17th Brazilian Congress of Thermal Sciences and Engineering
November 25th-28th, 2018, Águas de Lindóia, SP, Brazil

TOPOLOGICAL OPTIMIZATION OF DISCRETIZED FLUID-STRUCTURE SYSTEMS WITH UNSTRUCTURED MESHES

Fernanda Bichet Link
Vitor Souza Viana Silva
Gustavo Comerlato Rodrigues
Emanuel Moutinho Cesconeto
Ederval de Souza Lisboa
João Baptista Dias Moreira
Walter Jesus Paucar Casas
Mechanical Engineering Department, Federal University of Rio Grande do Sul, 425 Sarmento Leite st., Porto Alegre, RS 90050-170, Brazil.
feulink@gmail.com
vitorsvs83@hotmail.com
gustavocomerlato@gmail.com
baptista.moreira@ufrgs.br
emanuelemc@hotmail.com
ederval.lisboa@yahoo.com.br
walter.paucar.casas@ufrgs.br

Abstract. *This work presents a study of a methodology for structural topological optimization of a fluid-structure coupled system with the objective of maximizing structural rigidity while its structural volume is gradually removed in order to obtain, for a fraction of volume, a structure with rigidity close of the original. The structure is part of a fluid-structural system and is subject to static external loads as well as loads imposed by the fluid. The problems addressed were discretized with an unstructured and irregular mesh and the topological optimization through the BEFSO (Bi-Directional Evolutionary Fluid-Structural Optimization), method which is an improvement of the BESO method. The results obtained demonstrate the robustness of the program developed since it is able to reproduce the cases of optimization presented in the literature, as well as to solve complex problems solved by conventional methods that normally use regular meshes or commercial software.*

Keywords: *Topology optimization, FEM, fluid-Structure, unstructured mesh*

1. INTRODUCTION

As an alternative to the development of optimal structural concepts, the topological optimization method has been widely applied. This application is mainly directed in cases where the mass of the structure is critical, as in the sectors of the aerospace and aeronautics industry, for example. Research on this can be found in Paulino *et al.*, 2013. These concepts are used in the initial stages of a project, where an initial form of the structure is sought that will be improved throughout its development process and therefore is critical, because it limits the shape that the final structure can have.

There are several topology optimization methods, the main ones being BESO (Bidirectional Evolutionary Structural Optimization) and SIMP (Solid Isotropic Material with Penalization) as described by Huang and Xie, 2010. These methods differ in the treatment given to the optimized variable, in this case the elements of the structure: the BESO method treats them in a discrete way, that is, each element can have a value that represents its presence or another value that represents its absence, while the SIMP treats them continuously, that is, each element has a density value between 100% and 0%. A detailed analysis of the differences can be found in Huang and Xie, 2010.

This paper deals with a modification of the BEFSO (Two-dimensional Structural Evolutionary Fluid Optimization) method applied in fluid structure systems, which allows its use with unstructured meshes, providing its use in more complex geometries, such as in aircraft wings where the domain does not can be divided into regular elements. In such cases, the domain can be approached using a greater amount of regular elements, but with the proposed method it is possible to discretize the structure with fewer elements without losing the boundary profile. In this sense, the efficacy and the capacity of the method will be verified through the comparison with cases found in the literature.

2. THEORETICAL BACKGROUND

Topological optimization is used to generate a conceptual framework that is superior in some domain. In this article, it is sought to improve structural efficiency and for this, one should increase stiffness and decrease the material. In this way, initially there is a structure under loads and constraints, the optimization removes areas of material that are not very requested and, therefore, do not contribute much to the structural rigidity. These methods require domain discretization and an objective function that represents the performance of the structure and a set of constraints that limit the modifications.

In the present study, we intend to optimize systems of the fluid-structure type, which is comprised of fluid and structural domains. Some examples may be a tube, a pumping piston, an airplane wing, among others. A system is said to be coupled when each domain affects the other, which happens in dynamic cases and prevents their separation in two distinct problems, forcing their coupled resolution. However, in static cases, as in this work, the two domains could be solved separately; the fluid is calculated and the pressures resolved are applied to the structure. Although finite element calculation is done separately for each domain, structure modification during topology optimization can affect both domains simultaneously if some interface elements are added or removed. Thus, every time the structure is modified, the fluid needs to be recalculated. Therefore, some modifications to the method must be made.

The BEFSO method is used in this work, and this is a modification of the BESO method for systems with fluid-structure iteration. The objective function is the energy absorbed by the structure because it is a way of quantifying its rigidity, and the amount of material is considered as a constraint. A structure that uses more material has more rigidity, improves the objective function but with added material, which makes the structure less efficient. To avoid this, the amount of material that can be used is limited, and therefore the optimization seeks to improve the rigidity of a structure using less material. Compared to the ESO and SIMP (Bendsøe and Sigmund, 2003) methods, BESO (Huang e Xie, 2007) presents similar results but requires less processing time. In addition, this method stands out for its modularity and relative simplicity. Details of implementation can be found in Huang and Xie, 2010, while a general review will be presented here. The specific form used in this work is described in item 3.

