Soft Comput (2018) 22:2217–2243
https://doi.org/10.1007/s00500-017-2485-y

METHODOLOGIES AND APPLICATION

Modified Gbest-guided artificial bee colony algorithm with new probability model

Laizhong Cui[1] · Kai Zhang[1] · Genghui Li[1] · Xianghua Fu[1] · Zhenkun Wen[1] · Nan Lu[1] · Jian Lu[2]

Published online: 17 January 2017

Abstract Artificial bee colony (ABC) is a very effective and efficient swarm-based intelligence optimization algorithm, which simulates the collective foraging behavior of the honey bees. However, ABC has strong exploration ability but poor exploitation ability because its solution search equation performs well in exploration but badly in exploitation. In order to enhance the exploitation ability and obtain a better balance between exploitation and exploration, in this paper, a novel search strategy which exploits the valuable information of the current best solution and a novel probability model which makes full use of the other good solutions on onlooker bee phase are proposed. To be specific, in the novel search strategy, a parameter P is used to control which search equation to be used, the original search equation of ABC or the new proposed search equation. The new proposed search equation utilizes the useful information from the current best solution. In the novel probability model, the selected probability of the good solution is absolutely significantly larger than that of the bad solution, which makes sure the good solutions can attract more onlooker bees to search. We put forward a new ABC variant, named MPGABC by combining the novel search strategy and probability model with the basic framework of ABC. Through the comparison of MPGABC and some other state-of-the-art ABC variants on 22 benchmark functions, 22 CEC2011 real-world optimization problems and 28 CEC2013 real-parameter optimization problems, the experimental results show that MPGABC is better than or at least comparable to the competitors on most of benchmark functions and real-world problems.

Keywords Artificial bee colony algorithm · Search strategy · Probability model · Global optimization

1 Introduction

With the continuous development of science and technology, many global optimization problems (GOPs) constantly exist in all most of engineering and science fields, such as portfolio investment (Shalan and Ykhlef 2015), queueing system (Wei et al. 2013) and structural design (Aydogdu et al. 2016). Since these GOPs are more and more complex, and they are characterized as multimodality, discontinuity, highly non-linearity, non-differentiability and non-convexity, these kinds of GOPs are difficult or even impracticable to be solved by traditional optimization methods, especially for the gradient-based methods (Hu et al. 2015, 2016). Thus, many swarm-based intelligence evolution algorithms (EAs), such as Particle Swarm Optimization (PSO) (Kennedy and Eberhart 1995; Kuo et al. 2011), Ant Colony Optimization (ACO) (Dorigo et al. 1996; Mavrovouniotis and Yang 2011), Genetic Algorithm (GA) (Tang et al. 1996; Hunter and Chiu 2000), and Differential Evolution (DE) (Storm and Price 1997; Teo 2006), Artificial Bee Colony (ABC) algorithm, have been proposed to address this challenge task (Cui and Gao 2012) and they also have shown great potential.

In this paper, we focus on the study of artificial bee colony algorithm firstly proposed by Karaboga (2005), which inspired by the intelligent foraging behavior of real bee colony. The performance of ABC is firstly demonstrated by

Communicated by V. Loia.

✉ Genghui Li
li_genghui@126.com

Jian Lu
jianlu@szu.edu.cn

[1] College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, People's Republic of China

[2] College of Mathematics and Statistics, Shenzhen University, Shenzhen, People's Republic of China

the comparison with other EAs, such as GA, PSO and DE, on many benchmark functions. The results of the simulation experiments show that the performance of ABC is better than or at least comparable to PSO, GA and DE (Karaboga and Basturk 2007, 2008; Karaboga and Akay 2009). Due to its simply structure, ease to implementation and outstanding performance, ABC has attracted great attention and has been successfully applied to solve multi-objective optimization problems (Reza et al. 2012), constrained optimization problems (Karaboga and Akay 2011), binary optimization problems (Ozturk et al. 2015), clustering problems (Banharnsakun et al. 2013) and many practical optimization problems (Ma et al. 2011; Sun et al. 2011; Krink and Paterlini 2011). However, like other EAs, ABC often shows slow convergence speed since its solution search does well in exploration, but badly in exploitation (Karaboga 2005). It is well known that both exploration and exploitation are necessary for a population-based optimization algorithm, and the performance of the EAs depends on whether a suitable balance between exploration and exploitation can be found or not (Kiran et al. 2015; Gao et al. 2014, 2015c). Therefore, the performance of ABC can be improved by enhancing the exploitation ability and finding better balance between exploration and exploitation. A large number of improved ABC variants have been presented by exploiting the valuable information from the current best solution or other good solutions, such as literatures (Zhang et al. 2015; Gao and Liu 2011). The experimental results demonstrate that the population moderately guided by the current best solution or other good solutions can effectively improve the exploitation ability and enhance the performance of ABC. Based on this outcome, in this paper, we firstly propose a novel search strategy to make use of the current best solution, in which the original solution search equation and a new solution search equation exploiting the current best solution are corporately used according to a probability parameter P for generating a candidate food source. Secondly, in order to utilize other good solutions, we introduce a novel probability model on onlooker bee phase, in which the good food sources could attract more onlooker bees to search. The new search strategy and novel probability model are embedded into original ABC to form a new variant of ABC, named MPGABC.

To evaluate the performance of MPGABC, some extensive experiments are conducted on 22 common benchmark functions (Gao et al. 2013a), 22 real-world problems (Das and Suganthan 2010) and 28 real-parameter optimization problems (Liang et al. 2013). The comparison results with other variants of ABC (i.e., GABC Zhu and Kwong 2010, qABC Karaboga and Gorkemli 2014, best-so-far ABC Banharsakunm et al. 2011, dABC Kiran and Findik 2015 and MABC Gao and Liu 2012) validate the effectiveness and efficiency of MPGABC in terms of solution quality, robustness and convergence speed.

The remainder of this paper is organized as follows. Section 2 introduces the original ABC algorithm briefly. A review on the improved ABC variants is given in Sect. 3. The details of MPGABC, which mainly includes the novel search strategy and the novel probability model, are described in Sect. 4. The experimental results and corresponding analysis are given in Sect. 5. Finally, Sect. 6 concludes this paper.

2 The original ABC

The ABC algorithm is a process of searching the optimal solution by simulating the collective foraging behavior of honey bees. In ABC, a food source position denotes a possible solution of the optimal problem, and the nectar amount of each food source represents the quality of the corresponding solution. In order to find better food sources, three types of bees, namely employed bee, onlooker bee and scout bee, search collectively. Firstly, employed bees are composed of the first half of the colony, which are mainly responsible for randomly searching better food sources in the neighborhood of the corresponding parent food source. Moreover, employed bees will share the quality information of their food sources with onlooker bees after all employed bees find out a new food resource. Secondly, onlooker bees are consisted of the second half of the colony, which are mainly responsible for searching better food sources in the neighborhood of the good solutions according to the information provided by employed bees. Thirdly, if a better food source cannot be found by a preset number of times (*limit*) in the neighborhood of a certain food source, it will be abandoned by its employed bee, and this employed bee will become a scout bee to search a new food source randomly in all search space. The original ABC includes four phase, i.e., initialization phase, employed bee phase, onlooker bee phase and scout bee phase. After the initialization phase, ABC enters a loop of employed bee phase, onlooker bee phase and scout bee phase until the termination condition is met. The detailed descriptions of each phase are described as follows.

Initialization phase The initial population is generated according to Eq. (1), which contains *SN* food sources (solutions),

$$x_{i,j} = x_j^{\min} + rand(0,1) \cdot (x_j^{\max} - x_j^{\min}) \tag{1}$$

where $i = 1, 2, \ldots SN$, $j = 1, 2, \ldots D$. *SN* is the number of employed bees or onlooker bees; D is the dimensionality of the search space; $x_j^{\max}$ and $x_j^{\min}$ represent the upper bound and lower bound of the jth dimension, respectively. Moreover, the fitness value of each food source will be calculated by Eq. (2),

$$fit(x_i) = \begin{cases} \frac{1}{1+f(x_i)} & \text{if } (f(x_i) \geq 0) \\ 1 + \text{abs}(f(x_i)) & \text{otherwise} \end{cases} \tag{2}$$

where $fit(x_i)$ represents the fitness value of the ith food source x_i and $f(x_i)$ is the objective function value of food source x_i for the optimization problem.

Employed bee phase Each employed bee selects a distinct food source to search and generate a candidate food source in the neighborhood of this selected food source according to Eq. (3),

$$v_{i,j} = x_{i,j} + \phi_{i,j} \cdot (x_{i,j} - x_{k,j}) \tag{3}$$

where v_i is the ith candidate food source and x_i is the ith food source. x_k is randomly selected from the population, which is different from x_i. $\phi_{i,j}$ is the uniformly distributed random number in the range of $[-1, 1]$ and j is randomly selected from $\{1, 2, \ldots, D\}$. If the fitness value of v_i is better than its parent x_i, x_i will be replaced by v_i, and the *counter* which records the number of the consecutive unsuccessful updates of the food source x_i is reset to 0. Otherwise, x_i is kept to enter into the next generation and *counter* is increased by 1.

Onlooker bee phase After each employed bee finishes its search task, it will show the quality information of its food source with onlooker bees. Each onlooker bee will select a food source to search based on the probability of each food source, which is calculated as Eq. (4). Obviously, the better the fitness value is, the bigger the selection probability is. And then it will further search better food source in the neighborhood of the selected food source by using Eq. (3). If a candidate food source obtained by the onlooker bee is better than its parent food source, the parent food source will be replaced by the new one, and its *counter* is reset to 0. Otherwise, the old one is kept and *counter* is increased by 1.

$$p(x_i) = \frac{fit(x_i)}{\sum_{j=1}^{SN} fit(x_j)} \tag{4}$$

Scout bee phase The food source with the highest *counter* value is selected. If its *counter* value is bigger than the *limit* value, the selected food source will be abandoned by its employed bee, and then, this employed bee will become a scout bee to seek a new food source randomly according to Eq. (1). After the new food source is generated, the corresponding *counter* value is reset to 0, and the scout bee returns to be an employed bee.

Note that if the jth variable $v_{i,j}$ of the ith candidate food source violates the boundary constraints in employed bee phase and onlooker bee phase, it will be reset according to Eq. (1). The detailed procedure of ABC is shown in Fig. 1.

3 The improved ABC variants

Over the past decade, a lot of in-depth research works on ABC have been conducted to achieve a better performance on solving GOPs. A brief survey of the improved ABC approaches can be generally classified into three categories, i.e., modified solution search equation, combination with other technologies and hybridization of ABC with other EAs. They are, respectively, introduced as follows.

(1) *Modified solution search equation* Inspired by PSO, Zhu and Kwong (2010) propose an improved ABC algorithm, called as GABC, which incorporates the valuable information of the global best solution into their solution search equation to improve the exploitation ability of ABC. The experimental results show that GABC is better than the original ABC on most of cases. Banharnsakun et al. (2011) propose a new ABC variant, named best-so-far ABC, which introduces the best-so-far selection method and an adjustable search radius into the solution search equation. In best-so-far ABC, the current best solution is shared globally among the entire population. Inspired by the DE mutation operator DE/best/1 and DE/rand/1, Gao and Liu (2011; 2012) propose two novel search equations ABC/best/1 and ABC/rand/1. Besides, in order to appropriately take advantage of them, a selective probability P is used to control the frequency of employing ABC/rand/1 and ABC/best/1. Moreover, a new algorithm, named ABCgbest (Gao et al. 2012), is proposed by them, in which bees only search around the current best food source. Recently, Karaboga and Gorkemli (2014) present a new version of search equation for onlooker bee (qABC), which uses the valuable information of the best solution among the neighbors to improve the search efficiency of ABC. Gao et al. (2015a) propose a Gaussian search equation (BABC), which exploits the current best solution and Gaussian distribution to produce candidate solutions in the onlooker bee phase. Luo et al. (2013) present a new solution search equation for the onlooker bee, which exploits the best solution of the previous iteration to guide the search of new candidate solutions (called COABC). Gao et al. (2014, 2015c) propose two modified search equations. The one of them utilizes the beneficial information of best solutions in a cluster, which is employed in employed bee phase. And the other one learns from the current best solution, which is used in onlooker bee phase. Since different search equations have distinct advantages and perform differently on different problems or at different stages on the same problem, some methods with multiple search equations are proposed to enhance the comprehensive performance of ABC, such as MEABC (Wang et al. 2014), ABCVSS (Kiran et al. 2015) and MuABC (Gao et al. 2015b). In addition, Kiran and Findik (2015) add the directional information into ABC and design a new search mechanism, which selects search equation to produce candidate solution according to the previous directional information (named dABC).

```
Algorithm 1:The procedure of ABC
Initialization: Generate $SN$ solutions that contain $D$ variables according to Eq. (1)
While $FES < maxFES$
    for $i = 1$ to $SN$    // employed bee phase
        Generate a new solution $v_i^G$ in the neighbourhood of $x_i^G$ using Eq. (3)
        Evaluate the new solution $v_i^G$
        if $fit(v_i^G) \geq fit(x_i^G)$
            Replace $x_i^G$ by $v_i^G$
            $counter(i)=0$
        else
            $counter(i)= counter(i)+1$
        end if
    end for                                    // end employed bee phase
    Calculate the probability $p$ according to Eq. (4) // onlooker bee phase
    for $i =1$ to $SN$
        Select a solution $x_s^G$ from the population according to probability $p$
        Generate a new solution $v_s^G$ in the neighbourhood of $x_s^G$ using Eq. (3)
        Evaluate the new food source $v_s^G$
        if $fit(v_s^G) \geq fit(x_s^G)$
            Replace $x_s^G$ by $v_s^G$, $counter(s)=0$
        else
            $counter(s)= counter(s)+1$
        end if
    end for                             // end onlooker phase
    $FES=FES+2SN$
    Select the solution $X_{\max}^G$ with max counter value // scout phase
    if $counter(\max)>limit$
        Replace $X_{\max}^G$ by a new solution generated according to Eq.(1)
        $FES=FES+1$, $counter(\max)=0$
    end if                              // end scout phase
end while
Output: The food source (solution) with the smallest objective value
```

Fig. 1 The pseudo-code of original ABC

(2) *Combination with other technologies* Inspired by the concept of Grenade Explosion Method (GEM), two modified versions of ABC (namely GABC1 and GABC2) are proposed by Zhang et al. (2015), which show high robustness and fast convergence speed. Gao and Liu (2012; 2013a) employ the chaotic map (Feng et al. 2011) and opposition-based learning method in the initial phase and the orthogonal learning method to enhance the performance of ABC. Moreover, Gao et al. (2013b) also use the Powell's method as a local search tool to improve the exploitation ability of ABC. In addition, Kang et al. (2011a; 2011b) proposed two new ABC variants (called RABC and HJABC), in which the original ABC is used to realize the exploration ability and the exploitation phase is completed by the rotational direction method and Hooke Jeeves pattern search technology, respectively. Moreover, Akay and Karaboga (2012) propose two control parameters, i.e., modification rate (MR) and scaling factor (SF), to control frequency and magnitude of perturbation, respectively. Loubiere et al. (2016) apply a sensitivity analysis method, i.e., Morris' OAT method (One-At-Time), to select dimensions with high influence on the objective result for preferential evolvement. Xiang and An (2013) employ the chaotic search

technique on initialization phase and scout bee phase to, respectively, enhance the global convergence and prevent falling into local optimum (called ERABC). In addition, the memory-save technology has attracted much attention. Kiran and Babalik (2014) add a memory board to save the solutions whose qualities are better than the average fitness value. On the contrary, Bayraktar (2014) uses the short-term tabu list (STTL) of tabu search to memorize the abandoned solution. Moreover, Li and Yang (2016) introduce a new ABC variant named ABC with memory algorithm (ABCM), which memorizes the previous successful experiences of foraging behavior to guide the current foraging behavior.

