Advances in Engineering Software 80 (2015) 101–115

Contents lists available at ScienceDirect

Advances in Engineering Software

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

Ant colony approaches for multiobjective structural optimization problems with a cardinality constraint

Jaqueline S. Angelo [a,*], Heder S. Bernardino [b], Helio J.C. Barbosa [a,b]

[a] *Laboratório Nacional de Computação Científica, Petrópolis, RJ, Brazil*
[b] *Universidade Federal de Juiz de Fora, Juiz de Fora, MG, Brazil*

ARTICLE INFO

Article history:
Available online 25 October 2014

Keywords:
Structural optimization
Cardinality constraint
Multiobjective optimization
Ant colony algorithm
Adaptive penalty
Multicriteria tournament decision

ABSTRACT

Two Ant Colony Optimization algorithms are proposed to tackle multiobjective structural optimization problems with an additional constraint. A cardinality constraint is introduced in order to limit the number of distinct values of the design variables appearing in any candidate solution. Such constraint is directly enforced when an ant builds a candidate solution, while the other mechanical constraints are handled by means of an adaptive penalty method (APM). The test-problems are composed by structural optimization problems with discrete design variables, and the objectives are to minimize both the structure's weight and its maximum nodal displacement. The Pareto sets generated in the computational experiments are evaluated by means of performance metrics, and the obtained designs are also compared with solutions available from single-objective studies in the literature.

1. Introduction

Frequently in structural design problems, one is interested in finding the minimal weight of a framed structure subject to stress, displacements, or other constraints. In fact, besides these constraints, the problem of optimizing a framed structure can be formulated considering multiple and conflicting objective functions, for example, to minimize the weight of the structure and its nodal displacements.

In practice, the design variables are often to be chosen from commercially available sizes and/or types leading to a discrete optimization problem which is usually harder than its continuous counterpart. In fact, techniques from mathematical programming which are used in the continuous case must be adapted to deal with the discrete variables. Any attempt to "round" or substitute those obtained values by the "closest" available commercial sizes can potentially make the design infeasible (violating some constraint) or with unnecessarily degraded performance. Thus, the use of nature inspired metaheuristics, such as Ant Colony Optimization (ACO), becomes attractive.

Besides, it is a common practice in structural optimization to group certain sizing or shape variables into a single design variable. This procedure is used when symmetry conditions are to be imposed in the final design, and also to reduce the total number of design variables leading to a "smaller" search space. However, the effectiveness of this procedure depends on the designer's skill, and previous experience is valuable at this point in order to allocate members/variables to a group. As a result, it would be useful to the designer to leave to the optimizer algorithm the task of deciding how to group members and/or design variables. In addition, if the designer were able to limit the number of the different design variables (such as cross-sectional areas) he/she could achieve economies of bulk purchasing, and simplify the construction of the structure [1]. Those points can be achieved by introducing a cardinality constraint, in which the designer is able to enforce the maximum number of distinct design variables appearing in any candidate solution.

In [2], an ant colony approach was proposed to solve discrete and multiobjective structural optimization problems, where two ant colony algorithms equipped with an adaptive penalty method were compared. The algorithms proposed presented good performance according to the qualities expected for non-dominated solution sets.

In order to extend the research presented in [2], here, a cardinality constraint will be considered to solve the practically relevant multiobjective structural optimization problem. The two ant colony algorithms, MOAS and MOACS, will be modified to enforce the cardinality constraint. In this new implementation, the user is able to prescribe the maximum number m of different sizes or types he/she is willing to use in a particular design.

Furthermore, in order to incorporate the decision making process, we illustrate the use of a multicriteria decision method to

* Corresponding author.

http://dx.doi.org/10.1016/j.advengsoft.2014.09.015
0965-9978/

assist the decision maker in choosing the preferred Pareto-optimal solution. The Multicriteria Tournament Method (MTD) [3] is used in an *a posteriori* preference articulation approach to select the final solution of the multiobjective optimization problem.

This paper is organized as follows. Section 2 describes the multiobjective optimization problem and Section 3 the multiobjective structural optimization problem. Section 4 presents how the cardinality constraints are introduced in the structural optimization problem. Section 5 presents the ant colony metaheuristic. Section 6 describes the proposed ant colony algorithms and Section 6.1 details the adaptations in the process of solution construction so as to enforce the cardinality constraints. In Section 7a multicriteria decision method is presented so as to illustrate the use of a decision technique to assist the decision maker. The computational experiments are described in Section 8 and the paper ends with a Concluding Remarks section.

2. Multiobjective optimization

In multiobjective optimization problem (MOO) several objectives have to be simultaneously optimized. Without loss of generality, the MOOs considered here can be formulated as

$$\begin{array}{ll} \text{minimize} & f_i(\mathbf{x}) \quad i = 1, \ldots, k; \\ \text{subject to} & g_p(\mathbf{x}) \leqslant 0, \quad p = 1, \ldots, m; \\ & x_j^{(L)} \leqslant x_j \leqslant x_j^{(U)}, \quad j = 1, \ldots, n. \end{array} \tag{1}$$

where we have $k(\geqslant 2)$ *objective functions* f_i to be minimized, with $\mathbf{f}(\mathbf{x})$ denoting the vector of objective functions. A solution $\mathbf{x}$ is the vector of *n decision variables*: $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. The *feasible region* is defined by $\mathcal{S} := \{\mathbf{x} : \mathbf{x}^{(L)} \leqslant \mathbf{x} \leqslant \mathbf{x}^{(U)}, g_p(\mathbf{x}) \leqslant 0\}$, where $\mathbf{x}$ is called a feasible candidate solution whenever $\mathbf{x} \in \mathcal{S}$. The image of the feasible region is called *feasible objective region*, denoted by $\mathcal{Z}(= \mathbf{f}(\mathcal{S}))$. Its elements are the points $\mathbf{z} = (z_1, z_2, \ldots, z_k)^T$ in the objective space, with $z_i = f_i(\mathbf{x})$, for all $\mathbf{x} \in \mathcal{S}$.