2.1. Domain discretization and optimized variable

The domain must be discretized in several elements, forming a mesh. This discretization will be used in the analysis of the system by a numerical method, which in this work consists of the finite element method. In the BESO method each element can assume a value of two possible values: one represents its presence and another absence, as follows:

$$x_i = \begin{cases} 1: \text{if the element is present} \\ 0: \text{if the element is absent} \end{cases} \tag{1}$$

The x_i value shown is used to calculate the elastic modulus E_i for the ith element, using the original coefficient E_i^0 for the material of the structure, as shown in Eq. (2):

$$E_i = \left(x_i\right)^p E_i^0 \tag{2}$$

Where p is an exponent called a penalty factor. If x_i equals 0, that is, if the element is left absent, it will not participate in the FEM analysis and this implies that its elasticity modulus becomes effectively zero. However, in this work implementation the method of removing the elemental stiffness of the global stiffness matrix, before solving the problem was used, and in this way any DOF belonging to a node not connected to any active element can be removed from the system, reducing its complexity to as the structure loses its volume.

The objective function of the optimization problem quantifies a structural characteristic to be optimized that depends on the configuration of the structure in each iteration. This function is used to set the sensitivity function as well as the stop condition. Among the most frequently used functions are frequency response (Lisboa *et al.*, 2017) and rigidity (Vicente, 2013), but many others also exist, such as frequency response (Yoon *et al.*, 2007), energy dissipation rate due to the viscosity of a fluid (Hansen *et al.*, 2005) and effective nutrient diffusivity through a porous structure (Chang, 2015). The following section presents the problem formulation proposed in this article.

3. PROBLEM FORMULATION

In this work the optimization search to minimize the energy absorbed by the structure and thus, the characteristic to be maximized is the structural rigidity. We must then minimize the following objective function, according to Eq. (3).

$$\mathbf{C} = \frac{1}{2}\mathbf{f}^{\mathrm{T}}\mathbf{u} \tag{3}$$

Where **u** is the displacement vector of the structure and **f** is its force vector. This force vector can be written as a function of the displacement and stiffness of the system, **f** = **Ku**. The global stiffness matrix is composed of the elementary stiffness matrices $\mathbf{k}_i$, which depend on the elastic modulus E_i of each structural element, and this depends on the variable x_i, as shown in Eq. (2). As such, we can write the matrix **K** as shown in Eq. (4).

$$\mathbf{K} = \sum_{i=1}^{n} \left(x_i \right)^{p} \mathbf{k}_i^{0e} \tag{4}$$

Where $\mathbf{k}_i^{0e}$ is the elemental stiffness matrix of the i-element as a function of E_i^0 instead of E_i, and n is the number of elements in the mesh.

The optimization of the structure occurs with the optimization of the values of the vector **x**, for this, we take the derivative of the objective function in relation to each of the components of **x**. When we derive Eq. (3) with respect to x_i, we estimate how it behaves when x_i is altered, and with this we can determine which elements are more sensitive to structure in structural terms. To the result of the derivatives of the objective function we call sensitivities.

If we differentiate **K** by a specific x_i element we get as a result $p = (x_i)^{p-1}\mathbf{k}_i^{0e}$, because $\partial x_j/\partial x_i = 0$ for $j \neq i$. Thus, if we differentiate the objective function with regards to x_i we get Eq. (5), which is the sensibility of the system in regards to element i, as follows:

$$\alpha_i = p\left(xi\right)^{(p-1)} \frac{1}{2}\left(\mathbf{u}_i^e\right)^{\mathrm{T}} \mathbf{k}_i^{0e}\mathbf{u}_i^e \tag{5}$$

Where the e superscript denotes that those are elemental values. Because they are not functions of x_i, the **u** vectors are unchanged when we differentiate the objective function but all the components that do not refer to the DOFs of the i element are multiplied by zeros in the reduced stiffness matrix k_i^{0e} and thus can be discarded, with the elemental $\mathbf{u}_i^e$ displacement vector remaining. In the implementation used in this paper p is equal to one, and the equation is simplified. This is called hard-kill and is explained in details by Huang and Xie, 2010.

3.1. Stabilization filter

The sensitivity filter aims to modify the sensitivity value of each element. For this, it takes into account the values that each of them had in previous iterations. An example is the simple mean, where the sensitivity of the current iteration of an element is given by the average of the calculated value in the present iteration and of the calculated in the previous iteration; this filtering scheme modifies the sensibility value of each element, as defined in Eq. (6):

$$\alpha_i^f = \frac{1}{2}\left(a_i + a_{(i-1)}^f\right) \tag{6}$$

Where α_i^f is the filtered sensibility value for the i iteration, α_i is the non-filtered value for the same element in the same iteration, and $\alpha^f_{(i\text{-}1)}$ is its filtered sensibility in the previous iteration. This way, the whole historical of the sensibility values are taken in consideration during the calculation of the new value, with more distant iterations having a smaller influence. This is needed to smooth out sudden jumps in the objective function that could result from the removal or addition of an element, and that could lead to instabilities in the optimization process.

