XLI CILAMCE
XLI Ibero-Latin American Congress on Computational Methods in Engineering
16-19 November, 2020 FOZ DO IGUAÇU PARANÁ - BRAZIL

Isogeometric analysis using Bézier elements

Francisco D. P. Silva[1], Elias S. Barroso[2], Evandro Parente Jr.[1]

[1]*Laboratório de Mecânica Computacional e Visualização, Departamento de Engenharia Estrutural e Construção Civil, Universidade Federal do Ceará*
Campos do Pici, 60440-900, Fortaleza, Ceará, Brazil
davydsilva.eng@gmail.com, evandro@ufc.br
[2]*Departamento de Computação, Universidade Federal do Ceará*
Campos do Pici, 60440-900, Fortaleza, Ceará, Brazil
elias.barroso@gmail.com

Abstract. Isogeometric Analysis (IGA) is a numerical analysis approach that integrates the concepts of CAD (Computer-Aided Design) and CAE (Computer-Aided Engineering). It uses the same basis functions employed by CAD systems to describe the model geometry (e.g. Bézier and NURBS surfaces) to approximate the solution field (e.g. displacement or temperature). Therefore, the geometry of the models is exactly represented for any level of discretization. The use of well-known geometric modeling algorithms (e.g. knot insertion and degree elevation) simplifies the model refinement procedure, allowing an easy application of discretization schemes while preserving the initial geometry. The modeling paradigm adopted in CAD systems uses boundary models that cannot be directly used by CAE. The use of Bézier elements is a good alternative, as it facilitates the connection between the geometric model, based on a Boundary Representation (B-Rep), and the isogeometric analysis model. High-order mesh generation techniques are used to discretize geometric models through rational Bézier elements, preserving the exact geometry and creating a suitable analysis model. This paper discusses the definition of two-dimensional triangular Bézier elements and their performance in numerical analysis. The numerical results are compared with an analytical solution, where the convergence rate of the Bézier elements is assessed in linear elasticity problem.

Keywords: Isogeometric Analysis, Bézier elements, Convergence

1 Introduction

The Isogeometric Analysis (IGA) is a numerical method introduced by Hughes et al. [1] to integrate CAD (Computer-Aided Design) and CAE (Computer-Aided Engineering). The main feature of the IGA is that the geometry of the structure is exactly represented, as the same basis functions employed by CAD systems to describe the model geometry (e.g. Bézier and NURBS surfaces) are used to approximate the solution fields. Thus, well-known geometric modeling algorithms (e.g. knot insertion and degree elevation) can be used to refine the analysis model [2].

The CAD systems usually adopt the Boundary Representation (B-Rep) paradigm, where the model geometry is represented by a collection of bounding surfaces and cannot be used directly by FEM based CAE systems. This occurs because of these representations do not provide an explicit parameterization of the domain to be analyzed [3]. A large body of work was developed in order to solve this problem using different basis functions like B-Splines [4], NURBS [5], T-Splines [6]. However, they had difficulties in automatically generating suitable isogemetric meshes for complex geometries.

The use of Bézier elements is a good alternative to overcome such issues, as it facilitates the connection between CAD and CAE. These elements allow the use of high-order mesh generation techniques to discretize domain of the problem, preserving the exact geometry of the model and yielding suitable analysis models. Thus, the Bézier elements in IGA has been receiving considerable attention [3, 7–10].

This paper discusses the definition of two-dimensional triangular Bézier elements, implemented by Barroso et al. [8, 10], and their performance in numerical analysis. The mesh generation is carried out using the academic program Plane Mesh Generator (PMGen), which features a graphical user interface for 2D mesh generation for finite element and isogeometric analysis [11]. The numerical results are compared with an analytical solution, where the convergence rate of the Bézier element is assessed in linear elasticity problem.

This paper is organized as follows. Section 2 review some concepts of Bézier elements. Section 3 describes the formulation of IGA used in this work. Section 4 presents the numerical example. The conclusions of this work are presented in Section 5.

2 Geometry Modeling

In this section, we show important concepts related to geometric modeling using Bézier and NURBS elements such as rational Bézier curves, NURBS curves, rational Bézier triangles, and Nurbs surfaces.

