Applied Soft Computing 36 (2015) 315–333

Contents lists available at ScienceDirect

Applied Soft Computing

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

Lightning search algorithm

Hussain Shareef*, Ahmad Asrul Ibrahim, Ammar Hussein Mutlag

Department of Electrical Engineering, United Arab Emirates University, P.O. Box 15551, 1 Al-Ain, United Arab Emirates

ARTICLE INFO

Article history:
Received 11 September 2014
Received in revised form 23 July 2015
Accepted 24 July 2015
Available online 4 August 2015

Keywords:
Benchmark functions
Constrained optimization
Lightning search algorithm
Nature-inspired algorithms

ABSTRACT

This paper introduces a novel metaheuristic optimization method called the lightning search algorithm (LSA) to solve constraint optimization problems. It is based on the natural phenomenon of lightning and the mechanism of step leader propagation using the concept of fast particles known as projectiles. Three projectile types are developed to represent the transition projectiles that create the first step leader population, the space projectiles that attempt to become the leader, and the lead projectile that represent the projectile fired from best positioned step leader. In contrast to that of the counterparts of the LSA, the major exploration feature of the proposed algorithm is modeled using the exponential random behavior of space projectile and the concurrent formation of two leader tips at fork points using opposition theory. To evaluate the reliability and efficiency of the proposed algorithm, the LSA is tested using a well-utilized set of 24 benchmark functions with various characteristics necessary to evaluate a new algorithm. An extensive comparative study with four other well-known methods is conducted to validate and compare the performance of the LSA. The result demonstrates that the LSA generally provides better results compared with the other tested methods with a high convergence rate.

1. Introduction

Optimization is a process of finding the best solution. Solutions are labeled good or bad after determining the objective function that states the relations between system parameters and constraints. The objective function is often formulated based on application, and it can be in the form of fabrication cost, process efficiency, and so on.

Numerous techniques have been used to deal with optimization problems. Most classical point-by-point methods (e.g., direct methods and gradient-based methods) use a deterministic procedure to reach the optimum solution [1]. However, finding the optimum solution for such problems using classical techniques becomes complicated as the size of the search space increases with the dimension of the optimization problem [2]. Recently, computational intelligence optimization algorithms have been extensively used to solve complex optimization problems in various domains, including science, commerce, and engineering, because of their ease of use, broad applicability, and global perspective.

Computational intelligence optimization algorithms are nature-inspired computational methodologies that address complex real-world problems. These algorithms can be further divided into swarm intelligence methods and evolutionary algorithms (EAs). Swarm intelligence optimization algorithms generally use reduced mathematical models of the complex social behavior of insect or animal groups. The most popular swarm intelligence methods are particle swarm optimization (PSO) [3], artificial bee colony (ABC) [4], and ant colony optimization (ACO) [5]. The PSO mimics the movements of bird flocking or fish schooling [6]. Inspired by the food-searching mechanism of honey bees, the ABC method uses the foraging behavior of these insects [7]. Meanwhile, ACO was developed based on the behavior of ants when seeking the optimal path between their colony and food source [5]. However, these swarm intelligence methods are limited by factors such as trapping in local minima and premature convergence [6–8]. To overcome these problems, variants of these algorithms have been developed with superior performance [6,8–10]. Other swarm intelligence methods, such as the gravitational search algorithm (GSA) [11], the harmony search algorithm (HSA) [12], biogeography-based optimization [13], and the grenade explosion method [14], have also been developed.

EAs derive their working principles from natural genetic evolution. At each generation, the best individuals of the current population survive and produce offspring resembling them; hence, the population gradually comprises enhanced individuals. Operations such as recombination, crossover, mutation, selection, and adaptation are involved in this process [15]. The renowned

* Corresponding author. Tel.: +60 12 7714885; fax: +60 3 89216146.
E-mail addresses: hussain_ln@yahoo.com (H. Shareef), asrul@eng.ukm.my (A.A. Ibrahim), ammar_alqiesy@yahoo.com (A.H. Mutlag).

http://dx.doi.org/10.1016/j.asoc.2015.07.028
1568-4946/

paradigms of EAs are the genetic algorithm (GA) [15], evolutionary programming [16], differential evolution [17], evolutionary strategy [16], and genetic programming [18]. These algorithms are based on the principles of Darwinian theory and evolution theory of living beings. However, each algorithm follows specialized recombination, crossover, mutation, selection, and adaptation strategies. Similar to other metaheuristic algorithms, the aforementioned methods also have some drawbacks, such as slow convergence rate, difficulty in solving multimodal functions, and stagnation in local minima [19–21]. Advanced versions of EAs have been developed in recent years to improve the efficiency and performance of the aforementioned EAs; these advanced algorithms include stud genetic algorithm [21], fast evolutionary programming [20], adaptive differential evolution algorithm [22], and covariance matrix adaptation evolution strategy [23]. Not all algorithms and their variants provide superior solutions to some specific problems. Therefore, new heuristic optimization algorithms must be continuously searched to advance the field of computational intelligence optimization.

