Chapter 27
Gravitational Search Algorithm:
Concepts, Variants, and Operators

Hossein Nezamabadi-pour
Shahid Bahonar University of Kerman, Iran

Fatemeh Barani
Higher Education Complex of Bam, Iran

ABSTRACT

During the last decades, several metaheuristics have been developed to solve complex engineering optimization problems which most of them have been inspired by natural phenomena and swarm behaviors. Metaheuristics are the most selected techniques to find optimal solution intelligently in many areas of scheduling, space allocation, decision making, pattern recognition, document clustering, control objectives, image processing, system and filter modeling, etc. These algorithms have promised better solutions in single and multi-objective optimization. Gravitational search algorithm (GSA) is one of the recent created metaheuristic search algorithms, which is inspired by the Newtonian laws of gravity and motion. GSA was first proposed by Rashedi et al. and in the short time it became popular among the scientific community and researchers resulting in a lot of variants of the basic algorithm with improved performance. This chapter book presents a detailed review of the basic concepts of GSA and a comprehensive survey of its advanced versions. We propose a number of suggestions to the GSA community that can be undertaken to help move the area forward.

1 INTRODUCTION

With the growth of computer technology, storage devices, and soft computing (SC) techniques, it is now very accessible to solve more difficult and complex real-world problems in the fields of system modelling and optimization. For years, the researchers have been looking into nature to find heuristic approaches for inspiration to handle complicated computational problems. Optimization is at the heart of various natural processes like Darwinian evolution. Over millions of years, each species had to adapt itself to fit to the environment it was in (Das et al., 2011). The evolutionary computation (EC) techniques as an

DOI: 10.4018/978-1-4666-9644-0.ch027

important paradigm of computational intelligence (CI) have been developed by mimicking biological evolution to accomplish complex real-world optimization. EC is a form of stochastic optimization search.

Optimization tasks are unavoidable in many disciplines ranging from arts and design, business and finance to science and engineering (Ong et al., 2009). Classical optimization algorithms cannot provide a suitable solution in so many complex fields due to the increase of search space with the problem size, dependency of these algorithms on initial solutions, etc. Therefore, solving these problems using classical techniques is impractical and this causes a growth interest in metaheuristic search algorithms (Talbi et al., 2002; Sarafrazi et al., 2012).

There are two distinct forms of nature-inspired EC techniques which are evolutionary algorithms (EA), and swarm intelligence (SI)-based algorithms (Karaboga et al., 2014; Davarynejad et al., 2014). EAs are metaheuristic global search methods and optimization algorithms modeled from natural genetic principles such as natural selection. The basic idea of natural selection is "Select the best, discard the rest". It means that better individuals get higher chance to survive. The important methods in the field of EAs are Genetic Algorithms (GA) proposed by Holland (1975), Evolutionary Programming (EP) proposed by Fogel et al. (1966), Evolutionary Strategies (ES) proposed by Rechenberg (1973) and Schwefel (1975), Genetic Programming proposed by Koza (1992) and Differential Evolution (DE) proposed by Storn and Price (1995).

Swarm Intelligence (SI) refers to a newly developed group of population-based algorithms for multi-agent search and optimization. SI studies the collective behavior of systems made up of a population of simple agents interacting locally with each other and with their environment. In the SI systems the agents follow very simple rules, although there is no centralized control structure dictating how individual agents should behave. Social interactions (locally shared knowledge) provide the basis for unguided problem solving. In recent years, the swarm intelligence metaheuristics have received tremendous attention in research, mainly as Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO). ACO is the first family of SI-based search algorithms which was proposed by Dorigo et al. (1991), where the behavior of ants is modeled in finding the shortest path from nest to food source. PSO, which was proposed by Kennedy and Eberhart (1995), mimics the flocking behavior of birds and fish. Some other successful instances of swarm intelligence metaheuristics are bees algorithm (Jung, 2003; Karaboga, 2005), bacterial foraging optimization (BFO) (Passino, 2002), monkey algorithms (Soleimanpour, 2013), and gravitational search algorithm (GSA) (Rashedi et al., 2009).The interested reader could follow refs. (Vasant, 2012; Vasant, 2014).

In the late-2000s, gravitational search algorithm was introduced by E. Rashedi and her colleagues (Rashedi, 2007a, 2007b, 2007c, 2007d, 2009) as a novel nature inspired population based metaheuristic for solving continuous (real-valued) optimization problems. GSA belongs to the class of metaheuristics which are stochastic global optimization methods. It is inspired by Newtonian laws of gravity and motion and notion of mass interactions. The GSA uses the theory of Newtonian physics and its searcher agents are the collection of objects. In GSA, there is an isolated system of objects. Using the gravitational force, every object in the system can see the situation of other objects. The gravitational force is therefore a way of transferring information between different objects (Rashedi et al., 2009). In other words, in GSA, searcher agents are considered as objects which their masses are considered corresponding to their performance. All these objects attract each other by a gravity force, and this force causes a movement of all objects globally towards the objects with heavier masses. The heavy objects correspond to good solutions of the problem. The position of the agent corresponds to a solution of the problem, and

Related Content