Computers and Structures 130 (2014) 10–21

Contents lists available at ScienceDirect

Computers and Structures

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

Democratic PSO for truss layout and size optimization with frequency constraints

A. Kaveh *, A. Zolghadr

Centre of Excellence for Fundamental Studies in Structural Engineering, School of Civil Engineering, Iran University of Science and Technology, Tehran 16, Iran

ARTICLE INFO

Article history:
Received 5 February 2013
Accepted 16 September 2013
Available online 12 October 2013

Keywords:
Democratic Particle Swarm Optimization (DPSO)
Frequency constraints
Optimal design
Truss structures

ABSTRACT

This paper represents a new algorithm for structural optimization with frequency constraints. The new algorithm is termed Democratic Particle Swarm Optimization. The emphasis is placed upon alleviating the premature convergence phenomenon which is believed to be one of flaws of the original PSO. When considering frequency constraints in a structural optimization problem, the search spaces happen to be highly non-linear and non-convex hyper-surfaces with numerous local optima and naturally the problem of premature convergence is amplified. The proposed algorithm is capable of coping with this problem. Four numerical examples are presented to demonstrate the viability of the algorithm.

1. Introduction

Natural frequencies of a structure provide useful information about the dynamic behavior of the system. In fact, in most of the low frequency vibration problems, the response of the structure is primarily a function of its fundamental frequencies and mode shapes [1]. In particular, it is sometimes desirable to control the natural frequencies of a structure in order to keep out the unwelcome resonance phenomenon.

Despite being introduced by Bellagamba and Yang in 1980s [2] structural optimization with frequency constraints has not been yet completely addressed. Different researchers have conducted research on the field with a variety of methods. Grandhi and Venkayya [3] investigated the problem using an algorithm where the resizing and scaling procedure to locate the boundary constraints was carried out using an optimality criterion based on uniform Lagrangian density. Sedaghati et al. [4] performed the frequency analysis using an integrated finite element force method. A mathematical programming technique was used to optimize both frame and truss structures with frequency constraints. Wang et al. [5] used the differentiation of the Lagrangian function to form an optimality criterion. Layout and size optimization of three-dimensional truss structures was considered simultaneously. An infeasible starting point with the minimum weight increment was utilized. Lingyun et al. [6] studied the problem of mass minimization of trusses using a hybridization of the simplex search method and genetic algorithms called niche genetic hybrid algorithm (NGHA). Lin et al. [7] considered the problem of minimum weight design of structures under static and dynamic constraints proposing a bi-factor algorithm based on the Kuhn–Tucker criteria. Gomes [8] investigated simultaneous layout and size optimization of truss structures utilizing the standard Particle Swarm Optimization algorithm. Kaveh and Zolghadr [9] employed the Charged System Search (CSS) algorithm and its enhanced form and a Hybridized CSS–BBBC with trap recognition capability for weight optimization of trusses on layout and size [10].

Weight minimization of a structure with frequency constraints, especially when the frequencies are lower bounded, is believed to be a demanding problem [8]. Frequency constraints are highly nonlinear, non-convex and implicit with respect to the design variables [1] and thus, the problem includes several local optima.

According to Sergeyev and Mroz [11] natural frequencies of a structure are much more sensitive to layout alterations than to size modifications. This might be because of the fact that the vibration modes may switch due to layout modifications that can lead to significant changes in natural frequencies. Moreover, when considering layout and size optimization simultaneously the orders of the variables involved can be different. The two above mentioned facts might lead to divergence. Therefore, the optimization algorithm to be used in these problems should be capable of maintaining proper balance between the diversification and the intensification inclinations. Diversification is the exploration of the search space while intensification is the exploitation of the best solutions found [12].

Particle Swarm Optimization (PSO) initially developed by Kennedy and Eberhart [13] is one of the most widely used multi-agent

* Corresponding author. Tel.: +98 21 77240104; fax: +98 21 77240398.
E-mail address: alikaveh@iust.ac.ir (A. Kaveh).

0045-7949/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.compstruc.2013.09.002

meta-heuristic algorithms. High capability of finding suboptimal solutions in a reasonable amount of time together with relative ease of implementation and little number of parameters has continually encouraged researchers in using PSO for a diverse range of optimization problems in different disciplines. In structural engineering, PSO has been successfully applied to different types of optimization problems ([14–20] among others). However, despite having the above-mentioned benefits, the standard PSO is infamous of premature convergence [21,22]. Improving the exploration ability of the PSO has been an active research topic in recent years [23].

In this paper a Democratic Particle Swarm Optimization (DPSO) is proposed in order to improve the exploration capabilities of the PSO and thus to address the problem of premature convergence. As the name suggests, in the Democratic PSO all eligible particles have the right to be involved in decision making. The details of the proposed modifications will be represented in the upcoming sections.

The remainder of this paper is organized as follows: In Section 2, truss layout and size optimization problem with frequency constraints is stated. The optimization algorithm is proposed after a brief introduction to the standard PSO in Section 3. Four numerical examples are studied in Section 4 in order to show the capability of the proposed algorithm. Finally, in Section 5 some concluding remarks are provided.

