



# Enhancing artificial bee colony algorithm using refraction principle

Peng Shao<sup>1</sup> · Le Yang<sup>1</sup> · Liang Tan<sup>1</sup> · Guangquan Li<sup>1</sup> · Hu Peng<sup>2</sup>

Published online: 17 March 2020  
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

## Abstract

The artificial bee colony algorithm (ABC), as one of the excellent intelligent optimization technologies, has presented very good optimization performance for many complex problems due to its simplicity and easiness of implementation. However, ABC has a very good performance at exploration relatively, but for some complex problems it still results in slower convergent speed and lower convergent accuracy in the later stage of algorithms. Meanwhile, ABC has relatively poor performance at exploitation. To overcome these drawbacks further, the enhancing ABC algorithm using refraction principle is proposed (EABC-RP) in this paper. In EABC-RP, on the one hand, in order to enhance its exploration further, the unified opposition-based learning (UOBL) based on refraction principle is employed to generate refraction solutions (new food sources) for employed bees, which helps to increase population diversity and guide search direction close to the global optimal solution. On the other hand, for exploitation, when ABC has fallen into the local optimal solution, the UOBL based on refraction principle is employed for mutation to increase the probability of jumping out of the local optimal solution for scout bees. A lot of experiments are conducted on 23 benchmark functions to verify the effectiveness of EABC-RP. The experimental results show that EABC-RP achieves higher solution accuracy and faster convergent speed in most cases and outperforms other ABC variants. In addition, EABC-RP is used to optimize finite impulse response (FIR) low-pass digital filter which obtains the better filtering performance, which validates the effectiveness of the EABC-RP algorithm further.

**Keywords** Evolutionary algorithms · Artificial bee colony algorithm · Opposition-based learning · Refraction principle

## 1 Introduction

In real world, especially in nowadays with information technology developing rapidly, many fields have faced or are facing a variety of complex optimization problems. However, with the increasing complexity of problems, traditional optimization methods have been difficult to deal with them. Fortunately, evolutionary algorithms (EAs) have very good optimization performance over these complex problems and play a more and more important role in the field of optimization. In order to solve various complex optimization problems, more and more EAs have been proposed such as

particle swarm optimization (PSO) (Kennedy and Eberhart 1995), differential evolution (DE) (Storn and Price 1997), genetic algorithm (GA) (Goldberg 1989; Holland 1992) and artificial bee colony (ABC) (Karaboga 2005).

In this paper, the research object is the ABC algorithm which was proposed by Karaboga in 2005 inspired by the foraging behavior of honey bee (Karaboga 2005). The ABC algorithm has risen wide spread attention from researchers due to its outstanding optimization performance over many complex optimization problems in various fields such as artificial neural networks (Yeh and Hsieh 2012; Bullinaria and Alyahya 2014), load balancing (Xing et al. 2018), scheduling problem (Zhang et al. 2013; Sundar et al. 2015; Hornig 2017; Pang and Shan 2019), power flow optimal problem (Adaryani and Karami 2013; Bai et al. 2017), optimal filter design (Bose et al. 2014), parameter estimation (Oliva et al. 2017; Chen et al. 2018) and sensor placement (Yang and Peng 2018) and so forth. Compared with traditional optimization methods, ABC has few requirements of objective function and constraints such as continuity and non-convexity. Moreover, it does not use external information in the searching process and only takes fitness function as the basis of evolution

---

Communicated by V. Loia.

✉ Peng Shao  
sp198310@163.com

<sup>1</sup> School of Computer and Information Engineering, Jiangxi Agricultural University, Nanchang 330045, People's Republic of China

<sup>2</sup> School of Information Science and Technology, Jiujiang University, Jiujiang 332005, People's Republic of China

(Karaboga and Basturk 2007). ABC has some advantages such as simple operation, fewer control parameters, high search accuracy and strong robustness. But it has a disadvantage of slower convergence, especially in the middle and last stages of the search process (Zhou et al. 2016b). In order to overcome these demerits, scholars have proposed some excellent improved ABC variants which mainly concentrated on balancing the exploration and exploitation.

In this paper, on the one hand, for the sake of enhancing the exploration of ABC further, the unified opposition-based learning (UOBL) based on refraction principle (Shao et al. 2015), an improved strategy of opposition-based learning (OBL) (Tizhoosh 2005), is employed in ABC to improve its performance. The main idea of UOBL is to improve the opposite process of OBL according to the refraction phenomena of light, which can increase the population diversity to enhance the probability of finding the global optimal solution. Based on the idea of the strategy, UOBL has been employed in the performance improvement in PSO (Shao et al. 2015), and in the improved PSO a large number of experimental results demonstrate that it has achieved very good optimization performance. On the other hand, the UOBL strategy is used again in ABC as a mutation strategy. When ABC algorithm traps into the local optimal domain, UOBL can generate the mutation solution based on refraction principle and guide the algorithm to search for the global optimal solution rather than random mutation resulting in aimless search. To verify the effectiveness of the adopted UOBL in ABC, 23 benchmark functions are tested in the next experiments, and extensive experimental results demonstrate that EABC-RP can achieve better optimization performance over the majority of them compared with other ABC variants.

The other parts of this paper can be organized as follows: In Sect. 2, related works including the basic idea of ABC and some other excellent ABC variants are introduced briefly. The main design idea of proposed ABC is described in detail in Sect. 3. Section 4 demonstrates the experimental results and analysis of the results. Finally, in Sect. 5 the conclusions are drawn.

## 2 Related works

### 2.1 Artificial bee colony algorithm

Similar to other swarm intelligent algorithms such as PSO, ABC is proposed inspired by collective intelligent behavior of biology in the nature and it simulates the collaborative foraging behavior of a swarm of honey bees. In the nature, a bee swarm is categorized into three groups in process of collaborative foraging, and they are the employed bees, the onlooker bees and the scout bees. Similarly, in ABC, the entire search process of the algorithm is also divided into three phases, the

employed bee phase, the onlooker bee phase and the scout bee phase, and each specified phase plays itself role in ABC. The employed bees play a role that searches food sources, and after all the employed bees finishing their search of food sources, they will share some information including the positions and the nectar amounts of food sources on the dance area to the onlooker bees. The onlooker bees make a decision to choose a better food source according to the information provided by the employed bees and then continue exploiting other food sources around the chosen food source. The food source that has more amount of nectar has a higher probability to be chosen by the onlooker bees compared with one of less nectar. If a food source is exhausted, the scout bee is translated from its associated employed bee, which abandons the food source and continues searching a new food source randomly (Zhou et al. 2016b; Bao and Zeng 2011).

In ABC, the position of each food source is generated randomly in the initial population and it is regarded as a candidate solution  $X_i$  for optimization problems which is expressed as  $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$  and  $D$  is the dimensional size of optimization problems. The main idea of ABC algorithm can be described as follows:

Firstly, the employed bees are in charge of searching food sources generated randomly. After all employed bees complete the search process, some information including the nectar amounts and the positions of food sources is shared by employed bees to the onlooker bees by the way of dancing in dancing area. Each employed bee generates a new food source  $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$  in the neighborhood of its parent position  $X_i$  by the following equation:

$$v_{ij} = x_{ij} + r_{ij} \cdot (x_{ij} - x_{kj}) \quad (1)$$

where  $k \in \{1, 2, \dots, SN\}$  and  $j \in \{1, 2, \dots, D\}$  are selected randomly, but  $k$  has to be different from  $i$  and  $k$  is a solution in the neighbor domain of  $i$ ;  $r_{ij} \in [-1, 1]$  is a random number which controls the generated range of the neighbor domain of  $x_{ij}$ ;  $SN$  is the number of generated food sources randomly and the number of employed bees corresponds to the number of food sources.

Then, if the fitness of the new food source  $V_i$  generated by Eq. (1) is better than the one of its parent  $X_i$ , and  $X_i$  is replaced with  $V_i$ . The onlooker bees continue searching other food sources according to Eq. (1) after receiving the related information including nectar amounts and positions of food sources from the employed bees. A food source is chosen by onlooker bees according to the probability related with amount of nectar, and the more amount of nectar food sources have, the greater the probability they are chosen. The probability  $p_i$  depends on the nectar amount (fitness value) of food sources, which is defined as follows:

$$p_i = \frac{\text{fitness}_i}{\sum_{j=1}^{SN} \text{fitness}_j} \quad (2)$$

where  $p_i$  is the selection probability and the parameter  $fitness_i$  is the fitness value of the solution  $X_i$  evaluated by its employed bee, and from Eq. (2) we can see that the number of the onlooker bees is also equal to the one of the food sources.

Lastly, if the quality of a food source is not improved further in predetermined generations, it can be considered that the algorithm has trapped into the local optimal domain. Therefore, it is taken into consideration to abandon the food source. The employed bee associated with the food source would be transformed into the scout bee to continue searching a new food source randomly. In ABC, the parameter *limit* is used to control the role changing of employed bees and scout bees. When the value of *limit* exceeds the predetermined generations, a new food source is generated randomly by scout bees to replace the abandoned food source according to the following equation.

$$x_{ij} = a_j + rand_j \cdot (b_j - a_j) \quad (3)$$

where  $[a_j, b_j]$  is the boundary constraint for the  $j$ th dimensional variable and the parameter  $rand_j$  is a distributed random number within the range  $[0, 1]$  uniformly.

