

Some modeling features for two-dimensional isoparametric and isogeometric finite element analysis

João Carlos L. Peixoto¹, Danilo S. Bomfim², Rodrigo L. Soares¹, Luiz F. Bez³, Pedro C. F. Lopes³, André M. B. Pereira³, Rafael L. Rangel⁴, Luiz F. Martha¹

¹*Dept. of Civil and Environmental Engineering, Pontifical Catholic University of Rio de Janeiro
Rua Marques de São Vicente, 225, 22451-900, 22451-900, Gávea, Rio de Janeiro - RJ, Brazil
joaoclpeixoto@gmail.com, rodrigolucassoares@gmail.com, lfm@tecgraf.puc-rio.br*

²*Dept. of Structural Engineering, The São Carlos School of Engineering - University of São Paulo
Av. Trab. São Carlense, 400 - Parque Arnold Schmidt, São Carlos - SP, 13566-590, Brazil
dsbomfim2@hotmail.com*

³*Institute of Computing, Fluminense Federal University
Av. Gal. Milton Tavares de Souza, s/n - São Domingos, 24210-310, RJ/Niterói Brazil
luizf.bez@gmail.com, pedrocortez@id.uff.br, andre@ic.uff.br*

⁴*International Center for Numerical Methods in Engineering, Polytechnic University of Catalonia (UPC)
C. Gran Capità S/N, 08034, Barcelona, Spain
rrangel@cimne.upc.edu*

Abstract. This work describes some modeling features of an interactive graphics system for finite element simulations in two dimensions. The specific objective of this article is to describe the strategy for domain decomposition into patches ready for isoparametric and isogeometric analysis. The main objective of this system in the future is to provide students and researchers of computational mechanics with an open educational tool for understanding the integration of geometric modeling, mesh generation, finite element analysis, and visualization of results. The presented geometric modeling resources are based on the Half-Edge topological data structure. The system is developed in Python, based on the object-oriented programming paradigm, and uses the Qt user interface system. Modeling is performed through the interactive creation of parametric curves of various types: polygonal lines, quadratic and cubic Bezier curves, composite cubic Bezier curves, circles, circle arcs, ellipses, ellipse arcs, and NURBS (Non-Uniform Rational B-splines). The system automatically recognizes the creation of closed regions.

Keywords: finite element modeling, isoparametric and isogeometric analysis, Python application, educational software.

1 Introduction

Creating geometric models in the context of mesh generation is an essential task when dealing with the numerical simulation of real engineering objects, such as the model shown in Fig. 1, inspired by the book of Bathe [1]. In planar models, such as the one depicted in Fig. 1, it seems natural to represent a solid through an explicit description of its boundary, the so-called boundary representation (B-rep) [2]. This type of representation is based on data structures that describe the adjacency relationships of the vertices, edges, and faces of a solid.

However, numerical simulation requires additional modeling capabilities. For example, the Finite Element Method (FEM) or the Boundary Element Method (BEM) need discrete models, i.e., domain meshes or boundary meshes, respectively, that are generated based on the geometric description of the model's domain. Fig. 2 shows a FEM mesh of the model of Fig. 1 that was generated using an arbitrary triangulation technique [3]. In addition, the construction of the discrete models usually involves decomposing the model's domain into several patches, resulting in a complex multi-region solid. Fig. 3 shows a FEM of the model of Fig. 1 that was generated by

decomposing the domain into quadrilateral patches and using a finite element mesh generation algorithm based on a discrete transfinite mapping [4] in each patch.

