

Spline parameterization method for 2D and 3D geometries based on T-mesh optimization

J.I. López^a, M. Brovka^a, J.M. Escobar^a, R. Montenegro^a, G.V. Socorro^a

^a*University of Las Palmas de Gran Canaria, University Institute for Intelligent Systems and Numerical Applications in Engineering (SIANI), Spain.*

Abstract

We present a method to obtain high quality spline parameterization of 2D and 3D geometries for their use in isogeometric analysis. As input data, the proposed method demands a boundary representation of the domain, and it constructs automatically a spline transformation between the physical and the parametric domains. Parameterization of the interior of the object is obtained by deforming isomorphically an adapted parametric T-mesh onto the physical domain by applying a T-mesh untangling and smoothing procedure, which is the key of the method. The T-mesh optimization is based on the mean ratio shape quality measure. The spline representation of the geometry is calculated by imposing interpolation conditions using the data provided by one-to-one correspondence between the meshes of the parametric and physical domains. We give a detailed description of the proposed technique and show some examples. Also, we present some examples of the application of isogeometric analysis in geometries parameterized with our method.

Keywords: spline parameterization, isogeometric analysis, T-mesh optimization, hexahedral mesh optimization, shape quality metric

1. Introduction

One of the open problems of isogeometric analysis [1, 2, 3] is how to obtain a spline parameterization of the interior of a computational domain from the description of its boundary. CAD models usually provide only the boundary surface of a solid. The application of isogeometric analysis requires a full volumetric rep-

Email address: mbrovka@siani.es (M. Brovka)

resentation of the geometry. A parameterization is suitable for analysis if there are no self-intersections. Moreover, in order to obtain an accurate numerical result, it is necessary to have a good quality parameterization. Orthogonality and uniformity are desirable for the tensor-product structured parameterization. It is not a trivial task to obtain a good quality smooth global parametric mapping for complex domains, and it can be very time-consuming. IGA requires an effective method to construct a spline parameterization from the boundary of the object. Here are some attempts to tackle the problem.

Zhang et al. described in [4] a procedure to construct a T-spline representation of a genus-zero solid from its boundary triangulation. The method is based on the control net optimization procedure. To remove tangled elements, the authors maximize the worst Jacobian of the trilinear map for each element, and the smoothing is performed by moving each node towards the mass center of its neighboring elements. In [5] the method was extended for an arbitrary genus topology. Some additional techniques based on polycubes decomposition was developed in [6, 7] for treating more complex cases. An optimization-based method is used in [8, 9]. To find the optimal values for the control points of B-spline planar domain the authors minimize the Winslow functional with certain constraints to assure the validity of the mapping. Another similar approach is developed in [10], where the B-spline parameterization is found by solving a constraint optimization problem. Constraints are defined by imposing injectivity sufficient conditions, and the optimization consists in minimizing some energy functional in order to reach a good orthogonality and uniformity of the parametric mapping. The idea was extended for 3D case in [11]. Another similar technique was proposed by these authors in [12, 13]. They use a harmonic mapping obtained by solving an optimization problem for the control points. Some additional terms are added to the objective function in order to improve the quality of the spline mapping. The use of harmonic mapping is a common characteristic of several works dealing with 2D and 3D parameterization methods. For example, Li et al. [14] construct a harmonic volumetric mapping through a meshless procedure by using a boundary method. Martin et al. [15] present a methodology based on discrete harmonic functions to parameterize a solid, where the input data are surface triangulation and a tetrahedral mesh of the solid. Several Laplace equations are solved using FEM to obtain two orthogonal harmonic functions, which are used to construct a structured hexahedral mesh of the solid. In [16] the authors proposed harmonic mapping based parameterization method for planar domains using THB-splines. First, a harmonic mapping from the physical to the parametric domain is found via the boundary element method. Then, this harmonic mapping should be inverted to provide the

desired mapping from the parametric to the physical domain. An approximation of this inverse is obtained via a minimization problem with the control points as unknown variables. The objective function also includes some additional terms to obtain a better quality spline parameterization.

In this paper, we propose a different approach. To obtain the parameterization of the object Ω (physical domain) we deform a T-mesh of the parametric domain, the unitary cube $\widehat{\Omega} = [0, 1]^d$, $d = 2, 3$, until it achieves the shape of the object. This deformation only affects the positions of the nodes, that is, there is not any change in their connectivities: we say that both meshes are isomorphic. Given that a point can be fully determined by the local coordinates relative to the cell in which it is contained, we can define a one-to-one mapping between parametric domain $\widehat{\Omega}$ and the object. Mesh deformation is accomplished by means of the simultaneous T-mesh untangling and smoothing procedure based on a pointwise mean ratio quality measure. **This quality metric gains both good orthogonality and uniformity of the isoparametric curves and forces the mapping to be as conformal as possible. Besides, in contrast to other optimization-based methods, we do not need to impose additional constraints to guarantee the validity of the mapping. Then, spline representation of the object is calculated by imposing interpolation conditions** using the data provided by one-to-one correspondence between the mesh of the parametric domain and the mesh of the physical object. The interpolation is performed using polynomial spline spaces constructed via a technique proposed in our previous work [17]. This simple strategy allows to define easily a cubic spline space with nice properties over a given strongly balanced quadtree/octree T-mesh. However, any other spline basis defined over a T-mesh can be used.

Globally smooth 3D parametrization entails some important limitations due to the tensor-product nature of spline functions. First, for complex domains it is necessary to subdivide the object and perform several-patches parameterization with loss of regularity in some zone of the merging. Second, for the given boundary surface of a solid it is not trivial to find optimal partition of this surface into six patches corresponding to six faces of the parametric cube. Besides, to guarantee regular parameterization (positive Jacobian), the boundary patches must merge in a convex local region, which can be impossible for some geometries. Here, we are not going to tackle these problems inherent to tensor-product spline parameterization. Thus, in this paper, we deal with genus-zero geometries given by six boundary patches that merge forming convex regions around the junction.

The paper is organized as follows. In next section we describe the main steps of the proposed algorithm. In Section 3 we explain the process of the construction of the T-mesh adapted to the singularities of the object boundary. Section 4

describes the T-mesh optimization procedure which is the key of the method. The modeling of the geometry by means of cubic splines is developed in Section 5. A quality improvement strategy, based on adaptive refinement, is described in Section 6. In Section 7 we show some examples of the application of isogeometric analysis in geometries parameterized with our method. Finally, in section 8 we present the conclusions and set out some future lines.

2. General scheme of the method

The method is based on the ideas of our previous works on mesh untangling and smoothing and the Meccano method [18, 19, 20].

The algorithm includes the following stages:

1. *Construction of an adapted T-mesh:* An adapted T-mesh is generated by refining the initial mesh until the input boundary is approximated with a prescribed tolerance. During this process, the boundary nodes of the parametric domain are mapped to the boundary of the object.
2. *T-Mesh optimization:* The inner nodes of the physical T-mesh are relocated by applying a simultaneous untangling and smoothing procedure.
3. *Construction of a spline representation of the geometry:* Spline representation of the object is calculated by interpolating the data provided by the transformation between the parametric and physical meshes of the object.
4. *Adaptive refinement to improve the mesh quality (optional step):* If the quality of the mesh is not satisfactory, we apply an adaptive refinement of the mesh in order to increase the degree of freedom in the areas with high distortion. Then, we return to step 2 and repeat the process until reaching a good spline parameterization.

The T-meshes used in this work are hierarchical meshes with a quadtree/octree structure [21], which are frequently used in engineering. Due to their simplicity, tree-structured meshes are an attractive tool for performing adaptive refinement in IGA and geometric modelling. For spline representation of the object we use polynomial spline spaces constructed via the technique described in [17]. This strategy allows to define easily a cubic spline space with nice properties over a given strongly balanced quadtree/octree T-mesh. A strongly balanced mesh implies that any cell has contact only with cells that differ at most in one level of depth, and it can be straightforwardly obtained using a standard balancing procedure.

As was said before, the proposed parameterization algorithm is suitable for 2D and 3D geometries. It is worth noting that our 3D parameterization method is a direct generalization of the 2D case. To facilitate the illustration of the whole process, we are going to give a description of all steps of the method both for 2D and 3D domains.

In the next sections we describe the details of each stage.

3. Construction of an adapted T-mesh

Here, we describe the process to construct an adapted T-mesh that allows to represent the features of the physical object Ω with a desired tolerance.

3.1. 2D case

For planar domains we assume that the input boundary is given by four spline curves or polygonals. Also, we assume that the four curves (polygonals) meet in a convex zone (the inner angle is less than 180°). If the input boundary is given by spline parametric curves, we take the sides of the square as their parametric spaces to map them to the input boundary. If the input boundary are polygonals, each edge of the unit square are mapped into its corresponding polygonal via chord-length parameterization.

Let Γ^i ($i = 1, \dots, 4$) be the four boundary curves of the object Ω , and $\widehat{\Gamma}^i$ be the four sides of the parametric cube $\widehat{\Omega} = [0, 1]^2$. Given the four parametrizations $\Pi_b^i : \widehat{\Gamma}^i \rightarrow \Gamma^i$, we can define a global parametric mapping between the cube boundary $\partial\widehat{\Omega} = \widehat{\Gamma} = \bigcup_{i=1}^4 \widehat{\Gamma}^i$ and the boundary of the object $\partial\Omega = \Gamma = \bigcup_{i=1}^4 \Gamma^i$:

$$\Pi_b : \widehat{\Gamma} \rightarrow \Gamma \quad (1)$$

Next, we construct an adapted quadtree T-mesh to approximate the input boundary with a predefined tolerance ε . Let consider that $\widehat{\Omega}_k$ is a T-mesh of $\widehat{\Omega}$ resulting after applying several quadtree refinements according to some approximation error criterion. Then, after mapping $\widehat{\Gamma}_k = \partial\widehat{\Omega}_k$ onto the physical space using Π_b , we obtain a new boundary Γ_k which is an approximation for the input boundary Γ , i.e.

$$\widehat{\Gamma}_k \rightarrow \Pi_b(\widehat{\Gamma}_k) =: \Gamma_k \approx \Gamma \quad (2)$$

In order to improve this approximation we must refine the cells of $\widehat{\Omega}_k$, that contact with the boundary, in such a way that the *distance* between Γ_k and Γ decreases until reaching a prescribed tolerance ε . In this work, the following simple approximation criterion is used. We evaluate the area of the triangle formed by the edge

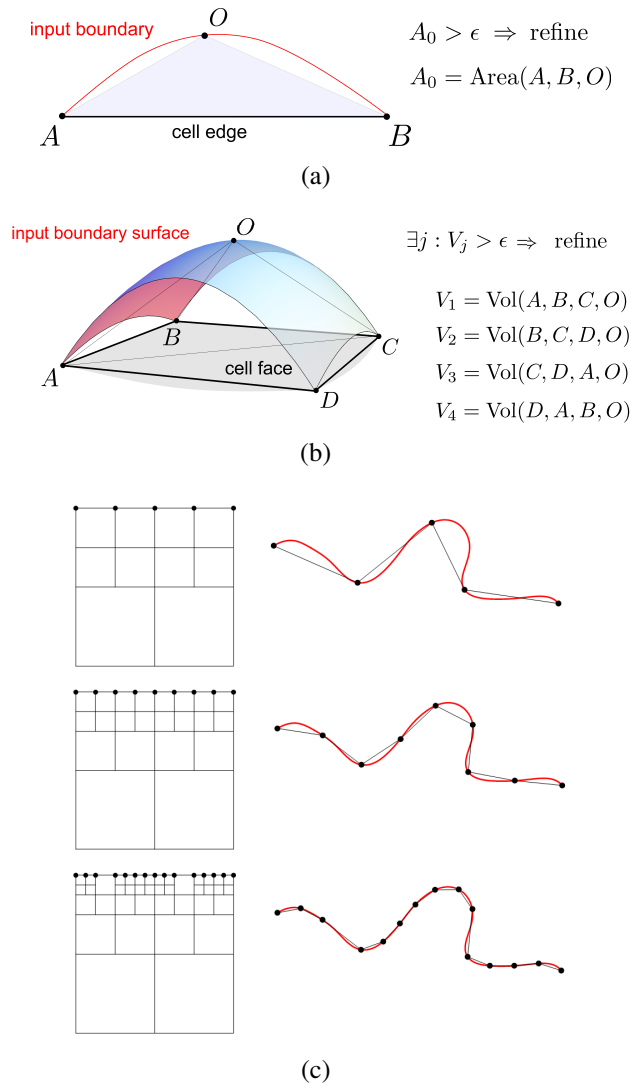


Figure 1: Step 1 of the method: construction of an adapted T-mesh. (a) Error criterion for the boundary approximation in 2D. (b) Error criterion for the boundary surface approximation in 3D. (c) Example of adaptive refinement to approximate the object boundary with a desired tolerance.