2.1 Rational Bézier Curves

A rational Bézier curve of degree p is defined by the linear combination:

$$C(\xi) = \sum_{i=0}^{p} R_{i,p}(\xi)\, \mathbf{p}_i, \qquad R_{i,p}(\xi) = \frac{B_{i,p}(\xi)\, w_i}{\sum_{\hat{i}=0}^{p} B_{\hat{i},p}(\xi)\, w_{\hat{i}}}, \tag{1}$$

where ξ is the parametric coordinate, $\mathbf{p}_i$ is a set of control points, $R_{i,p}$ are a rational bases, w_i is the weight associated with the control point p_i and $B_{i,p}$ are the Bernstein polynomials, defined as:

$$B_{i,p}(\xi) = \frac{p!}{i!\ (p-i)!}(1-\xi)^{p-i}\, \xi^i, \quad i = 0, 1, ..., p, \tag{2}$$

Bernstein polynomials have important properties like: linear independence, non-negativity, partition of unity, and symmetry [12].

2.2 NURBS Curves

NURBS (Non-Uniform Rational B-splines) curves of degree p are defined by the linear combination:

$$C(\xi) = \sum_{i=1}^{n} R_{i,p}(\xi)\, \mathbf{p}_i, \qquad R_{i,p}(\xi) = \frac{N_{i,p}(\xi)\, w_i}{\sum_{\hat{i}=1}^{n} N_{\hat{i},p}(\xi)\, w_{\hat{i}}}, \tag{3}$$

where n is the number of bases and $N_{i,p}$ are the B-Spline base functions, which require a knot vector, a set of non-negative and non-decreasing parametric values, $\Xi = [\xi_1, \xi_2, \xi_3, ..., \xi_{m+p+1}]$. Given the knot vector, the B-spline basis functions are defined via the Cox–de Boor recursion formula as [12]:

$$N_{i,0}(\xi) = \begin{cases} 1, & \xi_i \leq \xi \leq \xi_{i+1} \\ 0, & \text{otherwise,} \end{cases} \tag{4}$$

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi). \tag{5}$$

B-spline basis functions present important properties, including linear independence, non-negativity, partition of unity, and compact support.

2.3 Rational Bézier Triangles

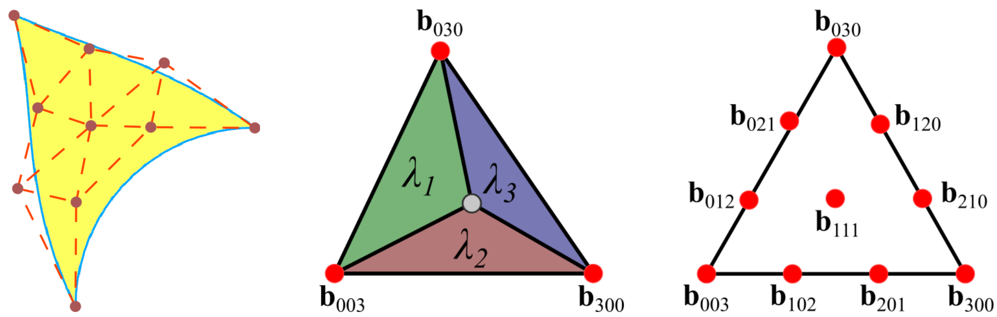
Rational Bézier triangles are bivariate surfaces defined by a set of control points, arranged in a triangular structure. It is worth mentioning that the edges of the Bézier triangles are Bézier curves. These triangular surfaces are defined by [3]:

$$T(\boldsymbol{\lambda}) = \sum_{i+j+k=p} R_{ijk}^{p}(\boldsymbol{\lambda})\, \mathbf{p}_{ijk}, \qquad R_{ijk}^{p}(\boldsymbol{\lambda}) = \frac{B_{ijk}^{p}(\boldsymbol{\lambda})\, w_i}{\sum_{i+j+k=p} B_{ijk}^{p}(\boldsymbol{\lambda})\, w_i}, \tag{6}$$