## 2.2 Some excellent ABC variants

Since the original ABC is proposed, it has attracted many scholars attention and many excellent ABC variants have been proposed in the past decade. To overcome the poor exploitation ability further and enhance the optimization performance of ABC algorithm, some good strategies have been introduced to improve its optimization performance. For example, Zhou et al. (2016b) proposed an improved ABC based on Gaussian bare-bones (GBABC). Zheng et al. (2010) proposed a Gbest-guided artificial bee colony algorithm (GABC) for numerical function optimization. The GABC improved the exploitation by incorporating the information of global best solution into the solution search equation. Huo et al. (2015) proposed a discrete Gbest-guided ABC (DGABC) to simulate the search for the optimal service composition solution. Cui et al. (2017) proposed a modified Gbest-guided ABC algorithm with new probability model (MPGABC). In MPGABC, the novel search strategy was proposed by utilizing the useful information from the current best solution and the new probability selected model was proposed. The OBL strategy was applied into ABC algorithm by some scholars. For instance, El-Abd (2011) proposed an opposition-based ABC and after 1 year El-Abd (2012) used a generalized opposition-based learning strategy in initialization population and generation jumping to improve ABC (GOABC). Guo et al. (2015) proposed an elite opposition-based artificial bee colony algorithm for global optimization (EOABC). In EOABC, it executed the elite opposition-based

learning strategy with a preset learning probability to enhance its exploitation capacity. Moreover, Wang et al. (2014) combined several strategies into ABC algorithm to propose multi-strategy ensemble artificial bee colony algorithm (MEABC). Zhou et al. (2016a) proposed an improved multi-strategy ensemble artificial bee colony algorithm with neighborhood search (MEABC). Zhou et al. (2015) used the neighbor search strategy to propose an enhancing the modified artificial bee colony algorithm (MABC-NS). Fan et al. (2018) proposed a hybrid artificial bee colony algorithm with variable neighborhood search and memory mechanism (HABC). Zhong et al. (2017) adopted modified neighborhood based on update operator and independent-inheriting-search strategy to propose and improve artificial bee colony algorithm for global optimization (MNIIABC). In the MNIIABC algorithm, a modified neighborhood-based update operator was applied in the employed bee stage and an independent-inheriting-search strategy was used in the onlooker stage.

In this paper, what the search strategy mainly adopts is the UOBL model based on refraction principle to enhance the optimization performance of ABC algorithm. Note that some prominent ABC variants are listed briefly and other excellent ABC variants are not listed in this paper.

## 3 The proposed algorithm design

### 3.1 OBL based on refraction principle

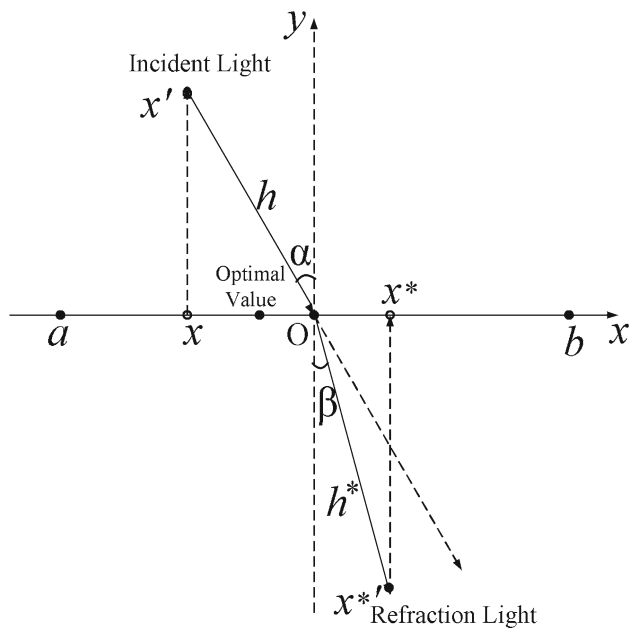
#### 3.1.1 Opposition-based learning

Opposition-based learning (Tizhoosh 2005) (OBL), proposed by Tizhoosh, is an excellent scheme for machine intelligence, which has been applied to various optimization algorithms and proved to be effective. The main idea of OBL is that the candidate solution for a given problem is computed to obtain its opposite solution according to Eq. (4), which increases the probability of finding a candidate solution closer to the global optimum solution. Tizhoosh gives the following definition for OBL.

Let  $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$  be as a candidate solution in  $D$ -dimensional space and the value of  $X_i$  be a real number defined in interval  $[a_j, b_j]$ . The opposite solution  $X_i^*$  of  $X_i$  can be defined as follows:

$$x_{ij}^* = a_j + b_j - x_{ij}, \quad j = 1, 2, 3, \dots, D \quad (4)$$

The OBL strategy has been applied to improve the global optimization of intelligent algorithms such as PSO and ABC, and it has obtained very good optimization performance. However, in the later stage of algorithms, due to the narrowing of the search interval, the opposition solution still falls into the local extremum region with a larger probability,



**Fig. 1** Opposite learning process based on refraction principle (UOBL)

which results in trapping into the local optimal solution easily (Wang et al. 2007). Some scholars proposed some improved OBL strategies to enhance its optimization performance. For example, Wang et al. (2011a) proposed a generalized opposition-based learning strategy to optimize PSO algorithm. Shao et al. (2015) proposed an improved OBL strategy based on refraction principle to optimize PSO algorithm.

### 3.1.2 The unified model of opposition-based learning based on refraction principle

As mentioned above, OBL is easy to trap into the local optimum solution in the later stage of algorithms so that some improved OBL strategies are proposed such as UOBL using the refraction principle (Shao et al. 2015) to improve its opposite process.

The refraction of light is a common physical phenomenon in the nature. When the light slants from a medium into another medium, its propagation direction will be changed, that is, a part of light is refracted. According to the refraction phenomenon of light, in our previous work (Shao et al. 2015), an improved OBL (UOBL) is proposed to ameliorate its performance, and the improvement process of UOBL is described in Fig. 1.

In Fig. 1, the parameter  $x^*$  represents the refraction solution of the candidate solution  $x$ , and the incident angle and the refraction angle are  $\alpha$  and  $\beta$ , respectively. According to the refraction law of light and Fig. 1, it can be concluded that the refractive index  $n$  is expressed by the following equation:

$$n = \sin \alpha / \sin \beta \quad (5)$$

Let  $k = h/h^*$ , and it can be seen from Fig. 1 that we can conclude the computational formula of refraction solution  $X_i^* = (x_{i1}, x_{i2}, \dots, x_{iD})$  is presented as follows:

$$x_{ij}^* = (a_j + b_j)/2 + (a_j + b_j)/(2kn) - x_{ij}/kn \quad (6)$$

where  $a_j$  and  $b_j$  are the minimum and maximum values of the  $j$ th dimension, respectively, in current population and  $x_{ij}$  represents the value of the  $j$ th dimension of the  $i$ th particle in current population and  $x_{ij}^*$  is the refraction solution of  $x_{ij}$ ;  $h$  and  $h^*$  are the length of the incident light and refraction light, respectively. When  $n = 1$  and  $k = 1$ , Eq. (6) can be simplified to the OBL model (Eq. (4)). Hence, the improved OBL model based on refraction principle can be regarded as the unified model of opposition-based learning (UOBL). Due to very good optimization performance of UOBL model, it is employed to enhance the global optimization performance of PSO in our previous work (Shao et al. 2015). Meanwhile, the enhanced PSO based on UOBL model is applied to optimize FIR digital filters, which achieves very good filtering performance.

### 3.2 Modified employed bee phase

The original ABC has excellent global optimization performance for many complex problems. That is to say, ABC has good exploration for optimization problems. The OBL is also a global optimization strategy, which has been applied to improve the global optimization of evolutionary algorithms such as PSO and ABC and achieves very good optimization performance. In order to enhance the performance of OBL further, the UOBL model is used to enhance the global searching ability of ABC algorithms. In this paper, UOBL is adopted to modify the search process of employed bees in ABC.

In ABC, the employed bees are responsible for searching optimal solutions in an initiate population generated randomly. The diversity of population has a great influence on the evolution of species. Similarly, in swarm intelligence algorithm, the diversity of population has a great influence on finding the optimal solution. For the sake of enhancing the population diversity in ABC, the UOBL model [as shown in Eq. (6)] which has a better optimization performance over PSO (Shao et al. 2015) is introduced to the employed bees to improve the diversity of the population and then enhance the global search of ABC algorithm.

The main idea of modified employed bee phase is that firstly each employed bee generates a new food source (a candidate solution)  $V_i$  according to Eq. (1) and its fitness value is calculated. And then the refraction solution  $X_i^*$  of the candidate solution  $V_i$  is obtained according to Eq. (6) and its fitness value is also computed. If the fitness value of the refraction solution is better than the one of the candidate solu-

**Algorithm 1** : Modified employed bees

- Step 1: for  $i=1$  to  $SN$
- Step 2: The new candidate solution  $V_i$  is achieved according to Eq. (1);
- Step 3: Calculate the fitness  $f(V_i)$  of the candidate solution;
- Step 4: The refraction solution  $X_i^*$  of the candidate solution is calculated according to Eq. (6) ;
- Step 5: Calculate the fitness  $f(X_i^*)$  of the refraction solution;
- Step 6: if  $f(X_i^*) < f(V_i)$  then
- Step 7:  $V_i$  is replaced with  $X_i^*$ ;
- Step 8: end if
- Step 9: end for

tions, the candidate solution is replaced with the refraction solution. The modified process of the employed bee phase is described in algorithm 1.

**3.3 Probability selection for the onlooker bee phase**

As mentioned above, after all employed bees finished their search of food sources according to Eq. (1), the onlooker bees will receive nectar amounts and positions of food sources that all employed bees provide. And the food source which has more nectar amounts is chosen with higher probability by the onlooker bees. In ABC, the probability of selecting food source is calculated according to Eq. (2). From the above equation, we can see that the probability is proportional to the fitness value, that is, the food source with higher fitness value is bound to be selected, which adopts a greedy selection method to choose a better food source actually, and its disadvantage is that the algorithm is easy to fall into the local optimum solution with the very big probability.

For the above problem, in EABC-RP, Eq. (2) is modified modestly and a linear equation as Eq. (7) is adopted to reduce the probability of falling into local optimum region.

$$p_i = \max Value * \frac{Fitness_i}{\max(Fitness)} + \min Value \quad (7)$$

where the parameters *maxValue* and *minValue* are numbers between [0, 1] and the parameter *maxValue* is greater than the parameter *minValue*. The *max(Fitness)* represents the maximal fitness value of all bees.