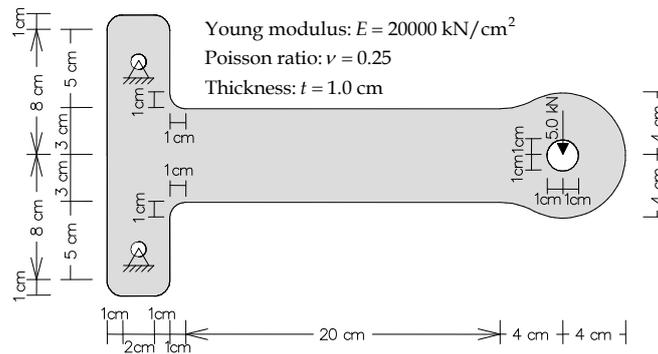


Figure 1. Engineering problem of steel bracket [1]

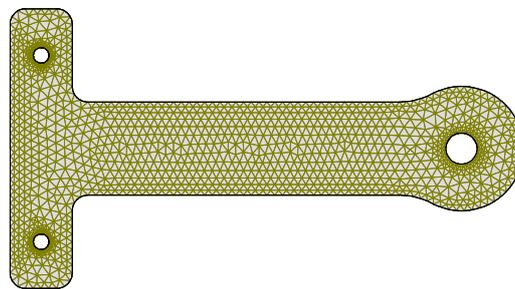


Figure 2. FEM mesh of the model of Fig. 1 generated using an arbitrary triangulation technique [3]

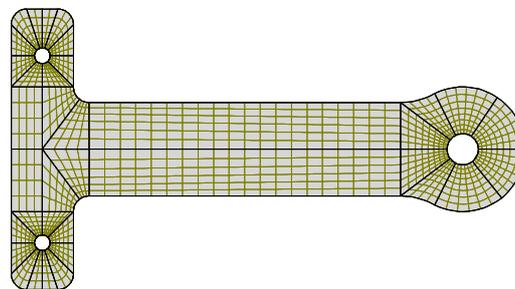


Figure 3. FEM mesh of the model of Fig. 1 generated using a finite element mesh generation algorithm based on a discrete transfinite mapping [4] in each patch

One strategy for two-dimensional modeling objects and creating FEM meshes, such as the ones in Fig. 1, Fig. 2, and Fig. 3, is the direct manipulation of curve segments. Rossignac and O'Connor [5] have proposed the concept of geometric complex, in which an object is decomposed into disjoint portions called cells (curves segments or vertices in the present context), each one homogeneous in dimension and satisfying the condition that the intersection of the boundaries of any two cells is necessarily equal to the union of other cells of the decomposition. Cavalcanti et al. [6] generalized this concept and presented a modeling approach based on spatial subdivision. In this approach, any object is represented by a subdivision of the entire Euclidian space (including the external infinite region). An incremental approach for model creation considers the inclusion of a cell (a curve segment, for example) at a time, with the corresponding updating of the data structure after each insertion. Inserting a curve segment (or vertex) is a fundamental operation executed by the spatial subdivision scheme that may result in the refinement of new and existing cells due to geometric intersection. New surface patches (cells in the decomposition) may result from curve insertions. This approach provides a basis for representing complex objects in two steps: first, a subdivision phase and, second, a phase with a selection of active cells. For instance, in a finite

element analysis of a solid with a cavity, the infinite external region and the cavity would be inactive cells.

In addition to the problem of geometric modeling and meshing, several other issues are essential in a numerical simulation. For example, one needs to define physical model parameters and other modeling attributes. For example, it is necessary to determine material properties, essential boundary conditions (displacement constraints), and natural boundary conditions (loads and other external solicitations). In summary, it is necessary to define and create simulation attributes and associate them with the topological and geometric entities of the model.

An essential issue in the simulation of physical problems is that some characteristics are common to various simulation types and other features are particular to each type of simulation. The common parts are geometric modeling and meshing. The specific aspects are related to the simulation attributes. For example, in a simulation of solid elasticity problems, elastic properties of materials and applied forces are usual attributes. On the other hand, in the simulation of thermal diffusion problems, the properties of the materials will be related to their thermal conductivity, and prescribed temperature or heat flux are usual boundary conditions. Therefore, to be general, a numerical modeler must address generic management of simulation attributes in addition to geometric modeling and meshing issues. The generic management of attributes in a numerical simulation based on geometric modeling is a point that is generally overlooked in other numerical simulation works. This subject is essential for the present research group. Several dissertations and theses from the research group addressed this topic [7-10]. Recently, a published article [11] also addresses this issue.

