

High-Order discontinuous boundary elements: Formulation, convergence and performance analysis on 3D elasticity

Cristiano J.B. Ubessi, Rogério J. Marczack

*Departamento de Engenharia Mecânica, Universidade Federal do Rio Grande do Sul - UFRGS
Sarmiento Leite, 425, 90046-902, RS, Brazil
cristiano.ubessi@ufrgs.br, rato@mecanica.ufrgs.br*

Abstract. The Boundary Element Method (BEM) is known for its ability to accurately approximate difficult engineering problems in elasticity, such as stress concentration, contact problems, etc. Due to its inherent complexity, element order has been historically kept as low as possible in the literature, being most of the time restricted to linear and quadratic elements, as they present an already adequate precision. Due to Green's function singularities, a large part of the BEM integration process is performed with high-order Gaussian quadratures, such that increasing element order is, in many cases, more efficient than mesh size refinement. Moreover, it was recently highlighted that the singularity subtraction method may impose the usage of high-order elements. This work's objective is to investigate the possible effects of employing a p-refinement on some selected benchmark problems. For that purpose, we present a newly developed library and its formulation for computing arbitrary order shape functions considering continuous or discontinuous, and Lagrangian or Serendipity type QUAD elements. This library (as well as the BEM code) is written in GNU Octave, where the shape functions are automatically generated and tested using a Computer Algebra System. This process avoids the manual implementation of these functions which in general is cumbersome and prone to errors. The meshes for the benchmark cases are generated using GMSH, which can generate arbitrary-order QUAD elements. Solution convergence is analyzed in terms of the L2 error norm. The continuous versus discontinuous element subject is also briefly discussed.

Keywords: Boundary Element Method, High-Order Elements, Discontinuous Serendipity Elements

1 Introduction

The Boundary Element Method (BEM) is known for its ability to accurately approximate difficult engineering problems in elasticity, such as stress concentration, contact problems, etc. Most of its ability comes from its formulation which, similarly to a Trefftz method, employs the exact solution for an auxiliary problem, in this case, Green's function for infinite space. This allows one to write Betti's reciprocal theorem in such a manner that, both potential and its spatial derivatives are involved in the approximation. This property turns BEM attractive to application in many elasticity problems, as surface traction is approximated with the same degree as displacement. To this date, there are very few papers focusing on increasing this element order with the BEM. One of the many interesting BEM features is its ability to exactly calculate stress fields in the domain based only on the results obtained at the boundary. Due to the singularities arising from the Green's function, the integral of these results close to the surface becomes extremely difficult, raising the necessity to employ singularity dealing methods, such as analytical treatment, special coordinate transformations, or subdivision schemes. This difficulty becomes even higher when dealing with thin domains, where singularities are present in many of the elements, not only in the one close to the interesting point, and near-singular integrals also occur, where complex poles of singularity may arise.

The self-regularized formulations enable one to model extremely thin structures with the BEM without these special treatments. This method employs the element shape function derivatives to cancel out the singularity of the fundamental solution. From this perspective, the higher the element order, the higher singularities can be canceled out, which enables one to obtain stress near the surface, which otherwise would require extreme mesh refinements. In this specific case, the issue has nothing to do with the domain relative thickness, but the need to calculate stress fields localized near the surface of an object whose element size is many times larger than this distance. This is a problem that arises not only in the BEM but is also a problem in finite element methods, since they would require localized refinements which are not simple to perform, even when using tetrahedral elements, for instance, see [1].

In contrast with the FEM, which depends on the lowest possible number of element nodes for reducing the sparse matrix bandwidth and memory usage, and in some cases linear elements are preferred, even though its convergence rate is lower, in the BEM this is not the case: the matrices are fully populated, and if the number of equations is the same, there is no solution time or memory difference in storing and solving a system of equations with linear, quadratic, or n th-order elements. Nonetheless, due to vectorization efficiencies, the performance per DOF in the system assembly is usually higher for higher-order elements. Moreover, in a recent paper from [2], the authors demonstrated for potential problems that higher order elements may improve the solution convergence by up to an order of magnitude.

For the above reasons, the investigation of high-order elements in the BEM is justified and considered relevant for several fields of application.

Although standard high-order elements can be easily formulated by using the Lagrange polynomials, the formulation for incomplete polynomial (or serendipity type) elements of a higher order than two, to the best of our knowledge, is not readily found in the literature. The discontinuous shape functions for these elements, therefore, is another contribution of this research.

