A Novel Hybrid Simplex-Genetic Algorithm For The Optimum Design Of Truss Structures

M. Aslani*, R. Najian Asl[a], R. Oftadeh[b], M. Shariat Panahi[c]

***Abstract*— Optimum design of complex truss structures often requires searching a high-dimensional, heavily constrained solution space. For the very same reason, the problem has been established as a standard benchmark for evaluating the effectiveness of multivariate optimization algorithms. Moreover, the popularity of such structures justifies the development, or at least custom-tuning, of numerical search algorithms tailored to determine the optimum topology/ dimensions of large truss structures at minimum computational cost.**

We present a new hybrid algorithm for the weight minimization of large truss structures. The algorithm combines the fundamental elements of standard Genetic Algorithms with those proposed by Nelder and Mead in their Simplex algorithm. This would result in a search tool that inherits the power of GAs to quickly spot the promising regions of the search space and the ability of Simplex to effortlessly approach the optimum in convex subspaces. Furthermore, we improve the performance of the algorithm by incorporating a modified Tournament Selection and augment the resulting algorithm with a dynamic penalty method to continuously confine the search to the borders of the feasible region.

To demonstrate the applicability and effectiveness of the proposed algorithm, we apply it to a variety of structural design problems and the results are compared with those reported in literature.

***Keywords*—Multivariate Optimization, Hybrid Simplex-Genetic Algorithm, Truss Structures**

I. INTRODUCTION

Optimal design of structures dates back to the times of such great scientists as Galileo, Bernoulli, Lagrange, and Navier who studied basic methods to find the "best" shapes for structural elements to stand given loads while satisfying certain strength requirements [1, 2]. Since then, the topic has grown to become a challenging one, generally known as Structural Optimization, which seeks to determine the structure's topology/dimensions/materials that would render the structure as light/inexpensive as possible while keeping its performance characteristics (e.g. stresses and displacements) within allowable limits.

The large number of design variable and their mixed (discrete/continuous) nature has rendered the problem a perfect benchmark for large scale search algorithms. The emergence of Evolutionary Algorithms, with their ability to simultaneously examine numerous areas of vast, multimodal search spaces resulted in more reliable, noticeably less expensive designs.

One of the most promising and popular evolutionary algorithms is Genetic Algorithms (GAs) which was inspired by the Darwinian principle of *Survival of the Fittest*. Using a number of nature-like genetic operators and multiple stochastic decision parameters, GAs basically make a population of artificial "chromosomes" (each representing a potential solution) to "evolve" toward global/near global solution(s).

The Simplex algorithm, basically an unconstrained numerical optimization algorithm, on the other hand builds on the presumption that moving the worst member of a set of randomly selected points towards the centroid of the other members and beyond that would improve the point and continuing in this way would lead the centroid of the set to the optimum provided that the search region is convex.

The idea of applying GAs to the optimal design of structures was first implemented by Goldberg and Samtani [3]. They considered the use of GAs to optimize a 10-bar planar truss. Other researchers have also applied the technique to the design of plane frames [5], welded beams [4]. Pham and Yang [6] presented interesting work on the optimization of multi-modal discrete functions with GAs and used truss structures as an example. GÄatzi[7] combined GAs and Ansys™ and applied the combination to some structural design problems.

II. THE BASIC GENETIC ALGORITHMS

Before describing our improved version, we need to briefly introduce the basic elements of standard GAs.

✓ Binary Coded Genetic Algorithms

In a binary coded GA each chromosome presents a possible solution for the problem, operators act on binary coded solutions as what happens in nature.