(3) *Hybridization of ABC with other EAs* Kang et al. (2009) take the advantages of Nelder–Mead simplex method and ABC to develop a hybrid ABC, named HSABCA. Xiang et al. (2014) propose a well-known hybrid algorithm named hABCDE, which incorporates a modified ABC with a modified DE. In addition, Shi et al. (2010) present a hybrid intelligent algorithm (named IABAP) based on the information exchange process, in which all population share beneficial information between particle swarm and bee colony. Marinakis et al. (2009) propose a hybrid algorithm for clustering by combining ABC optimization with the greedy randomized adaptive search procedure (GRASP method). Xiao and Chen (2011) propose a novel hybrid algorithm, which consists of ABC algorithm and artificial immune network algorithm to address multi-mode resource constrained multi-project scheduling problem. Sharma and Pant (2011) embed the DE mutation operators into the framework of ABC algorithm and develop a hybrid ABC (called DE-ABC). Hsieh et al. (2012) introduce PSO into ABC and propose a new hybrid algorithm, named EABC-PGSVM. Abraham et al. (2012) invent a novel hybrid differential artificial bee colony algorithm (called HDABCA), which combines DE strategy with standard ABC algorithm. Besides, Tuba and Bacanin (2014) combine ABC with firefly algorithm, Fister et al. (2012) mix ABC with memetic search, and Chen et al. (2012) integrate ABC with annealing algorithm, and so on.

4 The proposed algorithm

In this section, the proposed algorithm is described in detail. Firstly, we give the motivations of our proposed algorithm. Secondly, the novel search strategy and the novel probability model are presented, respectively. Finally, the complete proposed algorithm is shown.

Algorithm 2: The novel search strategy

```
if  $rand(0,1) < P$
    $v_{i,j} = x_{i,j} + \phi_{i,j} \cdot (x_{i,j} - x_{k,j})$
else
    $v_{i,j} = x_{i,j} + \varphi_{i,j} \cdot (x_{best,j} - x_{i,j})$
end
```

Fig. 2 The pseudo-code of the novel search strategy

4.1 Motivations

Generally, the performance evaluation of Evolutionary Algorithms (EAs) depends on that a suitable trade-off between exploration and exploitation can be maintained. Usually, exploiting the information of the current best solution and other good solutions could enhance the exploitation ability, while excessively exploiting could degrade the exploration ability. Therefore, the key issue is how to exploit the current best solution and other good solutions. On the one hand, in the search Eq. (3) of the original ABC, only the target solution and a randomly selected solution are used to generate new solution, which does well in exploration but badly in exploitation. On the other hand, in the search equations of other ABC variants (Zhang et al. 2015; Wang et al. 2014), the current best solution is always used, which performs well in exploitation but relatively underperforms in exploration. Therefore, the incorporation of the original solution search equation and a new search equation utilizing the best solution could get a better balance between exploration and exploitation. Moreover, in the onlooker bee phase of original ABC, the onlooker bee selects a food source to search according to the information provided by employed bees. In theory, the food source with higher quality could attract more onlooker bees to search. However, if the qualities (fitness values) of all food sources have no significant difference, all food sources will obtain nearly the same probability according to Eq. (4), which makes that the good solutions are not fully exploited. Therefore, a new probability model should make sure that the good solutions are more likely to be chosen by onlooker bee for searching, which could enhance the performance of ABC.

4.2 The novel search strategy

In order to enhance the exploitation ability and accelerate the convergence speed of ABC, Zhu and Kwong (2010) propose a new solution search equation in GABC as Eq. (5),

$$v_{i,j} = x_{i,j} + \phi_{i,j} \cdot (x_{i,j} - x_{k,j}) + \varphi_{i,j} \cdot (x_{\text{best},j} - x_{i,j}) \quad (5)$$

where x_{best} is the current best solution of the population, and x_i is the ith food source. x_k is a randomly selected food source from the population, which is different from x_k. $\phi_{i,j}$ is the uniformly distributed random number in the range of $[-1, 1]$ and $\varphi_{i,j}$ is a uniform random number in the range of $[0, 1.5]$; j is randomly selected from $\{1, 2, \ldots, D\}$.

Although Eq. (5) exploits the current best solution, there are two drawbacks. Firstly, as claimed in Gao et al. (2013a), Eq. (5) may cause the oscillation phenomenon when the guidance of the last two terms ($\phi_{i,j} \cdot (x_{i,j} - x_{k,j})$ and $\varphi_{i,j} \cdot (x_{\text{best},j} - x_{i,j})$) is in opposite direction, which could degrade convergence. Secondly, Eq. (5) may easily make the new solution $v_{i,j}$ violate the boundary constraints when the last two terms are in identical direction, which also could hinder convergence. In order to address these issues, we propose a novel search strategy shown in **Algorithm 2** (Fig. 2).

In Algorithm 2, P is a parameter defined by the user. x_{best} is the current best solution of the population, and x_i is the ith food source. x_k is a randomly selected food source from the population, which is different from x_i. $\phi_{i,j}$ is the uniformly distributed random number in the range of $[-1, 1]$ and $\varphi_{i,j}$ is a uniform random number in the range of $[0, 1.5]$. j is randomly selected from $\{1, 2, \ldots, D\}$. Obviously, with the guidance from only one term, the novel search strategy can easily avoid the oscillation phenomenon and violating the boundary constraints. Moreover, parameter P could be used to control how to appropriately exploit the valuable information of the current best solution. The effectiveness of the novel search strategy is validated through extensive experiments in Sect. 5.2, and the sensitiveness of parameter P is analyzed experimentally in Sect. 5.4.

4.3 The novel probability model

In the onlooker bee phase of the original ABC, onlooker bees mainly select good solutions to search according to the information shared by the employed bees. From our above analysis, the original probability model Eq. (4) for onlooker bee phase is unable to make sure that the good solutions could attract more onlooker bees than the bad solutions to search. In order to address this issue, we propose a novel probability model as Eq. (6),

$$p(x_i) = \frac{0.8}{(\mathrm{e}^{r(x_i)/SN})\sqrt{r(x_i)}} \tag{6}$$

where $p(x_i)$ is the selected probability of the ith food source and SN is the number of employed bees or onlooker bees. $r(x_i)$ is the ranking of the ith food source in ascending order among all food sources according to the objective function value. For example, the ranking of the current best food source and worst food source is 1 and SN, respectively. To explain clearly, when $SN = 50$, the selection probability of each food source is illustrated in Fig. 3, where X axis denotes the ranking of the food source, and Y axis denotes the corresponding probability. Obviously, the selected probability of a good solution is significantly larger than a bad solution. The effectiveness of the novel probability model is validated in Sect. 5.2.

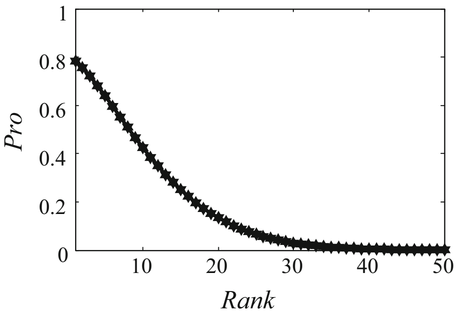


Fig. 3 An illustrated example of the novel probability model

4.4 The complete proposed algorithm (MPGABC)

In this subsection, we put forward a novel ABC variant (named MPGABC) by combining the novel search strategy and the novel probability model with the basic framework of ABC. The pseudo-code of the complete MPGABC is demonstrated in Fig. 4. Obviously, the differences of **Algorithm 1 (ABC)** and **Algorithm 3 (MPGABC)** are shown in lines **4, 13** and**16.**

5 Experimental results

5.1 Benchmark test functions and experimental settings

In this paper, 22 well-known benchmark functions (Gao et al. 2013a; Kiran et al. 2015; Zhang et al. 2015) with low dimension ($D = 30$), middle dimension ($D = 50$) and high dimension ($D = 100$) are employed to validate the performance of MPGABC. These functions have different characteristics, such as continuous unimodality function ($f_1 - f_6$ and f_8), multimodality functions ($f_{11} - f_{22}$), discontinuous step function (f_7) and noisy-quartic function (f_9). Particularly, f_{10} is the Rosenbrock function, which is unimodal for $D = 2$ and $D = 3$, but it may have many optimal solutions when $D > 3$. Generally speaking, the unimodal functions can be used to test the exploitation ability and the multimodal functions can be employed to demonstrate the exploration ability. The mathematical expression, search range, the global optimal value and the "acceptable value" of each function are listed in columns 2, 3, 4 and 5 of Table 1, respectively. When the objective function value of the best solution obtained by an algorithm in a run is less than the acceptable value, this run is regarded as a *successful run*.

Table 1 Benchmark functions in experiments

Name	Function	Range	Min	Accept
Sphere	$f_1(x) = \sum_{i=1}^{D} x_i^2$	$[-100,\ 100]^D$	0	1×10^{-8}
Elliptic	$f_2(x) = \sum_{i=1}^{D} (10^6)^{\frac{i-1}{D-1}} x_i^2$	$[-100,\ 100]^D$	0	1×10^{-8}
SumSquare	$f_3(x) = \sum_{i=1}^{D} i x_i^2$	$[-10,\ 10]^D$	0	1×10^{-8}
SumPower	$f_4(x) = \sum_{i=1}^{D} \lvert x_i \rvert^{(i+1)}$	$[-1,\ 1]^D$	0	1×10^{-8}
Schwefel 2.22	$f_5(x) = \sum_{i=1}^{D} \lvert x_i \rvert + \prod_{i=1}^{D} \lvert x_i \rvert$	$[-10,\ 10]^D$	0	1×10^{-8}
Schwefel 2.21	$f_6(x) = \max\{\lvert x_i \rvert,\ 1 \le i \le n\}$	$[-100,\ 100]^D$	0	1×10^{0}
Step	$f_7(x) = \sum_{i=1}^{D} (\lfloor x_i + 0.5 \rfloor)^2$	$[-100, 100]^D$	0	1×10^{-8}
Exponential	$f_8(x) = \exp\left(0.5 \times \sum_{i=1}^{D} x_i\right)$	$[-10, 10]^D$	0	1×10^{-8}
Quartic	$f_9(x) = \sum_{i=1}^{D} i x_i^4 + \text{random}[0, 1]$	$[-1.28, 1.28]^D$	0	1×10^{-1}
Rosenbrock	$f_{10}(x) = \sum_{i=1}^{D-1} \left[100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right]$	$[-5,\ 10]^D$	0	1×10^{-1}
Rastrigin	$f_{11}(x) = \sum_{i=1}^{D} \left[x_i^2 - 10\cos(2\pi x_i) + 10\right]$	$[-5.12, 5.12]^D$	0	1×10^{-8}
NCRastrigin	$f_{12}(x) = \sum_{i=1}^{D} \left[y_i^2 - 10\cos(2\pi y_i) + 10\right]$ $y_i = \begin{cases} x_i & \lvert x_i \rvert < \frac{1}{2} \\ \frac{\text{round}(2x_i)}{2} & \lvert x_i \rvert \ge \frac{1}{2} \end{cases}$	$[-5.12, 5.12]^D$	0	1×10^{-8}
Griewank	$f_{13}(x) = 1/4000 \sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600,\ 600]^D$	0	1×10^{-8}
Schwefel2.26	$f_{14}(x) = 418.98288727243380 * D - \sum_{i=1}^{D} x_i \sin\left(\sqrt{\lvert x_i \rvert}\right)$	$[-500, 500]^D$	0	1×10^{-8}
Ackley	$f_{15}(x) = 20 + e - 20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right)$ $- \exp\left(\frac{1}{D}\sum_{i=1}^{D} \cos(2\pi x_i)\right)$	$[-50, 50]^D$	0	1×10^{-8}
Penalized1	$f_{16}(x) = \frac{\pi}{D}\left\{10\sin^2(\pi y_1) + \sum_{i=1}^{D-1} (y_i - 1)^2 \left[1 + 10\sin^2(\pi y_{i+1})\right]\right.$ $\left. + (y_D - 1)^2\right\} + \sum_{i=1}^{D} u(x_i, 10, 100, 4)$ $y_i = 1 + 1/4(x_i + 1),\ u_{x_i,a,k,m} = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a \le x_i \le a \\ k(-x_i - a)^m & x_i < a \end{cases}$	$[-100, 100]^D$	0	1×10^{-8}
Penalized2	$f_{17}(x) = \frac{1}{10}\left\{\sin^2(\pi x_1) + \sum_{i=1}^{D-1} (x_i - 1)^2 \left[1 + \sin^2(3\pi x_{i+1})\right]\right.$ $\left. + (x_D - 1)^2 \left[1 + \sin^2(2\pi x_{i+1})\right]\right\} + \sum_{i=1}^{D} u(x_i, 5, 100, 4)$	$[-100,\ 100]^D$	0	1×10^{-8}
Alpine	$f_{18}(x) = \sum_{i=1}^{D-1} \lvert x_i \cdot \sin(x_i) + 0.1 \cdot x_i \rvert$	$[-10,\ 10]^D$	0	1×10^{-8}
Levy	$f_{19}(x) = \sum_{i=1}^{D-1} (x_i - 1)^2 \left[1 + \sin^2(3\pi x_{i+1})\right] + \sin^2(3\pi_1)$ $+ \lvert x_D - 1 \rvert \left[1 + \sin^2(3\pi x_D)\right]$	$[-10,\ 10]^D$	0	1×10^{-8}
Weierstrass	$f_{20}(x) = \sum_{i=1}^{D} \left(\sum_{k=0}^{k_{\max}} \left[a^k \cos\left(2\pi b^k (x_i + 0.5)\right)\right] - D \sum_{k=0}^{k_{\max}} \left[a^k\right.\right.$ $\left.\left. \cos\left(2\pi b^k 0.5\right)\right], a = 0.5, b = 3, k_{\max} = 20$	$[-1,\ 1]^D$	0	1×10^{-8}
Himmelblau	$f_{21}(x) = 1/D \sum_{i=1}^{D} \left(x_i^4 - 16x_i^2 + 5x_i\right)$	$[-5,\ 5]^D$	−78.33236	−78
Michalewicz	$f_{22}(x) = -\sum_{i=1}^{n} \sin(x_i) \sin^{20}\left(\frac{i \times x_i^2}{\pi}\right)$	$[0,\ \pi]^D$	−30,−50, −100	−29,−49,−99

To evaluate the performance of MPGABC, three evaluation metrics are used in our experiments. The detailed descriptions are given as follows:

(1) The mean and standard deviation (mean/std): They are utilized to evaluate the accuracy of the best objective function value for different algorithms. For minimum optimization problem, the smaller the value of mean and standard deviation is, the higher quality/accuracy of the solution has.

(2) The average number of function evaluation (AVEN): It is required to reach the acceptable value, which is adopted to evaluate the convergence speed. The smaller AVEN is, the faster the convergence speeds is. Note that AVEN will be only calculated for the successful runs. If an algorithm cannot find any solution whose objective function value is smaller than the acceptant value in all runs, AVEN will be denoted by "NAN."

(3) The successful rate (SR): The successful rate (SR %) of the 25 independent runs is utilized to evaluate the robustness or reliability of different algorithms. The

greater the value of this metric is, the better the robustness/reliability is.

There are five parts in our experiments. *Experiment* **1** validates the effectiveness of our proposed algorithmic components (i.e., the novel search strategy and the novel probability model). *Experiment* **2** evaluates the performance of MPGABC by the comparison with other outstanding ABC variants on some common benchmark functions. *Experiment* **3** analyzes the sensitiveness of parameter P on the performance of MPGABC. *Experiment* **4** demonstrates the performance of MPGABC by the comparison of MPGABC and other ABC variants on real-world problems of CEC 2011. *Experiment* **5** investigates the effectiveness of MPGABC by the comparison with ABC variants on the real-parameter optimization problems of CEC 2013.

5.2 Experiment 1: the effectiveness of the proposed algorithmic components

In order to demonstrate the effectiveness and efficiency of the two proposed algorithmic components (i.e., the novel search strategy and the novel probability model), the original ABC, GABC and the following three ABC variants (MGABC, PABC and MPGABC) are used to study in this experiment on 22 benchmark functions with 30D.

(1) ABC only combines with the novel search strategy, denoted by MGABC (the modified GABC).
(2) ABC only combines with the novel probability model, denoted by PABC.
(3) ABC combines with the novel search strategy and the novel probability model, denoted by MPGABC.

Each algorithm will be conducted 25 times (Shan et al. 2015; Omidvar et al. 2014) independent run for all test functions. For a fair comparison, we use the maximum number of function evaluation (*maxFES*) as the termination condition, which is set to 150000 (Wang et al. 2014). Moreover, the same parameter settings ($SN = 50$, $limit = SN \cdot D$) are used in all compared algorithms. P is set to 0.3 in the novel search strategy. The experimental results are shown in Table 2.

