ARTICLE IN PRESS
JID: KNOSYS [m5G;October 30, 2017;13:33]

Knowledge-Based Systems 000 (2017) 1–11

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

A bilevel improved fruit fly optimization algorithm for the nonlinear bilevel programming problem

Guangmin Wang [a,*], Linmao Ma [a], Jiawei Chen [b]

[a] School of Economics and Management, China University of Geosciences, Wuhan 430074, China
[b] School of Mathematics and Statistics, Southwest University, Chongqing 400715, China

ARTICLE INFO

Article history:
Received 14 April 2017
Revised 28 September 2017
Accepted 30 September 2017
Available online xxx

Keywords:
Bilevel programming problem
Nonlinear bilevel programming problem
Fruit fly optimization algorithm
Improved fruit fly optimization algorithm

ABSTRACT

This paper proposes a bilevel improved fruit fly optimization algorithm (BIFOA) to address the nonlinear bilevel programming problem (NBLPP). Considering the hierarchical nature of the problem, this algorithm is constructed by combining two sole improved fruit fly optimization algorithms. In the proposed algorithm, the lower level problem is treated as a common nonlinear programming problem rather than being transformed into the constraints of the upper level problem. Eventually, 10 test problems are selected involving low-dimensional and high-dimensional problems to evaluate the performance of BIFOA from the aspects of the accuracy and stability of the solutions. The results of extensive numerical experiments and comparisons reveal that the proposed algorithm outperforms the compared algorithms and is significantly better than the methods presented in the literature; the proposed algorithm is an effective and comparable algorithm for NBLPP.

1. Introduction

The Bilevel programming problem (BLPP), a kind of Stackelberg game [1], can be used to model realistic optimization problems involving at least two stockholders in many areas, such as engineering design, economic policy, traffic problems, and so on. Thus, monographs [2–5] and reviews [6–10] regarding the theory, algorithm and application of bilevel programming have been conducted. Recently, a monograph by Zhang et al. [11] combined decision theories, methods, algorithms and applications of bi-level and tri-level decision-making effectively to present how a decision support system can support managers in practice. Lu et al. [12] systematically reviewed multilevel decision-making techniques to help researchers and practical professionals understand developments in theoretical research results and applications of multilevel decision-making techniques.

However, solving the bilevel programming problem is also a very difficult task due to its hierarchical structure, although many researchers have performed deep research in this field. Jeroslow [13] firstly pointed out that the bilevel programming problem is a non-convex problem. Subsequently, this conclusion is shown in the ensuing studies by Bard [14] and Ben-Ayed and Blair [15]. Thereafter, Vicente et al. [16] further emphasized that the linear BLPP is also a NP-hard problem, even if performing a mere search of one local optima. Consequently, in addition to the theory and application of BLPP, the idea of how to efficiently solve BLPP has attracted much attention.

The current methods developed to address BLPP can be categorized into the following two main types: conventional methods and evolutionary algorithms. The former methods are based on conventional mathematical methods including the Kth-best algorithm, exact penalty function, descent direction method, complementary pivoting algorithm, etc. 9]. These methods usually supposed the problem is differentiable and convex. However, the realistic problem is always opposite to this supposition. Thus, the conventional methods are limited to various specific problems. For the other method, evolutionary algorithms are generally inspired by intelligent algorithms consisting of a genetic algorithm (GA) [17,18], particle swarm optimization (PSO) [19], tabu search (TS) [20], artificial neural networks (ANN) [21,22], an estimation of distribution algorithm [23], intuitionistic fuzzy method [24], evolutionary algorithm [25], etc. The performance of these algorithms is related to the parameters controlling the evolutionary process due to the nature of intelligent algorithms. Moreover, the procedure is commonly complex for the readers.

However, the fruit fly optimization algorithm (FOA) is a novel meta-heuristic method simulated by the foraging behavior of fruit flies. This algorithm has a huge advantage in that it has a simple structure with very few parameters to tune, which makes it easily understood and implemented. Therefore, FOA has received much interest and has received further investigation presenting

* Corresponding author.
E-mail address: gmwang@cug.edu.cn (G. Wang).

https://doi.org/10.1016/j.knosys.2017.09.038
0950-7051/

Please cite this article as: G. Wang et al., A bilevel improved fruit fly optimization algorithm for the nonlinear bilevel programming problem, Knowledge-Based Systems (2017), https://doi.org/10.1016/j.knosys.2017.09.038

a new variant, namely the improved fruit fly optimization algorithm (IFOA). Accordingly, a novel algorithm with a nested structure based on IFOA is developed to address the nonlinear BLPP (NBLPP) in line with game theory. This nested algorithm consisting of two common IFOA is called a bilevel IFOA (BIFOA), which is similar to the multi-population fruit fly optimization. One population is responsible for the search of the optimal solution to the upper level, while the other is in charge of optimizing the lower level problem. During the optimization process, the two populations exchange information by virtue of taking the optimal solutions found for each other. That is, in this algorithm, the lower level problem is treated as a general nonlinear programming problem solved by a sole IFOA rather than transformed into the constraints of the upper level problem. Therefore, the proposed algorithm does not impose a limit on the property of the problems such as differentiability and convex, which makes it applicable to a wide range of practical optimization problems.