* M.Aslani, B.S. Student, School of *Mechanical Engineering, College of Engineering, University of Tehran, Tehran, Iran* (**Corresponding author** to provide phone: 00989123684934; e-mail: m.aslani@ut.ac.ir).

a- R.Najian Asl, B.S. Student, School of *Mechanical Engineering, College of Engineering, University of Tehran.*(e-mail: najianaslreza@gmail.com).

b- R.Oftadeh, M.S Student, School of *Mechanical Engineering, College of Engineering, University of Tehran.*(e-mail: roftadeh@ut.ac.ir).

c- M.Shariat Panahi, Associate Professor, School of *Mechanical Engineering, College of Engineering, University of Tehran.*(e-mail: mshariatp@ut.ac.ir).

Each member in a binary coded GA is a bit string. This bit string called chromosome, presents an available solution for the problem. Genetic operators are applied on these strings as what happens in nature. For simple GA crossover and mutation are basic operators.
Here are some of its fundamental operators:

✓ **Crossover**

Crossover is the most important operator in GA. In crossover the couples chosen from mating pool are merged with each other to make children who get their features from their parents but not a new one.
A very simple single point cross over scheme is as in figure [1].

✓ **Mutation**

Another main operator in GA is mutation. Upon its low probability to happen mutation enables us to find new search areas. In a binary coded GA, mutation if happens changes a bit of a gen from 0 to 1 or vice versa.

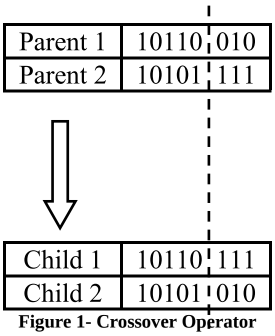

Figure 1- Crossover Operator

Parent	1**0**110**1**01
Child	1**1**110**0**01

Figure 2- Mutation Operator

✓ **Shifting Constrained Problem to Unconstrained**

A constrained problem is one in which the feasible region is defined by a set of implicit/explicit constraints is limited to some boundaries. It means that the validity of a solution is confined to its constraints defined for the problem. As an example to optimize weight of a truss we are searching for the lowest cross section of members and the constraints are stress and displacement of members. Here stress and displacement are indirect constraints as they are not applied directly on the cross section of the members i.e. the variables of the fitness function.
To shift the problem to a normal form we easily applied penalty to the solution which violates the constraints, so automatically the fitness of these chromosomes are lowered.
Many penalty methods are introduced and suggested for different problems. Here we have applied dynamic penalty as follow:
Obj. Function: $f(\vec{x})$
Constrained to: $g_i(\vec{x}) \leq 0\,,\ \ i = 1, q$
$F(\vec{x}) = f(\vec{x}) \pm p(\vec{x})$

$$p(\vec{x}) = \alpha(\#\ \text{iteration})^{\beta} \times \sum_{i=1}^{q} S_i(\vec{x}) \qquad Eq.(1)$$

$$S_i(\vec{x}) = \begin{cases} 0 & g_i(\vec{x}) \leq 0 \\ |g_i(\vec{x})| & \text{otherwise} \end{cases}$$

The sensitivity of the problem to this penalty method and its parameters are discussed further.
Dynamic penalty enables us to have a smooth rate of reaching to the global optimum. A wise tuning of the penalty parameters will tend the algorithm find the global optimum and preferably searches areas near to the boundaries of constraints where the validity of the solution is acceptable and the global point also lives there.

✓ **Tournament Selection**

A variety of methods has been introduced for reproduction in GA. Goldberg introduced the application of roulette wheel as the most famous tool for this operator. A very easy and more efficient way to choose parents is to randomly choose 2 chromosomes and comparing their fitness. The chromosome with better fitness will be the candidate for being a parent. This method is called Tournament Selection.