Numerical simulation of engineering problems also involves analysis and visualization of results. For example, Fig. 4 shows the visualization of the horizontal normal stress component (σ_{xx}) of the model of Fig. 1 for the mesh of Fig.3, displayed on top of the model's deformed configuration with a magnification factor of 50. This work aims to describe some modeling features of an integrated system for two-dimensional finite element simulation of computational mechanics problems. The software addresses geometric modeling and finite element mesh generation (pre-processing), configuration and application of simulation attributes to physical and mesh entities, isoparametric and isogeometric finite element analysis, and graphical post-processing of the results.

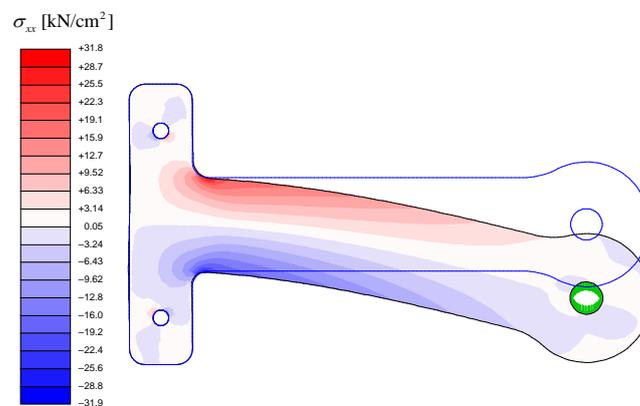


Figure 4. Contour of the horizontal stress component of model of Fig. 1 displayed on top of the model's deformed configuration. Deflections are drawn with a magnification factor of 50 together with the original configuration.

Conventional finite element programs usually implement the so-called isoparametric formulation [1,12]. In this formulation, the geometry of a finite element is interpolated using shape interpolation functions that are the same (iso) as the functions used to interpolate the main field of the numerical problem. Usually, polynomials functions are adopted as shape functions. One of the shortcomings of this type of approach is that the geometry of an isoparametric finite element model is that the geometry approximates the real object used in its original modeling. The original geometry is usually generated by CAD systems and uses parametric representations of curves and surfaces that are different from the shape functions usually adopted in isoparametric finite element modeling. The most commonly used mathematical representation for curves and surfaces in a CAD system is NURBS (Non-Uniform Rational B-Splines). The isogeometric formulation [13] was designed to approximate engineering design and finite element analysis and, while maintaining compatibility with existing practices,

reformulate the simulation process. A fundamental step is to focus on only one geometric model, which can be utilized directly as an analysis model, or from which geometrically precise analysis models can be automatically built. This methodology requires a change from classical finite element analysis to an analysis procedure based on CAD representations.

The objective of the software under development in this work is to provide students and researchers of computational mechanics with an open educational tool for understanding the integration of pre-processing, analysis, and post-processing in numerical simulators based on isoparametric and isogeometric finite element analysis. Geometric modeling is based on the Half-Edge topological data structure [2,10-11]. The system is developed in Python, based on the object-oriented programming paradigm, and uses the Qt user interface system. Modeling is performed through the interactive creation of parametric curves of various types: polygonal lines, quadratic and cubic Bezier curves, composite cubic Bezier curves, circles, circle arcs, ellipses, ellipse arcs, and NURBS (Non-Uniform Rational B-splines). The system automatically recognizes the creation of closed patches (regions of a spatial subdivision). For the generic treatment of the simulation, attributes associated with physics problems are configured by the end-user.