4. METHODOLOGY

In this study the 2D hard-kill BEFSO method implemented in MATLAB environment was programmed. Hard-kill was chosen instead of soft-kill because of its lower computational cost, even when using filtering methods needed to bring its results to the same level as them (Casas *et. al.*, 2016). The implementation details are described in this section. The general flow of the program is shown in Fig.1:

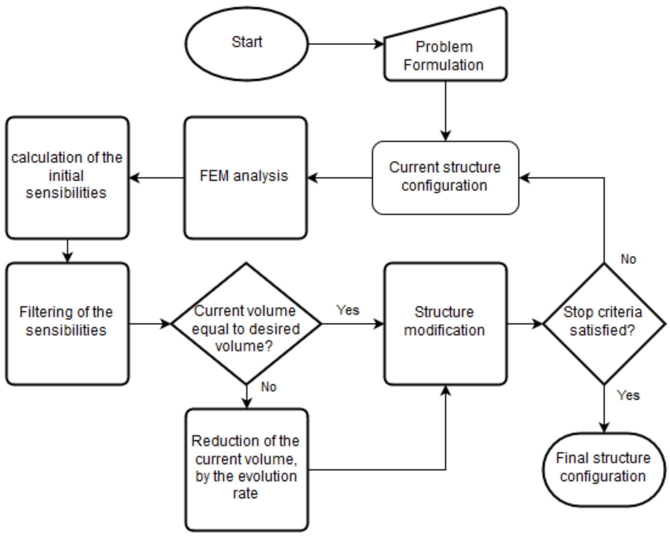

Figure 1. Flowchart showing the implemented BEFSO optimization procedure

4.1 Problem formulation

The analyzed problems involve the structural optimization of fluid-structure systems, seeking to minimize structural compliance, which is equivalent to maximizing rigidity. The mesh of the system is not changed during optimization. All problems are first described by a series of data concerning the mesh (node position and connectivity matrix), boundary conditions (forces and displacements and prescribed pressures, in certain degrees of freedom), and properties of the structural material (Poisson coefficient and Young's modulus) and initial regions of the fluid. The desired final volumetric fraction and rate of evolution as well as regions in which the structure cannot be removed are also reported.

4.2 Current configuration structure

The configuration of the structure in a given iteration is determined by a vector x of size equal to the number of elements used to discretize the domain. Each element of this vector can take a value of 0, representing an empty space or filled with fluid, or 1 representing a structural element. A second vector distinguishes between void elements having no stiffness, and thus they are removed from the total stiffness matrix, or fluid elements. An empty space will be occupied by fluid if there is fluid in the neighborhood. Some elements are always fluid, defined in the formulation of the problem. To determine the fluid elements, the flood fill is used, where empty elements beside fluid elements are occupied, repeating the process until there are no more empty elements in this condition.

4.3 Modification of the structure

After calculating all elementary filtered sensitivities, the elements are sorted according to their sensitivity value, and the elements with the lowest values are removed until the volume of the retained elements is equal to the volume defined by the constraint. This corresponds to the removal of elements that contribute less to the structure. That is, the least loaded elements are removed because their removal has the least impact on the structure as a whole when compared to the impact of the removal of other more loaded elements. In this sort are also considered previously removed elements, having a sensitivity value due to the filtering process. Thus, if a removed component has a value larger than an element that would be maintained, the first is re-included and the second is removed. However, only a certain volume of elements may be re-included.

4.4 Initial sensitivity calculation

The beginning of the calculation of sensitivities is given by Eq. (5) taking p = 1. Values are calculated with the elementary stiffness matrix and the displacement vector components that relate to the degrees of freedom of the elements, but only if the element is currently part of the structure. In this way, the sensitivity value is different for each element of the structure.

If an element is in the boundary of the structure with the fluid then the sensibility for it also takes into consideration the variation of the objective function due to the potential modification of the boundary. This variation is calculated based on the modification of the domain interface matrix, $\mathbf{L}_{sf}$, that would occur if the element were to be removed, and given by $(\mathbf{u}_s)^T \mathbf{L}^i_{sf}\, \mathbf{p}_f$.

The calculation of the sensitivity of each element is given by Eq. (7), based on that used by Vicente, 2013, but modified to consider the elemental volume according to Huang and Xie, 2010. If the element is not at the limit of the fluid structure, $(\mathbf{u}_s)^{\mathrm{T}}\,\mathbf{L}^{i}_{sf}\,\mathbf{p}_f = 0$.

$$\alpha_i = \frac{\partial \mathbf{C}}{\partial x_i} = \begin{cases} \dfrac{\left(\frac{1}{2}(\mathbf{u}_s)^{\mathrm{T}}\mathbf{K}^i_s\mathbf{u}_s + (\mathbf{u}_s)^{\mathrm{T}}\mathbf{L}^i_{sf}\,p_f\right)}{V_e} & \text{for } x_i = 1 \\ 0 & \text{for } x_i \neq 1 \end{cases} \tag{7}$$

4.5 Sensitivity filtering: mesh independency filter

In order to avoid the emergence of plaid patterns in the structure, a mesh independence filter is used. These patterns are fabric dependent and cannot be fabricated, which makes them undesirable. The filter must also be applied to allow elements that do not belong to the structure to receive a sensitivity value, which allows them to possibly be included in the structure again. Thus elements can be added to places that are very stressed because the high sensitivity values of the structural elements in these critical locations are smoothed over the empty elements around them.

