MECSOL 2019 - Proceedings of the 7th International Symposium on Solid Mechanics,
V. Tita, A. Ferreira, M.L. Ribeiro, A. Vieira and A.T. Beck (Editors), ABCM, Sao Carlos, SP, Brazil, April 15th to 17th, 2019

2D Shape Optimization Based on a NURBS Surface Velocity Field

Luiz G. Oliveira [1] and Pablo A. Muñoz-Rojas [1]

[1] Santa Catarina State University, Centro de Ciências Tecnológicas, Campus Universitário Prof. Avelino Marcante, 200 Bom Retiro - Joinville - SC – Brasil, CEP: 89223-100.

Abstract: This paper presents a procedure for shape optimization in plane elasticity problems solved using the finite element method. The strategy uses a B-Spline or a NURBS surface parameterization, which is differentiated with respect to boundary control points before discretization to obtain the velocity field. The position of domain control points can be made dependent on the coordinates of the boundary control points giving more flexibility to the method. By construction, the velocity field that outcomes from this approach is not affected by the finite element mesh quality. Furthermore, for a given mesh topology, this velocity field is shown to introduce reduced mesh distortions up to moderately large geometry modifications. This property is fundamental for a consistent shape optimization procedure, which must preserve the same mesh topology throughout the whole optimization process. The effectiveness of the method is compared to velocity fields obtained via Laplacian smoothing schemes. Since the NURBS based velocity field is performed in a parameterized domain, the approach can be directly extended for applications in the optimization of arbitrarily shaped shells.

Keywords: Shape, Optimization, Mesh, Sensitivity.

INTRODUCTION

Shape optimization deals with finding the best boundary contour of a given domain in order to maximize a performance measure while satisfying design constraints. In order to ensure mathematical consistency of the discretized version of the problem, the associated mesh topology must not be modified throughout the optimization process (Schleupen, 2000). To this end, in this paper we propose to use a velocity field based on a B-spline or a NURBS surface parameterization, which has shown robustness with respect to mesh distortion when compared to Laplacian smoothing techniques. The strategy is applicable without adaptations to plane problems and 3D surfaces.

DEFINITION OF THE SHAPE OPTIMIZATION PROBLEM

We state the 2D shape optimization problem as to find the optimal boundary shape of a mechanical component, aiming at weight minimization subject to nonlinear constraints, that is,

$$\begin{gathered} min\ W(\boldsymbol{a}) = \rho\, \psi(\boldsymbol{a}) \\ g_j(\boldsymbol{a}) = 0, \qquad j = 1, m_e \\ g_j(\boldsymbol{a}) \geq 0, \qquad j = m_e + 1, m \\ \boldsymbol{a}_l \leq \boldsymbol{a} \leq \boldsymbol{a}_u \end{gathered} \tag{1}$$

where ρ is the density of the material (in this work made equal to unity), ψ denotes the volume, $\boldsymbol{a}$ is the vector of design variables, $g_j(\boldsymbol{a})$ is the j-th design constraint, m_e is the number of equality constraints, m is the total number of constraints and a_l and a_u are the upper and lower limits of the design variables, respectively. In this work we consider stress and geometric constraints using a cluster strategy to be described.

The first step towards shape optimization is a proper geometry parameterization. Once the optimization method has been established for a given geometric description, it is necessary to evaluate the gradients of the objective function $W(\boldsymbol{a})$ and the constraints $g_j(\boldsymbol{a})$ with respect to the vector of design variables $\boldsymbol{a}$. The calculations of these gradients is known as sensitivity analysis. In this work, we perform such sensitivity analytically.

GEOMETRY PARAMETERIZATION

NURBS surfaces arise from the tensor product between two NURBS curves. The expression for a NURBS surface has the form (Piegl and Tiller, 1997),

$$S(u,v) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{ip}(u)\, N_{jq}(v)\, w_{ij}\, P_{ij}}{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{ip}(u)\, N_{jq}(v)\, w_{ij}}, \qquad [u,v] = \{x \in \mathbb{R}\colon 0 \leq x \leq 1\}, \tag{2}$$

where $S(u,v)$ returns a continuous field for the coordinates x, y and z belonging to the NURBS surface, evaluated at the parametric coordinates (u,v), N_{ip} and N_{jq} are the base functions for NURBS curves of order $p+1$ and $q+1$, which are obtained recursively as described in Piegl and Tiller (1997), P_{ij} is a control point of the $\boldsymbol{P}$ matrix of the $(n+1)\times(m+1)$ control points and w_{ij} is a weight value assigned to the control points. In addition to the control points, the definition of the NURBS surface requires that two knot vectors $\boldsymbol{U}$ and $\boldsymbol{V}$ be given, with $\boldsymbol{U}=\{u_0,\dots,u_h\}$, such that $u_l\le u_{l+1}$, $h=p+n$ and $0\le u_l\le 1$. In a similar way, $\boldsymbol{V}=\{v_0,\dots,v_h\}$ where $v_l\le v_{l+1}$, with $h=q+m$, e $0\le v_l\le 1$.

In order to parameterize the nodal coordinates of a finite element mesh with respect to the corresponding geometry described by a B-Spline or NURBS surface, it is necessary to identify the parametric coordinates u and v of the nodes through parametric recovery.

Parametric recovery

This problem is solved adapting the procedure proposed by Augusto (2012) so that plane or curved surfaces can be dealt with in a unified way.

For a finite element mesh, we want to know the parametric coordinates u and v in $\mathbb{R}^2$ that correspond to the coordinates x,y,z of k-th node in the $\mathbb{R}^3$ space. Therefore, the following equation must be satisfied,

$$\boldsymbol{f}(u_k,v_k)=\boldsymbol{f}_k=\begin{bmatrix}S^x(u,v)_k\\ S^y(u,v)_k\\ S^z(u,v)_k\end{bmatrix}-\begin{bmatrix}x_k\\ y_k\\ z_k\end{bmatrix}=0 \tag{3}$$

where $S^x(u,v)_k$, $S^y(u,v)_k$ and $S^z(u,v)_k$ are the coordinates x, y and z of the k-th node, which correspond to the implicit and unknown parametric coordinates u,v in accordance to Eq. (2). We solve Eq. (3) by the Newton-Raphson method expanding $\boldsymbol{f}_k$ in a Taylor series truncated at first order terms. The linearized version of Eq. (2), evaluated at an arbitrary point $(u,v)_k^*$ is given by

$$\boldsymbol{f}_k\cong\boldsymbol{f}_k^*+\nabla\boldsymbol{f}_k^*\begin{bmatrix}u-u^*\\ v-v^*\end{bmatrix}_k=0 \tag{4}$$