AB of the physical cell and the image of the edge center O over the input boundary, see Fig. 1(a). A cell is refined if the error is greater than ϵ . Each refinement produces a new boundary point that is projected over the input boundary, obtaining a more accurate approximation of the geometry, as illustrated in Fig. 1(c).

3.2. 3D case

For 3D geometries we assume that the input boundary is given by six spline surfaces, see Fig. 3, or six surface triangulations, merging in a convex region. If the input boundary is given by spline parametric surfaces, we take the faces of the cube as their parametric spaces. If the input surfaces are triangulations, the parameterization is accomplished by the method proposed by M. Floater in [22]. This method establishes a one-to-one mapping between the surface triangulation and a planar triangulation of the parametric domain, which is used to map a face of the unit cube to the input boundary.

Let Γ^i ($i = 1, 2, \dots, 6$) be the six boundary surfaces of the object Ω , and $\widehat{\Gamma}^i$ be the surfaces of the parametric cube $\widehat{\Omega} = [0, 1]^3$. Given the parametrizations $\Pi_b^i : \widehat{\Gamma}^i \rightarrow \Gamma^i$, we can define a global parametric mapping between the cube boundary $\partial\widehat{\Omega} = \widehat{\Gamma} = \bigcup_{i=1}^6 \widehat{\Gamma}^i$ and the boundary of the object $\partial\Omega = \Gamma = \bigcup_{i=1}^6 \Gamma^i$:

$$\Pi_b : \widehat{\Gamma} \rightarrow \Gamma \quad (3)$$

An adapted parametric octree T-mesh $\widehat{\Omega}_k$ is constructed analogously to 2D case, that is, $\widehat{\Omega}$ is refined according some approximation error criterion. As approximation error, we evaluate the volume of the four tetrahedra $ABCO$, $BCDO$, $CDAO$, $DABO$ constructed from the boundary cell face $ABCD$, where O is the image of the face center, see Fig. 1(b). The cell is refined if there is at least one volume greater than ε . A cell refinement introduces at most five new boundary points that are projected over the input surface, obtaining a more accurate approximation of the geometry. Then, after mapping the $\widehat{\Gamma}_k = \partial\widehat{\Omega}_k$ into the physical space, we obtain a new boundary Γ_k which is an approximation for the input boundary Γ , i.e.

$$\widehat{\Gamma}_k \rightarrow \Pi_b(\widehat{\Gamma}_k) =: \Gamma_k \approx \Gamma \quad (4)$$

4. T-mesh optimization

As a result of the previous stage, the boundary $\widehat{\Gamma}_k$ of the adapted T-mesh $\widehat{\Omega}_k$ is mapped to the boundary of the object, generating the surface Γ_k , which is the final approximation of Γ . The position of the inner nodes are determined by means of the T-mesh optimization procedure developed in this section, which is the key of the method. Once the inner nodes are relocated, the resulting physical T-mesh Ω_k is an approximation of the original geometry. The corresponding piece-wise volumetric parameterization is denoted as

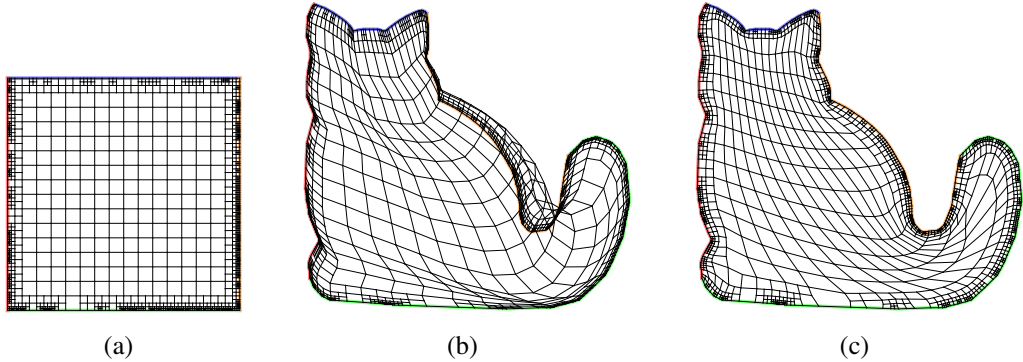


Figure 2: Stages of T-mesh construction for the *Cat* test model geometry. (a) Parametric T-mesh adapted to the boundary of the geometry; (b) previous relocation by Coons patch, tangled mesh; (c) optimized physical T-mesh. The color lines represent the correspondence between the parametric and physical boundaries.

$$\Pi : \widehat{\Omega}_K \rightarrow \Omega_K, \quad (5)$$

where any point p of a cell of $\widehat{\Omega}_K$ is mapped into a point q of the transformed cell of Ω_K by using an appropriate local mapping related to the cell.

To reduce the computational effort during the optimization process it is preferable to perform a previous relocation of the inner nodes. For this purpose Laplacian smoothing or Coons patch [23, 24] can be used. This previous relocation facilitates the untangling process, but in general does not obtain a satisfactory mesh quality and can produce self-intersections. Therefore, it is essential to apply an effective optimization algorithm. Figure 2(a) shows an example of the adapted parametric T-mesh constructed in the previous stage, Fig. 2(b) shows the tangled physical T-mesh after applying Coons patch, and Fig. 2(c) illustrates the resulting optimized T-mesh. An example of a 3D adapted parametric T-mesh and the corresponding optimized physical T-mesh of the solid is shown in Fig. 3.

The mesh optimization procedure consists in an iterative process where each node is moved to a new position in order to improve the quality of the *local sub-mesh*, which is the set of elements connected to the free node. This new position of the node is determined by minimizing an objective function based on the *mean ratio* shape quality measure. This algebraic quality measure was originally introduced for a triangle, and it is defined in terms of the Jacobian matrix of the affine mapping from the ideal (target) triangle to the given one. It represents the

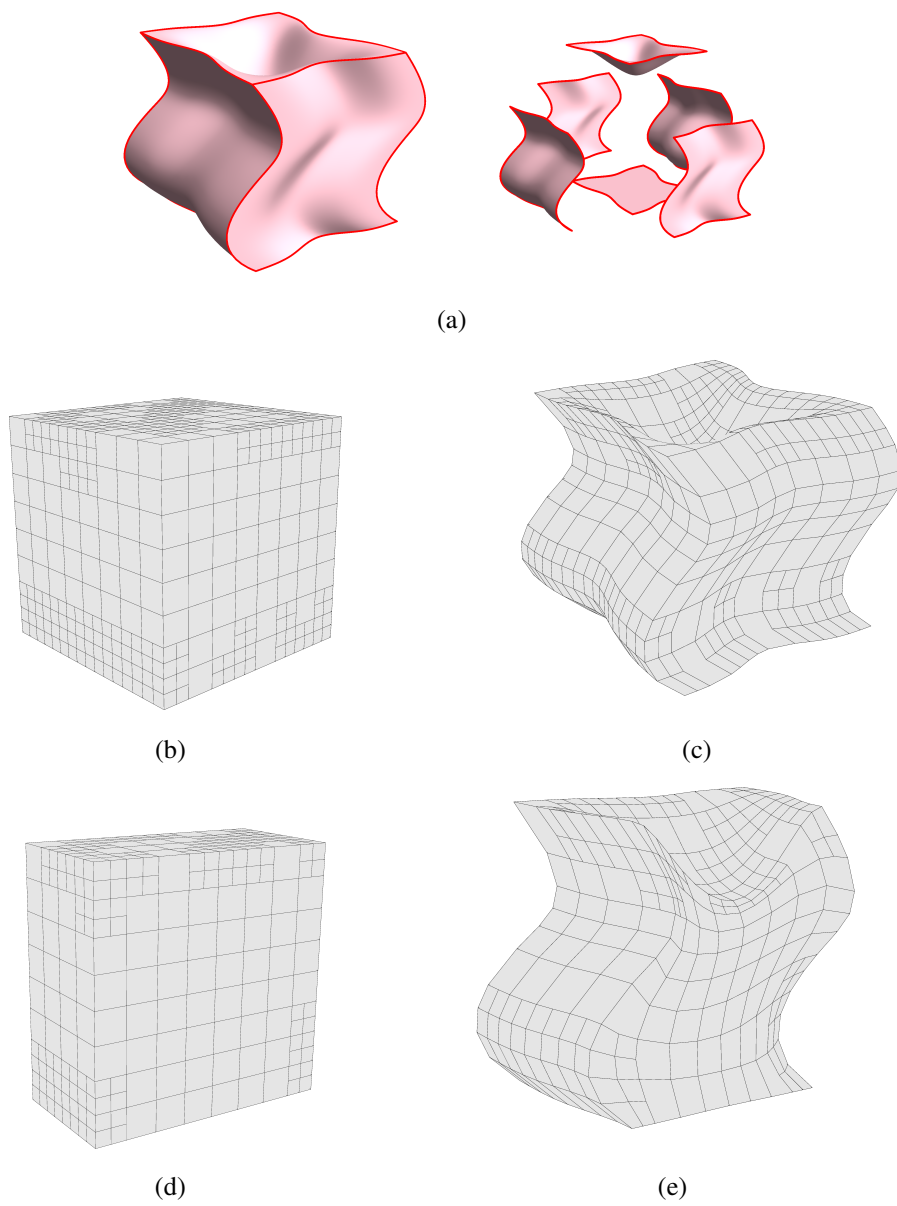


Figure 3: Test model 1. (a) Parametric T-mesh adapted to the boundary of the geometry; (b) optimized physical T-mesh; (c) a section of parametric T-mesh ; (d) a section of physical T-mesh.

deviation of the given triangle from the ideal one. The distortion of an element is defined as the inverse of its quality.

The objective function for a given free node is constructed as a sum of shape distortion measures of all elements of its local submesh. For each cell of the physical T-mesh, the corresponding parametric cell is used as its ideal element. Repeating the optimization procedure for all inner nodes we achieve to minimize the deformation of the physical mesh with respect to the parametric one.

4.1. *Jacobian-based shape quality measure for a triangle/tetrahedral mesh optimization*

Here we briefly revise the concept of algebraic shape quality metric for a linear simplex element, see [25, 26], and the objective function we use for a linear triangle/tetrahedral mesh optimization. For more details see [27, 20, 28, 29].