It can be seen from Table 2 that MGABC and PABC are better than or at least comparable to GABC and the original ABC, respectively, in terms of solution accuracy and convergence rate on most cases. To be specific, MGABC is significant better than GABC on all functions except f_6, f_9, f_{10} and f_{13}. Moreover, PABC outperforms the original ABC on all functions excluding f_4, f_5, f_{14}, f_{15} and f_{19}. In addition, MPGABC outperforms MGABC and PABC on all cases excluding f_6 and f_{10}. This phenomenon effectively validates that each of our proposed components can make contribution to ABC and improve the performance of ABC.

Table 2 Comparisons of MPGABC variants and corresponding ABC on 22 functions with 30D

Alg	Metric	GABC	MGABC	ABC	PABC	MPGABC
f_1	Mean (SD)	6.97e−33 (4.93e−33)	1.54e−39 (1.66e−39)	8.05e−18 (6.08e−18)	1.77e−19 (1.60e−19)	5.41e−55 (1.95e−54)
	SR/AVEN	100/50130	100/33550	100/82934	100/66514	100/30494
f_2	Mean (SD)	1.92e−26 (2.12e−26)	3.55e−36 (8.59e−36)	4.77e−10 (3.76e−10)	1.49e−14 (3.49e−14)	5.24e−50 (2.20e−49)
	SR/AVEN	100/76150	100/45350	100/135230	100/99098	100/39274
f_3	Mean (SD)	2.98e−34 (2.38e−34)	4.90e−40 (5.01e−40)	1.55e−19 (1.31e−19)	2.45e−20 (1.59e−20)	1.89e−55 (4.07e−55)
	SR/AVEN	100/45478	100/33226	100/75366	100/59930	100/28214
f_4	Mean (SD)	1.83e−52 (6.33e−52)	4.72e−71 (2.11e−70)	2.41e−31 (9.09e−31)	3.90e−27 (6.74e−27)	1.40e−96 (6.98e−96)
	SR/AVEN	100/14106	100/12934	100/23266	100/16854	100/7922
f_5	Mean (SD)	5.95e−18 (1.76e−18)	2.00e−20 (1.33e−20)	6.55e−11 (2.12e−11)	1.33e−10 (3.87e−11)	2.60e−29 (8.89e−29)
	SR/AVEN	100/77478	100/66882	100/125030	100/119470	100/45478
f_6	Mean (SD)	2.55e−01 (1.30e−01)	2.19e+01 (5.46e+00)	4.35e+00 (8.60e−01)	8.67e−01 (1.92e−01)	9.87e+00 (3.32e+00)
	SR/AVEN	100/109060	0/NAN	0/NAN	72/136900	0/NAN
f_7	Mean (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)
	SR/AVEN	100/10314	100/12818	100/11314	100/18638	100/12530

Table 2 continued

Alg	Metric	GABC	MGABC	ABC	PABC	MPGABC
f_8	Mean (SD)	7.18e−66 (9.22e−77)	7.18e−66 (5.32e−76)	7.18e−66 (4.37e−73)	7.18e−66 (3.91e−73)	7.18e−66 (4.02e−79)
	SR/AVEN	100/150	100/150	100/150	100/150	100/150
f_9	Mean (SD)	2.80e−02 (6.51e−03)	3.37e−02 (7.12e−03)	6.42e−02 (1.37e−02)	3.83e−02 (9.20e−03)	2.46e−02 (6.43e−03)
	SR/AVEN	100/41966	100/45542	100/93186	100/50426	100/34834
f_{10}	Mean (SD)	8.21e−01 (3.73e+00)	3.61e+00 (1.49e+01)	6.79e−02 (5.93e−02)	4.46e−02 (5.10e−02)	7.43e−02 (1.82e−01)
	SR/AVEN	68/77515	52/96942	72/120030	88/90859	84/82945
f_{11}	Mean (SD)	0.00e+00 (0.00e+00)	0.00e+00 (0.00e+00)	2.68e−14 (1.03e−13)	6.11e−15 (7.56e−15)	0.00e+00 (0.00e+00)
	SR/AVEN	100/68134	100/67434	100/99214	100/90818	100/55254
f_{12}	Mean (SD)	0.00e+00 (0.00e+00)	0.00e+00 (0.00e+00)	4.25e−13 (1.57e−12)	4.04e−14 (3.11e−14)	0.00e+00 (0.00e+00)
	SR/AVEN	100/76642	100/78382	100/110050	100/98918	100/61378
f_{13}	Mean (SD)	4.51e−08 (2.25e−07)	2.89e−06 (1.40e−05)	3.08e−04 (1.54e−03)	2.96e−04 (1.48e−03)	2.11e−09 (1.06e−08)
	SR/AVEN	96/61688	92/65763	96/96783	96/82013	96/44275
f_{14}	Mean (SD)	2.18e−13 (6.03e−13)	3.57e−12 (3.64e−13)	4.51e−12 (1.59e−12)	4.37e−12 (1.05e−12)	3.49e−12 (5.04e−13)
	SR/AVEN	100/65670	100/63854	100/84338	100/98478	100/63062
f_{15}	Mean (SD)	1.49e−14 (2.92e−15)	1.46e−14 (3.96e−15)	3.83e−09 (2.27e−09)	9.91e−09 (3.84e−09)	1.42e−14 (3.59e−15)
	SR/AVEN	100/89178	100/83434	96/144000	60/147960	100/54498
f_{16}	Mean (SD)	1.57e−32 (5.59e−48)	1.57e−32 (5.59e−48)	1.29e−18 (1.76e−18)	2.39e−20 (2.79e−20)	1.57e−32 (5.59e−48)
	SR/AVEN	100/45786	100/35010	100/79398	100/57114	100/28018
f_{17}	Mean (SD)	4.06e−33 (2.30e−33)	1.50e−33 (0.00e+00)	8.19e−18 (1.71e−17)	3.87e−19 (3.42e−19)	1.50e−33 (0.00e+00)
	SR/AVEN	100/49750	100/36930	100/84730	100/68458	100/30430
f_{18}	Mean (SD)	3.88e−07 (6.54e−07)	8.13e−07 (7.41e−07)	3.15e−06 (1.85e−06)	2.49e−07 (2.54e−07)	1.83e−08 (4.32e−08)
	SR/AVEN	16/129980	4/132550	0/NAN	0/NAN	76/110410
f_{19}	Mean (SD)	1.39e−31 (1.41e−32)	1.35e−31 (2.23e−47)	8.23e−14 (1.25e−13)	3.03e−12 (9.21e−12)	1.35e−31 (2.23e−47)
	SR/AVEN	100/50934	100/42786	100/91734	100/85838	100/29430
f_{20}	Mean (SD)	3.60e−02 (4.19e−02)	6.26e−03 (1.21e−02)	3.06e−02 (3.75e−02)	2.02e−02 (2.16e−02)	1.16e−03 (2.15e−03)
	SR/AVEN	0/NAN	16/118280	0/NAN	0/NAN	28/132380
f_{21}	Mean (SD)	−7.833e+01 (5.02e−15)	−7.833e+01 (2.90e−15)	−7.83e+01 (4.10e−15)	−7.83e+01 (0.00e+00)	−7.833e+01 (2.90e−15)
	SR/AVEN	100/15986	100/17334	100/26934	100/13578	100/9774
f_{22}	Mean (SD)	−2.999e+01 (1.01e−03)	−2.999e+01 (2.25e−04)	−2.999e+01 (8.26e−04)	−3.00e+01 (8.23e−05)	−3.00e+01 (9.51e−05)
	SR/AVEN	100/21778	100/19222	100/25362	100/10694	100/9586

```
Algorithm 3:The procedure of MPGABC
Initialization: Generate $SN$ solutions that contain $D$ variables according to Eq. (1)
while $FES < maxFES$
    for $i$ =1 to $SN$   // employed bee phase
        Generate a new solution $v_i^G$ in the neighbourhood of $x_i^G$ using Algorithm 2
        Evaluate the new solution $v_i^G$
        if $fit(v_i^G) \geq fit(x_i^G)$
            Replace $x_i^G$ by $v_i^G$
            $counter(i)$=0
        else
            $counter(i)$= $counter(i)$+1
        end if
    end for                                  // end employed bee phase
    Calculate the probability $p$ according to Eq. (6) // onlooker bee phase
    for $i$ =1 to $SN$
        Select a solution $x_s^G$ from the population according to probability $p$
        Generate a new solution $v_s^G$ in the neighbourhood of $x_s^G$ using Algorithm 2
        Evaluate the new food source $v_s^G$
        if $fit(v_s^G) \geq fit(x_s^G)$
            Replace $x_s^G$ by $v_s^G$
            $counter(s)$=0
        else
            $counter(s)$= $counter(s)$+1
        end if
    end for                           // end onlooker phase
    $FES$=$FES$+2$SN$
    Select the solution $X_{\max}^G$ with max $counter$ value // scout phase
    if $counter(\max) > limit$
        Replace $X_{\max}^G$ by a new solution generated according to Eq.(1)
        $FES$=$FES$+1, $counter(\max)$=0
    end if                            // end scout phase
end while
Output: The food source (solution) with the smallest objective value
```

Fig. 4 The pseudo-code of MPGABC

Besides, through the combination of two proposed algorithmic components, the performance of ABC can be further significantly improved.

To clearly show the convergence rate, the AVEN of GABC, MGABC, the original ABC, PABC and MPGABC are plotted in Fig. 5a, which clearly indicates that MGABC and PABC are better than GABC and ABC, respectively, regarding to convergence speed on most test functions. This result demonstrates that the novel search strategy and the novel probability model can independently enhance the exploitation of ABC and accelerate the convergence speed. Moreover, MPGABC is better than GABC and MGABC on most cases, which means that the combination of the novel search strategy and the novel probability model can further enhance the exploitation ability of ABC. Note that some points are vacant in Fig. 5a, such as f_6 and f_8. The reason is that some algorithms cannot find an acceptable solution in 25 independent runs.

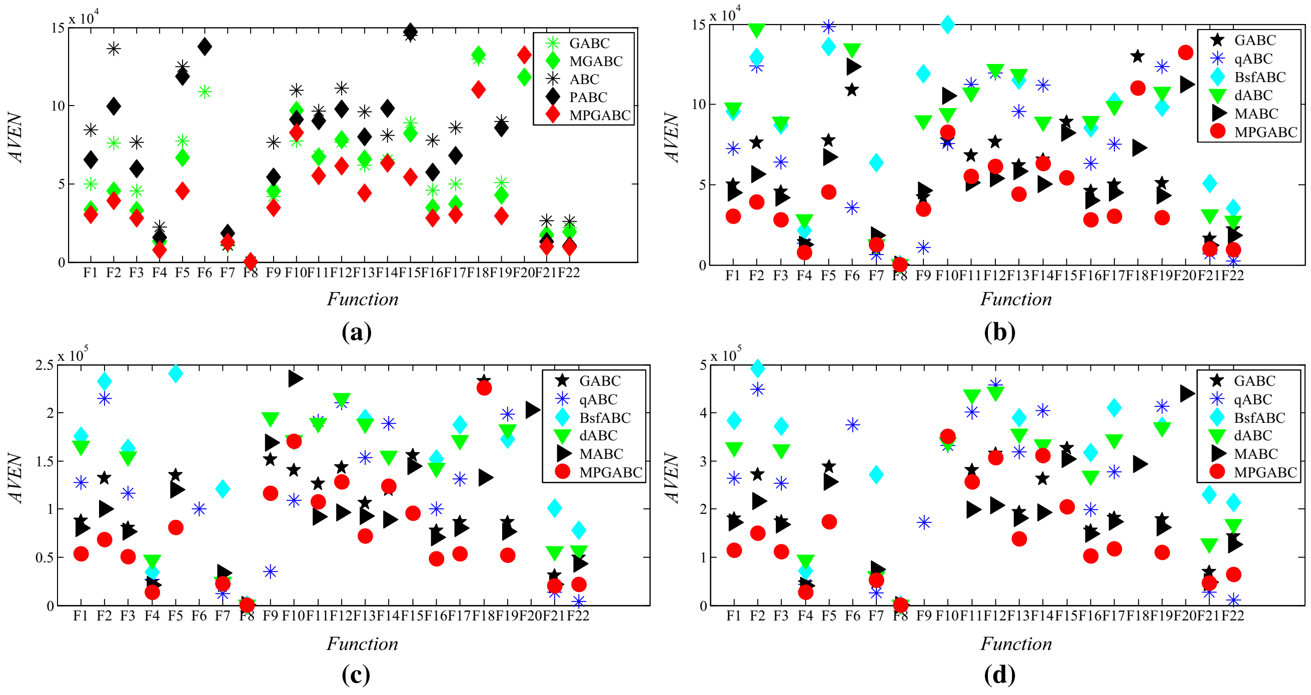


Fig. 5 Average FES are needed to reach the accept value (**a**), 30D (**b**), 50D (**c**) and 100D (**d**)

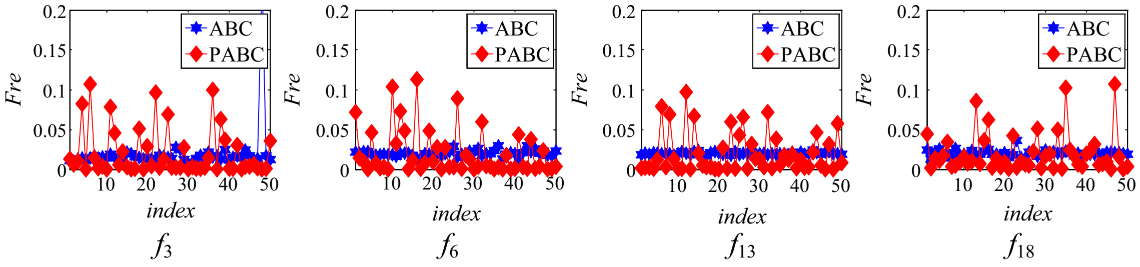


Fig. 6 The evolutionary frequency of each individual

Overall, it can be concluded that either the novel search strategy or the novel probability model can make a contribution to ABC. Moreover, the performance of ABC can be further enhanced by integrating these two proposed algorithmic components.

As shown in Table 2, PABC is better than ABC on most test functions especially in terms of convergence speed. The distinction between PABC and ABC is mainly caused by the novel probability model. In order to clearly show the difference between the novel probability model and the original probability model, the evolutionary frequency (the number of evolution/*maxFES*) of each individual in population on some selected representative functions with 30D (unimodal functions f_3, f_6, and multimodal functions f_{13}, f_{18}) is illustrated in Fig. 6, where X axis denotes the index of the individual, and Y axis denotes the evolutionary frequency (*Fre*). Note that the same initial population is employed by PABC and ABC on each function. Obviously, there is no significant difference between individuals' evolutionary frequencies in ABC, which means that the original probability model is unable to ensure the good food sources attract more onlooker bees to search. However, some frequencies are always significantly larger than others in PABC, which demonstrates that the novel probability model can make sure the good food sources can be selected to search by more onlooker bees. Therefore, the novel probability model can fully exploit the valuable information hidden in the good food sources and enhance the exploitation ability of ABC.

5.3 Experkipment 2: comparison on benchmark functions

In this experiment, in order to verify the performance of our proposed MGPABC, five state-of-the-art ABC variants (i.e., GABC, qABC, best-so-far ABC, dABC and MABC) are used to compare with MPGABC on 22 test functions with 30D, 50D and 100D. To make a fair comparison, for all compared algorithms, *SN* is set to 50, and *maxFES* is set to

Table 3 Parameters setting of all compared algorithms

Algorithm	Parameters setting
GABC	$SN = 50, limit = SN \cdot D, C = 1.5$
qABC	$SN = 50, limit = SN \cdot D, r = 1$
best-so-far ABC	$SN = 50, limit = SN \cdot D, \omega_{\max}= 1, \omega_{\min} = 0.2$
dABC	$SN = 50, limit = SN \cdot D$
MABC	$SN = 50, limit = SN \cdot D, P = 0.7$
MPGABC	$SN = 50, limit = SN \cdot D, P = 0.3$

150,000, 250,000 and 500,000 (Wang et al. 2014; Zhang et al. 2015) corresponding to $30D$, $50D$ and $100D$, respectively. The detailed parameter settings of all algorithms are given in Table 3, which are set as the same as the original papers. Each algorithm will be conducted 25 (Shan et al. 2015; Omidvar et al. 2014) independent runs for all test functions. In addition, the Wilcoxon's rank-sum test (Shi et al. 2010), which is a nonparametric statistic test for independent samples, is also used in experiment results at 5% significance level to show the significant differences between MPGABC and other algorithms. The detailed experimental results are given in Tables 4, 5 and 6, and the best results are marked with boldface. It is noted that the symbols "−", "+", "=" denote that the performance of the corresponding algorithm is worse than, better than and similar to that of MPGABC, respectively, according to Wilcoxon's rank-sum test at a 0.05 significance level.