In isogeometric finite element analysis, an essential aspect in geometric modeling for mesh generation is the need to decompose the model domain into patches whose topology allows the generation of structured meshes in its interior. This strategy was used to generate the mesh indicated in Fig. 4. In this case, isoparametric quadrilateral finite elements were generated by mapping algorithms. It is evident that this type of meshing strategy requires intensive modeling intervention by the end user. On the other hand, a great part of the computational mechanics community understands that meshes generated in this way, with quadrilateral finite elements, provide better results than a mesh generated by arbitrary triangulation, such as the one shown in Fig. 3. The advantage of meshes generated by arbitrary triangulation of the model's domain is clear: after defining the model contour discretization, this mesh is automatically generated without user intervention. However, isogeometric analysis imposes a need to decompose the model domain into patches with topology prone to mapping algorithms. The specific objective of this article is to describe some modeling features that allow for domain decomposition into patches ready for isoparametric and isogeometric analysis.

This article is organized into three more sections. The following section describes an automatic hierarchical domain decomposition strategy of quadrilateral regions into patches in which a mapping algorithm may generate a mesh. Section 3 argues about T-spline patches, which have several advantages, and proposes solutions to some problems faced by NURBS patches. T-Spline models require fewer control points, which makes patch decomposition much faster and more efficient. Finally, Section 4 draws some conclusions and addresses future developments for continuing this work.

2 Hierarchical decomposition of regions

As already highlighted, the decomposition of the domain of a model into subdomains is a requirement demanded in isogeometric analysis. In this section, fundamental concepts for the generation of meshes in sub-regions will be discussed, going through definitions regarding the structured or unstructured form of a mesh and a characterization of the hierarchical decomposition method. Some domain decomposition templates will be discussed and used according to verified essential conditions.

The bilinear transfinite mapping is a traditional and widespread method for generating quadrilateral finite element meshes [4]. However, it is noteworthy that this method has some limitations, such as the need to have four boundary curves delimiting the domain, in addition to the fact that opposite curves need to have the same number of edges. Thus, the bilinear transfinite mapping only allows the generation of structured meshes, such as the mesh shown in Fig. 5, formed from 20 edges on the upper and lower curves and 16 edges on the left and right curves.

Quadrilateral structured meshes have a constant valence equal to 4 at the interior nodes. The valence of a node is understood as the number of elements it makes with the nodes in its surroundings. For unstructured meshes, the valence at the internal nodes has variable values, as can be seen in Fig. 6, where some highlighted interior nodes assume a valence equal to three. In contrast, most others have a valency of four. This figure shows a quadrilateral region with an unstructured mesh decomposed into sub-regions with structured meshes. This mesh was created by an algorithm of automatic hierarchical decomposition [14-15] that works for quadrilateral and triangular regions.

Structured meshes are pretty relevant, as they are essential for isogeometric analysis. Therefore, a strategy

that decomposes regions into sub-regions that can be meshed with mapping algorithms is necessary. Furthermore, it is desired that the meshing process occurs for any number of boundary curves and for any number of curve edges. In this sense, mesh generation using the cited algorithm of hierarchical decomposition is divided into three phases. In the first phase, the model's domain is decomposed by the user, using the modeling capabilities of the system, into quadrilateral and triangular regions. In the second phase, the curves on the boundaries of the regions are subdivided into an arbitrary number of edges. In the third phase, an automatic process of dividing the domain into subdomains (or sub-regions) is carried out so that it is possible to generate in these sub-regions a mesh directly by bilinear transfinite mapping.

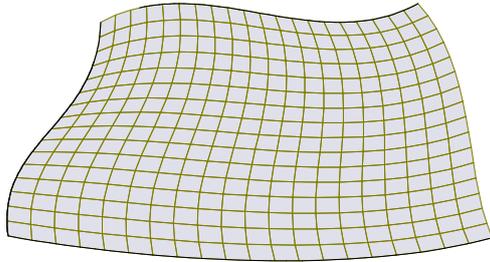


Figure 5. Quadrilateral patch with a FEM mesh generated by a discrete transfinite mapping

