Applied Soft Computing 14 (2014) 363–380

Contents lists available at ScienceDirect

Applied Soft Computing

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

A modified objective function method with feasible-guiding strategy to solve constrained multi-objective optimization problems

Licheng Jiao, Juanjuan Luo*, Ronghua Shang, Fang Liu

Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, Xidian University, Xi'an 710071, PR China

ARTICLE INFO

Article history:
Received 18 May 2012
Received in revised form 9 September 2013
Accepted 7 October 2013
Available online 16 October 2013

Keywords:
Constrained multi-objective optimization
Constraint handling
Modified objective function method
Feasible-guiding strategy

ABSTRACT

For constrained multi-objective optimization problems (CMOPs), how to preserve infeasible individuals and make use of them is a problem to be solved. In this case, a modified objective function method with feasible-guiding strategy on the basis of NSGA-II is proposed to handle CMOPs in this paper. The main idea of proposed algorithm is to modify the objective function values of an individual with its constraint violation values and true objective function values, of which a feasibility ratio fed back from current population is used to keep the balance, and then the feasible-guiding strategy is adopted to make use of preserved infeasible individuals. In this way, non-dominated solutions, obtained from proposed algorithm, show superiority on convergence and diversity of distribution, which can be confirmed by the comparison experiment results with other two CMOEAs on commonly used constrained test problems.

1. Introduction

In real world, we often encounter problems that at least two objectives need to be optimized simultaneously and a set of constraint conditions must be satisfied in the meantime. All these problems are called constrained multi-objective optimization problems (CMOPs). Solving CMOPs is an important part of the optimization field. In contrast to multi-objective optimization problems (MOPs), CMOPs have to deal with various limits on decision variables, the interference resulting from constraints, and the relationship between objective functions and constraints [1].

There are a large amount of constraint handling methods in solving constrained optimization problems. According to [2,3], the commonly used constraint handling methods can be roughly classified into four categories:

(1) Use of penalty functions:

Method based on penalty functions is the simplest and most commonly used constraint handling approach. It combines constraint violations with objective functions through the penalty coefficients that are used to keep the balance between them. For penalty function method, its challenge is how to regulate the penalty coefficients to preserve individuals [1]. It can be classified into different categories according to their values. If the penalty coefficients stay constant during the whole search process, then it is called static penalty function method. If the penalty coefficients change with generation number, then it is dynamic penalty function method. In adaptive penalty function method, a feedback taken from the searching progress is added to control the amount of penalty. In death penalty function method, infeasible individuals are rejected and this method has the drawback of not extracting any information from infeasible individuals [3].

(2) Maintaining a feasible population by special representations and genetic operators:

According to literature [1], the main purpose of this kind of method is to generate feasible individuals, to remove infeasible region from the search space, or to recover infeasible individuals to feasible individuals. In Lawrence Davis' handbook of genetic algorithms [4], several special representations and genetic operators were used to solve complex real world problems. GENOCOP [5], a method developed by Michalewicz, which generates feasible individuals by handling linear constraint with eliminating equalities and designing special genetic operators. The drawback of this method is the need of a feasible starting point and only linear constraints exist. In Ray's paper [6], an infeasibility driven evolutionary algorithm was proposed to handle constraints by maintaining a small percentage of infeasible solutions close to the constraint boundaries. In [7–9], some decoders methods were proposed to solve constrained problems, and also some repair algorithms to handle constraints, such as [10–13]. In [14], a gene silencing operator was proposed to solve constrained optimization problems.

* Corresponding author. Tel.: +34 657569939.
E-mail address: ljjinxd@163.com (J. Luo).

1568-4946/$ – see front matter.
http://dx.doi.org/10.1016/j.asoc.2013.10.008

(3) Separation of objectives and constraints:

There are several approaches that a clear distinction is made to handle constraints and objective functions. In [15], a stochastic ranking approach was proposed to balance objective and penalty functions stochastically. A probability factor determines that the rank of each individual is decided by objective functions or penalty functions. In [16], Deb proposed a criteria that feasible solutions always take advantage of infeasible solutions, the individual with better objective value is preferred when two feasible solutions are compared together, and individual with smaller constraint violations is preferred when two infeasible individuals are compared. In [17,18], an α level comparison and ε level comparison method was adopted to transform an algorithm for unconstrained problems into an algorithm for constrained problems with some tolerance of constraint violation. In [19], a two-population based method was proposed to solve constrained problems. Solutions that satisfy all the constraints and solutions that are potential for the problem are saved in two populations respectively, and similar method was adopted in [20]. Besides that, there are also some algorithms to solve constrained problems in stages, such as [21].

(4) Hybrid method:

The property of this category is to combine two or more above constraint handling methods together to achieve better performance since different technique has its own advantages and fits for only a subset of problems [22], such examples can be seen in [1,23,24].

