LIGHTWEIGHT STRUCTURES in CIVIL ENGINEERING
CONTEMPORARY PROBLEMS
Monograph from Scientific Seminar
Organized by Polish Chapters of
International Association for Shell and Spatial Structures
Lodz University of Technology
Faculty of Civil Engineering, Architecture and Environmental Engineering
XXIV LSCE
Łódź, 7th of December 2018 (Friday)

TEACHING-LEARNING-BASED OPTIMIZATION ALGORITHM FOR DESIGN OF BRACED DOME STRUCTURES

M. Grzywiński
Adjunct, Faculty of Civil Engineering, Czestochowa University of Technology, Poland, mgrzywin@bud.pcz.pl

ABSTRACT: In this paper, a genetic algorithm is proposed to solve the weight minimization problem of dome structures considering configuration, shape, and sizing design variables. The design of braced dome structures is optimized by an efficient optimization algorithm called Teaching-Learning-Based Optimization (TLBO). The process of TLBO is divided into two parts: the first part consists of the 'Teacher Phase' and the second part consists of the 'Learner Phase'. Analyses of structures are performed by a finite element code in MATLAB which is used in conjunction with an optimization code based on TLBO. The effectiveness of TLBO algorithm is demonstrated through two benchmark braced domes (52-bar, and 56-bar).

Keywords: braced dome, teaching-learning based optimization (TLBO), size and shape optimization, minimum weight.

1. INTRODUCTION

Domes are one of the oldest magnificent structural systems. They consist of one or more layers of elements that are arched in all directions. Domes are used to cover large areas such as exhibition halls, stadium and concert halls. They provide a completely unobstructed inner space and economy in terms of materials. They are lighter compared with the more conventional forms of structures (Ref.1).

Structural optimal design has always been a concern for engineers in practice. The focus is not only in construction cost, but also in geometry of structures. It is responsible for engineers to design structures with high reliability and low cost. For these purposes, many optimal algorithms were investigated to accomplish the tasks including the classical methods and the innovative algorithms.

In the early 1990s, the genetic algorithm was presented in Ref.2, then, it was made significant achievements in structural optimization fields (Ref. 3-5).

Teaching-learning-based optimization (TLBO) is a recently proposed metaheuristic algorithm [6-8]. The easy and effectiveness of TLBO were supported by research works of many other researchers published (Ref. 9-11). In the problem of size and geometry optimization of truss structures, the cross-sectional area and the geometry of primary structures both increase the dimension of the design space. It has been proved that TLBO algorithm performs well in problems with large dimensions (Ref. 12).

2. TLBO ALGORITHM

One of the recently developed metaheuristics is teaching-learning-based- optimization (TLBO) algorithm (Ref. 6, 8). TLBO has many similarities to evolutionary algorithms (EAs): an initial population is randomly selected, moving on the way to the teacher and classmates is comparable to mutation operator in EA, and selection is based on comparing two solutions in which the better one always survives. Similar to most other evolutionary optimization methods, TLBO is a population-based algorithm inspired by learning process in a classroom. The searching process consists of two phases, i.e. Teacher Phase and Learner Phase. In teacher phase, learners first get knowledge from a teacher and then from classmates in learner phase. In the entire population, the best solution is considered as the teacher $X_{teacher}$. On the other hand, learners learn from the teacher in the teacher phase. In this phase, the teacher tries to enhance the results of other individuals X_i by increasing the mean result of the classroom X_{mean} towards his/her position $X_{teacher}$. In order to maintain stochastic features of the search, two randomly-generated parameters r and T_F are applied in update formula for the solution X_i as:

$$X_{new} = X_i + r(X_{teacher} - T_F \cdot X_{mean}) \tag{1}$$

where r is a randomly selected number in the range of 0 and 1 and T_F is a teaching factor which can be either 1 or 2:

$$T_F^i = round[1 + rand(0,1)\{2-1\}] \tag{2}$$

Moreover, X_{new} and X_i are the new and existing solution of i, (Ref. 6).

