ASIAN JOURNAL OF CIVIL ENGINEERING (BHRC) VOL. 16, NO. 5 (2015)
PAGES 567-585

A FAST MARGINAL FEASIBILITY SEARCH METHOD IN SIZE OPTIMIZATION OF TRUSS STRUCTURES

B. Dizangian and M.R. Ghasemi*
[1]Department of Civil Engineering, University of Sistan and Baluchestan, Zahedan, Iran

Received: 20 January 2015; **Accepted:** 10 April 2015

ABSTRACT

Maybe it is reasonable that in optimization problems based on sensitivity analysis one could reach the vicinity of the optimum point with a minor number of analyzes. Besides, it is also fair to accept the fact that the optimum point activates at least one constraint in constrained optimization problems. Based on these concepts a new method is proposed in the present study. It utilizes four well-organized operators to reach the global optimum solution; ordered by first a Subspace Search (SS) operator that transforms the whole design space into a series of subspaces in order to rapidly reach the Feasible-Non-Feasible (FNF) margin at the early stages by doing a few number of analyzes. It is then followed by a Marginal Sensitivity Analysis (MSA) operator that determines the sensitivity degree of each design variable to constraints violation, near the margin of FNF region. Next, the Marginal Search (MS) operator is used to determine a local optimum point near the FNF border in the feasible region. Finally, the roulette wheel (RW) operator is employed to select, in a random manner, only one variable for updating in each iteration. The robustness and effectiveness of the proposed method is verified on several well-known benchmark truss examples. The results show that the proposed method not only speeds up the optimization procedure, but also it ensures the non-violated global optimum design point without a need for multiple runs.

Keywords: Marginal; feasibility search; size optimization; truss structures.

1. INTRODUCTION

Over the lasts decades, a number of optimization methods have been developed for optimization of trusses. Genetic algorithms ([1-5]) and simulated annealing algorithms ([6, 7]) are the most notable stochastic-based optimization techniques used for the optimum design of trusses.

More recently, another branch of nature-inspired algorithms have attracted the attention of researchers in all optimization field including structural problems. Algorithms belonging

*E-mail address of the corresponding author: mrghasemi@hamoon.usb.ac.ir (M.R. Ghasemi)

to this field imitate the collective behavior of a group of social insects like bees, cuckoo and ants (see e.g., [8-15]). In all of nature-inspired methods, a number of individuals called agents have a duty of finding better solution by means of time consuming step-by-step iterations; besides optimization procedure starts from fully random design points. However, the main problem arises while as the number of design variables increases, raising the number of agents is inevitable.

The introduced method here exhibits a less time-consuming procedure comparing to those recent optimization methods. However, in most of problems, a method that could soon find the near global optimum point in early stages may be necessary. Details of the proposed sensitivity-based method may be summarized into the four following operators; first the SS operator is activated whereby the searching design space is remarkably confined in the early stages of the procedure; second the sensitivity values of constraints violation of each design variable near the margin of FNF region are determined by employing the MSA operator; third the MS operator that is used to search near the border-line of FNF to find better solutions and fourth, RW operator that includes random selective approach for the most decision-maker variables that finally causes a movement into the feasible region to first of all passing probably over at least a local optimum point and secondly gives more space to other variables to further modify their values for a better objective value.

A serial exploitation of these four operators through successive iterations ensures the fact that 1. The founded global optimum point is feasible, 2. The searching space is confined at early stages, 3. The more sensitive and decisive variables are known at each iteration through optimization procedure and finally 4. It avoids a simultaneous alteration of all design variables, a fact that existed in other optimization procedures. These steps reason a major reduction on the computational time required through structural optimization procedure.

2. SRUCTURAL DESIGN OPTIMIZATION

2.1 General formulation

Most engineering optimization problems may be expressed as minimizing (or maximizing) a function subject to inequality and equality constraints and can be stated as the general form

$$\begin{array}{lll} \text{Minimize/Maximize} & f(\boldsymbol{X}) & \\ \text{subject to} & g_i(\boldsymbol{x}) \le 0 & i=1,\ldots,m \\ \text{and/or} & h_j(\boldsymbol{x}) = 0 & j=1,\ldots,l \\ \text{and/or} & \boldsymbol{x}^L \le \boldsymbol{x} \le \boldsymbol{x}^U & \end{array} \tag{1}$$

where $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T$ is a column vector of *n* design variables. In Eq. (1) , f is the objective or cost function, g's are inequality constraints, and h's are equality constraints. $\boldsymbol{x}^L$ and $\boldsymbol{x}^U$ are allowable lower and uper bounds of $\boldsymbol{x}$, respectively. A design $\boldsymbol{x}$ that satisfies all inequality and equality constraints is feasible.

2.2 Size optimization of trusses

Size optimization of structural systems involves arriving at optimum values for member cross-sectional areas A that minimize an objective function f, usually the structural weight W. This is expressed mathematically as

$$\text{Minimize} \quad f = W(\boldsymbol{A}) = \sum_{j=1}^{n} A_j\, L_j \rho_j \tag{2}$$

Where A_j,L_j and ρ_j are the cross-sectional area, length and unit weight of jth member, respectively and n is the total number of members. The vector $\boldsymbol{A}$ is selected between lower A^l and upper A^u bounds. Eq. (2) is subjected to the following normalized constraints ([16]).

$$\begin{aligned}
s_m(\boldsymbol{A}) &= \max\left(0, \frac{\sigma_m}{\sigma_{m,allowed}} - 1\right), \quad m = 1,2,\dots,NE \\
b_m(\boldsymbol{A}) &= \max\left(0, \frac{\lambda_m}{\lambda_{m,allowed}} - 1\right), \quad m = 1,2,\dots,NE \\
d_k(\boldsymbol{A}) &= \max\left(0, \frac{u_k}{u_{k,allowed}} - 1\right), \quad k = 1,2,\dots,ND
\end{aligned} \tag{3}$$

Where NE and ND denotes the number of members and the number of degrees of freedom of structure, respectively; s_m, b_m, and d_k are respectively, the member stress, member buckling and nodal displacement normalized constraint functions; σ_m and λ_m are the stress and the slenderness ratio of mth member; $\sigma_{m,allowed}$ and $\lambda_{m,allowed}$ are the allowable axial stress and allowable slenderness ration for mth member, respectively; $u_{k,allowed}$ and u_k are the allowable displacement and nodal displacement of kth degrees of freedom, respectively. In Eq. (3), all the normalized constraint functions are activated when the violated constraints have values larger than zero. In this paper the Squared Normalized Degree of Constraints Violation (SNDCV) Z and penalty function γ are used, where

$$Z(\boldsymbol{x}) = \left(\sum_{m=1}^{NE} (s_m + b_m) + \sum_{k=1}^{ND} d_k\right)^2 \tag{4}$$

And;

$$\gamma(\boldsymbol{x}) = (1 + Z) \tag{5}$$

Then, the penalized objective function of the structure (in a minimization problem) is considered as Eq. (6)

$$F'(\boldsymbol{x}) = f(\boldsymbol{x})\gamma(\boldsymbol{x}) \tag{6}$$

3. AN EFFICIENT SUBSPACE SEARCH (SS) OPERATOR

One of the most important tasks of the proposed method is to find a first local optimum point in early stages. To do so, concept of uniform random numbering is utilized. Stochastic-based methods begin their search from random points, a number of which depends on the number of variables and complexity of the problem. For instance, in Genetic Algorithm (GA) Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO) etc., the initial design points are spread randomly in all parts of searching space. In such methods, if number of these initial random points are not consistent with the number of variables or the complexity of the problem, it affects the convergence history through the optimization procedure and therefore affects the quality of the global optimum solution ([17 -19]). The major feature of the proposed subspace search is the independence of number of initial samples produced to the complexity and/or number of design variables of the problem.

