Computer-Aided Design 45 (2013) 1562–1574

Contents lists available at ScienceDirect

Computer-Aided Design

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

A hybrid differential evolution augmented Lagrangian method for constrained numerical and engineering optimization

Wen Long [a,b,*], Ximing Liang [b,c], Yafei Huang [b], Yixiong Chen [b]

[a] Guizhou Key Laboratory of Economics System Simulation, Guizhou University of Finance & Economics, Guiyang, Guizhou 550004, PR China
[b] School of Information Science and Engineering, Central South University, Changsha, Hunan 410083, PR China
[c] School of Science, Beijing University of Civil Engineering and Architecture, Beijing 100044, PR China

HIGHLIGHTS

- A method hybridizing augmented Lagrangian multiplier and differential evolution algorithm is proposed.
- We formulate a bound constrained optimization problem by a modified augmented Lagrangian function.
- The proposed algorithm is successfully tested on several benchmark test functions and four engineering design problems.

ARTICLE INFO

Article history:
Received 8 April 2013
Accepted 29 July 2013

Keywords:
Constrained optimization problem
Differential evolution algorithm
Modified augmented Lagrangian multiplier method
Engineering optimization

ABSTRACT

We present a new hybrid method for solving constrained numerical and engineering optimization problems in this paper. The proposed hybrid method takes advantage of the differential evolution (DE) ability to find global optimum in problems with complex design spaces while directly enforcing feasibility of constraints using a modified augmented Lagrangian multiplier method. The basic steps of the proposed method are comprised of an outer iteration, in which the Lagrangian multipliers and various penalty parameters are updated using a first-order update scheme, and an inner iteration, in which a nonlinear optimization of the modified augmented Lagrangian function with simple bound constraints is implemented by a modified differential evolution algorithm. Experimental results based on several well-known constrained numerical and engineering optimization problems demonstrate that the proposed method shows better performance in comparison to the state-of-the-art algorithms.

1. Introduction

In real-world applications, most optimization problems are subject to different types of constraints. These problems are known as constrained optimization problems. In the minimization sense, general constrained optimization problems can be formulated as follows:

$$\begin{array}{lll} \min & f(\vec{x}) & \text{(a)} \\ s.t. & g_j(\vec{x}) = 0, \quad j = 1, 2, \dots, p & \text{(b)} \\ & g_j(\vec{x}) \le 0, \quad j = p+1, \dots, m & \text{(c)} \\ & l_i \le x_i \le u_i, \quad i = 1, 2, \dots, n & \text{(d)} \end{array} \qquad (1)$$

where $\vec{x} = (x_1, x_2, \dots, x_n)$ is a dimensional vector of n decision variables, $f(\vec{x})$ is an objective function, $g_j(\vec{x}) = 0$ and $g_j(\vec{x}) \le 0$ are known as equality and inequality constraints, respectively. p is the number of equality constraints and $m - p$ is the number of inequality constraints, l_i and u_i are the lower bound and the upper bound of x_i, respectively.

Evolutionary algorithms (EAs) have many advantages over conventional nonlinear programming techniques: the gradients of the cost function and constraint functions are not required, easy implementation, and the chance of being trapped by a local minimum is lower. Due to these advantages, evolutionary algorithms have been successfully and broadly applied to solve constrained optimization problems [1–10] recently. It is necessary to note that evolutionary algorithms are unconstrained optimization methods that need additional mechanism to deal with constraints when solving constrained optimization problems. As a result, a variety of EA-based constraint-handling techniques have been developed [11,12].

Penalty function methods are the most common constraint-handling technique. They use the amount of constraint violation to punish an infeasible solution so that it is less likely to survive into the next generation than a feasible solution [13]. The augmented Lagrangian is an interesting penalty function that avoids the side-effects associated with ill-conditioning of simpler penalty and barrier functions. Recent studies have used different augmented

* Corresponding author at: Guizhou Key Laboratory of Economics System Simulation, Guizhou University of Finance & Economics, Guiyang, Guizhou 550004, PR China. Tel.: +86 08516772696.
E-mail address: lw084601012@gmail.com (W. Long).

0010-4485/$ – see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cad.2013.07.007

Lagrangian multiplier methods with an evolutionary algorithm. Kim and Myung [14] proposed a two-phase evolutionary programming using the augmented Lagrangian function in the second phase. In this method, the Lagrangian multiplier is updated using the first-order update scheme applied frequently in the deterministic augmented Lagrangian methods. Although this method exhibits good convergence characteristics, it has been tested only for small-scale problems. Lewis and Torczon [15] proposed an augmented Lagrangian technique, where a pattern search algorithm is used to solve the unconstrained problem, based on the augmented Lagrangian function presented by Conn et al. [16]. Tahk and Sun [17] used a co-evolutionary augmented Lagrangian method to solve min–max problems by means of two populations of evolution strategies with annealing scheme. Krohling and Coelho [18] also formulated constrained optimization problems as min–max problems and proposed the co-evolutionary particle swarm optimization using Gaussian distribution. Rocha et al. [19] used an augmented Lagrangian function method along with a fish swarm based optimization approach for solving numerical test problems. Jansen and Perez [20] implemented a serial augmented Lagrangian method in which a particle swarm optimization algorithm is used to solve the augmented function for fixed multiplier values.