**3.4 Mutation for the scout bee phase**

As previously mentioned, if food sources are abandoned, the scout bees are transformed from a few employed bees and then generate new food sources randomly according to Eq. (3). However, the way of randomly generating new ones still has its disadvantage such as easy to get caught in the local optimal region when the search space becomes smaller in the later stage of the algorithm. Furthermore, if food sources are directly abandoned, the more help-

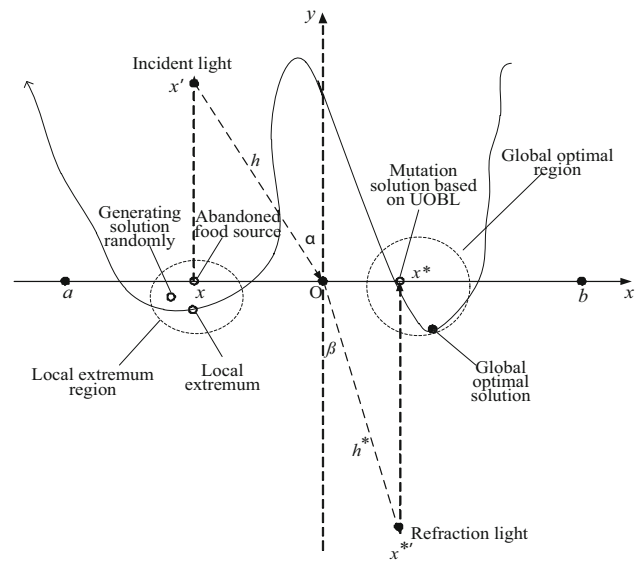


Fig. 2 Diagram of mutation for scout bees

ful search information obtained by the abandoned food source during the process of evolution would be lost, which makes it more difficult to find the optimal solution.

To overcome these drawbacks and enhance the performance of ABC, Eq. (6) is also employed in scout bee phase as a mutation operation to generate refraction solutions of the abandoned food sources. On the one hand, as mentioned above, in a narrow search space, the random mutation method is extremely vulnerable to falling into the local extremum region. But the method of generating the refraction solution can enlarge the search space and increase the probability of jumping out the local extremum region. On the other hand, by computing the refraction solution of the abandoned food source, the more helpful information that the abandoned food source contains can be preserved to guide the search direction close to the optimal solution. The mutation diagram of scout bees is described in Fig. 2.

In summary, based on the above basic idea, the pseudocode of EABC-RP is described in detail in algorithm 2.

**Algorithm 2:** Pseudo-code of EABC-RP algorithm

---

```

Step 1: Generate  $SN$  food sources  $X_i$  ( $i=1,2, \dots, SN$ ) as candidate solutions randomly and other related
parameters such as number of iteration  $FES$ , maximum number of iterations  $MaxFES$  and dimensional
size  $D$  and so on;
Step 2: Calculate the fitness values  $f(X_i)$  in accordance with the candidate solutions in the random population;
Step 3: while  $FES \leq MaxFES$  do
Step 4:   for  $i=1:SN$  do /*The employed bees*/
Step 5:     Choose  $j$  from 1 to  $D$  randomly and generate a random number  $r_{ij}$ ;
Step 6:     Choose  $k$  from 1 to  $SN$  randomly and choose a solution  $x_{kj}$  from the current population;
Step 7:     Generate a new candidate solution  $V_i$  according to Eq. (1) and calculate its fitness value  $f(V_i)$ ;
Step 8:     if  $f(V_i) < f(X_i)$  then
Step 9:        $X_i$  is replaced with  $V_i$ ;
Step 10:      Call Eq. (6) to generate a refraction solution  $X_i^*$ ;
Step 11:      Calculate the fitness value  $f(X_i^*)$  of the refraction solution;
Step 12:      if  $f(X_i^*) < f(V_i)$  then
Step 13:         $V_i$  is replaced with  $X_i^*$ ;
Step 14:      end if
Step 15:       $trial(i)=0$ ; //Set the iteration times
Step 16:    else
Step 17:       $trial(i) = trial(i)+1$ ;
Step 18:    end if
Step 19:     $FES=FES+1$ ;
Step 20:  end for
Step 21: /*Onlooker bees*/
Step 22: Calculate the choose probability  $p_i$  according to Eq. (7);
Step 23: while  $t=0 < SN$  do
Step 24:   if  $rand < p_i$  then //The parameter  $rand$  represents a random number
Step 25:      $t = t+1$ ;
Step 26:     Choose a food source  $X_j$  according to Eq. (7) which must be different from  $X_i$  ( $j \neq i$ );
Step 27:     Generate a new candidate solution  $V_i$  according to Eq. (1) and calculate its fitness value  $f(V_i)$ ;
Step 28:     if  $f(V_i) < f(X_j)$  then
Step 29:        $X_j$  is replaced with  $V_i$ ;
Step 30:        $trial(i)=0$ ; //Set the iteration times
Step 31:     else
Step 32:        $trial(i)+=1$ ;
Step 33:     end if
Step 34:   end if
Step 35:    $i = i+1$ ;
Step 36: end while
Step 37: /*Scout bees*/
Step 38: for  $i=1: SN$  do
Step 39:   if  $trial(i) > limit$  then
Step 40:      $trial(i)=0$ ; //The food source  $X_i$  without improvement in limited times is abandoned
Step 41:     Generate a mutation solution of the food source abandoned  $X_i$  according to Eq. (6) and
calculate its fitness value;
Step 42:   end if
Step 43: end for
Step 44: end while

```

---

## 4 Experimental verifications

### 4.1 Test problems and parameter setting

In order to verify the effectiveness of the proposed algorithm, in the next experiments, a set of 23 benchmark functions (F01–F23) are adopted, which are also tested widely in some excellent ABC variants mentioned above. The brief introduction of these benchmark functions is listed in Table 1, and more detailed description of them can be found in the following literatures (Xin et al. 1999; Xiong et al. 2014; Dervis and Bahriye 2009).

Note that, in Table 1, U, M, S and N, respectively, represent unimodal, multimodal, separable and non-separable. Furthermore, for the Rosenbrock function, it has the characteristic of multimodal functions when  $D$  is greater than 3.

For the sake of comparison fairness, the common parameters of all ABC variants are set as follows in the next experiments. The dimensional size  $D$  of benchmark functions is 30 and 50, respectively, and the corresponding maximum number of function evaluations ( $MaxFES$ ) is set to  $5000 \cdot D$ . Each benchmark function is tested for 30 times. In the proposed algorithm,  $maxValue$  and  $minValue$  are 0.9

**Table 1** 23 benchmark functions tested in the experiments

No.	Functions	Characteristics	Range	Global Optimum
F01	Sphere	US	[−100, 100]	0
F02	rosenbrock	UN	[−30, 30]	0
F03	ackley	MN	[−32, 32]	0
F04	griewank	MN	[−600, 600]	0
F05	rastrigin	MS	[−5.12, 5.12]	0
F06	schwefel	MS	[−500, 500]	−418.98 * D
F07	Quartic with noise	US	[−1.28, 1.28]	0
F08	Schaffer	MN	[−100, 100]	0
F09	Schewel	UN	[−10, 10]	0
F10	step	US	[−100, 100]	0
F11	SumDifferent	US	[−1, 1]	0
F12	SumSquares	US	[−10, 10]	0
F13	Zakharov	UN	[−5, 10]	0
F14	shiftedSphere	US	[−100, 100]	−450
F15	shiftedRosenbrock	MN	[−100, 100]	390
F16	shiftedRastrigin	MS	[−5, 5]	−330
F17	shiftedRotatedRastrigin	MN	[−5, 5]	−330
F18	Schwefel’s Problem 2.6 with Global Optimum on Bounds	UN	[−100, 100]	−310
F19	Shifted Rotated High Conditioned Elliptic Function	UN	[−100, 100]	−450
F20	Shifted Schwefel’s Problem 1.2	UN	[−100, 100]	−450
F21	Shifted Schwefel’s Problem 1.2 with Noise in Fitness	UN	[−100, 100]	−450
F22	Shifted Rotated Griewank’s Function without Bounds	MN	[0, 600]	−180
F23	Shifted Rotated Ackley’s Function with Global Optimum on Bounds	MN	[−32, 32]	−140

and 0.1, respectively, and the optimal combination of other parameters is set according to our previous work (Shao et al. 2015).

### 4.2 Comparison of EABC-RP with other excellent ABC algorithms

#### 4.2.1 Comparison of EABC-RP, ABC and GOABC-s

In this subsection, amounts of experiments are carried out at  $D = 30$  and  $D = 50$ , respectively, to verify the effectiveness of EABC-RP algorithm. In the next experiments, the parameters  $SN$  and  $limit$  are set to 30 and 100, respectively. A comprehensive comparison study is carried out between EABC-RP and other two ABC variants. The mean values and standard deviation (SD) are listed in Tables 2 and 3 at  $D = 30$  and  $D = 50$ , respectively. Furthermore, the Wilcoxon signed-rank test, a nonparametric statistical hypothesis test, is employed to compare the significant difference between two algorithms, which can be used as an alternative to the paired  $t$  test when the results do not obey the normal distribution (Garcia et al. 2010). “+,” “−” and “≈” indicate EABC-RP is better than, worse than and similar to that of other ABC variants, respectively, according to

the Wilcoxon signed-rank test when the significance level  $\alpha$  is 0.05. The experimental results are summarized as “ $w/l/t$ ,” which denotes that EABC-RP wins on  $w$  functions, loses on  $l$  functions and ties on  $t$  functions compared with other ABC variants, respectively.