Let S be the Jacobian matrix of the affine mapping that takes the ideal element (target) τ_I into the physical d -simplex element τ ($d = 2, 3$). Then, quality metrics of the simplex τ can be defined in terms of the matrix S . For example, *mean ratio*

$$q = \frac{d \sigma^{2/d}}{\|S\|^2},$$

is a scale, rotation and translation invariant algebraic quality metric of τ , where $\sigma = \det(S)$ and $\|S\|$ is the Frobenius norm of S . The maximum value of q is 1, and it is reached when $S = \mu R$, where μ is a scalar and R is a rotation matrix. In other words, q attains its maximum if and only if τ and τ_I are similar (same shape). Besides, any degenerate simplex has quality measure 0. The distortion measure is defined as the inverse of its quality, i.e.,

$$\eta = \frac{1}{q} = \frac{\|S\|^2}{d \sigma^{2/d}}.$$

Let $\mathbf{x} \in \mathbb{R}^d$ be a free node whose new position is to be determined. The objective function to be minimized is defined as

$$K(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^N \eta_i = \frac{1}{N} \sum_{i=1}^N \frac{\|S_i\|^2}{d \sigma_i^{2/d}},$$

where N is the number of elements connected to the free node \mathbf{x} and η_i is the distortion measure for the i -th element of the local mesh.

The objective function K becomes discontinuous when the area of any triangle tends to zero. Due to these singularities, the function K improves the quality of valid elements but it does not work properly when the mesh is tangled ($\sigma \leq 0$).

In [20] we proposed a modification of K by replacing the determinant σ by the positive, smooth and increasing function $h(\sigma) = \frac{1}{2}(\sigma + \sqrt{\sigma^2 + 4\delta^2})$. Then, the modified distortion becomes

$$\eta^* = \frac{\|\mathcal{S}\|^2}{dh(\sigma)^{2/d}}.$$

This regularization eliminates the asymptotes associated with their singularities and the new objective function

$$K^*(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^N \eta_i^*$$

becomes smooth in \mathbb{R}^d . In the feasible region (subset of \mathbb{R}^d where the free node could be placed for the local mesh to be valid) the modified objective function K^* approximates the original function K as $\delta \rightarrow 0$, and then, the minimum of the original and modified objective functions are nearly identical when δ is small. When this region does not exist, the minimum of the modified objective function is located in such a way that it tends to untangle the local mesh. Thus, the regularized objective function allows the simultaneous untangling and smoothing of the mesh. The value of δ is selected in terms of the local mesh under consideration, making it as small as possible and in such a way that the evaluation of the minimum of the modified function does not present any computational problem. For more details see [20, 30]. The unconstrained optimization problem can be easily solved with any standard method, see for example [31].

4.2. *Jacobian-based shape quality metric for arbitrary elements*

To extend the concept of the shape quality metric for non-simplicial elements (or for a higher-order simplicial ones) with non-constant Jacobian matrix we introduce the following concepts.

Let Ω_e be any type physical cell of the local mesh for the free node \mathbf{x} . First, we define a pointwise distortion measure for the given cell using some appropriate mapping $\mathbf{G} : \hat{\Omega}_e \rightarrow \Omega_e$ from the ideal element to the physical one. Here, we will use quadratic isoparametric mapping.

Let $J_G(\boldsymbol{\xi})$ be the Jacobian matrix of the mapping \mathbf{G} at point $\boldsymbol{\xi}$. Then, the regularized distortion at any point $\boldsymbol{\xi}$ of the ideal element is defined as

$$\eta^*(\boldsymbol{\xi}) = \frac{\|J_G(\boldsymbol{\xi})\|^2}{dh(\sigma(\boldsymbol{\xi}))^{2/d}}, \quad \boldsymbol{\xi} \in \hat{\Omega}_e.$$

where $\sigma(\boldsymbol{\xi}) = \det(J_G(\boldsymbol{\xi}))$.

To assess the distortion of the entire physical element Ω_e we can define a global distortion measure (similarly to [32, 33, 34]) by means of L^1 norm of the pointwise distortion, namely

$$\eta_{\Omega_e}^* = \frac{1}{V_{\widehat{\Omega}_e}} \int_{\widehat{\Omega}_e} \eta^*(\boldsymbol{\xi}) d\widehat{\Omega}_e, \quad (6)$$

where $V_{\widehat{\Omega}_e}$ is the volume of $\widehat{\Omega}_e$. Other norms, for example L^2 , can be used.

Remark 1. Note that for 2D case, using in (6) a not regularized inverse mean ratio distortion metric $\eta = \frac{\|S\|^2}{2\sigma}$, the resulting expression is equivalent to the Winslow functional frequently used for planar parameterization methods, see [8, 9]. However, our global distortion measure based on the regularized distortion η^* allows to perform unconstrained optimization pursuing two goals at the same time: a valid parametric mapping (positive Jacobian) and a mapping as conformal as possible.

The global regularized distortion $\eta_{\Omega_e}^*$ takes values between 1 and ∞ . If the physical element Ω_e coincides with the ideal one, then $\eta^*(\boldsymbol{\xi}) \rightarrow 1$ when $\delta \rightarrow 0$. On the other hand, this global distortion measure presents *pseudo-barriers*, that is, $\eta_{\Omega}^* \rightarrow \infty$ when $\sigma(\boldsymbol{\xi}) \leq 0$ and $\delta \rightarrow 0$.

The global quality of the element is defined as the inverse of its global distortion, i.e.

$$q_{\Omega_e}^* = \frac{1}{\eta_{\Omega_e}^*},$$

and it takes values from $[0, 1]$.

We define the objective function for a free node \mathbf{x} using the introduced global distortion measure for the cells

$$K^*(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^N \eta_{\Omega_i}^*(\mathbf{x}), \quad (7)$$

being N the number of cells connected to the free node \mathbf{x} and $\eta_{\Omega_i}^*(\mathbf{x})$ the global distortion measure for the i -th cell.

The integrals in (7) are evaluated by a numerical quadrature. Let $\{\boldsymbol{\xi}_j\}_{j=1,M}$ be a set of quadrature points over the ideal element $\widehat{\Omega}_e$. Then the objective function for the free node \mathbf{x} becomes

$$K^*(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^N \left(\sum_{j=1}^M w_j \eta_{\Omega_i}^*(\boldsymbol{\xi}_j) \right), \quad (8)$$

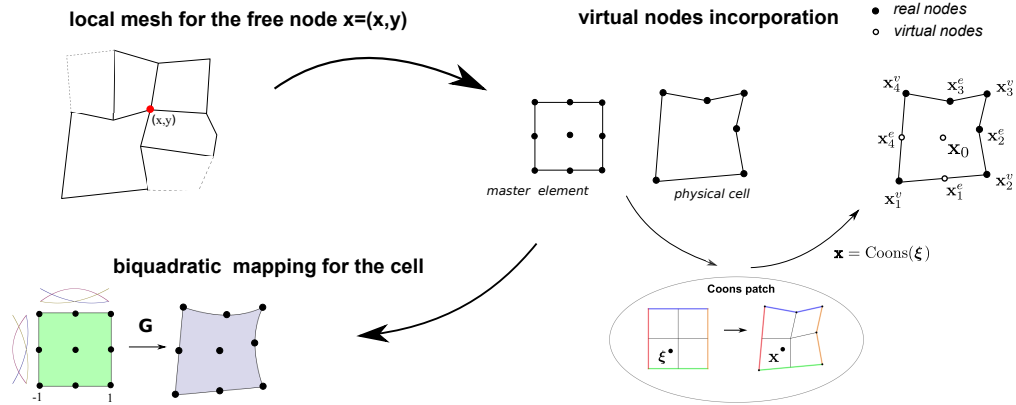


Figure 4: Defining a local quadratic mapping for each cell. Virtual nodes assigning for 2D case.

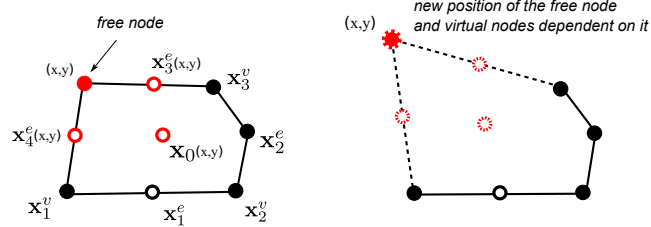


Figure 5: Virtual nodes of the cell connected to the free node update their position when the free node is moved during optimization process.

where w_j is the corresponding weight for the quadrature point ξ_j . Thus, in practice, the objective function is based on the pointwise distortion measure, evaluated at certain set of sample points within the ideal cell. Analogously to the triangular mesh, the unconstrained optimization problem can be easily solved with any standard method. To maintain a low computational cost of the minimization, it is preferable to use a few quadrature points. Also, it is preferable to evaluate the quality at the corner vertices of the cell, because usually the major distortion occurs at these points. Therefore, we use Gauss-Lobatto quadrature rule that includes boundary quadrature points. More specifically, we use three quadrature points $\{-1, 0, 1\}$ of the reference interval $[-1, 1]$.

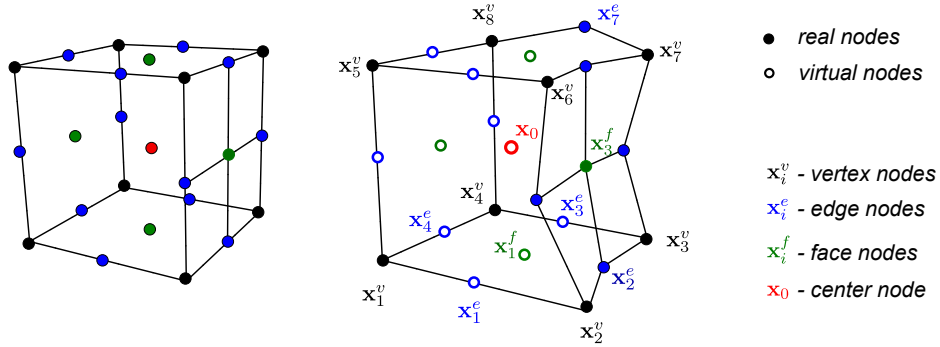


Figure 6: Defining a local triquadratic mapping for the cell. Virtual nodes assigning for 3D case.

4.3. Parametric mapping for the elements of T-mesh

Here we explain how we define the local parametric mapping \mathbf{G} for the elements of T-mesh used to define their global distortion measures.