Table 4 gives the experimental result on $30D$ functions. It can be clearly observed that MPGABC is significantly better than all compared algorithms in terms of solution accuracy and convnce rate on most test functions. To be specific, MPGABC outperforms all other algorithms on all unimodal functions ($f_1 - f_6$ and f_8) except f_6. Since the global optimal solution of Step function (f_7) is a region rather than a point, all algorithms can obtain the global optimal solution and show similar performance in terms of solution quality. But the convergence speed of MPGABC is better than all competitors except GABC and qABC. Moreover, MPGABC also outperforms or at least is competitive to all compared algorithms on Quartic function (f_9) and Rosenbrock function (f_{10}). Concerning multimodal functions $f_{11} - f_{22}$, MPGABC is better than or at least comparable to competitors on all functions excluding $f_{13} - f_{15}$, f_{18} and f_{20}. With respect to Griewank function f_{13}, MPGABC is beaten by qABC and best-so-far ABC, but it has better convergence rate than all compared algorithms. For Schwefel2.26 function f_{14}, MPGABC is superior to all competitors except GABC. On f_{15}, f_{18} and f_{20}, MPGABC is only beaten by MABC. Regarding to the remainder functions, MPGABC is better than or at least comparable to all compared algorithms. Overall, MPGABC outperforms GABC, qABC, best-so-far ABC (BsfABC), dABC and MABC on 9, 15, 18, 18 and 7 out of 22 functions. On the contrary, MPGABC is only beaten by GABC, qABC, best-so-far ABC (BsfABC) and MABC on 2, 2, 1, 1 and 4 functions, respectively. Furthermore, MPGABC is faster than all other competitors on a larger proportion of test functions (i.e., $f_1 - f_5$, f_8, f_{13}, $f_{15} - f_{17}$ and f_{19}). For the convenience and clearness of illustration, the convergence curves of mean objective function value for all functions are presented in Fig. 7, which clearly indicates that MPGABC has better solution accuracy and convergence rate than all other competitors on most test functions.

The results on functions with $50D$ and $100D$ are given in Tables 5 and 6, respectively, which also clearly show that MPGABC has better solution accuracy and convergence rate than all the competitors on most test functions. To be specific, regarding to $50D$ functions, MPGABC is better than GABC, qABC, best-so-far ABC(BsfABC), dABC and MABC on 14, 15, 18, 17 and 7 out of 22 functions, and MPGABC is only beaten by GABC, qABC, dABC and MABC on 1, 3, 1 and 4 functions, respectively. Concerning $100D$ functions, MPGABC is better than GABC, qABC, best-so-far ABC, dABC and MEABC on 13, 15, 19, 17 and 8 out of 22 functions, respectively, and MPGABC is only beaten by GABC, qABC, dABC and MEABC on 1, 4, 1 and 4 functions, respectively. Therefore, the superiority of MPGABC is not affected by the growth of the search space dimension. In order to clearly show the convergence rate of all compared algorithms, the AVEN of all compared algorithms on $30D$, $50D$ and $100D$ functions is, respectively, presented in Fig. 5b–d, which clearly illustrates that MGABC can obtain the best AVEN value on most functions.

In addition, according to the Friedman test, the final ranking of all ABC variants for each benchmark function with $30D$, $50D$ and $100D$ is shown in Table 7. The best results are marked in boldface. Evidently, the average ranking of MPGABC on all functions is better than that of all other ABC variants on $30D$ and $100D$. But MABC performs slightly better than MPGABC on $50D$. Therefore, it clearly shows that MPGABC is better than or at least competitive to all competitors on all the test functions.

Furthermore, we compare the average runtime between MPGABC and original ABC on 22 commonly benchmark functions with $30D$. MPGABC and ABC are conducted 25 independent runs on each function. The average runtime and Ratio are given in Table 8, where Ratio denotes the value that the average runtime of MPGABC is divided by that of original ABC. As shown in Table 8, the Ratio values are prominently less than 1 for all functions. This phenomenon indicates the average runtime of MPGABC is significantly less than that of original ABC on all cases. The reason is that in original ABC, the selection probabilities of all food source positions are similar and small, which means it is a time-consuming procedure that the onlooker bee selects a food source position to search by roulette wheel method. While in MPGABC, the selection probabilities of some good

Table 4 Comparisons of MPGABC with ABC variants on 22 test functions with 30D

Alg	Metric	GABC	qABC	best-so-far ABC	dABC	MABC	MPGABC
f_1	Mean (SD)	6.97e−33 (4.93e−33)−	1.60e−15 (1.32e−15)−	4.15e−17 (4.73e−17)−	4.67e−13 (5.20e−13)−	8.00e−40(5.33e−40)−	**5.41e−55** (**1.95e−54**)
	SR/AVEN	**100**/50130	**100**/72618	**100**/95566	**100**/97730	**100**/45207	**100/30494**
f_2	Mean (SD)	1.92e−26 (2.12e−26)−	1.53e−10 (3.87e−10)−	6.09e−11 (1.32e−10)−	1.12e−07 (2.67e−07)−	9.77e−37(7.36e−37)−	**5.24e−50** (**2.20e−49**)
	SR/AVEN	**100**/76150	**100**/123870	**100**/129190	16/147200	**100**/56455	**100/39274**
f_3	Mean (SD)	2.98e−34 (2.38e−34)−	3.14e−16 (2.92e−16)−	7.48e−19 (7.38e−19)−	1.38e−14 (9.85e−15)−	5.37e−41(4.78e−41)−	**1.89e−55** (**4.07e−55**)
	SR/AVEN	**100**/45478	**100**/63946	**100**/87102	**100**/88830	**100**/41773	**100/28214**
f_4	Mean (SD)	1.83e−52 (6.33e−52)−	3.01e−21 (1.31e−20)−	5.60e−31 (1.49e−30)−	4.21e−26 (1.15e−25)−	5.76e−85 (2.68e−84)−	**1.40e−96** (**6.98e−96**)
	SR/AVEN	**100**/14106	**100**/13342	**100**/21454	**100**/28050	**100**/12482	**100/7922**
f_5	Mean (SD)	5.95e−18 (1.76e−18)−	1.09e−08 (3.89e−09)−	5.84e−10 (2.31e−10)−	3.93e−08 (1.22e−08)−	1.82e−21(6.40e−22)−	**2.60e−29** (**8.89e−29**)
	SR/AVEN	**100**/77478	48/148340	**100**/135890	0/NAN	**100**/67210	**100/45478**
f_6	Mean (SD)	2.55e−01 (1.30e−01)+	**9.36e−02** (**1.79e−02**)+	1.55e+01 (9.40e−01)−	1.19e+00 (5.84e−01)+	2.38e+00(1.75e+00)+	9.87e+00 (3.32e+00)
	SR/AVEN	**100**/109060	**100/35898**	0/NAN	48/134830	24/123490	0/NAN
f_7	Mean (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**)
	SR/AVEN	**100**/10314	**100/6482**	**100**/63562	**100**/12634	**100**/18222	**100**/12530
f_8	Mean (SD)	**7.18e−66** (9.22e−77)=	**7.18e−66** (2.98e−72)=	**7.18e−66** (7.378e−71)=	**7.18e−66** (9.80e−71)=	**7.18e−66** (**1.00e−79**)=	**7.18e−66** (4.02e−79)
	SR/AVEN	**100/150**	**100/150**	**100/150**	**100/150**	**100**/166	**100/150**
f_9	Mean (SD)	2.80e−02 (6.51e−03)−	2.78e−02 (8.01e−03)=	8.40e−02 (1.62e−02)−	6.01e−02 (1.62e−02)−	2.99e−02(**5.82e−03**)−	**2.46e−02** (6.43e−03)
	SR/AVEN	**100**/41966	**100/11018**	80/118955	**100**/90014	**100**/46280	**100**/34834
f_{10}	Mean (SD)	8.21e−01 (3.73e+00)=	5.56e−01 (6.12e−01)−	4.90e−01 (3.55e−01)−	1.54e−01 (2.03e−01)−	1.93e+00(5.80e+00)−	**7.43e−02** (**1.82e−01**)
	SR/AVEN	68/77515	36/**75828**	4/NAN	52/94496	4/105350	**84**/82945
f_{11}	Mean (SD)	**0.00e+00** (**0.00e+00**)=	1.23e−10 (1.68e−10)−	2.05e+00 (6.77e−01)−	8.31e−13 (1.61e−12)−	**0.00e+00** (**0.00e+00**)=	**0.00e+00** (**0.00e+00**)
	SR/AVEN	**100**/68134	**100**/112510	0/NAN	**100**/107040	**100/51449**	**100**/55254
f_{12}	Mean (SD)	**0.00e+00** (**0.00e+00**)=	4.95e−10 (5.78e−10)−	2.93e+00 (9.50e−01)−	4.34e−11 (1.54e−10)−	**0.00e+00** (**0.00e+00**)=	**0.00e+00** (**0.00e+00**)
	SR/AVEN	**100**/76642	**100**/119730	0/NAN	**100**/121690	**100/53889**	**100**/61378
f_{13}	Mean (SD)	4.51e−08 (2.25e−07)=	2.48e−12 (6.35e−12)+	5.27e−11 (2.32e−10)+	3.06e−04(1.53e−03)−	**5.33e−17** (**2.66e−16**)=	2.11e−09 (1.06e−08)
	SR/AVEN	96/61688	**100**/95790	**100**/115254	96/118580	**100**/58453	96/**44275**
f_{14}	Mean (SD)	**2.18e−13** (6.03e−13)+	3.88e−10 (1.46e−09)−	4.13e+01 (5.17e+01)−	9.31e−12 (4.31e−12)−	3.20e−12 (9.51e−13)=	3.49e−12 (**5.04e−13**)
	SR/AVEN	**100**/65670	**100**/112170	0/NAN	**100**/88810	**100/50247**	**100**/63062

Table 4 continued

Alg	Metric	GABC	qABC	best-so-far ABC	dABC	MABC	MPGABC
f_{15}	Mean (SD)	1.49e−14 (2.92e−15)=	1.61e−06 (8.36e−07)−	4.61e−05 (2.75e−05)−	4.33e−07 (1.97e−07)−	**6.50e−15** (**1.42e−15**)+	1.42e−14 (3.59e−15)
	SR/AVEN	**100**/89178	0/NAN	0/NAN	0/NAN	**100**/82295	**100/54498**
f_{16}	Mean (SD)	**1.57e−32** (**5.59e−48**)=	4.12e−15 (7.77e−15)−	4.59e−19 (4.81e−19)−	6.68e−14 (7.59e−14)−	**1.57e−32** (**5.59e−48**)=	**1.57e−32** (**5.59e−48**)
	SR/AVEN	**100**/45786	**100**/63282	**100**/85358	**100**/89342	**100**/40270	**100/28018**
f_{17}	Mean (SD)	4.06e−33 (2.30e−33)=	1.83e−15 (1.51e−15)−	2.23e−16 (3.45e−16)−	1.54e−13 (1.25e−13)−	**1.50e−33** (**0.00e+00**)=	**1.50e−33** (**0.00e+00**)
	SR/AVEN	**100**/49750	**100**/75322	**100**/101674	**100**/98950	**100**/45073	**100/30430**
f_{18}	Mean (SD)	3.88e−07 (6.54e−07)−	1.43e−05 (3.92e−05)−	2.73e−04 (1.95e−04)−	8.52e−06 (5.36e−06)−	**2.88e−21** (**6.36e−21**)+	1.83e−08 (4.32e−08)
	SR/AVEN	16/129980	0/NAN	0/NAN	0/NAN	**100/73176**	76/110410
f_{19}	Mean (SD)	1.39e−31 (1.41e−32)=	9.29e−10 (9.59e−10)−	4.33e−15 (6.19e−15)−	1.68e−11 (3.02e−11)−	**1.35e−31** (**2.23e−47**)=	**1.35e−31** (**2.23e−47**)
	SR/AVEN	**100**/50934	**100**/123530	**100**/98418	**100**/107700	**100**/43306	**100/29430**
f_{20}	Mean (SD)	3.60e−02 (4.19e−02)−	8.71e−03 (8.44e−03)−	2.09e−01 (7.17e−02)−	3.00e−02 (3.79e−02)−	**0.00e+00** (**0.00e+00**)+	1.16e−03 (2.15e−03)
	SR/AVEN	0/NAN	0/NAN	0/NAN	0/NAN	**100/112440**	28/132380
f_{21}	Mean (SD)	−**7.833e+01** (5.02e-15)=	−**7.833e+01** (7.11e−15)=	−**7.833e+01** (6.12e−10)=	−**7.833e+01** (**0.00e+00**)=	−**7.833e+01** (5.80e−15)=	−**7.833e+01** (**2.90e−15**)
	SR/AVEN	**100**/15986	**100/6838**	**100**/50858	**100**/31282	**100**/12122	**100**/9774
f_{22}	Mean (SD)	−2.999e+01 (1.01e−03)−	−**3.000e+01** (1.12e−05)=	−2.999e+01 (1.08e−03)−	−2.999e+01 (1.11e−03)−	−**3.000e+01** (0.00e+00)=	−**3.000e+01** (9.51e−05)
	SR/AVEN	**100**/21778	**100/2310**	**100**/35170	**100**/27174	**100**/18516	**100**/9586
	+/=/−	2/11/9	2/5/15	1/3/18	1/3/18	4/11/7	

Table 5 Comparisons of MPGABC with ABC variants on 22 test functions with 50D

Alg	Metric	GABC	qABC	best-so-far ABC	dABC	MABC	MPGABC
f_1	Mean (SD)	1.03e−31 (7.56e−32)−	3.52e−15 (4.37e−15)−	1.08e−15 (9.47e−16)−	1.62e−12 (1.09e−12)−	4.64e−38 (3.73e−38)−	**2.47e−51** (**9.52e−51**)
	SR/AVEN	**100**/87354	**100**/127490	**100**/175546	**100**/165250	**100**/80425	**100/53526**
f_2	Mean (SD)	2.74e−25 (2.52e−25)−	4.44e−10 (7.92e−10)−	1.30e−09 (2.06e−09)−	2.70e−07 (4.96e−07)−	8.45e−35 (5.83e−35)−	**4.54e−49** (**1.32e−48**)
	SR/AVEN	**100**/132310	**100**/214880	**100**/232746	0/NAN	**100**/100270	**100/68698**
f_3	Mean (SD)	1.64e−32 (1.97e−32)−	1.11e−15 (1.09e−15)−	5.52e−17(6.49e−17)−	2.02e−13 (2.22e−13)−	9.74e−39 (4.80e−39)−	**6.05e−53** (**1.39e−52**)
	SR/AVEN	**100**/80426	**100**/116460	**100**/162730	**100**/154570	**100**/76558	**100/50754**
f_4	Mean (SD)	4.31e−54 (1.12e−53)−	4.57e−22 (8.36e−22)−	7.88e−31(2.14e−30)−	4.19e−26 (9.38e−26)−	1.13e−86 (3.86e−86)−	**8.51e−103** (**3.61e−102**)
	SR/AVEN	**100**/23742	**100**/27010	**100**/34346	**100**/47178	**100**/21152	**100/13562**
f_5	Mean (SD)	3.01e−17 (7.96e−18)−	1.91e−08 (4.64e−09)−	3.95e−09(1.17e−09)−	1.54e−07 (5.42e−08)−	2.86e−20 (5.74e−21)−	**2.95e−28** (**4.04e−28**)
	SR/AVEN	**100**/135020	0/NAN	**100**/240782	0/NAN	**100**/119870	**100/80894**
f_6	Mean (SD)	4.20e+00 (1.36e+00)−	**2.12e−01** (**3.38e−02**)+	2.63e+01(1.96e+00)−	4.20e+00 (9.01e−01)+	1.91e+01 (5.42e+00)=	2.12e+01 (4.04e+00)
	SR/AVEN	0/NAN	**100/100470**	0/NAN	0/NAN	0/NAN	0/NAN
f_7	Mean (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**)
	SR/AVEN	**100**/20146	**100/12530**	**100**/121010	**100**/24322	**100**/33424	**100**/22738
f_8	Mean (SD)	**2.67e−109** (1.65e−119)=	**2.67e−109** (1.84e−115)=	**2.67e−109** (4.25e−114)=	**2.67e−109** (4.82e−114)=	**2.67e−109** (**2.66e−123**)=	**2.67e−109** (3.06e−122)
	SR/AVEN	**100/150**	**100/150**	**100/150**	**100/150**	**100**/166	**100/150**
f_9	Mean (SD)	6.36e−02 (9.37e−03)−	**4.55e−02** (1.17e−02)+	1.93e−01(2.37e−02)−	1.30e−01 (2.98e−02)−	6.68e−02 (1.14e−02)−	5.02e−02 (**7.98e−03**)
	SR/AVEN	0/NAN	**100/34834**	0/NAN	20/195050	**100**/169040	**100**/116710
f_{10}	Mean (SD)	3.22e+00 (1.55e+01)=	6.83e−01 (6.17e−01)=	2.22e+00 (1.14e+00)−	**1.83e−01** (**1.77e−01**)−	1.35e+00 (2.20e+00)−	4.32e−01 (7.44e−01)
	SR/AVEN	**72**/140110	20/**108850**	0/NAN	44/171060	4/235690	52/170460
f_{11}	Mean (SD)	**0.00e+00** (**0.00e+00**)=	1.30e−10 (1.61e−10)−	4.96e+00(9.00e−01)−	2.76e−10 (1.38e−09)−	**0.00e+00** (0.00e+00)=	**0.00e+00** (**0.00e+00**)
	SR/AVEN	**100**/126330	**100**/191110	0/NAN	**100**/188710	**100/91876**	**100**/107620
f_{12}	Mean (SD)	**0.00e+00** (**0.00e+00**)=	9.17e−10 (9.55e−10)−	6.96e+00(1.04e+00)−	2.21e−10 (7.45e−10)−	**0.00e+00** (**0.00e+00**)=	**0.00e+00** (**0.00e+00**)
	SR/AVEN	**100**/142890	**100**/210650	0/NAN	**100**/215100	**100/96297**	**100**/128670
f_{13}	Mean (SD)	1.61e−10 (7.97e−10)−	9.79e−12 (4.64e−11)−	1.15e−11(5.63e−11)−	5.14e−12 (1.08e−11)−	**0.00e+00** (0.00e+00)=	**0.00e+00** (**0.00e+00**)
	SR/AVEN	**100**/106000	**100**/153360	**100**/194198	**100**/188460	**100**/92773	**100/72082**