where

$$\nabla\boldsymbol{f}_k=\begin{bmatrix}S^x{}_{,u} & S^x{}_{,v}\\ S^y{}_{,u} & S^y{}_{,v}\\ S^z{}_{,u} & S^z{}_{,v}\end{bmatrix}_k \tag{5}$$

and the comma stands for partial differentiation. The derivative of S with respect u and v can be found in Piegl and Tiller (1997). Pre-multiplying Eq. (4) by $(\nabla\boldsymbol{f}_k^*)^T$

$$(\nabla\boldsymbol{f}_k^*)^T\boldsymbol{f}_k^*+(\nabla\boldsymbol{f}_k^*)^T\,\nabla\boldsymbol{f}_k^*\begin{bmatrix}u-u^*\\ v-v^*\end{bmatrix}_k=0, \tag{6}$$

we have a 2×2 dimension system which, in recursive form returns

$$\begin{bmatrix}u\\ v\end{bmatrix}_k^{i+1}=\begin{bmatrix}u\\ v\end{bmatrix}_k^{i}-\left[\left(\nabla\boldsymbol{f}_k^i\right)^T\cdot\nabla\boldsymbol{f}_k^i\right]^{-1}\cdot\left(\nabla\boldsymbol{f}_k^i\right)^T\boldsymbol{f}_k^i \tag{7}$$

which is iterated until satisfaction of

$$\left\{\begin{bmatrix}u\\ v\end{bmatrix}_k^{i+1}-\begin{bmatrix}u\\ v\end{bmatrix}_k^{i}\right\}\le\epsilon \tag{8}$$

where ϵ is the convergence tolerance.

DISCRETIZED OBJECTIVE FUNCTION AND CONSTRAINTS

In a finite element framework, a normalized version of the objective function defined in Eq. (1) is given by

$$W(\boldsymbol{a})=\frac{1}{M_0}\rho\sum_{e=1}^{n_e}\int t\,|\boldsymbol{J}|_e\,d\xi d\eta \tag{9}$$

where the M_0 is the initial mass, t is the element thickness, $|\boldsymbol{J}|_e$ is the e-th element jacobian (which relates a differential area in the parametric element domain $(\xi,\eta$) to its counterpart in the “physical” domain (x,y)) and n_e is the total number of elements.

The stress constraints are evaluated on the whole domain using a modified P-norm and the cluster approach proposed by Holmberg et al. (2013). In this strategy, the stresses are evaluated, sorted and grouped in n_c stress clusters of equal size. For each stress cluster we define a constraint g_c

$$g_c = -\left[\frac{1}{N_c}\sum_{k\,\in\,\Omega_c}\left(\frac{\sigma_k^{vM}}{\sigma_{alw}}\right)^p\right]^{1/p} + 1 \geq 0, \qquad c = 1, n_c \tag{10}$$

where N_c is the size of the c-th set Ω of clustered stress points, σ_{alw} is the allowable design stress and σ_k^{vM} is the von Mises stress in k-th node,

$$\sigma_k^{vM} = \left(\bar{\sigma}_{11}^2 - \bar{\sigma}_{11}\bar{\sigma}_{22} + \bar{\sigma}_{22}^2 + 3\bar{\sigma}_{12}{}^2\right)_k^{1/2}. \tag{11}$$

The components of the stress tensor $\bar{\boldsymbol{\sigma}}$ in Eq. (11) are obtained via global L2 smoothing, as described in Wildemann and Muñoz-Rojas (2005),

$$\int_{\Omega_e} \boldsymbol{N}^T\boldsymbol{N}\,|\boldsymbol{J}|\,d\Omega_e\,\bar{\boldsymbol{\sigma}} = \int_{\Omega_e} \boldsymbol{N}^T\boldsymbol{\sigma}\,|\boldsymbol{J}|\,d\Omega_e \tag{12}$$

where $\boldsymbol{N}$ is the matrix of the interpolation functions of element e and $\boldsymbol{\sigma}$ are the stress components evaluated at the superconvergent points, according to

$$\boldsymbol{\sigma} = \boldsymbol{E}\,\boldsymbol{B}\,\boldsymbol{u}. \tag{13}$$

In Eq. (13), $\boldsymbol{E}$ is the constitutive matrix and $\boldsymbol{B}$is the strain-displacement matrix, which relates the element nodal displacements $\boldsymbol{u}$ with the strain tensor components. The element nodal displacements $\boldsymbol{u}$ are extracted from the global displacements vector $\boldsymbol{U}$, which is known by previous solution of the classical linear finite element system

$$\boldsymbol{KU} = \boldsymbol{F} \tag{14}$$

where $\boldsymbol{K}$ is the global stiffness matrix obtained by assembling a global system from the elements stiffness $\boldsymbol{k}_e$,

$$\boldsymbol{k}_e = \int_{\Omega_e} \boldsymbol{B}^T\boldsymbol{EB}\,|\boldsymbol{J}|\,d\Omega_e \tag{15}$$

and $\boldsymbol{F}$ is the load vector. Evaluating Eq. (12) in each element and assembling a global system, a linear system is obtained

$$\boldsymbol{M}\,\bar{\boldsymbol{\sigma}} = \boldsymbol{P}, \tag{16}$$

where $\boldsymbol{M}$ is known as mass matrix and $\boldsymbol{P}$ is the pseudo-load.

In this work an analogous approach is proposed for the geometric constraints, that is,

$$g_G = -\left[\frac{1}{N_G}\sum_{k\,\in\,\Omega_G}\left(\frac{\chi_{min}}{\chi_k}\right)^p\right]^{\frac{1}{p}} + 1 \geq 0, \qquad G = n_c + 1, n \tag{17}$$

where g_G is the G-th coordinate constraint and N_G is the size of the set Ω_G, χ_{min} is the minimum allowable coordinate value for the points in the set Ω_G and χ_a is the nodal coordinate of a parametric position which belongs to Ω_G.

The clusters can either be updated or kept constant during optimization iterations. For $n_c > 1$, updating the clusters may be favorable for satisfying constraints, however this is not rigorously consistent as it changes the mathematical optimization problem being solved, and can theoretically lead to convergence difficulties. Following Holmberg et al. (2013), we do adopt this updating in the present work but we remark that this effect should be better investigated in future developments.

SENSITIVITY ANALYSIS

Sensitivity analysis refers to the relationship between design variables and structure responses that are determined by the laws of mechanics (Haug et al., 1986). Three differentiation methods can be applied to evaluate sensitivities: numerical, semi-analytical or analytical. The main factors that may affect the sensitivity analysis are the mesh quality, the velocity field and the differentiation method (Beckers, 1991; Barthelemy, et al., 1988). In this paper we adopt the analytical sensitivity approach.