For the calculation of the nodal sensibility first we must determine which nodes are within a certain radius r_{min} around the centroid of each element. The value of this radius should be chosen so as to include more than only the nodes that belong to the element, but it is arbitrary. For these nodes we determine the distance between them and the centroid, as exemplified in Fig 2.

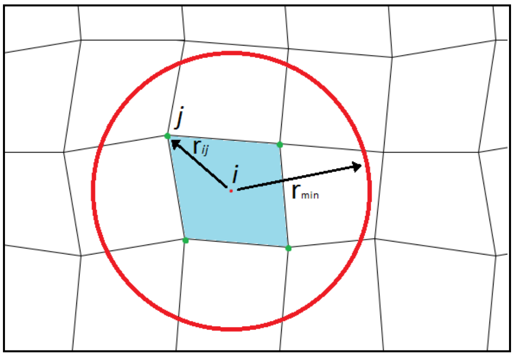

Figure 2. Schematic for determining the nodes within r_{min} of the centroid of an element and calculating the distance between them.

A nodal sensitivity value is assigned to each of these nodes and is given by the weighted mean of the element sensitivity values of all elements containing the node within a distance of r_{min}, according to Eq. (8).

$$\alpha^n_j = \sum_{i=1}^{M} \omega_{ij}\alpha^e_i \tag{8}$$

Where ω_{ij} are the weights relating to the interaction between the i element and the node j, M is the number of elements near the node j, and α_i^e is the sensibility value for the element i. These weights should obey $\sum_{i=1}^{M}\omega_{ij} = 1$, and are given by Eq. (9), where r_{ij} is the distance between node j and element i.

$$\omega_{ij} = \frac{1}{M-1}\left(1 - \frac{r_{ij}}{\sum_{i=1}^{M} r_{ij}}\right) \tag{9}$$

It should be noted that nodal sensitivity values are used only to calculate the filtered elemental sensitivity values, Eq. (10), Where W is the number of nodes close by element i and β_{ij} is equal to $r_{min} - r_{ij}$.

$$\alpha_i = \frac{\sum_{j=1}^{M} \beta_{ij} \alpha_j^n}{\sum_{j=1}^{W} \beta_{ij}} \tag{10}$$

In this way, a new way of applying the filter is proposed. After the weight calculations, the filter is applied by means of multiplications and simple sums. As such, it is possible to write it in vector form, given by Eqs. (11) and (12).

$$\alpha^n = \omega \alpha^e \tag{11}$$

$$\alpha = \beta \alpha^n \tag{12}$$

Where α^n, α^e and α are the nodal, elemental and filtered elemental sensibilities respectively, ω contains the nodal weights and β contains the elemental weights. α^e and α consist of $N \times 1$ vectors, where N is the number of elements, α^n is a $Y \times 1$ vector, where Y is the number of nodes, ω is a $Y \times N$ matrix and β is a $N \times Y$ matrix. Combining Eqs. (11) and (12) allows us to write, $\alpha = \beta\omega\alpha^e$, that can then be rewritten as in Eq. (13):

$$\alpha = \mathbf{H} \alpha^e \tag{13}$$

Where **H** consists of $N \times N$ matrix called smoothing matrix. This way there is no need to calculate or keep nodal sensibilities values, nor two separate matrixes. This matrix is still computationally expansive to generate, mas it depends only on the coordinates of the nodes and elements, and is thus unchanged throughout the optimization. It can then be computed on the first iteration only, being applied again in the next iteration and so on.

As this filter does not consider volume differences between the volume of elements, its utilization with meshes with big volume differences between elements can cause undesired results. To minimize this effect we must divide the sensibility of each element by the square root of its volume before and after applying the filter.

4.6 Sensitivity filtering: stabilization filter

As shown in Eq.(6), this is a simple filter that calculates the average of the elemental sensibilities of iteration i and iteration $i - 1$, so as to stabilize the convergence. If written in expanded form, it is possible to see that all previous values influence the current one, according Eq. (14):

$$\alpha_i^f = \frac{\alpha_i}{2} + \frac{\alpha_i - 1}{4} + \frac{\alpha_i - 2}{8} + \cdots \tag{14}$$

However, computationally only the current and previous iteration values are maintained.