4.3.1. 2D case

As was said before, for each cell of the physical T-mesh its counterpart in parametric domain is used as its ideal element. In our particular case of quadtree subdivisions all parametric cells are squares, so we take the master element $\widehat{\Omega}_e = [-1, 1]^2$ as the ideal element for all cells. Physical cells of our T-mesh, in general, are not straight-sided quadrilaterals since they can contain hanging nodes, which are not aligned with the corner nodes of the edge. For our balanced quadtree T-meshes an edge can contain at most one hanging node. To take into account all mesh nodes for the distortion measure of a given cell, we can consider the curvilinear second order element constructed via Lagrangian interpolation. Let $\xi_i \in \mathbb{R}^2$, $i = 1, \dots, 9$, be the nodes of the master element $\widehat{\Omega}_e$, and N_i are the nodal biquadratic Lagrangian shape functions defined on $\widehat{\Omega}_e$. Then, parametric mapping

$$\mathbf{G}(\xi) = (x(\xi, \eta), y(\xi, \eta)) = \sum_{i=1}^9 \mathbf{x}_i N_i(\xi) \quad (9)$$

maps $\widehat{\Omega}_e$ into the physical cell Ω_e with the nodes \mathbf{x}_i , $i = 1, \dots, 9$. So, to define the quadratic transformation \mathbf{G} , 9 cell nodes are needed. In our case some of these 9 nodes may not exist in the mesh, so we have to complement a given cell with some virtual nodes. The position of these virtual nodes depends on the real (existing) nodes of the cell, see Fig. 4(a). To assign reasonable positions to the

virtual nodes we appeal to Coons patch interpolation of the boundary edges of the cell. Suppose we define parameterizations of the four edges as $\mathbf{x}(0, \eta)$, $\mathbf{x}(1, \eta)$, $\mathbf{x}(\xi, 0)$ and $\mathbf{x}(\xi, 1)$. The bilinear Coons patch is defined as

$$\begin{aligned} \mathbf{x}(\xi, \eta) = & (1 - \xi)\mathbf{x}(0, \eta) + \xi\mathbf{x}(1, \eta) \\ & + (1 - \eta)\mathbf{x}(\xi, 0) + \eta\mathbf{x}(\xi, 1) \\ & - [1 - \xi \quad \xi] \begin{bmatrix} \mathbf{x}(0, 0) & \mathbf{x}(0, 1) \\ \mathbf{x}(1, 0) & \mathbf{x}(1, 1) \end{bmatrix} \begin{bmatrix} 1 - \eta \\ \eta \end{bmatrix} \end{aligned}$$

Doing so, we obtain the following positions for the virtual nodes of the cell. The missing edge nodes are placed in the midpoint of their corresponding edges, i.e., following the notation of the Fig. 4, the virtual node of the cell edge $(\mathbf{x}_k^v, \mathbf{x}_{k+1}^v)$ is defined as

$$\mathbf{x}_k^e = \frac{1}{2}(\mathbf{x}_k^v + \mathbf{x}_{k+1}^v), \quad k \in \{1, 2, 3, 4\}. \quad (10)$$

Then, to place the interior center node of the cell we apply $\mathbf{x}(1/2, 1/2)$ and obtain the following expression for the center node in terms of the boundary nodes of the cell

$$\mathbf{x}_0 = \frac{1}{2}(\mathbf{x}_1^e + \mathbf{x}_2^e + \mathbf{x}_3^e + \mathbf{x}_4^e) - \frac{1}{4}(\mathbf{x}_1^v + \mathbf{x}_2^v + \mathbf{x}_3^v + \mathbf{x}_4^v). \quad (11)$$

Note that virtual nodes do not introduce additional degrees of freedom for the mesh, since their position depends on the existing mesh nodes, in particular on the free node, too. Thus, the virtual nodes connected to it update their position when the free node is moved during the optimization process, that is, virtual nodes depend on coordinates (x, y) of the free node, see Fig. 5.

4.3.2. 3D case

The generalization of these ideas to a 3D T-mesh optimization is straightforward. To obtain a global shape distortion measure for each cell we define a tri-quadratic mapping from the master element $\widehat{\Omega}_e = [-1, 1]^3$ to the physical one, which requires 3^3 mesh nodes. A cell is complemented with virtual nodes calculated in terms of the existing nodes of the cell. First, virtual nodes are assigned to each face of the cell in order to define biquadratic mapping for the faces. For that we apply exactly the same procedure developed for the 2D case using Coons interpolation of the four edges of the face. Some faces may contain all the necessary 9 nodes, and therefore there is no need to add virtual nodes for them. If the face misses some nodes, then its virtual mid-side nodes and the face center node are assigned using the equations (10) and (11), respectively. When all the

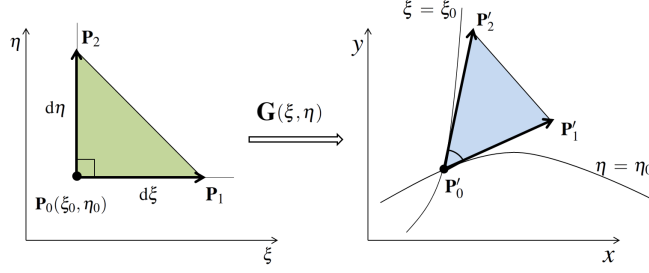


Figure 7: Mean ratio Jacobian as a pointwise quality metric of the parametric mapping \mathbf{G} . Mean ratio Jacobian at any point \mathbf{P}_0 represents the shape quality of the infinitesimal triangle $\mathbf{P}'_0\mathbf{P}'_1\mathbf{P}'_2$.

six faces have their biquadratic parametric representation, we define a 3D Coons body [35] interpolating the six faces in order to define the position of the central virtual node of the cell, see Fig. 6. The interior center node of the cell is assigned as

$$\mathbf{x}_0 = \frac{1}{8}(\mathbf{x}_1^v + \mathbf{x}_2^v + \cdots + \mathbf{x}_8^v) - \frac{1}{4}(\mathbf{x}_1^e + \mathbf{x}_2^e + \cdots + \mathbf{x}_{12}^e) + \frac{1}{2}(\mathbf{x}_1^f + \mathbf{x}_2^f + \cdots + \mathbf{x}_6^f),$$

where \mathbf{x}_i^v , $i = 1, \dots, 8$, are the corner-vertex nodes; \mathbf{x}_i^e , $i = 1, \dots, 12$, are the mid-side edge nodes, and \mathbf{x}_i^f , $i = 1, \dots, 6$, are the face center nodes.

Then, a triquadratic mapping from the master element $\widehat{\Omega}_e$ into the physical one Ω_e given by its nodes \mathbf{x}_i , $i = 1, \dots, 27$, is defined as

$$\mathbf{G}(\boldsymbol{\xi}) = (x(\boldsymbol{\xi}, \eta, \zeta), y(\boldsymbol{\xi}, \eta, \zeta), z(\boldsymbol{\xi}, \eta, \zeta)) = \sum_{i=1}^{27} \mathbf{x}_i N_i(\boldsymbol{\xi}), \quad (12)$$

where N_i are the triquadratic Lagrangian shape functions defined on the master element.

Remark 2. Note that the distortion measure for a triangle defined in terms of its constant Jacobian matrix is a particular case of global distortion measure based on the L^1 norm of the pointwise distortion defined for a given arbitrary element. Also, the pointwise shape distortion measure of any mapping \mathbf{G} is related to the shape distortion measure for a triangle as follows: the value of the distortion measure at any point \mathbf{P}_0 of the master element is the distortion measure of the infinitesimal triangle in physical domain formed by two isoparametric curves passing through the point $\mathbf{P}'_0 = \mathbf{G}(\mathbf{P}_0)$, as illustrated in Fig. 7.

Remark 3. *Another way to evaluate the distortion of a non-simplicial element is to decompose it into several overlapping simplicial elements and define the distortion as a sum of the distortions of each triangle of this decomposition. We have used this approach in our previous work about the parameterization method based on T-mesh optimization algorithm [36]. However, this approach has some shortcomings due to the large variety of possible simplicial decompositions, especially in 3D case.*

4.4. Weighted objective function

The objective function given in section 4.2 is a sum of the terms corresponding to the global distortion measure of each cell. A conformal uniform mesh is a particular case of our quadtree T-mesh. In this case, a free node is surrounded by the cells of the same scale, in the sense that their counterparts in parametric domain have the same size. For this situation it is reasonable to compose the objective function by the sum of global distortion measures. However, for a general quadtree T-mesh, a free node can be surrounded by cells of different scales. In computational experiments we have observed that for this case, a weighted objective function produces better results. Namely, we can define the objective function as follows

$$K^*(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^N w_i \eta_{\Omega_i}^*(\mathbf{x}),$$

where the weights w_i have some relation with the scale of the corresponding cell, that is, the cells of the same scale will have the same weight. For our balanced quadtree/octree meshes a free node can be surrounded by the cells of at most two different levels. So we need two different weights for our terms. Putting the smallest weight to 1 for the cell of level k , we need one value for the weight of the cells of level $k - 1$. Assigning different values to this weight we can obtain slightly different physical T-meshes. In practice, we have observed that good parametrizations are obtained using the weight $w_i = 2$ for the biggest cells of the local mesh. The weighted objective function tends to obtain a better orthogonality of the parametric curves in some cases.

5. Construction of a spline representation of the geometry

In this section we construct a spline representation of the geometry. For simplicity, the process is described for 2D, being the generalization to 3D obvious.

We have to obtain a global one-to-one spline transformation $\mathbf{S} : \widehat{\Omega} = [0, 1]^2 \rightarrow \Omega$ that maps the parametric domain into the physical one. For this purpose, we

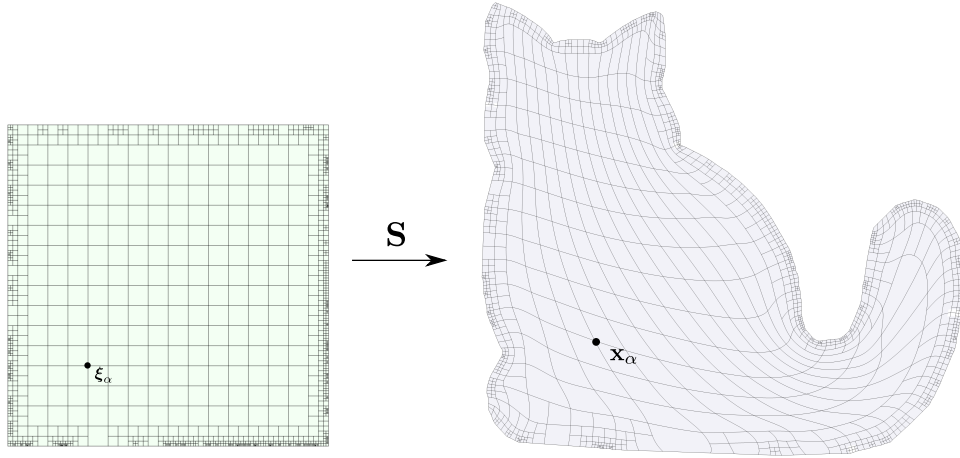


Figure 8: *Cat* test model. Parametric domain and spline representation of the physical domain.

use polynomial spline spaces constructed via the strategy proposed in our previous work [17, 37]. This strategy allows to define easily cubic spline spaces with nice properties over a given strongly balanced quadtree/octree T-mesh. We are going to refer to these spline functions as Extended Polynomial splines (EP-splines). According to the strategy, each regular node of the T-mesh has one blending function assigned. Similarly to T-splines [38], local knot vectors for each function are inferred by walking across the mesh and traversing its edges. Then, in order to span spaces with nice properties, some of the inferred function supports have to be modified applying certain modification rules.

5.1. Interpolation

Spline representation of the physical domain is build as a linear combination of EP-spline blending functions defined over the adapted parametric T-mesh T

$$\mathbf{S}(\boldsymbol{\xi}) = \sum_{\alpha \in A_T} \mathbf{P}_\alpha \hat{N}_\alpha(\boldsymbol{\xi}),$$

where $\mathbf{P}_\alpha \in \mathbb{R}^2$ is the control point corresponding to the α -th blending function.