- ABC: the original ABC;
- GOABC-s: the simplified GOABC. GOABC-s is an improved ABC based on generalized OBL (GOBL) for simplify. That is to say, GOABC-s removes other strategies and only retains the GOBL strategy compared with GOABC. Correspondingly, similar to EABC-RP, the GOBL strategy is also used to improve employed bees and scout bees, respectively, in GOABC-s, which is to compare the performance of GOBL and UOBL in improving ABC algorithm to prove the effectiveness of UOBL strategy.

Table 2 presents the comparison results of EABC-RP, ABC and GOABC-s at  $D = 30$ . As can be seen from the data presented in Table 2, the optimization performance of EABC-RP significantly outperforms the other two ABC algorithms for majority of benchmark functions. Specially, EABC-RP outperforms ABC over 16 out 23 benchmark functions while

**Table 2** Comparison of EABC-RP with ABC and GOABC-s at  $D = 30$ 

Functions	ABC Mean value $\pm$ SD	GOABC-s Mean value $\pm$ SD	EABC-RP Mean value $\pm$ SD
F01	4.71e-016 $\pm$ 5.01e-017 <sup>+</sup>	9.96e-021 $\pm$ 3.63e-020 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000
F02	3.23e-003 $\pm$ 2.49e-003 <sup>-</sup>	7.75e-001 $\pm$ 1.28e+000 <sup>+</sup>	1.52e-001 $\pm$ 8.63e-002
F03	4.16e-014 $\pm$ 2.38e-015 <sup>+</sup>	8.88e-016 $\pm$ 0.00e+000 <sup>~</sup>	8.88e-016 $\pm$ 0.00e+000
F04	2.36e-016 $\pm$ 1.73e-016 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000 <sup>~</sup>	0.00e+000 $\pm$ 0.00e+000
F05	9.47e-015 $\pm$ 2.11e-014 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000 <sup>~</sup>	0.00e+000 $\pm$ 0.00e+000
F06	-12569.5 $\pm$ 4.48e-012 <sup>~</sup>	-12569.5 $\pm$ 5.30e-012 <sup>~</sup>	-12569.5 $\pm$ 0.00e+000
F07	2.64e+000 $\pm$ 3.58e-001 <sup>+</sup>	3.86e-004 $\pm$ 3.33e-004 <sup>+</sup>	7.65e-006 $\pm$ 6.72e-006
F08	1.41e+001 $\pm$ 1.85e-003 <sup>+</sup>	1.41e+001 $\pm$ 5.86e-003 <sup>+</sup>	1.40e+001 $\pm$ 0.00e+000
F09	1.22e-015 $\pm$ 1.20e-016 <sup>+</sup>	5.63e-019 $\pm$ 8.06e-019 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000
F10	0.00e+000 $\pm$ 0.00e+000 <sup>~</sup>	0.00e+000 $\pm$ 0.00e+000 <sup>~</sup>	0.00e+000 $\pm$ 0.00e+000
F11	1.25e-016 $\pm$ 5.91e-017 <sup>+</sup>	1.78e-027 $\pm$ 7.15e-027 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000
F12	4.48e-016 $\pm$ 6.34e-017 <sup>+</sup>	2.20e-020 $\pm$ 6.71e-020 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000
F13	4.62e-016 $\pm$ 4.73e-017 <sup>+</sup>	4.67e-016 $\pm$ 5.70e-017 <sup>+</sup>	4.55e-016 $\pm$ 4.56e-017
F14	4.60e-016 $\pm$ 4.44e-017 <sup>~</sup>	4.14e-016 $\pm$ 6.36e-017 <sup>~</sup>	4.42e-016 $\pm$ 6.95e-017
F15	1.65e-001 $\pm$ 1.40e-001 <sup>+</sup>	3.86e-001 $\pm$ 3.59e-001 <sup>+</sup>	8.52e-002 $\pm$ 2.31e-001
F16	0.00e+000 $\pm$ 0.00e+000 <sup>~</sup>	5.32e-016 $\pm$ 1.30e-015 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000
F17	2.20e+002 $\pm$ 3.50e+001 <sup>-</sup>	3.50e+002 $\pm$ 3.08e+001 <sup>-</sup>	3.32e+002 $\pm$ 4.49e+001
F18	7.36e+003 $\pm$ 7.61e+002 <sup>+</sup>	7.17e+003 $\pm$ 8.76e+002 <sup>+</sup>	6.61e+003 $\pm$ 6.69e+002
F19	7.41e+006 $\pm$ 1.30e+006 <sup>+</sup>	9.33e+006 $\pm$ 1.87e+006 <sup>+</sup>	5.49e+005 $\pm$ 8.92e+005
F20	6.79e+002 $\pm$ 2.00e+002 <sup>+</sup>	6.02e+002 $\pm$ 1.37e+002 <sup>+</sup>	3.94e+002 $\pm$ 1.26e+002
F21	3.31e+004 $\pm$ 4.04e+003 <sup>-</sup>	2.86e+004 $\pm$ 3.99e+003 <sup>-</sup>	3.56e+004 $\pm$ 4.23e+003
F22	4.69e+003 $\pm$ 5.37e-011 <sup>+</sup>	8.74e+002 $\pm$ 1.70e+002 <sup>+</sup>	4.31e+002 $\pm$ 5.18e-010
F23	2.09e+001 $\pm$ 3.69e-002 <sup>+</sup>	2.02e+001 $\pm$ 3.81e-002 <sup>+</sup>	2.01e+001 $\pm$ 2.84e-002
<i>w/l/t</i>	16/3/4	15/2/6	-

ABC only obtains the better results on 3 out 23 benchmark functions (F02, F17 and F21). For F02, EABC-RP has no advantage compared with ABC when  $D$  is greater than 3, which is because its global optimal solution is located in a narrow parabolic-shaped flat valley which results in more difficulty to get better results, and the above three ABC variants also do not achieve better results for F02. However, for the most of other multimodal problems EABC-RP obtains better optimization performance than ABC or at least similar to ABC. For F17 and F21, EABC-RP slightly loses to ABC. For GOABC-s, EABC-RP also obtains significantly better performance than its over 15 out of 23 benchmark functions while GOABC-s merely slightly outperforms EABC-RP on 2 out of 23 benchmark functions (F17 and F21). Moreover, for F06, the above three ABC variants all obtain the optimal solution. But the difference among them is that the SD of EABC-RP is minimal, which indicates that the optimization performance of EABC-RP is the most stable than other two ABC algorithms.

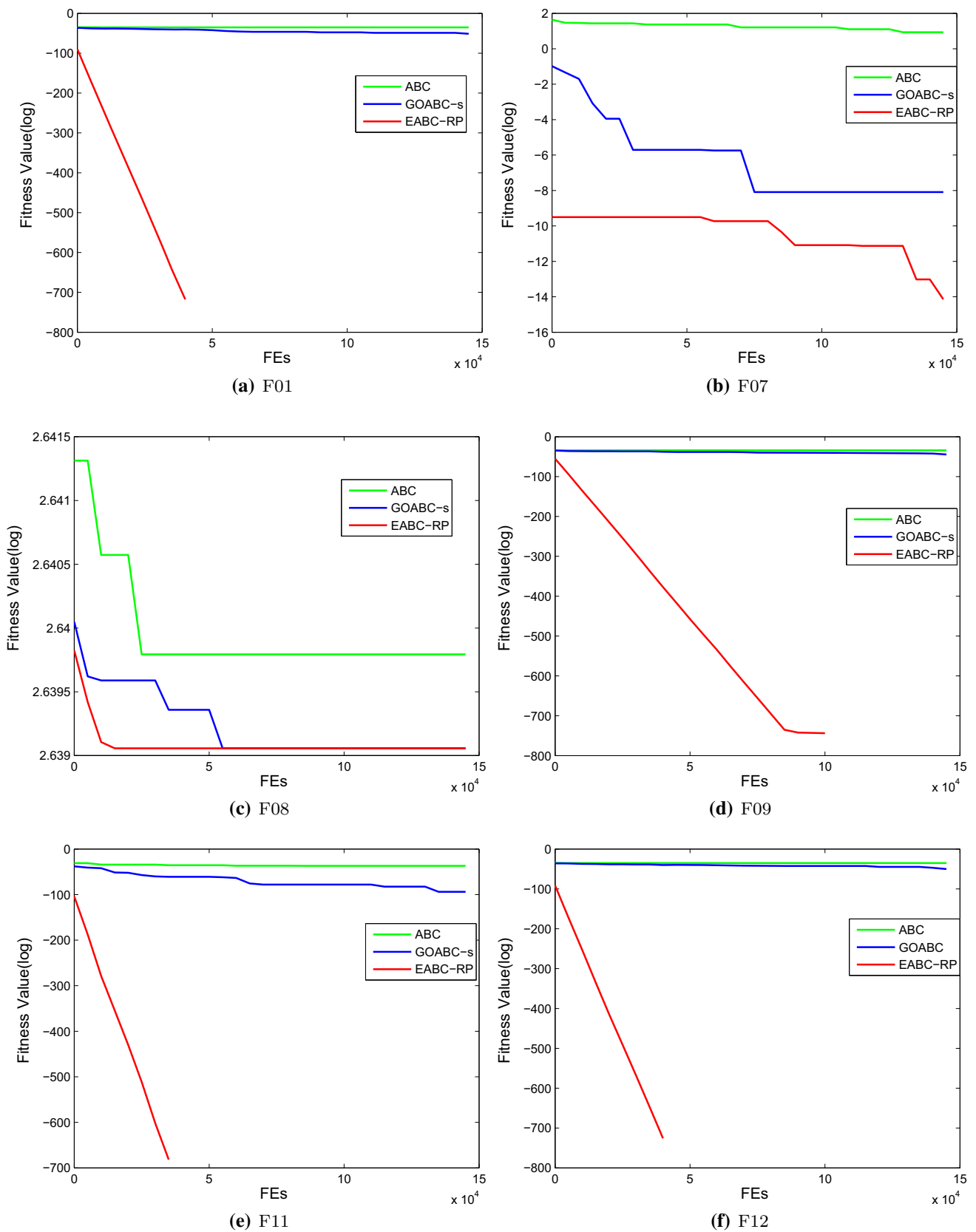
The convergent speed of EABC-RP, ABC and GOABC-s is presented in Fig. 3. As seen in Fig. 3, for the majority of benchmark functions the EABC-RP has faster convergent speed and lower convergent accuracy compared with other

two ABC variants. From this perspective, EABC-RP is better than the other two ABC variants.