4.7 Restrictions and evolution rate

As a restrictive condition, a volumetric restriction is applied, implemented as a fraction of the initial volume. Thus, the structure must satisfy the following condition over the iterations:

$$V_i = \begin{cases} V_0 \left(1 - iER\right) & \text{if } V_0\left(1 - iER\right) > V^* \\ V^* & \text{if } V_0\left(1 - iER\right) < V^* \end{cases} \tag{15}$$

Where V_i is the volume of the structure in the i iteration, computed as the sum of the volumes of all elements that make up the structure, V_0 is its initial volume, ER is the evolution rate, and V^* is the final desired volume, equal to $V_0 vp$ where vp is the final desired volume fraction. By way of this condition the structural volume is gradually reduced each iteration by the removal of structural elements, until the final volume is reached

4.8 Structure modification

In order to implement the desired optimization features, it is necessary to determine which previously removed elements can be included back in the structure. To do this, we use the bisection method to find a threshold for sensitivity values, so that elements removed with values above it can be considered reinstated. The purpose of this is to limit the total volume that can be included in each iteration in order to prevent the optimization from becoming unstable under certain circumstances. This limit value x 'is such that the sum of the volume of all elements removed with sensitivity values above it must be equal to the allowed inclusion volume V_{ad}, as shown in Equation. (16), below:

$$\begin{aligned}&\text{Find: } x' \mid y(x') = 0\\&\text{Subject to: } y(x) = \sum_{j=1}^{N} V'^{e}_{j}(x) - V_{ad}\\&\text{Where: } V^{e}_{j}(x) = \begin{cases} V^{e}_{j} & \text{if } j \text{ referes } \alpha_j > x \\ 0 & \text{otherwise} \end{cases}\end{aligned} \tag{16}$$

Where N is the number of elements, V_j^e is the volume of element j and α_j is its filtered sensibility. The considered elements are in this case the all elements that do not belong to the structure but could be added back, that is, all elements that unable to become part of the structure due to imposed restrictions. The potential elements are the ones whose sensibility values are larger or equal than x'.

With the list of potential elements calculated we repeat the process, but this time the considered elements are all structural elements plus those potential elements and V_{ad} is replaced by V_i, which is the desired structural volume for iteration i. The structural elements for the next iteration are the ones with $\alpha_j > x'$, regardless if they are presently part of the structure or not. Some structural elements can be defined as fixed, however, and the sensibility values for these elements are considered to be infinite for the purposes of this test.

4.9 Stop criteria

Three criteria are used to stop the optimization process. The first criterion is the restrictive condition, applied gradually over the iterations, must have been fully applied before the optimization can be interrupted. In other words, the optimization can only be interrupted if the structural volume is equal to the desired volume.

After reaching the first condition, some iteration is still calculated until the change in the objective function, in which case the structural compliance is below a tolerance value. During these iterations, there is no significant change in the volume of the structure, however, its configuration may change as the elements are removed from some areas and added to another. This condition is necessary because, as mentioned, the methods used to modify the structure are only approximations, and although they are sufficient to guide the volume reduction during the first part of the optimization, they are not able to find an ideal solution in just one iteration. Thus, during several iterations, the structure stabilizes and converges to its final and optimized form.

As the same time as the second condition is checked, the current value of the objective function is compared to the best value found since the achievement of the first condition. If the present value is better than the previous best, it is kept as the new best, and the structural configuration is saved. If the optimization fails to produce a structure with a new best after a certain number of iterations it is stopped. These conditions are represented by Eqs. (17) and (18):

$$V_i = V^* \tag{17}$$

$$\left| \frac{C_i - C_{i-1}}{C_i} \right| < tol \vee \; C_{(i-K)\cdots i} > C_{min} \tag{18}$$

Where V_i is the structural volume in iteration i, V^* is the desired volume, C_i is the value of the objective function in iteration i, tol is the acceptable tolerance to consider the structure as having converged, and K is the number of straight iterations that are allowed to have C values that are worse than the best value so far, C_{min}, V^*, tol and K are inputs to the optimization method.

5. FINITE ELEMENT ANALYSIS

5.1 Structural and Fluid Domain

The structural domain of the coupled system can be described by means of the differential equation of motion for a continuous body. As hypotheses for the structural domain are assumed small deformations and homogeneity and isotropy of the material, using the hypotheses of linear elasticity theory.

The equation of motion for a linear elastic continuous medium can be written as:

$$\tilde{\boldsymbol{\nabla}}^{\mathrm{T}} \boldsymbol{\sigma}_{\mathrm{S}} + \mathbf{b}_{\mathrm{S}} = \mathbf{q}_{\mathrm{S}} \tag{19}$$

Being $\boldsymbol{\sigma}_S$ the vector representation for the terms of Cauchy's tensor $\mathbf{b}_S$ is the field strength vector and $\mathbf{q}_S$ represents the inertial force vector. Dirichlet's conditions say that the displacement at the frontier of the domain must be equal to the displacement prescribed in it, and Neumann's conditions say that the efforts at the frontier should be equal to the efforts prescribed therein. With the Eq.(19), in addition to the boundary conditions, it is possible to calculate the field of voltages σ_S and displacements u_S of the domain.