2. Problem statement

In a frequency constraint truss layout and size optimization problem the aim is to minimize the weight of the structure while satisfying some constraints on natural frequencies. The design variables are considered to be the cross-sectional areas of the members and/or the coordinates of some nodes. The topology of the structure is not supposed to be changed and thus the connectivity information is predefined and kept unaffected during the optimization process. Each of the design variables should be chosen within a permissible range. The optimization problem can be stated mathematically as follows:

$$\begin{aligned}
&\text{Find} \quad X = [x_1, x_2, x_3, \ldots, x_n] \\
&\text{to minimizes} \quad P(X) = f(X) \times f_{penalty}(X) \\
&\text{Subjected to} \\
&\omega_j \le \omega_j^* \quad \text{for some natural frequencies } j \\
&\omega_k \ge \omega_k^* \quad \text{for some natural frequencies } k \\
&x_{imin} \le x_i \le x_{imax}
\end{aligned} \tag{1}$$

where X is the vector of the design variables, including both nodal coordinates and cross-sectional areas. Here n is the number of variables which is naturally affected by the element grouping scheme which in turn is chosen with respect to the symmetry and practice requirements. $P(X)$ is the penalized cost function or the objective function to be minimized; $f(X)$ is the cost function, which is taken as the weight of the structure in a weight optimization problem; $f_{penalty}(X)$ is the penalty function which is used to make the problem unconstrained. The non-zero values of the penalty function result from the violations of the constraints corresponding to the response of the structure [5]; ω_j is the jth natural frequency of the structure and ω_j^* is its upper bound. ω_k is the kth natural frequency of the structure and ω_k^* is its lower bound. x_{imin} and x_{imax} are the lower and upper bounds of the design variable x_i, respectively.

The cost function is expressed as:

$$f(X) = \sum_{i=1}^{nm} \rho_i L_i A_i \tag{2}$$

where ρ_i is the material density of member i; L_i is the length of member i; and A_i is the cross-sectional area of member i.

The penalty function is defined as [26]:

$$f_{penalty}(X) = (1 + \varepsilon_1 \cdot v)^{\varepsilon_2}, \quad v = \sum_{i=1}^{q} v_i \tag{3}$$

where q is the number of frequency constraints.

$$v_i = \begin{cases} 0 & \text{if} \quad \text{the } i\text{th constraint is satisfied} \\ \left|1 - \frac{\omega_i}{\omega_i^*}\right| & \text{else} \end{cases} \tag{4}$$

The parameters ε_1 and ε_2 are selected considering the exploration and the exploitation rate of the search space. In this study ε_1 is taken as unity, and ε_2 starts from 1.5 linearly increases to 6 in all test examples. These values penalize the unfeasible solutions more severely as the optimization process proceeds. As a result, in the early stages, the agents are free to explore the search space, but at the end they tend to choose solutions without violation.

3. Optimization algorithm

In this section the proposed algorithm is introduced as an improved version of Particle Swarm Optimization algorithm. Although PSO is a very well-known and commonly used optimization algorithm, in order to make the improvements visible, a basic form of the algorithm which is referred to here as the standard PSO will be briefly summarized first. Since PSO has been gradually improved by different researchers and in order to make the comparison more meaningful, the description of the standard PSO is taken from Ref. [8] in which it is used for the same type of optimization problems.

3.1. Standard PSO

Particle Swarm Optimization, first introduced by Kennedy and Eberhart [13], is a population-base meta-heuristic algorithm inspired by the social behavior of animals such as fishes schooling, insects swarming, and birds flocking. Like any other population-base meta-heuristic algorithm, PSO starts with a set of agents which are randomly spread in the multi-dimensional search space of problem. These agents are viewed as potential solutions of the optimization problem at hand. The quality of each candidate solution is measured using an objective function. As the optimization process continues these agents move around in the search space searching for better positions. By gradual improvement of the locations of the particles in a swarm the algorithm finally converges to a sub-optimal solution.

In their attempt to find better positions, agents make use of two different sources of information: their own best experience which is called a local best position and the swarm's best position so far which is called the global best position. Based on these two pieces of information an agent decides about the next position it is going to experience in iteration $(k+1)$ by forming a velocity vector as follows:

$$v_{i,j}^{k+1} = \chi\left[\omega v_{i,j}^k + c_1 r_1 \left(xlbest_{i,j}^k - x_{i,j}^k\right) + c_2 r_2 \left(xgbest_j^k - x_{i,j}^k\right)\right] \tag{5}$$

where, $v_{i,j}^k$ is the velocity or the amount of change of the design variable j of particle i, $x_{i,j}^k$ is the current value of the jth design variable of the ith particle, $xlbest_{i,j}^k$ is the best value of the design variable j ever found by ith particle, $xgbest_j^k$ the best value of the design variable j experienced by the entire swarm so far, r_1 and r_2 are two random numbers uniformly distributed in the range $(1,0)$, c_1 and c_2 are two parameters representing the particle's confidence in itself and in the swarm, respectively. In this paper, these parameters which determine the particle's inclination to move toward local and global best experiences are taken to be equal to 2 as reported to be suitable in the literature [24], however these had been taken as 1.5 in Ref.