To detail the step, first the initial values for design variables are generated using an integer value powered to ten multiplied by a vector of uniformly random values. This technique could be very efficient in cases where limiting bounds for the design variables are not known and as well, one could determine the near global optimum solution in a least amount of analyses to be taken place.

These initial random designs will be produced for the subspaces of positive powers of ten using Eq. (7) or for the subspaces of the negative powers of ten using Eq. (8) as follows

$$\boldsymbol{x}^s = rand(b_L^+, b_U^+) \quad \text{for } s \geq 0 \quad \text{(positive searching region)} \tag{7}$$
$$\boldsymbol{x}^s = rand(b_L^-, b_U^-) \quad \text{for } s < 0 \quad \text{(negative searching region)} \tag{8}$$

where the lower and upper bounds (b_L^+, b_L^- and b_U^+, b_U^-) are determined using the following relations

$$b_L^+ = |< s-1 > -s| \times 10^{(s-1)} \quad \text{for } s \geq 0 \quad \text{(positive searching region)} \tag{9}$$
$$b_L^- = |< -s-1 > +s| \times 10^{(s)} \quad \text{for } s < 0 \quad \text{(negative searching region)} \tag{10}$$
$$b_U^+ = 10^{(s)} \quad \text{for } s \geq 0 \quad \text{(positive searching region)} \tag{11}$$
$$b_U^- = 10^{(s-1)} \quad \text{for } s < 0 \quad \text{(negative searching region)} \tag{12}$$

where s is an integer number and | | is the absolute operator. Angle brackets < > are used base on the following definitions

$$< M >= \begin{cases} 0, & M < 0 \\ M, & M \geq 0 \end{cases} \tag{13}$$

This technique could be very efficient in cases where limiting bounds for the design variables are not known and as well one could determine the near global optimum solution in a least amount of analyses to be taken place.

4. MARGINAL SENSITIVITY ANALYSIS (MSA) OPERATOR

In order to perform sensitivity analysis, finite difference method is employed. The sensitivity of Squared Normalized Degree of Constraints Violation $Z(\boldsymbol{x})$ at the current design vector $\boldsymbol{x}$ using *backward and forward difference method* can be approximated as

$$S_i^{\pm} = \frac{Z(\boldsymbol{x} \pm \Delta\boldsymbol{x}(i)) - Z(\boldsymbol{x})}{\Delta\boldsymbol{x}(i)} \tag{14}$$

if $\boldsymbol{x}$ is feasible, the above equation can be reformulate as

$$S_i^{\pm} = \frac{Z(\boldsymbol{x} \pm \Delta\boldsymbol{x}(i))}{\Delta\boldsymbol{x}(i)} \tag{15}$$

Choosing the positive or negative sign in Eqs. (14 and 15) is such that the search carries into the non-feasible region rather than more feasible. In the present study $\Delta\boldsymbol{x}(i)$ is called self-adaptive dynamic perturbation value, given by Eq. (16)

$$\Delta\boldsymbol{x}(i) = \frac{\boldsymbol{x}(i)}{\alpha} \tag{16}$$

where $\boldsymbol{x}(i)$ is the *i*th variable and α is a fixed large number, called the sensitivity step length parameter. The Scaled Sensitivity Value SSV_i for *i*th variable, is calculated using the following equation

$$SSV_i = \frac{S_i}{\sum_1^n S_i} \tag{17}$$

where n is the total number of design variables. The feature of Eq. (17) is that it shows the sensitivity rate of a variable to other variables.

5. MARGINAL SEARCH (MS) OPERATOR

By having the *SSV* for each design variable, Marginal Search (MS) operator is used to perform an efficient search near the borderline of FNF inside the feasible region until a local optimum is found. The variables with less sensitivity play the main role of allowing the searching procedure. The pseudo-code of the Fig. 1 clarifies the MS operator.

In Fig. 1, $\boldsymbol{\lambda}(var)$ is called marginal search step-length which is determined by reducing value of the *var*th design variable $\boldsymbol{x}(var)$ as given in Eq. (19)

$$\boldsymbol{\lambda}(var) = \frac{\boldsymbol{x}(var)}{e} \tag{18}$$

where e parameter is related to adjusting accuracy to find a sequence of boundary point. That is, the greater e is, the smaller the step length $\boldsymbol{\lambda}(var)$ will be, because of which the border of a FNF region will be found causing a fast step towards the local optimum.

The signs + or – in Fig. 1 are chosen such that the search is carried out from the feasible region closer to the vicinity of the border of FNF.

Marginal Search (MS) operator

Input: Design vector $\boldsymbol{x}$, $\boldsymbol{x}^{Sorted}$
Output: local optimum point $\boldsymbol{x}^{\boldsymbol{local}}$, $\boldsymbol{x}^{best}$, f^{best}
<u>Start</u>
Set $\boldsymbol{x}^{best}$=$\boldsymbol{x}$, f^{best}=$f(\boldsymbol{x})$
for *var* = $\boldsymbol{x}^{Sorted}$; **do**:
  set $j = 0$; define $\boldsymbol{\lambda}$= a column search vector with n entries equal to zero
while $Z = 0$
$j = j + 1$
  $\boldsymbol{\lambda}(var) = \frac{x(var)}{e}$
$\boldsymbol{x}' = \boldsymbol{x} \pm j\boldsymbol{\lambda}$
    Determine the value of objective function f^{new} for $\boldsymbol{x}'$.
        Compute the Degree of Constraint Violation Z^{new} for $\boldsymbol{x}'$.
        if f^{new} is better than f^{best} and $V^{new}=0$
            $f^{best} = f^{new}$
            $\boldsymbol{x}^{best}$=$\boldsymbol{x}$
        end if
        if $Z^{new} = 0$
            $\boldsymbol{x}(var) = \boldsymbol{x}(var) \pm j\boldsymbol{\lambda}(var)$
        else
            break
        end if
 end while
end for
$\boldsymbol{x}^{local} = \boldsymbol{x}$
return $\boldsymbol{x}^{local}$, $\boldsymbol{x}^{best}$, f^{best}
<u>End</u>

Figure 1. The pseudo-code of Marginal Search (MS) operator

6. ROULETTE WHEEL (RW) OPERATOR

The pseudo-code of Fig. 2 clarifies the RW operator. Based on the computed vector of Scaled Sensitivity Values (SSV) using MSA operator, RW operator finally selects the desired variable for updating.

