Expert Systems with Applications 42 (2015) 7831–7845

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Backtracking Search Algorithm with three constraint handling methods for constrained optimization problems

Chunjiang Zhang, Qun Lin, Liang Gao *, Xinyu Li

The State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, Wuhan, Hubei, China

ARTICLE INFO

Article history:
Available online 29 May 2015

Keywords:
Backtracking Search Algorithm
Constrained optimization problem
Feasibility and dominance rules
ε-constrained method
Engineering optimization

ABSTRACT

A new evolutionary algorithm, Backtracking Search Algorithm (BSA), is applied to solve constrained optimization problems. Three constraint handling methods are combined with BSA for constrained optimization problems; namely feasibility and dominance (FAD) rules, ε-constrained method with fixed control way of ε value and a proposed ε-constrained method with self-adaptive control way of ε value. The proposed method controls ε value according to the properties of current population. This kind of ε value enables algorithm to sufficiently search boundaries between infeasible regions and feasible regions. It can avoid low search efficiency and premature convergence which happens in fixed control method and FAD rules. The comparison of the above three algorithms demonstrates BSA combined ε-constrained method with self-adaptive control way of ε value (BSA-SAε) is the best one. The proposed BSA-SAε also outperforms other five classic and the latest constrained optimization algorithms. Then, BSA-SAε has been applied to four engineering optimization instances, and the comparison with other algorithms has proven its advantages. Finally, BSA-SAε is used to solve the car side impact design optimization problem, which illustrates the wide application prospects of the proposed BSA-SAε.

1. Introduction

The research to find better optimization algorithms in science and engineering has never stopped. Since these science and engineering optimization problems are getting more complicated with larger scales, even a little improvement in the algorithm can make a considerable difference.

Currently, more researchers focus on evolutionary algorithms (EAs), which have been successfully applied to various complex problems, especially for nonlinear, non-differentiable and non-convex objective function problems. EAs are inspired by the natural evolution to find the best fitness individuals. There are two major categories of EAs, namely swarm intelligence optimization algorithms and genetic evolution algorithms. Swarm intelligence optimization algorithms are inspired by the natural biological living and hunting activities. Ant colony algorithm (Dorigo, 1992), firefly algorithm (Yang, 2008), Artificial Bee Colony (ABC) algorithm (Karaboga, 2005), cuckoo search algorithm (Yang & Deb, 2009), and Particle Swarm Optimization (PSO) algorithm (Kennedy & Eberhart, 1995) are examples of swarm intelligence algorithms. Genetic evolution algorithms are mainly based on basic selection, crossover, and mutation operators, for example, Genetic Algorithm (GA) (Holland, 1975), Covariance Matrix Adaption Evolution Strategy (CMAES) (Hansen, Müller, & Koumoutsakos, 2003), and Differential Evolution (DE) algorithm (Storn & Price, 1995). Some other researchers hybridize the above two or more algorithms together to enhance the optimization performance of the algorithm, i.e. GA-PSO (Sheikhalishahi, Ebrahimipour, & Shiri, 2013), PSO-DE (Vaisakh, Praveena, Rama Mohana Rao, & Meah, 2012), etc.

Backtracking Search Algorithm (BSA) was first introduced by Civicioglu (2013). It shows encouraging performance in dealing with boundary-constrained benchmark problems. In the paper, he takes three constrained real-world benchmark problems (Antenna, Radar and FM) (Das & Suganthan, 2010) to illustrate BSA's ability to solve real-world problems. However, there are only boundary constraints in the three problems. Boundary constraints are usually handled by some boundary handle techniques such as cutting off. Moreover, it is not clear how the algorithm handles constraints and this remains an area to be explored.

Due to its promising optimization performance, BSA has been widely used to solve many kinds of engineering optimization problems in recent years. Song, Zhang, Zhao, and Li (2015) applied BSA for surface wave analysis. Guney, Durmus, and Basbug (2014) and El-Fergany (2015) applied BSA to solve concentric circular antenna array synthesis optimization problems and multi-type distributed

* Corresponding author.
E-mail addresses: zhchj1989@gmail.com (C. Zhang), linqunhust@hotmail.com (Q. Lin), gaoliang@mail.hust.edu.cn (L. Gao), lixinyu@hust.edu.cn (X. Li).

http://dx.doi.org/10.1016/j.eswa.2015.05.050
0957-4174/

generators allocation optimization problems. For non-convex economic dispatch problems, Modiri-Delshad and Rahim (2014) verified BSA's excellent robustness and high efficiency in dealing with this kind of large-scale problem. Askarzadeh and dos Santos Coelho (2014) combined BSA with Burger's chaotic map (BCM) to solve parameter estimation problems of PEMFC electrochemical model. Wang et al. (2015) replace the mutate and crossover strategies in BSA with the mutate and crossover strategies in DE to solve unconstrained optimization problems. For the linear antenna array interference suppression problems, Das, Mandal, and Ghoshal (2014) combined BSA with DE, and proposed BSA-DE. To further enhance the convergence speed of BSA, Lin (2014) adopted Opposition-Based Learning (OBL) strategy and proposed Oppositional Backtracking Search Algorithm (OBSA) to solve hyper chaos system parameter identification optimization problems.