Table 3 demonstrates the comparison results of EABC-RP, ABC and GOABC-s at  $D = 50$ . From Table 3, we can draw that the optimization performance of EABC-RP is significantly better than ABC and GOABC-s at  $D=50$  on the whole. Specifically, EABC-RP outperforms ABC over 17 out 23 benchmark functions and outperforms GOABC-s on 15 out 23 benchmark functions. These benchmark functions that outperform ABC and GOABC-s at  $D = 50$  are almost same as at  $D = 30$  besides the slight different results of few functions such as F16, which shows that the optimization performance of EABC-RP is not significantly affected along with the complexity increase of problems.

Moreover, to validate the effectiveness of EABC-RP further, the Friedman test, a statistical analysis method, is employed to compare the significant difference of three ABC variants at a statistical level  $\alpha = 0.05$ . When the parameter  $p$  value is smaller than  $\alpha = 0.05$ , it shows that there is the significant difference among algorithms. In Table 4, average rankings are presented for three algorithms at  $D = 30$  and  $D = 50$ , respectively, and the best value is boldface. As can be seen from it, whether  $D = 30$  or  $D = 50$ ,  $p$  value is lower





**Fig. 3** Convergent curves of ABC, GOABC-s and EABC-RP on some representative functions at  $D = 30$

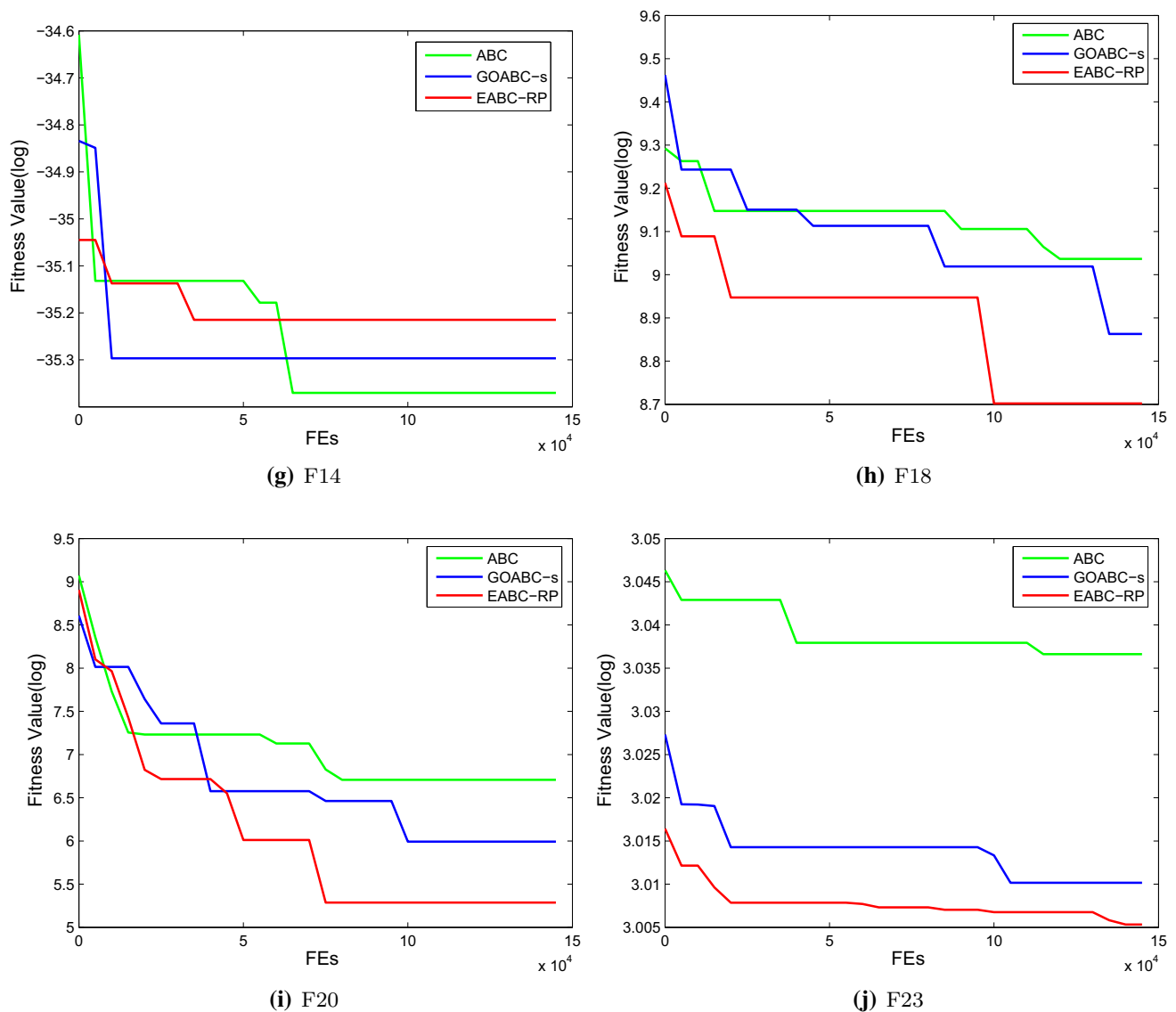


Fig. 3 continued

than 0.05, and the best average rankings are obtained by the proposed EABC-RP.

#### 4.2.2 Comparison of EABC-RP and other ABC variants

In this subsection, we present a comparison and analysis between EABC-RP and other excellent ABC variants at  $D = 30$  and  $D = 50$ . In the next experiments, the GBABC (Zhou et al. 2016b), an excellent improved ABC based on Gaussian bare-bones, is used to be compared with EABC-RP and ABC. The experimental results on 17 representative functions are listed in Table 5.

From Table 5, it can be seen that EABC-RP obtains better optimization performance compared with ABC on the majority of benchmark functions besides three benchmark

functions (F02, F17 and F21). For F02, EABC-RP outperforms GBABC, but they are defeated by ABC. For F17 and F21, EABC-RP has slight worse accuracy than those of other two ABC variants. Compared with GBABC, EABC-RP outperforms it on 8 benchmark functions and loses it on 5 benchmark functions while EABC-RP has similar results on 4 benchmark functions. From this view, the optimization performance of EABC-RP outperforms those of ABC and GBABC.

Table 6 presents experimental results of EABC-RP, ABC and GBABC at  $D = 50$ . As can be seen from it, EABC-RP outperforms the other two ABC variants. More specifically, EABC-RP outperforms ABC over 13 benchmark functions and outperforms GBABC on 10 benchmark functions.

**Table 3** Comparison of EABC-RP, ABC and GOABC-s at  $D = 50$

Functions	ABC Mean value $\pm$ SD	GOABC-s Mean value $\pm$ SD	EABC-RP Mean value $\pm$ SD
F01	1.13e-015 $\pm$ 1.23e-016 <sup>+</sup>	3.99e-020 $\pm$ 1.22e-019 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000
F02	2.79e-003 $\pm$ 3.14e-003 <sup>-</sup>	6.60e-001 $\pm$ 1.37e+000 <sup>+</sup>	4.23e-001 $\pm$ 2.06e-001
F03	9.29e-014 $\pm$ 7.30e-015 <sup>+</sup>	8.88e-016 $\pm$ 0.00e+000 <sup>~</sup>	8.88e-016 $\pm$ 0.00e+000
F04	7.69e-016 $\pm$ 2.71e-016 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000 <sup>~</sup>	0.00e+000 $\pm$ 0.00e+000
F05	1.40e-013 $\pm$ 6.99e-014 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000 <sup>~</sup>	0.00e+000 $\pm$ 0.00e+000
F06	-20949.1 $\pm$ 1.43e-007 <sup>~</sup>	-20949.1 $\pm$ 5.63e-007 <sup>~</sup>	-20949.1 $\pm$ 2.53e-008
F07	8.55e+000 $\pm$ 9.04e-001 <sup>+</sup>	2.55e-004 $\pm$ 2.73e-004 <sup>+</sup>	7.36e-006 $\pm$ 5.96e-006
F08	2.40e+001 $\pm$ 2.47e-003 <sup>~</sup>	2.40e+001 $\pm$ 6.75e-003 <sup>+</sup>	2.40e+001 $\pm$ 0.00e+000
F09	2.65e-015 $\pm$ 1.64e-016 <sup>+</sup>	1.80e-018 $\pm$ 1.51e-018 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000
F10	0.00e+000 $\pm$ 0.00e+000 <sup>~</sup>	0.00e+000 $\pm$ 0.00e+000 <sup>~</sup>	0.00e+000 $\pm$ 0.00e+000
F11	3.31e-014 $\pm$ 4.00e-014 <sup>+</sup>	3.26e-024 $\pm$ 1.73e-023 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000
F12	1.09e-015 $\pm$ 1.08e-016 <sup>+</sup>	7.10e-020 $\pm$ 1.62e-019 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000
F13	1.09e-015 $\pm$ 1.19e-016 <sup>+</sup>	1.08e-015 $\pm$ 1.04e-016 <sup>+</sup>	1.07e-015 $\pm$ 1.20e-016
F14	1.07e-015 $\pm$ 1.17e-016 <sup>+</sup>	1.00e-015 $\pm$ 1.18e-016 <sup>~</sup>	1.00e-015 $\pm$ 8.74e-017
F15	2.17e-001 $\pm$ 4.06e-001 <sup>+</sup>	4.28e-001 $\pm$ 2.72e-001 <sup>+</sup>	1.88e-001 $\pm$ 2.63e-001
F16	2.36e-015 $\pm$ 2.08e-015 <sup>-</sup>	2.35e-013 $\pm$ 5.04e-013 <sup>+</sup>	4.42e-015 $\pm$ 8.58e-015
F17	7.27e+002 $\pm$ 5.58e+001 <sup>-</sup>	7.61e+002 $\pm$ 3.62e+001 <sup>-</sup>	8.94e+002 $\pm$ 4.20e+001
F18	1.97e+004 $\pm$ 1.35e+003 <sup>+</sup>	1.96e+004 $\pm$ 1.45e+003 <sup>+</sup>	1.15e+003 $\pm$ 9.41e+002
F19	1.81e+007 $\pm$ 2.50e+006 <sup>+</sup>	2.04e+007 $\pm$ 2.61e+006 <sup>+</sup>	1.43e+006 $\pm$ 1.79e+006
F20	1.28e+004 $\pm$ 1.61e+003 <sup>+</sup>	9.97e+004 $\pm$ 1.37e+003 <sup>+</sup>	4.11e+003 $\pm$ 8.85e+002
F21	9.65e+004 $\pm$ 8.38e+003 <sup>+</sup>	8.87e+004 $\pm$ 9.16e+003 <sup>-</sup>	9.31e+004 $\pm$ 1.70e+004
F22	6.19e+003 $\pm$ 1.27e-012 <sup>+</sup>	1.97e+003 $\pm$ 3.84e+002 <sup>+</sup>	5.14e+002 $\pm$ 1.46e-012
F23	2.10e+001 $\pm$ 3.19e-002 <sup>+</sup>	2.03e+002 $\pm$ 4.03e-002 <sup>+</sup>	2.01e+001 $\pm$ 2.00e-002
w/l/t	17/3/3	15/2/6	-