III. THE HYBRID SIMPLEX-GENETIC ALGORITHM

As mentioned earlier, the main idea behind the Simplex algorithm is to scroll a given area to find the best point. Since the problem has changed to an unconstrained one, we can use the Simplex method to find better sub population.
In a Simplex algorithm with n variabes $n+1$ points are given to search the area. So to have better chromosome in a modified GA we first chose a best bunch of chromosomes, the number of them was dependant on the case. For example for a problem with 8 variables we should choose 9 chromosomes. Then by running the Simplex algorithm among these points at least a local optimum is achieved. Not being constrained to use only a set of points, we applied Simplex to more set of groups to upper the chance of finding the best points. To control this combination of algorithms, a parameter called *Simplex Covering (SC)* were defined to experimentally find the best number of groups to get into the Simplex algorithm. Comparison of the results shows an efficient convergence time, even when the SC is high. This is because what happens behind the Simplex is so simple but effective that does not affect the solving time but increases the chance to find the global optimum. This hybrid method is called *Hybrid Simplex-Genetic Algorithm* (HSGA).

IV. NUMERICAL EXAMPLES

Standard test problems that have been studied in related literature are presented here to demonstrate the efficiency of the HSGA. These cases include a 10-bar planar and a 25-bar space truss, subjected to a single load condition and a 72-bar space truss subjected to two load conditions. These truss structures were analyzed using the FEM displacement method. For all these cases the HSGA general solving parameters were as in table [1]. Table [2] presents the general parameters of the problems, the first one is a 2D truss the others are 3D.

Table 1-General Parameters of HSGA

Algorithm Parameters	10-bar	25-bar	72-bar
Population Size	200	100	300
CrossOver Rate	0.75	0.85	1
Mutation Probability	0.005	0.005	0.005
Simplex Covering(SC)	5	3	10
Penalty(α-β) Eq.1	100-2	10-1	50-1

Table 2- Parameters of Structure Problems

Problem's Parameters	10-bar (2D)	25-bar (3D)	72-bar (3D)
Material Density($\frac{lbf}{in^3}$)	0.1	0.1	0.1
Modulus of Elasticity(ksi)	10,000	10,000	10,000
Stress Limitation(ksi)	±25	±40	±25
Displacement Limitation(in)	±2.0	±0.35	±0.25
Number of Variables	10	8	16
Number of Constraints	22	55	264

A. 10-BAR PLANAR TRUSS

The cantilever truss, shown in Fig[3], was previously analyzed using various mathematical methods by Schmit and Farshi [8], Schmit and Miura [9], Venkayya [10], Gellatly and Berke [11], Dobbs and Nelson [12], Rizzi [13], Khan and Willmert [14], John et al. [15], Sunar and Belegundu [16], Stander [17], Xu and Grandhi [18],and Lamberti and Pappalettere [19,20] Kang Seok Lee, Zong Woo Geem [21].

In this example, P_1 and P_2 were 150 and 50 lbs respectively, see Fig [3] for geometry of the forces. Table [6] give the best solution vector for this case, and also show a comparison between the optimal design results reported in the literature and the present work.

The best solutions were approached by only 2 minutes on a regular computer. As it is shown in Fig [4] these results were given in only 400 hundred iterations which show high efficiency of the HSGA.

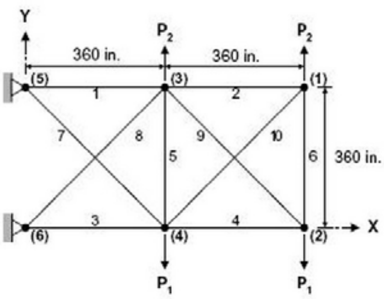

Figure 3- 10 bar planar truss

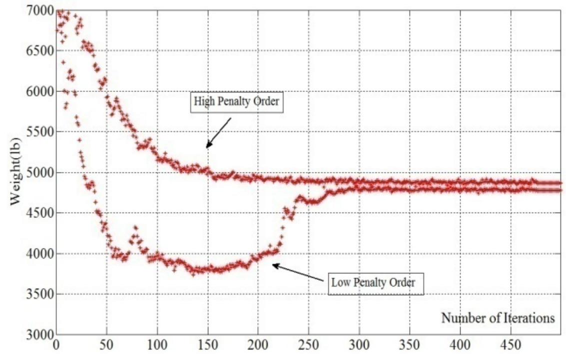

Figure 4 -Convergence History for Two Different Penalty Parameters

B. 25-BAR SPACE TRUSS

The second example considers the weight minimization of a 25-bar transmission tower as shown on Fig [5]. The design variables are the cross-sectional area for the truss members, which are linked in eight member groups as shown in Table [4]. Loading of the structure is presented on Table [3].