Objective function and constraints sensitivities

The normalized sensitivity of the objective function in Eq. (9) with respect to the design variables is obtained by partial differentiation as

$$W_{,a_z} = \frac{1}{M_0}\rho\sum_{e=1}^{n_e}\int_{\Omega_e} \boldsymbol{N}^T\boldsymbol{N}\frac{\partial|\boldsymbol{J}|}{\partial a_z}\,d\Omega_e \tag{18}$$

where the analytical sensitivities of the $|\boldsymbol{J}|$ with respect the design variables can be found in Olhoff et al. (1993) and Wang et. al. (1985).

The sensitivity of the constraints in Eq. (10) and (17) are,

$$\frac{\partial g}{\partial a_z} = -\left(\frac{1}{N}\right)^{\frac{1}{p}}\left[\sum_{k\,\in\,\Omega}(A_k)^p\right]^{\frac{1}{p}-1}\left(\sum_{k\,\in\,\Omega} A_k^{p-1}\,B_k\right) \tag{19}$$

where

$$\frac{\partial g_C}{\partial a_z}\begin{cases} A_k = \dfrac{\sigma_k^{vM}}{\sigma_{alw}} \\ B_k = \dfrac{1}{\sigma_{alw}}\left[\dfrac{\partial(\sigma_k^{vM})}{\partial a_z}\right] \end{cases}, \qquad \frac{\partial g_G}{\partial a_z}\begin{cases} A_k = \dfrac{\chi_{min}}{\chi_k} \\ B_k = -\dfrac{\chi_{min}}{\chi_k^2}(V_{kz}) \end{cases}. \tag{20}$$

The term V_{kz} is the sensitivity of the k-th nodal coordinate $\partial\chi_k/\partial a_z$, also known as velocity field, and its calculation will be detailed in the next Section. The term $\partial(\sigma_k^{vM})/\partial a_z$ is obtained by differentiating Eq. (11), which yields

$$(\sigma_k^{vM})_{,a_z} = \frac{1}{4\sigma_k^{vM}}\left[\frac{\partial\bar{\sigma}_{11}}{\partial a_z}(2\bar{\sigma}_{11}-\bar{\sigma}_{22}) + \frac{\partial\bar{\sigma}_{22}}{\partial a_z}(2\bar{\sigma}_{22}-\bar{\sigma}_{11}) + 6\frac{\partial\bar{\sigma}_{12}}{\partial a_z}\bar{\sigma}_{12}\right]_k . \tag{21}$$

The sensitivity of the smoothed stress components $\partial(\bar{\boldsymbol{\sigma}})/\partial a_z$ in Eq. (21) is obtained by differentiating Eq. (16)

$$\boldsymbol{M}\frac{\partial\bar{\boldsymbol{\sigma}}}{\partial a_z} = \frac{\partial\boldsymbol{P}}{\partial a_z} - \frac{\partial\boldsymbol{M}}{\partial a_z}\bar{\boldsymbol{\sigma}}, \tag{22}$$

where

$$\frac{\partial\boldsymbol{M}}{\partial a_z} = \int_{\Omega}\boldsymbol{N}^T\boldsymbol{N}\frac{\partial|\boldsymbol{J}|}{\partial a_z}\,d\Omega \tag{23}$$

and

$$\frac{\partial\boldsymbol{P}}{\partial a_z} = \int_{\Omega} N^T\left(\frac{\partial\boldsymbol{\sigma}}{\partial a_z}|\boldsymbol{J}| + \frac{\partial|\boldsymbol{J}|}{\partial a_z}\boldsymbol{\sigma}\right)d\Omega. \tag{24}$$

Notice that the stress sensitivity with respect to the design variables $(\partial\boldsymbol{\sigma}/\partial a_z)$ in Eq. (24) can be obtained by differentiating Eq. (13)

$$\frac{\partial\boldsymbol{\sigma}}{\partial a_z} = \boldsymbol{E}\frac{\partial\boldsymbol{B}}{\partial a_z}\boldsymbol{u} + \boldsymbol{EB}\frac{\partial\boldsymbol{u}}{\partial a_z} \tag{25}$$

where the procedure for evaluating the analytical sensitivity $\partial\boldsymbol{B}/\partial a_z$ will be omitted here for conciseness but can be found in texts as Olhoff et al. (1993). As the stresses depend on the nodal displacement $\boldsymbol{U}$, the sensitivity of the nodal displacements is obtained by differentiation of Eq. (14), yielding

$$\boldsymbol{K}\frac{\partial\boldsymbol{U}}{\partial a_z} = \left(\frac{\partial\boldsymbol{F}}{\partial a_z} - \frac{\partial\boldsymbol{K}}{\partial a_z}\boldsymbol{U}\right). \tag{26}$$

Solving Eq. (26) for $\partial\boldsymbol{U}/\partial a_z$ requires to calculate $\partial\boldsymbol{K}/\partial a_z$, which is obtained by assembling a global matrix from the differentiation Eq. (15),

$$\frac{\partial\boldsymbol{k_e}}{\partial a_z} = \int_{\Omega_e}\left\{\left[\left(\frac{\partial\boldsymbol{B}}{\partial a_z}\right)^T\boldsymbol{E}\,\boldsymbol{B} + \boldsymbol{B}^T\boldsymbol{E}\left(\frac{\partial\boldsymbol{B}}{\partial a_z}\right)\right]|\boldsymbol{J}| + \boldsymbol{B}^T\boldsymbol{EB}\,\frac{\partial|\boldsymbol{J}|}{\partial a_z}\right\}d\Omega_e. \tag{27}$$

Velocity fields

In our shape optimization problem the design variables are boundary control points, whose positions affect directly the contour shape. When the boundary changes, each parameterized domain point is mapped to a new position in the (x, y, z) coordinate system, so that a smooth mesh transition to the new geometry is achieved without changing the mesh topology. The velocity field is given by the sensitivity of given domain position (nodal coordinates) with respect to the design variables. Hence one way to obtain the velocity field is via direct differentiation of the NURBS surface, evaluating it at the nodal coordinates after parametric recovery. In this case the result depends only on the geometric model, not being affected by the mesh. Another possibility is to use Laplacian smoothing schemes as proposed by Duysinx et al. (1993), but in this case the mesh density and distortion has a strong influence on the result. Laplacian schemes are used for comparison with the method proposed here.