This paper is structured as follows: First, the basic BEM equations are presented, and then, a brief discussion on the theme of why discontinuous elements exist is presented. Then, the main motivation for this work, a detailed presentation is made on the formulation of standard Lagrangian and Serendipity type shape functions of arbitrary order. In the following sections, two kinds of study are performed: The first one is on the integration of the high-order elements. The second part of the study focuses on the actual solution of an elasticity problem using incomplete elements from order 1 to 6, and the results are compared in with linear and quadratic continuous elements from [3]. Some conclusions and insights for future research are drawn at the end of the manuscript.

1.1 Boundary integral equations

In this section, we briefly describe the basic equations from which the BEM is formulated. Let Γ be the boundary of a domain Ω . One can write the displacement boundary integral equation for the elastic problem as,

$$\mathbf{C}\mathbf{u} + \int_{\Gamma} \mathbf{T} \mathbf{u} \, d\Gamma = \int_{\Gamma} \mathbf{U} \mathbf{t} \, d\Gamma \quad (1)$$

where \mathbf{T} and \mathbf{U} are the BEM usual fundamental solutions, (e.g.: See [4], for instance). $\int(\cdot) \, d\Gamma$ denotes the Cauchy Principal Value of integral $\int(\cdot) \, d\Gamma$. After the boundary Γ is discretized with finite elements, eq. (1) for a i and field element e results into,

$$\mathbf{C}^i \mathbf{u}^i + \sum_{e=1}^N \tilde{\mathbf{H}}^{ie} \mathbf{u}^e = \sum_{e=1}^N \mathbf{G}^{ie} \mathbf{t}^e, \quad (2)$$

where $\tilde{\mathbf{H}}_i^e$ and \mathbf{G}_i^e for a source point i and field element e is calculated as:

$$\tilde{\mathbf{H}}^{ie} = \text{CPV} \int_{\Gamma^e} \mathbf{T} \bar{\phi} \, d\Gamma^e, \quad \mathbf{G}^{ie} = \int_{\Gamma^e} \mathbf{U} \bar{\phi} \, d\Gamma^e. \quad (3)$$

Performing a collocation process for all source points results in the following system of equations

$$\mathbf{H}\mathbf{u} = \mathbf{G}\mathbf{t}. \quad (4)$$

Switching the columns of \mathbf{H} and \mathbf{G} corresponding to the unknowns to the LHS forms the coefficients matrix \mathbf{A} , and multiplying the boundary conditions by the resulting RHS matrix, forms the RHS vector \mathbf{b} , as is usual in BEM [5], one obtains the solution for the unknown displacement or traction along the boundary.

1.2 The corner dilemma

Perhaps the main reason for adopting discontinuous elements is the fact that since the right-hand side of eq. (4) multiplies \mathbf{t}^e , which comes from the Cauchy formula $\mathbf{t} = \sigma \mathbf{n}$, if a collocation point is located at a corner, i.e., a node whose connected elements have different normal vectors, the surface traction at this node will result in six different components if two elements are sharing the same node, while there will be only three displacement components at this node. Analyzing this situation from an algebraic perspective, two possible scenarios can arise: If the boundary conditions at these elements are of prescribed traction only, these components will multiply the corresponding \mathbf{G} matrix, which has size $[3n \times 3nbc]$, and their contribution will be summed in the correctly sized right-hand side vector, with the same size as the unknown displacement components at this node.

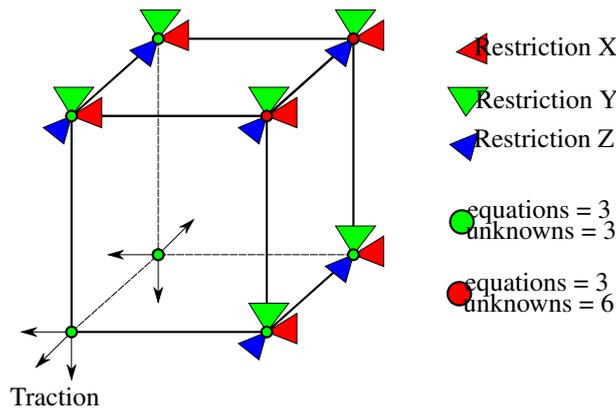


Figure 1. A possible situation that may generate excessive traction unknowns at corners.

