Applied Soft Computing xxx (2015) xxx–xxx

Contents lists available at ScienceDirect

Applied Soft Computing

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

Sizing optimization of truss structures using flower pollination algorithm

Gebrail Bekdaş [a], Sinan Melih Nigdeli [a,*], Xin-She Yang [b]

[a] Department of Civil Engineering, Istanbul University, 34320 Avcılar, Istanbul, Turkey
[b] Design Engineering and Mathematics, Middlesex University London, The Burroughs, London, UK

ARTICLE INFO

Article history:
Received 6 May 2015
Received in revised form 19 August 2015
Accepted 20 August 2015
Available online xxx

Keywords:
Flower pollination algorithm
Truss structures
Sizing optimization
Iterative constraint handling strategy

ABSTRACT

The recently developed flower pollination algorithm is used to minimize the weight of truss structures, including sizing design variables. The new algorithm can efficiently combine local and global searches, inspired by cross-pollination and self-pollination of flowering plants, respectively. Furthermore, it implements an iterative constraint handling strategy where trial designs are accepted or rejected based on the allowed amount of constraint violation that is progressively reduced as the search process approaches the optimum. This strategy aims to obtain always feasible optimized designs. The new algorithm is tested using three classical sizing optimization problems of 2D and 3D truss structures. Optimization results show that the proposed method is competitive with other state-of-the-art metaheuristic algorithms presented in the literature.

1. Introduction

The main objectives in structural design are to ensure the safety of structures and find a design with the maximum gain. Generally speaking, safety measures are defined as design constraints, while objective functions depending on design variables are defined as the maximum gain. In recent years, nature-inspired metaheuristic algorithms have been commonly used in engineering optimization. These iterative algorithms are very effective to find precise optimum values of challenging engineering problems with multiple variables and constraints. In addition, metaheuristic algorithms allow to account for design limitations by combining optimization process with accurate engineering analysis. Metaheuristic algorithms can be grouped as either trajectory-based algorithms or population-based algorithms. Simulated Annealing (SA) method developed by Kirkpatrick et al. [1] is a trajectory-based algorithm, while Harmony Search (HS) [2], Genetic Algorithm (GA) [3], Cuckoo Search [4], Particle Swarm Optimization (PSO) [5], Ant Colony Optimization (ACO) [6] are all population-based algorithms.

In addition, new metaheuristic algorithms are also being developed in order to improve the optimization capability and convergence behavior. For example, the Flower Pollination Algorithm (FPA) is a population-based metaheuristic method recently developed by Yang [7], which imitates the nature of flower pollination. In the development of the algorithm, the main characteristics of flower pollination were idealized into four rules. This study will apply FPA to solve sizing optimization problems of 3D and 2D truss structures.

Truss structures have been optimized using several approaches. For example, Adeli and Kamal optimized space trusses with a dual simplex algorithm to find a local optimum of the approximate problem, while the original problem was iteratively solved [8]. Rajeev and Krishnamoorthy used discrete variables and GA with a penalty parameter depending on constraint violation [9]. Cao also employed GA for the optimum design of frame structures [10]. Erbatur et al. employed GA for the optimum design of planar and space truss structures with continuous and discrete variables [11]. Schutte and Groenwold used PSO for the sizing and layout optimization of truss structures [12]. Camp and Bichon employed ACO to minimize the total weight of the structure subject to stress and deflection constraints [13]. Lee and Geem optimally designed trusses under multiple loading conditions by using HS algorithm and continuous design variables [14]. Big bang–big crunch (BB–BC) algorithm developed by Erol and Eksin [15] was employed in the optimum design methodology of space trusses by Camp [16]. Li et al. developed a heuristic particle swarm optimizer based on the particle swarm optimizer with passive congregation and a HS scheme; this method was successfully applied to the optimum design of planar and spatial truss structures [17].

In addition, Perez and Behdinan optimized truss structures with PSO [18]. Togan and Daloglu improved GA with an initial population strategy and self-adaptive member grouping [19]. Lamberti

* Corresponding author. Tel.: +90 2124737070x17949; fax: +90 2128891887.
E-mail address: melihnig@istanbul.edu.tr (S.M. Nigdeli).

http://dx.doi.org/10.1016/j.asoc.2015.08.037
1568-4946/

presented an efficient SA algorithm for sizing and layout optimization of truss structures [20]. A hybrid BB–BC/PSO algorithm, including Sub-Optimization Mechanism (SOM) was used for sizing optimization of space trusses [21]. Kaveh and Talatahari proposed a hybrid optimization method combining PSO, ACO and HS algorithm for truss structures with discrete and continuous variables [22,23]. Sonmez proposed an optimization methodology including the Artificial Bee Colony and Adaptive Penalty function approach (ABC–AP) in order to minimize the weight of truss structures [24]. Degertekin applied two different improved HS algorithms called the efficient HS algorithm and self-adaptive HS algorithm in order to optimize the size of truss structures [25]. Teaching Learning Based Optimization (TLBO) was applied to truss sizing optimization problems by Degertekin and Hayalioglu [26]. Also, Camp and Farshchin employed a modified TLBO for optimum design of truss [27]. Kaveh et al. developed hybrid particle swallow swarm optimization and the developed algorithm was tested with truss weight minimization problems [28].