In the second phase, i.e. the learner phase, the learners attempt to increase their information by interacting with others. Therefore, an individual learns new knowledge if the other individuals have more knowledge than him/her. Throughout this phase, the student X_i interacts randomly with another student X_j ($i \neq j$) in order to improve his/her knowledge. In the case that X_j is better than X_i (i.e. $f(X_j) < f(X_i)$ for minimization problems), X_i is moved toward X_j. Otherwise it is moved away from X_j:

$$X_{new} = X_i + r(X_j - X_i) \text{ if } f(X_i) > f(X_j) \tag{3}$$

$$X_{new} = X_i + r(X_i - X_j) \text{ if } f(X_i) < f(X_j) \tag{4}$$

If the new solution X_{new} is better, it is accepted in the population. The algorithm will continue until the termination condition is met. The pseudo code shown in Table 1 demonstrates the TLBO algorithm step-by-step.

Table 1. Pseudo code for TLBO

Set k=1;
Objective function f(X), $X=(x_1,x_2,\dots,x_d)^T$
d=no. of design variables
Generate initial students of the classroom randomly
X_i, $i=1,2,\dots,n$, n=no. of students
Calculate objective function f(X) for whole students of the classroom
WHILE (the termination conditions are not met)
{Teacher Phase}
Calculate the mean of each design variable X_{mean}
Identify the best solution (teacher)
FOR $i=1\rightarrow n$
Calculate teaching factor $T_F^i=round[1+rand(0,1)\{2-1\}]$
Modify solution based on best solution(teacher)
$X_{new}^i=X^i+rand(0,1)[X_{teacher}-T_F^i\cdot X_{mean}]$
Calculate objective function for new mapped student $f(X_{new}^i)$
IF X_{new}^i is better than X^i $X^i=X_{new}^i$
END IF {End of Teacher Phase}
{Student Phase}
Randomly select another learner X_j $(i\neq j)$
IF X^i is better than X^j
$X_{new}^i=X^i+rand(0,1)(X^i-X^j)$
ELSE
$X_{new}^i=X^i+rand(0,1)(X^j-X^i)$
END IF
IF X_{new}^i is better than X^i $X^i=X_{new}^i$
END IF {End of Student Phase}
END FOR
Set k=k+1
END WHILE

3. FORMULATION OF OPTIMIZATION

One of the most important factors in the structural design is the total structural weight. In this study, dome structures are designed to be the minimum weight. For this aim, the objective function for the dome structures is formulated as:

minimize:

$$W(A,x)=\sum_{i=1}^{n} L_i(x_i)\rho_i A_i \tag{5}$$

subject to:

$$\sigma_i \le \sigma_{up} \tag{6}$$

$$\delta_i \le \delta_{up} \tag{7}$$

$$A_{low} \le A_i \le A_{up} \tag{8}$$

$$x_{low} \le x_i \le x_{up} \tag{9}$$

where A (bar cross-sectional areas) and x (nodal coordinates) are the design variable, respectively; W is the weight of the dome; ρ_i and L_i is the material density and the length of the i-th element, respectively; *n* is numbers of bars in the dome; σ_i is element stress, and δ_i is node displacement of the dome. Inequality (8) and (9) indicates that the design variables including either a shape or sizing variable must take a value between lower and upper bounds.

$$c_i=\sigma_i/\sigma_{up} \quad \text{and} \quad c_i=\delta_i/\delta_{up} \tag{10}$$

Where c_i is 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:

$$C=\sum_{i=1}^{nc} c_i \tag{11}$$

Where nc is the number of the constraints. Objective function is changed to penalized objective function by adding penalty function to it. The penalized objective function, $\Phi(A,x)$, can be formulated as:

$$\Phi(A,x)=W(A,x)[1+P\cdot C] \tag{12}$$

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 EXAMPLES

To demonstrate the proposed solution method a braced dome structure is presented as a sizing problem with discrete design variables and second solution as a shape optimization with continuous design variables.