If this node has prescribed displacements, there can be more scenarios in development therein; for now, let us take a unit cube Fig. 1, discretized using six elements, one for each face as an example, and let the (0,0,0) corner be the 'problem' node. If one restricts this node in all directions, the unknown tractions will depend on the boundary conditions at the adjacent elements, and not on the node itself. If just one of the three elements forming this corner has its three displacement directions the number of unknown traction will be 3. If two or three elements forming this corner have all their displacements as BC, the number of unknown traction components will be 6 or 9. Therefore, in the latter, the entire stress tensor is unknown at this point, while only 3 displacement components are known.

There are several manners of dealing with this situation, many of them are listed in [3], but none of them seems to yield a definitive solution for the problem, as each one has its benefits and caveats.

For this reason, many researchers have adopted the usage of discontinuous elements in some form or another. Moving the collocation points to the interior of the element, at an offset from the element edges, the number of unknown traction and displacement components is always the same, as there are no more collocation points shared between elements. The element variable interpolation then adopts a discontinuous version of the shape functions, in which the Lagrange polynomial is modified to yield one or zero at the collocation points, not at the element corners. These points are generated from the standard mesh using the shape functions, by scaling down the intrinsic coordinates by a factor of $f = 1 - d_0$, with d_0 being a parameter that can vary from 0.05 to 0.5 element units. Since the collocation nodes are generated from the mesh nodes and stored in sequential order, no connectivity or incidence matrix is needed for the system assembly, as each element has its independent degrees of freedom (the collocation points). The results at the corners can then be extrapolated in post-processing if needed. This simplifies the system assembly because all the system nodes or degrees of freedom become independent of the assembly of different element contributions.

2 Lagrangian and serendipity elements

This section hopefully describes the necessary shape function formulation, which, in the author's opinion, is not found in any literature with the same clarity and usefulness as we wish to present. Therefore, this presentation is presented not only from a didactic perspective but also from the computational implementation standpoint, as it can serve as a guide for the implemented code. The standard scheme for generating the 2D shape functions is described as follows.

Let the standard element be composed of two intrinsic coordinates ξ and η , and have the same number of nodes nd ranging from 1 to nd in each direction, providing interpolation with complete polynomials of order $n = nd - 1$, with the total number of nodes being nd^2 . Let us define the Lagrangian matrices A and B , as

$$A_{ik}(\xi) = \frac{\xi - \bar{\xi}_k}{\bar{\xi}_i - \bar{\xi}_k}, \quad B_{ik}(\eta) = \frac{\eta - \bar{\eta}_k}{\bar{\eta}_i - \bar{\eta}_k} \quad (5)$$

where $\bar{\xi}_k$ and $\bar{\eta}_k$ are each of the intrinsic coordinates for the k -th node. To obtain the discontinuous shape functions, one needs to replace the standard intrinsic coordinates with the discontinuous ones, for instance, in the case of a 3-rd order element, $\bar{\eta} = (-1, -1/3, 1/3, 1)$ with $\bar{\eta} = (-(1 - d_0), -(1 - d_0)/3, (1 - d_0)/3, (1 - d_0))$.

If complete polynomials (standard Lagrangian elements) are used, then one can write the following equation for the shape function of the node at position jk .

$$\phi(\xi, \eta)^{jk} = \prod_{i=1, i \neq j}^{nd} A_{in}(\xi) \prod_{l=1, l \neq k}^{nd} B_{lk}(\eta) \quad (6)$$

Notice that, this grid-like structure of nodes makes it simpler to calculate the shape functions for general orders, but does not takes into account the commonly found linear node numbering in FEM and mesh generation codes. For this reason, a connectivity-type matrix is assembled which correlates the GMSH mesh generation software [6] node ordering with the corresponding j, k indices. This method also facilitates the computation of incomplete polynomial elements, which are described next.

To calculate the incomplete or serendipity element shape functions, there is a sequence of steps that must be followed. First, the four corner shape functions are obtained through eq. (6), assuming a linear element ($nd = 4$).

