

WEIGHT MINIMIZATION OF SPATIAL TRUSSES WITH GENETIC ALGORITHM

doi: 10.2478/cqpi-2019-0032

Date of submission of the article to the Editor: 25/03/2019

Date of acceptance of the article by the Editor: 04/05/2019

Maksym Grzywiński¹ – *orcid id: 0000-0003-4345-3897*

Jacek Selejdak¹ – *orcid id: 0000-0001-9854-5962*

¹Czestochowa University of Technology, **Poland**

Abstract: A genetic algorithm is proposed to solve the weight minimization problem of spatial truss structures considering size and shape design variables. A very recently developed metaheuristic method called JAYA algorithm (JA) is implemented in this study for optimization of truss structures. The main feature of JA is that it does not require setting algorithm specific parameters. The algorithm has a very simple formulation where the basic idea is to approach the best solution and escape from the worst solution. Analyses of structures are performed by a finite element code in MATLAB. The effectiveness of JA algorithm is demonstrated through benchmark spatial truss 39-bar, and compare with results in references.

Keywords: weight minimization, space truss, size and shape optimization, genetic algorithm

1. INTRODUCTION

Metaheuristic techniques have been developed to solve the structural optimization problems. Genetic algorithms (GA), particle swarm optimization (PSO), harmony search (HS), teaching-learning-based optimization (TLBO), and firefly algorithm (FA) are a variety of well-established methods for the optimal design of structures. Depending on the optimization purpose, cross-sectional areas of the members and/or nodal coordinates separately or simultaneously can be included as the design variables of the problem.

An interesting metaheuristic algorithm that has a very simple formulation and does not require internal parameters is the JAYA algorithm (JA) developed by (Rao, 2016). The JA algorithm was successfully tested on several benchmark functions. (Rao and Waghmare, 2017) later utilized the JA for solving constrained mechanical design problems such as robot gripper, multiple disc clutch brake, hydrostatic thrust bearing and rolling element bearing. The efficiency of the JA with respect to other metaheuristic algorithms was demonstrated also for these test problems.

The JA was used also for optimum design of steel grillage (Dede 2018), for truss optimization including six classical weight minimization problem for sizing, layout and

large-scale optimization problem (Degertekin et al. 2018), and for the optimum mass of braced dome structures with natural frequency constraints (Grzywiński et al., 2019).

2. JAYA ALGORITHM

The JAYA algorithm is a new metaheuristic optimization method proposed by (Rao, 2016). The word “Jaya” means “victory” in Sanskrit. This population based algorithm is based on the concept that the search process should always move toward the best design and avoid the worst design. The search engine continuously tries to get closer to success (i.e. to reach the best design) trying at the same time to avoid failure (i.e. by moving away from the worst design). A definite strength point of JA with respect to other metaheuristic optimization algorithms is that JA does not include any algorithm-specific parameters.

In fact, JA only requires two standard control parameters such as population size (i.e. number of truss designs in the population) and maximum number of iterations. In the optimization process, ndv is the number of design variables (i.e. number of member groups ng in sizing optimization problems; summation of number of member groups and number of shape variables in sizing-shape optimization problems) and np is the population size (i.e. number of truss designs).

The design corresponding to the lowest penalized objective function F_p^{best} is stored as the best design while the design corresponding to the highest penalized objective function F_p^{worst} is the worst design stored in the population.

Let $X_{k,l,it}$ denote the value of the k -th design variable (cross-sectional areas \mathbf{A} and nodal coordinates \mathbf{X}) for the l -th design of the population at the beginning of the it -th iteration. The JA algorithm perturbs this design variable using the following equation:

$$X_{k,l,it}^{new} = X_{k,l,it} + r_{1,k,it}(X_{k,best,it} - |X_{k,l,it}|) - r_{2,k,it}(X_{k,worst,it} - |X_{k,l,it}|) \quad (1)$$

where $X_{k,l,it}^{new}$ is the new value assigned to the design variable $X_{k,l,it}$, $r_{1,k,it}$ and $r_{2,k,it}$ are two randomly generated real numbers in the $[0,1]$ range for the k -th design variable in the it -th iteration. $X_{k,best,it}$ is the value of the k -th design variable for the best design of the population at the it -th iteration while $X_{k,worst,it}$ is the value of the k -th design variable for the worst design stored in the population.