Table 5 continued

Alg	Metric	GABC	qABC	best-so-far ABC	dABC	MABC	MPGABC
f_{14}	Mean (SD)	**7.42e−12** (3.56e−12)+	2.93e−10 (7.37e−10)−	2.13e+02(7.06e+01)−	1.99e−11 (3.50e−12)−	9.90e−12 (2.47e−12)=	1.05e−11 (**3.03e−12**)
	SR/AVEN	**100**/120530	**100**/189230	0/NAN	**100**/154880	**100/89407**	**100**/123880
f_{15}	Mean (SD)	3.61e−14 (3.97e−15)−	1.86e−06 (7.88e−07)−	3.69e−04 (3.26e−04)−	6.30e−07 (3.28e−07)−	**1.35e−14** (**1.91e−15**)+	2.51e−14 (4.55e−15)
	SR/AVEN	**100**/155340	0/NAN	0/NAN	0/NAN	**100**/144610	**100/95558**
f_{16}	Mean (SD)	1.18e−32 (2.60e−33)−	1.84e−15 (5.74e−15)−	7.89e−18 (8.38e−18)−	1.22e−13 (1.09e−13)−	**9.42e−33** (**1.40e−48**)=	**9.42e−33** (**1.40e−48**)
	SR/AVEN	**100**/77058	**100**/100420	**100**/152122	**100**/142050	**100**/70606	**100/48166**
f_{17}	Mean (SD)	6.47e−32 (3.17e−32)−	3.47e−15 (3.28e−15)−	9.48e−15(8.32e−15)−	8.52e−13 (8.53e−13)−	**1.50e−33** (**0.00e+00**)=	**1.50e−33** (**0.00e+00**)
	SR/AVEN	**100**/86354	**100**/131410	**100**/187542	**100**/171560	**100**/80610	**100/53730**
f_{18}	Mean (SD)	4.08e−06 (5.59e−06)−	1.47e−05 (4.51e−05)−	6.92e−03(3.87e−03)−	2.05e−05 (1.09e−05)−	**7.63e−20** (**5.93e−20**)+	3.21e−07 (4.58e−07)
	SR/AVEN	4/232250	0/NAN	0/NAN	0/NAN	**100/132490**	16/225930
f_{19}	Mean (SD)	1.75e−31 (4.95e−32)−	1.20e−09 (1.32e−09)−	1.70e−14(1.77e−14)−	1.50e−11 (2.02e−11)−	**1.35e−31** (**2.23e−47**)=	**1.35e−31** (**2.23e−47**)
	SR/AVEN	**100**/86402	**100**/198230	**100**/172718	**100**/182140	**100**/76393	**100/51914**
f_{20}	Mean (SD)	1.08e−01 (9.20e−02)−	2.90e−02 (1.81e−02)−	5.29e−01(1.16e−01)−	1.48e−01 (5.18e−02)−	**1.48e−14** (**4.27e−14**)+	1.56e−02 (2.04e−02)
	SR/AVEN	0/NAN	0/NAN	0/NAN	0/NAN	**100/203000**	0/NAN
f_{21}	Mean (SD)	**−7.833e+01** (7.11e−15)=	**−7.833e+01** (1.48e−14)=	**−7.833e+01** (1.45e−09)=	**−7.833e+01** (1.00e−14)=	**−7.833e+01** (**5.02e−15**)=	**−7.833e+01** (1.12e−14)
	SR/AVEN	**100**/30414	**100/13546**	**100**/100606	**100**/56126	**100**/21761	**100**/20514
f_{22}	Mean (SD)	−4.999e+01 (3.55e−03)=	**−5.000e+01** (2.67e−05)+	−4.999e+01 (2.66e−03)=	−4.999e+01 (1.82e−03)=	**−5.000e+01** (**7.40e−15**)+	−4.999e+01 (4.26e−04)
	SR/AVEN	**100**/48890	**100/3962**	**100**/78230	**100**/56538	**100**/43107	**100**/22182
	+/=/−	1/7/14	3/4/15	0/4/18	1/4/17	4/11/7	

Table 6 Comparisons of MPGABC with ABC variants on 22 test functions with 100D

Alg	Metric	GABC	qABC	best-so-far ABC	dABC	MABC	MPGABC
f_1	Mean (SD)	1.91e−30 (1.31e−30)−	7.20e−15 (4.39e−15)−	7.08e−14 (7.30e−14)−	4.78e−12 (6.96e−12)−	4.93e−36 (3.44e−36)−	**9.45**e−50 (**4.22e−49**)
	SR/AVEN	**100**/179330	**100**/264770	**100**/385830	**100**/327190	**100**/172670	**100/115300**
f_2	Mean (SD)	7.05e−24 (4.16e−24)−	5.81e−10 (7.44e−10)−	1.56e−07 (2.35e−07)−	1.43e−07 (2.01e−07)−	1.10e−32 (6.35e−33)−	**1.76e−45** (**7.71e−45**)
	SR/AVEN	**100**/272670	**100**/444430	32/484140	0/NAN	**100**/215990	**100/149510**
f_3	Mean (SD)	6.89e−31 (3.77e−31)−	3.32e−15 (2.06e−15)−	1.70e−14 (1.34e−14)−	1.85e−12 (1.26e−12)−	2.05e−36 (8.62e−37)−	**7.34e−50** (**2.51e−49**)
	SR/AVEN	**100**/174180	**100**/254010	**100**/372820	**100**/323350	**100**/168360	**100/111520**
f_4	Mean (SD)	6.86e−52 (3.03e−51)−	2.90e−22 (4.75e−22)−	2.97e−30 (7.91e−30)−	1.27e−25 (4.30e−25)−	3.98e−87 (1.68e−86)−	**1.31e−98** (**6.55e−98**)
	SR/AVEN	**100**/46478	**100**/54466	**100**/68822	**100**/94274	**100**/40010	**100/26786**
f_5	Mean (SD)	2.29e−16 (4.99e−17)−	4.18e−08 (7.30e−09)−	3.80e−08 (8.59e−09)−	4.89e−07 (1.51e−07)−	4.17e−19 (1.01e−19)−	**7.37e−27** (**1.19e−26**)
	SR/AVEN	**100**/288050	0/NAN	0/NAN	**100**/NAN	**100**/257170	**100/173740**
f_6	Mean (SD)	2.22e+01 (3.30e+00)+	**6.63e−01** (**6.30e−02**)+	4.10e+01 (1.57e+00)−	1.02e+01 (1.04e+00)=	4.06e+01 (3.60e+00)−	3.81e+01 (2.44e+00)
	SR/AVEN	0/NAN	**100/383500**	0/NAN	0/NAN	0/NAN	0/NAN
f_7	Mean (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**)
	SR/AVEN	**100**/46166	**100/27174**	**100**/270680	**100**/59858	**100**/74227	**100**/52222
f_8	Mean (SD)	**7.12e−218** (**0.00e+00**)=	**7.12e−218** (**0.00e+00**)=	7.13e−218 (**0.00e+00**)=	7.13e−218 (**0.00e+00**)=	**7.12e−218** (**0.00e+00**)=	**7.12e−218** (**0.00e+00**)
	SR/AVEN	**100/150**	**100/150**	**100/150**	**100/150**	**100**/166	**100/150**
f_9	Mean (SD)	1.64e−01 (1.69e−02)−	**6.49e−02** (1.61e−02)+	5.54e−01 (5.12e−02)−	2.91e−01 (5.88e−02)−	1.81e−01 (2.22e−02)−	1.24e−01 (**1.05e−02**)
	SR/AVEN	0/NAN	**100/199660**	0/NAN	0/NAN	0/NAN	0/NAN
f_{10}	Mean (SD)	1.04e+01 (2.61e+01)=	5.39e−01 (4.96e−01)−	1.27e+01 (2.64e+00)−	3.20e−01 (5.74e−01)−	1.63e+01 (2.44e+01)−	**1.73e−01** (**3.22e−01**)
	SR/AVEN	24/346470	16/355150	0/NAN	40/**337740**	0/NAN	**56**/350970
f_{11}	Mean (SD)	**0.00e+00** (**0.00e+00**)=	3.23e−10 (2.59e−10)−	1.33e+01(1.76e+00)−	8.22e−08 (4.09e−07)−	**0.00e+00** (**0.00e+00**)=	**0.00e+00** (**0.00e+00**)
	SR/AVEN	**100**/278790	**100**/401810	0/NAN	96/437840	**100/198330**	**100**/256250
f_{12}	Mean (SD)	**0.00e+00** (**0.00e+00**)=	1.61e−09 (1.12e−09)−	1.90e+01(1.88e+00)−	2.66e−01 (5.69e−01)−	**0.00e+00**(**0.00e+00**)=	**0.00e+00** (**0.00e+00**)
	SR/AVEN	**100**/309930	**100**/453910	0/NAN	52/443370	**100/207730**	**100**/307440
f_{13}	Mean (SD)	8.88e−18 (4.44e−17)=	2.35e−13 (6.14e−13)−	1.26e−13 (2.02e−13)−	8.10e−13 (1.12e−12)−	**0.00e+00** (**0.00e+00**)=	**0.00e+00** (**0.00e+00**)
	SR/AVEN	**100**/199410	**100**/305970	**100**/388990	**100**/356100	**100**/181160	**100/138210**

Table 6 continued

Alg	Metric	GABC	qABC	Best-so-far ABC	dABC	MABC	MPGABC
f_{14}	Mean (SD)	**2.04e−11** (4.20e−12)=	6.51e−10 (1.07e−09)−	7.06e+02 (1.54e+02)−	4.69e−11 (7.60e−12)+	3.64e−11 (**0.00e+00**)=	5.94e−11 (6.86e−12)
	SR/AVEN	**100**/262100	**100**/413690	0/NAN	**100**/333190	**100/192730**	**100**/310940
f_{15}	Mean (SD)	1.06e−13 (7.40e−15)−	1.97e−06 (4.82e−07)−	2.47e−03 (1.35e−03)−	1.65e-06 (7.69e−07)−	**3.46e−14** (**3.08e−15**)+	3.89e−14 (5.02e−15)
	SR/AVEN	**100**/327300	0/NAN	0/NAN	0/NAN	**100**/303580	**100/205370**
f_{16}	Mean (SD)	3.91e−32 (1.98e−32)−	6.34e−16 (1.01e−15)−	8.26e−17 (1.14e−16)−	2.17e−13 (2.05e−13)−	**4.71e**−33 (**6.98e−49**)=	**4.71e−33** (**6.98e−49**)
	SR/AVEN	**100**/155580	**100**/204780	**100**/321790	**100**/267930	**100**/148610	**100/102370**
f_{17}	Mean (SD)	1.13e−30 (6.61e−31)−	6.59e−15 (8.25e−15)−	5.47e−13 (4.11e−13)−	2.46e−12(2.56e−12)−	**1.50e-33** (**0.00e+00**)=	**1.50e−33** (**0.00e+00**)
	SR/AVEN	**100**/180510	**100**/280640	**100**/410220	**100**/344380	**100**/173590	**100/117090**
f_{18}	Mean (SD)	7.03e−05 (7.90e−05)−	4.82e−05 (7.35e−05)−	5.34e−02 (1.47e−02)−	3.58e−04 (3.45e−04)−	**6.02e−18** (**5.56e−18**)+	6.85e−06 (6.59e−06)
	SR/AVEN	0/NAN	0/NAN	0/NAN	0/NAN	**100/294360**	0/NAN
f_{19}	Mean (SD)	6.94e−31 (4.97e−31)−	1.19e−09 (1.43e−09)−	1.32e−13 (1.92e−13)−	2.59e−11 (4.04e−11)−	**1.35e−31** (**2.23e−47**)=	**1.35e−31** (**2.23e−47**)
	SR/AVEN	**100**/180620	**100**/383390	**100**/370530	**100**/369030	**100**/161280	**100/109890**
f_{20}	Mean (SD)	2.32e−01 (1.42e−01)−	5.87e−02 (2.48e−02)+	1.48e+00 (1.82e−01)−	5.09e−01 (1.82e−01)−	**1.08e−12** (**1.27e−12**)+	1.10e−01 (6.48e−02)
	SR/AVEN	0/NAN	0/NAN	0/NAN	0/NAN	**100/439450**	0/NAN
f_{21}	Mean (SD)	**−7.833e+01** (9.17e−15)=	**−7.833e+01** (3.49e−14)=	**−7.833e+01** (1.69e−08)=	**−7.833e+01** (1.88e−14)=	**−7.833e+01** (**5.02e−15**)=	**−7.833e+01** (3.80e−14)
	SR/AVEN	**100**/68702	**100/27854**	**100**/221890	**100**/128100	**100**/47671	**100**/46570
f_{22}	Mean (SD)	−9.996e+01 (6.58e−03)−	**−1.00e+02** (**7.57e−05**)+	−9.996e+01 (8.19e−03)−	−9.996e+01 (6.85e−03)−	**−1.000e+02** (1.51e−04)+	−9.999e+01 (1.78e−03)
	SR/AVEN	**100**/146990	**100/10582**	**100**/211420	**100**/167440	**100**/126170	**100**/64878
	+/=/−	1/8/13	4/3/15	0/3/19	1/4/17	4/10/8	