However, the analytical resolution of these equations is only possible for simple cases, so the finite element method is used. Linear weighting equations are used which describe the field of displacements and stresses in a body as a function of values calculated at specific points, called nodes. For the case studied 4-node quadrilateral elements are used, so there are 4 weighting functions, called shape functions.

These are chosen such that their value is 1 in their referent node, decreasing linearly to 0 on the other nodes. The displacement within the body is therefore given by the sum of these 4 functions, each multiplied by the displacement of its referent node. Only the displacements are considered on the two main axes, therefore 2 degrees of freedom per node. This can be written as follows:

$$\mathbf{u}_S^e = \mathbf{N}_S^e \tilde{\mathbf{u}}_S^e \tag{20}$$

Where $\mathbf{N}_S^e$ are the shape functions and are organized in a line vector, $\tilde{\mathbf{u}}_S^e$ are the values of the nodal displacements organized in a matrix 4 by 2, referring to the 4 nodes and 2 degrees of freedom of the elements used, and $\mathbf{u}_S^e$ is the vector of displacements in the domain of the element in a *xy*- position. The components of $\mathbf{N}_S^e$ and $\mathbf{u}_S^e$ are functions of *x* and *y*, whereas the components of $\tilde{\mathbf{u}}_S^e$ are constants.

For the stresses and displacements of the structure, an equation of the movement of elastic bodies, with acceleration and damping equal to zero, is used. Equation is given by:

$$\mathbf{Ku} = \mathbf{F} \tag{21}$$

To solve it, it is necessary to have the global stiffness matrix of structure$\mathbf{K}$. This matrix is assembled based on the stiffness matrices of each element. The elementary stiffness matrix $\boldsymbol{K^e}$ is given by:

$$\boldsymbol{K^e} = \int (\boldsymbol{\nabla}\mathbf{N}^{\mathrm{e}})^{\mathrm{T}} \mathbf{D_S} (\boldsymbol{\nabla}\mathbf{N}^{\mathrm{e}}) dv \tag{22}$$

Its development can be found in Petyt, 1990. Differentials of the form functions can be calculated analytically because they are only bilinear functions, but the integral must be calculated numerically using Gaussian integration. Since the equation involves the multiplication of functions, the domain may be described by a bi-quadratic function, or bi-cubic depending on the coordinate system. In both cases it is sufficient to use 4 points on the domain.

5.2 Interface

The interface between the two domains in the 2D case is given by a line S, where transmission of forces. The total load on the structure must be equal to the total force applied by the fluid. This force is given by the pressure distribution in the fluid, and occurs in the normal direction. The loading in the structure also follows a distribution along the line. The force applied by the fluid on the structure is then given by:

$$f_{Sf}^e = \int (\nabla \mathbf{N}_S^e)^T \mathbf{n} \mathbf{N}_f^e dS \tilde{\boldsymbol{p}}_f^e \tag{23}$$

Where $\mathbf{N}_S^e$and $\mathbf{N}_f^e$ are the shape functions of the structure and fluid respectively, along the line, $\mathbf{n}$ is the vector pointing in the normal direction and $\tilde{\boldsymbol{p}}_f^e$ is the vector of pressures on the nodes shared by the domains.

It is considered that the structure does not apply static loading on the fluid, i.e., that the fluid has a free surface. In the dynamic case the structure applies pressure waves to the fluid, but here only the static case is being considered. If standardized functions are used, Eq. (23) can still be used, being integrated over the local coordinate system but must be multiplied by the length of the contact line.

By separating the elementary pressure vector from the force it is possible to obtain a sub-matrix that can be placed in the global stiffness matrix, serving as a coupling between the domains. However, due to the hypothesis mentioned above, this causes the global matrix to lose its symmetry. This sub-matrix is given by:

$$\mathbf{L}_{Sf}^e = \int (\mathbf{N}_S^e)^T \mathbf{n} \ \mathbf{N}_f^e dS \tag{24}$$

5.3 Solution of the coupled system

Considering Eq. (21), the acceleration and damping plots are disregarded because only the static case is considered in this work. Expanding the matrices shows the division of domains

$$\begin{bmatrix} \mathbf{K}_S & \mathbf{L}_{Sf} \\ 0 & \mathbf{K}_f \end{bmatrix} \begin{bmatrix} \mathbf{u}_S \\ \mathbf{p}_f \end{bmatrix} = \begin{bmatrix} \mathbf{f}_S \\ \mathbf{f}_f \end{bmatrix} \tag{25}$$

This system of equations describes a weakly coupled system, that is, that can be solved in parts. In this case, it is possible to calculate the pressure field independently of the structure, and apply these pressures in the structure as a load:

$$\mathbf{p}_f = \mathbf{K}_f^{-1}\mathbf{f}_f \tag{26}$$

$$\mathbf{u}_S = \mathbf{K}_S^{-1}\left(\mathbf{f}_S - \mathbf{L}_{Sf}\mathbf{K}_f^{-1}\mathbf{f}_f\right) \tag{27}$$

6. RESULTS AND DISCUSSION

As an example of the capabilities of the program, a case of optimization of the internal transverse structure of an airplane wing was proposed. In this case, there is a fixed interface between the fluid and the structure, since the outer profile of the wing is determined by its aerodynamic behavior, which is not simulated here. A NACA 4412 profile was chosen, and the distribution of the external pressures caused by the air during its operation is given by Allen, 1939, as a function of the dynamic pressure, which is given by:

$$q = \frac{1}{2}\rho_{air}U^2 \tag{28}$$

Where ρ_{air} is the density of the air and U is the velocity of the fluid. Table 2 shows the values used for the simulation.