Velocity field based on a NURBS surface

For the NURBS surface given by Eq. (2), the velocity field is obtained by

$$V_{kz} = \frac{\partial \chi_k}{\partial a_z} = \frac{\partial S(u_k, v_k)}{\partial a_z} = \frac{\sum_{i=0}^{n} \sum_{j=0}^{m} \left[N_{ip}(u_k)\, N_{jq}(v_k) \frac{d(P_{ij})}{da_z} \right]}{\sum_{i=0}^{n} \sum_{j=0}^{m} N_{ip}(u_k)\, N_{jq}(v_k)\, w_{ij}} \tag{28}$$

where V_{kz} is the velocity field associated with the design variable a_z (the coordinate of a given boundary control point) and χ_k denotes generically the nodal coordinates x_k, y_k and z_k. The expression is evaluated at the parametric coordinates u and v of node k, which are previously known from parametric recovery by Eq. (7). Notice that the NURBS surface is a function of the whole set of control points and that the parametric coordinates u and v of all the mesh nodes are held constant throughout an optimization.

In the method proposed here, the basic velocity field scheme is obtained when the design variable a_z is a given boundary control point $\overline{P_{ij}}$ and there is no influence of its position on any other control point. In this case, when $P_{ij} = \overline{P_{ij}}$, then the derivative in Eq. (28) results $d(P_{ij})/da_z = 1,$ otherwise, $d(P_{ij})/da_z = 0$. Alternatively, the position of the domain control points can be made dependent on the boundary control points defined as design variables, that is,

$$d(P_{ij})/da_z = F(L, \lambda) \neq 0 \tag{29}$$

where F is a motion propagation function that depends on a weight factor λ and on the distance L between the domain control point (P_{ij}^d) and the z-th design variable. When $\lambda = 0$, the basic scheme of the velocity field is recovered.

In order to illustrate the weighing idea, we consider a geometry parametrization in which all the boundary control points are taken as design variables. Taking the set of the n_b boundary control points, the function $F(L, \lambda)$ in Eq. (29) is calculated for the d-th domain control point by

$$\frac{d(P_{ij}^d)}{da_z} = \sum_{z=1}^{n_b} (\gamma_z)_{ij} \cdot \alpha_z \ , \tag{30}$$

where the subscripts ij in the right side of the equality correspond to the point P_{ij}^d and γ is a motion parameter dependent on λ according to

$$(\gamma_z)_{ij} = \lambda^{\,L/L_{min}} \ , \qquad 0 \leq \lambda < 1. \tag{31}$$

L is the distance from the z-th design variable to the control point P_{ij} and L_{min} is the distance to nearest P_{ij} control point. The vector $\boldsymbol{\alpha}$ in Eq. (30) can be found by using the information known for the boundary controls points, that is, $d(P_{ij}^b)/da_z = 0$ when $P_{ij}^b \neq a_z$ and $d(P_{ij}^b)/da_z = 1$ when $P_{ij}^b = a_z$. This leads to the linear system

$$\sum_{z=1}^{n_b} (\gamma_z)_{ij} \cdot \alpha_z \ = \frac{d(P_{ij}^b)}{da_z}, \tag{32}$$

or in matrix form

$$[\boldsymbol{\gamma}_{\boldsymbol{b_b}}]\, \{\boldsymbol{\alpha}\} = \frac{d\boldsymbol{P^b}}{da_z}, \tag{33}$$

where $\boldsymbol{P^b}$ is the vector of boundary control points and $\boldsymbol{\gamma}_{\boldsymbol{b_b}}$ is a matrix containing the propagation factors of the boundary control points on the boundary control points. Solving for $\boldsymbol{\alpha}$, the derivative $d(P_{ij}^d)/da_z$ in Eq.(30) can be solved for all domain control points, in matrix form

$$\frac{d\boldsymbol{P^d}}{da_z} = [\boldsymbol{\gamma}_{\boldsymbol{b_d}}]\, \{\boldsymbol{\alpha}\} \tag{34}$$

where $\boldsymbol{\gamma}_{\mathbf{b_d}}$ is a matrix containing the motion propagation factors of the boundary control points on the domain control points.

The influence of a boundary control point on the domain ones (and, by extension, to the domain nodal coordinates) can be controlled using different λ values. However, it is important to note that such influence does not always increase monotonically when increasing λ, but may have a maximum for a given value of λ in the interval [0,1], after which, the influence decreases. For a fixed value of λ the influence of a boundary control point on the domain ones can be controlled using an additional parameter φ, such that

$$\frac{d(P_{ij}^d)}{da_z} = \left(\sum_{z=1}^{n_b} (\gamma_z)_{ij} \cdot \alpha_z \right)^{\varphi} , \tag{35}$$

where $0 < \varphi < 1$ allows to increase monotonically the desired influence. This can be important to prevent excessive mesh distortions in the vicinity of the contour for large shape modifications.

Velocity fields based on Laplacian smoothing techniques

As described by Duysinx et al., (1993), in Laplacian smoothing techniques a given node i has its position Q_i smoothed according to the way it is connected to its neighboring nodes, following the scheme (see Tab. 1),

$$Q_i = \frac{1}{\sum k_{ij}} \sum_{j=1}^{N_i} k_{ij} Q_i^j \tag{36}$$

where N_i is the number of nodes around the node at the position Q_i, k_{ij} is a stiffness term introduced between node i and its j-th neighboring node connected by a finite element edge or its diagonal. From Eq.(36) we obtain a system of linear equations, where the contour and domain points are denoted by sub-indexes b and the d, respectively

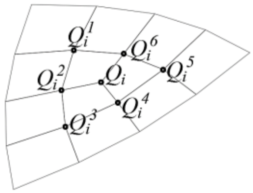
$$\begin{bmatrix} \boldsymbol{K}_{dd} & \boldsymbol{K}_{db} \\ \boldsymbol{K}_{bd} & \boldsymbol{K}_{bb} \end{bmatrix} \begin{Bmatrix} \boldsymbol{Q}_d \\ \boldsymbol{Q}_b \end{Bmatrix} = \begin{Bmatrix} 0 \\ \boldsymbol{R}_b \end{Bmatrix}. \tag{37}$$

$\boldsymbol{K}_{dd}$ is the submatrix that relates domain to domain nodes, $\boldsymbol{K}_{db}$ relates domain to boundary nodes and $\boldsymbol{K}_{db}$ relates boundary to boundary nodes. The vectors $\boldsymbol{Q}_d$ and $\boldsymbol{Q}_b$ denote the position of the domain nodes and boundary nodes and, by extension, the velocity field that is denoted by $\boldsymbol{V}_d$ and $\boldsymbol{V}_b$. Taking the first row of Eq.(37), one has

$$[\boldsymbol{K}_{dd}]\, \boldsymbol{V}_d = -\boldsymbol{K}_{db} \boldsymbol{V}_b \tag{38}$$

where the velocity field in the contour V_b is obtained analytically by parameterization of the contour by a NURBS curve, therefore the field of velocities in the contour is the same obtained by Eq. (28). Laplacian smoothing methods differ by the type of stiffness assigned to k_{ij} in Eq. (36), as described in Tab. 1.

