Expert Systems with Applications 42 (2015) 5976–5987

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Multi-objective optimization based reverse strategy with differential evolution algorithm for constrained optimization problems

Liang Gao, Yinzhi Zhou, Xinyu Li *, Quanke Pan, Wenchao Yi

State Key Laboratory of Digital Manufacturing Equipment & Technology, Huazhong University of Science and Technology, Wuhan 430074, PR China

ARTICLE INFO

Article history:
Available online 2 April 2015

Keywords:
Constrained optimization problems
Reverse model
Multi-objective optimization techniques
Differential evolution

ABSTRACT

Solving constrained optimization problems (COPs) has been gathering attention from many researchers. In this paper, we defined the best fitness value among feasible solutions in current population as *gbest*. Then, we converted the original COPs to multi-objective optimization problems (MOPs) with one constraint. The constraint set the function value $f(x)$ should be less than or equal to *gbest*; the objectives are the constraints in COPs. A reverse comparison strategy based on multi-objective dominance concept is proposed. Compared with usual strategies, the innovation strategy cuts off the worse solutions with smaller fitness value regardless of its constraints violation. Differential evolution (DE) algorithm is used as a solver to search for the global optimum. The method is called multi-objective optimization based reverse strategy with differential evolution algorithm (MRS-DE). The experimental results demonstrate that MRS-DE can achieve better performance on 22 classical benchmark functions compared with several state-of-the-art algorithms.

1. Introduction

In real world applications, many optimization problems, such as pressure vessel design problem (Hedar & Fukushima, 2006), welded beam design problem (Deb, 2000), can be formulated as constrained optimization problems (COPs). Without loss of generality, the general COPs can be modeled as follows (named as P):

$$\begin{aligned} &\min f(x) \\ (P)\quad &s.t.\ g_j(x) \leqslant 0, \quad j = 1, \ldots, q \\ &\quad h_j(x) = 0, \quad j = q+1, \ldots, m \end{aligned} \tag{1}$$

where $x \in R^n$, with the parametric constraints: $L \leqslant x \leqslant U$. L and U are lower and upper bound of variable x. The feasible region Ω can be defined as:

$$\Omega = \{x | x \in R^n, g_j(x) \leqslant 0,\ j = 1, \ldots, q,\ \ h_j(x) = 0,\ j = q+1, \ldots, m\} \tag{2}$$

For unconstrained optimization problems, meta-heuristic algorithms have proven its advantage over exact algorithms and have become mostly common used methods recently. However, almost all these meta-heuristic algorithms are designed for the unconstrained optimization problems. Therefore, the constraint-handling techniques have become the important supplements for the theory of meta-heuristic algorithms.

In the early years, the most common constraint-handling method is the penalty function method (Coello Coello, 2000; Smith & Coit, 1997). Its basic idea is to punish the infeasible solution by adding weighted penalty terms in objective function, so compared with feasible solution, the infeasible solution can barely survive into next iteration. The general penalty function formula is as the following:

$$\phi(x) = f(x) + \sum_{i=1}^{q} r_i \bullet \max(0, g_i(x))^2 + \sum_{j=q+1}^{m} c_j |h_j(x)| \tag{3}$$

where r_i and c_j are the positive constants called penalty factors. This approach converts the constraint problem to unconstraint problem. However, although different setting strategies have been proposed, how to determine the penalty factors reasonably is remained to be a challenge, which limits its application.

Unlike combining objective function and constraints into an unconstraint function, the idea of separating objective function and constraints has attracted many attentions and has made great impact on this area recently. In constrained optimization, the difficulty lays on how to evaluate the influence of constraint on function value. The idea of separating objective function and constraints provides a simple and efficient solution. There are several techniques that can be included into this filed, such as feasibility

* Corresponding author. Tel.: +86 27 87557742; fax: +86 27 87543074.
E-mail addresses: gaoliang@mail.hust.edu.cn (L. Gao), leitezhou@gmail.com (Y. Zhou), lixinyu@mail.hust.edu.cn (X. Li), panquanke@hust.edu.cn (Q. Pan), janety1989@gmail.com (W. Yi).

http://dx.doi.org/10.1016/j.eswa.2015.03.016
0957-4174/

rules, stochastic ranking, ε-constrained method, multi-objective concepts etc.

Feasibility rule proposed by Deb (2000) is a simple constraint-handling scheme for comparing two solutions. It includes three feasibility criteria:

(1) If one solution is feasible and another one is infeasible, the feasible solution is preferred to the infeasible solution;
(2) If two solutions are both feasible solutions, the one with better objective function will triumph;
(3) If two solutions are both infeasible solutions, the one with smaller degree of constraints violation will outperform the other one.

Stochastic ranking was proposed by Runarsson and Yao (2000). It provides a solution to the challenge of how to choose the proper penalty factors. It uses a self-defined probability parameter called P_f to control which criterion is used for comparison: based on their sum of constraint violation or based only on objective function value. Some future development can be seen in Zhang, Geng, Luo, Huang, and Wang (2006) and Mallipeddi, Suganthan, and Qu (2009).

