



Ειδική Υπηρεσία Διαχείρισης και Εφαρμογής Δράσεων  
στους τομείς Έρευνας, Τεχνολογικής Ανάπτυξης και Καινοτομίας

ΕΝΙΑΙΑ ΔΡΑΣΗ ΚΡΑΤΙΚΩΝ ΕΝΙΣΧΥΣΕΩΝ  
ΕΡΕΥΝΑΣ, ΤΕΧΝΟΛΟΓΙΚΗΣ ΑΝΑΠΤΥΞΗΣ  
& ΚΑΙΝΟΤΟΜΙΑΣ

## ΕΡΕΥΝΩ – ΔΗΜΙΟΥΡΓΩ – ΚΑΙΝΟΤΟΜΩ

### ΕΝΟΤΗΤΑ ΕΡΓΑΣΙΑΣ 3: ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΧΗΜΑΤΟΣ ΚΑΙ ΤΟΠΟΛΟΓΙΑΣ ΚΑΤΑΣΚΕΥΩΝ

#### Π3.2β Δημοσίευση σε περιοδικό

Κωδικός Έργου: **T1ΕΔΚ-05603**

Τίτλος Έργου: Ολιστική Υπολογιστική Πλατφόρμα  
Βελτιστοποίησης Υψηλών Επιδόσεων – HP-  
OCP

Φορείς: ΕΜΠ-ISAAR

Χρόνος: Από το πρόγραμμα: δεν προβλεπόταν

Παράδοσης: Υλοποιήθηκε: M20 (Μάιος 2020)





1st International Conference on Optimization-Driven Architectural Design (OPTARCH 2019)

## High performance topology optimization computing platform

Stefanos Sotiropoulos, Georgios Kazakis, Nikos D. Lagaros\*

*Veltion Engineering and Applied Sciences Optimization Lab, Institute of Structural Analysis & Antiseismic Research, School of Civil Engineering, National Technical University of Athens, 9, Heroon Polytechniou Str., Zografou Campus, GR-15780 Athens, Greece,  
E-mail: {st.sotirop@gmail.com, kzkgeorge@gmail.com, nlagaros@central.ntua.gr}*

---

### Abstract

One of the most challenging tasks in the construction industry nowadays, is to reduce the material demands and distribute, in the same time, the material among the structural system in the best possible way. Topology optimization is a design procedure that is increasingly used, to generate optimized forms of structures in several engineering fields. The current paper presents the Topology Optimization (TO) module of the High-Performance Optimization Computing Platform (HP-OCP) which focuses on civil engineering problems. More specifically the SIMP method [1] is implemented and the topology optimization problem is solved by using the OC algorithm. The HP-OCP is a platform which evaluates several objective functions, such as the volume of the structure, the compliance etc. and can solve constrained or unconstrained structural optimization problems. The above libraries are developed in C#. The core of the platform is created in such way that it can be integrated with any CAE program that has OAPI, XML or any other type of data exchange format. In the proposed work the structural analysis and design software SAP2000 is used. Theoretical aspects are discussed in order to implement the mathematical formulation in a commercial software. Basic and specific features are applied and representative examples are performed. One of the highlights of the proposed work is that the above module can be used for all kind of finite elements. Benchmark tests are presented with structures that are simulated by 2D plane-stress elements, 3D-solid elements and shell elements. Furthermore, it is independent of the type of the mesh, structured or unstructured, so both examples are presented. In the proposed work a powerful tool for both architects and civil engineers is introduced. The analysis and design of the structures are performed in SAP2000 software, in order to achieve a realistic result that could be a solution for a real-world structure.

© 2020 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

Peer-review under responsibility of the scientific committee of the 1st International Conference on Optimization-Driven Architectural Design

*Keywords:* Topology optimization; SAP2000; HP-OCP; C#.

---

\* Corresponding author. Tel.: +30 210 7722625; fax: +30 210 7721655.

E-mail address: [nlagaros@central.ntua.gr](mailto:nlagaros@central.ntua.gr)

## 1. Introduction

Over the recent years topology optimization is increasingly used in several engineering fields, such as aeronautics, automotive and aerospace industry. Consequently, most of commercial software that provide topology optimization solution are integrated into mechanical engineering-oriented analysis and design software. For example, OptiStruct by Altair [2], the Tosca Dassault Systems [3] that works with ABAQUS and MSC Nastran, Ansys topology optimization solution [4] and SOLIDWORKS [5]. The TO module of HP-OCP is developed in such way that it is independent of the software. In literature, several papers can be found that provide open source code for topology optimization for structured mesh in 2D [6,7] and 3D [8] problems. Talishi et al. [9] provided a MATLAB code that deal with unstructured mesh based on isoperimetric polygonal elements. Lagaros et al. [10] presented a C# code that solves topology optimization problems, which are discretized with 3D solid elements, using SAP2000. However, all of them focuses on specific types of elements and solve the classic linear elasticity problem. In the proposed work, the computation core of HP-TOCP is developed in C# programming language and is independent of the type of the mesh and the elements. Our software is able to be integrated with any commercial software analysis program and in this work the integration with SAP2000 is presented. We choose this program because we wanted to highlight the use of TO in a general-purpose structural software that is used mostly for civil engineering problems.

## 2. Mathematical Formulation

The basic idea of topology optimization is finding the shape of a structure by distributing material through an optimization algorithm. Three are the main approaches for dealing topology optimization problems, the SIMP method, the Level set and the BESO. In the present work the SIMP method is applied. The mathematical formulation of the compliance-based discrete topology optimization is defined as follows:

$$\min C(x) = F^T \cdot u(x) \quad (1a)$$

s.t.

$$K(x) \cdot u(x) = F \quad (1b)$$

$$\frac{V(x)}{V_0} = f \quad (1c)$$

$$0 < x_{min} \leq x \leq 1 \quad (1d)$$

where  $C$  is the compliance,  $F$  is the load vector,  $u$  the displacement vector,  $K$  the stiffness matrix and  $f$  the value of the final volume of the structure. The objective function is the minimization of the compliance and the constraints are the equilibrium of the finite element analysis and the volume that the final structure covers inside the initial domain. The design variables are the densities of the finite elements and can take values in the range of  $(0,1]$ , where 1 indicates that the material exists while 0 that it is absent. In the mathematical formulation a significantly small quantity  $x_{min}$  is defined in order to avoid numerical instabilities.

The above formulation can be solved using several mathematic algorithms like Optimality Criteria (OC), Sequential Linear Programming (SLP) and Method of Moving Asymptotes (MMA). In the proposed paper the OC algorithm is implemented. One of the most important part of OC algorithm is the calculation of the partial derivative of compliance with respect to the design variable. As it is shown in the classic formulation Eq. (2) we need to know the stiffness matrix of the structure.

$$\frac{\partial C(x)}{\partial x_e} = -u_e(x)^T \frac{\partial k_e(x)}{\partial x_e} u_e(x) \quad (2)$$

However, most of commercial analysis software don't provide this information. Lagaros *et al.* [10] proposed the following modified expression:

$$k_e(x_e) = x_e^p \cdot k_e^0 \Leftrightarrow k_e^0 = \frac{k_e(x_e)}{x_e^p} \quad (3)$$

$$\frac{\partial k_e(x_e)}{\partial x_e} = p \cdot x_e^{p-1} \cdot k_e^0 = p \cdot x_e^{p-1} \cdot \frac{k_e(x_e)}{x_e^p} = p \cdot \frac{k_e(x_e)}{x_e} \quad (4)$$

$$\frac{\partial C(x)}{\partial x_e} = -u_e(x)^T \cdot p \cdot \frac{k_e(x)}{x_e} u_e(x) = -p \frac{F^T u_e(x)}{x_e} = -\frac{p \cdot C_e}{x_e} \quad (5)$$

In this way the derivatives are calculated by just knowing the compliance of each element. In order to avoid the checkerboard problem and ensure that solution exists for the topology optimization problem a sensitivity filter is applied. In the next section the above implementation in SAP2000 is presented.

### 3. HP-TOCP

There are two basic stages that HP-TOCP is divided, the preprocessing and the iterative procedure. In both parts there are two independent processes that are performed. Firstly, the platform must retrieve all the necessary information from the static analysis program, herein the SAP2000. This is succeeded via creating some proper delegates functions, which are connected with the OAPI of SAP2000. In this way the libraries of HP-TOCP collect only the essential data from the commercial software. These functions remain the same independently to the static analysis program. The platform can be integrated using the above functions with any commercial program that provide OAPI, XML or any other type of data exchange format. The desired information are some basic features, like open save or close the program and some specific features like getting finite elements name, adding material property or modifying a material's Young Modulus. During the iterative part there is always interaction between the static program and the computation libraries, because after each analysis the updated analysis results must be retrieved.

#### 3.1. Preprocessing

The only prerequisite that TO module has, is the definition of the initial design domain model. At first, the initial geometry must be created in SAP2000 environment. Then the design domain must be discretized by any type of finite elements we want. The presented module is independent of the geometry and the type of the FE. Thus, the mesh may consist of either triangular or quadrilateral (2D elements) or tetrahedron, pentahedron and hexahedron (3D elements). In addition, the FE can be any type of plane, shell, plate or solid elements the user prefers. After the geometry and mesh generation, the boundaries conditions and the load cases and combinations must be defined. With the above procedure, TO module has all the necessary information to start the topology optimization method.

##### 3.1.1. Basic features

The first step of our module is to retrieve all the geometrical information of the SAP2000 model. Proper dictionaries are generated that contains all the names and coordinates of the nodes, the names and the connectivity of the finite elements and the load combinations' names. The next step is the definition of the finite elements and the

implementation of the topology optimization problem. The remarkable thing here is the way that SIMP method is applied. In order to specify any kind of finite element (2D or 3D) in SAP2000, first the equivalent section and material properties must be created. Each material property is produced by the SIMP equation and in the following it is assigned to a single section property. More specifically, the Young Modulus is modified according to the following equation:

$$E_i = E_{min} + (E_0 - E_{min}) \cdot \frac{i}{k} \quad (6)$$

where  $E_0$  is the initial Young modulus of the material, usually the unit value is used,  $E_{min}$  is a small quantity just to avoid numerical instabilities and  $k$  is the number of the material and section properties that was generated and stored in an array. In this way we have created a finite number of Young modulus values, but the design variable remains continuous. To deal with this problem, an extra function was developed in order to apply the power law. In the following the equation that identifies the index of the section property in the array and corresponds it to the equivalent density is presented:

$$index = Convert.ToInt32(x^p \cdot (k - 1)) \quad (7)$$

### 3.1.2. Specific features

In the above paragraph the absolutely necessary features were described in order to apply a topology optimization problem with the minimum requirements. In the following two specific functionalities in the preprocessing procedure are described. The first is the application of the sensitivity filter so to avoid the checkboard problem. Firstly, the centroid of each finite element is computed:

$$\bar{x}_i = \frac{\int_A x_i dA}{\int_A dA} \quad (8)$$

The above equation is independent of the shape, so it can be applied for all the types of finite elements. The next step is the evaluation of the distances between the centroid of all elements and their neighbors according to the given radius filter. Finally, the convolution operator matrix  $H_{e,i}$  of each element is created and the updated densities are evaluated from the equation below:

$$x_e = \frac{1}{\sum_{i \in N_e} H_{e,i}} \sum_{i \in N_e} H_{e,i} x_i \quad (9)$$

The second functionality is the definition of the Optimized and Non-optimized areas. In topology optimization problems sometimes, it is required some elements always to have material (e.g. a deck of a bridge) or the opposite (e.g. a hole for a pipe). In this case the user must create a group and named it “*NonOptimized*”. Then he must define the desired area and assign the equivalent elements to the above group. Next step is to define whether he wants this area to be void or full of material. The Optimized area is evaluated directly by the subtraction of the Non-optimized area from the full model. Finally, the user must determine if he wants the constraint of the volume fraction to be applied in the Optimized area or to the full model. Then, the TO module recognizes these areas and exclude them from the iterative part of the OC update scheme.

### 3.2. Iterative procedure

After performing the aforementioned computations, the TOCP is ready to start the iterative process. The first procedure in each iteration, is the assignment of the right property in each finite element using the update function.

Then, the static analysis is carried out by the SAP2000 software. The next step is to retrieve the necessary analysis results using the proper delegate functions. In the compliance-based topology optimization problem the only results that are required, are the joint forces and the displacements of each node of the elements, which are stored in arrays. This is repeated for all load combinations. The cross product between the arrays of displacements and forces over the nodes of each element results into the total compliance of the structural system. In the following, the partial derivative of each element is calculated according to Eq. (4) and stored in an array. Then, in order to confront the checkerboard problem, the values of the derivatives are filtered with convolution operator matrix  $H_{e,i}$ . Finally, the volume of the structure is computed in every inside loop of the OC algorithm until it converges. As it is said the design variable is continuous, so after algorithm convergence the elements' densities will have values from 0 to 1. However, in real world structures intermediate densities doesn't exist so an extra threshold is applied. In this way, if an element has density below of this value, it is considered that the material doesn't exist. The TO module recognize the elements that has these properties and delete them. In this way the optimized structure is visualized.

#### 4. Test Examples

In order to present all the capabilities of HP-TOCP, a wide range of different types of examples is performed and the results are illustrated in this section. More specifically, the results that are examined are some 2D test cases with different types of mesh (unstructured and structured), some additional features (filter, non-optimized areas, multiple load case) and different types of finite elements (plate and 3D solid elements).

##### 4.1. Cantilever beam

In the first example the cantilever beam problem is shown. The structure is discretized with 120 elements in the horizontal direction ( $nlx$ ) and 40 in the vertical ( $nly$ ) resulting into 4800 plane elements. All the nodes in the left side are restrained and a point load is applied in the bottom corner of the right edge. The value of the volume fraction ( $volfrac$ ) is 0.3, the number of the different section and material properties ( $k$ ) is 500, while the extra threshold that is used in order to delete the elements with lower density value is  $D_{lim} = 0.4$ . In Figure 1(b) the optimized layout without any filter scheme applied, is shown. The checkerboard problem is obvious and is overcome after the filter application with a radius ( $R$ ) equal to 3, as it is shown in Figure 1(c).

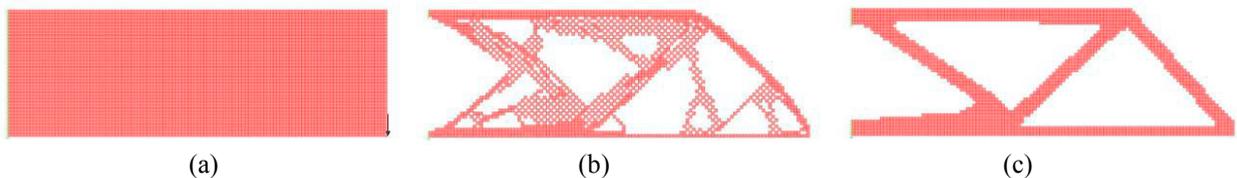


Fig. 1. Cantilever beam: (a) Initial domain, (b) Optimized structure without filter, (c) Optimized structure with filter

##### 4.2. Non-optimized and multiple load case

In Figure 2(a) the initial domain and the boundaries conditions of the bridge test case are shown. Both corners in the bottom side are supported while a distributed load is applied in the top side. The mesh of the problem remains the same as in 4.1 and the additional feature of non-optimized areas is implemented. More specifically, a group named “NonOptimized” was created in SAP2000 and the two rows of elements in the x direction at the top of the domain (with red color in Figure 2(a)) were assigned to this group. In this way, the optimized structure has material in these areas as it is shown in Figure 2(b). The rest of the parameters are:  $volfrac = 0.3$ ,  $k = 500$ ,  $D_{lim} = 0.5$ ,  $R = 3$ .

In order to highlight the integration of topology optimization problem in the conceptual design of civil engineering problems a Moment-resisting frames (MRF) test case is employed. The mesh of the initial domain is  $nlx = 40$ ,  $nly = 200$ , in the bottom corners the nodes are restrained and five horizontal point loads at equal

distances between them are applied at both sides, as it is shown in Figure 2(c). The rest of the parameters are:  $volfrac = 0.3$ ,  $k = 500$ ,  $D_{lim} = 0.2$ ,  $R = 3$ . In Figure 2(d), the optimized layout is presented and a structure that could be used as an MRF is depicted.

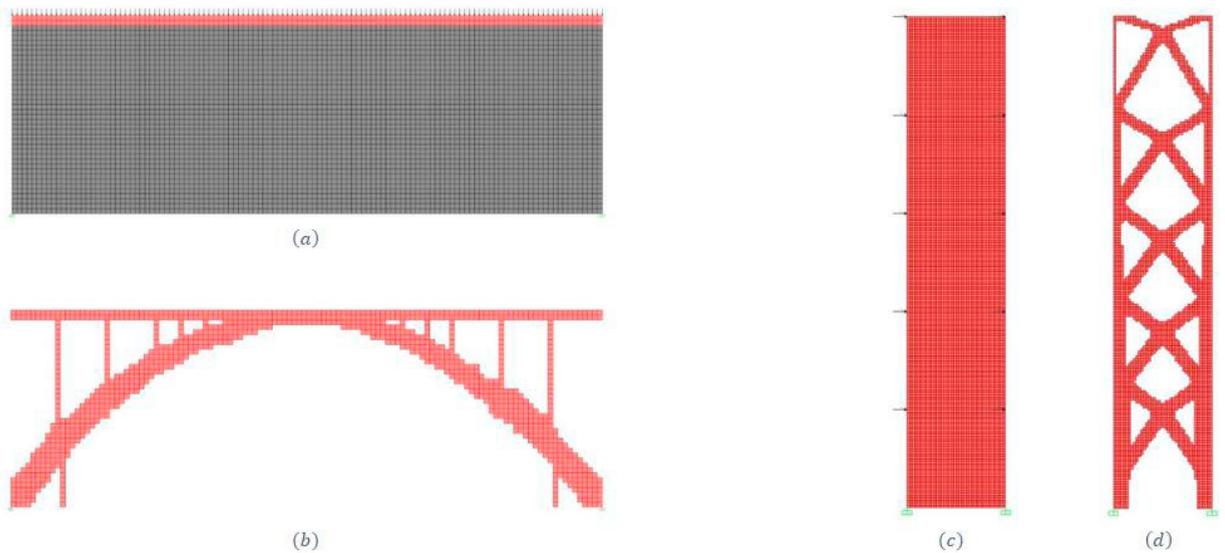


Fig. 2. (a) Initial domain and (b) optimized structure for Bridge problem, (c) Initial domain and (d) Optimized structure for MRF test case

#### 4.3. Unstructured mesh and Plate case

In Figure 3(a) the same cantilever beam with the example of section 4.1 is presented. In this case the initial domain is discretized with an unstructured mesh that consists of 5000 triangular elements that are generated through Delaunay triangulation. The rest of the parameters are the same with the previous cantilever beam example and in Figure 3(b) the optimized layout is illustrated. The structure is almost the same if it is considered that the sharp edges was created because of the mesh's triangles.

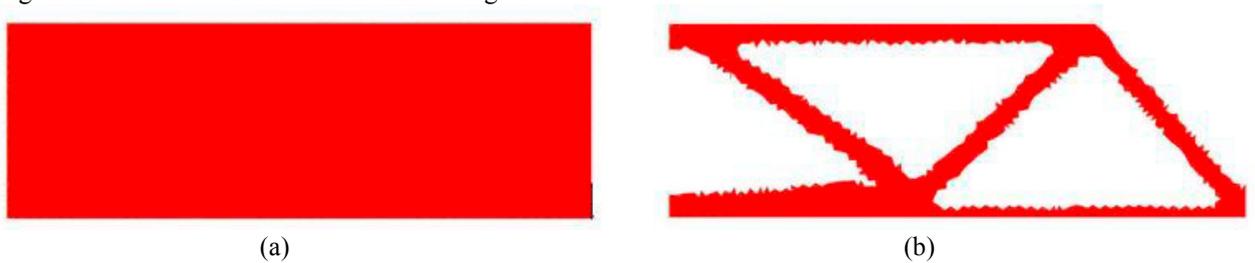


Fig. 3. Cantilever beam: (a) Initial unstructured mesh, (b) Optimized layout

In Figure 4(a) a plate test case is examined. The initial structure is simulated with plate finite elements and the parameters of the problem are:  $n_{lx} = 20$ ,  $n_{ly} = 20$ ,  $volfrac = 0.3$ ,  $k = 500$ ,  $D_{lim} = 0.4$ ,  $R = 3$ . In Figure 4(b) the final structured is presented. In this example, it is verified that TO module of HP-OCP can be used for different types of finite element, like plates and shell, and not only plane stress based, like Q4 and 3D solid elements.

#### 4.4. Typical 3D test cases

In the last part of this section two typical 3D test cases that are simulated by 8-node hexahedron solid elements are presented. In Figure 5(a) the 3D wheel problem is employed. The four corners of the bottom side are supported, a point load is applied in the center of the bottom side and the discretization of the domain is  $nlx = 40$ ,  $nly = 40$ ,  $nlz = 20$ , leading into 32000 solid elements. The rest of the parameters are:  $volfrac = 0.2$ ,  $k = 500$ ,  $D_{lim} = 0.4$ ,  $R = 3$ . In Figure 5(b) the final structure is shown.

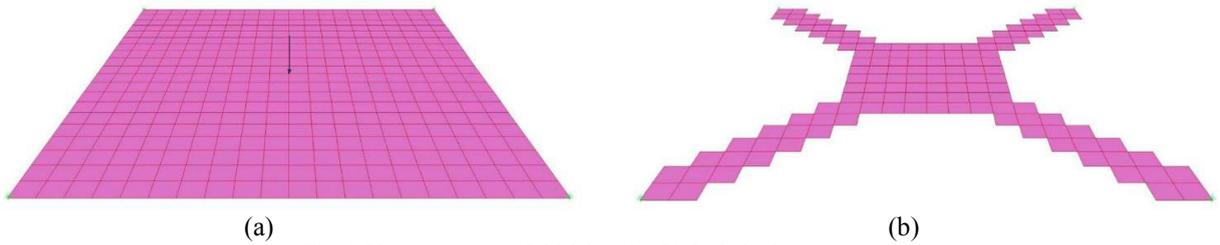


Fig. 4. Plate test case: (a) Initial domain, (b) Optimized structure

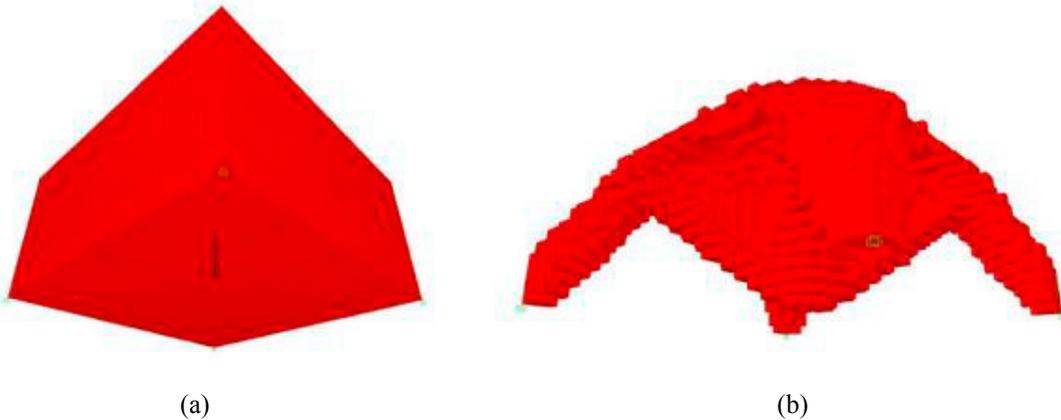


Fig. 5. 3D wheel test case: (a) Initial domain, (b) optimized structure

The last example is the classic 3D MBB beam problem. In Figure 6(a) the boundaries conditions and the mesh discretization are presented. More specifically, all the corners of the bottom side are restrained, a joint load is applied in the center of the top side and the discretization of the structure is  $nlx = 60$ ,  $nly = 10$ ,  $nlz = 10$ , leading into 6000 solid elements. The rest of the parameters are:  $volfrac = 0.2$ ,  $k = 500$ ,  $D_{lim} = 0.5$ ,  $R = 3$ . In Figure 6(b) the final structure is shown.