Table 1– Laplacian Smoothing schemes.

Pure Laplacian	$k_{ij} = 1.$	If i and j connected	 Q_i^1 Q_i^6 Q_i^2 Q_i Q_i^5 Q_i^4 Q_i^3
	$\boldsymbol{k_{ij} = 0}$	If $\boldsymbol{i}$ and $\boldsymbol{j}$ unconnected	
Power Laplacian	$\boldsymbol{k_{ij} = 1/L^r}$	$\boldsymbol{L}$ =Length interface ($\boldsymbol{ij}$)	

Velocity fields based on a NURBS surface and on Laplacian approaches: an example

Figure 1a shows the location of control points for the NURBS surface parameterization of a square domain. An unstructured mesh is generated in this domain and shown in Fig. 1b. We study the velocity fields with respect to the horizontal coordinate of the boundary control point indicated by the arrow. Figure 1c shows the sensitivity for different velocity fields on a horizontal cut of the domain at the same coordinate and direction indicated by the arrow in Figure 1b. Figure 1c evinces the difference between the behavior of the Laplacian fields and the fields based on the NURBS surface parameterization, for different values of λ and $\alpha = 1$.

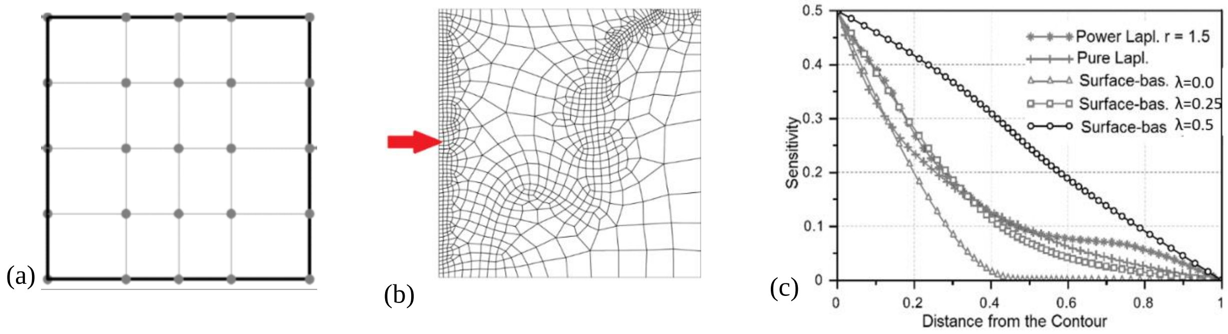


Figure 1 – a) Unstructured mesh generated on a cubic B-Spline surface, with 5x5 control points. b) Influence of velocity field for Laplacian smoothing and surface approach for several λ values.

Noteworthy, when $\lambda = 0$ the basic scheme of the velocity field is recovered, with no influence of the position of one control point on the position of the others. In addition, sensitivity by surface-based approaches the Laplacian methods when λ = 0.25 and $\varphi = 1$.

A color map of some of the aforementioned velocity fields is presented in the first row of Fig. 2. There, it is possible to see how much and how far a perturbation on the boundary affects the domain coordinates for each approach. The region of influence for the NURBS surface approach (Fig. 2a), can be easily controlled through the value of λ for $\varphi = 1$, as can be seen in Fig. 1c. The influence of the Power Laplacian velocity field can also be controlled, but less effectively, by modifying the power value (r).

The extension of the influence of a boundary perturbation on the domain nodes has a strong influence on mesh distortion, as displayed in the second row of the Figure. In order to show this, a large horizontal perturbation is applied in the position of the boundary control point indicated in Fig. 1b and the domain nodal coordinates are updated proportionally to the velocity fields.

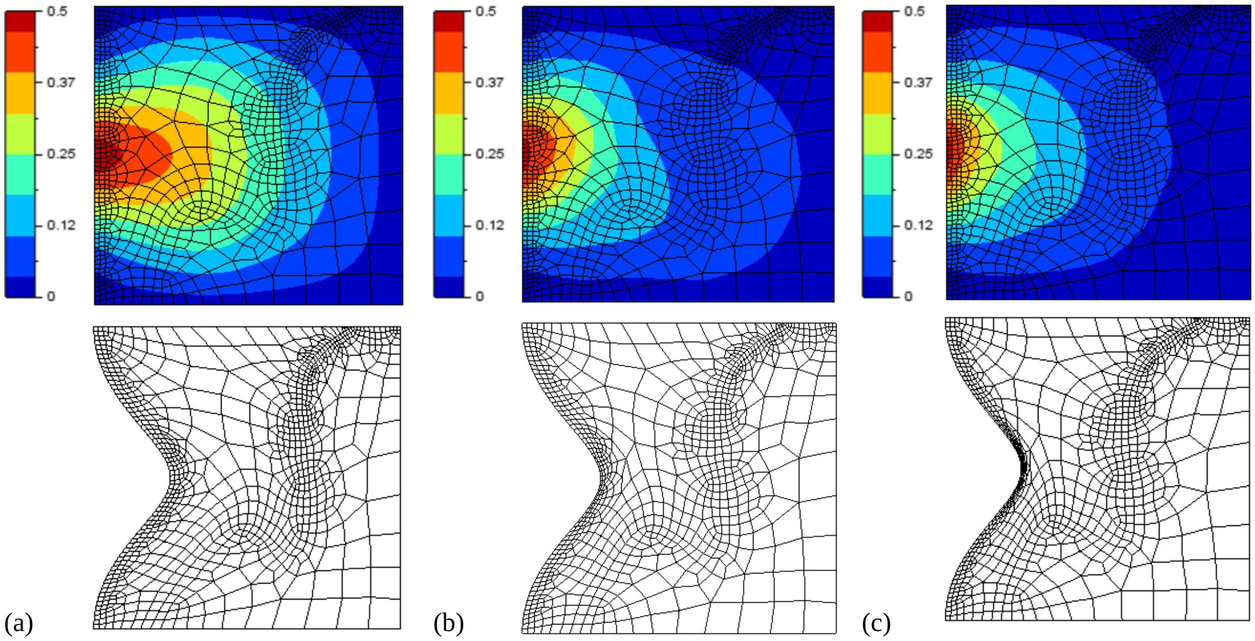


Figure 2 - Behavior of velocity fields in the mesh domain. a) Field based and B-Spline surface with $\lambda = 0.5$. b) Power Laplacian ($r = 5.0$). c) Pure Laplacian.