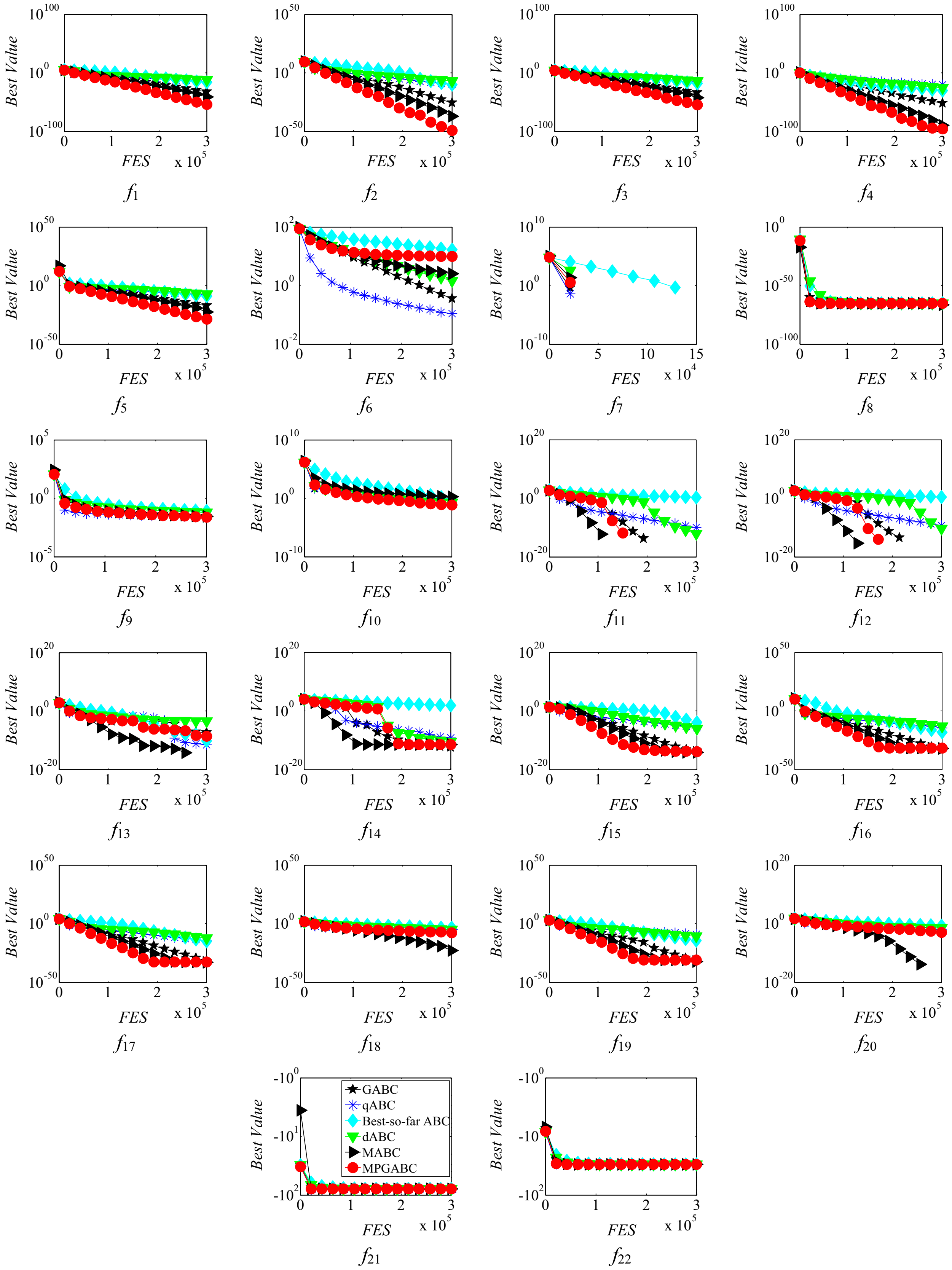


Fig. 7 Convergence curve of different ABCs on 22 test functions with $D = 30$

Table 7 The rankings of competitors for each problem based on the Friedman test

Fun	Alg																	
	30D						50D						100D					
	GABC	qABC	BsfABC	dABC	MABC	MPGABC	GABC	qABC	BsfABC	dABC	MABC	MPGABC	GABC	qABC	BsfABC	dABC	MABC	MPGABC
f_1	3.00	5.00	4.00	6.00	2.00	**1.00**	3.00	4.88	4.12	6.00	2.00	**1.00**	3.00	4.04	4.96	6.00	2.00	**1.00**
f_2	3.00	4.68	4.32	6.00	2.00	**1.00**	3.00	4.40	4.60	6.00	2.00	**1.00**	3.00	4.00	5.56	5.44	2.00	**1.00**
f_3	3.00	5.00	4.00	6.00	2.00	**1.00**	3.00	5.00	4.00	6.00	2.00	**1.00**	3.00	4.12	4.88	6.00	2.00	**1.00**
f_4	3.00	6.00	4.00	5.00	2.00	**1.00**	3.00	6.00	4.00	5.00	2.00	**1.00**	3.00	6.00	4.04	4.96	2.00	**1.00**
f_5	3.00	5.00	4.00	6.00	2.00	**1.00**	3.00	5.00	4.00	6.00	2.00	**1.00**	3.00	4.56	4.44	6.00	2.00	**1.00**
f_6	1.96	**1.08**	5.96	3.28	3.72	5.00	2.48	**1.00**	5.84	2.52	4.48	4.68	3.00	**1.00**	5.20	2.00	5.36	4.44
f_7	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**	**3.50**
f_8	3.00	4.00	5.04	5.96	**1.00**	2.00	3.00	4.00	5.00	6.00	**1.00**	2.00	3.00	4.00	5.00	6.00	**1.00**	2.00
f_9	2.68	2.64	5.92	4.92	2.88	**1.96**	3.16	**1.60**	5.96	5.04	3.28	1.96	3.32	**1.00**	600	4.88	3.80	2.00
f_{10}	2.56	4.00	4.72	2.80	4.96	**1.96**	**2.28**	3.64	5.52	2.68	4.12	2.76	2.76	3.16	5.60	2.28	5.08	**2.12**
f_{11}	**2.00**	5.00	6.00	4.00	**2.00**	**2.00**	**2.00**	4.96	6.00	4.04	**2.00**	**2.00**	**2.00**	4.84	6.00	4.16	**2.00**	**2.00**
f_{12}	**2.00**	4.96	6.00	4.04	**2.00**	**2.00**	**2.00**	4.88	6.00	4.12	**2.00**	**2.00**	**2.00**	4.52	6.00	4.48	**2.00**	**2.00**
f_{13}	2.12	4.68	4.44	5.64	**2.00**	2.12	2.32	4.92	4.40	5.44	**1.96**	**1.96**	**2.00**	4.48	4.72	5.80	**2.00**	**2.00**
f_{14}	**1.04**	5.00	6.00	3.92	2.44	2.60	**1.60**	5.00	6.00	3.92	2.16	2.32	**1.10**	5.00	6.00	2.98	2.06	3.86
f_{15}	2.54	4.96	6.00	4.04	**1.06**	2.40	2.98	4.96	6.00	4.04	**1.00**	2.02	3.00	4.84	6.00	4.16	**1.22**	1.78
f_{16}	**2.00**	5.08	4.00	5.92	**2.00**	**2.00**	2.72	5.00	4.00	6.00	**1.64**	**1.64**	3.00	4.92	4.08	6.00	**1.50**	**1.50**
f_{17}	2.72	4.96	4.04	6.00	**1.64**	**1.64**	3.00	4.12	4.88	6.00	**1.50**	**1.50**	3.00	4.00	5.04	5.96	**1.50**	**1.50**
f_{18}	3.08	3.96	6.00	4.84	**1.00**	2.12	3.36	3.32	6.00	4.92	**1.00**	2.40	3.68	2.88	6.00	4.84	**1.00**	2.60
f_{19}	2.08	5.96	4.00	5.04	**1.96**	**1.96**	2.76	6.00	4.00	5.00	**1.62**	**1.62**	3.00	5.96	4.00	5.04	**1.50**	**1.50**
f_{20}	4.12	3.68	6.00	3.76	**1.04**	2.40	4.04	3.08	6.00	4.56	**1.00**	2.32	4.00	2.36	6.00	4.68	**1.00**	2.96
f_{21}	2.76	3.60	6.00	3.04	**2.66**	2.94	**2.16**	4.20	6.00	3.10	2.20	3.34	1.64	4.04	6.00	4.96	**1.42**	2.94
f_{22}	4.72	2.12	5.72	4.56	**1.00**	2.88	5.08	2.00	5.44	4.48	**1.00**	3.00	4.88	1.56	5.24	4.88	**1.44**	3.00
Avg	2.72	4.31	4.98	4.74	2.13	**2.11**	2.88	4.16	5.06	4.74	**2.07**	2.09	2.90	3.85	5.19	4.77	2.15	**2.12**

Table 8 Average runtime (in seconds) used by ABC and MPGABC on 22 functions with D=30

Fun	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}
ABC	35.24	36.01	35.59	36.25	36.07	36.81	37.39	36.61	37.67	36.90	37.00
MPGABC	7.24	8.23	7.80	8.47	8.12	7.72	7.98	7.58	8.86	7.90	8.24
Ratio	0.2055	0.2286	0.2192	0.2337	0.2251	0.2097	0.2134	0.2071	0.2352	0.2141	0.2227
Fun	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}	f_{21}	f_{22}
ABC	38.34	37.28	37.08	36.34	40.10	35.90	35.62	36.13	78.88	36.78	36.73
MPGABC	9.31	8.59	8.63	9.18	13.02	8.79	8.41	8.84	52.06	10.20	10.14
Ratio	0.2428	0.2304	0.2327	0.2526	0.3247	0.2449	0.2361	0.2447	0.6600	0.2773	0.2761

food source positions are significantly enlarged by the novel probability model, and therefore, the time of the onlooker bee choosing a food source position to search by roulette wheel method is significantly less than that of ABC. Overall, the average runtime of MPGABC is better than that of ABC.

5.4 Experiment 3: sensitiveness analysis of parameter *P*

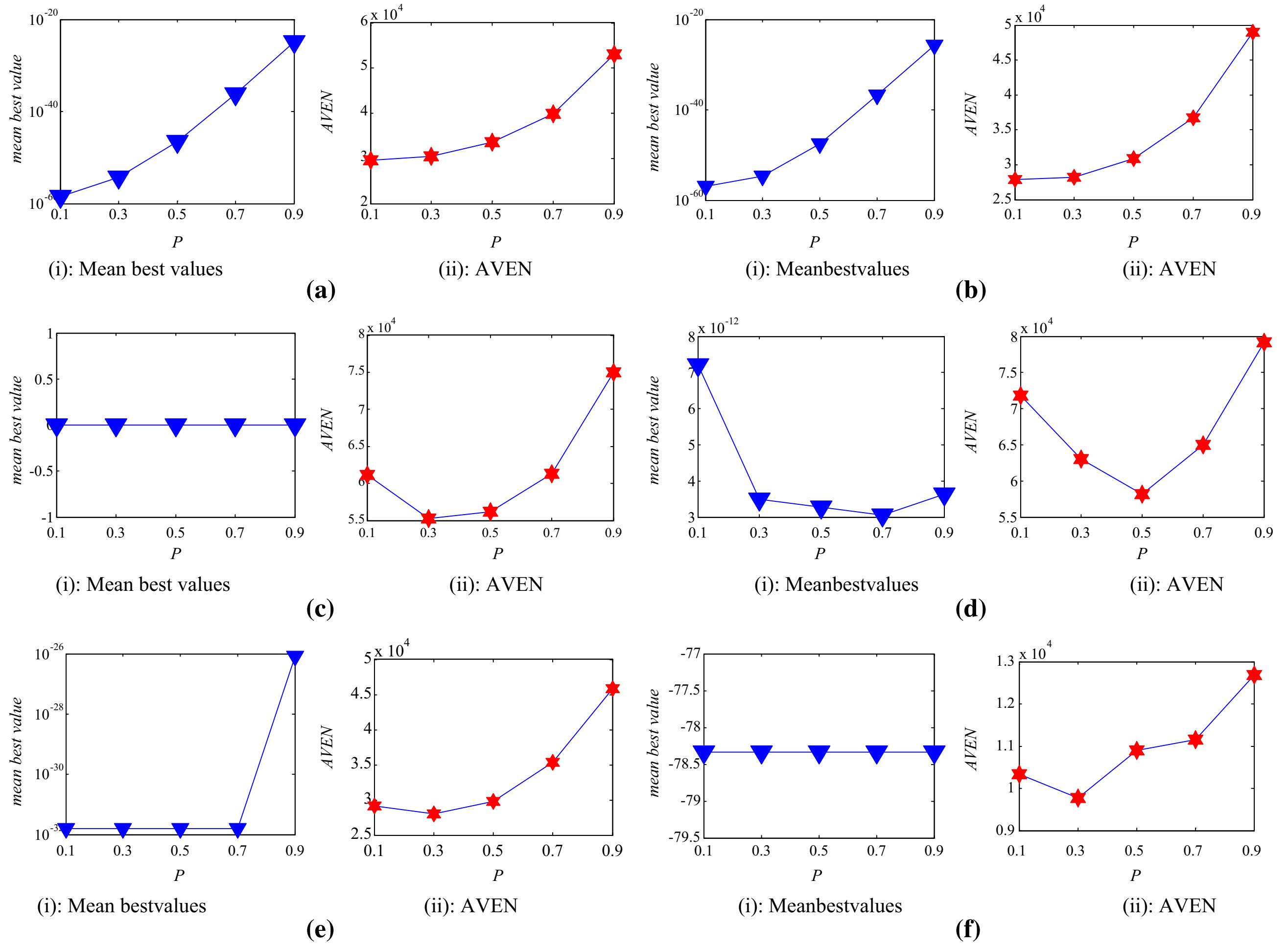
In MGABC, an additional parameter P is used to control the operating frequency of the original solution search equation (the operating frequency of the new solution search equation learning the beneficial information from the current best solution is $1-P$), which could adjust the exploration ability and exploitation ability of MPGABC. In order to analyze the influence of the parameter P on the performance of MPGABC, five different values of P (i.e., 0.1, 0.3, 0.5, 0.7 and 0.9) are utilized to study on some selected representative functions (unimodal functions f_1 and f_3, multimodal functions f_{11}, f_{14}, f_{15} and f_{21}) with $30D$. In this experiment, *SN* is set to 50, and *maxFES* is set to 150000. Each value on each test function is run independently by 25 times. The mean of the best objective function values and AVEN obtained by each value are used to evaluate their performance. The experimental results are illustrated in Fig. 8, which clearly shows that for unimodal functions, the smaller values of P (e.g., $P = 0.1$ or $P = 0.3$) can make MPGABC achieve stronger exploitation ability and better optimization performance according to *mean best value* and AVEN. Moreover, the smaller the value of P is, the better the performance is. The reason of this phenomenon is that since unimodal function has only one global optimal solution, the current best solution could always guide the correct evolution direction, and thus, the current best solution could be fully utilized to improve the performance. Regarding to multimodal functions, MGABC could obtain better performance when $P = 0.3$. This is because multimodal function has multiple local optimal solutions and thus the current best solution cannot always point to the correct direction. It means both the overuse and nonuse of the current best solution cannot obtain a better trade-off between exploration and exploitation. Therefore, based on the comprehensive consideration of unimodal and multimodal function, a proper value of P should be close to 0.3.

5.5 Experiment 4: comparison on CEC2011 real-world problems

In this subsection, MPGABC is tested on 13 kinds of real-world optimization problems (22 problems in total), which are all derived from CEC2011 (Das and Suganthan 2010), to further compare the performance of MPGABC with GABC, best-so-far ABC (BsfABC), dABC and MABC. To make a fair comparison, the parameter settings of all competitors are set the same as the settings used in their original papers. And according to the requirements of CEC2011 (Das and Suganthan 2010), *maxFES* is employed as the termination condition, which is set to 50,000, and all compared algorithms are independently conducted 25 runs on each real-world optimization problem. The mean and standard deviation of the objective functions value are used to evaluate the performance. The experimental results are given in Table 9, and the best result is highlighted in boldface.

5.6 Experiment 5: comparison on CEC2013 real-parameter problems

In this subsection, to further demonstrate the effectiveness of MPGABC in solving more complex problems, we compare MPGABC with three ABC variants (i.e., GABC, qABC and dABC) on 28 test functions with $D = 50$ and $D = 100$, which are derived from the CEC 2013 special session on real-parameter optimization (Liang et al. 2013). In this experimental study, the parameter settings of all competitors are set the same as the settings used in their original papers. According to the requirements of CEC2013 (Liang et al. 2013), the *maxFES* of all functions is set to $10000 \cdot D$, and each compared algorithm is independently conducted 51 runs on each function. The average and standard deviation of *the func-*



(i): Mean best values (ii): AVEN **(a)** (i): Meanbestvalues (ii): AVEN **(b)**

(i): Mean best values (ii): AVEN **(c)** (i): Meanbestvalues (ii): AVEN **(d)**

(i): Mean bestvalues (ii): AVEN **(e)** (i): Meanbestvalues (ii): AVEN **(f)**

Fig. 8 Mean best values and AVEN with different values of parameter *P*

tion error value $f(X_{\text{best}})$-$f(X^*)$ are employed to evaluate the optimization performance, where X_{best} is the best solution found by the algorithm in each run and X^* is the true global optimal solution of the test function. Moreover, the Wilcoxon's rank-sum test with the 5% significant level is conducted on the experimental results to obtain the reliable statistic conclusion. The experimental results are given in Tables 10 and 11 for $D = 50$ and $D = 100$, respectively. For the sake of clarity, the best results are highlighted in boldface.

