Applied Soft Computing 48 (2016) 137–150

Contents lists available at ScienceDirect

Applied Soft Computing

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

Enhanced artificial bee colony algorithm through differential evolution

Wei-feng Gao[a,*], Ling-ling Huang[a], Jue Wang[b], San-yang Liu[c], Chuan-dong Qin[d]

[a] School of Science, China University of Petroleum, Qingdao 266580, China
[b] Center for Forecasting Science, Academy of Mathematics and Systems Science, CAS, Beijing 100190, China
[c] School of Mathematics and Statistics, Xidian University, Xi'an, Shannxi 710071, China
[d] Mathematics and Information Science Colleges, Beifang University of Nationalities, Yinchuan 750021 China

ARTICLE INFO

Article history:
Received 29 April 2014
Received in revised form 30 April 2015
Accepted 21 October 2015
Available online 9 July 2016

Keywords:
Artificial bee colony algorithm
Differential evolution
Population initialization
Evaluation strategy

ABSTRACT

Artificial bee colony algorithm (ABC) is a relatively new optimization algorithm. However, ABC does well in exploration but badly in exploitation. One possible way to improve the exploitation ability of the algorithm is to combine ABC with other operations. Differential evolution (DE) can be considered as a good choice for this purpose. Based on this consideration, we propose a new algorithm, i.e. DGABC, which combines DE with gbest-guided ABC (GABC) by an evaluation strategy with an attempt to utilize more prior information of the previous search experience to speed up the convergence. In addition, to improve the global convergence, when producing the initial population, a chaotic opposition-based population initialization method is employed. The comparison results on a set of 27 benchmark functions demonstrate that the proposed method has better performance than the other algorithms.

1. Introduction

In the past decades, people have developed numerous optimization techniques [1] to deal with the complex optimization problems from nature, such as genetic algorithm (GA) inspired by the Darwinian law of survival of the fittest [2], particle swarm optimization (PSO) inspired by the social behavior of bird flocking or fish schooling [3], ant colony optimization (ACO) inspired by the foraging behavior of ant colonies [4], and biogeography-based optimization (BBO) inspired by the migration behavior of island species [5]. This kind of algorithms is named as evolutionary algorithms (EAs). Artificial bee colony algorithm (ABC) is such a novel technique proposed by Karaboga [6] based on simulating the foraging behavior of honey bee swarm. The performance of ABC has already been compared to other EAs, such as GA, DE, and PSO [7–9]. The results show that ABC is better than or at least comparable to the other compared methods. Due to its simple structure, easy implementation and outstanding performance, ABC has attracted increasing interest and has been applied to solve many real-world optimization problems [10–12] since its invention.

However, like other EAs, ABC also faces slow convergence. The reasons are as follows. It is well known that both the exploration and the exploitation are necessary for a population-based optimization algorithm. In EAs, the exploration refers to the ability to investigate the various unknown regions in the solution space to discover the global optimum. While, the exploitation refers to the ability to apply the knowledge of the previous good solutions to find better solutions. Actually, the two aspects contradict to each other. For the sake of the good performance on the optimization problems, the main challenge is how to strike a good balance between the exploration and the exploitation in the search process. However, the solution search equation of ABC is good at exploration but poor at exploitation [13], which results in slow convergence.

To improve the performance of the original ABC, some ABC variations have been developed. To mention a few, Gao et al. [14] proposed a novel solution search equation which looks like the crossover operation of GA. The experimental results demonstrate the effectiveness and efficiency of the modified search equation. Some researchers [13,15,16] took advantage of the information of the best solutions to improve the performance of ABC. Karaboga and Basturk [17] proposed a modified ABC which employs the frequency of perturbation and the ratio of the variance operation. Kang et al. [18] presented a memetic algorithm which combines Hooke–Jeeves pattern search with ABC. The experimental results show that the new algorithm is promising in terms of convergence speed, solution accuracy and success rate. Alatas [19] introduced a chaotic map into the initialization and the scouts phase and proposed a chaotic ABC.

In this paper, we propose a new algorithm, named as DGABC, which combines the good features of GABC and DE. Specifically, the

* Corresponding author.
E-mail address: gaoweifeng2004@126.com (W.-f. Gao).

http://dx.doi.org/10.1016/j.asoc.2015.10.070
1568-4946/

proposed algorithm adopts an evaluation strategy which selects the search method for each individual based on the previous search experience. The idea of this combination is to take advantage of the exploitation ability of DE to overcome the slow convergence speed of ABC during the search process. Additionally, to improve the global convergence, when producing the initial population, a chaotic opposition-based population initialization method is employed. The proposed algorithm is tested on a number of benchmark functions and compared with the other algorithms. The experimental results indicate the proposed approach has better convergence.

The rest of this paper is organized as follows: Section 2 describes the original ABC, while Section 3 provides a brief review of DE. The proposed algorithm is presented in Section 4. Section 5 presents and discusses the experimental results. Finally, the conclusion is drawn in Section 6.

2. Artificial bee colony algorithm

ABC [6] is a newly proposed optimization technique which simulates the intelligent foraging behavior of honey bee swarms. In ABC, a colony involves three different classes of bees: employed bees, onlookers and scouts. The position of a food source represents a solution of the optimization problem, and the nectar amount of each food source is the fitness of the corresponding solution. The first half of the colony includes employed bees, and the second half consists of onlookers. The pseudo-code of the original ABC is given below:

At the beginning, an initial population P is produced randomly which consists of SN solutions with D-dimensional vector of decision variables $X_i = \{x_{i,1}, x_{i,2}, \ldots, x_{i,D}\}$. X_i is conducted, defined by lower and upper bounds X_{min} and X_{max}.

$$x_{i,j} = x_{\min,j} + rand(0, 1)(x_{\max,j} - x_{\min,j}), \tag{2.1}$$

where $i = 1, 2, \ldots, SN$, $j = 1, 2, \ldots, D$.