Takahama and Sakai (2004) proposed an approach called the α–constrained method and improved the method to ε-constrained method (Takahama, Sakai & Iwane, 2005). It presents ε-level comparisons under the flame of feasibility rules. It modifies the second criteria listed above by relaxing the concept of feasible solution with ε value (not 0 in feasibility rules). Several variants of this method have been proposed by Takahama and Sakai (2006, 2008, 2013).

Multi-objective optimization techniques (MOTs) are relatively popular in recent literature (Mezura-Montes & Coello Coello, 2008). Its main idea is to convert the COPs into the unconstraint multi-objective optimization problems. Multi-objective concepts, including Pareto dominance (Coello Coello & Mezura-Montes, 2002; Liu, Zhong, & Hao, 2007), Pareto ranking (Reynoso-Meza, Blasco, Sanchis, & Martínez, 2010; Venter & Haftka, 2010) and non-dominated sorting (Ray, Singh, Isaacs & Smith, 2009), are utilized to tackle these multi-objective optimization problems. According to the number of objectives, these problems can be divided into bi-objective problems or many-objective problems.

In this paper, we propose a novel constraint-handling approach called multi-objective based reverse strategy (MRS). It converts the original constraints in (P) to the objectives, then sets the best feasible function value found at Gth generation as *gbest*, and adds $f(x) \leqslant gbest$ as the constraint to the new model. The new model can be regarded as a multi-objective problem with one constraint. The comparison criteria of solutions have been given in this article. Differential evolution (DE) algorithm is used to solve the COPs as optimizer. The details of the proposed approach will be provided in the following sections. Four experiments are conducted to evaluate the efficiency of the methods. The results show that the method has achieved significant improvement.

The rest of the paper is organized as follows: Section 2 describes the proposed MRS in detail; Section 3 introduces the classical DE algorithm after reviewing some modified DE algorithm for COPs, and then gives the working step of MRS-DE; In Section 4, the experimental results and the comparisons based on 22 benchmark problems are presented; Finally Section 5 concludes the paper with remarks.

2. Multi-objective Optimization based Reverse Strategy for Constraints Handling

In this section, a new constrained optimization method called multi-objective optimization based reverse strategy (MRS) is proposed.

2.1. Converted model

We change the objective function and constraints in (P) into constraints and objective function respectively (named as R).

$$\begin{aligned} &min\,\bar{f}(x) = \{\varphi(g_i(x),0), \varphi(|h_j(x)|,\delta), i=1,\ldots,q, j=q+1,\ldots,m\} \\ &(R) \\ &s.t.\, f(x) \leqslant gbest_G \end{aligned} \tag{4}$$

The explanation of this model is listed as follows: in each generation G, the best feasible function value is denoted as $gbest_G$. Set the objective function in next generation $(G+1)$ better than $gbest_G$, i.e. $f(x) \leqslant gbest_G$. $f(x)$ is the objective function value of (P) (in Section 1). We use this in equation as constraint. One point should be noted here is that $gbest_G$ is updated in each generation.

The constraints in (P) are converted to the objective functions, which can be expressed as:

$$\bar{f}(x) = \{\varphi(g_i(x),0), \varphi(|h_j(x)|,\delta), i=1,\ldots,q, j=q+1,\ldots,m\} \tag{5}$$

where $\varphi(\alpha,\beta) = \begin{cases} 0, & if\ \alpha \leqslant \beta \\ \alpha, & else \end{cases}$ is an indicator function. δ is the tolerance allowed for equality constraints. The minimum of $\bar{f}(x)$ is all the elements $\bar{f}_i(x)(i=1,\ldots,m)$ equal to 0, which means this solution satisfies all constraints in model (P).

We should mention that the initial solutions in evolutionary computation are randomly generated, which may be all infeasible. In this situation, *gbest* is defined as the smallest fitness value in the population. Once a feasible solution is found, *gbest* should be replaced by the fitness value of the feasible solution.

2.2. Comparison of two solutions in proposal strategy

In this part, we introduce the selection strategy to determine situations that offspring can replace parent. It can be summarized as follows. The fitness value and constraints represent the concepts in model (P).

(1) If parent and offspring are both feasible, and offspring has better fitness value, then offspring is better than parent.
(2) If parent is not feasible, and has smaller fitness value than *gbest*, then if offspring has smaller fitness value than *gbest*, choose the solution with smaller constraints violation; if offspring has better fitness value than *gbest*, then parent is better than offspring.
(3) If parent is not feasible, and has bigger fitness value than *gbest*, then if offspring has bigger fitness value than *gbest*, choose the solution with smaller constraints violation; if offspring has smaller fitness value than *gbest*, than offspring is better than parent.

So, the first rule in comparison strategy can assure that feasible solution will not be replaced by infeasible solution. The second and third rules choose better solution using model (R). If parent is infeasible, we first consider their fitness value. If one solution has smaller fitness value than *gbest*, and another one has not, then the former one is better solution. If both solutions are smaller or bigger than *gbest*, choose the dominating solution as the better solution.

The comparison strategy using the concepts of dominance can be summarized as follows:

(1) If $\bar{f}_i(a) \leqslant \bar{f}_i(b)$ for all $i=1,\ldots,m$, solution a has smaller objective function value.