The term $r_{1,k,it}(X_{k,best,it} - |X_{k,l,it}|)$ indicates the tendency of the solution to move closer to the best solution. The term $r_{2,k,it}(X_{k,worst,it} - |X_{k,l,it}|)$ indicates the tendency of the solution to avoid the worst solution.

3. OPTIMIZATION OF TRUSS STRUCTURES

One of the most important factors in the structural design is the total structural weight (Grzywiński and Selejda 2019). There are two types of design variables in this case: cross-sectional areas of members and nodal coordinates. For this aim, the objective function for the truss structures is formulated as:

$$\text{minimize} \quad W(X) = \sum_{k=1}^{ng} A_k \sum_{i=1}^{mk} \rho_i \cdot L_i(x_i) \quad (2)$$

where $W(X)$ is the weight of the truss; A_k (bar cross-sectional areas) and x_i (nodal coordinates) are the design variable, respectively; ρ_i and L_i is the material density and

the length of the i -th element, respectively; ng is total number of group, mk is total number of members in group k , nn is total number of nodes.

The constraints imposed on the structure are:

(a) member stress

$$\sigma_k^c \leq \sigma_k \leq \sigma_k^t \quad k = 1, 2, \dots, ng \quad (3)$$

where σ_k represents the stress for the k -th group elements. σ_k^t is the allowable tensile stress, and σ_k^c the allowable compressive stress, respectively.

(b) Euler buckling stress

$$\sigma_k \leq \sigma_k^b \quad k = 1, 2, \dots, ng \quad (4)$$

where σ_k^b is the Euler buckling compressive stress limit for the k -th group elements; it is usually taken as:

$$\sigma_k^b = \frac{K \cdot E \cdot A_k}{L_k^2} \quad k = 1, 2, \dots, ng \quad (5)$$

where K is a constant determined from the cross-sectional geometry (this case $K = 4$), and E is the Young's modulus of the material,

(c) nodal displacement

$$d_i \leq d_{max} \quad i = 1, 2, \dots, nn \quad (6)$$

in practical design, a specified nodal displacement is subject to a prescribed limit to achieve an appropriate structural rigidity,

(d) limit of design variables

$$A_{min} \leq A_k \leq A_{max} \quad k = 1, 2, \dots, ng \quad (7)$$

$$x_{min} \leq x_i \leq x_{max} \quad i = 1, 2, \dots, nn \quad (8)$$

Inequality (7) and (8) indicates that the design variables including either a shape and/or sizing variable must take a value between minimum and maximum bounds,

$$c_i = \frac{d_i}{d_{max}} \quad c_k = \frac{\sigma_k}{\sigma_{max}} \quad (9)$$

where c_i and c_k are the value of each constraints.

The objective function must be changed as independent of constraints. For this aim, a penalty function calculating value of violation of constraints is determined. By means of this function, the objective function is changed to a function including constraints.

Penalty function is given as:

$$C = \sum_{i=1}^{nn} c_i + \sum_{k=1}^{ng} c_k \quad (10)$$

Objective function is changed to penalized objective function by adding penalty function to it. The penalized objective function, $F(X)$, can be given as:

$$F(X) = W(X)[1 + P \cdot C] \quad (11)$$

where P is a positive constant which is a variable for each problem. This constant can be determined by the user to take into account of the constraints.

4. NUMERICAL EXAMPLE

The test problem is the combined sizing and shape optimization of the 39-bar spatial truss tower shown in Figure 1. Problem specifications are listed in Table 1. Fixed nodes' coordinates and elements' connectivity are presented in Table 2. The top and

bottom nodes have fixed position while middle nodes' coordinates are taken as design variables.

The symmetry of structure is maintained during the optimization process. The population size and allowable number of iterations is set to 40 and 100, respectively. The optimum design found by JA is demonstrated in Figure 2 while in Table 3 it is compared with those obtained from other methods. The solution obtained by JA is lighter than the cited references (Wang et al. 2002) and (Dede and Ayvaz 2015). This optimum design is attained at 100th generation (i.e. after 4000 structural analyses).