In ABC, an employed bee uses the following search equation to generate a candidate solution V_i from the old one X_i.

$$v_{i,j} = x_{i,j} + \phi_{i,j}(x_{i,j} - x_{k,j}), \tag{2.2}$$

where $k \in \{1, 2, \ldots, SN\}$ and $j \in \{1, 2, \ldots, D\}$ are chosen indexes randomly; k is different from i; $\phi_{i,j}$ is a random number in $[-1, 1]$.

Unlike the employed bees, the onlookers select a solution based on the probability value p_i as follows.

$$p_i = fit_i / \sum_{j=1}^{SN} fit_j, \tag{2.3}$$

where fit_i denotes the fitness value of the ith solution. Generally, the objective function value is directly used as fit_i in ABC. In this way, the employed bees exchange their information with the onlookers.

After the onlookers choose the solutions, each one generates a new solution by Eq. (2.2) and the greedy selection is applied to select the better one from the new solution and the old one. Here, we take the minimization problem as an example. The smaller the final result, the better it is. When a solution cannot be improved through a predetermined number of cycles, named *limit*, it is

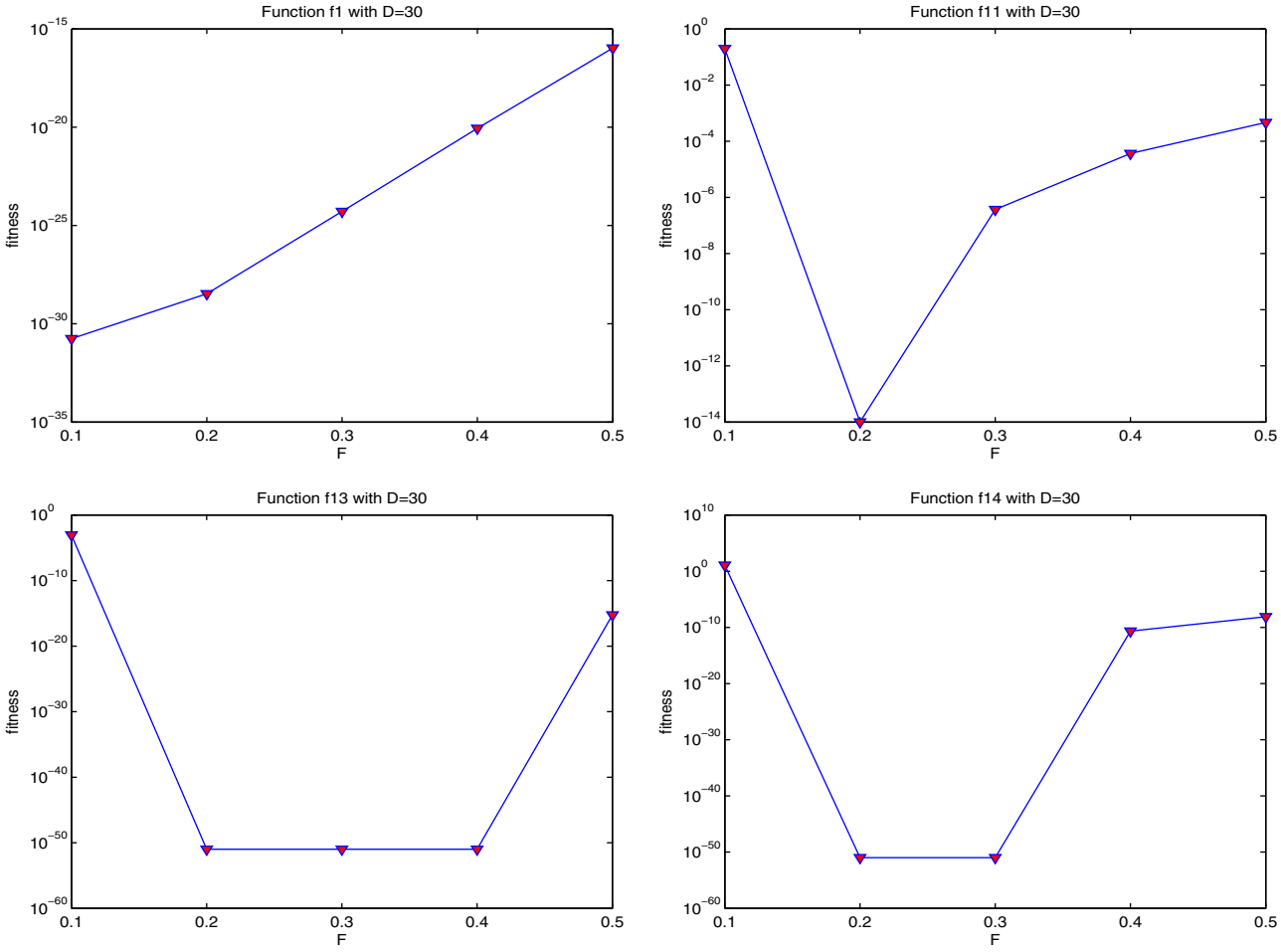

Fig. 1. DGABC's results on four test functions with different F.