Roulette Wheel (RW) operator

Input: Design vector $\boldsymbol{x}$**,** vector of the Scaled Sensitivity Values $\boldsymbol{SSV}$

Output: A desired variable x^{DV} for updating

<u>Start</u>

Tot=0;

count=0;

Produce uniform random number q in the range of 0 and 1.

While *Tot*<*q* **do**:

count= count+ 1;

*Tot= Tot+***SSV***(count);*

end while

return x^{DV}**=** $\boldsymbol{x}$(*count*)

<u>End</u>

Figure 2. The pseudo-code of Roulette Wheel (RW) operator

7. DESCRIPTION OF THE PROPOSED MARGINAL FEASIBILITY SEARCH METHOD

To describe the methodology of the algorithm, a flowchart is given as in Fig. 3. In the following, all the new formulations involved in the procedure will be introduced by using some steps towards finding the optimum solution $\boldsymbol{x}^G$.

Step 1. Initialization: Initialize optimization algorithm parameters such as maximum number of iterations ($Maxiter$) and step-length parameters α and e to *minimize* objective function $f(\boldsymbol{x})$. Here, the structural analyzer package was developed in MATLAB toolbox by introducing a self-written code, based on FEM.

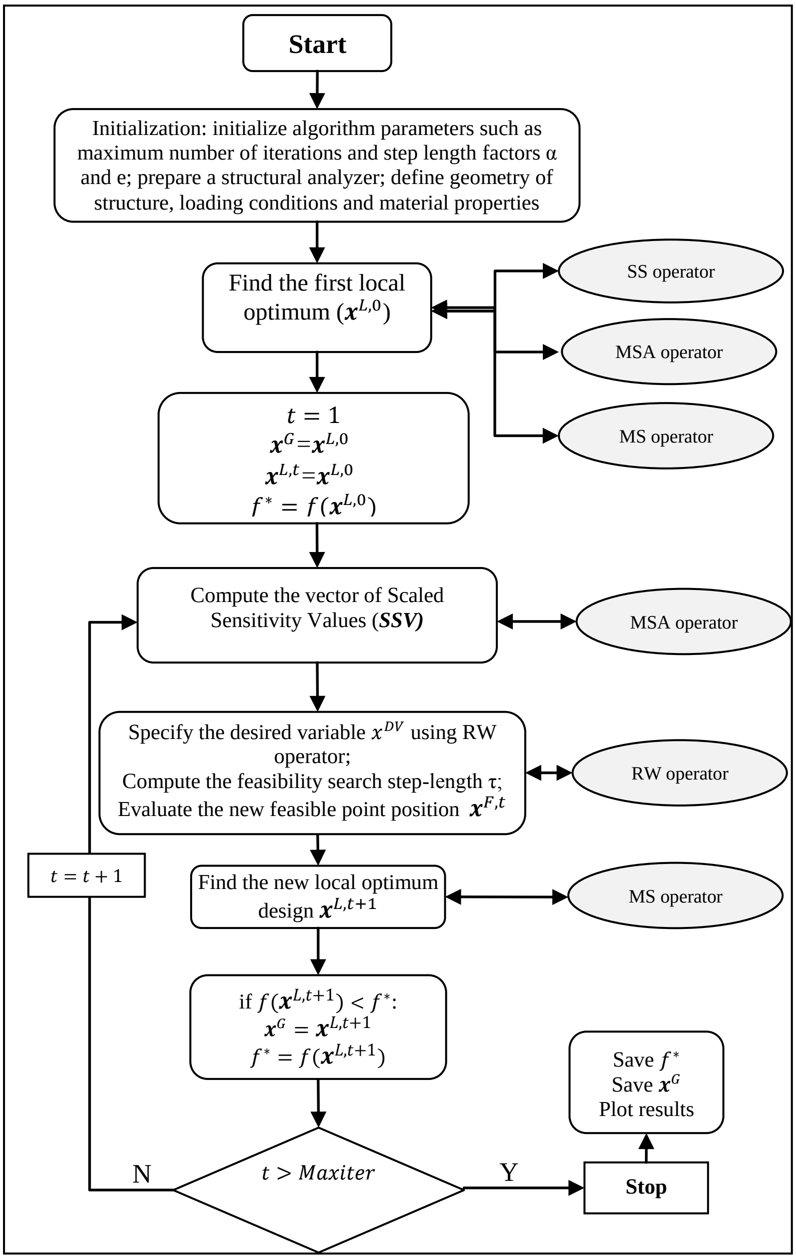

Figure 3. Flowchart of the proposed marginal feasibility search method

Step 2. Find the first local optimum point $\boldsymbol{x}^{L,0}$

Sub-step 2.1: Generate the initial random design using the formulations of SS operator in order to find the first feasible subspace. The first feasible random design is considered as the starting point.

Sub-step 2.2: Determine the Scaled Sensitivity Values *SSVs* of each variable at the starting point using Eqs. (15-17). Sort design variables by the values of *SSV* in ascending order.

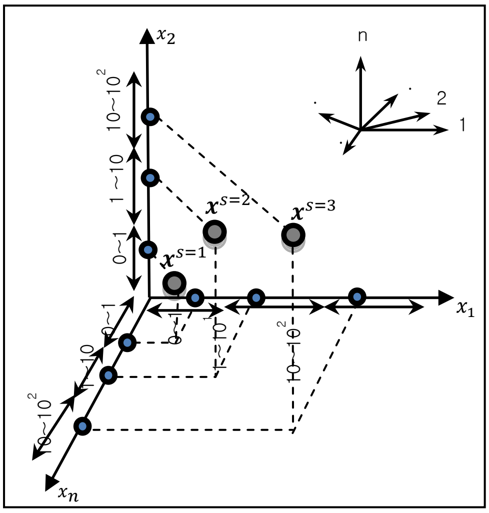

Figure 4. Generating the initial random designs in different subspaces by employing SS operator

Sub-step 2.3: determine the first local optimum point $\boldsymbol{x}^{L,0}$ using the MS operator.

Step 3. Main optimization procedure, find the global optimum solution $\boldsymbol{x}^G$

Sub-step 3.1: Initialize iteration counter $t = 1$, set the current optimum solution $\boldsymbol{x}^{L,t}$=$\boldsymbol{x}^{L,0}$ and $f^* = f(\boldsymbol{x}^{L,0})$.

Sub-step 3.2: Compute the Scaled Sensitivity Values *SSVs* at $\boldsymbol{x}^{L,t}$ for each variable using Eqs. (15-17).

Sub-step 3.3: Employ RW operator to specify the desired variable (DV) x^{DV}. The corresponding value of *SSV* for x^{DV} is called (*DSSV*).