In the above approaches, the augmented Lagrangian functions were used to deal with the constraints in constrained optimization problems. However, penalty vectors were only considered as fixed vectors of parameter. They were given at the beginning of the algorithms and kept unchanged during the whole process of solution. It is difficult and very important to choose some good penalty vectors. In addition, Mezura-Montes and Cecilia [21] established a performance comparison of four bio-inspired algorithms with the same constraint-handling technique (i.e., Deb's feasibility-based rule) to solve 24 benchmark test functions. These four bio-inspired algorithms are differential evolution, genetic algorithm, evolution strategy, and particle swarm optimization. The overall results indicate that differential evolution is the most competitive among all of the compared algorithms for this set of test functions.

In this paper, we presented a modified augmented Lagrangian technique, where a differential evolution algorithm is used to solve the unconstrained problem, based on the augmented Lagrangian function proposed by Liang [22]. The basic steps of the proposed method comprise an outer iteration, in which the Lagrange multipliers and various penalty parameters are updated using a first-order update scheme, and an inner iteration, in which a nonlinear optimization of the modified augmented Lagrangian function with bound constraints is solved by a differential evolution algorithm.

The rest of this paper is organized as follows. In Section 2, the modified augmented Lagrangian formulation method is described. In Section 3, the proposed hybrid method is discussed in sufficient detail. Simulation results based on constrained numerical optimization and engineering design problems and comparisons with previously reported results are presented in Section 4. Finally, the conclusions are given in Section 5.

2. Modified augmented Lagrangian formulation

In nonlinear constrained engineering optimization, the problem size ranges from a few hundred to several thousands of variables and constraints. Currently, the most frequently used solution methods are the generalized reduced gradient methods, successive quadratic programming methods, and the modified barrier function methods. These approaches are based on the linearization techniques and can be applied to problems with either a few variables, when used in full space, or a few degrees of freedom, when used in reduced space. Also, the presence of many inequality constraints (and bounds) may make their active-set based strategies quite inefficient. The modified barrier function method, which transforms the originally constrained problem to a series of unconstrained ones, has finite convergence as opposed to asymptotic convergence for the classical barrier function methods and their barrier parameters need not be driven to zero to obtain the solution. But the case of equality constraints poses a serious difficulty on the method. All these methods start from an initial point and iteratively produce a sequence to approach some local solution to the studied problem. The purpose of this work is to utilize the modified augmented Lagrangian multiplier method for constrained problems (1).

In formula (1), if the simple bound (1)(d) is not present, then one can use the modified augmented Lagrange multiplier method to solve (1)(a)–(c). For the given Lagrange multiplier vector λ^k and penalty parameter vector σ^k, the unconstrained penalty sub-problem at the kth step of this method is

$$\text{Minimize } P(x, \lambda^k, \sigma^k) \tag{2}$$

where $P(x, \lambda, \sigma)$ is the following modified augmented Lagrangian function:

$$P(x, \lambda, \sigma) = f(x) - \sum_{j=1}^{p}\left[\lambda_j g_j(x) - \frac{1}{2}\sigma_j (g_j(x))^2\right] - \sum_{j=p+1}^{m} \tilde{P}_j(x, \lambda, \sigma) \tag{3}$$

and $\tilde{P}_j(x, \lambda, \sigma)$ is defined as follows:

$$\tilde{P}_j(x, \lambda, \sigma) = \begin{cases} \lambda_j g_j(x) - \frac{1}{2}\sigma_j (g_j(x))^2, & \text{if } \lambda_j - \sigma_j g_j(x) > 0 \\ \frac{1}{2}\lambda_j^2/\sigma_j, & \text{otherwise.} \end{cases} \tag{4}$$

It can be easily shown that the Kuhn–Tucker solution (x^*, λ^*) of the primal problem (1)(a)–(c) is identical to that of the augmented problem (2). It is also well known that, if the Kuhn–Tucker solution is a strong local minimum, then there exists a constant $\bar{\sigma}$ such that x^* is a strong local minimum of $P(x, \lambda^*, \sigma)$ for all penalty vector σ which component not less than $\bar{\sigma}$; the Hessian of $P(x, \lambda, \sigma)$ with respect to x near (x^*, λ^*) can be made positive definite. Therefore, x^* can be obtained by an unconstrained search from a point close to x^* if λ^* is known and σ is large enough.

If the simple bound (1)(d) is present, the above modified augmented Lagrange multiplier method needs to be modified. In modified barrier function methods, the simple bound constraints are treated as the general inequality constraints $x_i - l_i \geq 0$ and $u_i - x_i \geq 0$, which enlarges greatly the number of Lagrange multipliers and penalty parameters. So, we make another modification to deal with the bound constraints. At the kth step, assume that the Lagrange multiplier vector λ^k and penalty parameter vector σ^k are given; we solve the following bound constrained sub-problem instead of (2):

$$\begin{cases} \min P(x, \lambda^k, \sigma^k) \\ \text{s.t. } l_i \leq x_i \leq u_i \end{cases} \tag{5}$$

where $P(x, \lambda, \sigma)$ is the same modified augmented Lagrangian function as in (3). Let $S \subseteq R^n$ designate the search space, which is defined by the lower and upper bounds of the variables (1)(d). The solution x^* to sub-problem (5) can be obtained by searching the search space if λ^* is known and σ is large enough. We will choose the differential evolution algorithm for the global search in (5). The details are discussed below.