Table 3- Loading of the 25-bar Space Truss

Node	F_x(kips.)	F_x(kips.)	F_x(kips.)
1	1	-10	-10
2	0	-10	-10
3	0.5	0	0
6	0.6	0	0

Fig [6] shows how the algorithm converges to the best answer. The effects of penalty parameters are also shown in the figure.

Table 4-Group cross Section of 25-bar Space Truss

Group Number	Members of Truss
1	1-2
2	1-4,2-3,1-5,2-6
3	2-5,2-4,1-3,1-6
4	3-6, 4-5
5	3-4, 5-6
6	3-10, 6-7,4-9, 5-8
7	3-8, 4-7,6-9,5-10
8	3-7,4-8,5-9,6-10

These results were captured with 200 to 300 hundred of iterations as it's obvious from the figure that the fitness function doesn't change after this number of iteration. We can see other's work on the internet with a better fitness function but unfortunately most of them handle values which violate the constraints.

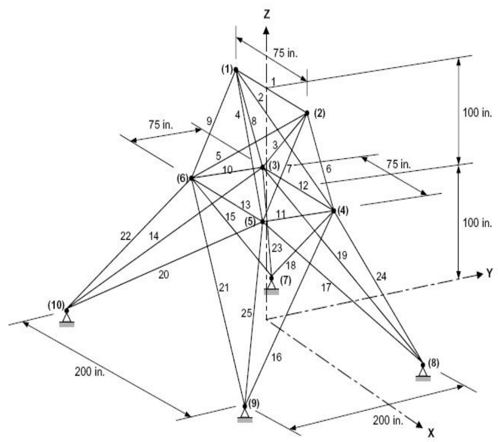

Figure 5- 25-bar Space Truss

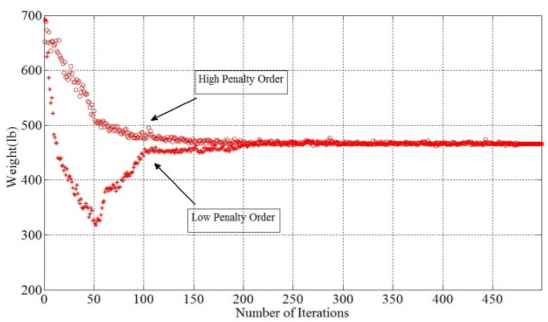

Figure 6- Convergence History for two Different Penalty Parameters

C. 72- BAR SPACE TRUSS

The 72-bar space truss, shown in Fig [7], has also been optimized by many researchers, including Schmit and Farshi [22], Schmit and Miura [23], Venkayya [24], Gellatly and Berke [25], Kang Seok Lee, Zong Woo Geem [21].
This space truss was subjected to the following two loading conditions:

Table 5- Loading Condition of 72-bar Space Truss

	Nodes	F_x(kips.)	F_x(kips.)	F_x(kips.)
Condition 1	1	5	5	-5
	2	0	0	0
	3	0	0	0
Condition 2	1	0	0	-5
	2	0	0	-5
	3	0	0	-5
	4	0	0	-5

The minimum cross-sectional area of 0.01 in^2 was considered in the solution.
Table [9] shows the cross section grouping of the members.

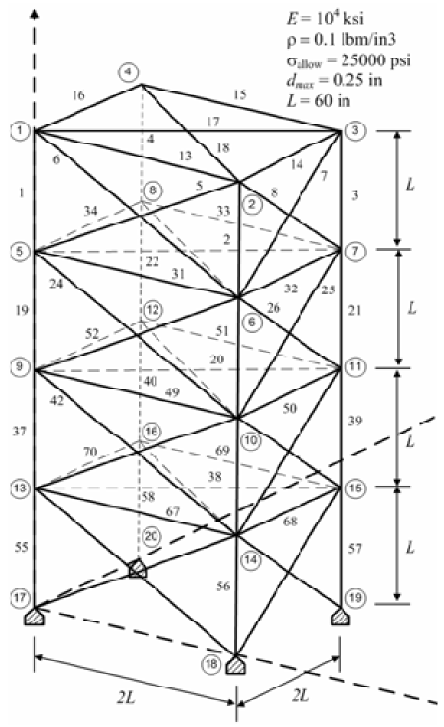

Figure 7- 72-bar Space Truss

For this case the best answer and the comparison of the results are available in table [8] .As you can see in Fig [8] the robustness of HSMA is obvious and such results achieved only by 500 iterations which took 3 minutes on a regular computer.

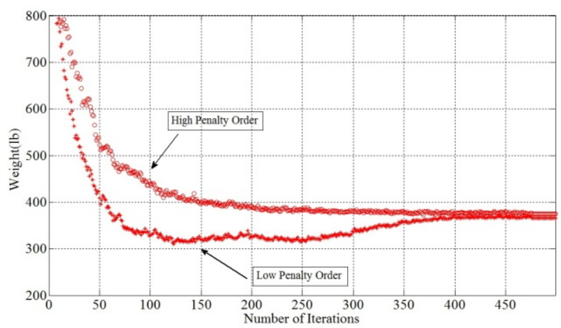

Figure 8- Convergence History for two different Penalty Parameters

V. SUMMARY AND CONCLUSIONS

A new hybrid search/optimization algorithm, called Hybrid Simplex-Genetic Algorithms (HSGA) was presented. The algorithm combines the exploration power of standard GAs and the exploitation capacity of the Nelder-Mead (Simplex) algorithm. The resulting algorithm has shown to benefit from the power of GAs to quickly spot the promising regions of the search space and the ability of Simplex to efficiently approach the optimum in convex subspaces. The performance of the algorithm was further improved by incorporating a modified Tournament Selection and augmenting it with a dynamic penalty method to continuously confine the search to the borders of the feasible region(s.)

Table 6-Comparison of optimal design for 10-bar space truss

Variables (in^2)		Schmit & Farshi	Schmit & Miura	Venkayy	Dobbs & Nelson	Rizzi	Khan & Willmert	Johnetal	HSGA (Thiswork)
1	A_1	24.29	23.55	25.19	25.81	23.53	24.72	23.59	24.8210
2	A_2	0.100	0.100	0.363	0.100	0.100	0.100	0.10	0.0100
3	A_3	23.35	25.29	25.42	27.23	25.29	26.54	25.25	24.7847
4	A_4	13.66	14.36	14.33	16.65	14.37	13.22	14.37	12.9048
5	A_5	0.100	0.100	0.417	0.100	0.100	0.108	0.10	0.0422
6	A_6	1.969	1.970	3.144	2.024	1.970	04.835	1.97	2.0719
7	A_7	12.67	12.39	12.08	12.78	12.39	12.66	12.39	13.4122
8	A_8	12.54	12.81	14.61	14.22	12.83	13.78	12.80	13.9488
9	A_9	21.97	20.34	20.26	22.14	20.33	18.44	20.37	18.5669
10	A_{10}	0.100	0.100	0.513	0.100	0.100	0.100	0.10	0.0100
Weight(lb)		**4691.84**	**4676.96**	**4684.11**	**4895.6**	**5059.7**	**4676.92**	**4676.93**	**4665.6**