Table2: Values used in the simulation of the NACA 4412 profile wing

Description	**Variable**	**Value**
Flow velocity	U	200km/h (55,57 m/s)
Air Density	ρ_{air}	1.007 Kg/m³
Dynamic pressure	q	1554.0 Pa
Level	H	2000 m
Wing Profile	α	13º57’
String	c	2 m
Thickness of interface fixed layer	t	10 mm
Number of elements	n_{el}	15790

The Fig. 3 shows the distribution of the nodes used for the interpolation of the prescribed dynamic pressures. This interpolation was divided into two regions with 10 nodes in each. In this way the pressure field is well defined.

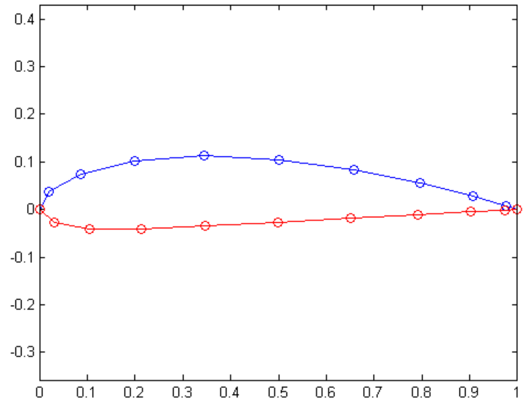

Figure 3: Distribution of the nodes used in dynamic pressure interpolation.

To generate the mesh of this case was used the free program Gmsh. A MATLAB® function has been developed that is capable of reading your output files and converting them to the format used by the developed software. Fig. 4 shows the mesh developed in Gmsh. The mesh used was unstructured and non-regular. The cantilever was chosen as being in a single beam. Aircraft wings different, even with the same profile, may have different structures. The simulated wing does not refer to any specific one, being used as an example of the functionality of the developed software.

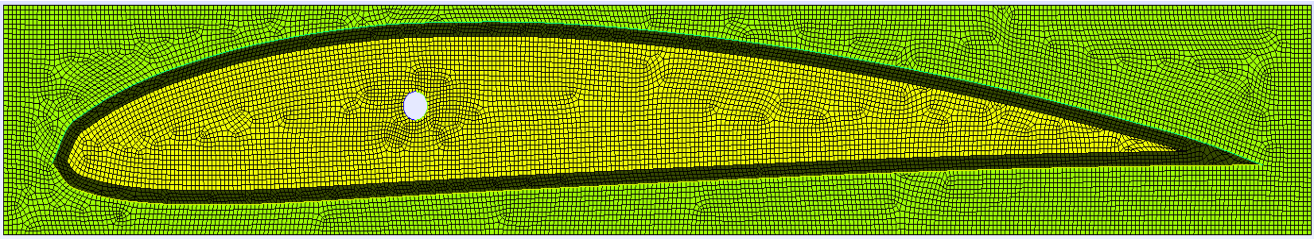

Figure 4: Non-structured and irregular quadrangular mesh developed in Gmsh

The result of the optimization is shown in the Fig. 5 (Casas *et. al.*, 2016). The colored regions represent the fluid, with the colors indicating the pressure field. The gray circular region represents a beam transverse to the image, the spar, and is where the crimping conditions are applied.

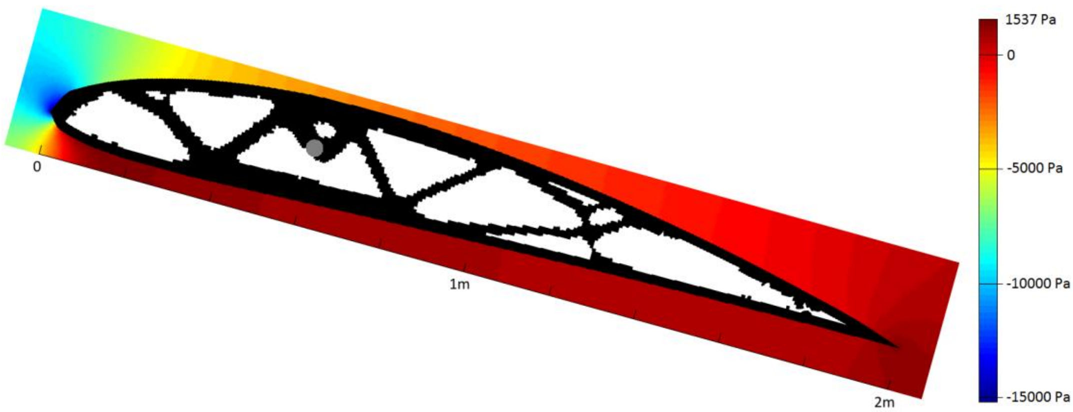

Figure 5: Result for the optimization of the internal transverse structure of a wing with the profile NACA 4412