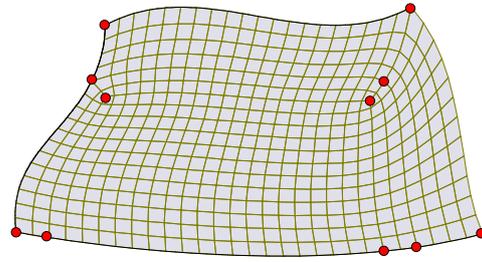


Figure 6. Hierarchical template-based mesh generation in one quadrilateral patch

In Fig. 6, it can be seen that each boundary curve has a different number of edges, 20 in the upper curve, 18 in the lower curve, 16 in the left curve, and 12 in the right curve. Thus, in this domain, forming a structured mesh through bilinear transfinite mapping is impossible. Therefore, the hierarchical decomposition process is applied, dividing the region into sub-regions capable of generating a structured mesh. In the scope of isogeometric analysis, the model geometry should be used in the analysis process. For this purpose, mathematical representations are adopted for the boundary curves of the NURBS type. The most significant advantage of NURBS lies in the fact that they are convenient and adaptive in modeling varied surfaces. In each sub-region, a NURBS surface patch is generated using a bilinear transfinite mapping equation [16].

The hierarchical decomposition of the domain is based on decomposition templates, which consist of how it can be divided, considering particular characteristics. In the present article, domain decomposition templates that result in quadrilateral sub-regions are discussed [14], as shown in Fig. 7.

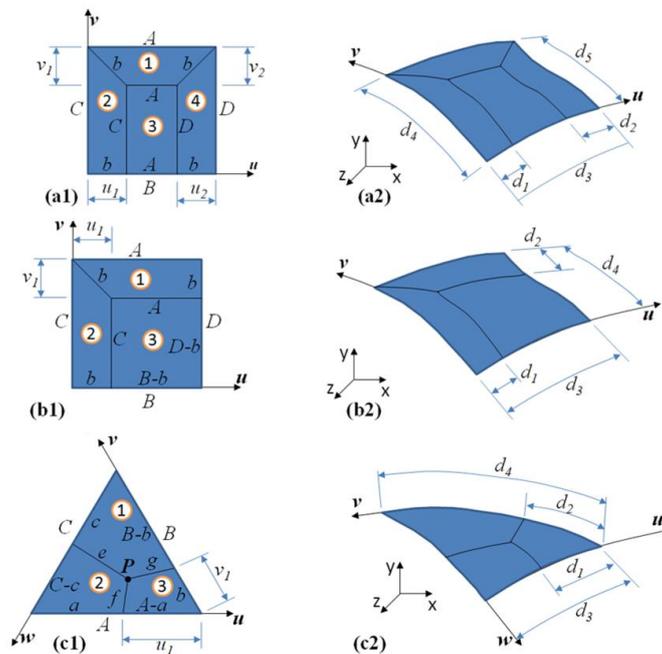


Figure 7. Patch decomposition of templates and Cartesian spaces [14]

The patterns (a1) and (b1) in Fig. 7 are the T1 and T2 templates, respectively, both made up of four curves. In contrast, the pattern (c1) in Fig. 7 is T3, formed by three curves. There is also the T0 pattern, which does not perform region decomposition, making it possible to generate a mesh of quadrilateral elements through transfinite mapping directly. In a simplified way, the hierarchical domain decomposition method comprises successive divisions of the regions through specific templates at different levels until the T0 is reached, from which the mesh generation is performed [14].

A fundamental point to be emphasized is the choice of the decomposition template to be adopted. This aspect fundamentally depends on the number of edges of the boundary curves [14-15]. In the case of template T0, the necessary condition is that the opposite curves have the same number of edges. Thus, it is possible to form a structured mesh, as in Fig. 5.

A good example can be seen in Fig. 6. At the first level, the domain is initially subdivided through the T2 template, forming three subdomains. In one of these sub-regions, the condition of the T0 template was not reached. Therefore, there was a new decomposition at the second level through the T1 template. The final result was the formation of 6 subdomains, subject to applying T0. It is noteworthy that within each subdomain, a structured mesh is formed. However, in a global analysis of the initial domain, it is observed that the resulting mesh assumes the unstructured form. Through this strategy, it is possible to generate a quadrangular mesh for a model with different numbers of edges in the delimiting curves.