Secondly, for each of the additional nodes ($n = 5 : nd$), the following consideration must be taken into account: At the element edge where the node lies, all the neighbor nodes exist (case 1), while for the other intrinsic direction, perpendicular to the edge where the current node lies, there is only one node at the opposite element edge, aligned with the current node (case 2). Therefore, due to this lack of nodes inside the element, the product of eq. (6) must be modified as follows:

$$\phi(\xi, \eta)^{jk} = \begin{cases} \prod_{i=\mathcal{S}(\xi(j))}^{nd} A_{ij}(\xi) \prod_{l=\mathcal{S}(\eta(k))}^{nd} B_{lk}(\eta) \end{cases} \quad (7)$$

with $\mathcal{S}(\xi(j))$ being the set of all co-linear nodes with node j , except for j , i.e.:

$$\mathcal{S}(\xi(j)) \begin{cases} [1, j] \cup (j, n] & \text{if } 1 < j < n, \\ n & \text{if } j = 1, \\ 1 & \text{if } j = n, \end{cases} \quad (8)$$

and $\mathcal{S}_{\eta(k)}$ following the same scheme. Notice that the third case of eq. (8) will be true for the edge direction, while the first two cases will be true for the opposite side node direction. Notice also that this rule is valid only for the edge nodes (not for the 4 corner nodes). By doing so, one generates the so-called incomplete shape functions, as they are linear in the direction where the mid-side nodes are missing, but of order (n) in the direction of the edge containing the node.

After calculation or determination of eq. (7) for node j , the two adjacent corner nodes in the same edge as node j must have their shape functions modified, by adding node j shape function, weighted such that the corner node function results null at node j position. This summation, in turn, enhances the corner node's shape function with the incomplete functions of the edge nodes.

Each new mid-side node shape function must be recursively added to the shape functions of the adjacent corners, as follows:

$$\phi(\xi, \eta)^{rs} = \phi(\xi, \eta)^{rs} - f_{jk}^{rs} \phi^{jk}(\xi, \eta) \quad (9)$$

where $[r, s]$, the position of each one of the adjacent corner nodes, depends on j and k , as follows:

$$\{[r1, s1], [r2, s2]\} = \begin{cases} \{[j, 1], [j, nd]\} & \text{if } j = 1 \text{ or } j = nd \\ \{[1, k], [nd, k]\} & \text{if } k = 1 \text{ or } k = nd \end{cases} \quad (10)$$

The multiplication factor needed to null out the original shape function rs at the midside node jk is obtained evaluating the non-modified linear function of the corner node rs at the position $\bar{\xi}^j, \bar{\eta}^k$, i.e., the current midside node, as follows:

$$f_{jk}^{rs} = \phi^{rs}(\bar{\xi}^j, \bar{\eta}^k) \quad (11)$$

Shape function generator and exporter

To provide closed-form shape functions in the same fashion as hand-written ones, we chose to develop a shape function generator and exporter function, which creates separate Octave function files for each element type. It can easily generate Fortran versions for these shape functions. The functions are exported in a manner that

enhances vectorization, allowing one to calculate both the shape functions and their derivatives for a complete set of integration points. The generated shape functions are vectorized in the sense that a complete set of intrinsic coordinates (usually integration points) can be passed into the shape function, and the complete set is returned, which in turn, facilitates the vectorization of numerical integration routines.

Each shape function routine has the same name as the corresponding GMSH element name, so the cross-correlation between those is direct both in the generation and its usage. After the shape functions and their derivatives are symbolically computed, they are tested for obedience to rules such as being unity at the corresponding node and zero at all other nodes. The derivatives are tested by calculating the Jacobian and the normal vector.

Integration rules

As a form to have correct order quadrature rules for integrating the various combinations of Green's function and element shape functions, the following study was carried out: An element of standard size was placed on the xy plane, centered at $(0, 0)$ (Fig. 2). Then, a set of exponentially distributed points was placed along the z axis, at $[0, 1, z]$, with $z = 1 \times 10^{-3} : 1 \times 10^3$.

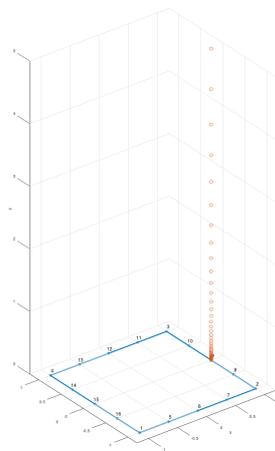


Figure 2. Element (4th order example) and source points considered in the integration test.