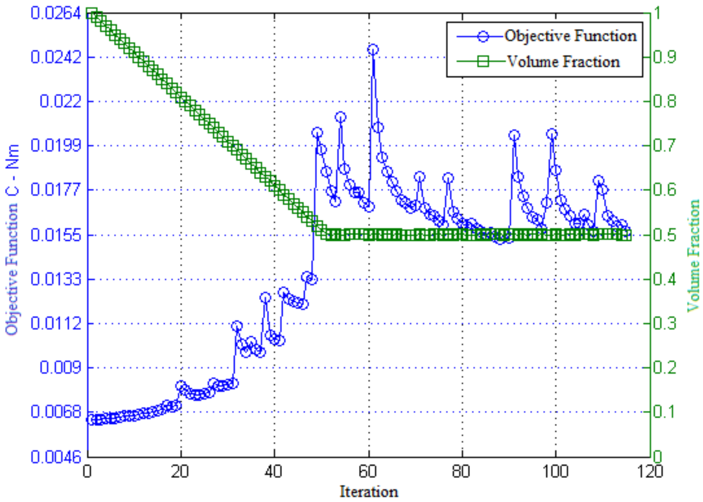

Figure 6: History of the objective function and fraction of the optimization volume of the NACA 4412 profile wing and 2m string

In Fig. 6 it is possible to see several peaks in the objective function throughout the optimization; these occur when a structure representing a beam is eliminated. The final result shown corresponds to the configuration of the structure that presented the lowest energy value, because the program saves the best configuration found during the optimization process.

7. CONCLUSION

The algorithm proposed as an extension to the BESO method proposed here contains an implementation of a FEM analysis (Petyt, 1990) as one of its parts and therefore independent of external software to perform optimizations, requiring only a MATLAB® console. The implemented algorithm was verified with a case found in the literature, and thus the goal of implementing an improvement for the BEFSO method was achieved. The proposed filter was also efficient.

8. ACKNOWLEDGEMENTS

The research described in this paper was financially supported by Brazilian national funding agencies, through the National Council of Scientific and Technological Development CNPq (grants 140081/2015-1, 148895/2016-6 and 148887/2016-3) from the Ministry of Science, Technology and Innovation; and the Coordination for the Improvement of Higher Education Personnel CAPES (grant 3686) from the Ministry of Education. Optional section must be placed before the list of references.

9. REFERENCE

Bendsøe, M. P.; Sigmund, O., 2003 "Topology optimization. Theory, methods, and applications". *Berlin: Springer-Verlag*, , p 370.

Casas, W. J. P., Cesconeto, E. M., Lisboa, E.S., Moreira, J. B. D., Medeiros, E. M. . and Ribeiro, T. S., 2016, "Topology Optimization of an Aircraft Component as a Fluid-Structure System with Unstructured Mesh", *Aerospace Technology Congress*, Solna, Stockholm.

Chang, C., 2015. *Design and Topology Optimization of Tissue Scaffolds*. Ph.D. thesis. University of Sydney, Australia.

Hansen, A. G., Sigmund, O., Haber, R. B., 2005. "Topology optimization of channel flow problems". *Struct. Multidisc. Optim.*, Vol.30, p.181–192.

Huang, X., Xie, Y.M., 2007. "Convergent and mesh-independent solutions for the bi-directional evolutionary structural optimization method". *Finite Elements in Analysis and Design*, vol. 43, 1039–1049.

Huang, X., Xie, Y. M., 2010 *Evolutionary Topology Optimization Of Continuum Structures: methods and applications.* Chichester, UK: John Wiley & Sons, Ltd, 1st edition.

Lisboa, E.S., Moreira, J. B. D., Cesconeto, E. M., Medeiros, J. E. G., Ribeiro, T. S. and Casas, W. J. P., 2017. "Optimization of dynamic parameters with bidirectional evolutionary structural method of continuum structures". *XVII International Symposium on Dynamic Problems on Mechanics*, São Sebastião, Brazil, March.

Petyt, M. 1990, *Introduction to Finite Element Vibration Analysis*, Cambridge University Press, New York, NY, USA.

Vicente, W.M., 2013. *Otimização topológica evolucionária aplicada a sistemas fluido-estrutura*. Ph.D. thesis. Universidade Estadual de Campinas.

Yoon, G. H., Jensen, J. S., Sigmund, O. 2007. "Topology optimization of acoustic–structure interaction problems using a mixed finite element formulation using matlab in numerical implementations". *International Journal for Numerical Methods in Engineerings*, USA, vol.70, p.1049–1075.

10. RESPONSIBILITY NOTICE

The author(s) is (are) the only responsible for the printed material included in this paper.