The experimental results on all functions with $D = 50$ are presented in Table 10. It can be seen that MPGABC is better than or at least comparable to all compared algorithms on these test functions. To be specific, with regard to unimodal functions (F1–F5), MPGABC outperforms all compared algorithms on F3 and F5, and MPGABC is only beaten by qABC on F2, F4, and dABC on F1. For basic multimodal functions (F6–F20), MPGABC is better than or at least comparable to all compared algorithms on F7, F11, F15, F17 and F19, respectively. Moreover, MPGABC can obtain the second best results, respectively, on F10, F12, F13, F18 and F20. In addition, MPGABC is only beaten by GABC on F9, qABC on F6 and F16, qABC and dABC on F14, respectively. Furthermore, all compared algorithms can obtain the similar performance on F8. Concerning composition functions (F21–F28), MPGABC is better than or at least comparable to all compared algorithms on F21–F23, F26 and F28. On the contrary, MPGABC is only beaten by GABC on F24, F25 and F27. Overall, MPGABC outperforms GABC, qABC and dABC on 14, 10 and 16 functions, respectively. On the contrary, MPGABC is only beaten by GABC, qABC and dABC on 8, 6 and 2 functions. According to the above analysis, it can be concluded that MPGABC is better than or at least comparable to all compared algorithms when considering all the test functions with $D = 50$.

In order to investigate the performance of MPGABC on the CEC2013 test functions with high dimensionality, we further compare MPGABC with GABC, qABC and dABC on these test functions with $D = 100$. The experimental results are given in Table 11. Clearly, as shown in Table 11, MPGABC is better than all other algorithms in terms

Table 9 Comparisons of MPGABC with four state-of-the-art ABC variants on 22 real-world problems

Pro	Alg				
	GABC	BsfABC	dABC	MABC	MPGABC
	Mean (SD)	Mean (SD)	Mean (SD)	Mean (SD)	Mean (SD)
T01MF	1.11e+01 (5.14e+00)≈	1.55e+01 (3.05e+00)−	1.23e+01 (4.19e+00)≈	1.03e+01 (5.95e+00)≈	**8.11e+00(5.80e+00)**
T02 L-J	−1.72e+01 (1.41e+00)−	−1.53e+01 (1.32e+00)−	−1.61e+01 (1.42e+00)−	−1.61e+01 (2.06e+00)−	**−1.95e+01 (2.51e+00)**
T03 BCB	**1.15e−05 (0.00e+00)**≈	**1.15e−05 (0.00e+00)**≈	**1.15e−05 (0.00e+00)**≈	**1.15e−05 (0.00e+00)**≈	**1.15e−05 (0.00e+00)**
T04 STR	1.80e+01 (3.42e+00)≈	**1.47e+01 (6.85e−01)**≈	1.75e+01 (2.52e+00)≈	1.89e+01 (2.95e+00)≈	1.80e+01 (3.44e+00)
T05Si(B)	−3.41e+01 (**8.79e−01**)−	−3.25e+01 (1.80e+00)−	−3.35e+01 (1.69e+00)−	−3.40e+01 (9.61e−01)−	**−3.45e+01** (1.00e+00)
T06Si(C)	−2.87e+01 (7.25e−01)≈	−2.70e+01 (2.19e+00)−	−2.79e+01 (1.37e+00)−	−2.83e+01 (1.35e+00)−	**−2.88e+01 (6.32e−01)**
T07 SPRP	1.32e+00 (9.12e−02)≈	1.30e+00 (9.66e−02)≈	**1.27e+00 (9.30e−02)**≈	1.31e+00 (9.17e−02)≈	1.30e+00 (9.91e−02)
T08 TNEP	**2.20e+02 (0.00e+00)**≈	**2.20e+02 (0.00e+00)**≈	**2.20e+02 (0.00e+00)**≈	**2.20e+02 (0.00e+00)**≈	**2.20e+02 (0.00e+00)**
T09 LSTP	8.42e+04 (3.05e+04)−	1.25e+06 (9.24e+04)−	6.51e+05 (9.06e+04)−	5.83e+04 (1.00e+04)−	**2.47e+04 (9.91e+03)**
T10 CAAD	−1.81e+01 (1.04e+00)−	−1.62e+01 (1.30e+00)−	−1.61e+01 (1.36e+00)−	−1.80e+01 (1.20e+00)−	**−1.89e+01 (1.31e+00)**
T11.1 DED1	**5.28e+04** (5.87e+02)+	1.27e+07 (4.79e+06)−	**5.28e+04 (5.74e+02)**+	6.09e+05 (2.16e+05)−	5.70e+04 (1.54e+03)
T11.2 DED2	5.53e+07 (1.04e+06)−	8.04e+06 (5.02e+05)−	4.22e+06 (3.64e+05)−	2.83e+06 (1.59e+05)−	**1.38e+06 (7.35e+04)**
T11.3 ELD1	1.55e+04(4.19e+00)−	1.55e+04(4.99e+00)−	1.55e+04(4.09e+00)−	1.55e+04(5.66e+00)−	**1.54e+04(2.45e+00)**
T11.4 ELD2	2.40e+07 (**7.60e−09**)−	**1.91e+04** (8.72e+01)≈	**1.91e+04** (9.51e+01)≈	1.93e+04 (1.86e+02)−	**1.91e+04** (1.18e+02)
T11.5 ELD3	1.46e+08 (**0.00e+00**)−	3.31e+04 (3.91e+01)−	3.31e+04 (4.45e+01)−	**3.30e+04** (3.96e+01)≈	**3.30e+04** (4.41e+01)
T11.6 ELD4	3.98e+06 (3.07e+05)−	2.48e+06 (2.33e+05)−	**1.37e+05 (1.50e+03)**≈	**1.37e+05** (3.48e+03)≈	**1.37e+05** (2.01e+03)
T11.7 ELD5	3.97e+11 (1.66e+09)−	2.02e+09 (6.96e+08)−	4.48e+07 (7.25e+07)−	1.08e+07 (1.56e+07)−	**3.62e+06 (3.73e+06)**
T11.8 HS1	2.20e+07 (3.17e+06)≈	3.47e+07 (1.88e+06)−	2.98e+07 (**1.76e+06**)−	**1.64e+07** (2.52e+06)+	2.36e+07 (2.06e+06)
T11.9 HS2	1.71e+08 (5.91e+06)−	3.34e+07 (5.48e+06)−	2.90e+07 (**1.86e+06**)−	**1.42e+07** (2.63e+06)+	2.03e+07 (3.47e+06)
T11.10 HS3	1.71e+08 (5.55e+06)−	3.74e+07 (3.25e+06)−	2.95e+07 (**1.87e+06**)−	**1.82e+07** (2.43e+06)+	2.36e+07 (2.08e+06)
T12 (me)	2.07e+01 (2.94e+00)≈	2.50e+01 (3.08e+00)−	2.14e+01 (2.70e+00)≈	2.02e+01 (2.90e+00)≈	**1.99e+01 (2.47e+00)**
T13 (Ca)	4.07e+01 (3.91e+00)−	2.42e+01 (2.97e+00)−	2.14e+01 (3.40e+00)≈	**1.85e+01 (3.03e+00)**≈	1.99e+01 (3.08e+00)
−/≈/+	13/8/1	17/5/0	12/9/1	10/9/3	

"−", "≈" and "+," respectively, denote that the performance of the corresponding algorithm is worse than, better than, and similar to that of MPGABC according to the Wilcoxon's rank test at a 0.05 significance level

Table 10 Comparisons of MPGABC with ABC variants on CEC2013 functions with 50D

Pro	Alg			
	GABC	qABC	dABC	MPGABC
	Mean (SD)	Mean (SD)	Mean (SD)	Mean (SD)
F1	2.76e−13 (9.44e−14)−	2.50e−13 (6.83e−14)−	**8.92e−14** (1.12e−13)+	2.27e−13 (**0.00e−00**)
F2	1.35e+07 (5.63e+06)−	**5.84e+06** (**1.91e+06**)+	1.18e+07 (3.21e+06)≈	1.10e+07 (2.39e+06)
F3	1.30e+09 (1.19e+09)−	1.18e+09 (9.67e+08)−	1.53e+09 (1.16e+09)−	**8.15e+08** (**4.23e+08**)
F4	1.59e+05 (**1.48e+04**)≈	**1.52e+05** (1.71e+04)+	1.62e+05(1.59e+04)≈	1.60e+05 (1.50e+04)
F5	2.74e−13 (5.65e−14)−	4.12e−13 (6.81e−14)−	2.32e−13 (**2.23e−14**)−	**2.01e−13** (4.87e−14)
F6	4.36e+01 (**1.82e+00**)≈	**4.15e+01** (4.85e+00)+	4.22e+01(3.80e+00)≈	4.35e+01 (2.05e+00)
F7	**1.43e+02** (**1.29e+01**)≈	1.44e+02 (1.72e+01)≈	1.52e+02 (1.54e+01)−	**1.43e+02** (1.57e+01)
F8	**2.11e+01** (3.51e−02)≈	**2.11e+01** (4.40e−02)≈	**2.11e+01** (**2.97e−02**)≈	**2.11e+01** (3.72e−02)
F9	**5.88e+01** (**2.43e+00**)+	6.00e+01 (2.67e+00)≈	6.12e+01(2.55e+00)≈	6.02e+01 (2.93e+00)
F10	3.17e+00 (8.48e−01)−	**1.74e−01** (**8.24e−02**)+	1.62e+00 (3.06e−01)−	3.60e−01 (1.47e−01)
F11	8.25e−14 (2.86e−14)−	1.11e−13 (1.11e−14)−	**5.46e−14** (1.11e−14)≈	5.68e−14 (**0.00e−00**)
F12	**4.64e+02** (**5.64e+01**)+	6.42e+02 (7.33e+01)−	6.62e+02 (8.48e+01)−	5.64e+02 (7.11e+01)
F13	5.81e+02(5.60e+01)+	7.48e+02 (6.02e+01)−	7.63e+02 (**5.41e+01**)−	6.57e+02 (6.96e+01)
F14	3.16e+00 (1.28e+00)−	**6.64e−02** (**2.70e−02**)+	7.08e−01(7.53e−01)+	1.71e+00 (7.44e−01)
F15	7.57e+03 (6.11e+02)−	**7.30e+03** (5.12e+02)≈	7.59e+03 (4.84e+02)−	7.42e+03 (**3.85e+02**)
F16	1.53e+00 (2.57e−01)−	**1.22e+00** (2.27e−01)+	1.42e+00 (**1.83e−01**)≈	1.39e+00 (1.97e−01)
F17	**5.08e+01** (1.06e−02)≈	**5.08e+01** (**3.33e−04**)≈	5.10e+01 (5.18e−02)−	**5.08e+01** (1.31e−02)
F18	**5.05e+02** (**5.02e+01**)+	6.79e+02 (7.41e+01)−	7.65e+02 (7.57e+01)−	5.91e+02 (5.17e+01)
F19	9.38e−01 (2.94e−01)−	5.80e−01 (1.97e−01)≈	9.77e−01 (2.07e−01)−	**5.34e−01** (**1.55e−01**)
F20	**2.34e+01** (6.93e−01)+	2.43e+01 (**2.90e−01**)−	2.42e+01 (4.79e−01)−	2.37e+01 (7.31e−01)
F21	2.10e+02 (9.04e+01)−	2.00e+02 (**1.12e+01**)≈	**1.92e+02**(2.58e+01)≈	2.06e+02 (1.70e+01)
F22	2.26e+01 (2.64e+01)−	5.90e+00 (3.71e+00)−	2.68e+01 (3.46e+00)−	**1.34e+01** (**1.80e+00**)
F23	9.69e+03 (8.78e+02)−	9.55e+03 (7.05e+02)≈	9.73e+03 (**6.65e+02**)−	**9.51e+03** (6.76e+02)
F24	**3.66e+02** (**7.56e+00**)+	3.75e+02 (8.90e+00)≈	3.77e+02 (9.97e+00)−	3.73e+02 (**7.56e+00**)
F25	**4.13e+02** (**9.05e+00**)+	4.31e+02 (1.08e+01)−	4.36e+02 (1.15e+01)−	4.24e+02 (9.80e+00)
F26	2.02e+02 (3.95e−01)−	**2.01e+02** (**1.57e−01**)≈	2.02e+02(2.93e−01)−	**2.01e+02** (2.46e−01)
F27	**1.58e+03** (6.27e+02)+	1.66e+03 (6.29e+02)≈	1.67e+03 (6.44e+02)≈	1.74e+03 (**5.68e+02**)
F28	**4.00e+02** (5.67e−13)≈	**4.00e+02** (1.05e−02)≈	**4.00e+02** (2.15e−04)≈	**4.00e+02** (**4.57e−13**)
−/≈/+	14/6/8	10/12/6	16/10/2	

"−", "≈" and "+," respectively, denote that the performance of the corresponding algorithm is worse than, similar to, and better than that of MPGABC according to the Wilcoxon's rank test at a 0.05 significance level

of solution accuracy on unimodal functions F3 and F5, and MPGABC also can obtain the second best results, respectively, on F1 and F2. Moreover, MPGABC is only beaten by qABC on F4. With regard to basic multimodal functions (F6–F20), MPGABC is better than or at least comparable to all compared algorithms on F6, F11, F15, F16, F17, F19 and F20, respectively. Moreover, MPGABC is only beaten by GABC on F7, F12, F13 and F18, and qABC on F10, respectively. In addition, all compared algorithms show similar performance on F8 and F9. However, MPGABC performs worst on F14, which may be caused by the property that the number of local optimal solution of F14 is huge and the second better local optimal solution is far from the global optimum. For composition functions (F21–F28), MPGABC is better than or at least comparable to all compared algorithms on F26, and MPGABC also exhibits the second best performance on F21, F23–F25 and F28, respectively. But MPGABC is, respectively, beaten by qABC and dABC on F22, GABC and dABC on F27. In summary, MPGABC outperforms GABC, qABC and dABC on 16, 13 and 15 functions, respectively. On the contrary, MPGABC is only beaten by GABC, qABC and dABC on 9, 7 and 4 functions. Therefore, MPGABC can also perform better than the recent ABC variants on these complex test functions with $D = 100$. According to the above analysis, it can be concluded that the superiority of MPGABC is not affected by the growth of the dimensions in search space.