Control points \mathbf{P}_α are found by imposing interpolation conditions. A standard choice for an interpolation with B-splines is Greville abscissae, which is the averages of the knots. These points correspond approximately to the points where the B-spline functions attain their maximum. The generalization of this

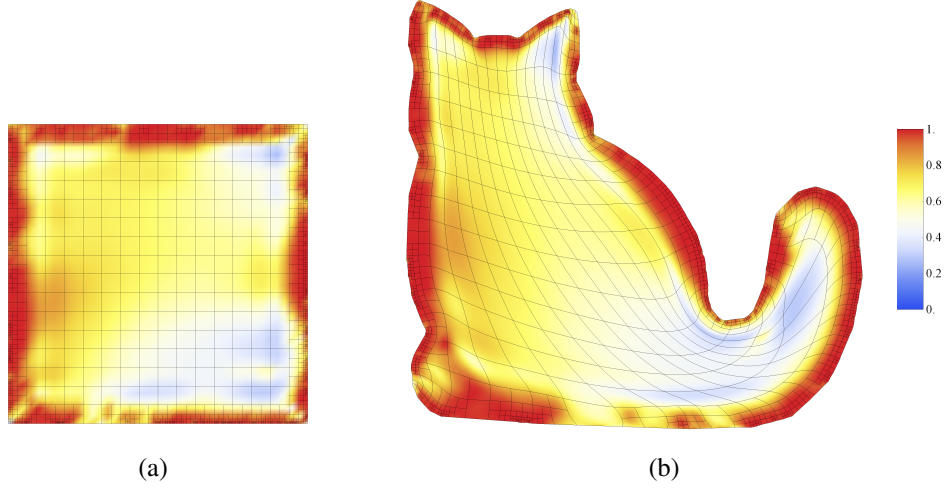


Figure 9: *Cat* test model. Colormap of mean ratio Jacobian. (a) Parametric domain. (b) Physical domain.

concept for the T-mesh leads to the so-called Greville collocation points. Dealing with a T-mesh, we do not have a global knot vector, so a set of Greville points is obtained as the knot average of each local knot vector, that is, for the local knot vectors $\Xi_\alpha = \{\xi_1, \xi_2, \xi_3, \xi_4, \xi_5\}$ and $\mathcal{H}_\alpha = \{\eta_1, \eta_2, \eta_3, \eta_4, \eta_5\}$ of the function \widehat{N}_α , its Greville point is defined as $\xi_\alpha = (\xi_\alpha, \eta_\alpha)$, where

$$\xi_\alpha = \frac{\xi_2 + \xi_3 + \xi_4}{3} \quad \text{and} \quad \eta_\alpha = \frac{\eta_2 + \eta_3 + \eta_4}{3}.$$

In the case of our EP-spline functions, Greville points coincide with their anchors and thus with the vertices of the T-mesh, except for the functions that contain three repeated boundary knots. For each point ξ_β in the parametric space its image \mathbf{x}_β in the physical space is determined by the piece-wise mapping Π obtained by the mesh optimization process developed in section 4.

Finally, the control points are found from the linear system of equations

$$\mathbf{x}_\beta = \mathbf{S}(\xi_\beta) = \Pi(\xi_\beta) = \sum_{\alpha \in A_T} \mathbf{P}_\alpha \widehat{N}_\alpha(\xi_\beta), \quad \forall \xi_\beta, \beta \in A_T.$$

We remark that the construction of the spline representation can also be performed with any other spline basis functions defined over a T-mesh.

6. Quality assessment and its improvement

6.1. Quality assessment of the mapping. Mean ratio Jacobian

To assess the pointwise quality of the obtained parameterization we appeal to the same idea used in the optimization procedure. Namely, our objective function was based on the pointwise quality measure of the local quadratic mapping for each cell. Now we can analyze the pointwise quality measure of the constructed spline transformation. That is, we calculate at each point of the domain the mean ratio Jacobian given by

$$q_S(\boldsymbol{\xi}) = \frac{d \det(J_S)^{2/d}}{\|J_S\|^2},$$

where J_S is the Jacobian matrix of the mapping \mathbf{S} at the point $\boldsymbol{\xi}$ of the parametric domain. In contrast to the scaled Jacobian, that represents the quality of the mapping \mathbf{S} in the sense of the orthogonality of its isoparametric curves, the mean ratio Jacobian represents both: a quality in the sense of the orthogonality and uniformity. The mean ratio Jacobian equals 1 at the point \mathbf{P}_0 if the mapping conserves orthogonality and produces the same length distortion in both parametric directions, i.e. the mapping is conformal at this point.

Figure 9 shows the resulting spline representation of the *Cat* test model and the colormap of the mean ratio Jacobian, whose minimal value is 0.26. Other examples of spline parametrization of a 2D domain are shown in Fig. 10 and Fig. 11. The minimal value of the mean ratio Jacobian for the *Rabbit* geometry is 0.17, and it is 0.03 for the *Map* geometry.

Figures 12 and 13 show 3D parameterization examples with their corresponding colormap of mean ratio Jacobian, whose minimal value is 0.34 and 0.032, respectively.

6.2. Adaptive refinement

The parameterization of complex geometries entails a severe distortion that can lead to low quality mesh cells, or even with negative Jacobian. In some cases, the optimization procedure cannot obtain a good-quality mesh due to the lack of degrees of freedom provided by the inner nodes; in other words, the refinement level of the mesh is not sufficient to parameterize the given geometry. In order to improve the mesh quality in this case, we have to increase the number of degrees of freedom in the area. We adopt the following adaptive strategy: we refine the cells with low quality and optimize again the refined mesh. More precisely, for each cell of the mesh, the mean ratio Jacobian is calculated at some sample points

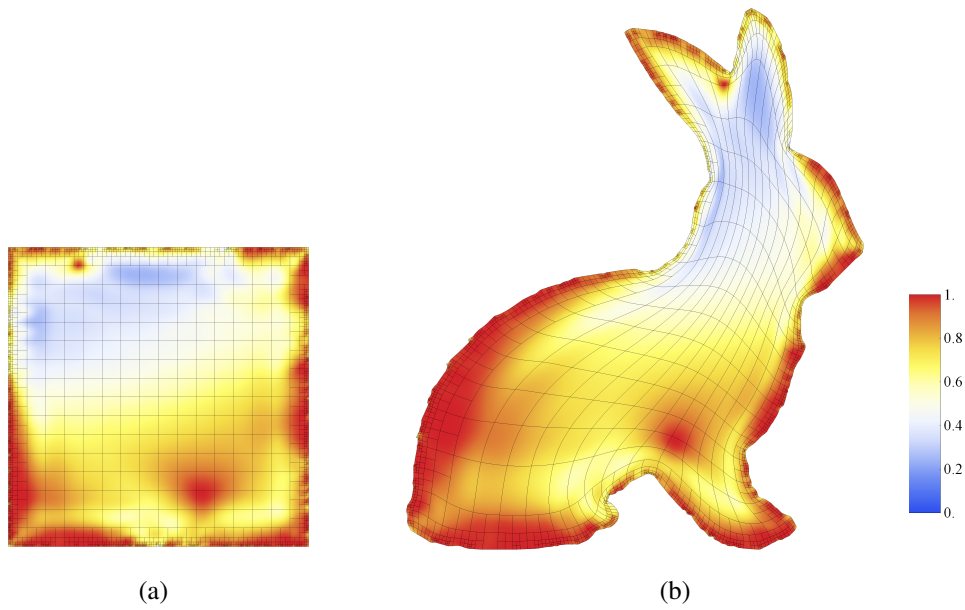


Figure 10: *Rabbit* geometry. Colormap of mean ratio Jacobian. (a) Parametric domain. (b) Physical domain.

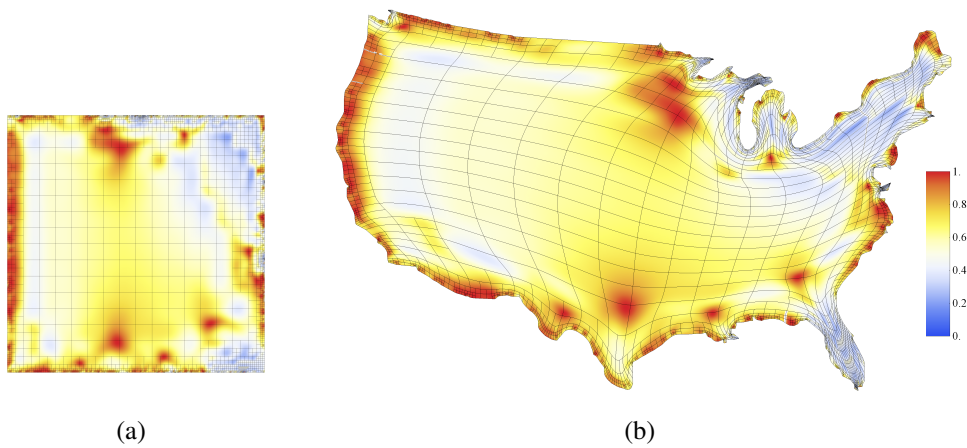


Figure 11: *Map* geometry. Colormap of mean ratio Jacobian. (a) Parametric domain. (b) Physical domain.

(normally quadrature points). A cell $\widehat{\Omega}_e$ is marked to refine if, at least, one of its

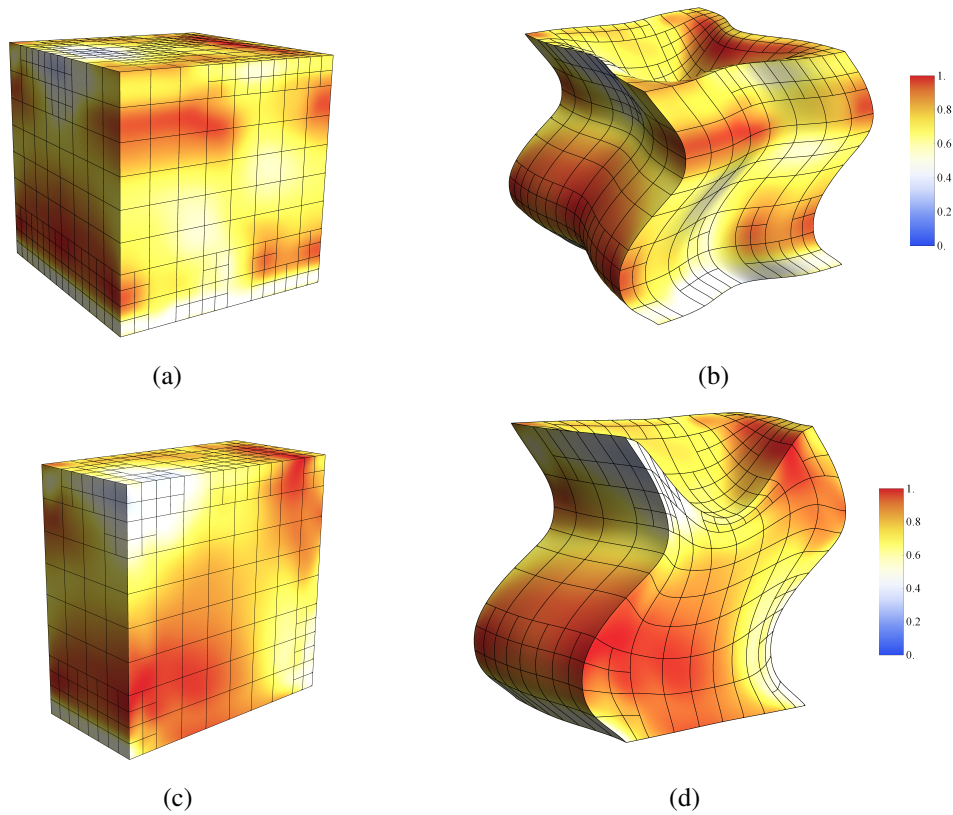


Figure 12: Test model 1. Colormap of mean ratio Jacobian. (a) Parametric domain; (b) spline representation of the physical domain; (c) a section of the parametric domain; (d) corresponding section in the physical domain.

sample points has mean ratio Jacobian less than a certain threshold. The refined T-mesh is optimized again, and the process is repeated until a satisfactory mesh quality is obtained.