The main contribution of this paper is the development of a nested structure for IFOA to solve NBLPP involving high-dimensional problems based on the process of bargaining mentioned in game theory. The construction of a nested IFOA is presented, and the mechanism for the connection between two populations is illustrated. Actually, this nested structure can be used for other intelligent algorithms. To verify the performance of the proposed algorithm, the proposed bilevel IFOA is applied to solve 10 test problems involving 7 SDM test problems and 3 practical test problems including low-dimensional and high-dimensional problems. Moreover, the computational results are compared with those obtained by the current similar intelligent algorithms.

The remainder of this paper is organized as follows. In Section 2, a brief introduction of the basic concepts of NBLPP is provided. Section 3 develops a BIFOA algorithm in detail after describing the original FOA and IFOA. Section 4 presents the test problems for the numerical experiments and the numerical experimental results to demonstrate the performance of the proposed algorithm. Finally, Section 5 concludes the paper.

2. Basic concepts of NBLPP

In this paper, the nonlinear bilevel programming problem is formulated as follows:

$$(\text{NBLPP}) \quad \begin{array}{l} \min\limits_{x} F(\mathbf{x},\mathbf{y}) \\ s.t.\ \ g(\mathbf{x},\mathbf{y}) \le 0, \end{array}$$

where y solves the following problem:

$$\begin{array}{l} \min\limits_{y} f(\mathbf{x},\mathbf{y}) \\ s.t.\ \ h(\mathbf{x},\mathbf{y}) \le 0, \end{array}$$

where $\mathbf{x} \in R^{n_x}$, $\mathbf{y} \in R^{n_y}$ are decision variables of the upper and lower problems, respectively. $F, f : R^{n_x+n_y} \to R$ are the objective functions of the upper and lower level problems, respectively. $\mathbf{g} : R^{n_x+n_y} \to R^{n_u}$, $\mathbf{h} : R^{n_x+n_y} \to R^{n_l}$ are the constraint functions, respectively. For a nonlinear BLPP, at least one of these functions is nonlinear.

As mentioned above, BLPP is a mathematical model of the Stackelberg game and can be solved according to the following process. Firstly, the upper level problem is optimized with respect to the upper level decision variables independently. Secondly, for a given $\mathbf{x}$, the lower level problem is optimized with respect to the lower level decision variables. The optimal lower decision variables are obtained and transferred to the upper level problem. However, the lower decision variables affect the upper level objection value. Therefore, the upper level problem is optimized again while $\mathbf{y}$ is fixed. This procedure is a process of exchanging information between the upper and lower level problems. Notably, when both the two level objective function values remain stable, NBLPP is solved successfully.

Based on this process, this paper develops a bilevel IFOA with a nested structure for solving NBLPP. This algorithm passes over the transformation of the lower problem into the constraints of the upper level problem but considers NBLPP as two general nonlinear programming problems. The nested construction can avoid the difficulty arising from the hierarchical structure of NBLPP.

3. Overview of IFOA

This section provides a brief overview of the improved fruit fly optimization algorithm (FOA) and its application for solving the practical optimization problems as well as various improvements.

3.1. Basic concept of IFOA

FOA is a novel intelligent algorithm premised on the foraging food behavior of fruit flies. According to previous biology studies, fruit flies can easily find remote food resources by virtue of their osphresis and vision organs and have proven to be superior to other species. So the process of finding foods consists of the following two main stages: osphresis and vision foraging stages. Firstly, fruit flies smell food sources based on their acute osphresis function and fly towards the approximate locations; then, they fly to the food directly taking advantage of their sensitive vision function when they get close to the source.

In the initial research, FOA was firstly presented and used to optimize the financial distress model by Pan [26]. This algorithm has received much interest from researchers in many areas since FOA has a simple computational process and structure so it can be easily understood and implemented for users. Therefore, FOA has been applied for many optimization problems such as the analysis of service satisfaction [27], high-dimensional continuous function optimization problems [28–30], annual power load forecasting [31,32], electricity consumption forecasting [33], PID controller tuning [34,35], fuzzy PID controller parameters tuning [36], the knapsack problem [37,38], inverse estimation of particle size distribution [39], joint replenishment problems [40], the semiconductor final testing scheduling problem [41], identification of dynamic protein complexes [42] and range image registration [43]. Nevertheless, existing studies do not address the bi-level programming problem.

In view of recent research on FOA, it is obvious that the basic FOA is not capable of addressing such intricate practical problems containing negative variables because the variables view the distance as always being a positive value. For the main drawbacks of FOA, including but not limited to this one, refer Shan et al [28] and Zhang et al [44]. To overcome these limitations, further research on FOA is conducted to enhance its performance and expand the application range. For instance, Shan et al proposed a new method to generate a candidate solution [28]; Li et al and Zhang et al proposed two different modified expressions of smell concentration [34,45]. Zheng and Wang developed a two-stage adaptive fruit fly optimization algorithm [46], and Pan et al [29] provided a novel method to dynamically adjust the search radius of the fruit fly. In addition, other more elaborate improvements are presented based on a cloud model [38], chaotic operation [47], knowledge-guided improvement [48], multi-scale cooperative mutation operation [44] and multi-swarm operation [30]. Consequently, the corresponding results demonstrate that these improvements can promote the performance of the basic FOA in some sense. Obviously, the more subtle the improvements, the more parameters are introduced to the algorithm, which in turn destroys the simple structure of FOA to some extent. Eventually, for the comprehensive consideration of the complexity and efficiency of the algorithm, the