Instead of using some random singular function, we chose to integrate the standard kernel for the computation of the H matrix for all nodes of this element, i.e., $\mathbf{H}_{ij}^{(k)} = \int_{\Gamma^e} T_{ij} N^{(k)}$. The error considered for this analysis is calculated by evaluating the same integrals using a 256-point quadrature rule, and, since \mathbf{H} is a set of matrices for all nodes, we chose to take the Euclidean norm of the error, through the following equation:

$$\|e\|_{int} = \sum \frac{\|\mathbf{H}^{(k)} - \mathbf{H}^{REF}\|}{\|\mathbf{H}^{REF}\|}. \quad (12)$$

Fig. 3(a) shows the integral convergence for the second-order element, and Fig. 3(b) for the sixth-order element. It is possible to observe that for large relative distances the 3-point rule is sufficient for the former, while this rule cannot be used for the latter under any circumstance. It is also possible to notice that for closer distances and where the kernel singularity needs a higher number of integration points, the convergence is similar for both elements.

3 Convergence results on classical elasticity problem

For evaluating the mesh convergence a relatively short beam is considered, with length = 10, width = 1, and height = 2, In order to simplify the analytical solution, a pure moment load was applied by considering a linearly varying surface traction at $x_3 = L$, with $t_x = \sigma_{xx} n_x$, Fig. 4. Notice that, due to the necessity of completely restricting (hinge) the beam support, as well as the model being three-dimensional, it is not possible to exactly reproduce this problem solution using only the standard beam theories, so a highly refined, converged FEM solution is used as a reference for the convergence analysis.

The maximum displacement at the beam tip is presented in Fig. 5(a) and the convergence for error in terms of the L2 norm is shown in Fig. 5(b). The norm is calculated as:

$$\|e\|_{L2} = \frac{\int_{\Gamma} |u_z - u_z^{(ref)}|}{\int_{\Gamma} |u_z^{(ref)}|} \quad (13)$$

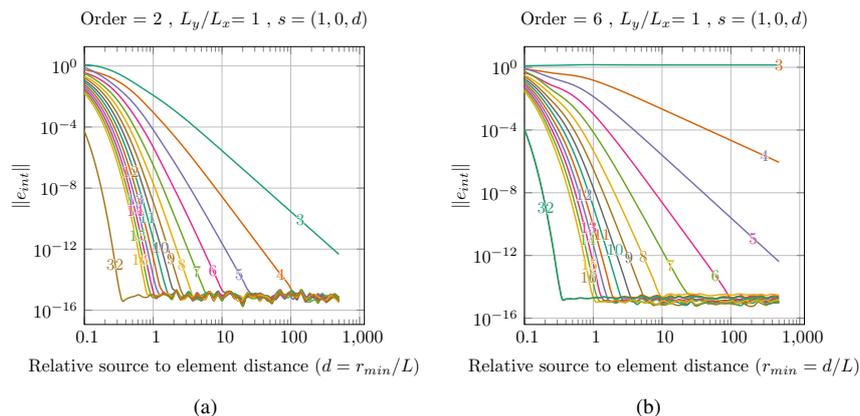


Figure 3. Relative error norm e for varying source point relative distances to the reference square element. Plot labels refer to the Gaussian quadrature rule in each element direction.

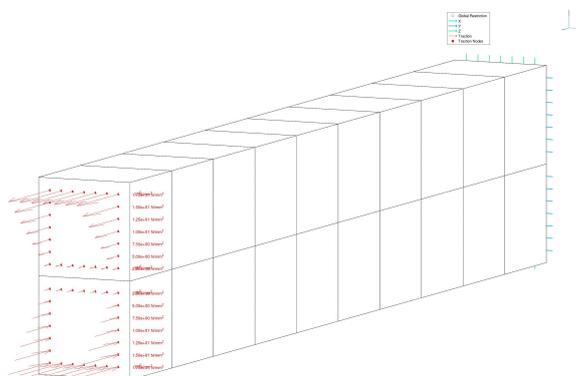


Figure 4. Pure moment boundary conditions for the 6th-order serendipity element at the collocation nodes.

In Fig. 5(b), is possible to observe the convergence oscillations where the refinement stretches some of the elements. Several discontinuity factors were tested, ranging from 2.5 to 20 % of the element’s characteristic length, and only the best-performing ones for each element type are included in Fig. 5(a) and Fig. 5(b).