where p is the degree, $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \lambda_3)$ are barycentric coordinates, $\mathbf{p}_{ijk}$ are the control points, R^p_{ijk} are the bivariate rational bases, and B^p_{ijk} are the bivariate Bernstein polynomials defined as:

$$B^p_{ijk}(\boldsymbol{\lambda}) = \frac{p!}{i!j!k!}\,\lambda_1^i\,\lambda_2^j\,\lambda_3^k, \qquad \lambda_1 + \lambda_2 + \lambda_3 = 1, \qquad 0 \le \lambda_i \le 1 \quad (i = 1, 2, 3), \tag{7}$$

where $i + j + k = p$ and $i, j, k \geq 0$. Fig. 1 illustrates a cubic Bézier triangle and its parametric space with the triple index scheme. It is worth pointing out that the bivariate Bernstein polynomials have similar properties as the univariate case discussed in the Section 2.1, as linear independence, non-negativity, partition of unity.



(a) Physical space. (b) Parametric space. (c) Index space.

Figure 1. Cubic Bézier triangle.

2.4 NURBS Surfaces

NURBS Surfaces are defined from tensor product of two univariate NURBS basis functions. Thus, a NURBS surface S of degrees p and q, in directions ξ and η, respectively, is defined by a linear combination of bivariate NURBS basis functions $\hat{R}_{ij}(\xi, \eta)$ and a matrix of control points $\mathbf{P}$ $(n \times m)$:

$$S(\xi, \eta) = \sum_{i=1}^{n}\sum_{j=1}^{m} \hat{R}(\xi, \eta)_{ij}\,\mathbf{P}_{ij}, \qquad \hat{R}(\xi, \eta) = \frac{w_{ij}\,B_{i,p}(\xi)\,B_{j,q}(\eta)}{W(\xi, \eta)}, \tag{8}$$

where $W(\xi, \eta)$ is the bivariate weight function:

$$W(\xi, \eta) = \sum_{\hat{i}=0}^{p}\sum_{\hat{j}=0}^{q} w_{\hat{i}\hat{j}}\,B_{\hat{i},p}(\xi)\,B_{\hat{j},q}(\eta). \tag{9}$$

It is important to note that the bivariate NURBS basis have similar properties defined for the univariate NURBS, as linear independence, non-negativity, partition of unity, and compact support.

2.5 Boundary Representation

The B-Rep is a geometric modeling paradigm usually adopted in CAD systems. It consists in representing models by its boundary entities, which correspond to a set of bounding surfaces in 3D models and bounding curves in plane models. There are many algorithms available for the manipulation and modification of B-Rep models, which make this approach extremely interesting in the geometric modeling field. Furthermore, the explicit definition of the model's boundary enables an easy application of rendering techniques through modern graphics API, such as OpenGL and DirectX.

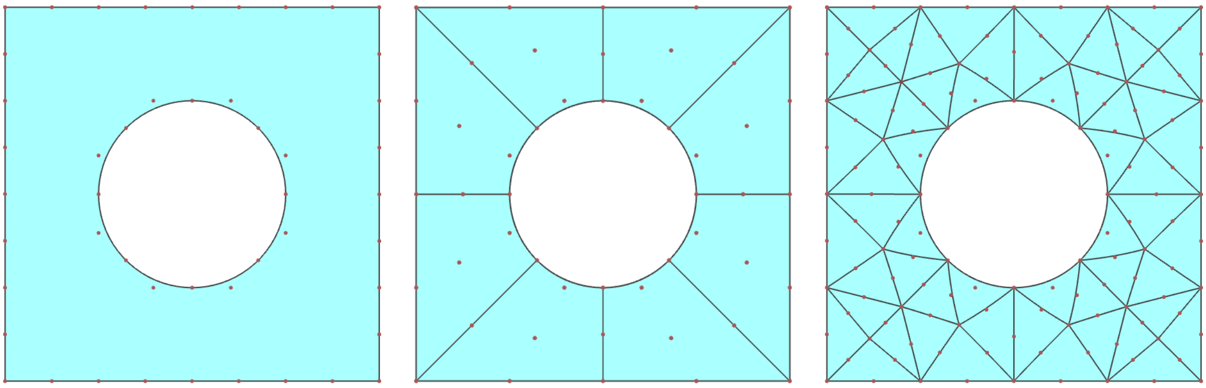
Despite these advantages, B-Rep does not to provide directly an analysis model required by finite element or isogeometric solvers. There is a surface-to-volume parametrization problem to be solved to obtain a valid numerical model. This issue has been tackled through the application of mesh generation techniques [13].