Table 11 Comparisons of MPGABC with ABC variants on CEC2013 functions with 100D

Pro	Alg			
	GABC	qABC	dABC	MPGABC
	Mean (SD)	Mean (SD)	Mean (SD)	Mean (SD)
F1	5.71e−13 (1.23e−13)−	6.51e−13 (9.12e−14)−	**2.27e−13** (**0.00e−00**)+	4.10e−13 (9.12e−14)
F2	3.43e+07 (9.75e+06)−	**1.09e+07** (**2.64e+06**)+	3.10e+07 (5.07e+06)−	2.64e+07 (3.43e+06)
F3	1.05e+10 (6.03e+09)−	1.20e+10 (5.84e+09)−	1.25e+10 (5.94e+09)−	**7.53e+09** (**3.03e+09**)
F4	3.51e+05 (2.41e+04)≈	**3.31e+05** (2.37e+04)+	3.41e+05 (2.51e+04)≈	3.43e+05 (**2.09e+04**)
F5	4.70e−13 (5.58e−14)−	8.78e−13 (9.40e−14)−	5.46e−13 (5.10e−14)−	**3.39e−13** (**1.59e−14**)
F6	1.77e+02 (**2.71e+01**)−	**1.46e+02** (3.38e+01)≈	1.54e+02 (2.91e+01)≈	**1.46e+02** (2.76e+01)
F7	**4.09e+02** (**1.58e+02**)+	2.01e+03 (1.13e+03)−	2.33e+03 (1.42e+03)−	1.18e+03 (7.24e+02)
F8	**2.13e+01** (**2.50e−02**)≈	**2.13e+01** (2.95e−02)≈	**2.13e+01** (2.89e−02)≈	**2.13e+01** (2.64e−02)
F9	**1.40e+02** (4.23e+00)≈	1.42e+02 (4.24e+00)≈	1.43e+02 (**3.56e+00**)≈	1.41e+02 (4.31e+00)
F10	1.43e+00 (1.92e−01)−	**1.31e−01** (**7.23e−02**)+	1.42e+00 (1.22e−01)−	9.32e−01 (3.07e−01)
F11	1.54e−13 (3.07e−14)−	2.36e−13 (2.09e−14)−	**1.10e−13** (**1.35e−14**)≈	**1.10e−13** (**1.35e−14**)
F12	**1.60e+03** (**1.31e+02**)+	2.03e+03 (1.84e+02)−	2.17e+03 (1.61e+02)−	1.86e+03 (1.82e+02)
F13	**1.81e+03** (**1.09e+02**)+	2.30e+03 (1.28e+02)−	2.31e+03 (1.47e+02)−	2.05e+03 (1.40e+02)
F14	2.85e+00 (1.39e+00)+	**9.24e−02** (**5.28e−02**)+	9.38e−01 (9.79e−01)+	5.65e+00 (1.25e+00)
F15	1.57e+04 (8.16e+02)−	**1.51e+04** (8.96e+02)≈	1.55e+04 (**8.01e+02**)−	**1.51e+04** (8.81e+02)
F16	2.07e+00 (**2.14e−01**)−	**1.85e+00** (2.24e−01)≈	1.93e+00(2.27e−01)≈	1.91e+00 (2.18e−01)
F17	1.07e+02 (2.35e−02)−	**1.02e+02** (**5.12e−04**)≈	**1.02e+02** (1.24e−01)≈	**1.02e+02** (2.29e−02)
F18	**1.76e+03** (**1.37e+02**)+	2.41e+03 (1.65e+02)−	2.60e+03 (1.96e+02)−	2.15e+03 (1.63e+02)
F19	2.25e+00 (5.07e−01)−	1.53e+00 (3.95e−01)−	2.63e+00 (3.74e−01)−	**1.27e+00** (**2.44e−01**)
F20	5.00e+01 (1.33e−01)−	5.00e+01 (2.82e−03)−	5.00e+01 (**1.11e−01**)−	**4.98e+01** (2.43e−01)
F21	3.67e+02 (4.91e+01)−	**3.03e+02** (**1.31e+01**)+	4.16e+02 (3.57e+01)−	3.65e+02 (4.83e+01)
F22	7.47e+01 (5.18e+01)−	**1.40e+01** (**2.14e+00**)+	4.34e+01 (5.52e+00)+	4.53e+01 (3.96e+01)
F23	2.16e+04 (1.45e+03)−	**2.08e+04** (1.43e+03)+	2.15e+04 (1.31e+03)≈	2.14e+04 (**1.19e+03**)
F24	**6.00e+02** (**1.17e+01**)+	6.25e+02 (1.53e+01)−	6.26e+02 (1.27e+01)−	6.12e+02 (1.33e+01)
F25	**7.16e+02** (**1.42e+01**)+	7.62e+02 (2.07e+01)−	7.70e+02 (1.94e+01)−	7.43e+02 (2.23e+01)
F26	2.07e+02 (2.15e+01)−	**2.02e+02** (**3.77e−01**)≈	**2.04e+02** (1.45e+00)≈	2.03e+02 (9.36e−01)
F27	**3.81e+03** (1.14e+03)+	4.12e+03 (9.46e+02)≈	4.05e+03 (**1.10e+03**)+	4.08e+03 (9.35e+02)
F28	**4.27e+03** (**1.18e+03**)+	8.28e+03 (2.83e+03)−	9.93e+03 (2.92e+03)−	6.74e+03 (2.48e+03)
−/≈/+	16/3/9	13/8/7	15/9/4	

"−", "≈" and "+," respectively, denote that the performance of the corresponding algorithm is worse than, similar to, and better than that of MPGABC according to the Wilcoxon's rank test at a 0.05 significance level

Furthermore, an insightful phenomenon should be point out that all ABC methods are able to obtain the near optimal solution on F1, F5 and F11, but fail to get the near optimal solution on others functions. The reason for this phenomenon may be that F1, F5, F11 and F22 are separable, while all other functions are non-separable. Therefore, the property of changing one variable at one time in ABCs may determine that ABCs are suitable for solving separable problems.

6 Conclusion and future work

In this paper, to improve the exploitation ability and further enhance the performance of ABC, we propose two new algorithmic components by suitably exploiting the current best solution and other good solutions. The one component is the novel search strategy, which exploits the useful information of the current best solution according to a probability parameter P. The other one is the novel probability model, which can make sure that the good solutions can always attract more onlooker bees to search. Through combining these two new algorithmic components with the basic framework of ABC, a new ABC variant is produced, named MPGABC. The performance of MPGABC has been validated by the comparison with other outstanding ABC variants (i.e., GABC, qABC, best-so-far ABC, dABC and MABC) on 22 benchmark test functions with 30D, 50D and 100D, 22 real-world problems and 28 CEC2013 real-parameter optimization problems with

50D and 100D in terms of solution accuracy, robustness and convergence speed.

Since the current best solution and other good solutions have valuable information, how to effectively employ them to further improve the performance of ABC is still worth studying in the future. Moreover, MPGABC could be extended to solve constraint optimization problems, multi-objective optimization problems (Lin et al. 2015) and real-world practice optimization problems.

Acknowledgements This work is supported by the National Natural Science Foundation of China under Grants 61402294, 61472258 and 61572328, Guangdong Natural Science Foundation under Grant S2013040012895, Foundation for Distinguished Young Talents in Higher Education of Guangdong, China, under Grant 2013LYM_0076, Major Fundamental Research Project in the Science and Technology Plan of Shenzhen under Grants KQCX20140519103756206, JCYJ20140418091413526, JCYJ20140509172609162, JCYJ20140828163633977, JCYJ20140418181958501, JCYJ20150630105452814, JCYJ20160310095523765 and JCYJ20160307111232895. The Open Research Fund of China-UK Visual Information Processing Lab.

Compliance with ethical standards

Conflict of interest All authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

Abraham A, Jatoth RK, Rajasekhar A (2012) Hybrid differential artificial bee colony algorithm. J Comput Theor Nanosci 9(2):249–257

Akay B, Karaboga D (2012) A modified artificial bee colony algorithm for real-parameter optimization. Inf Sci 192(1):120–142

Aydogdu I, Akin A, Saka MP (2016) Design optimization of real world steel space frames using artificial bee colony algorithm with Levy flight distribution. Adv Eng Softw 92:1–14

Banharnsakun A, Achalakul T, Sirrinaovakul B (2011) The best-so-far selection in artificial bee colony algorithm. Appl Soft Comput 11(2):2888–2901

Banharnsakun A, Sirinaovakual B, Achalakul T (2013) The best-so-far ABC with multiple patrilines for clustering problems. Neurocomputing 116:355–366

Bayraktar T (2014) A memory-integrated artificial bee algorithm for heuristic optimization, M. SC. thesis. University of Bedfordshire

Chen SM, Sarosh A, Dong YF (2012) Simulated annealing based artificial bee colony algorithm for global numerical optimization. Appl Math Comput 219(8):3575–3589

Cui Z, Gao X (2012) Theory and applications of swarm intelligence. Neural Comput Appl 21(2):205–206

Das S, Suganthan PN (2010) Problem definitions and evaluation criteria for CEC 2011 competition on testing evolutionary algorithms on real world optimization problems. Jadavpur University, India and Nanyang Technological University, Singapore; 2010 Technical report

Dorigo M, Maniezzo V, Colorni A (1996) Ant system: optimization by a colony of cooperating agents. IEEE Trans Syst 26(1):29–41

Feng JW, Dai AD, Xu C, Wang JY (2011) Designing lag synchronization for unified chaotic systems. Comput Math Appl 61:2123–2128

Fister I, Fjjr I, Brest J, Zumer V (2012) Memetic artificial bee colony algorithm for large-scale global optimization. IEEE Congress on Evolutionary Computation 2012 (pp 1–8). IEEE

Gao WF, Liu SY (2011) Improved artificial bee colony algorithm for global optimization. Inf Process Lett 111(17):871–882

Gao WF, Liu SY (2012) A modified artificial bee colony algorithm. Comput Oper Res 39(3):687–697

Gao WF, Liu SY, Huang LL (2012) A global best artificial bee colony algorithm for global optimization. J Comput Appl Math 236(11):2741–2753

Gao WF, Liu SY, Huang LL (2013a) A novel artificial bee colony algorithm based on modified search equation and orthogonal learning. IEEE Trans Cybern 43(3):1011–1024

Gao WF, Liu SY, Huang LL (2013b) A novel artificial bee colony algorithm with Powell's method. Appl Soft Comput 13(9):3763–3775

Gao WF, Liu SY, Huang LL (2014) Enhancing artificial bee colony algorithm using more information-based search equations. Inf Sci 270(1):112–133

Gao WF, Chan FTS, Huang LL, Liu SY (2015a) Bare bones artificial bee colony algorithm with parameter adaptation and fitness-based neighborhood. Inf Sci 316:180–200

Gao WF, Huang LL, Liu SY, Chan FTS, Dai C (2015b) Artificial bee colony algorithm with multiple search strategies. Appl Math Comput 271:269–287

Gao WF, Huang LL, Liu SY, Dai C (2015c) Artificial bee colony algorithm based on information Learning. IEEE Trans Cybern 45(12):2827–2839

Hsieh TJ, Hsiao HF, Yeh WC (2012) Mining financial distress trend data using penalty guided support vector machines based on hybrid of particle swarm optimization and artificial bee colony algorithm. Neurocomputing 82:196–206

Hu Y, Sim CK, Yang X (2015) A subgradient method based on gradient sampling for solving convex optimization problems. Numer Func Anal Opt 36(12):1559–1584

Hu YH, Yu CKW, Li C (2016) Stochastic subgradient method for quasi-convex optimization problems. J Nonlinear Convex Anal 174(4):711–724

Hunter A, Chiu KS (2000) Genetic algorithm design of neural network and fuzzy logic controllers. Soft Comput 4(3):186–192

Kang F, Li JJ, Xu Q (2009) Structural inverse analysis by hybrid simplex artificial bee colony algorithms. Comput Struct 87(13–14):816–870

Kang F, Li JJ, Ma ZY (2011a) Rosenbrock artificial bee colony algorithm for accurate global optimization of numerical functions. Inf Sci 181(16):3508–3531

Kang F, Li JJ, Ma ZY, Li H (2011b) Artificial bee colony algorithm with local search for numerical optimization. J Softw 6(3):490–497

Karaboga D (2005) An idea based on honey bee swarm for numerical optimization. Technical report-TR06, Erciyes University

Karaboga D, Akay B (2009) A comparative study of artificial bee colony algorithm. Appl Math Comput 214(1):108–132

Karaboga D, Akay B (2011) A modified artificial bee colony (ABC) algorithm for constrained optimization problems. Appl Soft Comput 11(3):3021–3031

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

Karaboga D, Basturk B (2008) On the performance of artificial bee colony (ABC) algorithm. Appl Soft Comput 8(1):687–697

Karaboga D, Gorkemli B (2014) A quick artificial bee colony (qABC) algorithm and its performance on optimization problems. Appl Soft Comput 23:227–238

Kennedy J, Eberhart R (1995) Particle swarm optimization. IEEE Int Conf Neural Netw 4:1942–1948

Kiran MS, Babalik A (2014) Improved artificial bee colony algorithm for continuous optimization problems. J Comput Commun 2:108–116

Kiran MS, Findik O (2015) A directed artificial bee colony algorithm. Appl Soft Comput 26:454–462

Kiran MS, Hakli H, Gunduz M, Uguz H (2015) Artificial bee colony algorithm with variable search strategy for continuous optimization. Inf Sci 300:140–157

Krink T, Paterlini S (2011) Multiobjective optimization using differential evolution for real-world portfolio optimization. Comput Manag Sci 8(1):157–179

Kuo RJ, Wang MH, Huang TW (2011) An application of particle swarm optimization algorithm to clustering analysis. Soft Comput 15(3):533–542

Li X, Yang GF (2016) Artificial bee colony algorithm with memory. Appl Soft Comput 41:362–372

Liang JJ, Qu BY, Suganthan PN, Alfredo GH (2013) Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization. Technical report 201212, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and technical report, Nanyang Technological University, Singapore, January 2013

Lin QZ, Chen JY, Zhan ZH, Chen WN, Coello CAC, Yin YL, Lin CM, Zhang J (2015) A hybrid evolutionary immune algorithm for multiobjective optimization problems. IEEE Trans Evolut Comput 20(5):711–729

Loubiere P, Jourdan A, Siarry P, Chelouah R (2016) A sensitivity analysis method for driving the Artificial Bee Colony algorithm's search process. Appl Soft Comput 41:515–531

Luo J, Wang Q, Xiao XH (2013) A modified artificial bee colony algorithm based on converge-onlookers approach for global optimization. Appl Math Comput 219(20):10253–10262

Ma M, Liang J, Guo M, Fan Y, Yin YL (2011) SAR image segmentation based on artificial bee colony algorithm. Appl Soft Comput 11(8):5205–5214

Marinakis Y, Marinaki M, Matsatsinis N (2009) A hybrid discrete artificial bee colony—GRASP algorithm for clustering. In: Proceedings of the international conference on computers & industrial engineering 2009. IEEE, pp 548–553

Mavrovouniotis M, Yang SX (2011) A memetic ant colony optimization algorithm for the dynamic travelling salesman problem. Soft Comput 15(7):1405–1425

Omidvar MN, Li XD, Mei Y, Yao X (2014) Cooperative co-evolution with differential grouping for large scale optimization. IEEE Trans Evol Comput 18(3):378–393

Ozturk C, Hancer E, Karaboga D (2015) A novel binary artificial bee colony algorithm based on genetic operators. Inf Sci 297:154–170

Reza A, Hedayatzadeh R, Ziarati K, Hassanizadeh B (2012) A multiobjective artificial bee colony algorithm. Swarm Evol Comput 2(1):39–52

Shalan SAB, Ykhlef M (2015) Multi-objective portfolio optimization problem for Saudi Arabia stock market using hybrid clonal selection and particle swarm optimization. J Sci Eng 40(8):2407–2421

Shan H, Yasuda T, Ohkura K (2015) A self adaptive hybrid enhanced artificial bee colony algorithm for continuous optimization problems. Biosystems 132–133(7):43–53

Sharma TK, Pant M (2011) Differential operators embedded artificial bee colony algorithm. Int J Appl Evol Comput 2(3):1–14

Shi X, Li Y, Li H, Guan R, Wang L, Liang Y (2010) An integrated algorithm based on artificial bee colony and particle swarm optimization. IEEE Int Conf Neural Netw 5:2586–2590

Storm R, Price K (1997) Differential evolution-A simple and efficient heuristic for global optimization over continuous spaces. J Glob Optim 11(4):341–359

Sun Y, Zhang CY, Gao L, Wang XJ (2011) Multi-objective optimization algorithms for flow shop scheduling problem: a review and prospects. Int J Adv Mancuf Technol 55(5):723–739

Tang KS, Man KF, Kwong S, He Q (1996) Genetic algorithms and their applications. IEEE Signal Proc Mag 13(6):22–37

Teo J (2006) Exploring dynamic self-adaptive populations in differential evolution. Soft Comput 10(8):673–686

Tuba M, Bacanin N (2014) Artificial bee colony algorithm hybridized with firefly algorithm for cardinality constrained mean-variance portfolio selection problem. Appl Math Inf Sci 8(6):2831–2844

Wang H, Wu Z, Rahnamayan S, Sun H, Liu Y, Pan J (2014) Multi-strategy ensemble artificial bee colony algorithm. Inf Sci 279:587–603

Wei YH, Xu C, Hu QY (2013) Transformation of optimization problems in revenue management, queueing system, and supply chain management. Int J Prod Econ 146(2):588–597

Xiang WL, An MQ (2013) An efficient and robust artificial bee colony algorithm for numerical optimization. Comput Oper Res 40(5):1256–1265

Xiang W, Ma S, An M (2014) hABCDE: a hybrid evolutionary algorithm based on artificial bee colony algorithm and differential evolution. Appl Math Comput 238:370–386

Xiao R, Chen T (2011) Enhancing ABC optimization with Ai-net algorithm for solving project scheduling problem. ICNC 3:1284–1288

Zhang CQ, Zheng JG, Zhou YQ (2015) Two modified artificial bee colony algorithms inspired by grenade explosion method. Neurocomputing 151(3):1198–1207

Zhou XY, Wang H, Wang MW, Wan JY (2015) Enhancing the modified artificial bee colony algorithm with neighborhood search. Soft Comput. doi:10.1007/s00500-015-1977-x

Zhu G, Kwong S (2010) Gbest-guided artificial bee colony algorithm for numerical function optimization. Appl Math Comput 217(7):3166–3173