In order to compare the effect of bars configurations on the solution two schemas with different topologies were proposed.

Properties of applied material for both examples are shown in Table 2.

Table 2. Structural constraints and material properties

Properties / constraints	unit	value / notes
Modulus of elasticity	E (N/m^2)	2.1×10^{11}
Material density	ρ (kg/m^3)	7800
Displacement constraints	δ (m)	0.1 for each directions
Stress constraints	σ (N/m^2)	2.0×10^{8} for tension 1.5×10^{8} for compression

4.1. Example 1

The 52-bar dome is subjected to a downward vertical equipment loading of 30 kN at nodes 1, 6-13, and 60 kN at nodes 2-5, and simply supported at nodes 14-21. The geometry and nodal coordinates are presented in Fig. 1.

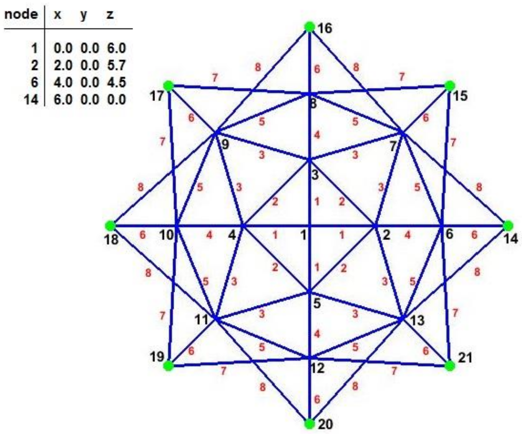

Fig. 1 Plan view and element grouping for 52-bar dome

4.2. Example 2

The 56-bar dome is subjected to a downward vertical equipment loading of 30 kN at nodes 1-17, and simply supported at nodes 18-25. The geometry and nodal coordinates are presented in Fig. 2.

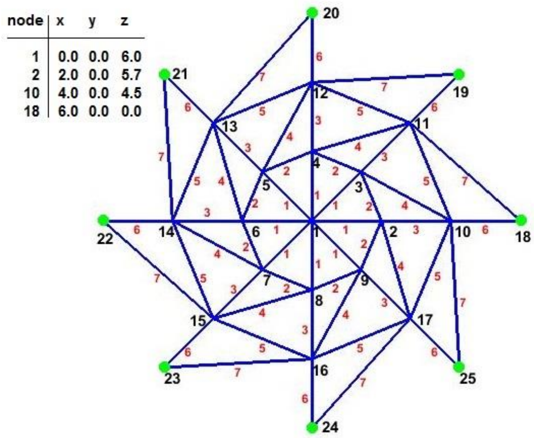

Fig. 2 Plan view and element grouping for 56-bar dome

4.3. Discrete size optimization

The structural element of example 1 (52-bar) is classified in 8 groups for the discrete size optimization. Example 2 (56-bar) has 7 groups, respectively. Thus, the cross-sections area are the design variables for the size optimization, respectively.

Table 3. Available cross-section area

Pipe D*t	CS [cm2]	Pipe D*t	CS [cm2]	Pipe D*t	CS [cm2]	Pipe D*t	CS [cm2]
20*2	1.13	-	-	-	-	-	-
30*2	1.76	30*3	2.54	-	-	-	-
40*2	2.39	40*3	3.49	40*4	4.52	-	-
50*2	3.01	50*3	4.43	50*4	5.78	50*5	7.07
-	-	60*3	5.37	60*4	7.03	60*5	8.64
-	-	70*3	6.31	70*4	8.29	70*5	10.21
-	-	80*3	7.25	80*4	9.55	80*5	11.78
-	-	90*3	8.20	90*4	10.80	90*5	13.35

D – pipe diameter [mm], t – pipe thickness [mm], CS – cross-section

Available cross-sections are shown in Table 3. The size of population is Pn=30 and the number of generation is Gn=100 for both examples. Result are shown in Table 4.

