Knowledge-Based Systems 104 (2016) 134–144

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Locally informed gravitational search algorithm

Genyun Sun[a,*], Aizhu Zhang[a], Zhenjie Wang[a], Yanjuan Yao[b], Jinsheng Ma[c], Gary Douglas Couples[c]

[a] School of Geosciences, China University of Petroleum (East China), Qingdao Shandong, 266580, China
[b] Satellite Environment Center (SEC), Ministry of Environmental Protection (MEP) of China, Beijing, 100094, China
[c] Institute of Petroleum Engineering (IPE), Heriot-watt University, Edinburgh Scotland, EH14 2AD, United Kingdom

ARTICLE INFO

Article history:
Received 27 May 2015
Revised 29 March 2016
Accepted 17 April 2016
Available online 22 April 2016

Keywords:
Gravitational Search Algorithm (GSA)
Environmental heterogeneity
k-neighborhood local search
Locally informed learning

ABSTRACT

Gravitational search algorithm (GSA) has been successfully applied to many scientific and engineering applications in the past few years. In the original GSA and most of its variants, every agent learns from all the agents stored in the same elite group, namely K_{best}. This type of learning strategy is in nature a fully-informed learning strategy, in which every agent has exactly the same global neighborhood topology structure. Obviously, the learning strategy overlooks the impact of environmental heterogeneity on individual behavior, which easily resulting in premature convergence and high runtime consuming. To tackle these problems, we take individual heterogeneity into account and propose a locally informed GSA (LIGSA) in this paper. To be specific, in LIGSA, each agent learns from its unique neighborhood formed by k local neighbors and the historically global best agent rather than from just the single K_{best} elite group. Learning from the k local neighbors promotes LIGSA fully and quickly explores the search space as well as effectively prevents premature convergence while the guidance of global best agent can accelerate the convergence speed of LIGSA. The proposed LIGSA has been extensively evaluated on 30 CEC2014 benchmark functions with different dimensions. Experimental results reveal that LIGSA remarkably outperforms the compared algorithms in solution quality and convergence speed in general.

1. Introduction

Evolutionary algorithms and population-based optimization algorithms have been widely used for solving various optimization problems in the past decades [4,5,7–9,11,14,20,34]. Gravitational search algorithm (GSA) is one of the latest population-based optimization algorithms, which is inspired from the Newton's law of gravity and motion[29]. In GSA, the performance of an agent is measured by its mass. The heavy masses correspond to good solutions. As Newtonian gravity states that "Every agent in the universe attracts every other agent with a force that is directly proportional to the product of their masses and inversely proportional to the square of the distance between them", the relevant force will cause a global movement of each agent towards those agents with heavier masses [25,41]. Hence, an agent can search for the global optimum iteratively by learning from all of the rest agents. In essential, it is a type of fully-informed learning strategy in nature, which makes GSA has an outstanding property: diverse search directions.

Although the fully-informed learning strategy is simple in theory and easy to use, it easily causes two problems: 1) suffering from high runtime consuming [2] and 2) performing a poor tradeoff between exploration and exploitation [29]. On one hand, for a population with N agents, to obtain the force of an agent exerted by the rest agents, N–1 times distance should be calculated. Consequently, performing one iteration in the population, $N(N-1)$ times distances between agents need to be computed, which results in high runtime consuming [2]. On the other hand, the fully-informed learning strategy makes each agent learns from the rest agents in all the time, which means every agent exactly has the same global neighborhood topology structure [40]. This type of global structure overly emphasis on exploitation and offends against the basic rules of population-based optimization algorithm: to achieve well balance, exploration must fade out and exploitation must fade in by the lapse of time [29, 31].

To obtain compromise between exploration and exploitation, a K_{best} model is employed in original GSA. The K_{best} model stores those superior agents after fitness sorting in each iteration. The size of K_{best} is a function of time, which is set to N at the beginning and linearly decreases with time down to one. In such a way, each agent is guided by the rest agents at the beginning while only by one agent at the end [29].

* Corresponding author. Tel.: +8613706307251.
E-mail address: genyunsun@163.com (G. Sun).

http://dx.doi.org/10.1016/j.knosys.2016.04.017

Although the K_{best} model plays a certain effect, some problems still remain. On one hand, the overall computational time of GSA is still high as the size of K_{best} decreasing slowly. On the other hand, in the later stages, each agent can only learn from few elite agents, which easily causes quick loss of search diversity and false convergence. In this case, once the prematurity occurs, the population will trap into local optima because there are no remedies. Moreover, this model weakens the role of the global best agent due to the fact that all the elite agents have the equal status in K_{best}. Especially, the historically global best agent is discarded once the population is updated. GSA therefore ignores the importance of the global best agent in guaranteeing the convergence speed and accuracy [27]. The biggest problem lies in the topology structure of K_{best} model that is still a global neighborhood topology structure, in which every agent learns form the same group of elite. Due to the single topology structure, GSA overlooks the influence of environmental heterogeneity on individual behavior.