Furthermore, chaotic swarming of particles which is the combination of swarm intelligence and chaos theory was developed for optimization of truss structures [29]. Dede and Ayvaz developed a methodology for sizing and layout of trusses using TLBO [30]. Kaveh et al. also used an improved magnetic charged system search in order to solve truss optimization problems with continuous and discrete variables [31]. Colliding bodies optimization (CBO), developed by Kaveh and Mahdavi [32], reproduces the mechanisms of the collisions of moving bodies [33]: the algorithm has successfully been utilized in truss optimization. Kaveh and Ilchi Ghazaan later developed an enhanced colliding bodies optimization algorithm (ECBO) which stores some best solutions into the memory in order to optimize truss structures with continuous and discrete variables [34]. Another efficient algorithm for sizing and layout optimization of truss structures is ray optimization (RO) [35].

In the present study, the newly developed metaheuristic Flower Pollination Algorithm (FPA) is applied to structural optimization problems of planar and space trusses. In order to reach the global optimum, an iterative or adaptive constraint handling strategy is included in the search engine. The efficiency of the proposed approach is demonstrated by solving three classical weight minimization problems including sizing variables. Optimization results indicate that FPA is very competitive with other metaheuristic algorithms and can always find efficient designs within the predefined constraint tolerance.

2. Optimum design of truss structures

Trusses are structural systems, consisting of N bars joined by nodes. The system is subject to the external forces applied at the joints. The aim of structural optimization of truss systems is to minimize the total weight of the structure.

In the proposed methodology, the optimization process is encoded together with the structural analysis of the truss. The latter is carried out by using the stiffness method, and nodal displacements are calculated according to

$$\Delta = K^{-1}P. \quad (1)$$

In Eq. (1), Δ, K and P are the nodal displacement vector, system stiffness matrix and external load vector, respectively. The system stiffness matrix is constructed by merging the element stiffness matrices in global coordinates and erasing row and columns which correspond to zero displacements according to the boundary conditions. The stiffness matrix of a bar element with three degrees of freedom at each node is given by

$$K_i = EA_i \begin{bmatrix} l^2 & lm & nl & -l^2 & -lm & -nl \\ lm & m^2 & mn & -lm & -m^2 & -mn \\ nl & mn & n^2 & -nl & -mn & -n^2 \\ -l^2 & -lm & -nl & l^2 & lm & nl \\ -lm & -m^2 & -mn & lm & m^2 & mn \\ -nl & -mn & n^2 & nl & mn & n^2 \end{bmatrix} \quad (2)$$

where

$$l = \frac{L_{xi}}{L_i}, \quad m = \frac{L_{yi}}{L_i} \quad \text{and} \quad n = \frac{L_{zi}}{L_i}. \quad (3)$$

In Eqs. (2) and (3), the total length of the bars (L_i) and the dimensions of the length in x, y and z coordinates (L_{xi}, L_{yi} and L_{zi}) are calculated using the coordinates of the nodes and bounds of the elements which are determined as the design constants. Also, the elasticity modulus (E) and density (γ) of the material of bars are defined as design constants. The areas of bars (A_i) (from i = 1 to N) are the design variables (X) of the optimization problem. The aim of the optimization is to minimize the total structural weight. That is

$$\min \ W = \sum_{i=1}^{N} \gamma L_i A_i, \ (A_i \in \mathbb{R}) \quad (4)$$

for the design variables:

$$X^T = \{A_1, A_2, \ldots, A_N\} \quad (5)$$

within the ranges of Q4

$$A^L \leq A_i \leq A^U \quad i = 1, N \quad (5')$$

subject to the stress ($g_1(X) \leq 0$) and displacement ($g_2(X) \leq 0$) constraints

$$g_1(X): \sigma^L \leq \sigma_i \leq \sigma^U \quad i = 1, N$$
$$\text{and} \quad g_2(X): \delta^L \leq \delta_j \leq \delta^U \quad j = 1, N_j. \quad (6)$$

A^L and A^U are the lower and upper bounds of the solution ranges of the design variables. δ^L and δ^U are the displacement limits which are generally equal in absolute values, but with the opposite signs. The displacement of nodes (δ_j) from j = 1 to N_j (for a system with j nodes) are the components of the displacement vector:

$$\Delta = \begin{bmatrix} \delta_1 \\ \delta_2 \\ . \\ . \\ . \\ \delta_{N_j-1} \\ \delta_{N_j} \end{bmatrix} \quad (\delta_{1,N_j} \in \mathbb{R}). \quad (7)$$

σ^L and σ^U are two different types of stress limits which are for compression (σ^L in − sign) and tension (σ^U in + sign). The stresses of a bar (σ_i^G) in global coordinate is calculated by

$$\sigma_i^G = \frac{K_i \Delta_i}{A_i}, \quad i = 1, N \quad (8)$$

where Δ_i is the vector of the nodal displacements of ith bar. Axial stress on a bar (σ_i) is calculated by multiplying global stresses by directional cosines.

Please cite this article in press as: G. Bekdaş, et al., Sizing optimization of truss structures using flower pollination algorithm, Appl. Soft Comput. J. (2015), http://dx.doi.org/10.1016/j.asoc.2015.08.037