In the context of IGA, this problem remains an active field of research. There are many approaches in the literature aiming to solve the surface-to-volume parametrization problem, using multiple NURBS patches, T-Splines, and other representations, but these solutions lack in robustness. On the other hand, the use of rational Bézier elements allows the automatic generation of analysis-suitable meshes.

In the case of plane models described by NURBS curves, the unstructured rational Bézier triangle mesh generation algorithm is described as follows [11]:

- Initially, rational Bézier segments are extracted from NURBS curves, through Bézier extraction procedure;
- A linear mesh generator is processed, using the chords of Bézier segment as input edges;
- The Degree Elevation algorithm is applied to linear mesh elements, and the input boundary curves are restored, preserving geometry exactness;
- Lastly, post-processing smoothing can be applied to improve mesh quality.

For instance, we consider a simple plate with hole model illustrated in Figure 2. Since NURBS can not represent holes due to its tensor product nature, the model should be manually broken into multiple patches. Once the eight NURBS patches are found, the refinement procedure can be applied. On the other hand, a mesh composed of rational Bézier triangles can be obtained directly using the procedure described previously.



(a) B-Rep Model. (b) Multiple NURBS patches. (c) Bézier Triangle Mesh.

Figure 2. 2D model and its parametrizations using NURBS patches and rational Bézier triangles.

3 Isogeometric Analysis

Isogeometric analysis uses the same idea as the MEF isoparametric formulation, but the approximation sequence is reversed. Therefore, the displacement field is approximated using the same basis functions used by the CAD systems for geometric modeling. This feature allows the geometry of the structure to be exactly represented, independent of the discretization level adopted in the numerical analysis.

In this work, from the geometric modeling concepts discussed in the Section 2, a two-dimensional structure is described by:

$$x = \sum_{i=1}^{np} R_i\, x_i, \qquad y = \sum_{i=1}^{np} R_i\, y_i, \tag{10}$$

where np is the number of control points, R_i are the basis functions, and x_i, y_i are are the coordinates of the control points.

The displacements within the structure are approximate with the same basis functions used to describe the geometry of the solid:

$$u = \sum_{i=1}^{np} R_i\, u_i, \qquad v = \sum_{i=1}^{np} R_i\, v_i, \tag{11}$$

where R_i are the rational Bézier triangle bases presented in Eq. (6), u_i and v_i are the displacements of the control points. These equations can be written in matrix format as:

$$\hat{\mathbf{u}} = \begin{Bmatrix} u \\ v \end{Bmatrix} = \sum_{i=1}^{np} \begin{bmatrix} R_i & 0 \\ 0 & R_i \end{bmatrix} \begin{Bmatrix} u_i \\ v_i \end{Bmatrix} = \sum_{i=1}^{np} \mathbf{N}_i\, \mathbf{u}_i = \mathbf{N}\, \mathbf{u}, \tag{12}$$

where $\mathbf{u}$ is the vector of degrees of freedom and $\mathbf{N}$ is the approximation matrix given by:

$$\mathbf{N} = \begin{bmatrix} \mathbf{N}_1 & \mathbf{N}_2 & \dots & \mathbf{N}_{np} \end{bmatrix}. \tag{13}$$