3 T-Spline patches

T-spline curves are a more recent generalization of NURBS, which correct deficiencies and propose solutions to existing limitations. It is known that for isogeometric analysis, different technologies of computational geometry can be used in modelling. In the engineering field, NURBS are the most used in projects, gaining the position of industry standard, as they are very convenient to represent a wide variety of geometries [13] accurately. However, NURBS have disadvantages concerning the intersection of surfaces, as gaps and overlaps cannot be avoided [17].

The definition of a NURBS surface occurs through a set of control points inserted in a control mesh of rectangular shape. Thus, many of these control points are only needed to satisfy the topological issues adequately and may not provide new information to the model geometry. In this context, as an alternative, T-splines control meshes appear, which do not need to have a grid format, as in the case of NURBS, but rather T-shaped junctions, as seen in Fig. 8 (left), may be formed. From the generated sub-regions, T-spline patches with structured meshes are obtained, as in the right of Fig. 8. In a model performed by NURBS curves, most control points are redundant. From the designer and engineering point of view, more control points imply more modeling and analysis time and higher costs for a given project [17]. In specific models, the refinement process is often carried out, adding new control points without changing the geometry. It turns out that when refining in NURBS, it is necessary to add a whole line of control points [17]. On the other hand, T-splines allow for local refinement, which is quite advantageous.

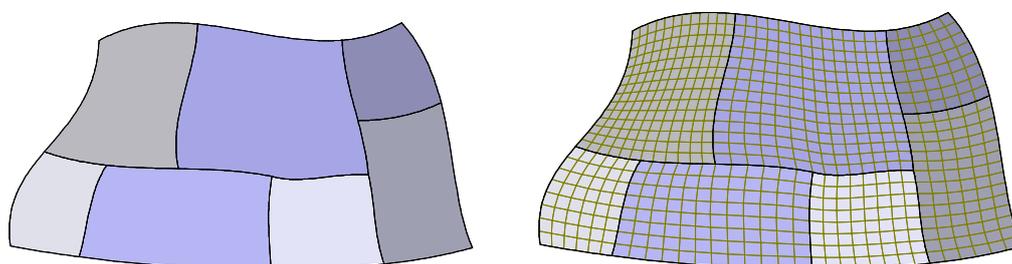


Figure 8. Patch decomposition for T-Spline representation and structured meshes generated in each patch

Another limitation of NURBS models is the intersection of surfaces. As a NURBS surface presents a control mesh with rectangular topology, many of the objects are modeled by joining several surfaces. The intersection is difficult, especially if there are corner points with a valence other than four [17]. With the use of T-splines models, it is possible to merge together with gap-free.

4 Conclusions

The decomposition of the domain is a fundamental strategy for isogeometric analysis. Through the templates described it is possible to perform this step by the hierarchical decomposition method automatically. In each subdomain obtained, it is possible to generate a structured mesh by transfinite mapping. An example was demonstrated showing how the use of these templates was performed during the hierarchical decomposition. It is interesting to note that through this strategy, it is possible to generate a quadrilateral mesh for a model with different numbers of edges in the bounding curves.

Furthermore, it was discussed about the decomposition of the regions through T-spline models, which present a series of solutions and advantages in relation to NURBS Surfaces.

Acknowledgements. The authors acknowledge the financial support from CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) – Finance Code 001, from FAPERJ (in Portuguese “Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro”), and from CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) – Fellowship Productivity in Research of the last author.