This study aims to introduce a novel metaheuristic optimization method called the lightning search algorithm (LSA) to solve constraint optimization problems. The LSA is based on the natural phenomenon of lightning. The proposed optimization algorithm is generalized from the mechanism of step leader propagation. It considers the involvement of fast particles known as projectiles in the formation of the binary tree structure of a step leader. Three projectile types are developed to represent the transition projectiles that create the first step leader population *N*, the space projectiles that attempt to become the leader, and the lead projectile that represent the best positioned projectile originated among *N* number of step leaders. The probabilistic nature and tortuous characteristics of lightning discharges, which depend on the type of projectile, are modeled using various random distribution functions. The LSA is explained in depth in Section 4.

2. Highlights of recent nature-inspired optimization algorithms

Many real-world optimization problems involve non-linearities and complex interactions among problem variables. Therefore, the capacity of nature-based algorithms to solve different optimization problems effectively and efficiently must be increased. Increasing the problem-solving capacity of these algorithms is generally achieved by modifying existing algorithms, hybridizing algorithms, and developing new algorithms. Several computational intelligence optimization algorithms have been proposed to overcome the limitations of their predecessors. The following descriptions highlight the recent methods published in the scientific literature.

2.1. Bat algorithm

The bat-inspired algorithm is a metaheuristic optimization algorithm developed by Xin-She Yang in 2010 [24]. The bat algorithm is based on the echolocation behavior of microbats with varying pulse rates of emission and loudness. Each virtual bat flies randomly with a velocity v_i at position (solution) x_i with a varying frequency or wavelength and loudness A_i. As the bat searches and finds its prey, the frequency, loudness, and pulse emission rate r are modified using a frequency-tuning technique to control the dynamic behavior of a swarm of bats. Search is intensified by a local random walk. The update of the velocities and positions of bats is similar to the standard updating procedure of PSO [24]. However, the bat algorithm features an intensive local search control, which is implemented by adjusting the loudness and pulse rate. The selection of the best continues until certain stop criteria are met. The main drawback of the standard bat algorithm is the abrupt switching to the exploitation stage by quickly varying A and r. This quick variation may lead to stagnation after the initial stages [25]. Many researchers have recognized this limitation of the bat algorithm and provided strategies to enhance the performance of this algorithm [26–28]. These strategies include using fuzzy logic [26], chaotic sequence [27], deferential operator, and Levy flight concepts [27,28].

2.2. Firefly algorithm (FFA)

The FA is a novel nature-inspired metaheuristic algorithm that is used to solve continuous multi-objective optimization problems based on the social behavior of fireflies [29]. The FA is an efficient technique for searching the Pareto optimal set, and its success rate and efficiency are better than those of PSO and the GA for both continuous and discrete problems [30]. The standard FA involves two important issues, namely, variation of light intensity I and formulation of attractiveness β. The attractiveness between fireflies is formulated as a function of the square of distance r between each other and the light absorption coefficient γ. As the fireflies search for the best solution, their movements are updated based on their current position, attractiveness, and a randomization term. When γ tends to be zero, the FA corresponds to the standard PSO [30]. However, the FA is superior to other algorithms because it is capable of automatic subdivision and dealing with multimodality [31].

2.3. Backtracking search algorithm (BSA)

Modeled based on the EA, the BSA is aimed at solving problems that are frequently encountered in EAs, such as excessive sensitivity to control parameters and premature convergence. Similar to the conventional EA, the BSA involves five processes, namely, initialization, initial selection, mutation, crossover, and second selection. In the initial selection, the BSA calculates the historical population as an indicator of the search direction. It can also redefine historical population at the beginning of each iteration in such a way that a population belonging to a randomly selected previous generation acts as a memory until it is changed. The mutation and crossover strategies of the BSA are different from those of the EA and its advanced versions. In generating trail population, only one parameter is used to control the amplitude of the search direction matrix in the mutation phase. However, crossover is complex. Two strategies are used to generate the final trial population. The first strategy uses mix rate to control the number of elements of individuals that will mutate in a trial, and the second strategy allows only one randomly chosen individual to mutate in each trial. In the second selection stage of the BSA, the population is updated using greedy selection, in which only individuals with high fitness values in the trail population are used. Despite its simple structure, the BSA may be memory and time consuming in computation because of the use of the dual population algorithm [32].

2.4. Krill herd algorithm (KHA)

The KHA is a newly developed swarm intelligence approach for optimization tasks based on the herding behavior of krill individuals [33]. It is a population-based method that consists of an idealized swarm of krills, each of which travels through a multi-dimensional search space to search for food. In the KHA, the positions of krill individuals are considered as different design variables, and the distance of the food from the krill individuals is equivalent to the fitness value of the objective function. In addition, krill individuals alter their position and travel to better positions. The position of each individual is affected by three principal processes: (i) movement affected by other krill individuals, (ii) foraging