**Table 4** Average rankings of ABC, GOABC-s and EABC-RP at  $D = 30$  and  $D = 50$

Algorithms	Average rankings			
	$D = 30$	$p$ value	$D = 50$	$p$ value
ABC	2.48		2.48	
GBABC	2.13	0.000	2.04	0.001
EABC-RP	<b>1.39</b>		<b>1.48</b>	

Moreover, the Friedman test is also conducted to compare the optimization performance of the above three ABC variants at  $D = 30$  and  $D = 50$ , respectively. The average rankings are presented in Table 7. As can be seen from it, whether  $D = 30$  or  $D = 50$ ,  $p$  value is lower than 0.05, which shows that there is the significant difference among algorithms. The best average rankings are obtained by EABC-RP and presented in boldface, which means that the sort of three ABC variants is EABC-RP, GBABC and ABC.

### 4.3 FIR low-pass digital filter optimization

#### 4.3.1 FIR digital filters

Digital filters, which can be used to filter out the noise of interfering signals, have been concerned over years and

achieved some intensive research works. Digital filters can be classified into two categories, namely infinite impulse response filters (IIR) and finite impulse response filters (FIR), according to the length of unit impulse response (Cheng 2009). In comparison, FIR digital filters have the characteristics of strict linear phase and arbitrary amplitude which makes the filters more stable. What's more, it is known to all that the communication channels that image processing and data transmission need must have the characteristic of linear phase. In other words, FIR digital filters have practical application more widely such as image processing, remote sensing, defense and medicine.

The transform function  $H(z)$  of FIR digital filters can be expressed using  $z$  transform by Eq. (8) (also called the systemic function) (Cheng 2009):

$$H(z) = \sum_{n=0}^{N-1} h(n)z^{-n}, \quad 0 \leq n \leq N - 1 \tag{8}$$

where  $N$  is the orders of the filter; the parameter  $h(n)$  is the unit impulse response, namely filter coefficients;  $(N+1)$  is the number of filter coefficients.

**Table 5** Comparison of EABC-RP, ABC and GBABC at  $D = 30$

Functions	ABC Mean value $\pm$ SD	GBABC Mean value $\pm$ SD	EABC-RP Mean value $\pm$ SD
F01	4.71e-016 $\pm$ 5.01e-017 <sup>+</sup>	2.63e-049 $\pm$ 5.27e-049 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000
F02	3.23e-003 $\pm$ 2.49e-003 <sup>-</sup>	3.63e+000 $\pm$ 3.40e+000 <sup>+</sup>	5.72e-001 $\pm$ 5.78e-001
F03	4.16e-014 $\pm$ 2.38e-015 <sup>+</sup>	4.44e-016 $\pm$ 0.00e+000 <sup>≈</sup>	8.88e-016 $\pm$ 0.00e+000
F04	2.36e-016 $\pm$ 1.73e-016 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000 <sup>≈</sup>	0.00e+000 $\pm$ 0.00e+000
F05	9.47e-015 $\pm$ 2.11e-014 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000 <sup>≈</sup>	0.00e+000 $\pm$ 0.00e+000
F07	2.64e+000 $\pm$ 3.58e-001 <sup>+</sup>	9.82e-005 $\pm$ 7.06e-005 <sup>+</sup>	7.65e-006 $\pm$ 6.72e-006
F10	0.00e+000 $\pm$ 0.00e+000 <sup>≈</sup>	0.00e+000 $\pm$ 0.00e+000 <sup>≈</sup>	0.00e+000 $\pm$ 0.00e+000
F14	4.60e-016 $\pm$ 4.44e-017 <sup>+</sup>	6.44e-014 $\pm$ 1.93e-014 <sup>+</sup>	4.42e-016 $\pm$ 6.95e-017
F15	1.65e-001 $\pm$ 1.40e-001 <sup>+</sup>	3.64e+001 $\pm$ 4.94e+001 <sup>+</sup>	8.52e-002 $\pm$ 2.31e-001
F16	0.00e+000 $\pm$ 0.00e+000 <sup>≈</sup>	6.63e-014 $\pm$ 2.12e-014 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000
F17	2.20e+002 $\pm$ 3.50e+001 <sup>-</sup>	1.28e+002 $\pm$ 1.77e+001 <sup>-</sup>	4.07e+002 $\pm$ 5.69e+001
F18	7.36e+003 $\pm$ 7.61e+002 <sup>+</sup>	2.43e+003 $\pm$ 4.72e+002 <sup>-</sup>	6.61e+003 $\pm$ 6.69e+002
F19	7.41e+006 $\pm$ 1.30e+006 <sup>+</sup>	2.80e+006 $\pm$ 1.02e+006 <sup>+</sup>	5.49e+005 $\pm$ 8.92e+005
F20	6.79e+002 $\pm$ 2.00e+002 <sup>+</sup>	2.94e+002 $\pm$ 1.11e+002 <sup>-</sup>	3.94e+002 $\pm$ 1.26e+002
F21	3.31e+004 $\pm$ 4.04e+003 <sup>-</sup>	7.38e+003 $\pm$ 1.46e+003 <sup>-</sup>	3.56e+004 $\pm$ 4.23e+003
F22	4.69e+003 $\pm$ 5.37e-011 <sup>+</sup>	1.86e-002 $\pm$ 1.63e-002 <sup>-</sup>	4.31e+002 $\pm$ 5.18e-010
F23	2.09e+001 $\pm$ 3.69e-002 <sup>+</sup>	2.09e+001 $\pm$ 8.62e-002 <sup>+</sup>	2.01e+001 $\pm$ 2.84e-002
w/l/t	12/3/2	8/5/4	-

**Table 6** Comparison of EABC-RP, ABC and GBABC at  $D = 50$

Functions	ABC Mean value $\pm$ SD	GBABC Mean value $\pm$ SD	EABC-RP Mean value $\pm$ SD
F01	1.13e-015 $\pm$ 1.23e-016 <sup>+</sup>	1.15e-040 $\pm$ 1.33e-040 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000
F02	2.79e-003 $\pm$ 3.14e-003 <sup>-</sup>	4.92e+001 $\pm$ 1.86e+001 <sup>+</sup>	4.23e-001 $\pm$ 2.06e-001
F03	9.29e-014 $\pm$ 7.30e-015 <sup>+</sup>	4.44e-016 $\pm$ 0.00e+000 <sup>≈</sup>	8.88e-016 $\pm$ 0.00e+000
F04	7.69e-016 $\pm$ 2.71e-016 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000 <sup>≈</sup>	0.00e+000 $\pm$ 0.00e+000
F05	1.40e-013 $\pm$ 6.99e-014 <sup>+</sup>	0.00e+000 $\pm$ 0.00e+000 <sup>≈</sup>	0.00e+000 $\pm$ 0.00e+000
F07	8.55e+000 $\pm$ 9.04e-001 <sup>+</sup>	9.08e-005 $\pm$ 7.19e-005 <sup>+</sup>	7.36e-006 $\pm$ 5.96e-006
F10	0.00e+000 $\pm$ 0.00e+000 <sup>≈</sup>	0.00e+000 $\pm$ 0.00e+000 <sup>≈</sup>	0.00e+000 $\pm$ 0.00e+000
F14	1.07e-015 $\pm$ 1.17e-016 <sup>≈</sup>	1.35e-013 $\pm$ 2.74e-014 <sup>+</sup>	1.00e-015 $\pm$ 8.74e-017
F15	2.17e-001 $\pm$ 4.06e-001 <sup>+</sup>	7.17e+001 $\pm$ 7.63e+001 <sup>+</sup>	1.88e-001 $\pm$ 2.63e-001
F16	2.36e-015 $\pm$ 2.08e-015 <sup>+</sup>	1.63e-013 $\pm$ 2.84e-014 <sup>+</sup>	4.42e-015 $\pm$ 8.58e-015
F17	7.27e+002 $\pm$ 5.58e+001 <sup>-</sup>	1.61e+002 $\pm$ 1.52e+001 <sup>-</sup>	8.94e+002 $\pm$ 4.20e+001
F18	1.97e+004 $\pm$ 1.35e+003 <sup>+</sup>	5.47e+003 $\pm$ 7.62e+002 <sup>+</sup>	1.15e+003 $\pm$ 9.41e+002
F19	1.81e+007 $\pm$ 2.50e+006 <sup>+</sup>	7.11e+006 $\pm$ 2.86e+006 <sup>+</sup>	1.43e+006 $\pm$ 1.79e+006
F20	1.28e+004 $\pm$ 1.61e+003 <sup>+</sup>	5.61e+003 $\pm$ 1.01e+003 <sup>+</sup>	4.11e+003 $\pm$ 8.85e+002
F21	9.65e+004 $\pm$ 8.38e+003 <sup>+</sup>	3.33e+004 $\pm$ 3.56e+003 <sup>-</sup>	9.31e+004 $\pm$ 1.70e+004
F22	6.19e+003 $\pm$ 1.27e-012 <sup>+</sup>	1.39e-002 $\pm$ 1.39e-002 <sup>-</sup>	5.14e+002 $\pm$ 1.46e-012
F23	2.10e+001 $\pm$ 3.19e-002 <sup>+</sup>	2.10e+001 $\pm$ 8.13e-002 <sup>+</sup>	2.01e+001 $\pm$ 2.00e-002
w/l/t	13/2/2	10/3/4	-