Fig. 2a illustrates the velocity field and the relocated mesh for such large perturbation in the case of the surface-based method for λ=0.5. It is clear that albeit the geometry change is considerable, this velocity field approach was successful in leading to a little distorted mesh. Figure 2b shows that the Power Laplacian smoothing makes smaller elements stiffer than the larger ones when compared to the pure Laplacian method in Fig. 2c. Both Laplacian schemes result in considerable mesh distortion when applied to this problem

2D SHAPE OPTIMIZATION EXAMPLE: SQUARE PANEL WITH A CENTRAL HOLE SUBJECT TO BI-AXIAL TRACTION

In this Section we present a plane stress shape optimization example defined according to the optimization problem given by Eq. (1). The aim is to minimize the weight, subject to stress constraints using the modified P-norm as proposed in Eq. (10). The domain is parameterized by a NURBS surface defined by cubic splines at the boundaries. We adopt displacement based quadrilateral 8 node elements and the nodal stresses are obtained via global L2 smoothing with values evaluated at the Gauss points following a 2×2 rule. We solve this problem using the proposed method and the Inverse Power Laplacian velocity field highlighting that, by construction, at the boundary the velocity fields are equal for both approaches. We compare the influence of using two different meshes, one semi-structured (mesh A) and another one unstructured (mesh B). The optimization problem is solved using sequential quadratic programming SQP with non-monotone line-search (Yu-Hong and Schittkowski, 2008).

The problem was modeled as shown in Fig. 3a, where the geometry of the hole must be optimized to minimize the weight, subject to a maximum allowable stress of 200 MPa. The two finite element meshes employed in the study are depicted in right side of Fig. 3a, both with same geometry parametrization (the position of the control points are also shown). The control points located at the hole contour are assigned as design variables. We compare the effect of solving this problem using the proposed method with $\lambda = 0.8$ and $\varphi = 0.8$ for both meshes, and by the Inverse Power Laplacian scheme with $r = 5$ for mesh A and $r = 8$ for mesh B. The r parameters for the Laplacian approach were selected based on experience for each mesh in order to obtain good Laplacian velocity field results.

The stress constraints are imposed as proposed by Holmberg et al. (2013), using the modified P-norm with P=20 and five stress clusters. The initial smoothed von Mises stresses are displayed for both meshes in left side of Fig. 3a. Notice that the stress constraint is being violated and a shape modification is necessary to obtain an acceptable design even if no mass reduction is attained.

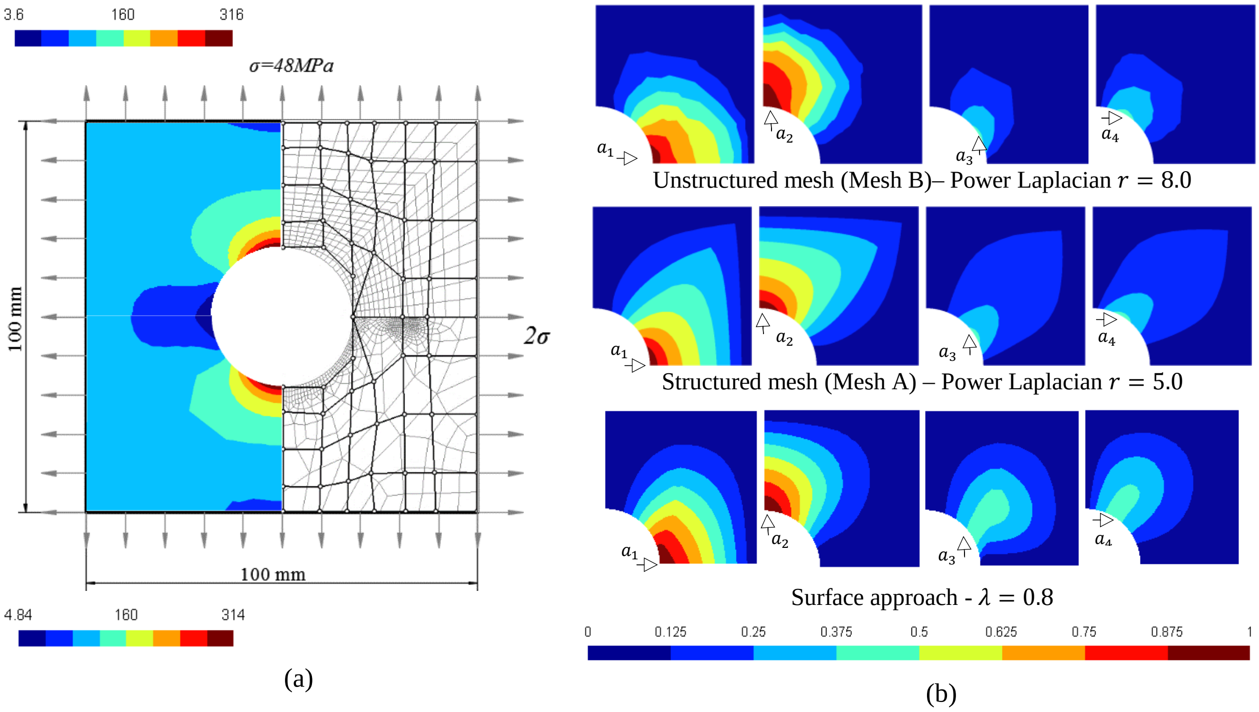

3.6
160
316
σ=48MPa
100 mm
2σ
100 mm
4.84
160
314
(a)
a_1
a_2
a_3
a_4
Unstructured mesh (Mesh B)– Power Laplacian $r = 8.0$
Structured mesh (Mesh A) – Power Laplacian $r = 5.0$
Surface approach - $\lambda = 0.8$
0
0.125
0.25
0.375
0.5
0.625
0.75
0.875
1
(b)

Figure 3 – a) Geometry and loading acting on the square panel, geometry parameterization using a NURBS surface and Inicial meshes (semi-structured in the top and unstructured below), Smoothed von Mises stress field in the initial configuration for both meshes (semi-structured in the top and unstructured below).
b) Influence of the mesh on the velocity fields.