Sub-step 3.4: Consider the desired variable x^{DV} as the feasibility search direction. Compute the feasibility search step-length τ by the following equation

$$\tau = DSSV\ x^{DV} \tag{19}$$

Sub-step 3.5: Determine the new feasible design $\boldsymbol{x}^{F,t}$ by updating the desired variable using Eq. (20)

$$x^{DV}_{new} = x^{DV} + \tau = (1 + DSSV)x^{DV} \tag{20}$$

It must be noted that updating is done only to the value of desired variable and other variables are kept fixed. This effort is established by causing degradation of the expense of the desired variable and as a result allowing other design variables to increase their significance by reducing their quantitative value. This will create the possibility of reaching to another local or perhaps the global optimum. Thus, from here onwards the cyclic procedure of reducing values of variables by using the MS operator is carried out until the global optimum is reached.

Sub-step 3.6: Evaluate the new local optimum design $\boldsymbol{x}^{L,t+1}$ from the feasible design $\boldsymbol{x}^{F,t}$ using MS operator.

Sub-step 3.7: If the new local optimum solution is better than the best solution $f(\boldsymbol{x}^{L,t+1}) < f^*$, then $f^* = f(\boldsymbol{x}^{L,t+1})$ and $\boldsymbol{x}^G = \boldsymbol{x}^{L,t+1}$.

Sub-step 3.8: Set $t = t + 1$.

Sub-step 3.9: If iteration counter t is greater than *Maxiter*, go to next step; otherwise, go to *substep 3.2*.

Sub-step 3.10: Stop the procedure, save the best solution $\boldsymbol{x}^G$ and corresponding objective value f^* and plot the convergence history of best objective value f^*.

8. TEST EXAMPLES

In this section, in order to just perform the robustness and efficiency of the proposed technique, a number of numerical benchmark examples of two and three-dimensional truss structures were attempted. They include a 10-bar planar truss subjected to a single load condition, a 25-bar and a 72- bar space trusses subjected to multiple load conditions. For all examples, the parameters α and e were set as a fixed value of 100.

8.1 Ten-bar planar truss

This problem was first addressed in Venkayya [20] and has since become a benchmark problem to test and verify the efficiency of optimization methods. The topology and nodal numbering of the 10-bar planar truss structure are shown in Fig. 5. The truss was previously studied by many researchers such as Camp et al. [21], Li et al. [22], Lee and Geem [23] and Sonmez [14]. The material has a modulus of elasticity of 10^7 psi (6.895 Gpa.) and a mass density of 0.1 lb./in^3 (27200 N/m^3). The constraints contain maximum displacement limitations of ± 2 in. (5.08 cm) on any node in both *x* and *y* direction and the allowable axial stress of ± 25 ksi (172.25 MPa) in any member. The truss is subjected to two loadings condition: Load case 1; in which the point load P_1=100 kips (445 kN) and P_2=0 and load case 2 consists of point loads P_1=150 kips (667.5 kN) and P_2=50 kips (222.5 kN).

According to Fig. 6 at the start of phase 1 for both loading cases, the third initial design $\boldsymbol{x}^{s=3}$ that was generated randomly within (10-100 in^2) bounds, offered a solution without penalization. It was therefore considered as the starting point.

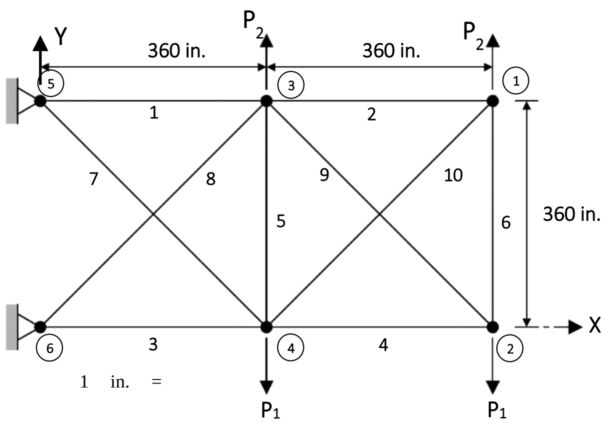

Figure 5. Ten-bar planar truss

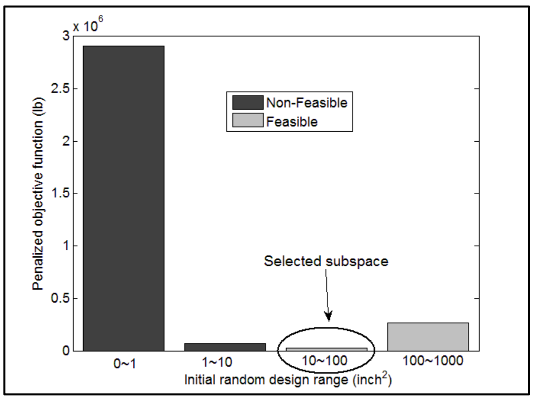

Figure 6. Selecting initial random designs by dividing the searching space into subspaces; the 10-bar truss example

The bar graphs of Figs. 7 and 8 show the results of MSA operator at $\boldsymbol{x}^{L,0}$ for the two loading cases.

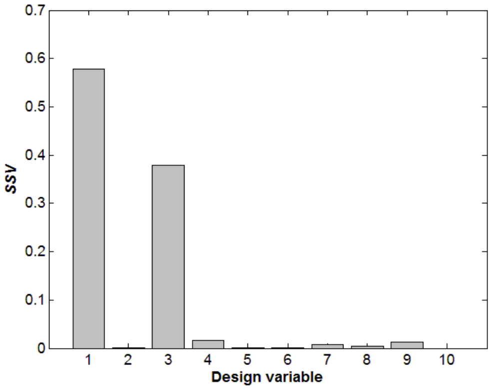

Figure 7. First sensitivity analysis at $\boldsymbol{x}^{L,0}$; 10-bar truss, load case 1

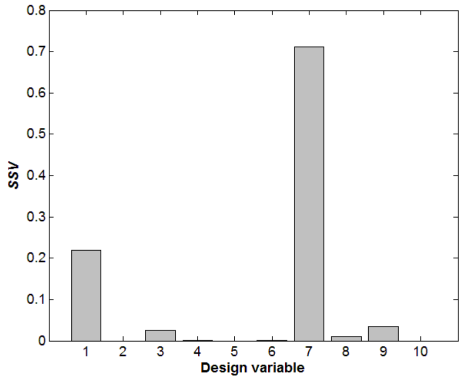

Figure 8. First sensitivity analysis at $\boldsymbol{x}^{L,0}$; 10-bar truss, load case 2

Tables 1 and 2 provide optimum results reported in the literature compared to that of the present work. As indicated, the proposed method leads to solutions with no constraints