Usually, the objective functions are conflicting and possibly non-commensurable; hence there is no single solution that is optimal with respect to all objectives. In this case there is a set of alternatives that are superior to the remainder when all the objectives are considered. This set, composed by the so-called non-dominated solutions, provides many options to the decision-maker.

Assuming a problem in which all objectives should be minimized, a solution $\mathbf{x} \in \mathcal{S}$ dominates another solution $\mathbf{x}' \in \mathcal{S}$ ($\mathbf{x}$ is *non-dominated* by $\mathbf{x}'$) when

$$f_i(\mathbf{x}) \leqslant f_i(\mathbf{x}') \quad \forall i \in \{1, \ldots, k\} \quad \text{and} \quad \exists j \in \{1, \ldots, k\} : f_j(\mathbf{x}) < f_j(\mathbf{x}'), \tag{2}$$

i.e., the solution $\mathbf{x}$ is no worse than $\mathbf{x}'$ in all objectives and better in at least one of them. Notice that all possible pairwise solutions must be compared in order to find those that are non-dominated. Finally, the *Pareto-optimal set* is the set of non-dominated solutions, and the corresponding image in the objective space defines the *Pareto front*, which contains the best collection of solutions found for the problem.

3. Multiobjective structural optimization

We are interested in finding a set of discrete cross-sectional areas $\mathbf{A} = \{A_1, A_2, \ldots, A_N\}$ which minimizes both the total weight $w(\mathbf{A})$ of a given truss (pin-jointed) structure and the maximum displacement $d(\mathbf{A})$ of its nodes, subject to stress constraints. Notice that the first objective (total weight) corresponds to the material cost of the structure while the second one (maximum nodal displacement) is associated with mechanical performance. It is not hard to see that those objectives are conflicting and non-commensurable. Formally, the problem can be expressed as minimizing both functions

$$w(\mathbf{A}) = \sum_{k=1}^{N} \gamma A_k L_k \tag{3}$$

$$d(\mathbf{A}) = \max(|u_{i,l}|), \tag{4}$$

subject to the (normalized) stress constraints

$$\frac{|s_{j,l}|}{s_{adm}} - 1 \leqslant 0. \tag{5}$$

In (3), $A_i \in T$ is the cross-sectional area of the ith bar, T is a table of commercially available cross sectional areas, N is the number of bars in the truss structure, L_k is the length of the kth bar, and γ is the specific weight of the material. In (4), $u_{i,l}$ is the nodal displacement of the ith translational degree of freedom with $1 \leqslant i \leqslant M$, and $1 \leqslant l \leqslant N_L$, where N_L is the number of load cases applied to the structure. In (5), s_j is the stress at the jth bar, and s_{adm} is the allowable stress for the material.

Although the function $w(\mathbf{A})$ from Eq. (3) is simple, the function $d(\mathbf{A})$ (Eq. (4)) and the constraints are complex implicit functions of the design variables $\mathbf{A}$, and they require the solution of the equilibrium equations of the discrete finite element model, which can be written in the linear case as

$$\mathbf{K}(\mathbf{A})\mathbf{u}_l = \mathbf{f}_l, \quad 1 \leqslant l \leqslant N_L, \tag{6}$$

where $\mathbf{K}$ is the symmetric and positive-definite stiffness matrix of the structure, and $\mathbf{u}_l$ and $\mathbf{f}_l$ are the vector of nodal displacements and the vector of nodal forces for the lth load condition, respectively. For each load condition, the system (Eq. (6)) should be solved for the displacement field and the stress in the jth bar is then calculated according to Hooke's Law as

$$s_{j,l} = E\delta(\mathbf{u}_l), \tag{7}$$

where E is the Young's modulus of the material and δ is the unit change in length of the bar.

4. A cardinality constraint in structural optimization

It is clear that the optimal solution will usually employ as many sizes as the number of design variables defined for the problem. Variable linking, a common procedure which groups different design parameters of the problem in a single design variable, allows for a reduction in the total number of design variables and frequently also in the complexity of the search problem. This procedure is also useful, for instance, when symmetry conditions are to be enforced. Notice that each choice of design variable linking leads to a particular optimization problem with, consequently, a different optimal solution. Although it may be simple to define the groups in order to enforce the symmetry conditions of the structure, in general the choice (made *a priori* by the analyst) of which variables should be grouped together is not trivial. As a result, the final set of independent design variables may be far from optimal and the corresponding optimal solution more expensive than necessary, due to inadequate design variable linking.

The solution adopted here provides the designer with the possibility of enforcing the maximum number of different sizes to be used in a given problem, and transfers to the search technique the task of finding a good grouping for the design variables. This constraint, called a cardinality constraint, originally introduced by means of a genetic algorithm encoding in [4] and latter applied to structural optimization problems in [1,5] would allow for: