

Document downloaded from:

<http://hdl.handle.net/10251/133375>

This paper must be cited as:

Marco, O.; Ródenas, J.J.; Fuenmayor Fernández, F.; Tur Valiente, M. (2018). An extension of shape sensitivity analysis to an immersed boundary method based on Cartesian grids. *Computational Mechanics*. 62(4):701-723. <https://doi.org/10.1007/s00466-017-1522-0>



The final publication is available at

<https://doi.org/10.1007/s00466-017-1522-0>

Copyright Springer-Verlag

Additional Information

An extension of shape sensitivity analysis to an Immersed Boundary Method based on Cartesian grids

Onofre Marco · Juan José Ródenas ·
Francisco Javier Fuenmayor · Manuel Tur

Received: date / Accepted: date

Abstract Gradient-based shape optimization processes of mechanical components require the gradients (sensitivity) of the magnitudes of interest to be calculated with sufficient accuracy. The aim of this study was to develop algorithms for the calculation of shape sensitivities considering geometric representation by parametric surfaces (i.e. NURBS or T-splines) using 3D Cartesian h -adapted meshes independent of geometry. A formulation of shape sensitivities was developed for an environment based on Cartesian meshes independent of geometry, which implies, for instance, the need to take into account the special treatment of boundary conditions imposed in non body-fitted meshes. The immersed boundary framework required to implement new methods of velocity field generation, which have a primary role in the integration of both the theoretical concepts and the discretization tools in shape design optimization. Examples of elastic problems with three-dimensional components are given to demonstrate the efficiency of the algorithms.

Keywords Cartesian Grid-FEM · sensitivity analysis · velocity field · NURBS

1 Introduction

In optimal structural design, sensitivity analysis is the calculation of the derivatives of structural response (displacements, stresses, natural frequencies, etc.) with respect to design variables. The initial development of sensitivity analysis focused on the size design variables, i.e. thickness or cross-sectional areas of structural components, etc. In many structural problems it is necessary to consider shape as a design variable. This is particularly important in the optimal design of structural components. This paper will focus on the analysis of the sensitivities of the design variables that describe the geometry of the component to be optimized.

O. Marco · J.J. Ródenas · F.J. Fuenmayor · M. Tur
Centro de Investigación en Ingeniería Mecánica (CIIM), Universitat Politècnica de València,
Valencia 46022, Spain
E-mail: onofremarco@gmail.com

A large number of references have been published in the field of sensitivity analysis in shape design, especially during the 90s. Currently, the research efforts in the topic are focused on new implementations of the well-known approaches and their application to new problems. In a brief overview, four different approaches can be distinguished:

1. Global finite differences [1, 24, 34, 75]: finite difference expressions are used to obtain the derivatives from the output of repeated Finite Element Analysis (FEA) when small perturbations of the design variables are introduced.
2. Continuum approach. [2, 3, 12, 36, 37, 49, 55]. Derivatives are obtained differentiating the governing elasticity equations. For shape design variables, the two main approaches are the material derivative and the control volume approach. These relate changes in the geometrical shape with the structural characteristics leading to a set of continuum sensitivity equations that are then discretized and solved.
3. Discrete approach [32, 46, 53, 57]. The procedure derivation-to-discretization is reversed and the components of the discretized system of equations are differentiated with respect to the design variables.
4. Computational differentiation [52, 73] is related to the automatic differentiation of the routines within the computational code[6, 22, 68].

The different approaches can be evaluated from the point of view of accuracy, their relation to discretization and cost in computational and implementation terms. Their relationships and comparisons can be found in [13, 25, 26, 31, 50].

Gradient-based optimization requires the evaluation of the sensitivities to drive the optimization process. In shape optimization, this involves adapting or regenerating the Finite Element (FE) mesh for the different geometries to run the numerical simulation of each of these geometries. Reference [63] showed that the behavior of the optimization algorithm is strongly influenced by the accuracy of the results used to drive the process (objective function, constraints and their derivatives). Any inaccuracy in these results can pollute the behavior of the optimization algorithm and reduce the convergence rate to the optimal solution, induce the convergence to a non-optimal or unfeasible solution or even prevent convergence. This requires high-quality FE analyses that can involve a considerable computational cost for each geometry and hence of the overall optimization process. The analysis cost of each geometry can be partially alleviated by the use of adaptive analysis techniques, which are intended to provide the optimal cost-effective FE models to obtain numerical results of the prescribed accuracy. Numerous methods of alleviating the mesh burden are reviewed in [40].

One of the options is the immersed boundary approach[43, 54, 76], which is a natural platform for structural shape optimization processes because its properties are suited to simplifying the mesh generation stage. Immersed Boundary Methods (IBM) have been studied by a number of authors for a wide range of problems such as shape optimization[27, 35] or bio-mechanics, see for instance[20, 44]. An example of this type of approach can be seen in Figure 1. The geometrically complex domain, Ω_{Phys} , is embedded into a geometrically simpler domain, Ω , see Figure 1a. The embedding domain is often simply a cube (or a rectangular cuboid in general) that can be effortlessly discretized using a Cartesian mesh made out of hexahedra to create what we call an approximation mesh, Ω_{Approx} , see Figure

1b. During the integration step, only the internal elements and the internal part of the elements cut by the boundary will be considered (Figure 1c).

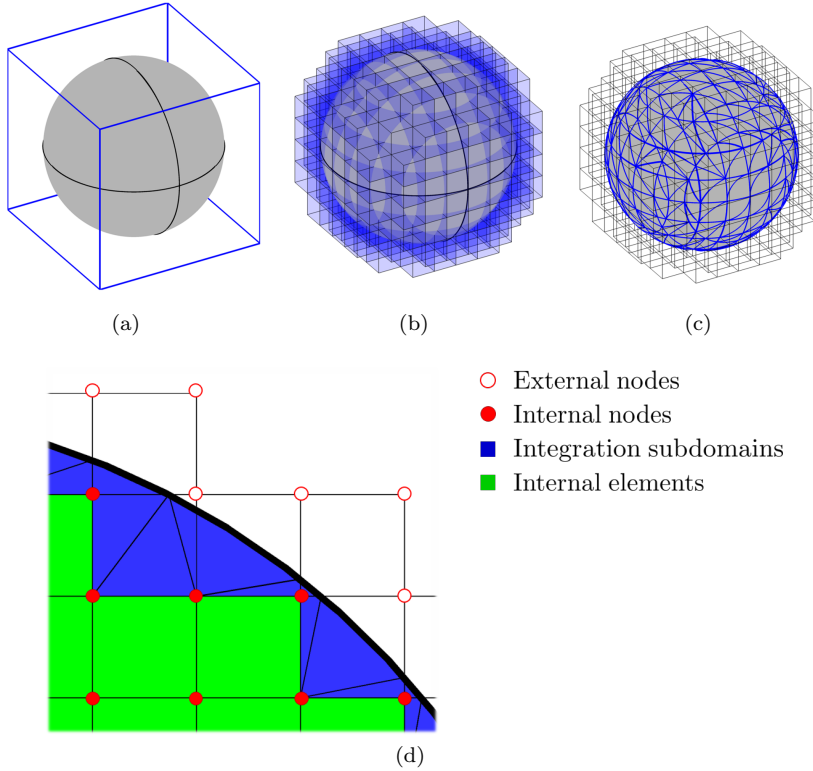


Fig. 1: Typical Immersed Boundary Method environment. (a) Model Ω_{phys} within the embedding domain, Ω . (b) Discretization Ω_{Approx} of the embedding domain. (c) Integration domain, Ω_{Approx}^h and (d) detail of a section.

Another approach is to improve the geometrical accuracy of the models by integrating CAD representations with the FEM codes. Isogeometric Analysis (IGA)[28, 51] is a recent trend in this direction. The main idea is that the meshing procedure is circumvented since an existing CAD geometry is directly used for analysis, all the while keeping the exact geometry. However, in its finite element form, generating an analysis-suitable solid discretization is an open topic[17, 42, 78]. In order to bypass the internal domain parametrization, Boundary Element Methods have also been used in shape optimization [39, 41]. Studies on IGA sensitivity analysis can be found in [9, 23, 58].

The NURBS-enhanced Finite Element Method (NEFEM) [66, 67] employs NURBS for the geometric representation of the boundary, whilst maintaining the flexibility of FEM by using polynomial interpolation. NEFEM conveniently merges the accurate representation of the geometry alleviating the difficulty of generating

interior isogeometric elements. However, mesh generation in NEFEM presents a difficulty similar to that found in FEM.

In [47, 48] we introduced cgFEM (Cartesian Grid FEM) as an alternative to solve these drawbacks. The 3D version of this methodology, based on the use of Cartesian grids independent of the geometry, was implemented in a computer code, named FEAVox [45], for the structural analysis of components. cgFEM is distinguished from other immersed approaches by considering the exact CAD representation of the boundary of the domain by means of the use of the NURBS-enhanced integration techniques [66] to perform the numerical integration over the true computational domain.

In the approach proposed here, the properties related to all the immersed methods (automatic domain discretization, creation of hierarchical data structures for simple data transfer and re-use of calculations, etc.) are merged with the ability to consider the exact geometrical representation, instead of simplifying the embedded boundary (for instance using triangular facets for its definition).

From this point of view, we propose the implementation of sensitivity analysis in cgFEM, exploiting the features of our embedded methodology, aiming for the efficient calculation of sensitivities to reduce the computational cost of the optimization process. The strategy proposed in this paper belongs to the group of techniques denoted as *discrete semi-analytical*. This specification, in our case, means that some of the discrete derivatives rely on analytical derivation and some on finite difference approximations. We use finite differences in order to differentiate the nodal locations with respect to design variables, the so-called velocity field, which is a challenging issue considering the immersed nature of the cgFEM.

The paper is organized as follows: the generation of velocity fields, both on the boundary and inside the domain, is addressed in Section 2. The formulation of shape sensitivity analysis using an immersed boundary approach is described in Section 3. The numerical results showing the performance of the proposed technique are given in Section 4 and the conclusions are reported in Section 5.

2 Design velocity fields

Sensitivity analysis is intended to find the change in the magnitude of response (displacements, stresses, etc.) with respect to design variables. In the case of shape design problems, the position of the material points depends on the design variables. Defining \mathbf{a} as the vector of design variables, in a previous step, the evaluation of shape sensitivities defines how to vary the position of material points of the domain in relation to the design variables, i.e. the sensitivity of the coordinates of the material points, usually called velocity fields, which for an arbitrary design variable a_m is defined as:

$$\mathbf{V}_m = \frac{\partial \mathbf{p}}{\partial a_m} \quad (1)$$

The quality of the velocity field influences the accuracy of the numerical solution, which determines the effective convergence rate of the gradient-based optimization algorithms. The analysis domain will be usually defined by a parametric description of its boundary as a function of the design variables. Differentiating

this function with respect to the design variables will provide the velocity field on the boundary, but there is no closed conformation of this field in the interior.

The determination of the velocity fields is based on the theoretical features of the sensitivity expressions and practical requirements obtained from the features of the FE solution [10, 11]. Theoretically, the velocity field should have the same regularity as the displacements field and depend linearly on the alteration of the design variables. In practical terms, different applications can also impose certain practical additional requirements on the velocity field, such as the need to maintain the mesh topology, to provide FE nodes necessarily located on the boundary of the domain, to produce non-distorted meshes, to be naturally related with the design parameters of the CAD models or to be efficient and general.

Magnitudes like the sensitivity of the strain energy are not affected by the values of the velocity fields in the interior of the domain, provided the velocity fields meet the theoretical requirements and the exact structural response is used in the evaluation of this sensitivity. However, in practice, the FE approximation will be used instead of the exact structural response. As a consequence of this, the final sensitivity of the strain energy will be affected by the velocity fields considered in the interior of the domain[61]. In fact, the stability of the sensitivity of the strain energy can be used to assess the quality of the different techniques that can be used to define the design velocity field[61].

As mentioned above, the design velocity fields must be defined in the whole domain. To do this, the velocity field is usually defined at the nodes of the FE mesh and then interpolated using the shape functions used to interpolate the displacements, so that the velocity fields and the displacements field will have equivalent regularity. In the following subsections we will first describe a procedure to define the design velocity fields along the boundary of the domain, followed by the procedure to define it inside the domain. We will consider the special characteristics of cgFEM aimed at the development of an efficient procedure for shape sensitivity analysis.

Some of the methods found in the literature for velocity field definition are following:

- Finite Difference (FD) method[74]. This method defines the parametric nodal positions on the boundary and evaluates the change of the position due to a perturbation of the design variables using an FD scheme. After that, an interpolation technique has to be used to give the values of the velocity field to the internal nodes.
- Structured meshes[29]. These methods are based on the rules to generate structured meshes that provide a formulation for the position of internal nodes as a function of the boundary nodes.
- Boundary elements method[5]. This strategy considers a null velocity field on the domain except in the subdomain defined by the boundary elements. This reduces the shape sensitivity calculations to a small fraction of the domain.
- Physical approach[4]. This approach defines the evaluation of the velocity field as an equivalent linear elasticity problem where the velocity field evaluated on the boundary is considered as displacements applied on the boundary. Solving the elasticity problem will provide the displacements in the interior of the domain that will be interpreted as velocity field.

- Laplacian method[8]. As in the previous case, the velocity field is considered analogous to a displacement field. The velocity field on the boundary is considered as a perturbation (displacement) of the boundary. Laplacian smoothing is then used to improve the nodal positions. The final displacements of the nodes will be considered as the velocity field at each node.
- Domain triangulation method[61]. This method uses an initial step of the Delaunay triangulation procedure, where the nodes of the triangulation are only placed on the boundary, to interpolate the velocity field through the domain.

Comparative studies of some of these methods can be found in [10, 11, 60, 77].

Due to the nature of IBM, the above-described algorithms cannot be used directly in an IBM context. In this paper we will therefore discuss some alternatives that can be used to generate adequate velocity fields for a Cartesian grid framework, taking into account the features of embedded methods.

2.1 Generation of boundary velocity fields

NURBS (Non-Uniform Rational B-Spline) curves and surfaces [56, 64] were used in the present study to describe the boundary of 2D and 3D domains. Existing works in the literature show the use of NURBS for sensitivity analysis have been used both as an analysis tool, i.e. Isogeometric Analysis[9, 33] and for the geometric description of the models[7, 65, 70].

A rational B-spline curve is given by

$$\mathbf{C}(\lambda) = \frac{\sum_{i=1}^n N_i^{(p)}(\lambda) w_i \mathbf{P}_i}{\sum_{i=1}^n w_i N_i^{(p)}(\lambda)} \quad (2)$$

Here \mathbf{P}_i are the n control points given in d -dimensional space \mathbb{R}^d , w_i is referred to as the i -th weight, typically $w_i \geq 0 \forall i$ and $N_i^{(p)}(\lambda)$ are normalized B-spline basis functions of order p , which are defined recursively as

$$N_i^{(0)}(\lambda) = \begin{cases} 1 & \lambda_i \leq \lambda \leq \lambda_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

$$N_i^{(q)}(\lambda) = \frac{(\lambda - \lambda_i) N_i^{(q-1)}(\lambda)}{\lambda_{i+q} - \lambda_i} + \frac{(\lambda_{i+q+1} - \lambda) N_{\lambda+1}^{(q-1)}(\lambda)}{\lambda_{i+q+1} - \lambda_{i+1}} \quad (4)$$

for $q = 1, \dots, p$ and $i = 1, \dots, n$, where λ_i are the knots, which are assumed ordered $0 \leq \lambda_i \leq \lambda_{i+1} \leq 1$, forming the so-called knot vector

$$A = \{\lambda_1, \dots, \lambda_{n+p+1}\}, \quad (5)$$

which uniquely describes the B-spline basis functions. The multiplicity of a knot, i.e. the number of times it is repeated in the knot vector, determines the decrease in the number of continuous derivatives, so they are \mathcal{C}^{p-1} -continuous where the knots are not repeated. If a knot has multiplicity k , the basis is \mathcal{C}^{p-k} -continuous at that knot. Other properties of the basis functions can be found in [56, 64].

NURBS surfaces are obtained from a tensor product through two knot vectors $\Xi = \{\xi_1, \dots, \xi_{n+p+1}\}$ and $\Gamma = \{\eta_1, \dots, \eta_{m+q+1}\}$. The $n \times m$ control points $\mathbf{P}_{i,j}$

form a control net. The NURBS surface $\mathbf{S}(\xi, \eta)$ is defined on the one-dimensional basis functions $N_i^{(p)}$ and $M_j^{(q)}$ (with $i = 1, \dots, n$ and $j = 1, \dots, m$) of order p and q , respectively, as

$$\mathbf{S}(\xi, \eta) = \sum_{i=1}^n \sum_{j=1}^m \frac{N_i^{(p)}(\xi) M_j^{(q)}(\eta) w_{i,j} \mathbf{P}_{i,j}}{\sum_{i=1}^n \sum_{j=1}^m N_i^{(p)}(\xi) M_j^{(q)}(\eta) w_{i,j}} \quad (6)$$

An example of a NURBS surface is represented in Figure 2 with the corresponding control net.

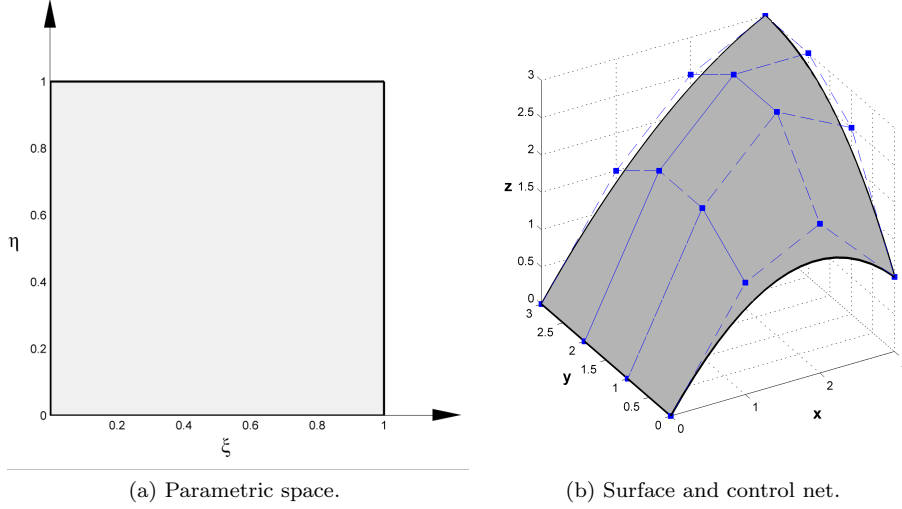


Fig. 2: NURBS surface example.

For a specific design variable a_m the calculation of the velocity field on the parametrized boundary $\mathbf{S}(\xi, \eta, \mathbf{a})$, is simple and would be expressed as:

$$\mathbf{V}_{m,\Gamma}(x(\xi, \eta), y(\xi, \eta), z(\xi, \eta)) = \frac{\partial \mathbf{S}(\xi, \eta, \mathbf{a})}{\partial a_m} = f\left(\frac{\partial \mathbf{P}(\mathbf{a})}{\partial a_m}\right) \quad (7)$$

The boundary velocity field on the discrete model is achieved by the evaluation of (7) using the parametric coordinates (ξ, η) of each surface point. The analytical evaluation of the derivatives of the NURBS and trimmed NURBS (see Subsection 2.1.1) can be cumbersome, because of the lack of explicit expressions to evaluate the control points as a function of \mathbf{a} , and will depend on how each CAD system generates the surfaces as a function of the parameters defined by the user. Therefore, for the sake of generality, we propose to approximate the analytical evaluation of these derivatives by a finite differences approximation as shown in the following equation:

$$\mathbf{V}_{m,\Gamma} \cong \frac{\Delta \mathbf{S}(\xi, \eta, \mathbf{a})}{\Delta a_m} = \frac{\mathbf{S}(\xi, \eta, \mathbf{a} + \Delta a_m) - \mathbf{S}(\xi, \eta, \mathbf{a})}{\Delta a_m} \quad (8)$$

where Δa_m is a perturbation of the design variable a_m .

In practical terms, the evaluation of the shape sensitivities will require the information of the design velocity field at certain points on the boundary. These points are: a) the points used for the numerical integration of boundary integrals, and b) the points of intersection of the NURBS with the edges of the elements, i.e. with the Cartesian axes that define the mesh. Equation (8) will be used to obtain the velocity field at the parametric coordinates (ξ, η) of these points.

2.1.1 Velocity field on trimmed surfaces

NURBS surfaces are inherently four-sided patches that do not allow for the presence of holes nor the direct creation of irregular shapes. Due to the limitation of a strict rectangular topology, trimming is a valuable procedure to devise complex objects. A trimmed NURBS surface consists of: a) a tensor product NURBS surface and, b) a set of properly arranged trimming curves lying within the parametric rectangle of the surface. The trimming curves can be of any form but, when dealing with NURBS entities, it is useful to represent them in NURBS form.

Assume that n_c NURBS curves are given defined as:

$$\mathbf{C}_k(\lambda) = (\xi_k(\lambda), \eta_k(\lambda)) \quad k = 1, 2, \dots, n_c \quad (9)$$

The curves $\mathbf{C}_k(\lambda)$ are all properly oriented forming loops. A loop establishes the boundary of the trimmed region such that, when advancing along the piecewise curve as indicated by its numbering, the real surface material is always on the same side, see Figure 3a. The trimmed surface boundaries are then retrieved by mapping the 2D trimming curves onto the surface. That is,

$$\mathbf{S}(\xi_k(\lambda), \eta_k(\lambda)) \quad k = 1, 2, \dots, n_c \quad (10)$$

are surface curves bounding the trimmed surface. Figure 3b shows the 3D mapping of the trimming loop.

Regarding the evaluation of velocity fields in geometries including trimmed NURBS, the trimming procedure in general does not allow application of the previously explained procedure to evaluate the boundary velocity field. The reason for this is that, when dealing with trimmed entities, in some cases the generation of new geometries is obtained by modifying the trimming curve into the parametric space but not the parametric space itself, i.e. without modifying the control points of the surfaces, leading to $V_{m,\Gamma} = 0$ in (7).

Let us consider a trimmed NURBS surface defined as:

$$\mathbf{S}(\mathbf{C}_k(\lambda, \mathbf{a}), \mathbf{a}) = \mathbf{S}(\xi_k(\lambda, \mathbf{a}), \eta_k(\lambda, \mathbf{a}), \mathbf{a}) \quad k = 1, \dots, n_c \quad (11)$$

where \mathbf{C} are NURBS curves and $(\xi_k(\lambda, \mathbf{a}), \eta_k(\lambda, \mathbf{a}))$ are the surface parametric coordinates of the k trimming curve function of the parameter λ and the design variable vector, \mathbf{a} . In this definition, we assume that \mathbf{a} can influence both the surface \mathbf{S} and the trimming curves \mathbf{C} .

Figure 4a contains the representation of a surface as defined in (11) with the trimmed curves and the parametric subspace Γ_T bounded by them. In Figure 4b, the mapping of the parametric subspace to the physical space is shown along with the diamond-shaped control point polygon.

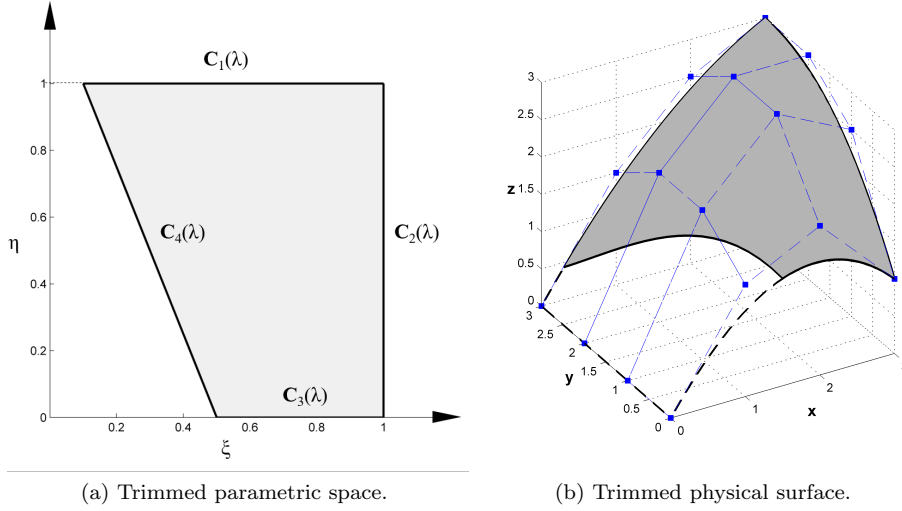


Fig. 3: NURBS surface example.

Now let us assume a change in the design variables such that $\tilde{\mathbf{a}} = \mathbf{a} + \Delta a_m$, where Δa_m is a small increment in a single design variable. Figure 4c shows an illustration of this change in the parametric space Γ_T , that leads to the new domain $\Gamma_{\tilde{T}}$. Figure 4d confirms how a change in the trimming loop yields a different mapping of the subspace, while keeping the control polygon in place. However, it is still necessary to evaluate the value of the velocity field for the points in the domain represented in Figure 4a.

Let A in Figures 4a and 4b be a point of interest of coordinates $(\xi, \eta)_A$ in the parametric space and $(x, y)_A$ in the physical space. The perturbation of the design variable a_m will modify the trimmed surface in the parametric space and hence in the physical space. This will perturb the position of A to \tilde{A} (see Figures 4c and 4d) of coordinates $(\xi, \eta)_{\tilde{A}}$ in the parametric space and $(x, y)_{\tilde{A}}$ in the physical space. The evaluation of the design velocity field at A will require the evaluation of this perturbation in the physical space. We therefore need to find how $(\xi, \eta)_A$ is mapped to $(\xi, \eta)_{\tilde{A}}$. This can be evaluated on the trimming curves $\mathbf{C}_k(\lambda, \mathbf{a})$ but we also need this information in the interior of Γ_T .

The velocity field on the trimming curve \mathbf{C}_k boundary for this particular problem can be written, using approximation (8), as:

$$\mathbf{V}_{m, \mathbf{C}_k} = \frac{\partial \mathbf{S}(\mathbf{C}_k(\lambda, \mathbf{a}))}{\partial a_m} \cong \frac{\mathbf{S}(\mathbf{C}_k(\lambda, \mathbf{a} + \Delta a_m)) - \mathbf{S}(\mathbf{C}_k(\lambda, \mathbf{a}))}{\Delta a_m} \quad (12)$$

Equation (12) will evaluate the velocity field only on the trimming curves. This means that the parametric coordinates of the points (ξ, η) on Γ_T will have to be updated to consider the change of the parametric subspace that leads to $\Gamma_{\tilde{T}}$. Figure 5 shows a general case in which a number of points of interest on the surface, defined by their original parametric coordinates $\{\xi_a, \eta_a\}$, should be

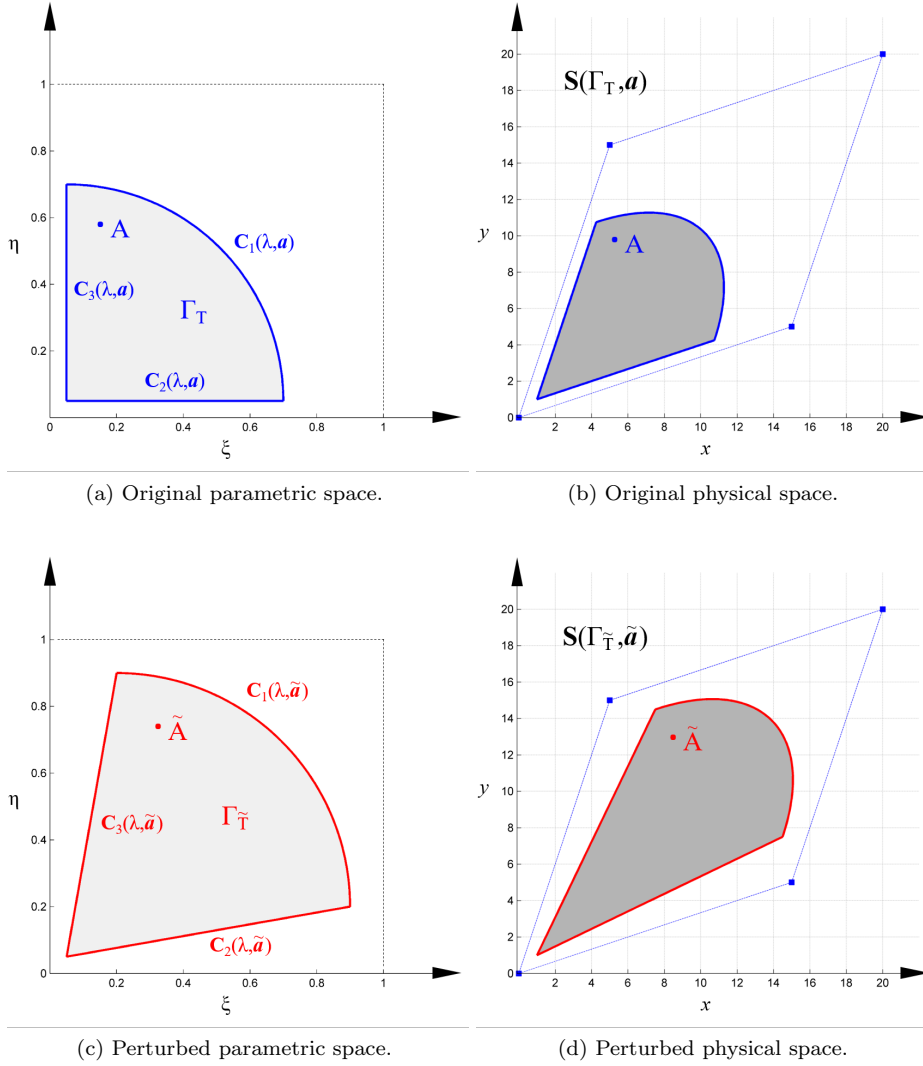


Fig. 4: Trimmed NURBS surface example. Modifying trimming curves.

updated to new coordinates $\{\xi_{\tilde{a}}, \eta_{\tilde{a}}\}$ to obtain a transformation consistent with the trimming curves.

We adapted the idea of the physical approach [4] to obtain this update from $\{\xi_a, \eta_a\}$ to $\{\xi_{\tilde{a}}, \eta_{\tilde{a}}\}$. Hence, we propose solving an auxiliary elasticity problem with imposed displacements on the boundary. It is possible to create a 2D finite element system in the original parametric space from information that is already at our disposal. In this case the intersections between the surface with the Cartesian axes would be the nodes and the elements would be defined by the faces of the

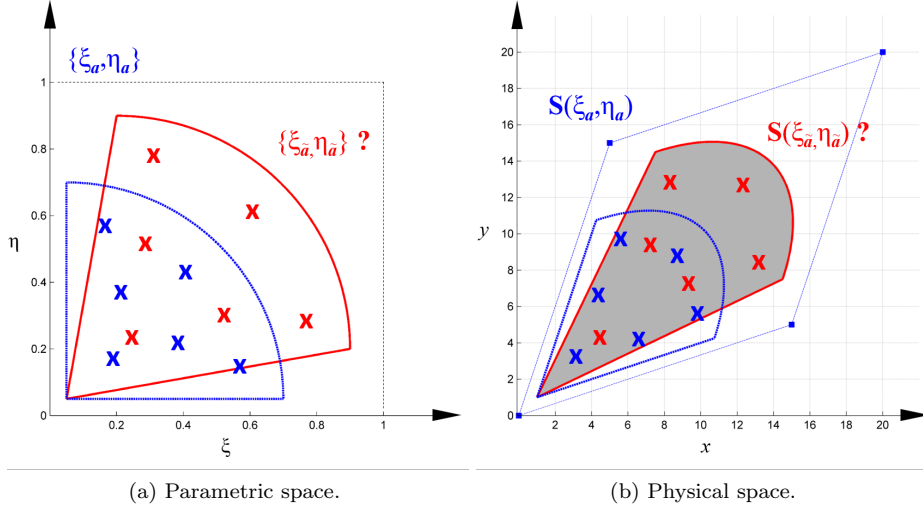


Fig. 5: Problem transforming points within a trimmed NURBS surface.

integration subdomains on the surface (see [45]). Figure 6a shows this proposal. The discretized system of equations of the auxiliary problem can be written as:

$$\mathbf{K}\mathbf{P} = \mathbf{F} \quad (13)$$

where \mathbf{K} is the stiffness matrix, \mathbf{F} is the vector of equivalent nodal forces. In this case $\mathbf{F} = 0$ as Neumann boundary conditions are not applied, and \mathbf{P} contains the prescribed displacements on the boundary and the unknown values of the field in the interior of the domain. These 'prescribed displacements' will correspond to the values of the change on the trimming curves coordinates such that:

$$\mathbf{P}_{m,k} = (\xi_k(\lambda, \mathbf{a} + \Delta a_m), \eta_k(\lambda, \mathbf{a} + \Delta a_m)) - (\xi_k(\lambda, \mathbf{a}), \eta_k(\lambda, \mathbf{a})) \quad k = 1, \dots, n_c \quad (14)$$

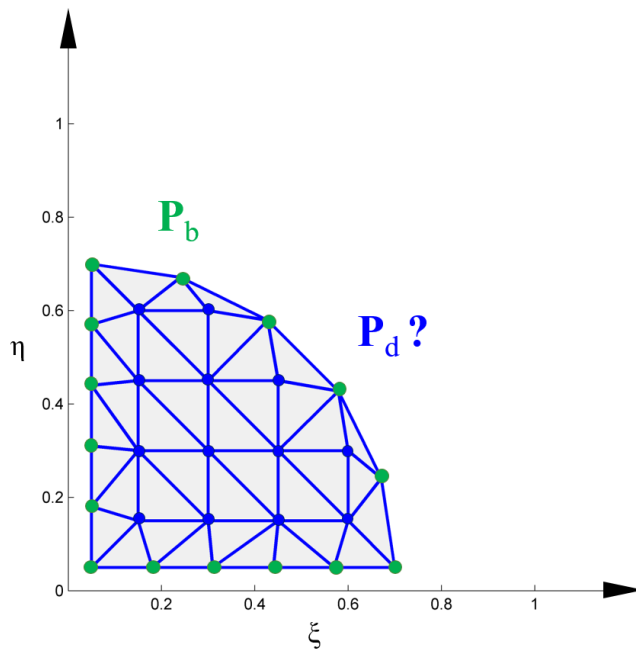
In this way the displacements imposed are those that change the position of the trimming curves in the parametric space associated with the design variable under study.

Solving (13) after applying the Dirichlet boundary conditions provides the perturbation of the position of all the nodes of the mesh shown in Figure 6b, which can be interpolated into the elements. The result will be the position of the original points mapped into the new subspace defined by the perturbed boundary in the parametric space:

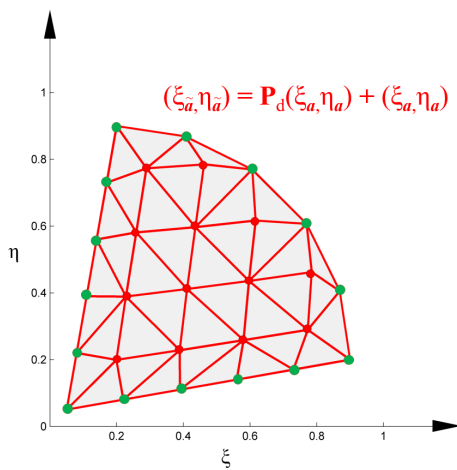
$$\{\tilde{\xi}, \tilde{\eta}\} = \{\xi_{\mathbf{a} + \Delta a_m}, \eta_{\mathbf{a} + \Delta a_m}\} = \{\xi_{\mathbf{a}}, \eta_{\mathbf{a}}\} + \mathbf{P}_m(\xi_{\mathbf{a}}, \eta_{\mathbf{a}}) \quad (15)$$

The velocity field on these surfaces will be calculated as:

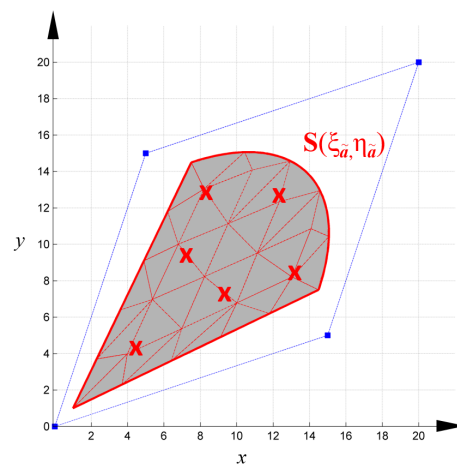
$$\mathbf{V}_{m,\Gamma} \cong \frac{\Delta \mathbf{S}(\xi, \eta, \mathbf{a})}{\Delta a_m} = \frac{\mathbf{S}(\tilde{\xi}, \tilde{\eta}, \mathbf{a} + \Delta a_m) - \mathbf{S}(\xi, \eta, \mathbf{a})}{\Delta a_m} \quad (16)$$



(a) FEM system using the parametric space of a trimmed NURBS surface.



(b) Solution in the parametric space.



(c) Solution in the physical space.

Fig. 6: NURBS surface example solved using the proposed strategy.

To our knowledge, none of the previous works in the bibliography are related to the evaluation of the design velocity field for trimmed surfaces. The procedure

proposed to evaluate the design velocity field for these surfaces involves solving a 2D FE problem. However, the associated computational cost is low, as: a) the FE mesh used for the analysis is that of a previously evaluated triangulation of the trimmed surface in the parametric space required for intersecting the surface with the Cartesian mesh, b) the mesh is a relatively coarse 2D mesh, thus involving a low computational cost and c) the factorization of \mathbf{K} obtained during the process is common to all the design variables.

2.2 Generation of domain velocity fields

After describing the method of evaluating the design velocity field at any point on the boundary of the domain, this section deals with the methods used to obtain the velocity fields in the domain of the models from the boundary values. As explained above, due to the fixed Cartesian configuration of the meshes used in FEAVox, standard velocity field generation techniques cannot be used directly. Figure 7a uses a 2D case to show that, using fixed Cartesian grids, it is possible to find nodes external to the domain (green dots) that will be involved in the evaluation of the design velocity field. Strategies are needed to assign the velocity field both to internal and external nodes so that we can interpolate the velocity field at any point on the elements.

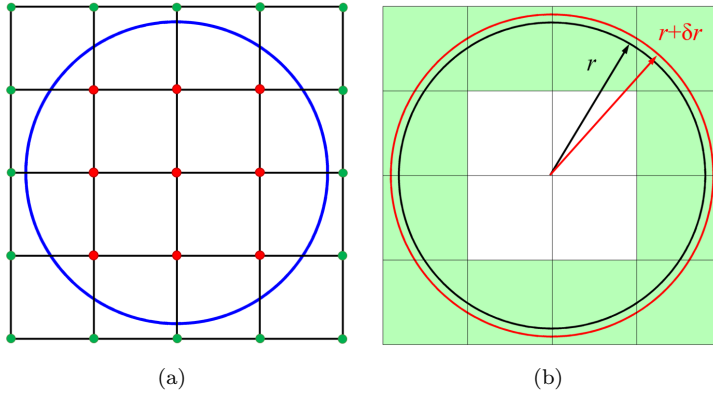


Fig. 7: Embedded methods and velocity fields. (a) Existence of external nodes and (b) perturbation of geometry only in boundary elements.

It should be noted that a perturbation of the boundary will not induce a perturbation of the Cartesian nodes internal to the surface. This is an important feature imposed by the use of Cartesian grids, since the velocity field will be zero in the internal elements, thus reducing the computational cost associated with the evaluation of their volume integrals for shape sensitivity analysis, e.g. the non-shaded elements in Figure 7b.

We propose two different velocity field generators that will represent the geometry changes only in these elements. Both methods can be classified as boundary

element methods [5], as the velocity fields will be non-zero only on the band of elements intersected by the parametrized boundary.

Figure 8 shows an example of the velocity field for the example in Figure 7 that would be obtained by the proposed methods using the radius as design variable. Figure 8a shows the interpolation of the velocity field even in the external nodes, while Figure 8b represents the actual velocity field necessary to evaluate the integrals of the sensitivity analysis only of the internal elements and the integration subdomains of the boundary elements.

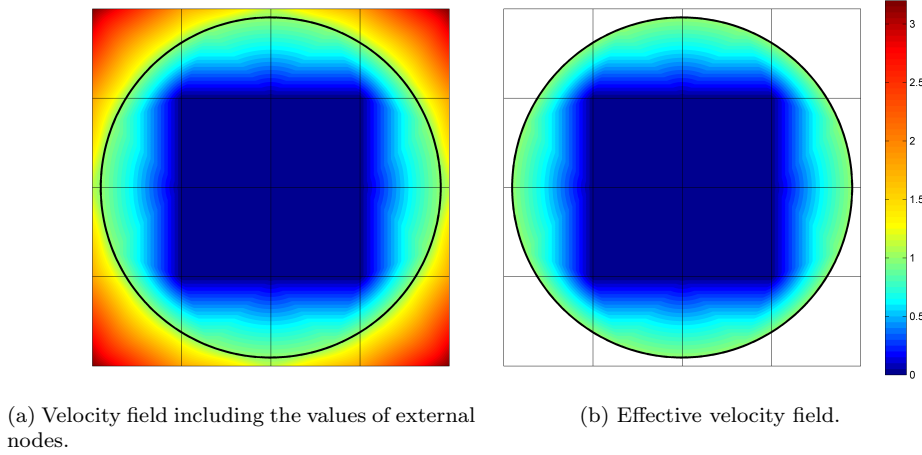


Fig. 8: Representation of a velocity field with the proposed strategies.

2.2.1 Least squares approach

In this first method we use a least squares procedure to extrapolate the values to the external nodes imposing the velocity field on the boundary and the zero velocities on the internal nodes of the elements of the boundary of the layer.

An FE nodal interpolation for each component of the design velocity field is fitted into each element with the velocity field values at the surface integration points of the elements and to $\mathbf{V} = 0$ at the internal nodes of the boundary elements. By using a least square approach we obtain the linear system of equations:

$$\mathbf{M}\mathbf{V}_{m,q} = \mathbf{G}_{m,q} \quad q = x, y \text{ and } z \quad (17)$$

The system matrix \mathbf{M} is obtained by the assembly of the mass matrix-type array of each element along the boundary. The global mass matrix is given by:

$$\mathbf{M} = \sum_{e=1}^{n_e} \int_{\Gamma_D^e} \mathbf{N}^T \mathbf{N} |\mathbf{J}| d\Gamma \quad (18)$$

where

Γ_D^e is the portion of the boundary within the element,
 \mathbf{N} corresponds to the matrix of finite element interpolation functions.

On the other side of the equation, the vector $\mathbf{G}_{m,q}$ is evaluated by adding the contribution of elements:

$$\mathbf{G}_{m,q} = \sum_{\Gamma_D^e} \int_{\Gamma_D^e} \mathbf{N}^T \mathbf{V}_{m,q}^e |\mathbf{J}| d\Gamma \quad q = x, y, z \quad (19)$$

with $\mathbf{V}_{m,q}^e$ as the q^{th} component of the velocity field on the boundary related to the design variable m within each element. Note that this is a low-cost procedure as it only involves the elements along the boundary, which is of interest for 3D domains.

2.2.2 Physical approach

This method consists of solving a linear elasticity problem in which the velocity field on the boundary is considered as the displacements applied on the boundary. This auxiliary problem will have, for example, the following characteristics:

- The body Ω_{phys} is characterized with a linear elastic material with Young modulus equal to one and zero Poisson ratio;
- The discretization used to evaluate the design velocity field is the discretization used to evaluate the displacements;
- Every single shape design variable gives a non-zero velocity field on the elements cut by the boundary and zero velocity on the rest of the domain, which ensures the equilibrium of each auxiliary problem. The unknowns are the velocities for all external nodes of the actual FE mesh Ω_{approx} .

Even though this method needs the resolution of a system of equations as large as the original problem, the associated computational cost is reduced, as: a) the FE mesh used for the analysis is the same Cartesian mesh used for the sensitivity analysis, b) we can remove the internal nodes from the system since the velocity field is set to 0 on these nodes, leading to a problem only associated with the domain's boundary, which can be seen as a 2D problem, and c) the stiffness matrix \mathbf{K} can be factorized during the process and used for all the design variables.

3 Calculating shape sensitivities with FEAVox

This section describes the adaptation of the discrete analytical method to evaluate shape sensitivities when using the cgFEM methodology. In order to do this we have to take into consideration that imposing Dirichlet boundary conditions in an immersed boundary environment requires different strategies from those used in standard FEM. This also has to be considered for the calculation of the shape sensitivities dealt with in this section.

The case of the Neumann boundary conditions can be easily undertaken by simply considering that the integration surface can cut the element and does not necessarily has to correspond with the element faces. In cgFEM we use a stabilized

method [72] similar to Niche's method that modifies the classical structure of the FE linear elastic stiffness matrix and force vector.

The global stiffness matrix is obtained by the contribution of the classical stiffness matrix of each element \mathbf{k}^e and a stabilization term \mathbf{k}_D^e for all the boundary elements containing the Dirichlet boundary.

The stiffness matrix of each element is computed by

$$\mathbf{k}^e = \int_{\Omega^e} \mathbf{B}^T \mathbf{D} \mathbf{B} |\mathbf{J}| d\Omega \quad (20)$$

where

Ω^e is the domain in local element coordinates,

\mathbf{B} is the nodal strains-displacements matrix,

\mathbf{D} is the stiffness matrix that relates stresses with strains. In this work we consider linear elasticity where, under isotropic behavior, this matrix depends only on E , the Young modulus, and ν , the Poisson ratio of the material,

$|\mathbf{J}|$ is the determinant of the matrix \mathbf{J} , representing \mathbf{J} the Jacobian matrix of transformation of the global coordinates (x, y, z) to the local element coordinates (ξ, η, τ) .

and the stabilization term:

$$\mathbf{k}_D^e = \int_{\Gamma_D^e} \frac{\kappa^*}{h} \mathbf{C}^T \mathbf{C} |\mathbf{J}| d\Gamma \quad (21)$$

where

Γ_D^e is the portion of the Dirichlet boundary within the element,

κ^* is the penalty constant, being $\kappa^* = \kappa \cdot E$ and $\kappa > 0$,

h is the element size,

\mathbf{C} is the matrix of finite element interpolation if Dirichlet conditions are applied on the three displacement components x , y and z .

$$\mathbf{C} = \mathbf{N} = \begin{bmatrix} N_1 & 0 & 0 & N_2 & 0 & 0 & N_3 & 0 & 0 & \dots & N_{nnod} & 0 & 0 \\ 0 & N_1 & 0 & 0 & N_2 & 0 & 0 & N_3 & 0 & \dots & 0 & N_{nnod} & 0 \\ 0 & 0 & N_1 & 0 & 0 & N_2 & 0 & 0 & N_3 & \dots & 0 & 0 & N_{nnod} \end{bmatrix}$$

with $nnod$ as the number of nodes per element. Otherwise $\mathbf{C} = \mathbf{S}\mathbf{N}$, where $\mathbf{S}_{ii} = \sum_d \delta_{id}$ would be a diagonal matrix, d is the direction where Dirichlet boundary conditions are applied and δ is the Dirac delta function.

On the other side of the equation, the equivalent force vector \mathbf{f} is evaluated by adding the contribution of the standard FE vector of equivalent forces on nodes \mathbf{f}_q , the stabilization term of the Dirichlet boundary \mathbf{f}_g and the stabilizing stress component \mathbf{f}_s .

The vector \mathbf{f}_q is the standard FE vector due to point forces, volumetric forces, forces distributed over the Neumann surface of the element, evaluated assembling the contribution \mathbf{f}_q^e of every element e on the domain:

$$\mathbf{f}_q^e = \int_{\Gamma_N^e} \mathbf{N}^T \mathbf{t} |\mathbf{J}| d\Gamma + \int_{\Omega^e} \mathbf{N}^t \mathbf{b} |\mathbf{J}| d\Omega + \mathbf{p} \quad (22)$$

where vectors \mathbf{t} , \mathbf{b} and \mathbf{p} correspond to the surface, body and point loads, respectively.

The vector \mathbf{f}_g is due to the non-homogeneous Dirichlet condition $\mathbf{u}^h = \mathbf{g}$ on Γ_D and it is evaluated assembling the contribution of every element on the Dirichlet boundary:

$$\mathbf{f}_g^e = \int_{\Gamma_D^e} \frac{\kappa^*}{h} \mathbf{C}^T g |\mathbf{J}| d\Gamma \quad (23)$$

Finally, \mathbf{f}_s is the stabilizing term which depends on the stress field. In our formulation we use the recovered tractions on Γ_D evaluated from the recovered stress field $\boldsymbol{\sigma}^*$ [62] to stabilize, solving the problem iteratively updating the stress field value [71, 72], $\boldsymbol{\sigma}^*(\hat{\mathbf{u}}^h)$ being the FE recovered stress field calculated for an FE solution from a previous iteration (or mesh) $\hat{\mathbf{u}}^h$. The traction on the boundary is defined as $\mathbf{T}(\hat{\mathbf{u}}^h) = \boldsymbol{\sigma}^*(\hat{\mathbf{u}}^h) \cdot \mathbf{n}$ where \mathbf{n} is the unit vector normal to the boundary, then

$$\mathbf{f}_s^e = \int_{\Gamma_D^e} \mathbf{C}^T \mathbf{T}(\hat{\mathbf{u}}^h) |\mathbf{J}| d\Gamma \quad (24)$$

which modifies the global system such that:

$$(\mathbf{K} + \mathbf{K}_D) \mathbf{u} = \mathbf{f}_q + \mathbf{f}_g + \mathbf{f}_s \quad (25)$$

The derivative of (25) with respect to any design variable a_m provides the sensitivity of the calculation

$$\left(\frac{\partial \mathbf{K}}{\partial a_m} + \frac{\partial \mathbf{K}_D}{\partial a_m} \right) \mathbf{u} + (\mathbf{K} + \mathbf{K}_D) \frac{\partial \mathbf{u}}{\partial a_m} = \frac{\partial \mathbf{f}_q}{\partial a_m} + \frac{\partial \mathbf{f}_g}{\partial a_m} + \frac{\partial \mathbf{f}_s}{\partial a_m} \quad (26)$$

then, rearranging, yields

$$(\mathbf{K} + \mathbf{K}_D) \frac{\partial \mathbf{u}}{\partial a_m} = \left(\frac{\partial \mathbf{f}_q}{\partial a_m} + \frac{\partial \mathbf{f}_g}{\partial a_m} + \frac{\partial \mathbf{f}_s}{\partial a_m} \right) - \frac{\partial \mathbf{K}}{\partial a_m} \mathbf{u} = \mathbf{f}_{ps_m} \quad (27)$$

The discrete analytical method consists of obtaining analytical expressions of the sensitivities of the external forces and stiffness matrix. In our case we used the finite differences approximation of Eqs. (8) and (16) in the evaluation of the velocity field that will be used to obtain the derivatives of the previous equation. Therefore the method used to evaluate the shape sensitivities can be classified as a discrete semi-analytical method. Then using (27) the sensitivities of the displacements are obtained. From these sensitivities other response magnitudes are calculated.

3.1 Evaluation of derivatives

In this section we derive the components of (27) to be able to evaluate the shape sensitivities in the Cartesian grid framework. First, starting with \mathbf{k}^e and considering that the derivative of \mathbf{D} with respect to design variables is zero

$$\frac{\partial \mathbf{k}^e}{\partial a_m} = \int_{\Omega^e} \left[\frac{\partial \mathbf{B}^T}{\partial a_m} \mathbf{D} \mathbf{B} + \mathbf{B}^T \mathbf{D} \frac{\partial \mathbf{B}}{\partial a_m} \right] |\mathbf{J}| d\Omega + \int_{\Omega^e} \left[\mathbf{B}^T \mathbf{D} \mathbf{B} \frac{\partial |\mathbf{J}|}{\partial a_m} \right] d\Omega \quad (28)$$

As can be found in [16], this expression depends on known magnitudes and the factors $\frac{\partial \mathbf{B}}{\partial a_m}$ and $\frac{\partial |\mathbf{J}|}{\partial a_m}$, which are a function of the velocity field evaluated above.

To evaluate $\frac{\partial \mathbf{k}_D^e}{\partial a_m}$ it is necessary to take into account that, as the Cartesian grid will not be modified by the design variables, i.e. h and \mathbf{C} do not depend on a_m . Therefore, the only non-zero partial derivative with respect to a_m is $\frac{\partial |\mathbf{J}|}{\partial a_m}$, which leads to:

$$\frac{\partial \mathbf{k}_D^e}{\partial a_m} = \int_{\Gamma_D^e} \frac{\kappa^*}{h} \mathbf{C}^T \mathbf{C} \frac{\partial |\mathbf{J}|}{\partial a_m} d\Gamma \quad (29)$$

In general, the sensitivity with respect to design variables of the nodal equivalent forces \mathbf{f}_q will have two terms, one dependent on the variation of the forces (punctual, volumetric, etc.), with respect to design variables, and the second depending on the velocity field. In the first case the expression defining the dependence of the acting forces with respect to the design variables must be available but in this work we considered constant acting forces. In the second case the derivatives had to be calculated by the same procedure as for the stiffness matrix components.

The remaining components of \mathbf{f} have to be derived as

$$\frac{\partial \mathbf{f}_g^e}{\partial a_m} = \int_{\Gamma_D^e} \frac{\kappa^*}{h} \mathbf{C}^T g \frac{\partial |\mathbf{J}|}{\partial a_m} d\Gamma \quad (30)$$

where we have assumed that the Dirichlet boundary conditions are not a function of the design variables,

$$\frac{\partial \mathbf{f}_s^e}{\partial a_m} = \int_{\Gamma_D^e} \left[\mathbf{C}^T \frac{\partial \mathbf{T}(\hat{\mathbf{u}}^h)}{\partial a_m} |\mathbf{J}| + \mathbf{C}^T \mathbf{T}(\hat{\mathbf{u}}^h) \frac{\partial |\mathbf{J}|}{\partial a_m} \right] d\Gamma \quad (31)$$

where

$$\frac{\partial \mathbf{T}(\hat{\mathbf{u}}^h)}{\partial a_m} = \frac{\partial \boldsymbol{\sigma}^*}{\partial a_m} \mathbf{n} + \boldsymbol{\sigma}^* \frac{\partial \mathbf{n}}{\partial a_m} \quad (32)$$

As mentioned above, the term \mathbf{T} used to stabilize the Lagrange multipliers is a recovered stress field obtained from the FE solution. To evaluate the stresses in linear elasticity we consider the general expression for the calculation of the FE stresses $\boldsymbol{\sigma}_h$ in continuous isoparametric elements

$$\boldsymbol{\sigma}_h = \mathbf{D} \mathbf{B} \mathbf{u}_h^e \quad (33)$$

\mathbf{u}_h^e being the vector of nodal displacements of element e . Taking the derivative with respect to the design variable a_m yields

$$\frac{\partial \boldsymbol{\sigma}}{\partial a_m} = \mathbf{D} \mathbf{B} \frac{\partial \mathbf{u}_h^e}{\partial a_m} + \mathbf{D} \frac{\partial \mathbf{B}}{\partial a_m} \mathbf{u}_h^e \quad (34)$$

where all terms on the right can be evaluated using the development of the preceding sections. Once we have evaluated both $\boldsymbol{\sigma}$ and $\frac{\partial \boldsymbol{\sigma}}{\partial a_m}$ we can apply the construction of the smoothing field based on a recovery technique shown in [62].

Remark 1 To simplify the evaluation of $\frac{\partial \boldsymbol{\sigma}^*}{\partial a_m}$ we considered $\frac{\partial \boldsymbol{\sigma}^*}{\partial a_m} = \left(\frac{\partial \boldsymbol{\sigma}}{\partial a_m} \right)^*$. The numerical results will show that this approximation, previously used in [19], does not influence the results.

4 Numerical examples

From the premise of the approximate nature of the FEM, the error of the solution associated with the size of the elements of the FE mesh can be termed FE discretization error. Usually this error is quantified in terms of the energy norm $\|\cdot\|$ as:

$$\|e(\mathbf{u})_{ex}\|^2 = \int_{\Omega} (\boldsymbol{\sigma}_h - \boldsymbol{\sigma})^T \mathbf{D}^{-1} (\boldsymbol{\sigma}_h - \boldsymbol{\sigma}) d\Omega \quad (35)$$

where $\boldsymbol{\sigma}_h$ and $\boldsymbol{\sigma}$ are the FE (approximate) and the exact stresses respectively.

The sensitivity analysis results evaluated by the FEM are also influenced by the discretization error associated with the FE model. Therefore, a way must be defined to evaluate the discretization error in the evaluation of the sensitivities. Following [19] we use the sensitivity of the squared energy norm with respect to each design variable, i.e.

$$\chi_m = \frac{\partial \|e(\mathbf{u})_{ex}\|^2}{\partial a_m} = \frac{\partial}{\partial a_m} \int_{\Omega} \boldsymbol{\sigma}^T \mathbf{D}^{-1} \boldsymbol{\sigma} d\Omega \quad (36)$$

and following a similar procedure to that used to derivate the expression (28) we obtain:

$$e(\chi_m)_{ex} = \sum_{\Omega_e}^{n_e} \int_{\Omega_e} (\boldsymbol{\sigma}_h - \boldsymbol{\sigma})^T \mathbf{D}^{-1} \left(2 \left(\frac{\partial (\boldsymbol{\sigma}_h - \boldsymbol{\sigma})}{\partial a_m} \right) + \frac{(\boldsymbol{\sigma}_h - \boldsymbol{\sigma})}{|\mathbf{J}|} \frac{\partial |\mathbf{J}|}{\partial a_m} \right) |\mathbf{J}| d\Omega_e \quad (37)$$

The following definition of relative error in sensitivities can be used to make the error in sensitivities comparable with the error in energy norm in relative terms:

$$\eta(\chi_m)_{ex} = \sqrt{\left| \frac{e(\chi_m)_{ex}}{\chi_m} \right|} \quad (38)$$

In the absence of singularities, with this definition, the optimal convergence rate of the relative error in sensitivities with respect to the number of degrees of freedom will have the convergence rate of the relative error in energy norm. The theoretical predicted convergence rate in energy norm is $O(h^p)$. Therefore, following the rationale of [80], taking into account that the number of degrees of freedom N in 3D is approximately inversely proportional to h^3 the convergence rate can be written as $O(N^{-p/3})$. Therefore, the convergence rate of $\eta(\chi)$ as defined in (38) will be $-1/3$ for tri-linear elements and $-2/3$ for tri-quadratic elements in 3D problems.

In addition, we know from [61] that there is a relationship between the discretization error in energy norm and the so-called sensitivity discretization error, such that:

$$\frac{e(\chi_m)_{ex}}{\|e(\mathbf{u})_{ex}\|^2} \approx R_m \quad (39)$$

This expression shows that, only with the presence of the discretization error, the discretization error in the sensitivity of the squared energy norm and the squared discretization error in energy norm will both be related by a constant R_m . This relationship between the two types of errors can be adopted as an indicator to measure the quality of the procedure to generate the velocity field.

4.1 Thick wall infinite cylinder under internal pressure

The geometrical model for this test is represented in Figure 9. A linear-elastic analysis is performed on a domain whose boundary representation is defined by NURBS. Appropriate symmetry boundary conditions allow only 1/4 of the section to be modeled. The internal and external surfaces are of radius a and b , with $a = 5$ and $b = 20$. Young's modulus is $E = 1000$, Poisson's ratio is $\nu = 0.3$ and the applied load is $P = 1$. The shape sensitivity analysis of this example considers only one design variable corresponding to the outer radius of the cylinder, thus taking $a_m = b$.

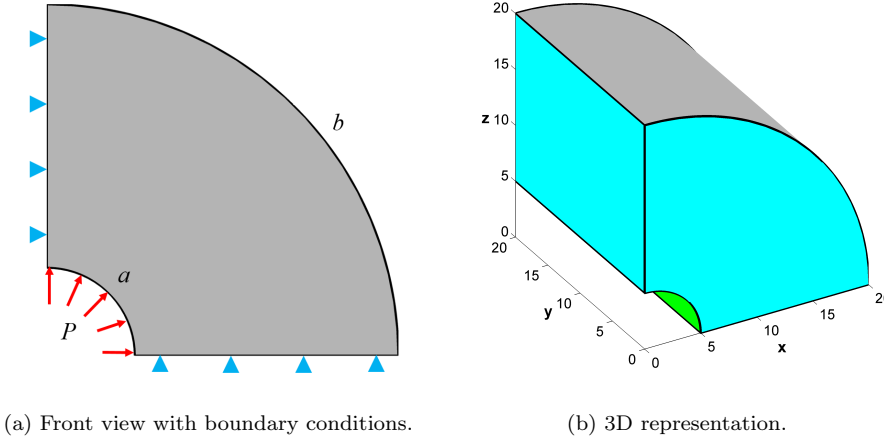


Fig. 9: Model of a cylinder under internal pressure.

The exact solution for displacements and stresses is given by:

$$u_r = \frac{P(1+\nu)}{E(k^2-1)} \left(r(1-2\nu) + \frac{b^2}{r} \right), \quad u_y = 0 \quad (40)$$

$$\sigma_r = \frac{P}{k^2-1} \left(1 - \frac{b^2}{r^2} \right), \quad \sigma_\phi = \frac{P}{k^2-1} \left(1 + \frac{b^2}{r^2} \right), \quad \sigma_y = \nu(\sigma_x + \sigma_z) \quad (41)$$

where $k = b/a$, $r = \sqrt{x^2 + z^2}$.

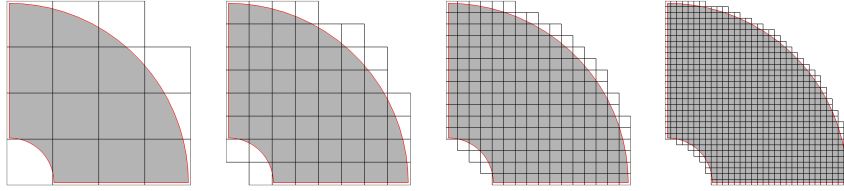
The analytical sensitivity of the squared energy norm (1/4 of cylinder) is:

$$\chi = \frac{\partial \|\mathbf{u}_{ex}\|^2}{\partial a_m} = 2 \frac{\partial \Pi}{\partial a_m} = 2\pi \frac{p^2(1+\nu)}{E} + \frac{a^4 b(\nu-1)}{(a^2-b^2)^2} \quad (42)$$

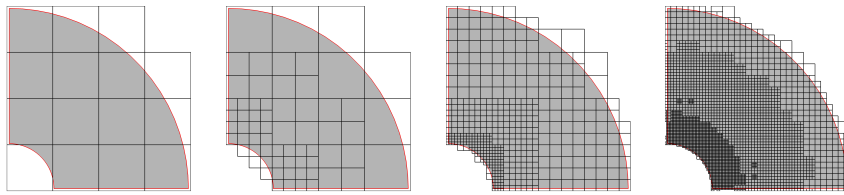
For the data used in the model we will have:

$$\begin{aligned} \|\mathbf{u}_{ex}\|^2 &= 0.055815629478779 \\ \chi &= -5.082398781807488 \cdot 10^{-4} \end{aligned} \quad (43)$$

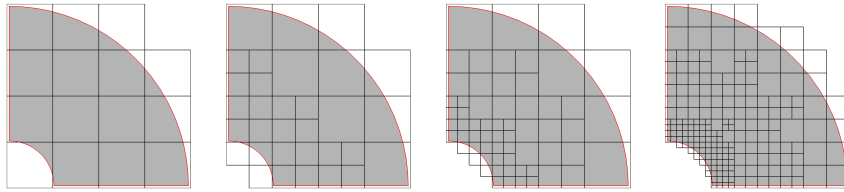
For this problem we will analyze the behavior of the methods used to generate the velocity field, i.e. the proposed least squares approach (LS) and the physical approach (PA), in various FE analyses. In the first analysis we will study the convergence for tri-linear elements (L8) with meshes uniformly refined and h -adapted meshes. The meshes used in this simulation can be seen in Figures 10a and 10b.



(a) First four meshes of the uniform refinement analysis (L8 and Q20).



(b) h -adaptive meshes with tri-linear elements (L8).



(c) h -adapted meshes with tri-quadratic elements (Q20).

Fig. 10: 2D view of the first four meshes of the h -refinement analyses.

Figure 11a shows the relative error of the sensitivity analysis $\eta(\chi)$ and Figure 11b shows its convergence rate as a function of the number of degrees of freedom. The theoretical convergence rate for tri-linear elements ($-1/3$) is indicated in the plot with a black horizontal line.

The convergence plots show almost optimal rates for tri-linear elements using the two velocity fields proposed in this contribution, although the physical approach provides more stable results. Regarding the quality of the velocity fields, estimated using R_m (see Figure 11c) we can conclude that all the analyses show good behavior, but the velocity field calculated using the physical approach shows slightly better stability.

In Figure 12 we compare the results for tri-quadratic elements (Q20). Repeating strategy, we analyze uniform meshes (Fig. 10a) and h -adapted meshes (Fig.

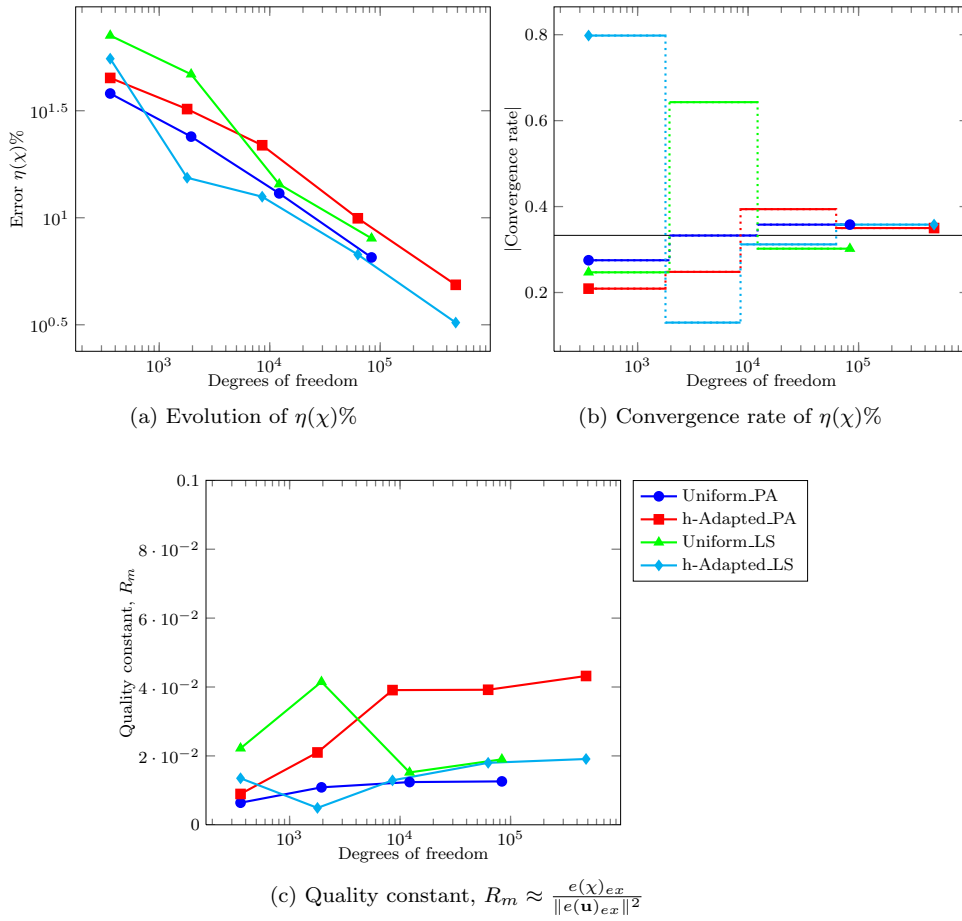


Fig. 11: Thick wall infinite cylinder. Analysis with tri-linear elements (L8)

10c). Figure 12a shows similar behavior in the convergence of the relative error in sensitivities for the two methods proposed to construct the velocity field. For 3D problems with a non-singular solution, the theoretical convergence rate is $-2/3$ if tri-quadratic elements are used. Figure 12b shows that both velocity fields provide optimal convergence rates. In addition, h -adapted meshes present convergence above the optimal rate, which could indicate that the analysis is still in the pre-asymptotic range.

Finally, the quality of the velocity fields quantified by R_m (see Figure 12c) show less stability than that obtained for tri-linear elements, although the interval of oscillation is narrow enough to consider the proposed velocity fields acceptable.

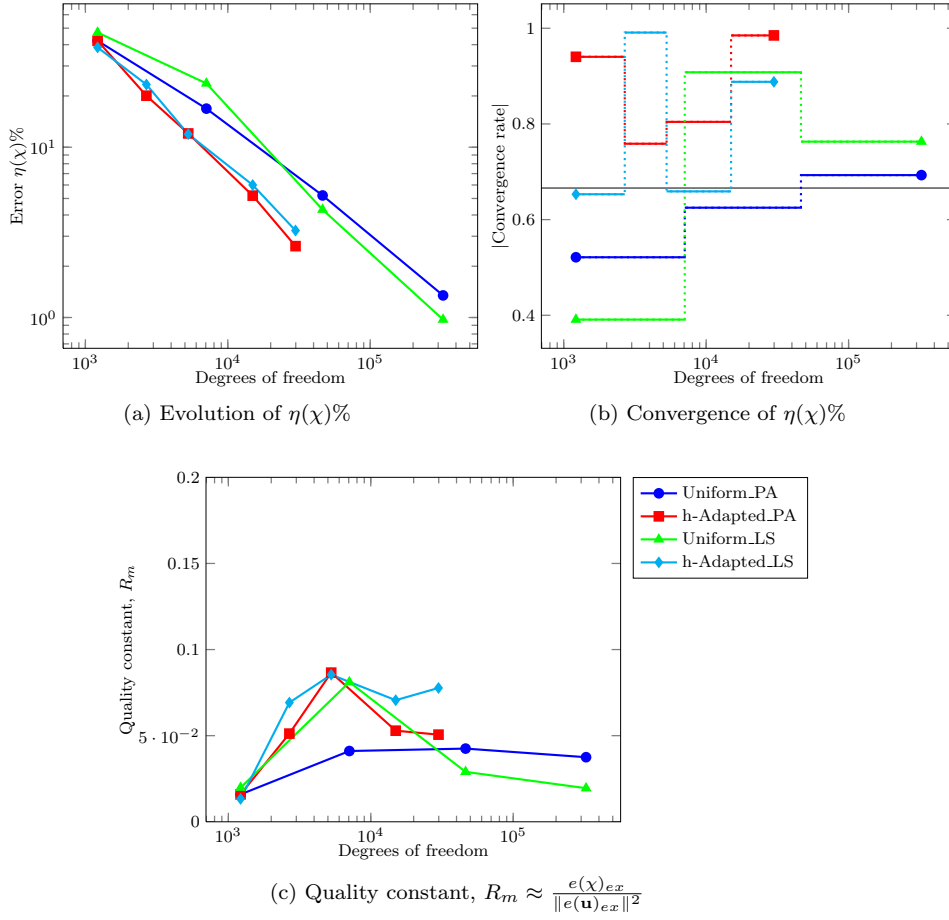


Fig. 12: Thick wall infinite cylinder. Analysis with tri-quadratic elements (Q20)

4.2 Sequence of collinear cracks in an infinite plate

In problems of Linear Elastic Fracture Mechanics (LEFM), the Stress Intensity Factor (SIF) is the parameter that characterizes the stress field near a crack tip. This parameter is vital to assess the maximum allowable stress, critical crack size, fatigue life of a component with cracks, etc.

The evaluation of the SIF represents an interesting challenge for shape sensitivity analysis, given the singular nature of the problem [14, 15, 18, 21]. In fact, the quantity called *energy release rate* \mathcal{G} is the variation of the total potential energy of a component as a function of the crack size growth.

The energy release rate \mathcal{G} for a two-dimensional LEFM problem under mode I loads can be defined as:

$$\mathcal{G} = -\frac{d\Pi_p}{da} = \frac{d\Pi}{da} \quad (44)$$

where

Π_p is the total potential energy. In the case where the load remains constant. $\Pi_p = -\Pi$, where Π is the strain energy and a is the length of the crack.

Since the energy norm squared is equal to twice the strain energy ($\|\mathbf{u}\|^2 = 2\Pi$) it follows:

$$\mathcal{G} = \frac{1}{2} \frac{d \|\mathbf{u}\|^2}{da} \quad (45)$$

We can therefore evaluate \mathcal{G} using shape sensitivity analysis, assuming that a , the crack length, is the design variable. From (45) and from the definition of χ_m in (36) we have:

$$\mathcal{G} = \frac{1}{2} \chi \quad (46)$$

On the other hand, the energy release rate \mathcal{G} and the SIF \mathbf{K}_I are related by the following expression:

$$\mathbf{K}_I = \sqrt{E' \mathcal{G}} \quad (47)$$

where $E' = E$ in plane stress and $E' = E/(1 - \nu^2)$ for plane strain, E is the elasticity modulus and ν is the Poisson's ratio.

To evaluate the effectiveness of the sensitivity calculation module in singular problems we are going to use the problem of a sequence of collinear cracks in an infinite plate (see Figure 13).

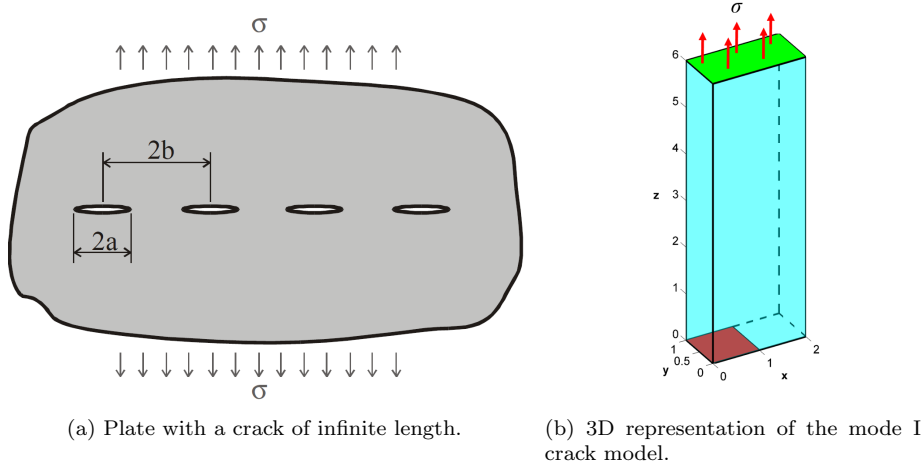


Fig. 13: Infintie plate with a sequence of collinear cracks.

Considering the symmetries of the problem in Figure 13a, the FE model involves taking into account the strip bounded by the center of the crack and the

equidistant point between the ends of two consecutive cracks, together with the appropriate boundary conditions, as shown in Figure 13b. On the blue surfaces the normal displacements have been blocked to simulate the symmetry and plain strain conditions. The red surface is free and the constant stress σ is applied on the top surface. A considerable height was used in the analysis to make certain that the effect of the finite height on the SIF was insignificant.

In an infinite sequence of collinear cracks subjected to constant stress σ , the exact value of Stress Intensity Factor \mathbf{K}_I is given by equation (Kanninen and Popelar [30]):

$$\mathbf{K}_I = \sigma \sqrt{\pi a} \sqrt{\frac{2b}{\pi a} \tan\left(\frac{\pi a}{2b}\right)} \quad (48)$$

For the data used in the model it yields:

$$\begin{aligned} \mathbf{K}_I &= 200 \\ \chi &= 2\mathcal{G} = 0.007112888 \end{aligned} \quad (49)$$

Since the crack has a top and a bottom, the value of χ in the upper part would be obtained when modeling both sides. However, the model used in the numerical analysis uses only the top of the crack, so that the value of χ obtained directly through the shape sensitivity analysis approximates to half the value displayed. Thus the value of χ to be compared with the numerical results is:

$$\chi = 0.003556444 \quad (50)$$

In order to evaluate the behavior of the velocity fields, as in the previous problem, we consider using the ratio R_m defined previously. However there is no expression to evaluate the exact energy norm for this problem. To determine the error in strain energy we have taken, as a reference, the solution obtained from a very refined 2D h -adapted mesh (23811 degrees of freedom, 12059 nodes, 5918 elements) with quadratic triangular elements, with an estimated error of 0.0753%, evaluated using the ZZ error estimator [79] i.e. using (35) but substituting the exact stress field $\boldsymbol{\sigma}$ by a recovered stress field $\boldsymbol{\sigma}^*$ obtained by the recovery technique described in [62]. In this mesh, the value of $\|\mathbf{u}\|^2$, which will be considered as exact in the analyses, is:

$$\|\mathbf{u}_{ex}^*\|^2 = 0.009582263 \quad (51)$$

The J integral [59] is commonly used to characterize the singularity at the crack tip in LEFM problems. The behavior of this contour integral can be considerably improved by means of the Equivalent Domain Integral (EDI) method [38, 69]. The transformation of the J contour integral into a domain integral leads to the appearance of an auxiliary function, usually denoted as q , which must be defined by the analyst. This function used in the EDI method is equivalent to the velocity field used to characterize the singularity through the use of shape sensitivity analysis. Therefore, in this problem, we will compare the behavior of the velocity field based on the physical approach (PA) with the behavior of an auxiliary function q commonly used in the EDI method, a *Plateau* function. Hence, as in the case of the auxiliary function q , the Plateau velocity field will be defined as a vector field

in the direction of the crack propagation, with a maximum constant value within the volume defined by a radius R_{in} around the crack tip and a linear decrease to 0 within R_{in} and an outer radius R_{out} . Figure 14 shows the appearance of these velocity fields.

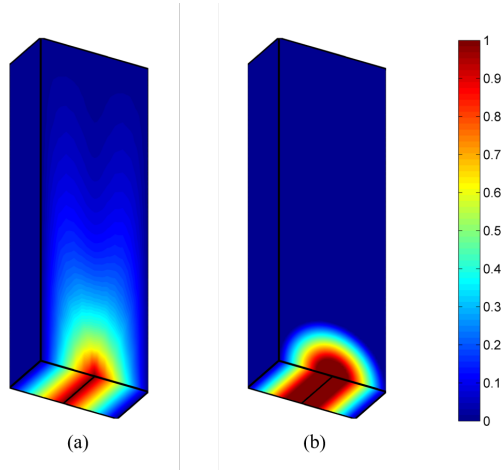
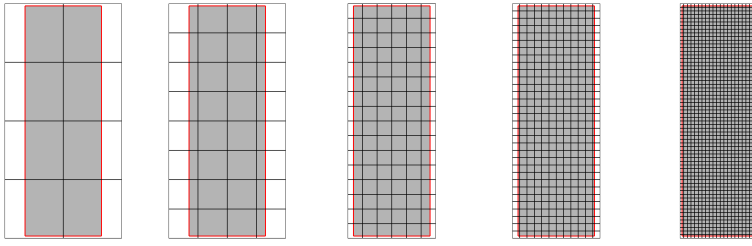


Fig. 14: Velocity fields for the crack model. (a) Physical approach and (b) Plateau function.

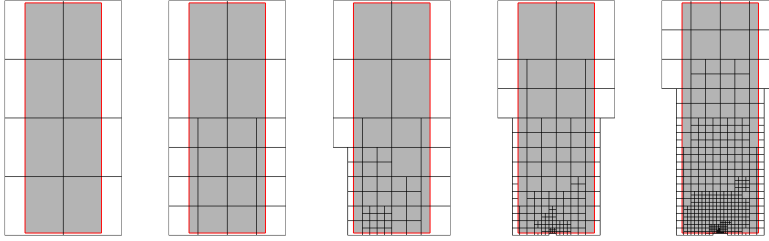
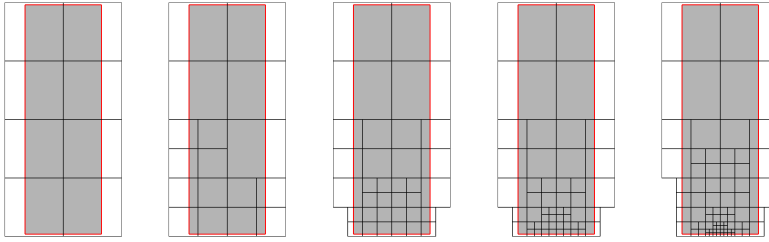
As in the previous test problem, a sequence of uniformly refined meshes has been used with both tri-linear (L8) and tri-quadratic (Q20) elements, Figure 15a, and two sequences of h -adapted meshes, one for linear elements, Figure 15b, and another for quadratic elements, presented in Figure 15c.

Figure 16a shows the variation of the relative error in sensitivities $\eta(\chi)$ obtained for each velocity field, while Figure 16c represents the evolution of \mathbf{K}_I in terms of the number of degrees of freedom. Tri-linear elements and both uniform and h -adaptive refinement have been used in the analysis represented. Figure 16b shows the convergence rate of $\eta(\chi)$ with respect to the number of degrees of freedom for the different meshes. We can see how only the h -adaptive simulations reach the optimal theoretical convergence rate of $-1/3$, for 3D tri-linear elements. The convergence rate for singular problems and uniform refinement is dominated by the intensity of the singularity λ and not by the degree of the interpolation of the solution. In this case $\lambda = 0.5$, thus, the convergence rate in terms of the number of degrees of freedom will be $-\lambda/d$, d being the dimensionality of the problem. Thus, for $\lambda = 0.5$ and $d = 3$, the theoretical convergence rate with respect to the number of degrees of freedom for uniform refinement is $-1/6$. The convergence rate for uniform refinements is close to this value. Regarding the quality of the velocity fields evaluated using R_m , we observe in Figure 16d how the velocity field defined using the physical approach is more stable for both uniform meshes and for h -adapted meshes but especially for uniform refinement processes.

The same analyses were carried out with tri-quadratic elements. Figure 17b shows the convergence rate for the different meshes. We can observe that only the



(a) First five meshes of the uniform refinement analysis (L8 and Q20).

(b) h -adaptive meshes with tri-linear elements (L8).(c) h -adaptive meshes with tri-quadratic elements (Q20).Fig. 15: Infinite sequence of cracks. First five meshes of the h -refinement process.

h -adaptive simulations reach the optimal theoretical convergence rate of $-2/3$, for 3D tri-quadratic elements. The uniformly refined meshes have a convergence rate close to the convergence rate for tri-linear elements with uniform refinement, i.e. close to the theoretical convergence rate for this type of mesh, $-1/6$, as expected. Figure 17c represents the evolution of \mathbf{K}_I with respect to the number of degrees of freedom. The evolution of the quality constant for tri-quadratic elements shown in Figure 17d is similar to that of tri-linear elements. The results obtained by the physical approach are clearly more stable than with the plateau velocity, which produces oscillatory behavior, especially in h -adapted meshes.

5 Conclusions

This paper proposes an extension of a methodology for the calculation of shape sensitivities, together with the definition of two methodologies that define the de-

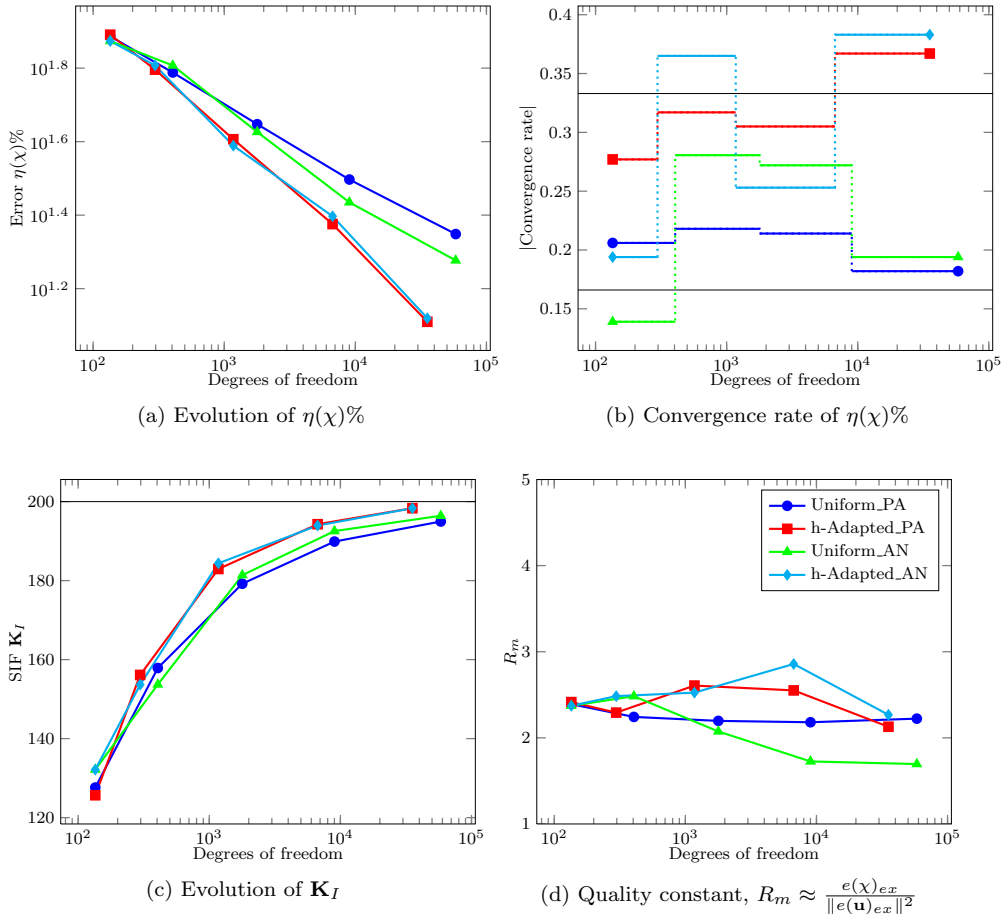


Fig. 16: Infinite sequence of cracks. Analysis with tri-linear elements (L8)

sign velocity field for an immersed boundary method where an h -adapted Cartesian grid is used to mesh the embedding domain. This includes adapting the formulation of shape sensitivities, taking into account the special treatment of boundary conditions required by the use of non body-fitted meshes.

Two problems with known analytical solutions, one with a smooth and another with a singular solution, were used in the section devoted to numerical examples. These examples were used to compare the performance of the proposed methodologies in defining the design velocity field and to demonstrate their appropriate behavior in shape sensitivity analysis, within the framework of immersed boundary techniques based on Cartesian grids.

The proposed physical approach for the definition of the design velocity field provided the best behavior. Furthermore, this approach outperforms the behavior of an analytical velocity field commonly used in the context of fracture mechanics to evaluate the Stress Intensity Factor.

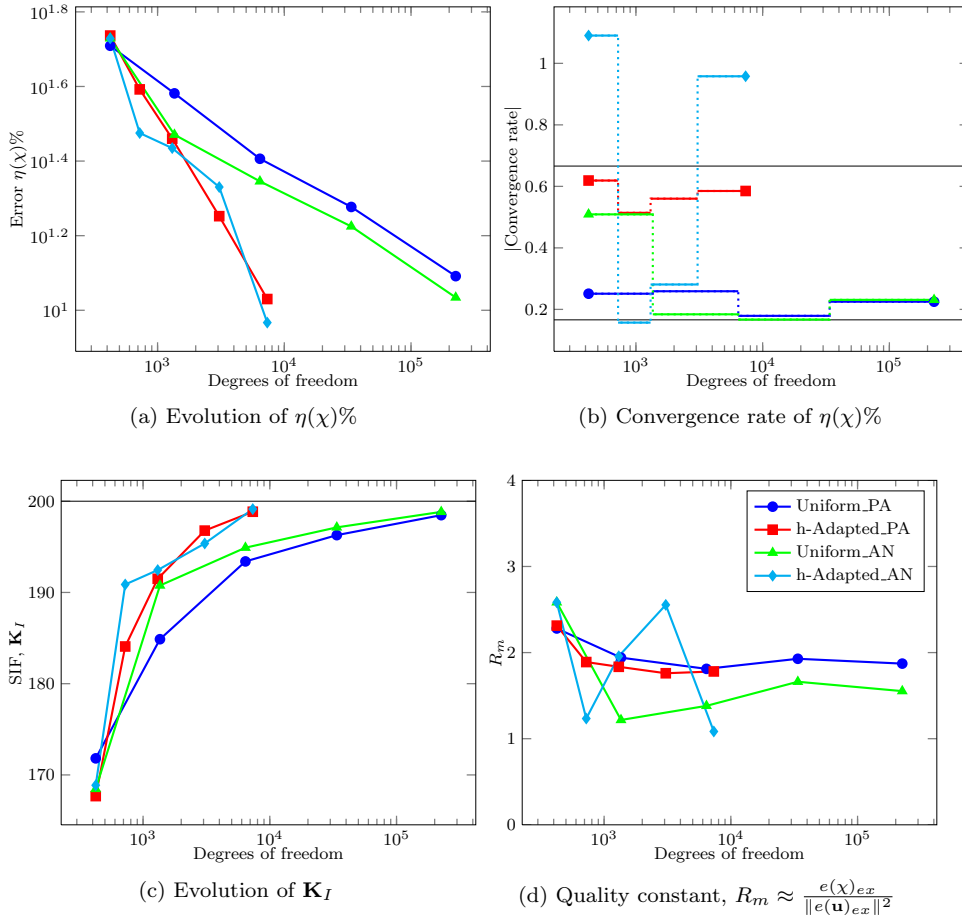


Fig. 17: Infinite sequence of cracks. Analysis with tri-quadratic elements (Q20)

Acknowledgements The authors wish to thank the Spanish *Ministerio de Economía y Competitividad* for the financial support received through the project DPI2013-46317-R and the FPI program (BES-2011-044080), and the *Generalitat Valenciana* through the project PROM-ETEO/2016/007.

References

1. Akgün MA, Garcelon GH, Haftka RT (2001) Fast exact linear and nonlinear structural reanalysis and the sherman-morrison-woodbury formulas. *International Journal for Numerical Methods in Engineering* 50(7):1587–1606
2. Arora JS (1993) An exposition of the material derivative approach for structural shape sensitivity analysis. *Computer Methods in Applied Mechanics and Engineering* 105(1):41–62

3. Arora JS, Lee TH, Cardoso JB (1992) Structural shape sensitivity analysis: relationship between material derivative and control volume approaches. *AIAA Journal* 30(6):1638–1648
4. Belegundu D, Zhang S, Manicka Y, Salagame R (1991) The Natural Approach for Shape Optimization with Mesh Distortion Control. Tech. rep., Penn State University
5. Bennett JA, Botkin ME (1985) Structural Shape Optimization with Geometric Problem Description and Adaptive Mesh Refinement. *AIAA Journal* 23(3):459–464
6. Bischof C, Carle A, Khademi P, Mauer A (1996) The adifor 2.0 system for the automatic differentiation of fortran 77 programs. *IEEE Computational Science and Engineering* 3(3):18–32
7. Braibant V, Fleury C (1984) Shape optimal design using b-splines. *Computer Methods in Applied Mechanics and Engineering* 44(3):247–267
8. Bueda G, Oliver J (1993) A General Methodology for Structural Shape Optimization Problems Using Automatic Adaptive Remeshing. *International Journal for Numerical Methods in Engineering* 36(18):3161–3185
9. Cho S, Ha SH (2009) Isogeometric shape design optimization: exact geometry and enhanced sensitivity. *Structural and Multidisciplinary Optimization* 38(1):53–70
10. Choi K, Kim N (2005) Structural sensitivity analysis and optimization, mechanical engineering series, vol. 1, Springer-Verlag, Berlin, Heidelberg
11. Choi KK, Chang KH (1994) A Study of Design Velocity Field Computation for Shape Optimal Design. *Finite Elements in Analysis and Design* 15(4):317–341
12. Choi KK, Duan W (2000) Design sensitivity analysis and shape optimization of structural components with hyperelastic material. *Computer Methods in Applied Mechanics and Engineering* 187(1–2):219–243
13. Choi KK, Twu SL (1989) Equivalence of continuum and discrete methods of shape design sensitivity analysis. *AIAA Journal* 27(10):1419–1424
14. Choi MJ, Cho S (2014) Isogeometric shape design sensitivity analysis of stress intensity factors for curved crack problems. *Computer Methods in Applied Mechanics and Engineering* 279:469–496
15. Chowdhury MS, Song C, Gao W (2014) Shape sensitivity analysis of stress intensity factors by the scaled boundary finite element method. *Engineering Fracture Mechanics* 116:13–30
16. El-Sayed MEM, Zumwalt KW (1991) Efficient design sensitivity derivatives for multi-load case structures as an integrated part of finite element analysis. *Computers & Structures* 40(6):1461–1467
17. Escobar JM, Montenegro R, Rodríguez E, Cascón JM (2014) The mecano method for isogeometric solid modeling and applications. *Engineering with Computers* 30(3):331–343
18. Fuenmayor FJ, Domínguez J, Giner E, Oliver JL (1997) Calculation of the stress intensity factor and estimation of its error by a shape sensitivity analysis. *Fatigue & Fracture of Engineering Materials & Structures* 20(5):813–828
19. Fuenmayor FJ, Oliver JL, Ródenas JJ (1997) Extension of the Zienkiewicz-Zhu error estimator to shape sensitivity analysis. *International Journal for Numerical Methods in Engineering* 40(8):1413–1433
20. Gil AJ, Arranz-Carreño A, Bonet J, Hassan O (2010) The Immersed Structural Potential Method for Haemodynamic Applications. *Journal of Computational*

- Physics 229(22):8613–8641
21. Giner E, Fuenmayor FJ, Besa AJ, Tur M (2002) An implementation of the stiffness derivative method as a discrete analytical sensitivity analysis and its application to mixed mode in LEFM. *Engineering Fracture Mechanics* 69(18):2051–2071
 22. Griewank A, Juedes D, Utke J (1996) Adol-c, a package for the automatic differentiation of algorithms written in c/c++. *ACM Transactions on Mathematical Software (TOMS)* 22(2):131–167
 23. Ha SH, Choi K, Cho S (2010) Numerical method for shape optimization using T-spline based isogeometric method. *Structural and Multidisciplinary Optimization* 42(3):417–428
 24. Haftka RT (1993) Semi-analytical static nonlinear structural sensitivity analysis. *AIAA Journal* 31(7):1307–1312
 25. Haftka RT, Adelman H (1989) Recent developments in structural sensitivity analysis. *Structural Optimization* 1(3):137–151
 26. Haftka RT, Barthelemy B (1989) *Discretization Methods and Structural Optimization — Procedures and Applications: Proceedings of a GAMM-Seminar October 5–7, 1988, Siegen, FRG*, Springer, Berlin, Heidelberg, chap On the Accuracy of Shape Sensitivity Derivatives, pp 136–144
 27. Haslinger J, Jedelsky D (1996) Genetic algorithms and fictitious domain based approaches in shape optimization. *Struc Optim* 12:257–264
 28. Hughes TJR, Cottrell JA, Bazilevs Y (2005) *Isogeometric Analysis: CAD, Finite Elements, NURBS, Exact Geometry, and Mesh Refinement*. *Computer Methods in Applied Mechanics and Engineering* 194:4135–4195
 29. Iman MH (1982) Three-dimensional Shape Optimization. *International Journal for Numerical Methods in Engineering* 18(5):661–673
 30. Kanninen MF, Popelar CH (1985) *Advanced Fracture Mechanics*, Oxford Engineering Science Series. Oxford University Press
 31. van Keulen F, Haftka R, Kim N (2005) Review of options for structural design sensitivity analysis. Part I: linear systems. *Computer Methods in Applied Mechanics and Engineering* 194(30-33):3213–3243
 32. Kibsgaard S (1992) Sensitivity analysis-the basis for optimization. *International Journal for Numerical Methods in Engineering* 34(3):901–932
 33. Kiendl J, Schmidt R, Wüchner R, Bletzinger KU (2014) Isogeometric shape optimization of shells using semi-analytical sensitivity analysis and sensitivity weighting. *Computer Methods in Applied Mechanics and Engineering* 274:148–167
 34. Kirsch U (2002) *Design-oriented Analysis: a Unified Approach*. Springer Netherlands
 35. Kunisch K, Peichl G (1996) Numerical gradients for shape optimization based on embedding domain techniques. *Comput Optim* 18:95–114
 36. Lee BY (1996) Consideration of body forces in axisymmetric design sensitivity analysis using the bem. *Computers & Structures* 61(4):587–596
 37. Lee BY (1997) Direct differentiation formulation for boundary element shape sensitivity analysis of axisymmetric elastic solids. *International Journal of Solids and Structures* 34(1):99–112
 38. Li FZ, Shih CF, Needleman A (1985) A comparison of methods for calculating energy release rates. *Engineering Fracture Mechanics* 21(2):405–421

39. Li K, Qian X (2011) Isogeometric analysis and shape optimization via boundary integral. *Computer-Aided Design* 43(11):1427–1437
40. Lian H, Bordas SPA, Sevilla R, Simpson RN (2012) Recent developments in the integration of computer aided design and analysis. *Computational Technology Reviews* 6:1–36
41. Lian H, Kerfriden P, Bordas SPA (2016) Implementation of regularized isogeometric boundary element methods for gradient-based shape optimization in two-dimensional linear elasticity. *International Journal for Numerical Methods in Engineering* 106(12):972–1017
42. Liu L, Zhang Y, Hughes TJR, Scott MA, Sederberg TW (2014) Volumetric T-spline Construction using Boolean Operations. *Engineering with Computers* 30(4):425–439
43. Liu WK, Tang S (2007) Mathematical Foundations of the Immersed Finite Element Method. *Computational Mechanics* 39(3):211–222
44. Liu WK, Liu Y, Darell D, Zhang L, Wang XS, Fukui Y, Patankar N, Zhang Y, Bajaj C, Lee J, Hong J, Chen X, Hsu H (2006) Immersed Finite Element Method and its Applications to Biological Systems. *Computer Methods in Applied Mechanics and Engineering* 195(13):1722–1749
45. Marco O, Sevilla R, Zhang Y, Ródenas JJ, Tur M (2015) Exact 3D boundary representation in finite element analysis based on Cartesian grids independent of the geometry. *International Journal for Numerical Methods in Engineering* 103:445–468
46. Moita JS, Infanta J, Mota CM (2000) Sensitivity analysis and optimal design of geometrically non-linear laminated plates and shells. *Computers & Structures* 76(1–3):407–420
47. Nadal E (2014) Cartesian Grid FEM (cgFEM): High Performance h-adaptive FE Analysis with Efficient Error Control. Application to Structural Shape Optimization. PhD Thesis. Universitat Politècnica de València
48. Nadal E, Ródenas JJ, Albelda J, Tur M, Tarancón JE, Fuenmayor FJ (2013) Efficient Finite Element Methodology based on Cartesian Grids: Application to Structural Shape Optimization. *Abstract and Applied Analysis* 2013
49. Navarrina F, López-Fontán S, Colominas I, Bendito E, Casteleiro M (2000) High order shape design sensitivities: a unified approach. *Computer Methods in Applied Mechanics and Engineering* 188(4):681–696
50. Neittaanmäki P, Salmenjoki K (1989) *Computer Aided Optimum Design of Structures: Recent Advances*, Springer-Verlag, Berlin, Heidelberg, chap Comparison of Various Techniques for Shape Design Sensitivity Analysis, pp 367–377
51. Nguyen VP, Anitescu C, Bordas SPA, Rabczuk T (2015) Isogeometric analysis: An overview and computer implementation aspects. *Mathematics and Computers in Simulation* 117:89–116
52. Ozaki I, Kimura F, Berz M (1995) Higher-order sensitivity analysis of finite element method by automatic differentiation. *Computational Mechanics* 16(4):223–234
53. Pandey PC, Bakshi P (1999) Analytical response sensitivity computation using hybrid finite elements. *Computers & Structures* 71(5):525–534
54. Peskin CS (1977) Numerical Analysis of Blood Flow in the Heart. *Journal of Computational Physics* 25:220–252

55. Phelan DG, Haber RB (1989) Sensitivity analysis of linear elastic systems using domain parametrization and a mixed mutual energy principle. *Computer Methods in Applied Mechanics and Engineering* 77(1–2):31–59
56. Piegl L, Tiller W (1995) *The NURBS Book*. Springer-Verlag
57. Poldneff MJ, Rai IS, Arora JS (1993) Implementation of design sensitivity analysis for nonlinear structures. *AIAA Journal* 31(11):2137–2142
58. Qian X (2010) Full analytical sensitivities in NURBS based isogeometric shape optimization. *Computer Methods in Applied Mechanics and Engineering* 199(29–32):2059–2071
59. Rice JR (1968) A Path Independent Integral and the Approximate Analysis of Strain Concentration by Notches and Cracks. *Journal of Applied Mechanics* 35(2):379–386
60. Ródenas JJ (2001) Error de Discretización en el Cálculo de Sensibilidades mediante el Método de los Elementos Finitos. PhD Thesis
61. Ródenas JJ, Fuenmayor FJ, Tarancón JE (2003) A Numerical Methodology to Assess the Quality of the Design Velocity Field Computation Methods in Shape Sensitivity Analysis. *International Journal for Numerical Methods in Engineering* 59(13):1725–1747
62. Rodenas JJ, Tur M, Fuenmayor FJ, Vercher A (2007) Improvement of the superconvergent patch recovery technique by the use of constraint equations: the SPR-C technique. *International Journal for Numerical Methods in Engineering* 70(6):705–727
63. Ródenas JJ, Bugeda G, Albelda J, Oñate E (2011) On the need for the use of error-controlled finite element analyses in structural shape optimization processes. *International Journal for Numerical Methods in Engineering* 87(11):1105–1126
64. Rogers DF (2001) *An Introduction to NURBS: with Historical Perspective*. Elsevier
65. Schramm U, Pilkey WW (1993) The coupling of geometric descriptions and finite elements using NURBS a study in shape optimization. *Finite Elements in Analysis and Design* 15(1):11–34
66. Sevilla R, Fernández-Méndez S, Huerta A (2011) 3D-NURBS-enhanced Finite Element Method (NEFEM). *International Journal for Numerical Methods in Engineering* 88(2):103–125
67. Sevilla R, Fernández-Méndez S, Huerta A (2011) NURBS-enhanced Finite Element Method (NEFEM): A Seamless Bridge Between CAD and FEM. *Archives of Computational Methods in Engineering* 18(4):441–484
68. Shiriaev D, Griewank A (1996) Adol-f: Automatic differentiation of fortran codes. *Computational Differentiation: Techniques, Applications, and Tools (SIAM)* 1:375–384
69. Shivakumar KN, Raju IS (1992) An equivalent domain integral method for three-dimensional mixed-mode fracture problems. *Engineering Fracture Mechanics* 42(6):935–959
70. Silva CAC, Bittencourt ML (2007) Velocity fields using NURBS with distortion control for structural shape optimization. *Structural and Multidisciplinary Optimization* 33(2):147–159
71. Tur M, Albelda J, Nadal E, Ródenas JJ (2014) Imposing dirichlet boundary conditions in hierarchical cartesian meshes by means of stabilized lagrange multipliers. *International Journal for Numerical Methods in Engineer-*

- ing 98(6):399–417
72. Tur M, Albelda J, Marco O, Ródenas JJ (2015) Stabilized Method to Impose Dirichlet Boundary Conditions using a Smooth Stress Field. *Computer Methods in Applied Mechanics and Engineering* 296:352–375
 73. Wujek BA, Renaud JE (1998) Automatic differentiation for more efficient system analysis and optimization. *Engineering Optimization* 31(1):101–139
 74. Yang RJ, Fiedler MJ (1987) Design Modelling for Large-Scale Three-dimensional Shape Optimization Problems. *ASME Computers in Engineering* 15:177–182
 75. Yoon BG, Belegundu AD (1988) Iterative methods for design sensitivity analysis. *AIAA Journal* 26(11):1413–1415
 76. Zhang L, Gerstenberger A, Wang X, Liu WK (2004) Immersed Finite Element Method. *Computer Methods in Applied Mechanics and Engineering* 293(21):2051–2067
 77. Zhang W, Beckers P (1989) *Computer Aided Optimum Design of Structures: Recent Advances*, Springer-Verlag, Berlin, Heidelberg, chap Comparison of Different Sensitivity Analysis Approaches for Structural Shape Optimization, pp 347–356
 78. Zhang Y, Wang W, Hughes TJR (2013) Conformal Solid T-spline Construction from Boundary T-spline Representations. *Computational Mechanics* 6(51):1051–1059
 79. Zienkiewicz OC, Zhu JZ (1987) A Simple Error Estimator and Adaptive Procedure for Practical Engineering Analysis. *International Journal for Numerical Methods in Engineering* 24(2):337–357
 80. Zienkiewicz OC, Taylor RL, Zhu J (2013) *The Finite Element Method: Its Basis and Fundamentals*, seventh edition edn. Butterworth-Heinemann, Oxford