Table 1: Comparing optimal designs for the 10-bar planar truss, load case 1

		Optimal cross sectional area (in.^{2})				cross sectional area (in.^{2})		
Variable group	Bar areas	Camp et al.[1] [21]	Li et al.[2] [22]	Lee and Geem[3] [23]	Sonmez[4] [14]	Current work $\boldsymbol{x}^{s=3}$	Current work $\boldsymbol{x}^{L,0}$	Current work *Optimal* $\boldsymbol{x}^{G}$
1	A_1	28.92	30.704	30.150	30.548	80.824	12.087	*31.1650*
2	A_2	0.1	0.100	0.102	0.100	70.280	0.1	*0.1000*
3	A_3	24.07	23.167	22.710	23.180	35.144	16.814	*23.1000*
4	A_4	13.96	15.183	15.270	15.218	81.304	30.242	*14.7230*
5	A_5	0.1	0.100	0.102	0.100	52.374	9.168	*0.1000*
6	A_6	0.56	0.551	0.544	0.551	36.692	0.1	*0.4139*
7	A_7	7.69	7.460	7.541	7.463	42.877	36.330	*7.5712*
8	A_8	21.95	20.978	21.560	21.058	57.974	56.010	*21.1630*
9	A_9	18.7	21.508	21.450	21.501	93.570	93.570	*21.4230*
10	A_{10}	0.1	0.100	0.100	0.100	103.590	3.431	*0.1000*
Weight (lb.)		5076.31	5060.92	5057.88	5060.880	28011	12106	*5064.4*
No. of analyses		NA	75,000	400,000	500,000	3	80	650
Constraints Violation		3.4×10^{-2}	3.1×10^{-4}	8.23×10^{-3}	None	None	None	None

Note: 1 in.^{2}=6.452 cm^2; 1 lb. =4.45 N.
1- Ant colony optimization
2- Heuristic particle swarm optimization
3- Harmony search algorithm
4- Artificial bee colony

Table 2: Comparing optimal designs for the 10-bar planar truss, load case 2

		Optimal cross sectional area (in.^{2})				cross sectional area (in.^{2})		
Variable group	Bar areas	Rizzi[1] [24]	Li et al.[2] [22]	Lee and Geem[3] [23]	Sonmez[4] [14]	Current work $\boldsymbol{x}^{S=3}$	$\boldsymbol{x}^{L,0}$	*Optimal* $\boldsymbol{x}^{G}$
1	A_1	24.29	23.353	23.250	23.4692	84.927	11.267	*25.0000*
2	A_2	0.1	0.100	0.102	0.1005	20.092	0.1	*0.1000*
3	A_3	23.35	25.502	25.730	25.2393	64.859	38.380	*25.0000*
4	A_4	13.66	14.250	14.510	14.3540	96.899	70.420	*14.4320*
5	A_5	0.1	0.100	0.100	0.1001	14.332	0.1	*0.1000*
6	A_6	1.969	1.972	1.977	1.9701	74.368	8.8924	*1.9876*
7	A_7	12.67	12.363	12.210	12.4128	25.497	12.017	*12.4250*
8	A_8	12.54	12.894	12.610	12.8925	22.173	19.285	*12.3590*
9	A_9	21.97	20.356	20.360	20.3343	22.642	22.642	*19.9820*
10	A_{10}	0.1	0.101	0.100	0.1000	55.643	0.1	*0.1000*
Weight (lb.)		4691.84	4677.29	4668.810	4677.077	19210	7401.3	*4682.476*
No. of analyses		NA	75,000	400,000	500,000	3	98	1000
Constraints Violation		1.9×10^{-4}	6.17×10^{-4}	6.38×10^{-2}	None	None	None	None

Note: 1 in.^{2}=6.452 cm^2; 1 lb. =4.45 N.
1- Optimality criteria
2- Heuristic particle swarm optimization
3- Harmony search algorithm
4- Artificial bee colony

violation and less computational efforts. For case 1, the 650 FEM analyses were needed with a minimum weight of 5064.4 lb. For case 2, after 1000 FEM analyses, a minimum truss weight of 4678.1 lb. was found.

Figs. 9 and 10 show convergence histories on the weight minimization of the truss structure in different stages of the optimization procedure, together with the maximum number of iterations normally allowed.

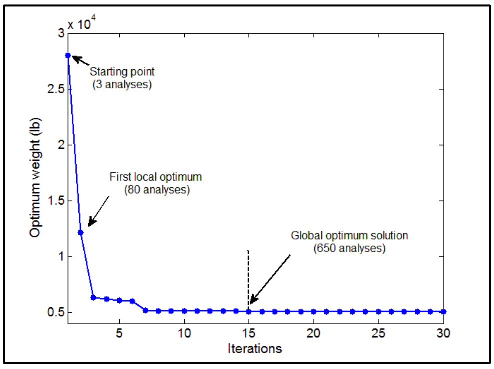

Figure 9. Convergence history of optimum weight; 10-bar planar truss, Load case 1

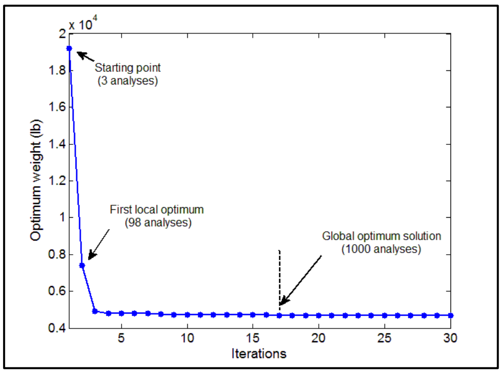

Figure 10. Convergence history of optimum weight; 10-bar planar truss, Load case 2

8.2. Twenty-Five-bar space truss

Fig. 11 shows a 25-bar transmitting tower space truss. This problem has been studied by many researchers including Li et al. [22], Lee and Geem [23], Sonmez [14] and Kaveh et al. [13]. The material density was 0.1 lb./in.^{3} (2767.990 kg/m^3) and the modulus of elasticity was 10,000 ksi (68.950 GPa). This space truss was subjected to two loading conditions as shown in Table 3.

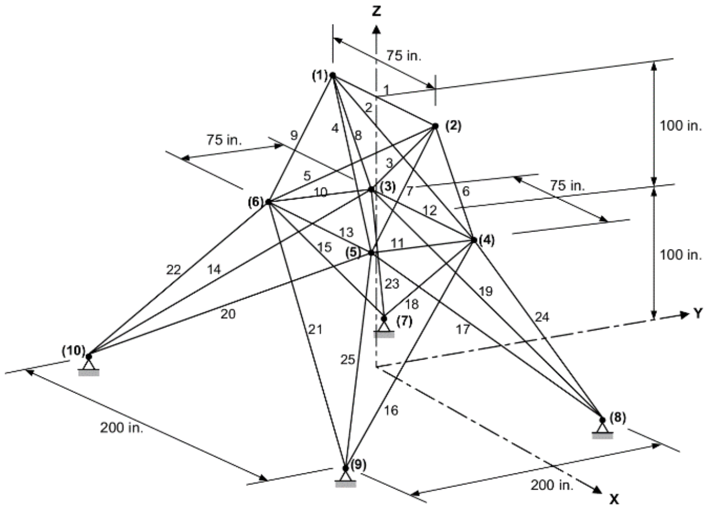

Figure 11. 25-bar space truss

Table 3: Loading conditions (kips) for 25-bar space truss

Loading cases	Node	x	y	z
1	1	1.0	10.0	-5.0
	2	0	10.0	-5.0
	3	0.5	0	0
2	5	0	20.0	-5.0
	6	0	-20.0	-5.0

Note: 1 kips= 4.45 kN.

