Available online at www.sciencedirect.com

ScienceDirect

Advances in Engineering Software 38 (2007) 717–721

ADVANCES IN ENGINEERING SOFTWARE

www.elsevier.com/locate/advengsoft

Short Communication

Fuzzy multiobjective optimization of truss-structures using genetic algorithm

Omer Kelesoglu

Department of Construction Science, Firat University, 23119 Elazig, Turkey

Received 12 January 2006; received in revised form 2 March 2007; accepted 2 March 2007
Available online 9 May 2007

Abstract

In this paper, a method for solving fuzzy multiobjective optimization of space truss with a genetic algorithm is proposed. This method enables a flexible method for optimal system design by applying fuzzy objectives and fuzzy constraints. The displacement, tensile stress, fuzzy sets, membership functions and minimum size constraints are considered in formulation of the design problem. An algorithm was developed by using MATLAB programming. The algorithm is illustrated on 56-bar space truss system design problem and the results are discussed.

Keywords: Multiobjective; Fuzzy sets; Genetic algorithm; Membership function; Truss system

1. Introduction

Many real world engineering design problems can be formulated as multiobjective optimization problems. Although multiobjective optimization can be used in all areas of engineering, it was originally used mainly for optimization of structures in civil engineering. The first paper on multiobjective optimization was by Pareto [1]; since then determination of a compromise set of multiobjective is called Pareto optimization [2,3].

Zadeh first introduced the concept of fuzzy set theory [4]. Then Zimmermann applied the fuzzy set theory concept with some suitable membership functions to solve a linear programming problem with several objective functions [5]. Rao et al. applied a fuzzy optimization technique with membership functions to solve the multiobjective engineering problem [6]. Today, fuzzy theory and membership function values are widely used in engineering design. Use of membership functions is also a tool to solve multiobjective design problems. For such cases, the application of fuzzy set theory is effective, and numerous multiobjective fuzzy optimization techniques have been developed [7–10].

In this paper, a fuzzy multiobjective optimization approach is presented. The fuzzy λ-formulation is combined with a genetic algorithm for solving fuzzy multiobjective optimization problems with design variables. Objective functions including the volume of the minimum weight and minimum displacement are considered in the numerical example. The volume of construction weight, displacement, geometrical properties, cross sectional areas, member degrees, and upper and lower limit values of the stress elements are used as constraints. Finally, the proposed method is applied to complex design problem of a space truss.

2. Fuzzy multiobjective optimization

2.1. Fuzzy optimization

Fuzzy optimization is the name given to formulation of optimization problems with flexible, approximate or uncertain constraints and objectives by using fuzzy sets. Fuzzy optimization problems associate fuzzy input data by fuzzy

E-mail address: okelesoglu@firat.edu.tr

0965-9978/$ - see front matter
doi:10.1016/j.advengsoft.2007.03.003

membership functions. A fuzzy optimization model assumes that objectives and constraints in an imprecise and uncertain situation can be represented by fuzzy sets. In fuzzy optimization the fuzziness of available resources is characterised by membership functions that span a tolerance range. In the present study, objective functions are considered as fuzzy sets and inflows are considered in the form of chance constraints. The complete formulation is available in Rao's paper [6]. The mathematical λ-formulation of a multiobjective fuzzy optimization can be written as

$$\text{maximize} \quad \lambda \tag{1}$$

$$\text{subject to} \quad \lambda - \mu_{f_i}(x) \leqslant 0, \quad i = 1, 2, \ldots, k, \tag{2}$$

$$\lambda - \mu_{g_j}(x) \leqslant 0, \quad j = 1, 2, \ldots, m, \tag{3}$$

$$x_i^l \leqslant x_i \leqslant x_i^u, \quad i = 1, 2, \ldots, n. \tag{4}$$

where x is the vector of design variables, $\mu_{f_i}(x)(i = 1, 2, \ldots, k)$ the set of objective functions, and $\mu_{g_j}(x) \leqslant 0(j = 1, 2, \ldots, m)$ the set of constraints.

