Expert Systems with Applications 42 (2015) 4310–4323

Contents lists available at ScienceDirect

Expert Systems with Applications

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

An improved fruit fly optimization algorithm and its application to joint replenishment problems

Lin Wang [a], Yuanlong Shi [a], Shan Liu [b,*]

[a] School of Management, Huazhong University of Science and Technology, Wuhan 430074, China
[b] Economics and Management School, Wuhan University, Wuhan 430072, China

ARTICLE INFO

Article history:
Available online 30 January 2015

Keywords:
Fruit fly optimization algorithm
Swarm collaboration
Random perturbation
Function optimization
Joint replenishment problem

ABSTRACT

Fruit fly optimization algorithm (FOA) is one of the recent evolutionary computation approaches. This paper presents an effective and improved FOA (IFOA) for optimizing numerical functions and solving joint replenishment problems (JRPs). In the proposed IFOA, a new method of maintaining the population diversity is developed to enhance the exploration ability. Fruit flies with better fitness values use vision to fly toward a new location, and the others fly randomly in initial search space based on swarm collaboration. In addition, a new parameter to avoid the acquisition of local optimal solution is introduced to implement intelligent searching. Random perturbation is added to the updated initial location to jump out of the local optimum. Comparisons are carried out using 18 benchmark functions to verify the performance of the IFOA. Experimental results show that IFOA has better comprehensive performance than the original FOA, differential evolution algorithm, and particle swarm optimization algorithm. The IFOA is also utilized to solve the typical JRPs that have been proven as non-deterministic polynomial hard problems. Comparative examples reveal that the proposed IFOA can find better solutions than the current best algorithm; thus, it is a potential tool for various complex optimization problems.

1. Introduction

Optimization problems are used extensively in science, engineering, and business. Over the last few decades, a number of meta-heuristic optimizations have been developed. These optimizations are based on biotic factors, such as genetic algorithm (Aryanezhad & Hemati, 2008; Deb, Pratap, Agarwal, & Meyarivan, 2002; Holland, 1975), differential evolution algorithm (Storn & Price, 1997; Wang, Fu, Lee, & Zeng, 2013; Wang, Zeng, & Chen, 2014), and particle swarm optimization algorithm (Kennedy & Eberhart, 1995; Ladyzynski & Grzegorzewski, 2013; Trelea, 2003).

A novel biological species inspired the fruit fly optimization algorithm (FOA) that was first proposed by Pan (2012) based on swarm intelligence. Pan (2012) provided an easy and powerful approach to handle the complex optimization problems. The main inspiration of FOA is the sensitive osphresis and vision of fruit fly. More recently, FOAs have been applied in a variety of fields, such as power load forecasting (Li, Guo, Zhao, Su, & Wang, 2012), web auction logistics service (Lin, 2013), neural network parameter optimization (Chen, Lin, Huang, & Pan, 2013; Li, Guo, Li, & Sun, 2013), PID controller parameter tuning (Sheng & Bao, 2013), design and optimization of key control characteristic optimization (Xing, 2013), and multi-dimensional knapsack problem (Wang, Zheng, & Wang, 2013), layout of IMUs (Dai et al., 2014), steelmaking casting problem (Li, Pan, Mao, & Suganthan, 2014), inverse estimation of the particle size distribution (He, Qi, Yao, & Ruan, 2014), semiconductor testing scheduling problem (Zheng, Wang, & Wang, 2014).

To improve the search efficiency and global search ability, researchers designed several improved FOAs (IFOAs) (Dai, Zhao, Lu, & Dai, 2014; Pan, 2013; Pan, Sang, Duan, & Gao, 2014; Shan, Cao, & Dong, 2013; Yuan, Dai, Zhao, & He, 2014). However, the improvements are conducted only by continuous optimization problem. A typical improvement presented by Shan et al. (2013) replaced the non-linear generation mechanism of candidate solution (NGMS) with a linear generation mechanism of candidate solution (LGMS). However, more results of benchmark functions tests prove the performance should be further improved (Pan et al., 2014). Yuan et al. (2014) presented a novel multi-swarm FOA which can take full advantage of swarm scale of fruit flies, but only six benchmark functions were utilized which may be insufficient to verify its applicability.

There is a common disadvantage of all these FOAs. All the individuals of the swarm use vision to fly towards current best smell

---

* Corresponding author.

E-mail addresses: wanglin982@gmail.com (L. Wang), shiyuanlong@gmail.com (Y. Shi), shan.l.china@gmail.com (S. Liu).

http://dx.doi.org/10.1016/j.eswa.2015.01.048
0957-4174/© 2015 Elsevier Ltd. All rights reserved.

position, which implies that fruit flies are around the current best smell position that may be a local extreme. The diversity loss occurs when the global optimal is shifted away from a too converged swarm. This kind of swarm behavior is apt to being trapped in local optimal or premature especially in multi-modal optimization problems. Just like viewpoints in social psychology, if partial individuals do not follow "general trend" and have an independent "thinking", the diversity can be maintained. Or if the pioneer has an ability to distinguish local extreme, he can lead the swarm to walk out of predicament.

This study aims to propose an IFOA by incorporating two improvements on the original FOA and use it to solve both continuous optimization problems and combinatorial optimization problems. First, when flies are hunting for food, they show "team work" or "swarm collaboration" instead of scrambling; the flies share the work and cooperate with each in sharing information. This finding is useful to maintain the diversity of the swarm and to avoid being trapped in the local optimal. Flies are divided into two groups for different search jobs. Second, a new method is designed, which can urge the circulation jump out of the local extreme, and a random perturbation is added to the updated initial location. The inspiration also comes from the flies that will not stay on a small food all the time. The benchmark function test results show the improved performance by IFOA in solving high-dimension function optimization. The IFOA is also used to solve the practical joint replenishment problems (JRPs), which are NP-hard problems with important mathematics significance. Results also show that the proposed IFOA can find better solutions than the current best algorithm. The main contributions are as follows: We propose an improved FOA with swarm collaboration and random perturbation. And this is first time that FOA is used to solve the practical JRPs.

The remaining parts of the paper are organized as follows. Section 2 summarizes the review of FOA technique. Section 3 discusses the proposed IFOA in detail. Section 4 provides the comparisons of IFOA with the original FOA and another FOA variant by 18 benchmark function test. Section 5 contains test by JRPs and analysis. Section 6 proposes the conclusions and future research directions.

2. The basic FOA and analysis

2.1. Overview of original FOA

Fruit fly is an insect that exists widely in temperate and tropical climate zones worldwide. It is extensively used in genetics and developmental biology research. Fruit fly is superior to other species in vision and osphresis. During hunting for food, a fruit fly initially smells a particular odor by using its osphresis organs, sends and receives information from its neighbors and compares the current best location and fitness. Flies identify the fitness values by taste, and fly toward the location with better fitness. They use their sensitive vision to find food and fly toward that direction further. Fig. 1 shows the food finding iterative process of fruit fly swarm.

Pan (2012) proposed an FOA technique, which is the latest evolutionary computation technique, based on the food finding characteristics of fruit fly swarm. The steps of FOAs can be described as follows.

Step 1: Initialization. The swarm location range (LR), iteration number ($Maxgen$), and population size ($sizepop$) are initialized. According to Shan et al. (2013), the random fly direction and distance zone of fruit fly (FR) should be initialized first.

$$\begin{aligned} X\_axis &= rand(LR), \\ Y\_axis &= rand(LR). \end{aligned} \tag{1}$$

Step 2: Osphresis searching process.

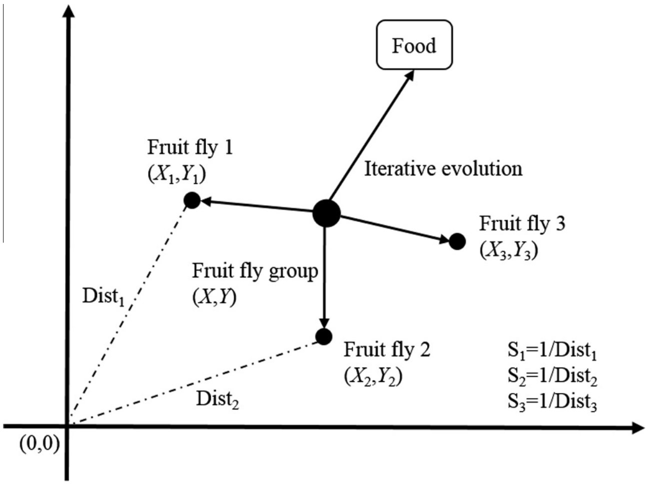

Fig. 1. Food searching iterative process of fruit fly swarm (source: Shan et al., 2013).

Step 2.1: Given the random direction and distance for food searching of any individual fruit fly,

$$\begin{aligned} X_i &= X\_axis + rand(FR), \\ Y_i &= Y\_axis + rand(FR). \end{aligned} \tag{2}$$

Step 2.2: The distance of food source to the initialization location is calculated by using the equation

$$Dist_i = \sqrt{X_i^2 + Y_i^2}. \tag{3}$$

Step 2.3: The smell concentration judgment value (S_i) is calculated by using the equation

$$S_i = \frac{1}{Dist_i}. \tag{4}$$

Step 2.4: The smell concentration ($Smell_i$) of the individual fruit fly location is calculated by inputting the S_i into the $Smell_i$ judgment function

$$Smell_i = F(S_i). \tag{5}$$

Step 2.5: The fruit fly with maximal smell concentration among the swarm is determined by using the following equation

$$[bestSmell\ bestIndex] = \max(Smell). \tag{6}$$

Step 3: Vision searching process
The maximal concentration value and X, Y coordinate are maintained. The fruit fly swarm flies toward the location by using vision.

$$\begin{aligned} Smellbest &= bestSmell, \\ X\_axis &= X(bestIndex), \\ Y\_axis &= Y(bestIndex). \end{aligned} \tag{7}$$

Step 4: The iterative optimization is entered to repeat the implementation of Steps 2–3. When the smell concentration reaches the preset precision value or the iterative number reaches the maximal $Maxgen$, the circulation stops.

Fig. 2 shows the pseudo code of original FOA.

2.2. Shortcomings of the original FOA

2.2.1. The exploration ability is weak

To evaluate the performance of an algorithm, we strictly considered the exploration and exploitation abilities. For example, in the