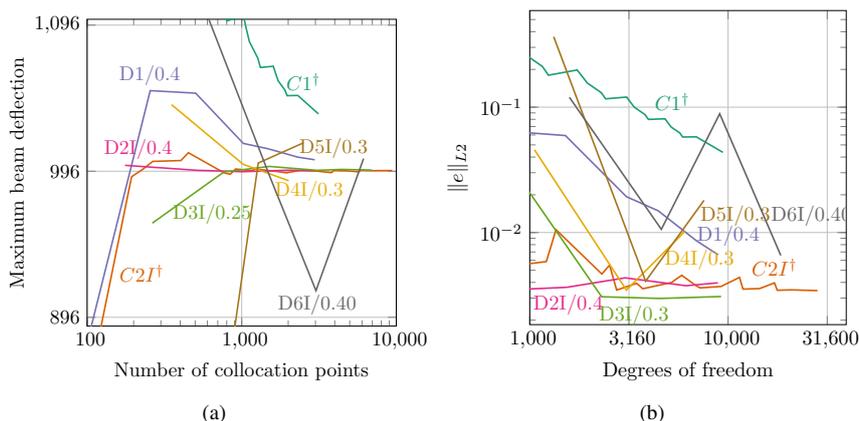


Figure 5. (a) Displacement at the beam tip for different cases analyzed (Incomplete elements only). (b) Relative error norm $\|e\|_{L_2}$. † - Continuous 4 and 8 node elements from [3] sample program. Element codes: C = continuous. D = discontinuous. n = order n. I = incomplete. e.g.: D3I/0.3 = discontinuous, 3rd-order, incomplete, $d_0 = 0.3$.

The main conclusion that can be drawn from this preliminary study is that discontinuous elements can provide better performance under the same system size. Although the minimum system size possible is higher, by selecting appropriate discontinuity factors, the discontinuous elements can yield faster convergence than the continuous ones. Although in this prismatic problem this is feasible, there may be cases where more elements are needed

for geometric representation, then using continuous elements may alleviate system size. Another remark is that, when mesh refinements caused some elements to be stretched, the convergence rate for this problem has oscillated, which indicates some form of sensitivity to the element stretch, or to the distribution of collocation nodes such as is seen in other similar methods [3], which must be further investigated. The elements with higher order than 4 also present oscillatory convergence in the problem considered, which also needs a more profound analysis.

The results presented herein consider only the incomplete or serendipity type elements, the complete interpolation element results will be included in a forthcoming paper, as well as other aspects such as computational efficiency, and the system matrix conditioning, not included in this paper.

Further development is also needed to evaluate the convergence rate of the continuous higher order elements, since these are not implemented by the authors, nor in any other BEM elasticity software to the best of the author's knowledge.

Other elasticity problems whose analytical solution may be better suited for the convergence analysis are also under investigation and should be featured in a forthcoming paper.

Acknowledgements. CJBU is grateful to CNPq for previously provided doctoral scholarship and to CAPES for previously provided PDSE scholarship. RJM is grateful to CAPES and CNPq for their financial support through grant 317140/2021-3.

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] C. Richter, U. Krupp, M. Zeißig, and G. Telljohann. Plastic effects on high cycle fatigue at the edge of contact of turbine blade fixtures. *Journal of Engineering for Gas Turbines and Power*, vol. 140, n. 4, pp. 042501, 2018.
- [2] J. P. Marshall and J. Richardson. A three-dimensional, p-version bem: High-order refinement leveraged through regularization. *Engineering Analysis with Boundary Elements*, vol. 122, pp. 13–20, 2021.
- [3] G. Beer, I. Smith, and C. Duenser. *The Boundary Element Method with Programming*. Springer Wien, New York, 2008.
- [4] F. C. Buroni, C. J. B. Ubessi, G. Hattori, A. Saez, and R. J. Marczak. A fast and non-degenerate scheme for the evaluation of the 3D fundamental solution and its derivatives for fully anisotropic magneto-electro-elastic materials. *Engineering Analysis with Boundary Elements*, vol. 105, pp. 94–103, 2019.
- [5] C. A. Brebbia, J. C. F. Telles, and L. C. Wrobel. *Boundary element techniques: theory and applications in engineering*. Springer Science & Business Media, 2012.
- [6] C. Geuzaine and J.-F. Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, vol. 79, n. 11, pp. 1309–1331, 2009.