Table 4. Optimal solutions for size optimization

Design variables		Example 1 52-bar	Pipe D*t	Example 2 56-bar	Pipe D*t
Size optimization cross-sec. area (cm2)	A_1	3.49	40*3	1.76	30*2
	A_2	1.76	30*2	3.01	50*2
	A_3	2.54	30*3	4.43	50*3
	A_4	5.37	60*3	1.13	20*2
	A_5	2.54	30*3	2.39	40*2
	A_6	4.52	40*4	5.37	60*3
	A_7	1.13	20*2	1.13	20*2
	A_8	1.13	20*2	-	-
Weight (kg)		401.7679		392.1861	
Pn/Gn		30/100		30/100	
Run		10		10	

4.3. Continuous shape optimization

The structural element for example 1 (52-bar) is classified 8 groups for the discrete size optimization and nodal points are classified 5 groups for the continuous shape optimization by preserving the structural symmetry. Thus, the cross-sections area and the coordinates are the design variables for the size and shape optimization, respectively. The example 2 (56-bar) has 7 groups for the discrete size optimization and 5 group of the continuous shape optimization, respectively.

The size of population is Pn=30 and the number of generation is Gn=100 for both examples. The optimal solutions for example 1 and 2 are summarized in the Table 5 and 6.

Table 5. Optimal solutions for shape optimization

Design Variables	Initial	Example 1 52-bar	Example 2 56-bar
Z1	6.00	6.3246	5.5642
X2	2.00	2.3529	2.5480
Z2	5.70	4.7277	4.9855
X6 or X10	4.00	3.1168	3.7788
Z6 or Z10	4.50	3.5272	3.5060

To show the performance of the used algorithm the best solution, mean solution, standard deviation, number of independent runs and the number of function evaluation are given in the same tables. The history of the best solution, mean solution and the standard deviation obtained using the TLBO algorithm for example 1 and 2 are given the Fig. 3 and Fig. 4, respectively.

Table 6. Optimal solutions for size optimization

Design Variables		Example 1 52-bar	Pipe D*t	Example 2 56-bar	Pipe D*t
Size optimization cross-sec. area (cm2)	A_1	1.13	20*2	1.13	20*2
	A_2	1.13	20*2	1.13	20*2
	A_3	1.76	30*2	3.01	50*2
	A_4	4.43	50*3	1.13	20*2
	A_5	1.13	20*2	1.13	20*2
	A_6	3.01	50*2	5.37	60*3
	A_7	2.39	40*2	1.13	20*2
	A_8	1.13	20*2	-	-
Weight (kg)		295.1062		288.2896	
Mean		297.3514		301.2919	
Std		3.1918		4.4355	
Pn/Gn		30/100		30/100	
Run		20		20	

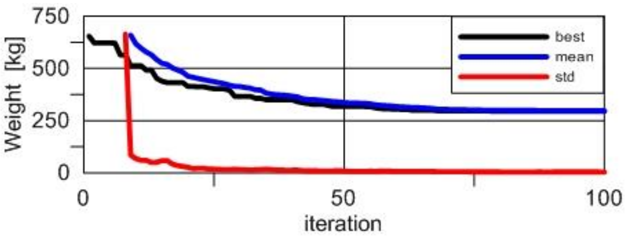

Fig. 3 Convergence graph for 52-bar dome

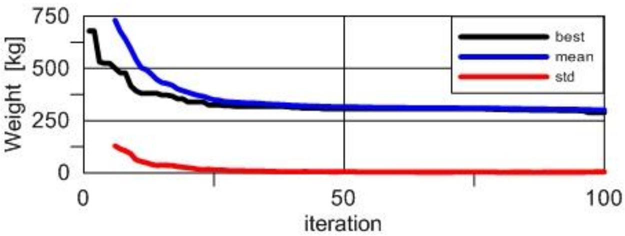

Fig. 4 Convergence graph for 56-bar dome

To show the diversity of the independent runs for the both examples the weight of the related structure are given in the Table 7. The elapsed mean times to solve each example are given in this table. The computer used to solve these examples has the properties as; Intel(R) Core(TM) i7-4510U CPU @ 2.60GHz 8,00 GB, Windows 8.1.