Considering that the displacements are small, the strains are calculated as:

$$\epsilon = \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix} = \begin{Bmatrix} u_{,x} \\ v_{,y} \\ u_{,y} + v_{,x} \end{Bmatrix} = \sum_{i=1}^{np} \begin{bmatrix} R_{i,x} & 0 & 0 \\ 0 & R_{i,y} & 0 \\ R_{i,y} & R_{i,x} & 0 \end{bmatrix} \begin{Bmatrix} u_i \\ v_i \end{Bmatrix} = \mathbf{B}\,\mathbf{u}. \tag{14}$$

where the strain-displacement matrix ($\mathbf{B}$) has the same format of the approximation matrix ($\mathbf{N}$), defined in Eq. (13). Furthermore, in linear elasticity the stresses ($\boldsymbol{\sigma}$) are computed using the Hooke's law:

$$\boldsymbol{\sigma} = \mathbf{C}\,\boldsymbol{\epsilon}. \tag{15}$$

Using the Principle of Virtual Work, the equilibrium equations of the model can be written as:

$$\delta W_{int} = \delta W_{ext} \quad \Rightarrow \quad \int_V \delta\epsilon^T \sigma \, dV = \int_V \delta\hat{\mathbf{u}}^T \, \mathbf{b} \, dV + \int_S \delta\hat{\mathbf{u}}^T \, \mathbf{q} \, dS, \tag{16}$$

where $\mathbf{b}$ is the body load, $\mathbf{q}$ is the surface load, V is the volume, S is boundary of the structure, and $\delta\epsilon$ is the virtual strain, given by:

$$\delta\epsilon = \mathbf{B}\,\delta\mathbf{u}, \tag{17}$$

Substituting Eq. (12) and Eq. (17) in Eq. (16), the equilibrium equations of the model can be written as:

$$\delta\mathbf{u}^T\mathbf{g} = \delta\mathbf{u}^T\mathbf{f}, \tag{18}$$

where $\mathbf{f}$ is the external forces vector and $\mathbf{g}$ is the internal force vector, given by:

$$\mathbf{f} = \int_V \mathbf{N}^T \, \mathbf{b} \, dV + \int_S \mathbf{N}^T \, \mathbf{q} \, dS, \tag{19}$$

$$\mathbf{g} = \int_V \mathbf{B}^T \sigma \, dV = \int_V \mathbf{B}^T \mathbf{C}\,\epsilon \, dV = \int_V \mathbf{B}^T \mathbf{C}\,\mathbf{B} \;\; dV \mathbf{u} = \mathbf{K}\,\mathbf{u}, \tag{20}$$

where $\mathbf{K}$ is the stiffness matrix, given by:

$$\mathbf{K} = \int_V \mathbf{B}^T \mathbf{C}\,\mathbf{B} \;\; dV. \tag{21}$$

Finally, replacing the relationship $\mathbf{g} = \mathbf{K}\ \mathbf{u}$ in Eq. (18) and like virtual displacements ($\delta\mathbf{u}$) are arbitrary, the equilibrium equation can be written as:

$$\mathbf{K}\mathbf{u} = \mathbf{f}. \tag{22}$$

Note that the element stiffness matrix ($\mathbf{K}_e$) and force vector ($\mathbf{f}_e$) are evaluated through numerical integration on each element domain. The triangular quadrature presented in [14] is adopted here. The global equilibrium system shown in the Eq. (22) is obtained using the same techniques used in FEM.

The presented isogeometric formulation was implemented in the structural analysis software FAST [15], developed in C ++ programming language and using the object-oriented programming (OOP) paradigm.

4 Numerical Example

This example deals with a infinite plate with circular hole under constant in-plane stress in the x-direction. The infinite plate is modeled as a finite plate as shown in Fig. 3. The material propriety adopted is $E = 10^5$ and Poison's ratio is $\nu = 0.3$. The analytical solution to this problem can be found in Hughes et al. [1]:

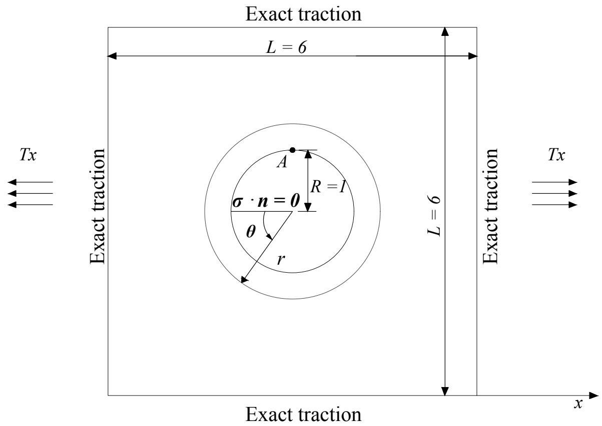
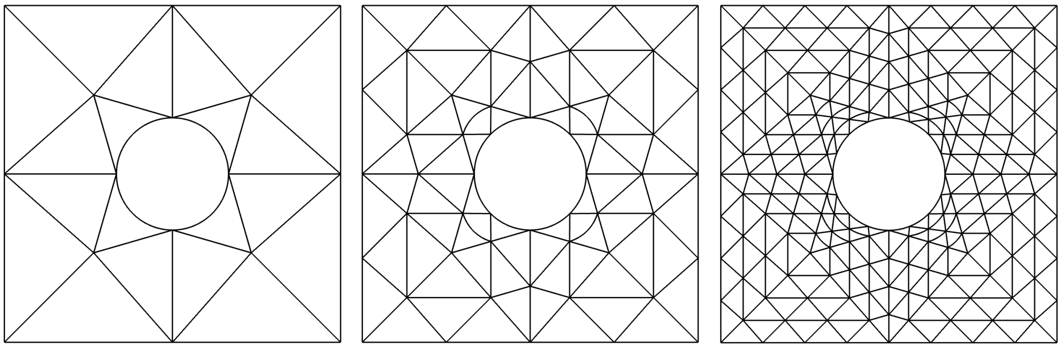


Figure 3. Elastic plate with a circular hole: problem definition

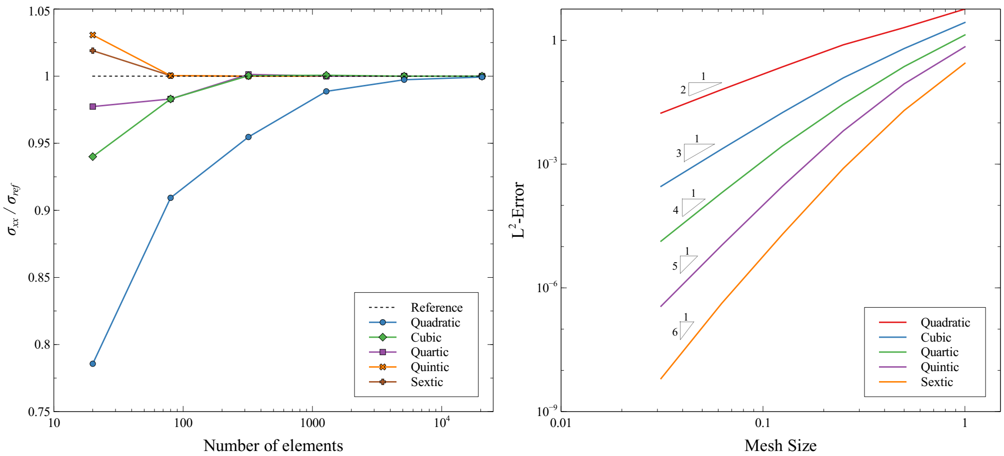
The meshes were generated using PMGen program, and uniformly refined [3]. The numerical problem is solved using FAST. Figure 4 shows the meshes under several levels of refinement.



(a) $h = 1.0$ (b) $h = 0.5$ (c) $h = 0.25$

Figure 4. Meshes used at refinement-h ($p = 2$)

Figure 5(a) shows the results obtained for the stress concentration of σ_{xx} at the point A, in the Fig. 3. The values shown are the ratio between the numerical value obtained and the value of the analytical solution. All results converged to the exact solution of the problem with the discretization of the model. Convergence results in L^2-norm of stresses are show in Fig. 5(b). As expected, the convergence rates for quadratic, cubic, quartic, quintic and sextic Bézier elements are approximately 2, 3, 4, 5 and 6, respectively.