Table 1
Input data for the spatial 39-bar truss problem

Properties / constraints	unit	value / notes
Modulus of elasticity	E (GPa)	210
Material density	ρ (kg/m ³)	7800
Stress constraints	σ (MPa)	240 for tension -240 for compression
Displacement constraints	δ (cm)	0.4 for Y directions (nodes 13, 14, 15)
Nodes forces	F (kN)	10 for Y directions (nodes 13, 14, 15)
Euler buckling	σ_e (MPa)	$\sigma_e \leq \frac{-K_e EA_e}{L_e^2}$

Table 2
Initial shape and member grouping for the spatial 39-bar truss problem

Shape variables				Sizing variables	
Joint	x (m)	y (m)	z (m)	Cross-area	node-node
1	0.000	1.000	0.000	A1	(1-4), (2-5), (3-6)
2	-0.866	-0.500	0.000	A2	(4-7), (5-8), (6-9)
3	0.866	-0.500	0.000	A3	(7-10), (8-11), (9-12)
13	0.000	0.280	4.000	A4	(10-13), (11-14), (12-15)
14	-0.242	-0.140	4.000	A5	rest of the element
15	0.242	-0.140	4.000		

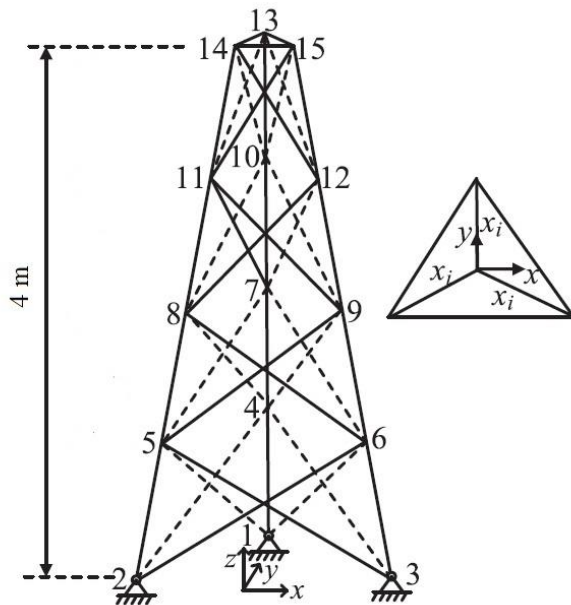


Fig. 1. Initial shape 39-bar truss



Fig. 2. Optimal shape 39-bar truss

Table 3

Comparison results of 39-bar truss problem

Design Variables	Wang et al. 2002	Dede & Ayvaz 2015	This study
A1	11.01	11.9650	12.0000
A2	8.63	11.1457	10.1794
A3	6.69	7.8762	6.5537
A4	4.11	2.7013	2.1396
A5	4.37	2.4058	1.7422
Y4	0.805	0.8996	0.8926
Z4	1.186	1.3507	1.1098
Y7	0.654	0.6917	0.6514
Z7	2.204	2.3122	2.5000
Y10	0.466	0.4825	0.4115
Z10	3.092	3.3031	3.4962
Weight (kg)	203.18	154.13	134.62

5. CONCLUSION

The JAYA algorithm showed a good performance when searching minimum weight of the truss system. It does not require control parameters as in other optimization technique. The design results are compared with the results given in literature. Comparison between the results clearly shows that the proposed algorithm named JAYA can be effectively used in the design of truss structures. To optimize the truss structures a new and efficient algorithm called JAYA is coded in the Matlab.

REFERENCES

- Dede, T., Ayvaz, Y., 2015. *Combined size and shape optimization of structures with a new meta-heuristic algorithm*. Applied Soft Computing, 28, 250–258.
- Dede, T., 2018. *Jaya algorithm to solve single objective size optimization problem for steel grillage structures*. Steel and Composite Structures, 26(2), 163–170.

- Degertekin, S.O., Lamberti, L., Ugur, I.B., 2018. *Sizing, layout and topology design optimization of truss structures using the Jaya algorithm*. Appl. Soft Comput. J., 70, 903–928.
- Grzywiński, M., Dede, T., Özdemir, Y.I., 2019. *Optimization of the braced dome structures by using Jaya algorithm with frequency constraints*. Steel and Composite Structures, 30(1), 47–55.
- Rao, R.V., 2016. *Jaya, A simple and new optimization algorithm for solving constrained and unconstrained optimization problems*. Int. J. Ind. Eng. Comput., 7, 19–34.
- Rao, R.V., Waghmare, G.G., 2017. *A new optimization algorithm for solving complex constrained design optimization problems*. Eng. Optim., 49(1), 60–83.
- Wang, D., Zhang, W., and Jiang, J., 2002. *Combined shape and sizing optimization of truss structures*. Computational Mechanics, 29(4-5), 307–312.
- Grzywiński, M., Selejdak, J., 2019. *Truss optimization using genetic algorithm*. Proceedings of Academicsera 48th International Conference, New York, Institute For Research and Journals, 1-3.