Table 7-Comparison of optimal design for 25-bar space truss

Variables (in^2)		Zhu	Rizz	Schmit	Rajeev	Tulleo(GA)	HSGA (This work)
1	A_1	0.1	0.01	0.01	0.1	0.1	0.01
2	A_2	1.9	1.988	1.964	1.8	0.7	0.0635
3	A_3	2.6	2.991	3.033	2.3	3.2	3.6641
4	A_4	0.1	0.01	0.01	0.2	0.1	0.01
5	A_5	0.1	0.01	0.01	0.1	1.4	1.9614
6	A_6	0.8	0.684	0.67	0.8	1.1	0.7430
7	A_7	2.1	1.676	1.68	1.8	0.5	0.1650
8	A_8	2.6	2.662	2.67	3	3.4	3.9297
Weight(lb)		**562.93**	**545.16**	**545.22**	**546.01**	**493.94**	**464.872**

Table 8- Comparison of optimal design for 72-bar space truss

Variables (in^2)		Adeli & Park	Sarma & Adeli		Lee & Geem	HSGA (This work)
			Simple GA	Fuzzy GA		
1	$A_1 \approx A_4$	2.755	2.141	1.732	1.963	1.8226
2	$A_5 \approx A_{12}$	0.510	0.510	0.522	0.481	0.5409
3	$A_{13} \approx A_{16}$	0.010	0.054	0.010	0.010	0.0105
4	$A_{17} \approx A_{18}$	0.010	0.010	0.013	0.011	0.0117
5	$A_{19} \approx A_{22}$	1.370	1.489	1.345	1.233	1.1040
6	$A_{23} \approx A_{30}$	0.507	0.551	0.551	0.506	0.5119
7	$A_{31} \approx A_{34}$	0.010	0.057	0.010	0.011	0.0129
8	$A_{35} \approx A_{36}$	0.010	0.013	0.013	0.012	0.0110
9	$A_{37} \approx A_{40}$	0.481	0.565	0.492	0.538	0.5608
10	$A_{41} \approx A_{48}$	0.508	0.527	0.545	0.533	0.5437
11	$A_{49} \approx A_{52}$	0.010	0.010	0.066	0.010	0.0120
12	$A_{53} \approx A_{54}$	0.643	0.066	0.013	0.167	0.1574
13	$A_{55} \approx A_{58}$	0.215	0.174	0.178	0.161	0.1644
14	$A_{59} \approx A_{66}$	0.518	0.425	0.524	0.542	0.5560
15	$A_{67} \approx A_{70}$	0.419	0.437	0.396	0.478	0.4012
16	$A_{71} \approx A_{72}$	0.504	0.641	0.595	0.551	0.5196
Weight(lb)		**376.3**	**372.4**	**364.7**	**364.33**	**363.3821**

ISBN: 978-988-18210-7-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Table 9-Cross Section Grouping of 72-Bar Truss

Area Members Group	Truss Members
A1	1, 2, 3, 4
A2	5, 6, 7, 8, 9, 10, 11, 12
A3	13, 14, 15, 16
A4	17, 18
A5	19, 20, 21, 22
A6	23, 24, 25, 26, 27, 28, 29, 30
A7	31, 32, 33, 34
A8	35, 36
A9	37, 38, 39, 40
A10	41, 42, 43, 44, 45, 46, 47, 48
A11	49, 50, 51, 52
A12	53, 54
A13	55, 56, 57, 58
A14	59, 60, 61, 62, 63, 64, 65, 66
A15	67, 68, 69, 70
A16	71, 72

Through a number of standard structural design examples of various complexities, the proposed algorithm was demonstrated to outperform the standard GA and some hybrid methods reported in literature.

The proposed algorithm can be applied to a wide range of engineering problems with large, highly nonlinear and heavily constrained solution spaces and mixed (discrete-continuous) variables.

REFERENCES

[1] Galileo Galilei. Dialogues Concerning Two New Sciences. Evanston Ill. Northwestern University Press, 1950. Originally published in 1665.