Table 4: Allowable stresses (ksi) for the 25-bar space truss

Design variables	Members	Compression	Tension
1	A_1	35.092	40.000
2	A_{2-5}	11.590	40.000
3	A_{6-9}	17.305	40.000
4	A_{10-11}	35.092	40.000
5	A_{12-13}	35.092	40.000
6	A_{14-17}	6.759	40.000
7	A_{18-21}	6.959	40.000
8	A_{22-25}	11.082	40.000

Note: 1 ksi= 6.89725 MPa.

Design constraints were set as the maximum allowable displacement of ±0.35 in. (±8.89 mm) imposed to all nodes in every direction and the allowable stresses for all members are given as in Table 4. The minimum cross-sectional areas of all members were set equally as 0.01 in^2 (6.45 mm^2). As for consistency with the literature, all members were classified into eight groups as given in Table 4.

As the bar graph of Fig. 12 shows, the initial random designs were generated up to the second subspace ($s=2$), the second subspace offered solution without penalization. Therefore, the second random design $\boldsymbol{x}^{s=2}$, having generated for the cross sectional areas, set between 1 and 10^1 in^2, was selected as the starting design.

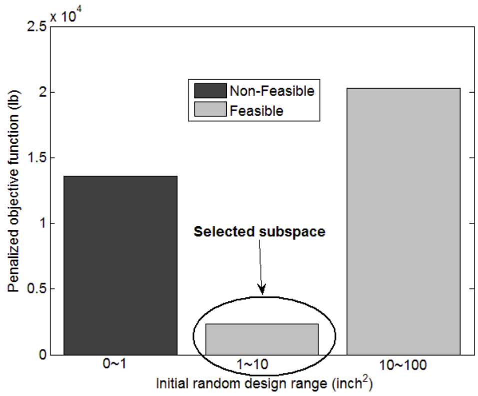

Figure 12. Selecting initial random designs by dividing the searching space into subspaces; the 25-bar truss example

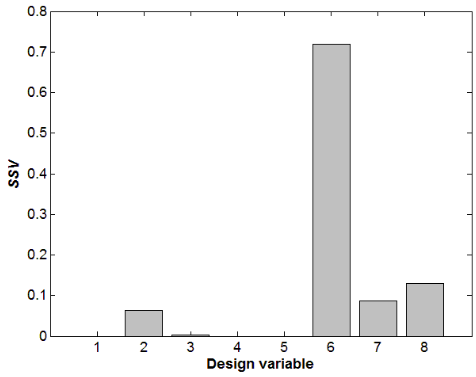

Figure 13. First sensitivity analysis at $\boldsymbol{x}^{L,0}$; 25-bar truss example

Fig. 13 shows the primary sensitivity results at $\boldsymbol{x}^{L,0}$. As illustrated in Table 5, the proposed method required 2340 analyses to find a minimum global weight of 545.14 lb. after 17 iterations. It was approved with the results recorded by other researches.

Table 5: Comparing optimal designs for the 25-bar truss

		Optimal cross sectional area (in.^{2})				cross sectional area (in.^{2})		
						Current work		
Variable group	Bar areas	Li et al.[1] [22]	Lee and Geem[2] [23]	Sonmez[3] [14]	Kaveh et al.[4] [13]	$\boldsymbol{x}^{s=2}$	$\boldsymbol{x}^{L,0}$	*Optimal* $\boldsymbol{x}^G$
1	A_1	0.010	0.047	0.011	0.010	5.3881	0.01	*0.0100*
2	A_{2-5}	1.970	2.022	1.979	1.910	1.2902	0.742	*1.8360*
3	A_{6-9}	3.016	2.950	3.003	2.798	9.2069	8.988	*3.1399*
4	A_{10-11}	0.010	0.010	0.010	0.010	2.3264	0.01	*0.0100*
5	A_{12-13}	0.010	0.014	0.010	0.010	1.3801	0.01	*0.0100*
6	A_{14-17}	0.694	0.688	0.690	0.708	2.7384	0.7319	*0.6809*
7	A_{18-21}	1.681	1.657	1.679	1.836	5.0973	3.090	*1.7539*
8	A_{22-25}	2.643	2.663	2.652	2.645	1.7588	1.758	*2.6068*
Weight (lb.)		545.190	545.380	545.193	545.09	1218.3	794	*545.14*
No. of analyses*		N/A	15,000	300,000	17,500	2	240	2340

Constraints Violation	1.9×10^{-4}	6.17×10^{-4}	6.38×10^{-2}	None	None	None	*None*

Note: 1 in.^{2}=6.452 cm^2; 1 lb. =4.45 N.
1- Heuristic Particle Swarm Optimization
2- Harmony Search algorithm
3- Artificial Bee Colony
4- Chaotic Swarming of Particles

Fig. 14 shows the convergence history of the truss problem towards the global optimum weight. The idea here is to show how fast the proposed technique guides the pointer towards the global optimum in early stages.