Before proceeding to the optimization results themselves, it is worth observing how the different mesh discretizations affect the Power Laplacian velocity fields compared to the NURBS surface approach in this specific problem. Figure 3b shows the velocity fields for the four control point positions taken as design variables. It is noticeable that the mesh grading and the power r, affects strongly the Laplacian schemes. The unstructured mesh has a very fine mesh near the design boundary region, which rapidly becomes coarse in the domain. The mesh grading gradient is much smoother in the semi-structured mesh. The inverse power Laplacian scheme attributes a high stiffness to small elements when compared to their neighbors while in the pure Laplacian this stiffness is constant no matter the element size. Hence, we can observe that the power Laplacian scheme applied to the unstructured mesh B with r=8 (first row of Fig. 3b), shows a more intense velocity fields near the boundary than applied to the semi-structured mesh A with r=5 (second row of Fig. 3b). We also notice that the semi-structured mesh induces an elongation in the velocity fields along a line at 45 degrees due to the mirroring adopted to generate the mesh. The third row in Fig. 3b shows the NURBS based velocity fields, which are insensitive to the mesh characteristics.

Now, concerning optimization itself, Fig. 4 shows the optimized shape configurations for both meshes. In both cases the proposed approach led to less severely distorted elements when compared to the inverse Laplacian scheme, a fact of concern especially when the optimization problem requires stress evaluation. Notice that for the unstructured mesh, the proposed approach succeeded to end up with an acceptable distortion level whereas the Laplacian alternative had the optimization interrupted due to excessive distortion of the elements in the contour of the hole.

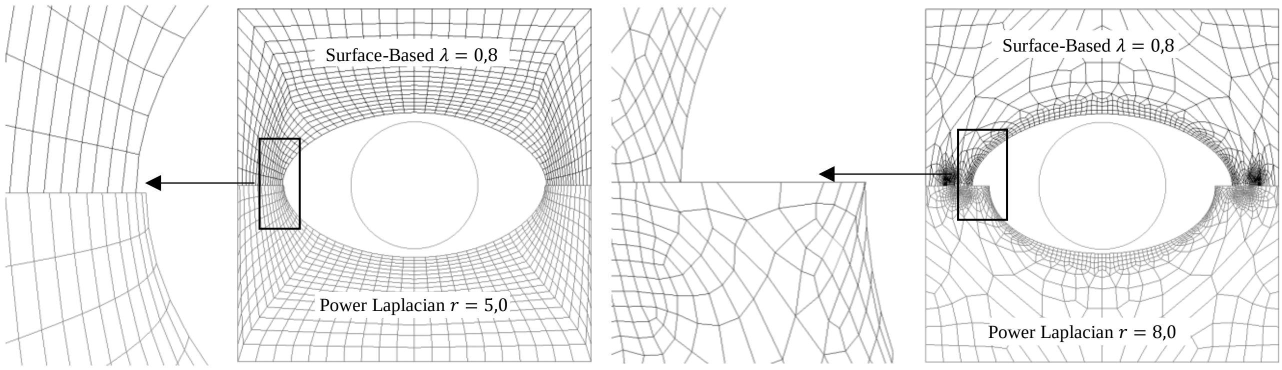
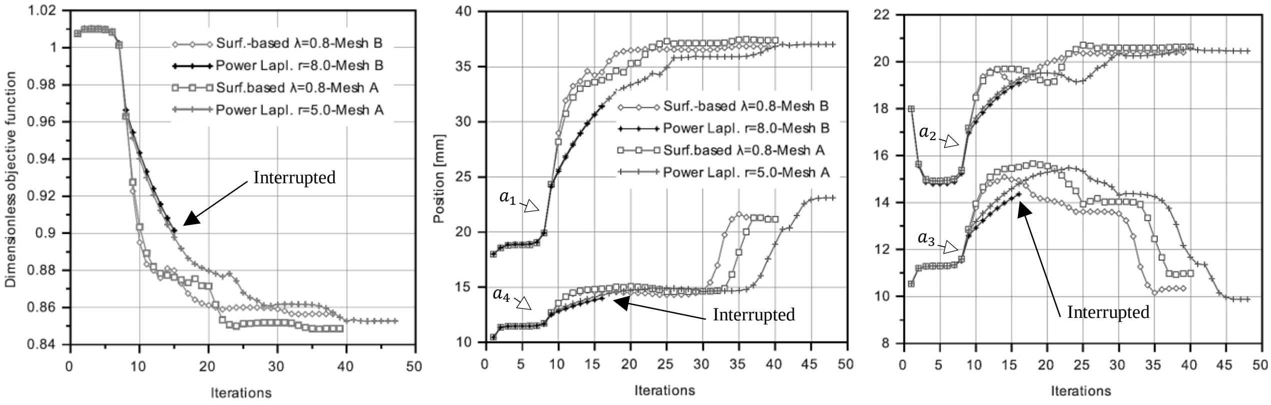

Surface-Based $\lambda = 0,8$
Power Laplacian $r = 5,0$
Surface-Based $\lambda = 0,8$
Power Laplacian $r = 8,0$

Figure 4 - a) Optimal shapes for the semi-structured mesh. b) Optimal shapes for the unstructured mesh.

Figure 5 displays the objective function and design variables evolution along external iterations of the SQP. For the unstructured mesh, the surface-based method leaded to a better optimal shape than the Laplacian counterpart. In addition, the latter shows a great influence of the mesh topology on the optimization result. A mass reduction of approximately 15% (see Tab. 2) was reached with both approaches, except for the power Laplacian approach applied to the unstructured mesh (mesh B), which had the optimization interrupted because of excessive mesh distortion.



(a) (b) (c)

Figure 5 - a) Objective function convergence; b) a_1 and a_4 design variables convergence; c) a_2 and a_3 design variables convergence;

Table 2– Numerical values of the dimensionless objective function (normalized with respect to the initial value)

	Semi-Structured Mesh						Unstructured Mesh					
	Objective Function	Stress constraints ***g***					**Objectve Function**	Stress Constraints ***g***				
Surface-based	**0.848**	**0,0**	**0,15**	**0,24**	**0,34**	**0,45**	**0.856**	**0,0**	**0,10**	**0,24**	**0,26**	**0,28**
Power Lapl.	**0.853**	**0,0**	**0,12**	**0,23**	**0,33**	**0,47**	**0.902***	**-0,007**	**0,18**	**0,42**	**0,44**	**0,46**

Figure 6a shows that in all the cases, the smoothed von Mises stress fields slightly exceed the maximum allowable stress of 200MPa. This means that although stress constraints are well handled using Holmberg's procedure, the five stress clusters sorted by stress level are not sufficient to enforce the stress constraints rigorously. As shown in Tab. 2, only the first constraint ends active. Figure 6b illustrates the convergence of the constraints that correspond to the cluster containing the highest stresses for all meshes and approaches.