[2] Using Genetic Algorithms for Optimal Design of Trusses. Carlos A. Coello Coello Michael Rudnick Alan D. Christiansen. Department of Computer Science, Tulane University, New Orleans, LA 70118.W. M. Jenkins.

[3] Towards structural optimization via the genetic algorithm. Computers and Structures, 40(5):1321-7, 1991

[4] Kalyanmoy Deb. Optimal design of a welded beam via genetic algorithms. AIAA Journal, 29:2013-15, November 1991.

[5] W. M. Jenkins. Plane frame optimum design environment based on genetic algorithm. Journal of Structural Engineering, 118(11):3103-13, November 1992

[6] D. T. Pham and Y. Yang. Optimization of multimodal discrete functions using genetic algorithms. In Institute of Mechanical Engineers (Part D), pages 53- 9, 1993

[7] Roman GÄatzi, Marion Uebersax, Oliver KÄonig. Structural Optimization Tool using Genetic Algorithms and Ansys, June 2000

[8] Schmit Jr LA, Farshi B. Some approximation concepts for Structural synthesis. AIAA J 1974; 12 (5):692–9.

[9] Schmit Jr LA, Miura H. Approximation concepts for efficient structural synthesis. NASA CR-2552, Washington, DC: NASA; 1976.

[10] Venkayya VB. Design of optimum structures. Comput and Structures 1971;1(1–2):265–309.

[11] Gellatly RA, Berke L. Optimal structural design. AFFDLTR- 70-165, Air Force Flight Dynamics Lab, Wright-Patterson AFB, OH; 1971.

[12] Dobbs MW, Nelson RB. Application of optimality criteria to automated structural design. AIAA J 1976;14(10):1436–43.

[13] Rizzi P. Optimization of multiconstrained structures based on optimality criteria. SAE 17th Structures, Structural Dynamics, and Materials Conference, King of Prussia, PA; 1976.

[14] Khan MR, Willmert KD, Thornton WA. An optimality criterion method for large-scale structures. AIAA J 1979; 17(7):753–61.

[15] John KV, Ramakrishnan CV, Sharma KG. Minimum weight design of truss using improved move limit method of sequential linear programming. Computer and Structures 1987;27(5):583–91.

[16] Sunar M, Belegundu AD. Trust region methods for structural optimization using exact second order sensitivity. Int J Numer Methods Eng 1991; 32:275–93.

[17] Stander N, Snyman JA, Coster JE. On the robustness and efficiency of the S.A.M. algorithm for structural optimization. Int J Numer Methods Eng 1995; 38:119–35.

[18] Xu S, Grandhi RV. Effective two-point function approximation for design optimization. AIAA J 1998;36(12): 2269 75.

[19] Lamberti L, Pappalettere C. Comparison of the numerical efficiency of different sequential linear programming based on algorithms for structural optimization problems. Computer and Structures 2000; 76:713–28.

[20] Lamberti L, Pappalettere C. Move limits definition in structural optimization with sequential linear programming— Part II: Numerical examples. Computer and Structures, 2003; 81:215–38.

[21] Kang Seok Lee, Zong Woo Geem, A new structural optimization method based on the harmony search algorithm Computers and Structures 82 (2004) 781-798.

[22] Schmit Jr LA, Farshi B. Some approximation concepts for structural synthesis. AIAA J 1974; 12(5):692–9.

[23] Schmit Jr LA, Miura H. Approximation concepts for efficient structural synthesis. NASA CR-2552, Washington, DC: NASA; 1976.

[24] Venkayya VB. Design of optimum structures. Computer and Structures 1971; 1(1–2):265–309.

[25] Gellatly RA, Berke L. Optimal structural design. AFFDLTR- 70-165, Air Force Flight Dynamics Lab., Wright- Patterson AFB, OH; 1971.

ISBN: 978-988-18210-7-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)