7. Isogeometric analysis applications

7.1. Helmholtz equation with variable frequency

Here we present an example of solving a Helmholtz equation in the *Cat* domain using isogeometric analysis with EP-splines.

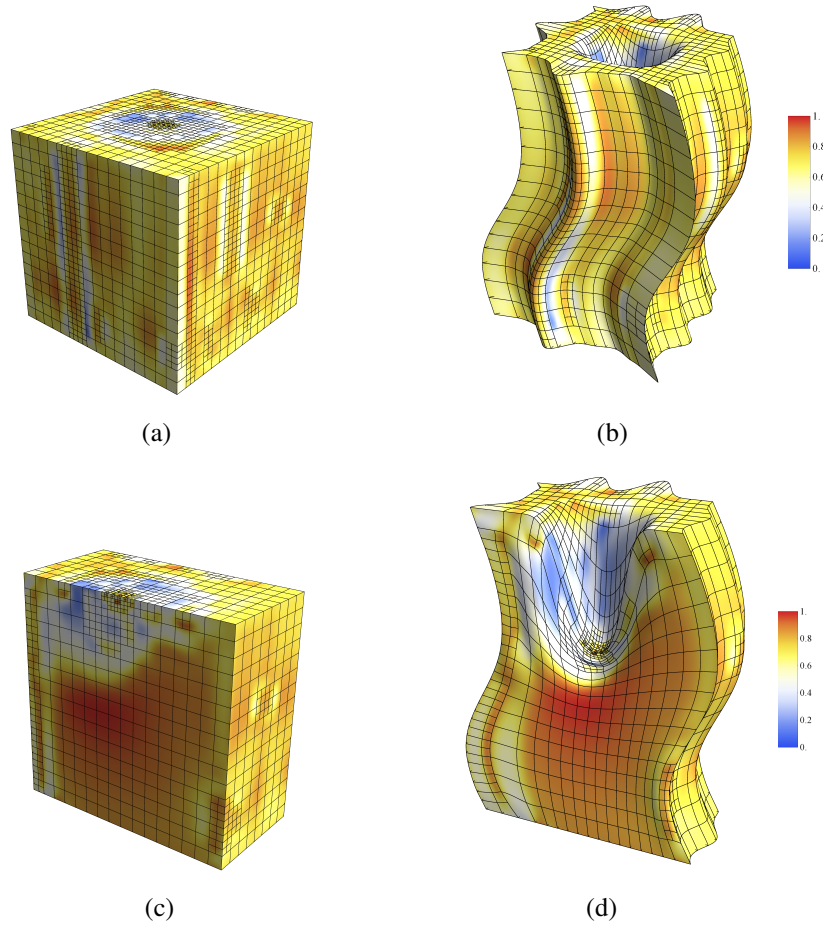


Figure 13: Test model 2. Colormap of mean ratio Jacobian. (a) Parametric domain; (b) spline representation of the physical domain; (c) a section of the parametric domain; (d) corresponding section in the physical domain.

We have the following problem with Dirichlet boundary conditions

$$\begin{aligned}
 -\Delta u - \frac{1}{(\alpha + r)^4} u &= f & \text{in } \Omega, \\
 u &= g & \text{on } \partial\Omega,
 \end{aligned} \tag{13}$$

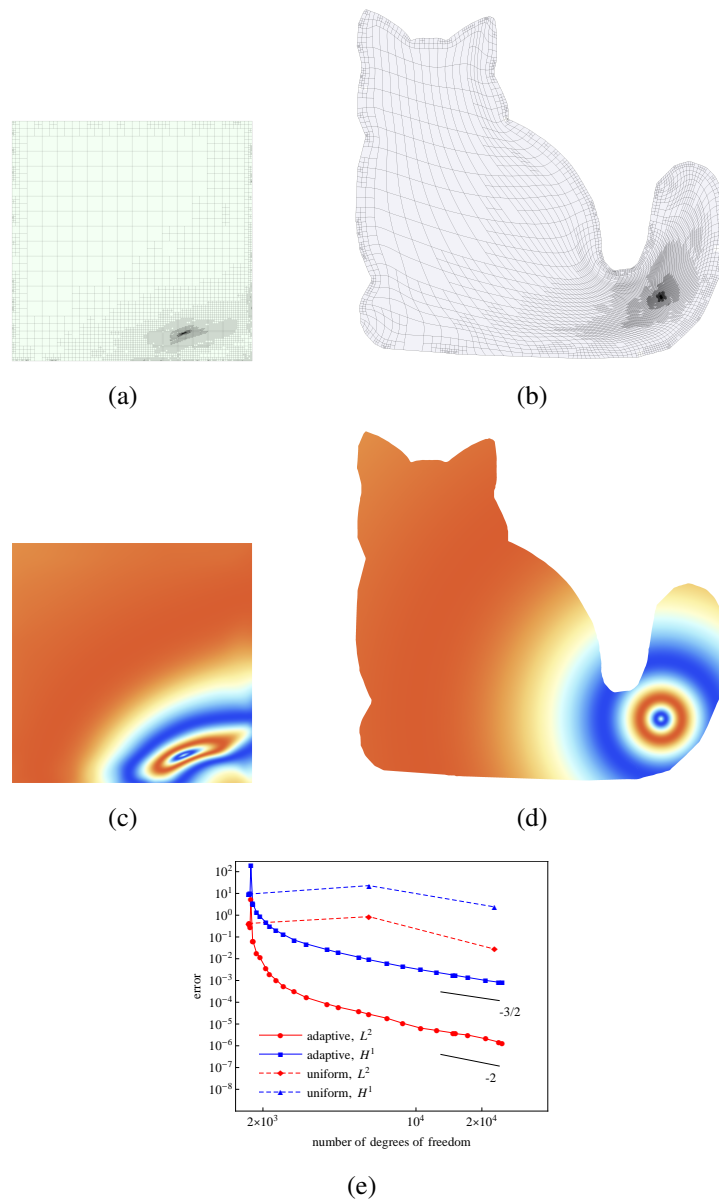


Figure 14: Results of the adaptive IGA with EP-splines for the Helmholtz equation in the *Cat* geometry. (a) Final refinement in the parametric domain. (b) Final refinement in the physical domain. (c) Numerical solution in the parametric domain. (d) Numerical solution in the physical domain. (e) Error convergence for the uniform and adaptive refinements.

where $r = \sqrt{(x-x_0)^2 + (y-y_0)^2}$. The exact solution of the problem is given by

$$u(r) = \sin\left(\frac{1}{\alpha+r}\right). \quad (14)$$

The function (14) is highly oscillatory near the point (x_0, y_0) , and the number of oscillations is determined by the parameter α . We have taken $\alpha = 1/(4\pi)$. Besides, it has a discontinuous gradient at the point (x_0, y_0) .

Let T be a T-mesh of the parametric domain $\widehat{\Omega}$, and $\widehat{V}_T = \text{span}\{\widehat{N}_i\}_{i \in I}$ is the finite dimensional space spanned by the spline blending functions defined over the mesh T . We construct the parametric mapping $\mathbf{S} : \widehat{\Omega} \rightarrow \Omega$ as $\mathbf{S} = \sum_{i \in I} \mathbf{P}_i \widehat{N}_i$. Then, the discrete approximation space V_T in the physical domain Ω is defined as follows: $V_T = \text{span}\{N_i : N_i = \widehat{N}_i \circ \mathbf{S}^{-1}, \text{ for all } \widehat{N}_i, i \in I\}$. The test function space is denoted by $V_{0,T} = \text{span}\{N_i\}_{i \in I_0}$. $V_{g_h, T}(\Omega)$ is the subspace of functions of V_T that are equal to g_h at the boundary, where g_h is an interpolant of g .

The discrete variational formulation consists in finding $u_h \in V_{g_h, T}$ such that

$$a(u_h, N_j) = F(N_j) \quad \forall N_j \in V_{0,T}.$$

where

$$a(u, v) = \int_{\Omega} (\nabla u \cdot \nabla v + k(r)^2 uv) \, d\Omega \quad \text{and} \quad F(v) = \int_{\Omega} f v \, d\Omega,$$

being $k(r) = \frac{1}{(\alpha+r)^2}$.

The adaptive refinement is performed using the residual-based error estimator given by

$$\eta(\Omega_e)^2 = h^2 \| (f + \Delta u_h + k(r)^2 u_h) \|_{L^2(\Omega_e)}^2,$$

where h is the diameter of the cell Ω_e . We have used the following marking strategy for the adaptive process. A cell Ω_e is marked to be refined if $\eta(\Omega_e) > \gamma \max_i \{\eta(\Omega_i)\}$, being $\gamma \in [0, 1]$. Normally we use $\gamma = 0.5$.

The numerical solution of the problem and the mesh corresponding to the final refinement iteration are shown in Fig. 14(a)-(d). As expected, during adaptive refinement the error estimator has marked the singularity zone. The evolution of the exact error in L^2 -norm and H^1 -seminorm and its comparison with the uniform refinement are shown in Fig. 14(e). As it can be observed, the adaptive refinement gains the optimal rates of convergence.

7.2. 3D Poisson problem

Now we consider a 3D Poisson problem with Dirichlet boundary conditions in Test model 1

$$\begin{aligned} -\Delta u &= f && \text{in } \Omega, \\ u &= g && \text{on } \partial\Omega. \end{aligned} \tag{15}$$

The problem is set up so that the exact solution is a function with a steep wave front given by

$$u(r) = \arctan(\alpha(r - r_0)),$$

where $r = \sqrt{(x - x_c)^2 + (y - y_c)^2 + (z - z_c)^2}$, the parameter α determines the steepness of the wave front and r_0 is its location. In this example, $\alpha = 200$ and $r_0 = 0.6$. The center of the wave front (x_c, y_c, z_c) is situated outside our computational domain, so the function is smooth in Ω . The adaptive refinement with EP-splines is performed using residual-based error estimator. The numerical solution of the problem and the mesh corresponding to the final refinement iteration are shown in Fig. 15. As expected, the error estimator has marked for refinement the zone of the wave front. The evolution of the exact error in L^2 -norm and H^1 -seminorm and its comparison with uniform refinement are shown in Fig. 15(c).

7.3. Wind field adjustment problem

The problem of wind field adjustment consists in finding a velocity field \vec{u} over a certain complex terrain that offers the best adjustment to some initial field \vec{u}_0 obtained from experimental measures. We use a mass consistent model based on the continuity equation of an incompressible flow with an impermeability boundary condition on the terrain surface Γ_b

$$\nabla \cdot \vec{u} = 0 \quad \text{in } \Omega \tag{16}$$

$$\vec{n} \cdot \vec{u} = 0 \quad \text{on } \Gamma_b \tag{17}$$

To find a field $\vec{u} = (u, v, w)$, adjusted to the given initial field $\vec{u}_0 = (u_0, v_0, w_0)$, we formulate a least-square problem in the domain Ω with the functional for a field $\vec{v} = (\tilde{u}, \tilde{v}, \tilde{w})$

$$E(\vec{v}) = \int_{\Omega} \left[(\tilde{u} - u_0)^2 + (\tilde{v} - v_0)^2 + (\tilde{w} - w_0)^2 \right] d\Omega. \tag{18}$$

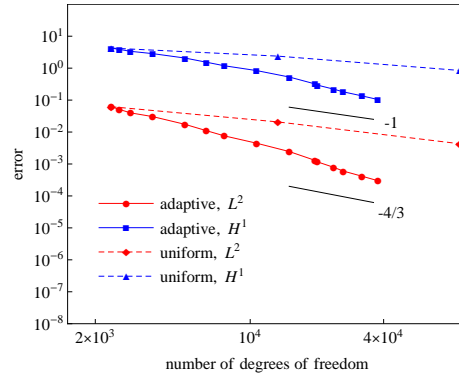
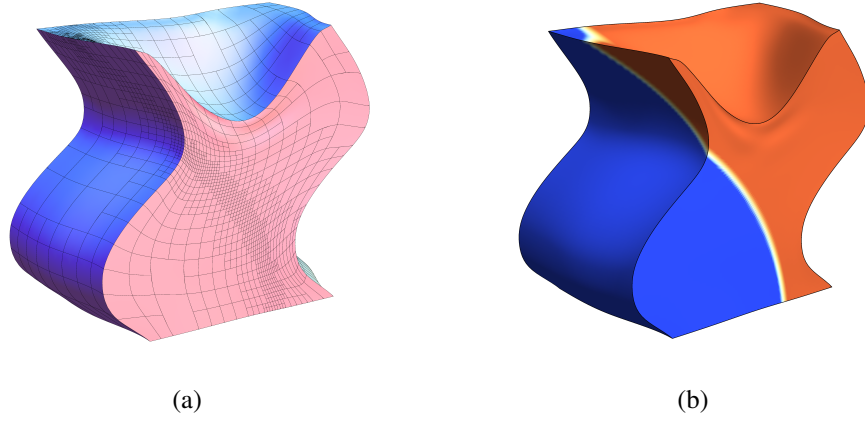


Figure 15: Results of the IGA adaptive refinement with EP-splines for 3D Poisson problem. (b) Final refinement in the physical domain. (d) Numerical solution in the physical domain. (e) Error convergence for the uniform and adaptive refinements.

Then the velocities field \vec{u} is the solution of the following problem: Find $\vec{u} \in \mathbb{K}$ such that

$$E(\vec{u}) = \min_{\vec{v} \in \mathbb{K}} E(\vec{v}) \quad (19)$$

$$\mathbb{K} = \{\vec{v} : \nabla \cdot \vec{v} = 0, \vec{n} \cdot \vec{v} = 0 \text{ on } \Gamma_b\}.$$

This problem is equivalent to find a saddle point (\vec{u}, λ) of the Lagrangian

$$L(\vec{v}, \lambda) = E(\vec{v}) + \int_{\Omega} \lambda \nabla \cdot \vec{v} d\Omega, \quad (20)$$

being $\lambda(x, y, z)$ the Lagrange multiplier.

Then the resulting wind field \vec{u} verifies the Euler-Lagrange equations

$$\vec{u} = \vec{u}_0 + \nabla\lambda. \quad (21)$$

Substituting (21) into (16) and (17), the variational approach results in an elliptic problem for λ

$$\begin{aligned} -\Delta\lambda &= \nabla \cdot \vec{u}_0 \quad \text{in } \Omega \\ -\vec{n} \cdot (\nabla\lambda) &= \vec{n} \cdot \vec{u}_0 \quad \text{on } \Gamma_b \\ \lambda &= 0 \quad \text{on } \Gamma_a \end{aligned} \quad (22)$$

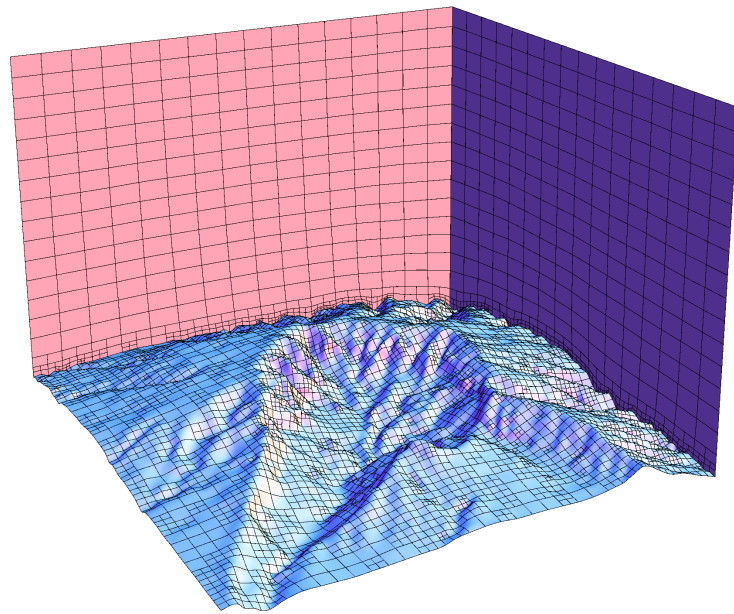
For more details about wind field adjustment model see [39, 40].

Numerical solution for the function λ is found via isogeometric analysis. Then, we obtain the wind field \vec{u} from equation (21). Here we solve the problem over the orography of the *La Palma* island with a constant initial wind velocity $\vec{u}_0 = (10, 0, 0)$ m/s. To solve the elliptic problem (22) we need to construct a domain parameterization adapted to the singularities of the terrain. Figure 16(a) shows the interior of the computational domain, which includes a 10 km air layer above the terrain surface. Streamlines of the adjusted wind field \vec{u} are shown in Fig. 16(b). The black arrow indicates the direction of the initial wind velocity \vec{u}_0 . It can be observed that the red streamlines started from the points close to the surface tend to follow the terrain orography. The blue streamlines start at the points with the same x, y coordinates but higher z coordinate, and it can be appreciated that they are less influenced by the orography.

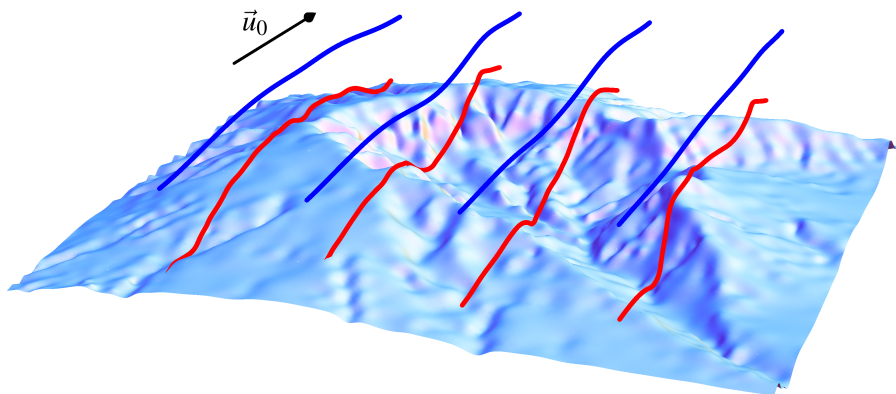
7.4. Application of the optimization procedure to high-order quadrilateral and hexahedral meshes

It is worth noting that the T-mesh optimization algorithm described in this work can also be used for FEM quadrilateral and hexahedral mesh optimization, in particular, second order meshes. The difference with respect to a T-mesh optimization is that we do not need to complement some elements with virtual nodes since all 9 (27 for 3D) nodes are given for the second order elements, and all of them should be relocated in order to improve the quality of the quadratic mapping of each cell.

Frequently, for example in shape optimization problems, it is necessary to change the boundary of the computational domain and solve the problem again and again over the changed domain. Remeshing the geometry at each step can



(a)



(b)

Figure 16: Wind field adjustment problem. (a) The interior of the computational domain (a 10 km air layer above the terrain surface). (b) Streamlines of the adjusted field \vec{u} . Red streamlines start from the points situated 300 m above terrain surface; blue streamlines start from the points with the same x, y coordinates and at height of 1500 m.

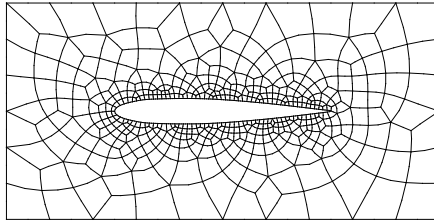
be very costly and can obstruct the automation of the process. Besides, a new mesh generation can introduce a different number of degrees of freedom for the problem. So, it is desirable to be able to adapt the initial mesh to the new boundary without changing its topology.

Here, we show the effectiveness of the optimization algorithm in the following scenario. Figure 17(a) shows an unstructured second order quadrilateral mesh generated by the *Gmsh* mesh generator [41]. The computational domain represents a simplified model of the airfoil where its surroundings fluid is meshed for applying FEM. The mesh contains 470 elements. Then we change the boundary of the domain by rotating the airfoil. New positions of the boundary nodes produce a tangled mesh, see Fig. 17(b). Our optimization algorithm is applied to relocate the interior nodes. The resulting optimized mesh adapted to the new boundary is shown in Fig. 17(c). Note that this mesh has the same topology as the initial one. Thus, the problem can be solved again over the new domain without remeshing the geometry. Figure 17(d) shows the comparison of the qualities of the meshes after optimization and after Laplacian previous relocation, which leaves 213 invalid cells.

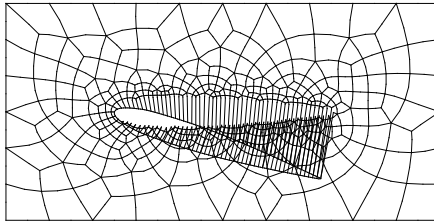
A similar example was performed for a hexahedral mesh of a wing. A 2D quadrilateral mesh was extruded with 25 layers forming a hexahedral mesh with 11750 elements, see Fig. 18(a). Then, one end of the wing was rotated $\pi/10$ degrees. The nodes over the internal surface are progressively rotated, obtaining a 3D twisted wing. These rotations produce a tangled quadrilateral mesh of the boundary surface and a tangled hexahedral elements in the interior, see Fig. 18(b). First, the boundary quadrilateral elements are optimized, and then, the hexahedral mesh is optimized, see Fig. 18(c). Figure 18(d) compares the elements quality after optimization and after Laplacian previous relocation, which produced 5136 invalid elements.

8. Conclusions and challenges

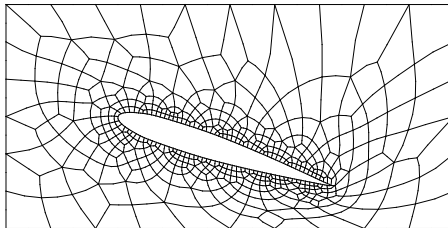
We have presented an affective tool for obtaining a single patch spline parameterization for 2D and 3D geometries. A new T-mesh untangling and smoothing procedure has been applied in order to define an isomorphic transformation between parametric and physical T-meshes. This transformation provides the data to perform interpolation and construct a spline representation of the object. The presented technique is simple and easy to implement. The algorithm has been tested in several 2D and 3D geometries and, for all of them, we have obtained a high quality parametric transformation with strictly positive Jacobian. Also,



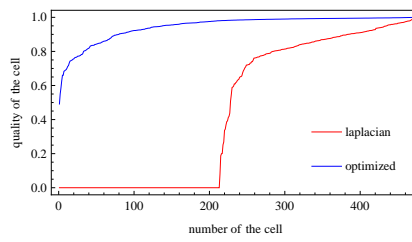
(a)



(b)

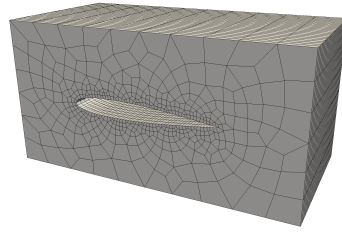


(c)

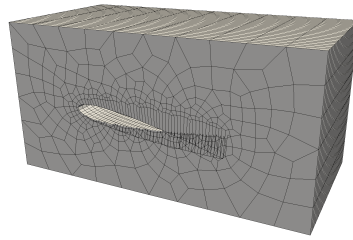


(d)

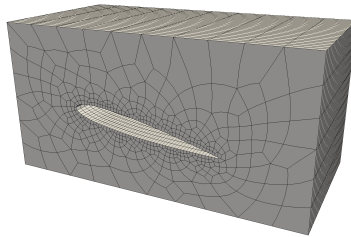
Figure 17: Unstructured quadrilateral mesh optimization. (a) Initial mesh. (b) Tangled mesh after changing the contour of the geometry. (c) Optimized mesh. (d) Quality of the mesh before and after optimization. A sorted list of qualities measures $q_{\Omega_e}^*$ of all elements of the mesh.



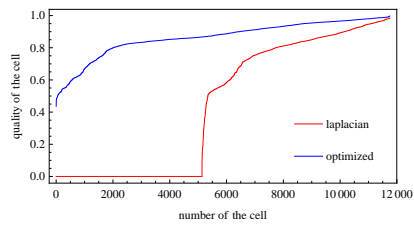
(a)



(b)



(c)



(d)

Figure 18: Unstructured hexahedral mesh optimization. (a) Initial mesh. (b) Tangled mesh after changing the contour of the wing. (c) Optimized mesh. (d) Quality of the mesh before and after optimization. A sorted list of qualities measures $q_{\Omega_e}^*$ of all elements of the mesh.

we have presented examples of solving partial differential equations using isogeometric analysis with EP-splines in domains parameterized with our technique. In our future work we plan to develop a more direct parameterization strategy based on the optimization of the control points of the spline mapping instead of its piecewise approximation. Also, in some cases, in order to obtain a better quality volume parameterization, it may be desirable to include the boundary nodes in the optimization process allowing their movement over the input surface. On the other hand, as was mentioned before, there are still considerable limitations (especially in 3D) for tensor product spline parameterization, since it may present singularities in some cases. It is necessary to study the possibility to obtain analysis-suitable parameterization for domains with non-convex merging of the boundary curves or surfaces.

Acknowledgements

This work has been supported by Spanish Government, “Secretaría de Estado de Universidades e Investigación,” “Ministerio de Economía y Competitividad” and FEDER, grant contract: CTM2014-55014-C3-1-R, “Programa de FPU 12/00202 del Ministerio de Educación, Cultura y Deporte”, “Programa de FPI propio de la Universidad de Las Palmas de Gran Canaria”.

- [1] Y. Bazilevs, V. M. Calo, J. A. Cottrell, J. A. Evans, T. J. R. Hughes, S. Lipton, M. A. Scott, T. W. Sederberg, Isogeometric analysis: Toward unification of computer aided design and finite element analysis, in: Trends in Engineering Computational Technology, Saxe-Coburg Publications, Stirling, 2008, pp. 1–16.
- [2] Y. Bazilevs, V. M. Calo, J. A. Cottrell, J. A. Evans, T. J. R. Hughes, S. Lipton, M. A. Scott, T. W. Sederberg, Isogeometric analysis using T-splines, *Comput. Meth. Appl. Mech. Eng.* 199 (2010) 229–263.
- [3] J. A. Cottrell, T. J. R. Hughes, Y. Bazilevs, *Isogeometric analysis: Toward integration of CAD and FEA*, 1st Edition, Wiley Publishing, 2009.
- [4] Y. Zhang, W. Wang, T. J. R. Hughes, Solid T-spline construction from boundary representations for genus-zero geometry, *Comput. Meth. Appl. Mech. Eng.* 249-252 (2012) 185–197.

- [5] W. Wang, Y. Zhang, L. Liu, T. J. R. Hughes, Trivariate solid T-spline construction from boundary triangulations with arbitrary genus topology, *Computer-Aided Design* 45 (2013) 351–360.
- [6] L. Liu, Y. Zhang, T. J. Hughes, M. A. Scott, T. W. Sederberg, Volumetric T-spline Construction Using Boolean Operations, *Eng. with Comput.* 30 (4) (2014) 425–439.
- [7] L. Liu, Y. Zhang, Y. Liu, W. Wang, Feature-preserving t-mesh construction using skeleton-based polycubes, *Computer-Aided Design* 58 (2015) 162 – 172.
- [8] J. Gravesen, A. Evgrafov, D.-M. Nguyen, P. Nørtoft, *Planar Parametrization in Isogeometric Analysis*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2014, pp. 189–212.
- [9] N. D. Manh, A. Evgrafov, A. R. Gersborg, J. Gravesen, Isogeometric shape optimization of vibrating membranes, *Computer Methods in Applied Mechanics and Engineering* 200 (1316) (2011) 1343 – 1353. doi:<http://doi.org/10.1016/j.cma.2010.12.015>.
- [10] G. Xu, B. Mourrain, R. Duvigneau, A. Galligo, Parametrization of computational domain in isogeometric analysis: Methods and comparison, *Comput. Meth. Appl. Mech. Eng.* 200 (2011) 2021–2031.
- [11] G. Xu, B. Mourrain, R. Duvigneau, A. Galligo, Optimal analysis-aware parameterization of computational domain in 3D isogeometric analysis, *Comput. Aided Des.* 45 (4) (2013) 812–821.
- [12] G. Xu, B. Mourrain, R. Duvigneau, A. Galligo, Variational harmonic method for parameterization of computational domain in 2D isogeometric analysis, in: *12th International Conference on Computer-Aided Design and Computer Graphics*, IEEE, Jinan, 2011, pp. 223–228.
- [13] G. Xu, B. Mourrain, R. Duvigneau, A. Galligo, Constructing analysis-suitable parameterization of computational domain from CAD boundary by variational harmonic method, *Journal of Computational Physics* 252 (2013) 275–289.
- [14] X. Li, X. Guo, H. Wang, Y. He, X. Gu, H. Qin, Harmonic volumetric mapping for solid modeling applications, in: *Proc. of ACM Solid and Physical*

Modeling Symposium, Association for Computing Machinery, Inc., 2007, pp. 109–120.

- [15] T. Martin, E. Cohen, R. Kirby, Volumetric parameterization and trivariate B-spline fitting using harmonic functions, *Comput. Aid. Geom. Design* 26 (2009) 648–664.
- [16] A. Falini, J. Špeh, B. Jüttler, Planar domain parameterization with THB-splines, *Comput. Aided Geom. Des.* 35 (C) (2015) 95–108.
- [17] M. Brovka, J. I. López, J. M. Escobar, R. Montenegro, J. M. Cascón, A simple strategy for defining polynomial spline spaces over hierarchical T-meshes, *Computer-Aided Design* 72 (2016) 140–156.
- [18] J. M. Cascón, R. Montenegro, J. M. Escobar, E. Rodríguez, G. Montero, A new *meccano* technique for adaptive 3-D triangulation, in: *Proc. of the 16th International Meshing Roundtable*, Springer, Berlin, 2007, pp. 103–120.
- [19] J. M. Cascón, R. Montenegro, J. M. Escobar, E. Rodríguez, G. Montero, The *meccano* method for automatic tetrahedral mesh generation of complex genus-zero solids, in: *Proc. of the 18th International Meshing Roundtable*, Springer, Berlin, 2009, pp. 463–480.
- [20] J. M. Escobar, E. Rodríguez, R. Montenegro, G. Montero, J. M. González-Yuste, Simultaneous untangling and smoothing of tetrahedral meshes, *Comput. Meth. Appl. Mech. Eng.* 192 (2003) 2775–2787.
- [21] H. Samet, *Foundations of Multidimensional and Metric Data Structures*, Morgan Kaufmann Publishers, Burlington, Massachusetts, 2006.
- [22] M. S. Floater, Parametrization and smooth approximation of surface triangulations, *Comput. Aid. Geom. Design* 14 (1997) 231–250.
- [23] S. A. Coons, *Surfaces for computer aided design*, Springfield, 1964.
- [24] G. Farin, D. Hansford, Discrete Coons patches, *Comput. Aid. Geom. Design* 16 (1999) 691–700.
- [25] P. M. Knupp, Algebraic mesh quality metrics, *SIAM J. Sci. Comput.* 23 (2001) 193–218.

- [26] P. M. Knupp, A method for hexahedral mesh shape optimization, *Int. J. Num. Meth. Eng.* 58 (2) (2003) 319–332.
- [27] R. Montenegro, J. M. Cascón, J. M. Escobar, E. Rodríguez, G. Montero, An automatic strategy for adaptive tetrahedral mesh generation, *Appl. Num. Math.* 59 (2009) 2203–2217.
- [28] J. M. Escobar, R. Montenegro, E. Rodríguez, J. M. Cascón, The meccano method for isogeometric solid modeling and applications, *Engineering with Computers* 30 (3) (2014) 331–343.
- [29] J. M. Escobar, J. M. Cascón, E. Rodríguez, R. Montenegro, A new approach to solid modeling with trivariate T-splines based on mesh optimization, *Comput. Meth. Appl. Mech. Eng.* 200 (2011) 3210–3222.
- [30] J. M. Escobar, E. Rodríguez, R. Montenegro, G. Montero, J. M. González-Yuste, SUS Code: Simultaneous mesh untangling and smoothing code, Technote, <http://www.dca.iusiani.ulpgc.es/SUScode/TechNote.pdf> (2003).
- [31] J. Dennis, R. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Classics in Applied Mathematics, Society for Industrial and Applied Mathematics, Englewood Cliffs, New Jersey, 1983.
URL <http://books.google.es/books?id=RtxcWd0eBD0C>
- [32] A. Gargallo-Peiró, X. Roca, J. Peraire, J. Sarrate, Distortion and quality measures for validating and generating high-order tetrahedral meshes, *Engineering with Computers* 31 (3) (2015) 423–437.
- [33] E. Ruiz-Gironés, X. Roca, J. Sarrate, High-order mesh curving by distortion minimization with boundary nodes free to slide on a 3D CAD representation, *Computer-Aided Design* 72 (2016) 52 – 64.
- [34] A. Gargallo-Peiró, X. Roca, J. Peraire, J. Sarrate, Optimization of a regularized distortion measure to generate curved high-order unstructured tetrahedral meshes, *International Journal for Numerical Methods in Engineering* 103 (5) (2015) 342–363.
- [35] R. Ďurikovič, S. Czanner, Modelling with three types of coons bodies, *International Journal of Modelling and Simulation* 24 (2) (2004) 97–101.

- [36] M. Brovka, J. I. López, J. M. Escobar, J. M. Cascón, R. Montenegro, A new method for T-spline parameterization of complex 2D geometries, *Engineering with Computers* 30 (4) (2014) 457–473.
- [37] M. Brovka, Construction of polynomial spline spaces over T-meshes for its application in Isogeometric Analysis, Ph.D. thesis, Universidad de Las Palmas de Gran Canaria.
- [38] T. W. Sederberg, J. Zheng, A. Bakenov, A. Nasri, T-splines and T-NURCCs, *ACM Trans. Graph.* 22 (2003) 477–484.
- [39] G. Montero, R. Montenegro, J. Escobar, A 3-d diagnostic model for wind field adjustment, *Journal of Wind Engineering and Industrial Aerodynamics* 7476 (1998) 249 – 261.
- [40] L. Ferragut, R. Montenegro, G. Montero, E. Rodriguez, M. Asensio, J. Escobar, Comparison between 2.5-d and 3-d realistic models for wind field adjustment, *Journal of Wind Engineering and Industrial Aerodynamics* 98 (1011) (2010) 548 – 558.
- [41] C. Geuzaine, 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* 79 (2009) 1309 – 1331.