Table 7. Diversity of the run for two example

Run no.	52-bar dome	56-bar dome
1	317.8353	334.1212
2	328.9207	325.8887
3	325.0644	288.2896
4	310.2879	323.5695
5	321.9327	312.0605
6	297.8441	320.9292
7	308.5499	322.2623
8	327.8622	298.7914
9	304.5463	334.1212
10	308.8461	334.1212
11	307.2416	322.2566
12	337.9143	325.4253

Table 7. Diversity of the run for two example (cont.)

Run no.	52-bar dome	56-bar dome
13	336.4490	320.9334
14	331.6691	301.9716
15	321.3125	292.1548
16	307.3391	320.9289
17	307.2191	293.7066
18	321.5014	297.0656
19	297.6162	283.6471
20	295.1062	282.9497
Best	295.1062	288.2896
CPU time [s]	300.7543	327.7893

5. CONCLUSION

Size and shape optimization of 3D dome structure are investigated in this study. To optimize the dome structures a new and efficient algorithm called TLBO is coded in the Matlab. Better topology is 56-bar dome. Higher weight reduces shape optimization, compared to size optimization.

Table 8. Results of optimization mass the domes

Braced dome	Size optimization	Shape optimization	Reduction mass
52-bar	401.77 kg	295.11 kg	26.5 %
56-bar	392.19 kg	288.29 kg	26.5 %
	2.4 %	2.4 %	

6. ACKNOWLEDGEMENTS

The author thanks Prof. Tayfun DEDE from Department of Civil Engineering, Karadeniz Technical University, 61080, Trabzon, Turkey, for his help and borrow program code in MATLAB.

REFERENCES

1. Z.S. Makowski (ed.): Analysis, design and construction of braced domes. Granada Publishing Ltd.; 1984.
2. D.E. Goldberg: Genetic algorithms in search, optimization, and machine learning, Addison Wesley, 1989.
3. C. Camp, S. Pezeshk, G.Z. Cao: Optimized design of two-dimensional structures using a genetic algorithm, Struct Eng, 124(5), 1998, pp. 471-476.
4. M. Grzywiński: Optimization of Single-Layer Braced Domes, Transactions of the VŠB – Technical University of Ostrava, Civil Engineering Series, 17(1), 2017, pp. 45-50.
5. S. Rajeev, C.S. Krishnamoorthy: Discrete optimization of structures using genetic algorithms, Struct Eng, 118(5), 1992, pp. 1233-1250.
6. R.V. Rao, V.J. Savsani, and D.P. Vakharia: Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems. Computer-Aided Design, 43(3), 2011, pp. 303-315.
7. R.V. Rao, V. Patel: An elitist teaching-learning-based optimization algorithm for solving complex constrained optimization problems, Inter J Indust. Eng. Comput., 3(1), 2012, pp. 535-560.
8. R.V. Rao: Teaching Learning Based Optimization Algorithm And Its Engineering Applications, Springer, 2016.
9. V. Toğan: Design of planar steel frames using teaching–learning based optimization, Eng Struct, 34(0), 2012, pp. 225-232.
10. T. Dede: Application of Teaching-Learning-based-Optimization Algorithm for the Discrete Optimization of Truss Structures. KSCE Journal of Civil Engineering, 18(6), 2014, pp. 1759-1767.
11. W. Cheng, F. Liu, L.J. Li: Size and geometry optimization of trusses using teaching-learning-based optimization, Int. J. Optim. Civil Eng., 3(3), 2013; pp. 431-444
12. R.V. Rao, V.J. Savsani, D.P. Vakharia: Teaching-learning-based optimization: An optimization method for continuous non-linear large scale problems. Information Sciences, 183, (2012), pp. 1-15.