In the past few years, many researches have focused on improving GSA. One active research trend is to introduce some new operators into the original GSA. In [32], a disruption operator was employed to further explore and exploit the search space. Then, Shaw et al. [33] used opposition-based learning to perform population initialization and generation jumping, and improved the exploitation ability of GSA in the last iterations. In [10], the Black Hole theory was utilized to prevent premature convergence and to improve the exploration and exploitation abilities of GSA. Another active research trend is to combine some state-of-art heuristic optimization algorithms with GSA. For example, Li et al. [22] integrated Differential Evolution (DE) into GSA to overcome the premature convergence existing in unconstrained optimization. Sun et al. [35] presented a hybrid GA and GSA (GAGSA) to overcome the premature convergence problem. In addition, the memory of particle swarm optimization (PSO) has been introduced to GSA for constructing some more promising variants of GSA. In the PSOGSA [25,26] and GGSA [27], social thinking was introduced to GSA to accelerate convergence speed in the last iterations. In gravitational particle swarm [37] and modified GSA [13], the movement of each agent is determined by velocity of PSO and acceleration of GSA. In improved GSA [16,17], both the chaotic perturbation operator and memory of the position of each agent were utilized. The chaotic operator can enhance its global convergence to escape from local optima, and the memory strategy provides a faster convergence and shares individual best fitness history to improve the search ability.

Essentially, most of the GSA variants mentioned above are presented to enhance the search performance of GSA by designing new learning strategies or promoting the population diversity. However, most of them treat every agent equally, i.e., every agent learns from the same elite group stored in the K_{best}. In other words, the sight range of each agent is exactly the same, which disregarding the local environment of agents and easily resulting in premature convergence and high runtime consuming.

The aforementioned issues prompt us to explore the effect of environmental heterogeneity on individual behavior and proposed a GSA variant called locally informed GSA (LIGSA). The novelties of LIGSA are in two areas as follows.

(1) A locally informed learning strategy is proposed. The environmental heterogeneity is taken into account by constructing unique local neighborhood for each agent. Learning from the k local neighbors promotes LIGSA fully explore the regions around each agent with low computational complexity as well as effectively prevent premature convergence.
(2) Historical experience of the population is introduced to GSA. Each agent can learn from the historically global best agent directly. This makes the historically global best agent play a remarkable role for guiding the convergence process. Thereby the convergence speed of LIGSA is accelerated.

The remainder of this paper is organized as follows. Section 2 briefly describes the framework of GSA as well as discusses the fully-informed learning mechanism of GSA. In Section 3, a detail introduction of the proposed LIGSA is given. The comparison experimental results and discussion are presented in Section 4. Finally, a conclusion is given in Section 5.

2. Overview of GSA

2.1. GSA framework

In GSA, every agent $\boldsymbol{x}_i = [x_{i1}, x_{i2}, \ldots, x_{iD}]$ $(i = 1, 2, \ldots, N)$ attracts each other by gravitational force in a D-dimensional search space according to the law of gravity [29]. The corresponding velocity of agent i is $\boldsymbol{v}_i = [v_{i1}, v_{i2}, \ldots, v_{iD}]$. Due to the force between two agents is directly proportional to their masses and inversely proportional to their distance, all the agents move towards those agents that have heavier masses [29,30]. The mass of each agent in generation t, denoted by M_i^t, is simply calculated by Eqs. (1) and (2) as follows:

$$mass_i^t = \frac{fit_i^t - worst^t}{best^t - worst^t}, \tag{1}$$

$$M_i^t = \frac{mass_i^t}{\sum_{j=1}^{N} mass_j^t}, \tag{2}$$

where fit_i^t represents the fitness value of the agent i in generation t. For a minimization problem, $worst^t$ and $best^t$ are defined in Eqs. (3) and (4) as follows:

$$best^t = \min_{j \in [j=1,2,\ldots,N]} fit_j^t, \tag{3}$$

$$worst^t = \max_{j \in [j=1,2,\ldots,N]} fit_j^t. \tag{4}$$

In an optimization problem, the force acting on the agent i from agent j at a specific time t is shown in Eq. (5) as follows:

$$F_{ij}^d(t) = G(t)\frac{M_i^t(t)M_j^t(t)}{R_{ij}^t(t) + \varepsilon}\left(x_{jd}^t(t) - x_{id}^t(t)\right), \tag{5}$$

where $G(t)$ is the gravitational constant in generation t, M_i^t and M_j^t are the gravitational mass of the agents i and j, x_{jd}^t is the position of the agent j and x_{id}^t represent the position of the agent i in the dth dimension, respectively. R_{ij}^t is the distance between the agents i and j, and ε is a small constant bigger than 0.

To give a stochastic characteristic to GSA, the total force that acts on the agent i in the dth dimension is set to be a randomly weighted sum of dth components of the forces exerted from other agents as shown in Eq. (6) as follows:

$$F_i^d(t) = \sum_{j=1, j\neq i}^{N} rand_j F_{ij}^d(t), \tag{6}$$

where $rand_j$ is a uniform random variable in the interval [0, 1].

Hence, by the law of motion, the acceleration of the agent i in generation t, and in the dth dimension, a_{id}^t, is given in Eq. (7) as follows:

$$a_{id}^t = \frac{F_{id}^t}{M_i^t}. \tag{7}$$

The gravitational constant, G, is initialized to G_0 at the beginning and decreases with time to control the search accuracy. It is defined in Eq. (8) as follows:

$$G(t) = G_0 \cdot e^{-\beta \frac{t}{T_{max}}}, \tag{8}$$