As existing research mainly focuses on the application of BSA in specific examples, as far as we know, there are no other researches about BSA for constrained optimization in literature at present, especially for nonlinear, non-convex, non-differential constrained problems. To fill this research gap, this paper mainly focuses on how to apply SBA on constrained optimization problems.

The rest of this paper is organized as follows: Section 2 reviews the mathematic model of constrained problem and several classic constraint handling methods. The original Backtracking Search Optimization Algorithm (BSA) is introduced in Section 3. Section 4 presents three different BSA based constrained optimization methods, including BSA combined with feasibility and dominance rules (BSA-FAD), BSA combined with ε-constrained method with fixed control way of ε value (BSA-FCε), and BSA with ε-constrained method using a self-adaptive control way of ε value (BSA-SAε) which is first proposed in this paper. Section 5 describes the experiments and analyzes the results, including the comparison among the three BSA-based methods, the comparison between the best BSA-based method (BSA-SAε) and some classic or latest constrained optimization algorithms, and BSA-SAε's applications in four engineering optimization instances. A case study of car side impact design is shown in Section 6. Section 7 summarizes the conclusions of the paper.

2. Constrained optimization problem and constraint handling methods

In general, constrained optimization problems can be mathematically formulated as a minimization problem as follows.

Min $f(X)$

$$\text{Subject to} \begin{cases} g_i(X) \leqslant 0 & i = 1, 2, \ldots, m \\ h_j(X) = 0 & j = 1, 2, \ldots, q \end{cases}$$

where $X = [x_1, x_2, x_3, \ldots, x_D] \in R^D$ is the solution vector; m is the total number of inequality constraints, and q is the total number of equality constraints. Each x_k, $k = 1, 2, 3, \ldots, D$ is bounded by $low_k \leqslant x_k \leqslant up_k$ which defines the D-dimensional search space. The set of solutions that satisfy all constraints is called feasible region, in which the best solution will be found. Both the objective function and constraints can be linear, nonlinear, non-differential, or non-convex. Equality constraints can be usually transformed into inequality constraints as follows: $|h_j(X)| - \delta \leqslant 0$ where δ is a very small positive value (usually 10^{-4}), called allowance tolerance.

There are number of constraint handling methods proposed in the literature, and the most commonly used five methods are penalty functions, feasibility and dominance (FAD) rules, stochastic ranking (SR), ε-constrained methods and multi-objectives concepts.

1) Penalty function

Transforming constrained optimization problem into unconstrained problem is the core idea of the penalty function, whose general formula is listed as follows:

$$\phi(X) = f(X) + p(X) \tag{1}$$

$$p(X) = \sum_{i=1}^{m} r_i * \max(0, g_i(X))^2 + \sum_{j=1}^{q} c_j * |h_j(X)| \tag{2}$$

where $\phi(X)$ is the extended objective function, $p(X)$ is the penalty value defined by the inequality constraints $g_i(X)$ and quality constraints $h_j(X)$, r_i and c_j are positive constants called "penalty factors". The penalty value is added to the fitness function because low values are preferred as expected in a minimization problem.

Even though the penalty function is very simple and popular, it is not trivial to define a set of appropriate penalty factors. To circumvent this problem, researchers have proposed many variants of penalty function method to deal with penalty factor values, such as dynamic method (Paszkowicz, 2009), adaptive method (Tessema & Yen, 2009), co-evolved method (Coello Coello, 2000), fuzzy method (Wu, Yu, & Liu, 2001), and some other methods (Lin, 2013). Among these methods, the adaptive penalty function is one of the most competitive approaches.

2) Feasibility and dominance rules

Feasibility and dominance (FAD) rules were first proposed for binary tournaments in Genetic Algorithms (Deb, 2000). They are usually used in a selection progress to select feasible solution and the relatively good solution in infeasible solutions.

Simplicity and flexibility make FAD rules suitable to be used with any optimization algorithms. Pulido and Coello Coello (2004) combined FAD rules with a global-best PSO; Ullah, Sarker, and Cornforth (2009) adopted FAD rules in their agent based memetic algorithm; Saha, Datta, and Deb (2010) used FAD rules in a real-coded GA with simulated binary crossover and adaptive polynomial mutation. Zhang, Li, Gao, and Wu (2013) combined FAD rules with a modified Electromagnetism-like Mechanism algorithm. One of the main common drawback of FAD rules is that they are prone to premature convergence due to its intense preference to feasible solutions (Mezura Montes & Coello Coello, 2005).

3) Stochastic ranking (SR)

Stochastic ranking (SR) was proposed by Runarsson and Yao (2000), in which a user-defined parameter called P_f controls the criterion used for comparison of infeasible solutions. A bubble-sort-like process is used in SR to rank the solutions in the population. It is very easy to be understood, but it just can be applied in algorithms with sort progress for population.

4) ε-constrained method

ε-constrained method is proposed by Takahama, Sakai, and Iwane (2005). This method is used to select a better solution from two solutions as FAD rules. An ε value is set as threshold value in ε-constrained method when comparing two solutions. When the constraint violation values of both solutions are smaller than ε value, the one with better objective function value is selected. Otherwise, the one with smaller constraint violation value is selected.