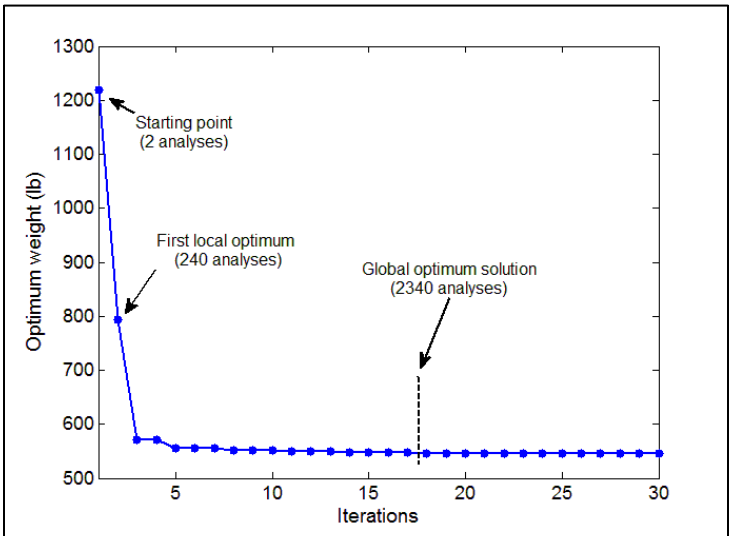

Figure 14. Convergence history of optimum weight; 25-bar truss

8.3 Seventy-two-bar space truss

Fig. 15 shows a 72-bar space truss with its node and element numbering schemes. This space truss, has been recently optimized by Perez et al. [25], Degertekin [26] and Kaveh et al. [13].The problem had a very complex and non-linear search space as it had 320 nonlinear constraints (72 tension, 72 compression, 8 positive displacements, 8 negative displacements for each loading case). The material density and modulus of elasticity were considered 0.1 lb./in.^{3} (2767.990 kg/m^3) and 10,000 ksi (68.950 GPa), respectively. The constraints contained the stress of ±25 ksi (±172.375 MPa) and displacement limitations of ±0.25 in. for uppermost nodes in all directions. This space truss was subjected to two loading conditions: Condition 1, in which $P_X=5.0$ kips, $P_Y=5.0$ kips (22.25 kN), and $P_Z=-5.0$ kips (22.25 kN) on node 17; and Condition 2, that $P_X=0.0$ kips, $P_Y=0.0$ kips, and $P_Z=-5.0$ kips (-22.25 kN) were acted on nodes 17, 18, 19, and 20. For design and manufacturing considerations truss members were classified into sixteen groups as shown in result Tables 6. For this problem, the value of lower bound limitation of design variables was 0.1 in.^{2}.

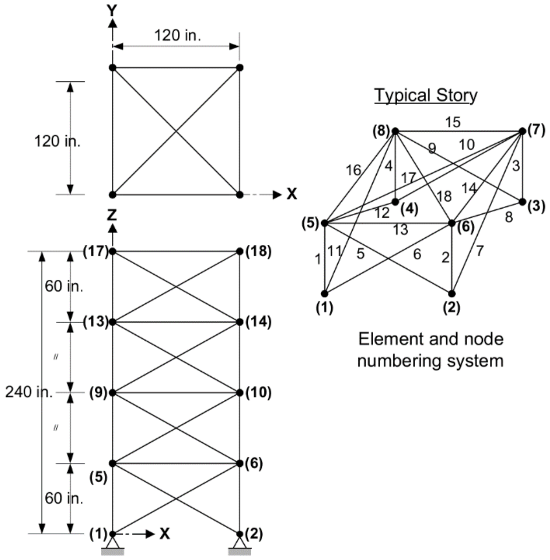

Figure 15. 72-bar space truss

Table 6: Optimal design comparison for the 72-bar space truss under multiple loading conditions

Variable group	Members	cross sectional area (in.^{2}) Perez et al.[1] [25]	Degertekin[2] [26]	Kaveh et al.[3] [13]	Current work $\boldsymbol{x}^{s=2}$	$\boldsymbol{x}^{L,0}$	*Optimal* $\boldsymbol{x}^{G}$
1	1~4	1.743	1.860	1.94459	1.955	0.1	*1.65344*
2	5~12	0.518	0.521	0.5026	9.579	3.590	*0.50681*
3	13~16	0.100	0.100	0.10000	8.130	2.140	*0.10000*
4	17~18	0.100	0.100	0.10000	6.786	0.1	*0.10000*
5	19~22	1.308	1.271	1.26757	3.361	2.762	*1.14299*
6	23~30	0.519	0.509	0.50990	2.877	2.278	*0.57423*
7	31~34	0.100	0.100	0.10000	10.345	0.1	*0.10000*
8	35~36	0.100	0.100	0.10000	2.306	0.1	*0.10000*
9	37~40	0.514	0.485	0.50674	8.478	2.488	*0.34987*
10	41~48	0.546	0.501	0.51651	8.913	5.319	*0.52909*
11	49~52	0.100	0.100	0.10752	4.710	0.1	*0.10000*
12	53,54	0.109	0.100	0.10000	4.816	0.1	*0.10000*
13	55~58	0.161	0.168	0.15618	7.403	0.1	*0.10000*
14	59~66	0.509	0.584	0.54022	5.314	1.720	*0.67830*
15	67~70	0.497	0.433	0.42229	3.954	2.157	*0.26164*
16	71,72	0.562	0.520	0.57941	6.897	3.304	*0.52311*
Weight (lb.)		381.779	380.837	379.974	5381	1854.6	*378.4304*
No. of analyses*		N/A	13,742	10,500	2	246	*6,890*
Constraints Violation		None	1.35×10^{-2}	None	None	None	*None*

Note: 1 in.^{2}=6.452 cm^2; 1 lb.=4.45 N.
* Two structural analysis is carried out for each design due to two loading conditions
1- Particle Swarm Optimization
2- Improved harmony search algorithms
3- Chaotic Swarming of Particles

The convergence history of 72-bar truss weight is given in Fig. 18. In addition, Table 6 contains the results for the same optimization task from different research efforts. As observed, first local optimum point $\boldsymbol{x}^{L,0}$ with the corresponding truss weight 1854.6 lb. was obtained after 246 analyzes. The proposed method showed a good and fast solution convergence while after 6890 analyzes the best solution was found. The best weight obtained by the proposed method is 50 to 66 percent less than the optimum truss weight results by other researchers.

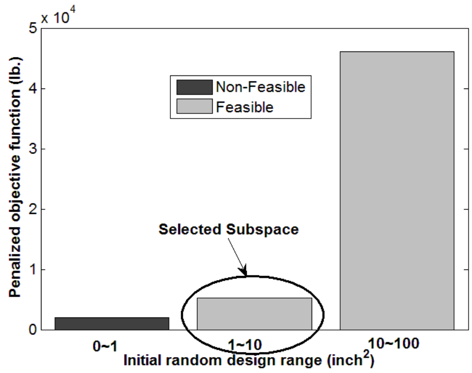

Figure 16. Selecting initial random designs by dividing the searching space into subspaces; the 72-bar truss example

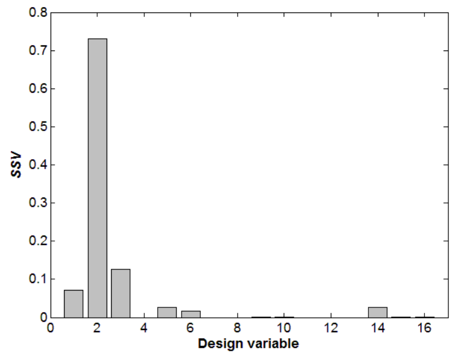

Figure 17. First sensitivity analysis at $\boldsymbol{x}^{L,0}$; 72-bar truss example

As in Table 6, results by Degertekin [26] explore certain amounts of penalty value; however, Perez et al. [25] and Kaveh et al. [13] offered a solution without penalization.