For solving multiobjective programming problems, the following multiobjective λ-formulation [6] is adopted for the fuzzy nonlinear problem. For the multiobjective optimization problem, Pareto optimal sets define the solution in this paper; the Pareto optimal solution concept is defined in

$$f_i(x) \leqslant f_i(x^*) \text{ for all } i, \quad i = 1, \ldots, k, \tag{5}$$

$$f_i(x) < f_i(x^*) \text{ for at least one } i, \quad 1 \leqslant i \leqslant k. \tag{6}$$

Given a set of feasible solutions P for the problem, a solution $x^* \in P$ is said to be a Pareto optimal solution for the problem if and only if there is no other solution $x \in P$, satisfying the conditions in Eqs. (5) and (6) [11].

3. Genetic algorithms approach

Genetic algorithms were more fully developed after the original work by Holland [12]. These approaches consist of optimization procedures based on principles inspired by natural evolution. Given a problem for which a closed-from solution is unknown, or impossible to obtain with classical methods, an initial randomly generated population of possible solutions is created. Its characteristics are then used in an equivalent string of genes or chromosomes that are later recombined with genes from other individuals. It can be shown that by using a Darwinian-inspired natural selection process, the method gradually converges towards the best-possible solution [13].

The GA manipulates a population of the potential solution for problems such as optimization. Genetic algorithms for multiobjective optimization problems have been proposed in the literature [11,14,15]. They can be used to compute the membership functions of fuzzy sets [16,17]. Kiyota et al. have proposed a multiobjective fuzzy optimization method by genetic algorithm [18]. Since then, several papers have evolved that use genetic algorithms fuzzy optimization [19–21].

To solve the problem given in Eq. (1), a GA is used. Since a GA seeks a set of solutions for multiple objectives in a group, it can offer several candidates to the engineer. An outline of the fuzzy multiobjective genetic algorithm for rule selection is as follows.

3.1. Representation and initialization

An initial population of size n is randomly generated from $[0,1]^{k+1}$ according to the uniform distribution in the closed interval $[0,1]$. Let the population be

$$x_i = (\mu_{i_0}, \mu_{i_1}, \ldots, \mu_{ik}), \tag{7}$$

where $i = 1,2,\ldots,n$ and μ_{ij} is a real number in $[0,1]$, $j = 0,1,2,\ldots,k$. Each individual x_i in a population is a chromosome. For each chromosome x_i, $i = 1,2,\ldots,n$, the centroid eval(x_i) is calculated as the fitness value. The chromosomes in the population can be rated in terms of their fitness values. Let the total fitness value of the population be $F = \sum_{i=1}^{n} \text{eval}(x_i)$. The cumulative fitness value for each chromosome, $S_i = \sum_{i=1}^{m} \frac{\text{eval}(x_i)}{F}$, $m = 1, 2, \ldots, n$, is calculated.

3.2. Calculation of the fitness value for each chromosome

Now, this paper calculates the fitness value, eval(x_i), for each chromosome $x_i (i = 1,2,\ldots,n)$ as follows. For each chromosome x_i membership function values for the objectives and constraints are first calculated. If $(\mu_{\min})_i$ is the membership function value corresponding to the ith chromosome, x_i, then

$$(\mu_{\min})_i = \min\{\mu_0[f_0(x_i)], \mu_1[f_1(x_i)], \ldots, \mu_k[f_k(x_i)]\}, \quad k = 1, 2, \ldots, m. \tag{8}$$

The fitness value for μ_i is calculated as

$$\text{eval}(x_i) = (\mu_{\min})_i. \tag{9}$$

3.3. Genetic operators

(1) *Crossover*
It is applied with a certain probability in order to combine genes from two chromosomes and create new ones. A randomly chosen two-point crossover is introduced, where two chromosomes swap their genes starting from two-point, followed by a repair operation to ensure no repetition of genes in new chromosomes. Let p be the probability of a crossover, $0 \leqslant p \leqslant 1$. Usually, p is between 0.6 and 0.9 [22], so, on average, 60% to 90% of the chromosomes undergo a crossover. The two-point crossover is used. An example is as follows: