An efficient Python code for modelling strut-and-tie tridimensional models for topological optimization using SESO and ESO methods

Higor E. V. O. Prado[1], Aline R. M. Madruga[1], Hélio Luiz Simonetti[2], Francisco de Assis das Neves[3], Virgil Del Duca Almeida[4]

[1]*Undergraduate, Control and Automation Engineering, Federal Institute of Minas Gerais (IFMG),*
Rua Itamarati 140, 32677-764, MG/Betim, Brasil
higorevop@gmail.com, madruga.alinerm@gmail.com
[2]*Doctor, Department of Mathematics, Federal Institute of Minas Gerais (IFMG)*
Rua Itamarati 140, 32677-764, MG/Betim, Brasil
helio.simonetti@ifmg.edu.br
[3]*Doctor, Department of Civil Engineering, Federal University of Ouro Preto (UFOP)*
Campus Morro do Cruzeiro, Bauxita, 35400-000, MG/Ouro Preto, Brasil
fassis@ufop.edu.br
[4]*MSc, Department of Automation and Control Engineering, Federal Institute of Minas Gerais (IFMG)*
Rua Itamarati 140, 32677-764, MG/Betim, Brasil
virgil.almeida@ifmg.edu.br

Abstract. This paper presents a methodology for the automatic generation of optimal strut-and-tie models in reinforced concrete structures using the Evolutionary Structural Optimization (ESO) and Smoothing Evolutionary Structural Optimization (SESO) methods. The presented methodology is developed to minimize the compliance and is implemented in Python. The proposed approach deals with the generation of projects with truss-like elements, addressing a console. The obtained results show a good accuracy compared with those described in the literature.

Keywords: Strut-and-tie, Topology Optimization, SESO, ESO, Python

1 Introduction

The regions of a structure in which Bernoulli's hypothesis of linear distribution of strains is considered valid, will be referred as B regions (B for Bernoulli). Its internal forces or stresses can be derived from moments, shear and axial forces analyzed through the static system of beams, frames, plates, among others. If there are no fractures, stresses are calculated using the bending of theory for linear elastic material. For B regions with fractures, Strut-and-Tie Models (STM) or standard methods set out in codes and standards are applied. These standard methods are not applicable to other regions and details of a structure where the strain distribution is significantly non-linear, e.g., next of concentrated loads, corners, bends, gaps and other discontinuities, Schlaich [1].

The internal flow of stresses in regions D, (D for Discontinuity), can be reasonably well described by STM. Not much precision is needed to determine the dividing sections between regions B and D. Regions D, Fig. 1, are considered as the parts of the structure within the range of geometric discontinuity or load, in which the hypothesis of linear distribution stress/strain is not applicable, Liang [2]. In Fig. 1, the sections can be assumed to be approximately at a distance h from the geometric discontinuity or from the concentrated load, where h is equal to the depth of the adjacent region B, Fig. 1. This hypothesis is justified by the Saint-Venant principle.

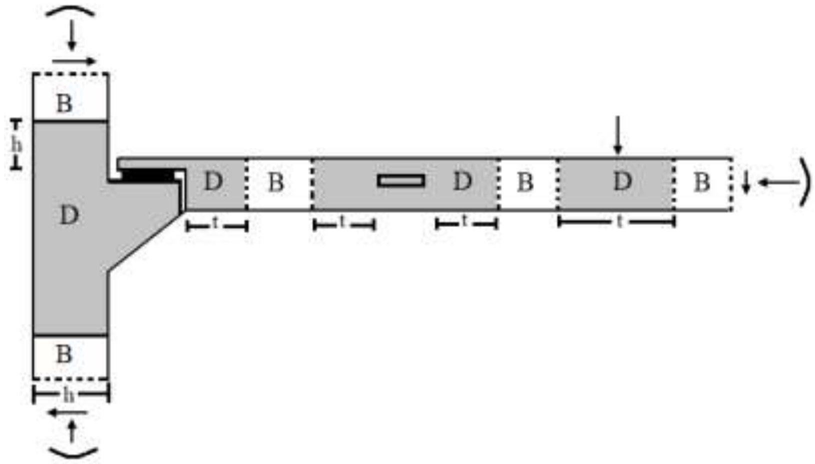

Figure 1 – Example of regions B and D

For a more realistic analysis of the physical behavior of these regions, it is common to use the strut-and-tie model, which is a generalization of the classical analogy of the truss model for beams. This analogy, presented by Ritter and Mörsch at the beginning of the 20th century, associates the reinforced concrete beam with an equivalent truss structure. The discrete elements (bars) represent the tensile stress fields (called tie) and compression (called strut) that arise internally to the structural element under the effect of bending. This analogy has been improved and is still used by technical standards in the design of beams in reinforced concrete, with bending and shear force, and establishes different criteria for determining the safety limits in their procedures. Kupfer [3] and Mörsch [4] have extended the truss models to the design of concrete beams under torsion. Lampert and Thürlimann [5] and Thürlimann [6] used the truss models for the design of concrete beams under torsion and bending. The strut-and-tie models, an extension of the truss analogy method, were proposed by Marti [7] for the design and detailing of regions of discontinuity in the design of prestressed concrete structures.

Although STM has been recognized as a rational approach for designing regions of discontinuity in reinforced concrete structures according to European, Canadian and American codes, these Code provisions still require improvements due to uncertainties in selecting optimal strut-and-tie, especially in the case of complex geometry or loading conditions, Liang [2]. To overcome these difficulties, automate the design of these models and improve the efficiency in the construction of STM in reinforced concrete structures. The Topological Optimization (TO) theory has been used as an alternative approach. Thus, the application of OT has proved to be a valuable path for research linked to projects of interest to industries, since it facilitates the molding of materials under certain design conditions.

Almeida et al. [8] introduce a numerical technique to determine STMs using the Smoothing-ESO (SESO) method. The basic idea of this technique is to identify the flow of stresses generated in the structure, established in members and support, and quantify their value for future structural design. Other works can be cited in the study of STMs, such as the models based on densities, Ramm and Maute [9], and Bruggi [10], using criteria for maximizing stiffness with prescribed volume. Liang [11,12,13] uses ESO in a two-dimensional linear elastic analysis to find the optimal STMs for reinforced concrete structures and prestressed concrete members. Such analyzes are elastic in plane state.

With the development of Industry 4.0, computational resources and advanced manufacturing methods are constantly improved. Thus, the TO, introduced by Bendsoe and Kikuchi [14], whose objective is to determine optimal values for the design variables, which represent the distribution of the material in the solution domain, constitutes an important tool with its lighter and attractive designs to the industry.

This article approaches the three-dimensional TO using the Python programming language in the implementation of a methodology that allows the automatic generation of STMs in reinforced concrete structures using the ESO and SESO methods.

Shobeiri [15] approaches STM using the Bi-directional Evolutionary Structural Optimization (BESO) method in a three-dimensional analysis. The 3D analysis, in the TO field using MEF, has highlights in the work developed by Liu and Tovar [16], who proposed a 169-line MATLAB code incorporating efficient strategies for the three-dimensional TO using a SIMP model modified with 8-node hexahedral elements with three degrees of freedom per node. Zhou and Wang [17] present a three-dimensional TO program using MATLAB. Its code is referred to as the 177-line program, a successor to the 99-line program by Sigmund [18]. Zuo and Xie [19] present a 100-line Python code for three-dimensional TO. The code adopts the Abaqus Scripting Interface, which provides convenient access to advanced finite element analysis. It is designed to minimize compliance with a volume constraint using the BESO method. In Vogiatzis [20] the authors developed a MATLAB code using the Level Set Method (LSM), which can be integrated with the TO procedure and convert the design into an STL file (STereoLithography), which is the format of 3D-printing.

Section 2 presents the formulation of the optimization problem for ESO and SESO. Section 3 succinctly shows some information regarding the Python language and the methodology used in this article. Section 4 describes the numerical example used in this work and the results that were obtained.

2 Problem Formulation

Considering the classical topology problem for the maximum stiffness of statically loaded linear elastic structures with a concentrated force, a mathematical formulation for TO of a continuum structure can be discussed. This equates to minimizing the total elastic energy for a structure in a state of equilibrium. Thus, minimizing the work performed by external forces can be expressed as:

$$\begin{aligned} &\text{Minimize: } W(u) \\ &\text{Subject to: } Ku = F \\ &V(X) = \sum_{i=1}^{NE} x_i V_i - \bar{V} \leq 0 \\ &X = \{x_1\ x_2\ x_3\ \dots\ x_n\},\ x_i = 1\ or\ x_i = 0 \end{aligned} \tag{1}$$

Considering the topology optimization problem as minimizing the deformation energy of a given structure considering the equilibrium, it follows that W=2U. The problem can then be defined as:

$$\begin{aligned} &\text{Minimize: } U(X) = \frac{1}{2} U^T K u = \sum_{1}^{NE} \frac{1}{2} \int_{Ve} \varepsilon_e^T E_e(x) \varepsilon_e dV_e \\ &\text{Subject to: } Ku = F \\ &V(X) = \sum_{i=1}^{NE} X_i V_i - \bar{V} \leq 0 \\ &X = \{x_1\ x_2\ x_3\ \dots\ x_n\},\ x_i = 1\ or\ x_i = 0 \end{aligned} \tag{2}$$

where: E_e is the element's elasticity matrix, ε_e is the element's strain vector, V_e is the volume of an element, NE is the number of finite elements of the mesh, K is the stiffness matrix, , $Ku = F$ is the equilibrium equation, F is the vector of loads applied to the structure, x_i is the design variable of the ith element, X is the vector of design variables.

2.1 ESO and SESO

Xie and Steven [4] developed a simple way to make changes in the topology of a structure through a heuristic of gradual and systematic removal of finite elements from the mesh corresponding to regions that did not effectively contribute to the good performance of the structure. This optimization methodology was called ESO.

SESO is based on the ESO philosophy and applies a weighting to the constitutive matrix so that the element that would be removed from ESO, is maintained and receives a relaxation, Simonetti [23]. This relaxation is processed through the application of a degradation in the value of its initial stiffness in such a way that it remains in the design domain and that, naturally, during the removal process, its influence can contribute and determine its permanence or definitive removal from the design domain.

In this article, the maximization of the structure stiffness, minimizing the strain energy, is evaluated. In this way, the strain energy of each element can be evaluated according to the following inequality:

$$U_e < RR_k * (U_i^{max}) \tag{3}$$

where U_e is the strain energy of the element, U_i^{max} is the maximum strain energy of the structure in the effective iteration k e RR is the Rejection Ratio of the k-th equilibrium state which can be defined according to Equation (4) and ER is the Evolutionary Ratio:

$$RR_{k+1} = RR_k + ER \tag{4}$$

The SESO procedure can then be interpreted as follows :

$$D(j) = \begin{cases} D_0, if\ j\ \in\ \Gamma_i \\ D_0 . \eta_j(\bar{\Gamma}), if\ j \in\ \Gamma_{GS} \\ 0, if\ j\ \in\ \Gamma_{LS} \end{cases} \tag{5}$$

Where: $D(j)$ is the constitutive matrix of element j, $\bar{\Gamma} = \Gamma_{LS} + \Gamma_{GS}$ is the domain of elements that meet Eq. (5), Γ_{LSi} is the domain of elements that must be effectively removed, Γ_{GSi} is the domain of elements that are returned to the structure, $0 \leq \eta(\bar{\Gamma}) \leq 1$ is a weighted function.

2.2 Performance Index

Liang [2] presents in his research on linear elastic structures an approach that defines an energy-based performance index with the following equation:

$$PI = \frac{C_0 V_0}{C_k V_k} \tag{6}$$

where PI is the performance index of the structure, C_0 is the initial compliance, V_0 is the initial volume of the structure, C_k is the compliance in the k-th iteration, and e V_k is the volume in the k-th iteration.

3 The Python Programming Language

Python is an interpreted, imperative, object-oriented, functional, dynamically typed programming language. It is designed with the philosophy of emphasizing the importance of programmer effort over computational effort. Therefore, it prioritizes code readability. Python is a general-purpose language that, in recent times, has been widely used in scientific projects for having an expressive syntax and a rich collection of built-in data types, Millman and Aivazis [21].

NumPy is a Python library that seeks to bring the computational power of languages like C and Fortran to Python. It is a package that supports multidimensional arrays and matrices and has a large collection of mathematical functions to work with these structures. The SciPy library is Open Source, made for mathematicians, scientists and engineers. It is based on NumPy, providing convenient and fast manipulation of n-dimensional arrays. PyVista is a visualization library for Python that seeks to simplify the creation of 3D meshes and plotting routines without compromising execution speed, Sullivan [22].

In this article, the Python programming language and the NumPy and SciPy libraries are used to build a code that allows the construction of STM in reinforced concrete structures using the ESO and SESO methods. In addition, the PyVista library is used to visualize the generated structures.

3.1 Methodology

For the development of the project, the formulation of the SESO method, described in Simonetti [23, 24], was applied.

For three-dimensional structure models it was necessary to formulate the finite element analysis in 3D and its implementation. For this purpose, the works of Torres [25], and Liu and Tovar [16] were used as basic references.

The implementation was based on the MATLAB code initially developed by Sigmund [18] and later used and modified by Liu and Tovar [16] and Simonetti [26]. To build the SESO-3D and ESO-3D code in Python, the NumPy and SciPy numerical libraries were used.

4 Numerical Example

In this section, an example is presented in order to demonstrate the efficiency of the ESO and SESO methods presented in the treatment of strut-and-tie modeling.

4.1 Console

This example shows the efficiency of the SESO method in handling the modeling of STM of a console structure. The geometry, boundary conditions and design domain of the structure are shown in Fig. 2a. This structure is subjected to a point load of $500kN$. The compressive strength of the concrete used in this example is $32MPa$. Young's modulus of concrete $\mathrm{E_c} = 28567\mathrm{MPa}$ and Poisson's coefficient $\upsilon = 0.15$ were adopted for this analysis. The prescribed volume is 12%, the filter radius was set to 1.5 mm and an evolutionary ratio of 2% was specified in the optimization process. It is noteworthy that for a smaller RR the optimization evolves to an optimal design, however, this optimal design is obtained with a higher computational cost. Note also that $0.01 \leq RR \leq 0.03$ has little effect on the final form of the optimal design. A 50x180x10 mesh consists of 54,000 eight-node hexahedral elements.

The topology optimized according to the ESO and SESO methods, Fig. 2b and 2c, resulted in a material reduction of the order of 85% of the initial structure and idealized for the STM model, see Fig. 2c. It can be seen that the optimal settings obtained for the strut-and-tie are similar and clearly represent the location of nodal zones and strut-and-tie. The results demonstrate that the SESO and ESO optimization procedures can be used successfully in the generation of STM models.

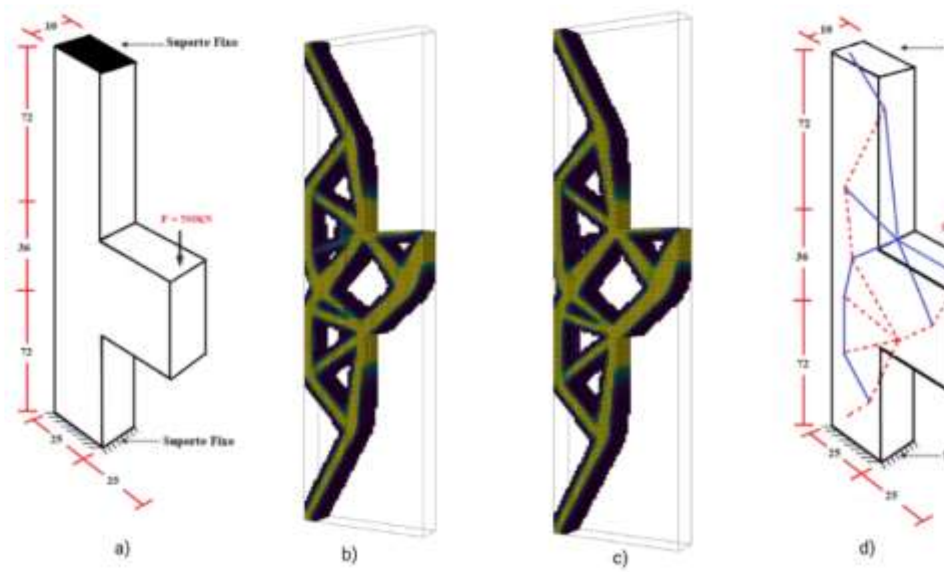

Figure 2. (a) Project domain, (b) Optimal Topology – ESO, (c) Optimal Topology – SESO and (d) STM Model

The PI is developed to monitor the optimization process and identify a viable stationary optimal for the project from the optimization history. In this article, this index is used as stopping criterion for the optimization algorithm. It was observed that the proposed approach tends to generate truss-type designs that are suitable for STMs when a small prescribed volume is defined. This is coherent with the minimal compliance approaches to optimization presented in Bruggi [10] and Shobeiri [15]. The history of the PI, the compliance evolution and the history of the optimization procedure for iterations 10 and 15 are shown in Fig. 3a and 3b for the ESO and SESO models, respectively. Note that the PI can monitor compliance movements, see Fig. 3a and 3b, for ESO and SESO models. It is noticed that due to the breaking of bars during the optimization procedure the compliance oscillates in iterations from 15 to 25 until it stabilizes, while the PI monitors these movements oscillating in the same iterations until it stabilizes at 2.82 in the optimal iteration.

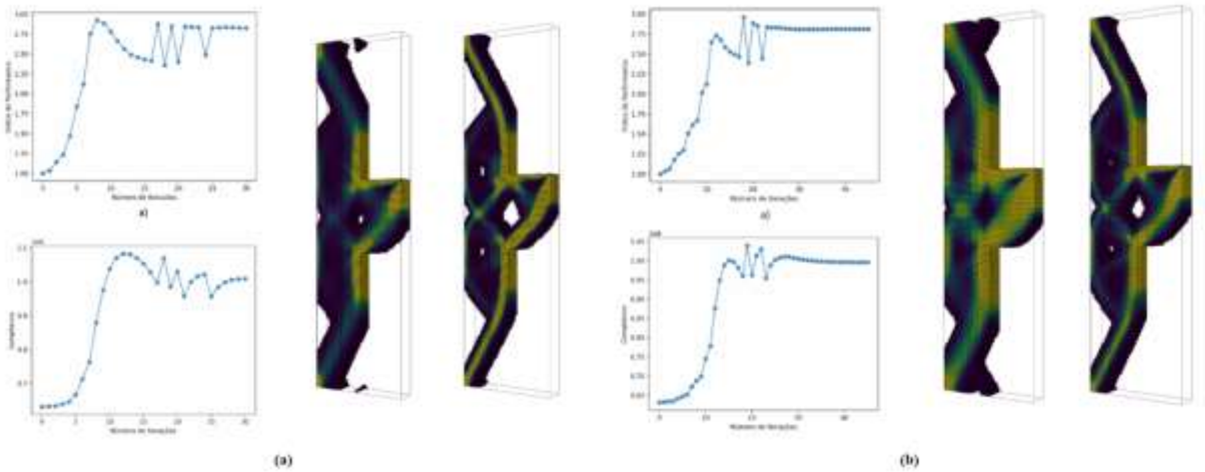

Figure 3. (a) SESO – Compliance and performance index by iteration and topologies in iterations 10 and 15. (b) ESO - Compliance and performance index by iteration and topologies in iterations 10 and 15.

5 Conclusions

In this article, the Python programming language was used to implement a methodology that allowed the automatic generation of strut-and-tie models in reinforced concrete structures using the ESO and SESO methods. For the three-dimensional structure models, it was necessary to formulate the finite element analysis in 3D and the implementation of its numerical calculation. The results showed that the ESO/SESO optimization procedure can be successfully used for STM models and that the Python programming language can be a good alternative for implementing topological optimization techniques.

Acknowledgements

The authors are grateful to the Federal Institute of Science and Technology of Minas Gerais, CNPq (National Council of Scientific and Technological Development), CAPES (the Coordination for Improvement of Higher-Education Personnel), FAPEMIG (Foundation for Research Support of Minas Gerais State) and Universidade Federal de Ouro Preto (UFOP) Numbers 30503/2018-5 for their financial support to this research.

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] Schlaich, J., & Schafer, K. Design and detailing of structural concrete using strut-and-tie models. *Structural Engineer*, *69*(6), 113-125, 1991.
[2] Liang, Q. Q. Performance-Based Optimization of Structures: Theory and Applications. New York: Taylor and Francis Group, 2005.
[3] Kupfer, H. "Generalization of Mörsch's Truss Analogy Using Principle of Minimum Strain Energy." CEB Bulletin d'information, Comite Ero-International du Beton 40: 44–57, 1964.
[4] Mörsch, E. Der Eisenbetonbau-Seine Theorie und Anwendung. Stuttgart: Wittwer 1 (Part 1), 1920.

[5] Lampert, P., and B. Thürlimann. "Ultimate Strength and Design of Reinforced Concrete Beams in Torsion and Bending." In Ultimate Strength and Design of Reinforced Concrete Beams in Torsion and Bending/Résistance et dimensionnement des poutres en béton armé soumises à la torsion et à la flexion/Bruchwiderstand und Bemessung von Stahlbetonbalken unter Torsion und Biegung, 107–131. Basel: Birkhäuser, 1971.
[6] Thürlimann, B., P. Marti, J. Pralong, P. Ritz, and B. Zimmerli. Vorlesung Zum Fortbildungskurs Für Bauingenieure. Zurich: Institut für baustatik und konstruktion, ETH, 1983.
[7] Marti, P. "Basic Tools of Reinforced Concrete Beam Design." ACI Structural Journal 82 (1): 46–56, 1985.
[8] Almeida, V. S., H. L. Simonetti, and L. O. Neto. "Comparative Analysis of Strutand-Tie Models Using Smooth Evolutionary Structural Optimization." Engineering Structures 56: 1665–1675, 2013.
[9] Ramm, E., and K. Maute. "Topology Optimization—A General Tool in Structural Design." In Proceedings of the Euro-C 94 Computational Modelling of Concrete Structures, edited byH.Mang, N. Bicanic and R. de Borst, 805–824. Swansea: Pineridge Press, 1994.
[10] Bruggi, M. "Generating Strut-and-Tie Patterns for Reinforced Concrete Structures Using Topology Optimization." Computers and Structures 87: 1483–1495, 2009.
[11] Liang, Q. Q., Y.M. Xie, andG. P. Steven. "Topology Optimization of Strut-and-Tie Models in Reinforced Concrete Structures Using an Evolutionary Procedure." ACI Structural Journal 97 (2): 322–330, 2000.
[12] Liang, Q. Q., Y. M. Xie, and G. P. Steven. "Generating Optimal Strut-and-Tie Models in Prestressed Concrete Beams by Performance-Based Optimization." ACI Structural Journal 98 (2): 226–232, 2001.
[13] Liang, Q. Q., B. Uy, and G. P. Steven. "Performance-Based Optimization for Strut-Tie Modeling of Structural Concrete." Journal of Structural Engineering 128 (5): 815– 823, 2002.
[14] Bendsoe, M.P. and Kikuchi, N. Generating Optimal Topologies in Structural Design Using a Homogenization Method. Computer Methods in Applied Mechanics and Engineering, 71, 197-224, 1988.
[15] Shobeiri, V., & Ahmadi-Nedushan, B. Bi-directional evolutionary structural optimization for strut-and-tie modelling of three-dimensional structural concrete. Engineering Optimization, 49(12), 2055-2078, 2017.
[16] Liu, Kai, and Andrés Tovar. "An efficient 3D topology optimization code written in Matlab." Structural and Multidisciplinary Optimization 50.6 : 1175-1196, 2014.
[17] Zhou S, Wang MY 3d structural topology optimization with the simp method. URL http://www2.acae.cuhk. edu.hk/~cmdl/download.htm, 2005.
[18] Sigmund O. A 99 line topology optimization code written in matlab. Structural and Multidisciplinary Optimization 21(2):120–127, 2001.
[19] Zuo, Zhi Hao, and Yi Min Xie. "A simple and compact Python code for complex 3D topology optimization." Advances in Engineering Software 85: 1-11, 2015.
[20] Vogiatzis, Panagiotis, Shikui Chen, and Chi Zhou. "A MATLAB Code for Integrated Additive Manufacturing and Level-Set Based Topology Optimization." ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers, 2016.
[21] Millman, K. J. and Aivazis M.; Python for Scientists and Engineers, in Computing in Science & Engineering, vol. 13, no. 2, pp. 9-12, March-April 2011, doi: 10.1109/MCSE.2011.36, 2011.
[22] Sullivan et al., . PyVista: 3D plotting and mesh analysis through a streamlined interface for the Visualization Toolkit (VTK). Journal of Open Source Software, 2019.
[23] Simonetti, H.L..; Almeida, V.S.; De Oliveira Neto, Luttgardes. A smooth evolutionary structural optimization procedure applied to plane stress problem. Engineering Structures, v. 75, p. 248-258, 2014.
[24] Simonetti, H.L.; Almeida, V.S.; De Assis das Neves, Francisco. Smoothing evolutionary structural optimization for structures with displacement or natural frequency constraints. Engineering Structures, v. 163, p. 1-10, 2018.
[25] Torres, I. F. R. Desenvolvimento e aplicação do método dos elementos finitos generalizados em análise tridimensional não-linear de sólidos. Tese (Doutorado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2003.
[26] Simonetti, H. L. ; Almeida, V.S. ; Neves, F. A. ; Almeida, V. D. . Topology Optimization for Elastic Analysis of 3D Structures using Evolutionary Methods. In: XL Ibero-Latin-American Congress on Computational Methods in Engineering (CILAMCE 2019), 2019, Natal. Topology Optimization for Elastic Analysis of 3D Structures using Evolutionary Methods, 2019.