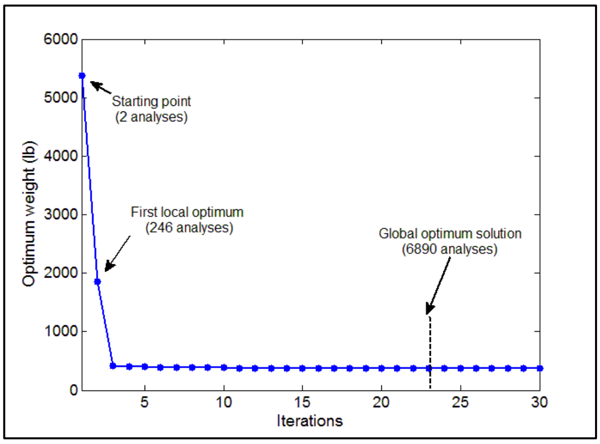

Figure 18. Convergence History of best weight of 72-bar space truss

9. CONCLUSIONS

In this article, a new and relatively fast optimization method based on a marginal feasibility search method for size optimization of truss problems has been introduced. It hybridizes four operators towards the optimum solution. First, it uses Subspace Search (SS) operator to reduce and confine the searching space by employing few arbitrary designs. Having found the target zone where the optimum point lies, it uses in order, a Marginal Sensitivity Analysis (MSA), Marginal Search (MS) and roulette wheel (RW) operators to find the global optimum solution.

Having applied the method into some challenging size optimization of trusses, a major advancement of the technique was observed on the optimum solution in terms number of analyses required for optimization where a minor improvement on the accuracy of the results were also observed in some cases. Investigations on the convergence history and results obtained emphasize on the performance of the proposed method to offer the feasible global optimum design point without multiple runs.

REFERENCES

1. Rahami H, Kaveh A, Gholipour Y. Sizing, geometry and topology optimization of trusses via force method and genetic algorithm, *Engineering Structures*, No. 9, **30**(2008) 2360-9.
2. Ghasemi MR, Hinton E, Wood RD. Optimization of trusses using genetic algorithms for discrete and continuous variables, *Engineering Computations*, No. 3, **16**(1999) 272-303. doi:10.1108/02644409910266403.
3. Dominguez A, Stiharu I, Sedaghati R. Practical design optimization of truss structures using the genetic algorithms, *Research in Engineering Design*, No. 2, **17**(2006), 73-84.
4. Toğan Vedat, Ayşe T. Daloğlu. Optimization of 3d trusses with adaptive approach in genetic algorithms, *Engineering Structures*, No. 7, **28**(2006) 1019-27. http://dx.doi.org/10.1016/j.engstruct.2005.11.007.
5. Camp Ch, Pezeshk Sh, Cao G. Optimized design of two-dimensional structures using a genetic algorithm, *Journal of Structural Engineering*, No. 5, **124**(1998) 551-9.
6. Hasançebi O, Erbatur F. On efficient use of simulated annealing in complex structural optimization problems, *Acta Mechanica*, Nos. 1-4, **157**(2002) 27-50. 10.1007/BF01182153
7. Hadidi A, Kaveh A, Farahmand Azar B, Talatahari S, Farahmandpour C. An efficient hybrid algorithm based on particle swarm and simulated annealing for optimal design of space trusses, *International Journal of Optimization in Civil Engineering*, No. 3, **1**(2011) 377-96.
8. Kaveh, A. *Advances in metaheuristic algorithms for optimal design of structures*, Springer, Switzerland, 2014.
9. Kaveh A, Talatahari S. A discrete particle swarm ant colony optimization for design of steel frames, *Asian Journal of Civil Engineering*, No. 6, **9**(2008) 531-542.
10. Ghasemi MR, Dizangian B. Size, shape and topology optimization of composite steel box girders using pso method, *Asian Journal Of Civil Engineering*, No. 6, **11**(2010) 699-715.
11. Kaveh A, Talatahari S. A hybrid particle swarm and ant colony optimization for design of truss structures, *Asian Journal of Civil Engineering*, No. 4, **9**(2008) 325-44.

12. Camp C, Bichon B. Design of Space Trusses Using Ant Colony Optimization, *Journal of Structural Engineering*, No. 5, **130(**2004**)** 741-51. doi: 10.1061/(ASCE)0733-9445(2004)130:5(741).
13. Kaveh A, Sheikholeslami R, Talatahari S, Keshvari-Ilkhichi M. Chaotic swarming of particles: A new method for size optimization of truss structures, *Advances in Engineering Software*, No. 0, **67**(2014) 136-47. http://dx.doi.org/10.1016/j.advengsoft.2013.09.006.
14. Sonmez M. Artificial bee colony algorithm for optimization of truss structures, *Applied Soft Computing*, No. 2, **11**(2011) 2406-18. http://dx.doi.org/10.1016/j.asoc.2010.09.003.
15. Gandomi AH, Xin-She Y, Alavi AH. Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems, *Engineering with Computers*, No. 1, **29(**2013**)**, 17-35. 10.1007/s00366-011-0241-y.
16. Belegundu AD, Chandrupatla TR. Optimization concepts and applications in engineering, *Cambridge University Press*, 2011.
17. Richards M, Ventura D. Choosing a starting configuration for particle swarm optimization, *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, **3**(2004) pp. 2309-2312.
18. Clerc M, Kennedy J. The particle swarm - explosion, stability, and convergence in a multidimensional complex space, *Evolutionary Computation, IEEE Transactions on*, No. 1, **6**(2002), 58-73. 10.1109/4235.985692.
19. Dorigo M, Thomas S. The ant colony optimization metaheuristic: algorithms, applications, and advances. In *Handbook of Metaheuristics*, edited by F. Glover and G. Kochenberger, Springer, US, 2003.
20. Venkayya VB. Design of optimum structures, *Computers and Structures*, No. 2, **1**(1971) 265-309.
21. Camp C, Bichon B. Design of space trusses using ant colony optimization, *Journal of Structural Engineering*, No. 5, **130**(2004) 741-51. doi: 10.1061/(ASCE)0733-9445(2004)130:5(741).
22. Li LJ, Huang ZB, Liu F, Wu QH. A heuristic particle swarm optimizer for optimization of pin connected structures, *Computers and Structures*, No. 7-8, **85**(2007) 340-9. http://dx.doi.org/10.1016/j.compstruc.2006.11.020.
23. Lee Kang Seok, Zong Woo Geem. A new structural optimization method based on the harmony search algorithm, *Computers and Structures*, No. 9-10, **82**(2004), 781-98. http://dx.doi.org/10.1016/j.compstruc.2004.01.002.
24. Rizzi P. Optimization of multiconstrained structures based on optimality criteria. In *AIAA/ASME/SAE 17th Structures, Structural Dynamics, and Materials Conference*, King of Prussia, PA, 1976.
25. Perez RE, Behdinan K. Particle swarm approach for structural design optimization, *Computers and Structures*, No. 19-20, **85**(2007) 1579-88. http://dx.doi.org/10.1016/j.compstruc.2006.10.013.
26. Degertekin SO. Improved harmony search algorithms for sizing optimization of truss structures, *Computers and Structures*, **92–93**(2012) 229-41. http://dx.doi.org/10.1016/j.compstruc.2011.10.022.