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] K.J. Bathe, *Finite Element Procedures in Engineering Analysis*, Prentice-Hall, Englewood Cliffs, New Jersey, 1982.
- [2] M. Mäntylä. *An Introduction to Solid Modeling*. Computer Science Press, 1988.
- [3] A.C.O. Miranda, J.B. Cavalcante Neto, and L.F. Martha, “An Algorithm for Two-dimensional Mesh Generation for Arbitrary Regions with Cracks”. *SIBGRAP'99 – XII Brazilian Symposium on Computer Graphics, Image Processing and Vision, UNICAMP/SBC, Campinas, SP, Out. 1999*, IEEE Computer Society Order Number PRO0481, ISBN 0-7695-0481-7, Eds.: J. Stolfi & C. Tozzi, pp. 29-38, 1999.
- [4] R. Haber, M.S. Shephard, J.F. Abel, R.H. Gallagher, and D.P. Greenberg, “A general two-dimensional graphical finite element preprocessor utilizing discrete transfinite mappings”. *International Journal for Numerical Methods in Engineering*, vol. 17, pp. 1015-1044, 1981.
- [5] J.R. Rossignac and M.A. O'Connor, “A dimensional-independent model for pointsets with internal structures and incomplete boundaries”. *Geometric Modeling for Product Engineering*, North Holland, pp.145-180, 1989.
- [6] P.R. Cavalcanti, P.C.P. Carvalho, and L.F. Martha, “Non-manifold Modeling: An Approach Based on Spatial Subdivision”. *Computer-Aided Design*, vol. 29, n. 3, pp. 209-220, 1997.
- [7] M.T.M. de Carvalho, *Uma Estratégia para Desenvolvimento de Aplicações Configuráveis em Mecânica Computacional*, Ph.D. thesis, Pontifical Catholic University of Rio de Janeiro. 1995.
- [8] E.S.S. da Silveira, *Um Sistema de Modelagem Bidimensional Configurável para Simulação Adaptativa em Mecânica Computacional*, M.Sc. dissertation, Pontifical Catholic University of Rio de Janeiro, 1995.
- [9] W.W.M. Lira, *Um Sistema Integrado Configurável para Simulações em Mecânica Computacional*, M.Sc. dissertation, Pontifical Catholic University of Rio de Janeiro, 1998.
- [10] D.S. Bomfim, *Uma estratégia de modelagem aberta e extensível para criação de modelos de subdivisões planares para mecânica computacional*, M.Sc. dissertation, Pontifical Catholic University of Rio de Janeiro, 2022.
- [11] D.S. Bomfim, R.L. Soares, L.F. Bez, P.C.F. Lopes, A.M.B. Pereira, and L.F. Martha, “Development of a Python Application Aiming at the Teaching-learning process of the Half-Edge Data Structure”, *Proceedings of the joint XLII Ibero-Latin-American Congress on Computational Methods in Engineering and III Pan-American Congress on Computational Mechanics, ABMEC-IACM, Rio de Janeiro, RJ, 2021*.
- [12] R.D. Cook, D.S. Malkus, M.E. Plesha, and R.J. Witt, *Concepts and applications of finite element analysis*, 4th edition, John Wiley & Sons, 2002.
- [13] J.A. Cottrell, T.J.R. Hughes, and Y. Bazilevs, *Isogeometric Analysis: toward integration of CAD and FEA*, John Wiley & Sons, 2009.
- [14] A.C.O. Miranda, L.F. Martha, “Hierarchical template-based quadrilateral mesh generation”. *Engineering with Computers*, vol. 33, n. 4, pp. 701-715, 2017.
- [15] E.Q. Ccapacca, *Construção de padrões para geração de malhas por decomposição hierárquica de domínio*, M.Sc. dissertation, Pontifical Catholic University of Rio de Janeiro, 2015.
- [16] W.J. Gordon, C. Hall, “Transfinite element methods: Blending-function interpolation over arbitrary curved element domains”. *Numerische Mathematik*, vol. 21, n. 2, pp. 109-129, 1973.
- [17] Y. Bazilevs, V.M. Calo, J.A. Cottrell, J.A. Evans, T.J.R. Hughes, S. Lipton, M.A. Scott, T.W. Sederberg, “Isogeometric analysis using T-splines”, *Computer Methods in Applied Mechanics and Engineering*, vol. 199, n. 5-8, pp. 229-263, 2010.