Applied Soft Computing 48 (2016) 230–239

Contents lists available at ScienceDirect

Applied Soft Computing

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

On the convergence of multivariant optimization algorithm

Baolei Li[a,b], Jianhua Chen[b], Xinling Shi[b,*], Yufeng Zhang[b], Danjv Lv[b], Lanjuan Liu[b], Qinhu Zhang[b]

[a] Oil Equipment Intelligent Control Engineering Laboratory of Henan Province, Physics and Electronic Engineering College, Nanyang Normal University, Nanyang 473061, China
[b] Department of Electronic Engineering, Information School, Yunnan University, Kunming 650091, Yunnan, China

ARTICLE INFO

Article history:
Received 19 April 2014
Received in revised form
18 December 2015
Accepted 1 July 2016
Available online 20 July 2016

Keywords:
Multivariant optimization algorithm
Convergence analysis
Multivariant search groups
Global optimization

ABSTRACT

In this paper, we introduce a new global optimization method and study its global convergence property through theoretical and experimental approaches. The proposed method is named as multivariant optimization algorithm (MOA) because the intelligent searchers, which are called as atoms, not only are divided into multiple subgroups but also are variant in responsibility. That is, global atoms explore the whole solution space in the hope of finding potential areas where local atoms start the local exploitation. The proposed method is characterized by two important features. On one hand, global atoms do the global exploration in each loop to jump out from local traps. On the other hand, global and local atoms conduct the global exploration and the local exploitation according to their own responsibility, respectively. These features contribute to increasing the chance of converging to the global best. To study the convergence property of MOA, we carried out the convergence analysis, numerical optimization experiments and the shortest path planning experiments. And the results demonstrate that MOA is globally convergent and superior to the compared methods in the global convergence accuracy and probability in solving complex challenging problems which have one or more features such as deceptiveness, randomly located optimum, asymmetry or multiple traps.

1. Introduction

Global optimization algorithms are designed to deal with optimization problems with multiple local extremal solutions [1]. In this study, we address the solution of the global optimization problem $\min f(\vec{x})$ where $\vec{x} \in \mathbb{R}^N, f : \mathbb{R}^N \to \mathbb{R}$ is a given objective function, N is the dimension of a problem. A global optimization algorithm is utilised to seek a solution $\vec{x}^* \in \mathbf{S} \subseteq \mathbb{R}^N$ such that $f(\vec{x}^*) \leq f(\vec{x}), \forall \vec{x} \in \mathbf{S}$, where the solution space $\mathbf{S}$ is some region of $\mathbb{R}^N$. Such a solution $\vec{x}^*$ is called a global minimum or a global optimum. A solution $\vec{x}'$ is a local minimum (trap) in a local neighborhood $\mathbf{S}_0 \subset \mathbf{S}$ if $f(\vec{x}') \leq f(\vec{x})$, $\forall \vec{x} \in \mathbf{S}_0$.

The "basic" versions of particle swarm optimization (PSO), genetic algorithm (GA), differential evolution (DE) and firefly algorithm (FA) [2] are widely regarded as classical global optimization algorithms, since they have many advantages such as briefness, compact and fast convergence speed [3,4]. A compact differential evolution (cDE) algorithm ne-cDE was proposed to efficiently perform an optimization process despite a limited memory requirement in [5]. However, a high likelihood of being trapped into local optima limits their performance [6]. The proper control of the global exploration and the local exploitation is critical to the global convergence probability of a global optimization algorithm [7]. However, there exists a contradiction between the global exploration and the local exploitation because a single swarm is responsible for both the global exploration and the local exploitation, in these "basic" algorithms [8]. As a result, a user has to balance the contradiction carefully to achieve a good performance [9].

Focusing on increasing the probability of locating the global optimum among numbers of local traps, different strategies were proposed and discussed. Most studies address the performance improvement of PSO, GA or DE through endowing with auxiliary local search, parameter adaptation or scalability strategies, which are briefly reviewed below. *(1) Auxiliary local search:* Dynamic multi-swarm PSO with harmony search (DMS-PSO-HS) is developed through combining the exploration capabilities of the dynamic multi-swarm particle swarm optimizer (DMS-PSO) with the stochastic exploitation of the harmony search (HS) algorithm [10]. A probabilistic memetic framework (APrMF) is a probabilistic memetic algorithm, which is able to analyze the probability of evolution or individual learning [11]. Global and local real-coded GA (HRCGA) is proposed by means of the parent-centric crossover

* Corresponding author.
E-mail address: xlshi@ynu.edu.cn (X. Shi).

http://dx.doi.org/10.1016/j.asoc.2016.07.001
1568-4946/

operators [12]. *(2) Parameter adaptation:* A new PSO version with adaptive ω, η_1, and η_2, called adaptive PSO(APSO), was proposed by Zhan et al., recently [13]. In APSO, four evolutionary states, including exploitation, exploration, convergence, and jumping out, are defined. Self-learning PSO (SLPSO) is an improved version of APSO. In SLPSO, each particle has a set of four strategies to cope with different situations in the search space [14]. By considering a time-varying population topology, the velocity update mechanism in fully informed PSO [15], and a decreasing inertia weight, Frankenstein's PSO (FPSO) was proposed in [16]. The covariance matrix adaptation evolution strategy algorithm (G-CMA-ES) employs a restart mechanism and an increasing population size strategy [17]. J-adaptive differential evolution (JaDE) is proposed by implementing a new mutation strategy "DE/current-to-pbest" with optional external archive and updating control parameters in an adaptive manner [18]. By means of learning from their previous experiences in generating promising solutions, a self-adaptive differential evolution (SaDE) algorithm, in which both trial vector generation strategies and their associated control parameter values are gradually self-adapted, was proposed [19]. *(3) Scalability:* comprehensive learning particle swarm algorithm (CLPSO) was proposed by using a novel learning strategy whereby all other particles' historical best information is used to update a particle's velocity [20].

