УДК 514.18

M. Nestorović, PhD dipl. eng. arch.,
J. Milošević, PhDc dipl. eng. arch.,
Z. Šobić, PhDc dipl. eng. arch.

FORM-FINDING USING BI-DIRECTIONAL EVOLUTIONARY STRUCTURAL OPTIMIZATION (BESO) METHOD

Belgrade University

This paper reviews the potentials of application of Bi-directional Evolutionary Structural Optimization (BESO) method in the architectural engineering design. The aim is to point on the advantage of application of BESO method in the phase of design conceptualization. Described example has a function to illustrate the effectiveness of this approach.

Introduction. In architecture, the form-finding projects deal with capacity of the form to translate desires, environmental constraints and technical limits into integrated morphological solutions. Form-finding methods enable generation of design while assuring of material and structural efficiency. Optimal structures are light, economic and of good performances. Observations from the nature are greatly used as foundation to generate form-finding strategies. In that respect, evolutionary-based methods, which apply concept of biological development, are frequently used approach. On the other hand, development of computational technologies facilitate gradual replacement of traditional physical models and limiting manual processes of form-finding and structural optimization. The creative contribution in using form-finding design tools has its historic significance, and new digital technologies offer possibility for development and application of new high reliable, precise and efficient design methods.

1. Basics of BESO method

Bi-directional Structural Optimization (BESO) is digital tool of mathematically based form-finding, introduced in late 1990s [2,4] as an improved version of ESO method [5,6]. ESO method is based on FEA and the simple concept of gradually removing inefficient material (elements) from the structure (FEA model). Through this process, the resulting structure will evolve towards its optimum shape and topology. On the other hand, in the BESO method, a bi-directional evolutionary strategy is applied which allows material (elements) to be removed and added simultaneously. The efficiency of the elements can be calculated by sensitivity analysis of the considered objective function or can be assigned intuitively.

Stiffness is one of the key factors that must be taken into account in the design of architectural structures. Commonly the mean compliance *C*, the inverse measure of the overall stiffness of a structure, is considered. Topology optimization is often aimed at searching for the stiffest structure with a given volume of material. The optimization problem with the volume constraint could be stated as:

Minimize $C=12f^Tu$ (1)

Subject to: $V^* - \sum_{i=1}^{N} V_i x_i = 1$ (2)

$x_i = 0$ or 1

(f - applied load; u - displacement; C - mean compliance; V- volume of an individual element; V^*- prescribed total structural volume; N - total number of element in the system; -
binary design variable, which declares the absence (0) or presence (1) of an element).

Sensitivity number of the elements are based on the elemental strain energy density. When a solid element is removed from the structure, the change of the mean compliance or total strain energy is equal to the elemental strain energy. This change is defined as the sensitivity number:

$$\alpha_i^e = \Delta C_i = \frac{1}{2} u_i^T K_i u_i \quad (4)$$

(u_i - nodal displacement vector of the ith element; K_i - elemental stiffness matrix)

In BESO, the sensitivity numbers used for material removal and addition are modified by introducing mesh-independency filter scheme which smoothes the sensitivity numbers through the design domain. The filter has a length scale r that does not change with mesh refinement. The primary role of the scale parameter $r$in the scheme is to identify the nodes that will influence the sensitivity of the ith element. Usually the value of $r$should be big enough to circumscribe sub-domain Ω (which size does not change with the mesh size) that covers more than one element. Nodes located inside the Ω contribute to the computation of the improved sensitivity number of the ith element as:

$$\alpha_i = \frac{\sum_{j=1}^{K} w(r_{i,j}) \alpha_j^n}{\sum_{j=1}^{K} w(r_{i,j})} \quad (5)$$

(K- total number of nodes in the sub-domain Ω_i; $(\mathbf{r}_{i,j})$ - linear weight factor defined as: $w(\mathbf{r}_{i,j}) = r_{min} - \mathbf{r}_{i,j}$ (j=1,2,…,K).

In BESO the convergence histories of the mean compliance and the structural topology are greatly improved by averaging the sensitivity numbers with their historical information. The simple averaging shame is given as:

$$\alpha_i = \frac{\alpha_i^k + \alpha_i^{k-1}}{2} \qquad (6)$$

(k-current iteration number).

The proposed filter and averaging schemes resolve problems of checkerboard and mesh-dependency.

Since the volume constraint (V^*) can be greater or smaller than the volume of the initial guess design, the target volume in each iteration may decrease or increase step by step until the constraint volume is achieved. The evolution of the volume can be expressed by:

$$V_{k+1} = V_k (1 \pm ER) \quad k=(1,2,3 \ldots) \qquad (7)$$

ER - evolutionary volume ratio.

Once the volume constraint is satisfied, the volume of the structure will be kept constant for the remaining iterations as:

$$V_{k+1} = V^* \qquad (8)$$

The elements are sorted according to the value of their sensitivity numbers (from the highest to the lowest). For solid element (1), it will be removed (switched to 0) if $\alpha_i \leq \alpha_{del}^{th}$. For void elements (0), it will be added (switched to 1) if $\alpha_i > \alpha_{add}^{th}$. α_{del}^{th} and α_{add}^{th} are the threshold sensitivity numbers for removing and adding elements. α_{del}^{th} is always less or equal to α_{add}^{th}.

The cycle of the FEA and element removal/addition continues until the objective volume (V^*) is reached and the following convergence criterion is satisfied:

$$error = \frac{\left|\sum_{i=1}^{N} C_{k-i+1} - \sum_{i=1}^{N} C_{k-N-i+1}\right|}{\sum_{i=1}^{N} C_{k-i+1}} \leq \tau \qquad (9)$$

(k-current iteration number; τ - allowable convergence tolerance; N-integer number).

In the BESO inefficient elements are completely eliminated from the mesh. This technique is referred as *hard kill* method. Oppose to *soft kill* method where the void elements are represented by a very soft material, in *hard kill* method only the non-void elements remain in the mesh and so the FEA can be performed faster. In *soft kill* method sensitivities of void elements are directly calculated; in *hard kill* methods the sensitivities of void elements cannot be calculated directly from the analysis results and should be extrapolated from the surrounding solid elements. [3]

BESO procedure, as described in [3], has following steps: /1/ Discretize the design domain using a finite element mesh and assign initial property values (0 or 1) for the elements to construct an initial design. /2/ Perform FEA and then calculate the element sensitivity number according to Equation (5). /3/ Average the sensitivity number with its history information using Equation (6) and then save the resulted sensitivity number for next iteration. /4/ Determine the target volume for the next iteration using Equation (7). /5/ Add and delete elements according to the described procedure. /6/ Repeat steps 2-5- until the constraint volume V is achieved and the convergence criterion (9) is satisfied.

2. Example

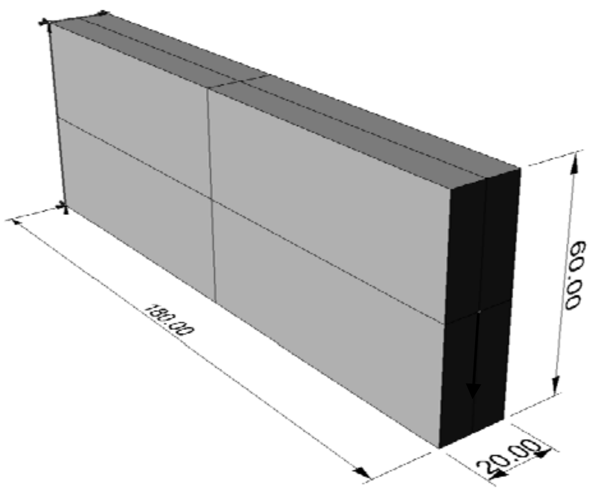

Fig. 1. Design domain and boundary conditions

Given example demonstrates the computational efficiency of BESO. BESO Engine Version 1.3. Beta plug-in for Rhinoceros® 4.0 (developed by Z. Zuo and M. Xie at RMIT University, Australia) was used for computation. The Fig.1. shows the design domain of a 3D cantilever (180L x 20W x 60H cm) with the concentrated load of F=1000N acting at the center of the free end.

The structure was modeled using 8340 eight-node brick elements. Young's modulus E=200GPa and Poisson's ratio v=0.3 are assumed. The objective volume is 10% of the total volume of the design domain. BESO parameters are: ER=2%, AR_{max}=50%, r_{min}=9mm, τ=0.1%.

Fig.2. gives the resulting topologies at various iterations.

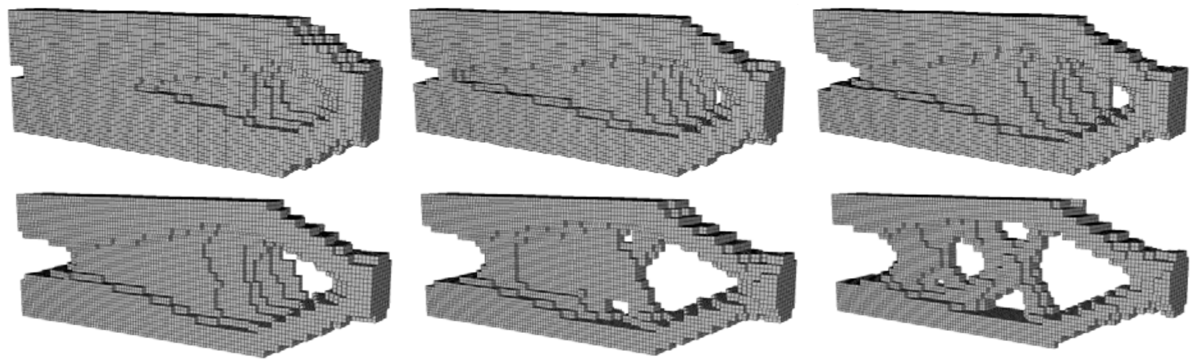

Fig. 2. Evolution of topology: (a) iteration 5; (b) iteration 10; (c) iteration 15; (d) iteration 20; (e) iteration 30; (f) iteration 40

Optimization stopped due to solution converged. The final optimal design is given in the Fig.3.

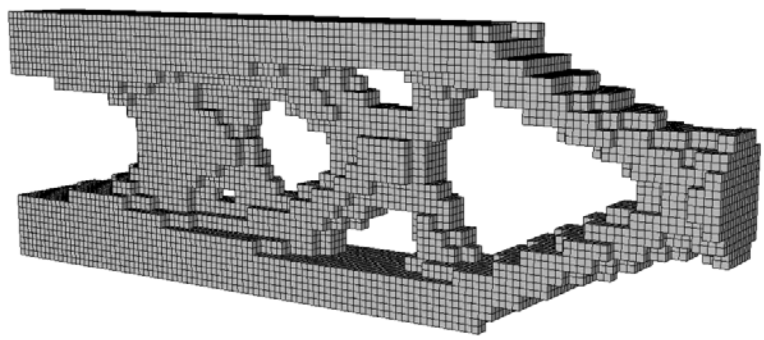

Fig. 3. Optimal design (iteration 44)

Conclusion. Standard for engineering, the derivation of form through performance based analysis and evaluations is still insufficiently exploited in the field of architectural design. The example presented in this paper on application of BESO method to practical design problem demonstrates benefit and potential of using the topology optimization technique as design tool. The BESO technique provides a useful tool for engineers and architects who are interesting in exploring structurally efficient forms and shapes during the conceptual design stage, it enables to expand the possible structural forms of their projects.

However, more research efforts should be directed towards improving its applicability to practical design problems and making the technology easily accessible to practicing architects.

Acknowledgement

Authors are supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia, Project No. TR 36008.

References

1. M. P. Bendsøe, O. Sigmund, (2003). Topology Optimization: Theory, Methods and Applications, Springer, Berlin.
2. O. M. Querin, G. P. Steven, Y. M. Xie, (1998). Evolutionary Structural Optimization (ESO) using bidirectional algorithm. Engineering Computations (1998)15(8): 1031-1048.
3. X. Huang, Y. M. Xie, (2010). Evolutionary Topology Optimization of Continuum Structures: Methods and Applications, John Wiley & Sons, Ltd, UK.
4. X. Y. Yang, Y. M. Xie, G. P. Steven, O. M. Querin, (1999). Bidirectionaly evolutionary method for stiffness optimization. AIAA Journal 37(11): 1483-1488
5. Y. M. Xie, G. P. Steven, (1993). A Simple Evolutionary Procedure for Structural Optimization. Computers & Structures 49(5): 885-896.
6. Y. M. Xie, G. P. Steven, (1997). Evolutionary Structural Optimization, Springer, London.

ПОИСК ФОРМЫ С ИСПОЛЬЗОВАНИЕМ СПОСОБА ДВУНАПРАВЛЕННОЙ ЭВОЛЮЦИОННОЙ СТРУКТУРНОЙ ОПТИМИЗАЦИИ (BESO)

М. Несторович, Е. Милошевич, З.Шобич

В настоящей статье рассматриваются возможности применения способа двунаправленной эволюционной структурной оптимизации (BESO) в архитектурном проектировании. Цель состоит в том, чтобы указать на преимущество применения метода BESO на стадии концептуального проектирования. Описанный пример иллюстрирует эффективность такого подхода.

ПОШУК ФОРМИ З ВИКОРИСТАННЯМ СПОСОБУ ДВОБІЧНОЇ ЕВОЛЮЦІЙНОЇ СТРУКТУРНОЇ ОПТИМІЗАЦІЇ (BESO)

М. Несторович, О.Мілошевич, З. Шобіч

У цій статті розглядаються можливості застосування способу двобічної еволюційної структурної оптимізації (BESO) в архітектурному проектуванні. Мета полягає в тому, щоб вказати на перевагу застосування методу BESO на стадії концептуального проектування. Описаний приклад ілюструє ефективність такого підходу.