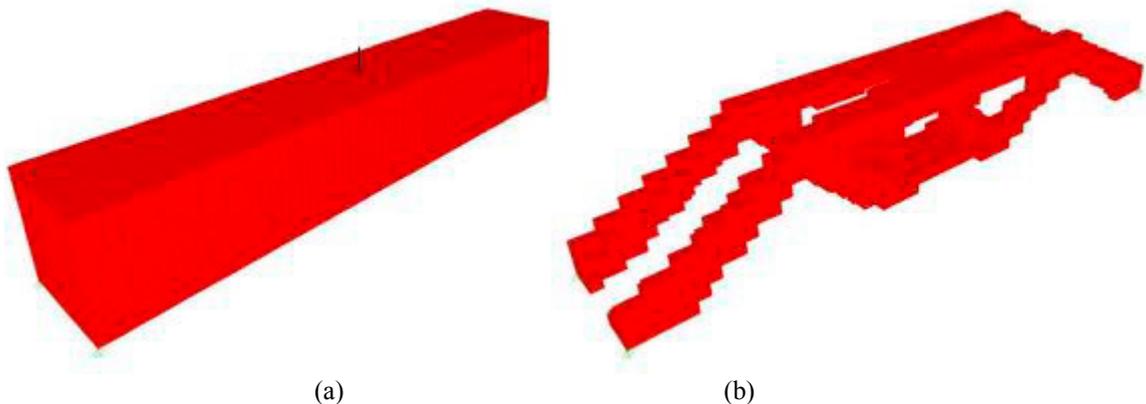


Fig. 6. MBB test case: (a) Initial domain, (b) optimized structure

## 5. Conclusion

In this work the capabilities of the TO module of HP-OCP are presented. The platform was developed in C# programming language and was structured in such way that it is independent of the static analysis software which performs the structural analysis. In the proposed paper, some theoretical aspects that must be taken into account in order to modify the classic formulation and make it valid for integration with commercial software are discussed. Some modifications of the SIMP method and the calculation of the OC algorithm are introduced and the connection with the OAPI provided by SAP2000 through some delegates functions is presented. In addition, some specific features like filter application, definition of non-optimized areas and multiple load case problem are described thoroughly. We performed compliance-based topology optimization in a wide range of different types of structures, in order to reveal all the capabilities of HP-TOCP. More specifically, 2D test cases are presented in order to demonstrate the filter, non-optimized areas and multiple load case application and the ability of handling structures with unstructured mesh. In addition, a structure simulated with plate finite elements and two typical 3D test cases are presented. In this way, TO module provides a holistic approach of dealing with topology optimization problems in field of civil engineering.

## Acknowledgements

This research has been co-financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH–CREATE–INNOVATE (project code: T1EDK-05603, “HP-OCP: Holistic, High Performance Optimization Computing Platform”).

## References

- [1] M.P. Bendsoe, O. Sigmund, Material interpolations in topology optimization, *Archive of Applied Mechanics*, 69 (1999) 635-654.
- [2] Optistruct: <http://www.altairhyperworks.com/product/OptiStruct/>, (last accessed November 2019)
- [3] Tosca: <https://www.3ds.com/products-services/simulia/products/tosca/>, (last accessed November 2019)
- [4] Ansys topology optimization: <http://www.ansys.com/products/structures/topology-optimization>, (last accessed November 2019)
- [5] SolidWorks: [http://help.solidworks.com/2018/english/WhatsNew/c\\_generative\\_design\\_study.htm](http://help.solidworks.com/2018/english/WhatsNew/c_generative_design_study.htm), (last accessed November 2019)
- [6] O. Sigmund, A 99 line topology optimization code written in Matlab, *Structural and Multidisciplinary Optimization*, 21(2) (2001) 120-127.
- [7] E. Andreassen, A. Clausen, M. Schevenels, B.S. Lazarov, O. Sigmund, Efficient topology optimization in Matlab using 88 lines of code, *Structural and Multidisciplinary Optimization*, (2011) 43(1): 1-16.
- [8] K. Liu, A. Tovar, An efficient 3D topology optimization code written in Matlab, *Structural and Multidisciplinary Optimization*, (2014) 50(6): 1175-1196.
- [9] C. Talischi, G.H. Paulino, A. Pereira, I.F.M. Menezes, Polytop: a MATLAB implementation of a general topology optimization framework using unstructured polygonal finite element meshes. *Structural and Multidisciplinary Optimization*, (2012) 45(3): 329-357.
- [10] N.D. Lagaros, N. Vasileiou, G. Kazakis, A C# code for solving 3D topology optimization problems using SAP2000, *Optimization and Engineering*, (2019), 20(1) 1-35.

## Σχόλια - Προβλήματα - Παρατηρήσεις

Δεν υπήρχαν παρατηρήσεις

	Επιστημονικός Υπεύθυνος Έργου	Συντονιστής Έργου
<b>Υπογραφή:</b>		
<b>Όνοματεπώνυμο :</b>	Ν. ΛΑΓΑΡΟΣ	Χ. ΚΩΣΤΟΠΑΝΑΓΙΩΤΗΣ
<b>Ημ/νία :</b>	10/09/2020	10/09/2020