In conclusion, the major issue of constraint handling method is how to deal with infeasible individuals throughout the whole searching progress. In recent years, a few researchers have focused their research on MOPs, a number of population based stochastic optimization algorithms such as evolutionary algorithms (EAs), particle swarm optimization (PSO), differential evolution (DE) [1,18,24], the human immune system (HIS) based algorithms [25,26] and some other algorithms inspired by nature [27] have been proposed to handle MOPs. Although there have been many approaches to handle constraints, most of them are aimed to deal with single objective optimization problems with constraints, few researchers focus on dealing with constraint handling and MOPs simultaneously.

In [28], a constraint-dominate principle, which feasible solutions always perform better than infeasible solutions and infeasible solutions with lower constraint violation are always better than those with larger constraint violations, was proposed to handle CMOPs. The main drawback of this principle is that it may lose some potential information of the infeasible region.

Different from constraint-dominate principle, an algorithm that explicitly maintains a small percentage of infeasible solutions close to constraint boundary was proposed in [6]. It adopted a user-defined parameter that determines the proportion of infeasible solutions and a constraint violation measure that based on the relative constraint violation ranking among the current population. Then, the constraint violation measure, which is sum of the relative constraint rankings, is added to form the modified objectives that can be made a non-dominated sorting in the following process. In this method, good infeasible solutions can rank higher than some feasible solutions so that some potential infeasible region information may come into use.

In [29], Ray et al. proposed a more elaborate constraint handling technique based on three different non-dominated rankings, which are objective ranking, constraint violation ranking and the combination of objective and violation ranking. Then the algorithm will perform according to the predefined rules.

In [30], CMOPs was converted into MOPs by using the modified objective functions. The final modified objective function formulation includes distance measure and adaptive penalty. Both of them are constituted adaptively by normalized constraint violations and normalized objective functions with a parameter r_f, which is decided by the proportion of feasible solutions in current population. Penalty function can make some infeasible solutions with good objective function values and low constraint violation values be selected.

In this case, a modified objective function method is proposed to handle constraints in this paper. Objective function values and constraint violation values are simply combined together by feasibility ratio to modify objective functions, which enable infeasible individuals with low constraint violation values and better objective function values participate in the searching of optimal solutions. This approach allows the selection to switch between feasibility and optimality during the evolution process. Different from constraint-dominant principle or ranking-based constraint handling method mentioned above, it is based on modifying objective function to preserve a proportion of infeasible individuals. Although proposed handling method was inspired by [30], it adopts a totally different modifying method for feasible and infeasible individuals in different situation, and the details will be shown in Section 3.1. Furthermore, proposed feasible-guiding strategy makes more use of preserved infeasible individuals to determine a feasible direction with the guide of feasible individuals, that can be treated as a DE/infeasible-to-feasible strategy, with which infeasible individuals in the local region can evolve towards feasible direction on some extent and the infeasible individuals can be fully used but not blindly. These two methods facilitate the searching of Pareto-optimal solutions not only go from feasible space but also from infeasible space. With all these methods, the algorithm is capable of finding feasible solutions even though the feasible region is smaller compared to the infeasible regions. Both of proposed methods can be extended easily to make an improvement on other CMOEAs. The performance of comparison experiments with other two CMOEAs on commonly used constrained test problems shows the superiority of proposed methods.

This paper is organized as follows. Section 2 presents a brief description of CMOPs. Then, a detailed description of the proposed constraint handling method is provided in Section 3. Next, experiments on various CMOP test problems are used to evaluate proposed constraint handling method in Section 4. Finally, a conclusion of this paper and future work are given in Section 5.

2. Problem description

A constrained multi-objective optimization problem (CMOP) can be mathematically formulated as follows:

$$\begin{aligned} \text{Minimize} \quad & f_i(x) = f_i(x_1, x_2, \ldots, x_n), && i = 1, 2, \ldots, k \\ \text{Subject to} \quad & g_j(x) = g_j(x_1, x_2, \ldots, x_n) \le 0, && j = 1, 2, \ldots, p \\ & h_j(x) = h_j(x_1, x_2, \ldots, x_n) = 0, && j = p+1, p+2, \ldots, m \\ & x_l^{min} \le x_l \le x_l^{max}, && l = 1, 2, \ldots, n \end{aligned} \tag{1}$$

where $x = (x_1, x_2, \ldots, x_n) \in \Omega$ is a n-dimensional decision variable vector, which is bounded in the search space Ω, x_l^{min} and x_l^{max} defines lower and upper boundaries of each dimension of search space Ω respectively. $f_i(x)$ is the ith objective function, and k is the number of objective functions. There are a total of m constraint functions to be satisfied with, of which p are inequality constraint functions, and the rest are equality constraint functions, which divide the search space into feasible space and infeasible space. $g_j(x)$ is the jth inequality constraint, and $h_j(x)$ is the jth equality constraint. When dealing with CMOPs, individuals that satisfy all