(a) Stress convergence (b) Error measured in the L^2-norm of stress vs. mesh size

Figure 5. Numerical results

5 Conclusions

This work presented the formulation of two-dimensional rational Bézier triangles for isogeometric analysis of 2D linear elasticity problems and studied its performance. Overall, the results found are excellent and show a good agreement between analysis with these elements and the analytical solution. The convergence rates presented the values corresponding to the theoretical rates for different element degrees. It is important to note that the use of rational Bézier triangles with the unstructured isogeometric mesh generator allows the analysis of complex models through automatic mesh generation geometrically exact without user interaction in which the model does not need to be subdivided into multiple patches.

Acknowledgements. This study was financed by the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) and the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. The authors gratefully acknowledge the financial support provided by these agencies.

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] Hughes, T., Cottrell, J., & Bazilevs, Y., 2005. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, vol. 194, n. 39-41, pp. 4135–4195.
[2] Cottrell, J. A., Hughes, T. J., & Bazilevs, Y., 2009. *Isogeometric Analysis: Toward Integration of CAD and FEA*. Wiley.
[3] Engvall, L. & Evans, J. A., 2016. Isogeometric triangular bernstein–bézier discretizations: Automatic mesh generation and geometrically exact finite element analysis. *Computer Methods in Applied Mechanics and Engineering*, vol. 304, pp. 378–407.
[4] Martin, T., Cohen, E., & Kirby, M., 2008. Volumetric parameterization and trivariate b-spline fitting using harmonic functions. In *Proceedings of the 2008 ACM symposium on Solid and physical modeling - SPM 08*. ACM Press.
[5] Akhras, H. A., Elguedj, T., Gravouil, A., & Rochette, M., 2016. Isogeometric analysis-suitable trivariate NURBS models from standard b-rep models. *Computer Methods in Applied Mechanics and Engineering*, vol. 307, pp. 256–274.
[6] Liu, L., Zhang, Y., Hughes, T. J. R., Scott, M. A., & Sederberg, T. W., 2013. Volumetric t-spline construction using boolean operations. *Engineering with Computers*, vol. 30, n. 4, pp. 425–439.
[7] Xia, S., Wang, X., & Qian, X., 2015. Continuity and convergence in rational triangular bézier spline based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, vol. 297, pp. 292–324.
[8] Barroso, E. S., Neto, J. B. C., Vidal, C. A., & Parente Jr, E., 2016. Análise isogeométrica utilizando triângulos de Bézier. In *Proceedings of the XXXVII Iberian Latin American Congress on Computational Methods in Engineering*.
[9] Engvall, L. & Evans, J. A., 2017. Isogeometric unstructured tetrahedral and mixed-element bernstein–bézier discretizations. *Computer Methods in Applied Mechanics and Engineering*, vol. 319, pp. 83–123.
[10] Barroso, E. S., Cavalcante Neto, J. B., Vidal, C. A., & Parente Jr, E., 2017. Geração de malhas isogeometricas utilizando mapeamento transfinito. In *Proceedings of the XXXVIII Iberian Latin American Congress on Computational Methods in Engineering*. ABMEC Brazilian Association of Computational Methods in Engineering.
[11] Barroso, E. S., Evans, J. A., Cavalcante Neto, J. B., Vidal, C. A., & Parente Jr, E., 2019. An algorithm for automatic discretization of isogeometric plane models. In *Proceedings of the XL Iberian Latin American Congress on Computational Methods in Engineering*.
[12] Piegl, L. A. & Tiller, W., 1997. *The Nurbs Book (Monographs in Visual Communication)*. Springer.
[13] Frey, P. & George, P., 2008. *Mesh Generation: Application to Finite Elements: Second Edition*. John Wiley & Sons, Inc.
[14] Dunavant, D. A., 1985. High degree efficient symmetrical gaussian quadrature rules for the triangle. *International Journal for Numerical Methods in Engineering*, vol. 21, n. 6, pp. 1129–1148.
[15] Barroso, E. S., 2015. Análise e otimização de estruturas laminadas utilizando a formulação isogeométrica. 2015. 161 f. Mestrado em engenharia civil, Centro de Tecnologia, Programa de Pós-Graduação em Engenharia Civil: Estruturas e Construção Civil, Universidade Federal do Ceará, Fortaleza.