Information Sciences 430–431 (2018) 87–103

Contents lists available at ScienceDirect

Information Sciences

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

Modifying the gravitational search algorithm: A functional study

M. Minárová [a], D. Paternain [b], A. Jurio [b], J. Ruiz-Aranguren [b], Z. Takáč [a], H. Bustince [b,*]

[a] *Slovak University of Technology in Bratislava, Radlinského 9,11, Bratislava 81005, Slovakia*
[b] *Public University of Navarra and Institute of Smart Cities, Campus Arrosadia s/n, 31006 Pamplona, Spain*

ARTICLE INFO

Article history:
Received 3 May 2017
Revised 27 September 2017
Accepted 17 November 2017
Available online 22 November 2017

Keywords:
Gravitational search algorithm
Aggregation function
t-norm, overlap function

ABSTRACT

In this paper we replace the product of the masses in the Gravitational search algorithm introduced by [17] by other bivariate functions with specific properties. We analyze the properties of these functions to guarantee convergence in the algorithm and we discuss an application to justify our theoretical study and the need of using functions other than the product.

1. Introduction

The development of algorithms inspired in the behavior of nature has a long history in artificial intelligence. In a short and non-exhaustive summary, we can recall evolutionary algorithms [1] or perceptron and neural networks [8].

Regarding physical laws, it is worth to mention that by the year 1977, Wright proposed an adaptation of Newton's Law of Gravitation for clustering problems [17]. More recently, in [16], an optimization algorithm which makes use of the gravitational law was proposed, in order to approach to the maximum (or the minimum) of a given function. The idea for this algorithm comes from two of the most relevant physical laws in the Newtonian framework. A simplified (and modified) form of the gravitational law, on the one hand, and the second law of dynamics, on the other hand. Roughly speaking, the Gravitational Search Algorithm (GSA) is an heuristic optimization algorithm where agents attract each other based on their particular masses. These masses are recalculated at each step according to the fitness evaluation of the agent.

It is worth mentioning that the gravitational law has also been used in image processing in the problem of edge detection. For instance, in [10] it is shown how the classical expression of the gravitational law can be modified in order to improve results of the edge detection algorithms. The good results provided by such a modification has suggested us to do a similar change, replacing the product of the masses in Newton's gravitational law by more general expressions in order to increase and/or modify the movements of the particles, so that the results of the algorithm may be increased. More specifically, our goals in this paper are the following:

* Corresponding author.
E-mail addresses: maria.minarova@stuba.sk (M. Minárová), daniel.paternain@unavarra.es (D. Paternain), aranzazu.jurio@unavarra.es (A. Jurio), zdenko.takac@stuba.sk (Z. Takáč), bustince@unavarra.es (H. Bustince).

https://doi.org/10.1016/j.ins.2017.11.033
0020-0255/

1. To propose an extension of the GSA algorithm by replacing the product of the masses in the calculation of the attracting force between particles by a more general function H, with suitable conditions;
2. to analyze the conditions under which the new GSA obtained by replacing the product by a function H is convergent;
3. to study the properties of the functions H which allow us to get the most suitable solution of an optimization problem.

In order to show the validity and usefulness of our generalized algorithm we discuss an example of application in an optimization problem which arises in image processing. Specifically, we consider the image compression algorithm [11–13] which makes use of the bidimensional fuzzy transform [14,15]. The goal of this kind of algorithms is to preserve as much information as possible, and the quality of the algorithm may be measured by first compressing the image, and then uncompressing again the result and comparing it with the original image. In this sense, the advantage of the bidimensional fuzzy transform is that it can be inverted, so both compression and uncompression can be defined in terms of it. For this reason, we propose as an example of application to optimize the partition upon which the bidimensional fuzzy transform is defined in order to minimize the mean squared error between the original image and the reconstructed one.

The paper is organized as follows: we start with some preliminary results and notations. In the next section, we recall the GSA and then we discuss our modification of the algorithm. Then we analyze the conditions that our functions H must fulfill so that we have convergence, and we display two particular cases: t-norms and overlap functions. Finally, we introduce an image application which justifies the modifications in the original GSA. We finish with some conclusions and references.

2. Preliminaries

2.1. Mathematical concepts and notations

We will denote by $\vec{X} = (x^1, \dots, x^k)$ a vector in the Euclidean space $\mathbb{R}^k$. By $\|\vec{X}\|$, we denote the Euclidean L^2 norm of the vector $\vec{X}$, i.e.

$$\|\vec{X}\| = \sqrt{(x^1)^2 + \cdots + (x^k)^2}$$

When we say that a sequence of vectors $\{\vec{X}_n\}$ converges to some other vector $\vec{X}_0$, we mean that it does so in the Euclidean norm. Note that $\vec{X}_n \to \vec{X}_0$ as $n \to \infty$ if and only if $x_n^d \to x_0^d$ as $n \to \infty$ for every $d \in \{1, \dots, k\}$.

For the sake of the utilizing along the entire paper, we at first introduce the functions that mathematically represent the decision making, the aggregation functions and their special types.

Definition 2.1. [3] An *aggregation function* is a function of $n > 1$ arguments that maps the (n-dimensional) cube onto an interval $\mathbb{I} = [a, b]$, $f : \mathbb{I}^n \to \mathbb{I}$, with the properties

(1) $f\,(\underbrace{a, a, \dots, a}_{n-times}) = a$ and $f\,(\underbrace{b, b, \dots, b}_{n-times}) = b$
(2) $\mathbf{x} \le \mathbf{y}$ implies $f(\mathbf{x}) \le f(\mathbf{y})$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{I}^n$.

Arithmetic, geometric, harmonic mean, minimum, maximum, product functions, etc. stand as examples of the aggregation functions. As a special type of the bivariate aggregation function, the t-norm and the overlap functions can be performed.

Definition 2.2. T-norm (triangular norm) is a function T: $[0, 1]^2 \to [0, 1]$ with the following properties:

(1) Commutativity: $T(a, b) = T(b, a)$
(2) Monotonicity: $T(a, b) \le T(c, d)$ if $a \le c$ and $b \le d$
(3) Associativity: $T(a, T(b, c)) = T(T(a, b), c)$
(4) Number 1 acts as identity element: $T(a, 1) = a$.

- A continuous t-norm is said to be Archimedean if $T(x, x) < x$ for all $x \in]0, 1[$.
- A strictly increasing continuous t-norm is called strict t-norm.

Remark 2.3. It is easy to overlook that the Łukasiewicz norm, defined as $T_L(a, b) = max\{0, a + b - 1\}$ is an example of t-norm.

Definition 2.4. [3] A mapping G_O: $[0, 1]^2 \to [0, 1]$ is an overlap function if it satisfies the following conditions:

(1) G_O is symmetric
(2) $G_O(x, y) = 0$ if and only if $xy = 0$
(3) $G_O(x, y) = 1$ if and only if $xy = 1$
(4) G_O is increasing
(5) G_O is continuous.

The product $G_O(x, y) = xy$ and the geometric mean $G_O(x, y) = \sqrt{xy}$ are examples of the overlap functions. The relations between t-norms and overlap functions can be performed by the following theorems [4].

Theorem 2.5. *Let $G_O(x, y)$ be an associative overlap function. Then $G_O(x, y)$ is a t-norm.*

Theorem 2.6. *If a t-norm T is an overlap function, then T belongs to one of the following three types:*