These improved methods, have shown a superiority over the "basic" algorithms in solving problems with lots of local traps [21–23,10]. However, the attractiveness in these improved methods may mislead intelligent searchers to move into a local optimum when solving deceptive functions. The reason is that the global minimum lies in a very narrow basin of attraction and at the same time there exists a strong local minimum with a wide basin of attraction in deceptive functions such as the Damavandi function [24]. So, these improved methods have difficulty in locating the global minimum of deceptive functions.

In this paper, a new "basic" heuristic search framework named multivariant optimization algorithm (MOA) is introduced to increase the global convergence probability in solving complex challenging problems. In MOA, the intelligent searchers (named as atoms) are divided into two categories according to their different responsibilities. They collaborate to search the solution space based on the historical searching information which is obtained by atoms and managed by a data structure. The data structure remembers and shares the historical useful searching information selectively. In each loop of MOA, a group of global atoms explore the whole solution space to locate areas which are more potential than those remembered by the data structure. Then, a group of local atoms exploit each potential area remembered by the data structure for a local refinement. After enough numbers of iteration, the global optimum is recorded in the data structure. From the brief description of MOA, it can be seen that a feature of this proposed method is that the global exploration is executed in each loop, which lessens the probability of being trapped into the local optima. Another is that multivariant search groups carry out the global exploration and the local exploitation respectively, which settles the contradiction between the global exploration and the local exploitation. These features make it a well-suited approach for solving global optimization problems with multiple local optima.

The purpose of this work is twofold. First, a new "basic" stochastic heuristic global optimization method was proposed. Further, we study the global convergence property of the proposed method through the convergence analysis, numerical optimization experiments and the shortest path planning experiments.

In the following sections, we describe the MOA method and prove that MOA is globally convergent. Then, MOA is compared with several state-of-the-art algorithms on twenty-three complex benchmark functions. The results suggest that MOA has a better performance in the global convergence accuracy and probability than the compared methods in solving complex challenging problems which are characterised by deceptiveness, randomly located optimum, asymmetry or multiple traps. Finally, MOA is used to solve the shortest path planning problems to assess the convergence property of MOA in application oriented problems.

2. Multivariant optimization algorithm

In this section, we present the MOA method for solving global optimization problems, introduce the data structure used to manage the communication and cooperation among multivariant search groups and describe the search strategy of MOA.

Without loss of generality, we take the solution space $\mathbf{S}$ as the hyperrectangle $\mathbf{S} = \{\vec{x} = (x_1, \ldots, x_N) | \min_i \leq x_i \leq \max_i, (i = 1, \ldots, N)\}$ where $\min_i$ and $\max_i$ are the lower and upper bounds of the ith dimension of the solution space, respectively. The global optimization problem considered in this paper is: find $\vec{x}^* = argmin\{f(\vec{x}) | \vec{x} \in \mathbf{S}\}$, where f is a given objective function.

In MOA, intelligent searchers called atoms search the solution space through cooperating with each other based on a data structure illustrated in Fig. 1. To simplify the description, we name the horizontal and vertical sorted doubly linked list as the queue and stack, respectively.

MOA is a stochastic heuristic optimization algorithm, where each iteration consists of two phases: a global exploration phase and a local exploitation phase. In the global exploration phase, global atoms explore the whole solution space randomly to produce a diverse set of potential areas. In the local exploitation phase, local atoms exploit each potential areas gained in the previous search for local refinements. The pseudo-code of MOA is outlined in Table 1, where

- *TL* is a temporary list used to record the newly generated global atoms and their fitness values;
- *TN* is a temporary node used to record a newly generated local atoms and its fitness value;
- *DS* is the used date structure;
- $DS(i, j)$ is the jth node from the top in the ith stack;
- $DS(i, j)$.A is an atom recorded in $DS(i, j)$ and $DS(i, j)$.Fv is its fitness value;

In step 1, new global search atoms explore the whole solution space according to:

$$atom_g = \{unifrnd(\min_1, \max_1), \ldots, unifrnd(\min_N, \max_N)\} \quad (1)$$

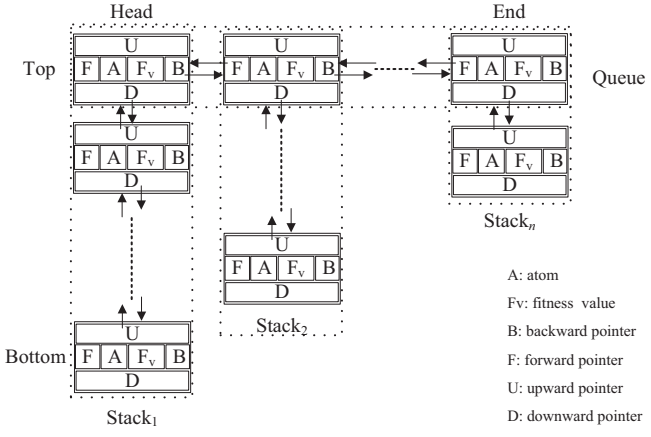

Fig. 1. Data structure of multivariant optimization algorithm.