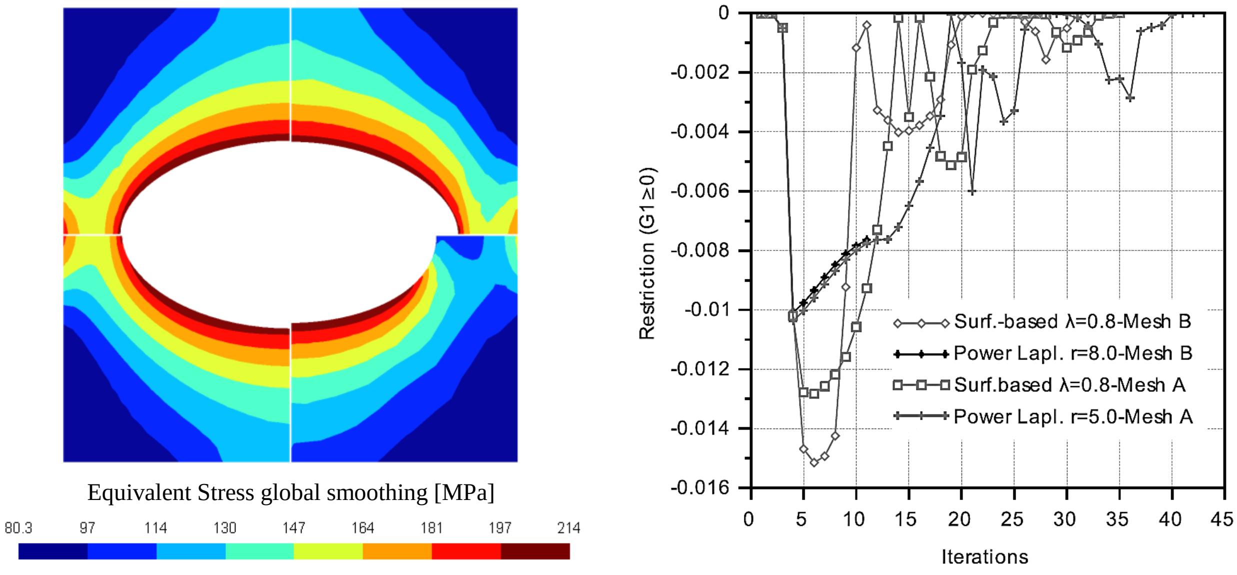


Figure 6 - a) Final stress distribution for the different velocity fields; b) main constraints (g_1) convergence.

CONCLUSIONS

Laplacian approaches are well established for calculating velocity fields in shape optimization problems discretized using finite elements. Notwithstanding, such strategies are strongly affected by mesh gradings and element distortions. Hence, a suitable mesh for discretization of a finite element problem cannot always be favorable for calculating a Laplacian velocity field. In this work we present a mesh independent procedure to evaluate 2D velocity fields based on NURBS surfaces, adopting boundary control points as design variables. A key and original contribution of the method developed is that domain control points can be affected by the design variables (or not) through a user-defined weight. Therefore we highlight the advantage of the proposed approach, which allows controlling the quality of the velocity field independent of the mesh topology.

A plane stress example is used to show the potential of the proposed method, which showed enhanced results with respect to Laplacian velocity fields, including a lower value for the weight objective function and a less pronounced distortion of the mesh near the boundary.

Holmberg's procedure must be better exploited in order to determine a relation of the P-norm and the number of clusters that must be adopted to enforce satisfactorily the stress constraints. Moreover, following Holmberg's recommendation (Holmberg et al, 2013) the clusters are modified and sorted along optimization iterations. Although this strategy can help to satisfy adequately the stress constraints, the mathematical optimization problem being solved is changed every time the clusters are reorganized, which is inconsistent. Theoretically this can bring convergence difficulties analogous to changing the mesh topology. Future works will evaluate this aspect in more depth.

Finally, it is worth mentioning that as the proposed approach is developed entirely on a parametric domain, we expect a straightforward extension for applications in shape optimization of arbitrarily curved shells.

ACKNOWLEDGEMENTS

The authors are sincerely grateful by Professor K. Schittkowski for the provision of the Fortran subroutine NLPQLP for solve nonlinear programming problems by a sequential quadratic programming (SQP) with distributed and non-monotone line search.

REFERENCES

Augusto, R. A., 2012, "Arquitetura Orientada por Objetos para o MEF de Alta Ordem com Aplicações em Mecânica Estrutural", Ph. D. Thesis. Universidade Estadual de Campinas.

Barthelemy, B., Chon. T. and Haftka, R. T., 1988, "Accuracy problems associated with semi-analytical derivatives of static response." Finite Elements in Analysis and Design 4, pp. 249-265.

Beckers, P., 1991, "Recent Developments in Shape Sensitivity: The Physical Approach", Engineering Optimization, Vol.18, pp. 67-78.

Duysinx, P., Fleury, C., and Zhang, W.H., 1993, "Sensitivity Analysis with Unstructured Free Mesh Generators in 2-D Shape Optimization", Report AO-31, Aerospace Laboratory, LTAS, University of Liége, Belgium.

Haug, E. J., Choi, K. K., and Komkov, V., 1986, "Design Sensitivity Analysis of Structural", Ed. Academic Press, Inc., Orlando, New York, 381 p.

Holmberg, E., Torstenfelt, B. and Klarbring, A., 2013, "Stress constrained topology optimization", Structural and Multidisciplinary Optimization, Vol.48, pp. 33-47.

Olhoff, N., Rasmussen, J. and Erik, L., 1993, "A Method of "Exact" Numerical Differentiation for Error Elimination in Finite-Element-Based Semi-Analytical Shape Sensitivity Analyses." Mechanics of Structures and Machines, Vol.21, No.1, pp. 1-66.

Piegl, L., and Tiller, W., 1986, "The NURBS book. 2^a ed.", Ed. Springer, Florida, 645 p.

Schleupen, A., Maute, K. and Ramm, E., 2000, "Adaptive FE-procedures in shape optimization", Structural and Multidisciplinary Optimization, Vol.19, pp. 282-303.

Wang, S., Sun, Y. and Gallagher, R. H., 1985, "Sensitivity Analysis in Shape Optimization", Computers and Structures, Vol.20, No. 5, pp. 855-867.

Wildemann, F. and Muñoz-Rojas, P., 2005, "Stress Recovery Techiniques Applied to Plane Elasticity and Solids of Revolution", In Proceedings of the XXVI CILAMCE-Iberian Latin American Congress on Computational Methods in Engineering, Espírito Santo, Brazil, 2005.

Yu-Hong, D., e Schittkowski, K., 2008, "A Sequential Quadratic Programming Algorithm with Non-Monotone Line Search", Pacific Journal of Optimization, Vol.4, No. 2, 335-351.

RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.