According to the characteristics of FIR digital filters such as the linear phase (Shao et al. 2015), Eq. (8) can be transformed as the following expression:

$$H(\omega) = \sum_{n=1}^{N/2} h\left(\frac{N}{2} - n\right) \bullet 2 \cos\left[\left(n - \frac{1}{2}\right)\omega\right] \tag{9}$$

where  $H(\omega)$  is the amplitude function.

**Table 7** Average rankings of ABC, GBABC and EABC-RP at  $D = 30$  and  $D = 50$

Algorithms	Average rankings			
	$D = 30$	$p$ value	$D = 50$	$p$ value
ABC	2.47		2.50	
GBABC	1.79	0.04	1.91	0.011
EABC-RP	<b>1.74</b>		<b>1.59</b>	

The types of digital filters, such as low pass and high pass, are determined by the coefficients  $h(n)$ . For evaluating the filtering performance of FIR digital filters, some generalized specifications such as the maximal pass band ripple (Cheng 2009; Wang et al. 2011b) have been taken into consideration.

What is the FIR low-pass digital filter? In signal filtering, the FIR low-pass digital filter allows the low-frequency components of the signal to pass through in the pass band and suppresses the high-frequency components in the stop band to remove the mixed noise in the signal. For FIR low-pass digital filter with linear phase, its ideal frequency response can be described as the following equation:

$$H(\omega) = \begin{cases} 1 & 0 \leq \omega \leq \omega_p \\ 0 & \omega_s \leq \omega \leq 1 \end{cases} \quad (10)$$

where  $\omega_p$  and  $\omega_s$  are the normalized cutoff frequency of the pass band and the stop band, respectively.

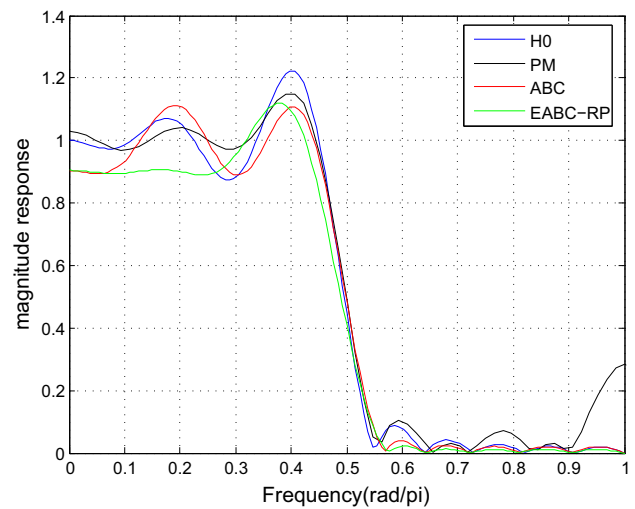
### 4.3.2 Fitness functions

The filtering performance of digital filters is related to their coefficients  $h(n)$  to a large extent, and to be exact, it is concerned with the combination of filter coefficients  $h(n)$  for FIR digital filter optimization. Hence, the process of obtaining the coefficient combination of digital filters can be regarded as the combinational optimization problem. In order to obtain a set of optimal combination of FIR digital filter coefficients, EABC-RP is employed to optimize FIR low-pass digital filter to verify its optimization performance and effectiveness for solving practical application.

It is of importance that designing or choosing a proper error function (fitness function) evaluates the performance of evolutionary algorithms. In this paper, Eq. (11) has been used as the fitness function because of achieving better digital filtering performance compared with other fitness functions which is adopted in Saha et al. (2013).

$$J = \sum abs[abs(|H(\omega)| - 1) - \delta_p] + \sum [abs(|H(\omega)|) - \delta_s] \quad (11)$$

where  $\delta_p$  and  $\delta_s$  are the ripples of the pass band and the stop band, respectively.



**Fig. 4** Magnitude response for FIR low-pass filters

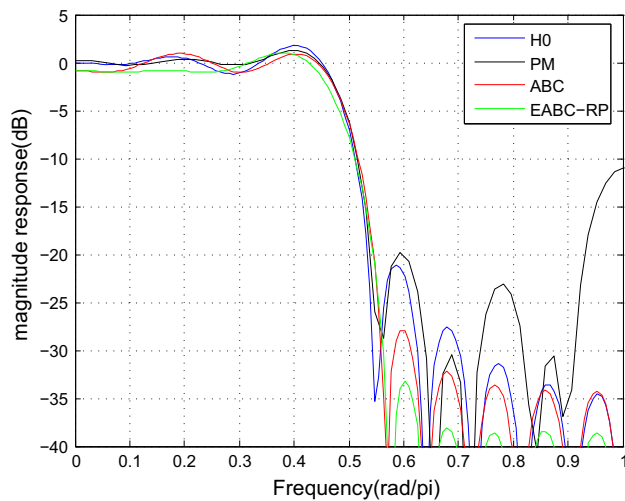
### 4.3.3 Results and analysis

In this subsection, the numerical experiments are presented for designing the FIR low-pass digital filter and each algorithm is performed in MATLAB for 500 iteration times. The parameters of FIR low-pass digital filters are set as follows: The order  $N$  is 20, and the number of coefficients is 21. The pass band ripple  $\delta_p$  and stop band ripple  $\delta_s$  are 0.1 and 0.01, respectively. The normalized pass band cutoff frequency  $\omega_p$  is 0.45, and normalized stop band cutoff frequency  $\omega_s$  is 0.55. The transition width ( $\omega_p - \omega_s$ ) is 0.1. In order to demonstrate the effectiveness of FIR low-pass digital filter designed by the EABC-RP algorithm, the zero-phase FIR digital filter as H0 (Ngamtawee and Wardkein 2013), the PM algorithm (Parks and McClellan 1972) and ABC are adopted for comparative algorithms to design the FIR low-pass digital filter, respectively.

Figures 4 and 5 demonstrate the magnitude responses and the magnitude attenuations of FIR low-pass digital filters which are designed by different comparative algorithms, respectively. In Fig. 4, from the magnitude responses we can see that the stop band ripple of FIR low-pass digital filter designed by EABC-RP is closest to 0, which shows that most high-frequency components are suppressed. From Fig. 5 as can be seen, the FIR low-pass digital filter designed by EABC-RP achieves the minimal magnitude attenuation compared with the ones optimized by H0, PM and ABC, respectively.

The maximal pass band ripples are listed in Table 8. From Table 8, we can see that the FIR low-pass digital filter designed by EABC-RP obtains the maximal pass band ripple compared with the ABC algorithm.

The coefficients of the FIR low-pass digital filters are listed in Table 9.



**Fig. 5** Magnitude attenuation for FIR low-pass filters

**Table 8** Maximal pass band ripples

Functions	Mean value	SD	Maximum
ABC	9.29e-001	3.46e-002	9.98e-001
EABC-RP	1.00e+000	1.90e-002	1.30e+000

**Table 9** Coefficients of FIR low-pass digital filter

Coefficients	ABC	EABC-RP
$h(1) = h(21)$	0.451645	0.452411
$h(2) = h(20)$	0.480228	0.448904
$h(3) = h(19)$	0.534912	0.446048
$h(4) = h(18)$	0.469924	0.526375
$h(5) = h(17)$	0.471997	0.418659
$h(6) = h(16)$	0.037409	0.037646
$h(7) = h(15)$	-0.000594	-0.001333
$h(8) = h(14)$	-0.002535	-0.001797
$h(9) = h(13)$	0.002448	0.001988
$h(10) = h(12)$	-0.001403	-0.000997
$h(11)$	0.550651	0.452259

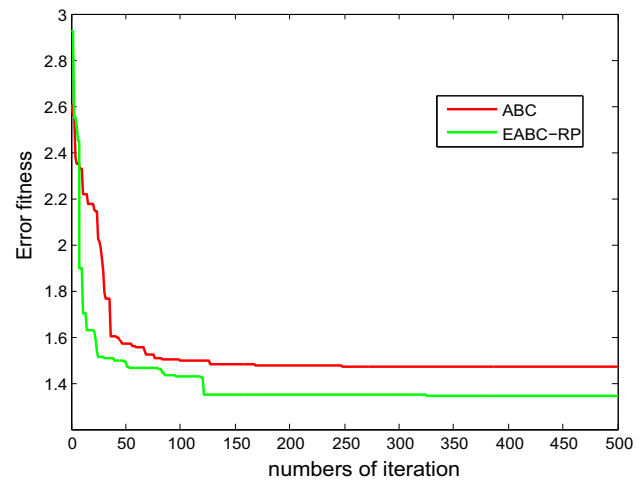
**Table 10** Comparison results of ABC and EABC-RP

Functions	Mean values	SD	Optimal values
ABC	1.31e+000	9.22e-002	1.28e+000
EABC-RP	1.28e+000	8.61e-002	1.26e+000

The comparison results of the mean values, SD and optimal values of two ABC variants are listed in Table 10.

Moreover, Fig. 6 presents the convergent speed of EABC-RP and ABC.

From Fig. 6, we can conclude that the EABC-RP has faster convergent speed and higher convergent accuracy compared



**Fig. 6** Convergent comparison of two ABC variants

with the ABC when they optimize the FIR low-pass digital filter, respectively.

To sum up, according to the above experimental results and analysis, it is evidently shown that the FIR low-pass digital filter designed by EABC-RP has obtained better filtering performance compared with the ones designed by the other algorithms, which also proves that the effectiveness of the EABC-RP algorithm.

## 5 Conclusions

In order to enhance the global search capability and local search capability further to improve the optimization performance of ABC, an improved OBL (UOBL) based on refraction principle is used to improve the search process of employed bees and mutation process of scout bees. On the basis of these, an enhancing ABC based on refraction principle (EABC-RP) is proposed. In the aspect of the global search of EABC-RP, the UOBL model is adopted to increase the population diversity by calculating the refraction solutions of the current candidate solutions to increase probability of finding the optimal solution. In the aspect of the local search of EABC-RP, to prevent trapping into the local optimal solution resulted from the mutation randomly and preserve the search experiences of the scout bees in ABC, UOBL plays an important role over leading the candidate solution to mutate toward the optimal solution to skip the local optimal solution with greater probability. The large number of experiment results and analysis shows that EABC-RP has a better optimization performance compared with other ABC variants. Meanwhile, the proposed EABC-RP is used to optimize the FIR low-pass digital filters and the better filtering performance is obtained by the designed FIR low-pass digital filters using EABC-RP,

which proves the effectiveness of the proposed algorithm further.

**Acknowledgements** This work is supported by the National Natural Science Foundation of China (Nos. 61763019 and 61862032) and the Science and Technology Plan Projects of Jiangxi Provincial Education Department (No. GJJ160409).

## Compliance with ethical standards

**Conflict of interest** The authors declare that they have no conflict of interest.

## References

- Adaryani MR, Karami A (2013) Artificial bee colony algorithm for solving multi-objective optimal power flow problem. *Int J Electr Power Energy Syst* 53(1):219–230
- Bai W, Eke I, Lee KY (2017) An improved artificial bee colony optimization algorithm based on orthogonal learning for optimal power flow problem. *Control Eng Pract* 61:163–172
- Bao L, Zeng JC (2011) A bi-group differential artificial bee colony algorithm. *Control Theory Appl* 28(2):266–272
- Bose D, Biswas S, Vasilakos AV, Laha S (2014) Optimal filter design using an improved artificial bee colony algorithm. *Inf Sci* 281(281):443–461
- Bullinaria JA, Alyahya K (2014) Artificial bee colony training of neural networks: comparison with back-propagation. *Memet Comput* 6(3):171–182
- Chen X, Xu B, Mei C, Ding Y, Li K (2018) Teaching-learning-based artificial bee colony for solar photovoltaic parameter estimation. *Appl Energy* 212:1578–1588
- Cheng P (2009) *Digital signal processing*. Tsinghua University Press, Beijing
- Cui L, Zhang K, Li GH, Fu XH, Wen ZK, Lu N, Lu J (2017) Modified gbest-guided artificial bee colony algorithm with new probability model. *Soft Comput* 22(2):1–27
- Dervis K, Bahriye A (2009) A comparative study of artificial bee colony algorithm. *Appl Math Comput* 214(1):108–132
- El-Abd M (2011) Opposition-based artificial bee colony algorithm. In: *Genetic & evolutionary computation conference*, pp 109–119
- El-Abd M (2012) Generalized opposition-based artificial bee colony algorithm. In: *Evolutionary computation*, pp 1–4
- Fan C, Qiang FU, Long G, Xing Q (2018) Hybrid artificial bee colony algorithm with variable neighborhood search and memory mechanism. *J Syst Eng Electron* 29(2):405–414
- Garcia S, Fernandez A, Luengo J, Herrera F (2010) Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: experimental analysis of power. *Inf Sci* 180(10):2044–2064
- Goldberg D (1989) *Genetic algorithm in search, optimization, and machine learning*. Addison-Wesley, Reading, MA
- Guo ZL, Wang S, Yue XZ, Jiang D, Li K (2015) Elite opposition-based artificial bee colony algorithm for global optimization. *Int J Eng (IJE)* 28(9):1268–1275
- Holland JH (1992) *Adaptation in natural and artificial system*. MIT Press, Cambridge
- Hornig SC (2017) Combining artificial bee colony with ordinal optimization for stochastic economic lot scheduling problem. *IEEE Trans Syst Man Cybern Syst* 45(3):373–384
- Huo Y, Zhuang Y, Gu JJ, Ni S, Xue Y (2015) Discrete gbest-guided artificial bee colony algorithm for cloud service composition. *Appl Intell* 42(4):661–678
- Karaboga D (2005) An idea based on honey bee swarm for numerical optimization. Technical report TR06, Erciyes University
- Karaboga D, Basturk B (2007) A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *J Glob Optim* 39(3):459–471
- Kennedy J, Eberhart R (1995) Particle swarm optimization. In: *Proceedings of IEEE international conference on neural networks*, vol 4, pp 1942–1948
- Ngamtawee R, Wardkein P (2013) Linear-phase fir design using PSO method with zero-phase pre-design. In: *International conference on electrical engineering/electronics*, pp 1–5
- Oliva D, Ewees AA, Aziz MAE, Hassanien AE, Cisneros MP (2017) A chaotic improved artificial bee colony for parameter estimation of photovoltaic cells. *Energies* 10(7):865
- Pang C, Shan G (2019) Sensor scheduling based on risk for target tracking. *IEEE Sens J* 19(18):8224–8232
- Parks T, McClellan J (1972) Chebyshev approximation for nonrecursive digital filters with linear phase. *IEEE Trans Circuit Theory* 19(2):189–194
- Saha SK, Ghoshal SP, Mandal D, Kar R (2013) Cat swarm optimization algorithm for optimal linear phase fir filter design. *ISA Trans* 52(6):781–794
- Shao P, Wu ZJ, Zhou XY, Deng CS (2015) Improved particle swarm optimization algorithm based on opposite learning of refraction. *Acta Electron Sin* 25(18):4117–4125
- Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J Glob Optim* 11(4):341–359
- Sundar S, Suganthan PN, Jin CT, Cai TX, Chong CS (2015) A hybrid artificial bee colony algorithm for the job-shop scheduling problem with no-wait constraint. *Soft Comput* 7(3):1–10
- Tizhoosh HR (2005) Opposition-based learning: a new scheme for machine intelligence. In: *International conference on computational intelligence for modelling, control & automation, & international conference on intelligent agents, web technologies & internet commerce*, pp 695–701
- Wang H, Li H, Liu Y, Li C, Zeng S (2007) Opposition-based particle swarm algorithm with Cauchy mutation. In: *IEEE congress on evolutionary computation*, pp 4750–4756
- Wang H, Wu ZJ, Rahnamayan S, Liu Y, Ventresca M (2011a) Enhancing particle swarm optimization using generalized opposition-based learning. *Inf Sci* 181(20):4699–4714
- Wang YL, Wang SH, Ji RD (2011b) An extreme simple method for digital FIR filter design. In: *2011 third international conference on measuring technology and mechatronics automation*
- Wang H, Wu ZJ, Rahnamayan S, Sun H, Liu Y, Pan JS (2014) Multi-strategy ensemble artificial bee colony algorithm. *Inf Sci* 279:587–603
- Xin Y, Liu Y, Lin G (1999) Evolutionary programming made faster. *IEEE Trans Evol Comput* 3(2):82–102
- Xing H, Song F, Yan L, Wei P (2019) A modified artificial bee colony algorithm for load balancing in network-coding-based multicast. *Soft Comput* 23:6287–6305
- Xiong GJ, Shi DY, Duan XZ (2014) Enhancing the performance of biogeography-based optimization using polyphyletic migration operator and orthogonal learning. *Comput Oper Res* 41(1):125–139
- Yang JH, Peng ZR (2018) Improved ABC algorithm optimizing the bridge sensor placement. *Sensors* 18(7):2240
- Yeh W, Hsieh TJ (2012) Artificial bee colony algorithm-neural networks for S-system models of biochemical networks approximation. *Neural Comput Appl* 21(2):365–375

- Zhang R, Song S, Wu C (2013) A hybrid artificial bee colony algorithm for the job shop scheduling problem. *Int J Prod Econ* 141(1):167–178
- Zheng F, Gong Z, Li Q, Wan D, Zheng X, Wang T, Wang G (2010) Gbest-guided artificial bee colony algorithm for numerical function optimization. *Appl Math Comput* 217(7):3166–3173
- Zhong F, Li H, Zhong S (2017) An improved artificial bee colony algorithm with modified-neighborhood-based update operator and independent-inheriting-search strategy for global optimization. *Eng Appl Artif Intell* 58:134–156
- Zhou X, Wang H, Wang MW, Wan JY (2015) Enhancing the modified artificial bee colony algorithm with neighborhood search. *Soft Comput* 21(10):1–11
- Zhou X, Wang MW, Wan JY, Zuo JL (2016a) An improved multi-strategy ensemble artificial bee colony algorithm with neighborhood search. In: *International conference on neural information processing*
- Zhou X, Wu ZJ, Wang H, Rahnamayan S (2016b) Gaussian bare-bones artificial bee colony algorithm. *Soft Comput* 20(3):907–924

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