

# Optimization of a regularized distortion measure to generate curved high-order unstructured tetrahedral meshes

A. Gargallo-Peiró<sup>1</sup>, X. Roca<sup>2\*</sup>, J. Peraire<sup>2</sup> and J. Sarrate<sup>1</sup>

<sup>1</sup> *Laboratori de Càlcul Numèric (LaCàN), Departament de Matemàtica Aplicada III, Universitat Politècnica de Catalunya, Jordi Girona 1, E-08034 Barcelona, Spain.*

<sup>2</sup> *Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA.*

## SUMMARY

We present a robust method for generating high-order nodal tetrahedral curved meshes. The approach consists of modifying an initial linear mesh by first, introducing high-order nodes, second, displacing the boundary nodes to ensure that they are on the CAD surface, and third, smoothing and untangling the mesh obtained after the displacement of the boundary nodes to produce a valid curved high-order mesh. The smoothing algorithm is based on the optimization of a regularized measure of the mesh distortion relative to the original linear mesh. This means that whenever possible, the resulting mesh preserves the geometrical features of the initial linear mesh such as shape, stretching and size. We present several examples to illustrate the performance of the proposed algorithm. Furthermore, the examples show that the implementation of the optimization problem is robust and capable of handling situations in which the mesh before optimization contains a large number of invalid elements. We consider cases with polynomial approximations up to degree ten, large deformations of the curved boundaries, concave boundaries, and highly stretched boundary layer elements. The meshes obtained are suitable for high-order finite element analyses. Copyright © 2014 John Wiley & Sons, Ltd.

**KEY WORDS:** unstructured high-order methods; mesh generation; CAD; mesh optimization; mesh quality; curved meshing

## 1. INTRODUCTION

In the last decade, unstructured high-order methods [1, 2, 3, 4, 5] have attracted considerable attention from the computational mechanics community. This attention has been prompted by the ability of these methods to approximate with high-fidelity the solution of partial differential equations on complex domains. It is well known that for problems with smooth solutions, the approximation obtained with high-order methods converges exponentially with the order of the approximating polynomial. More generally, high-order methods have been shown to deliver higher accuracy with a lower computational cost than low-order methods in many practical applications [6, 7, 8, 9, 10, 11, 12, 13, 14, 15]. In addition, the accurate approximation of the domain geometry eliminates the spurious effects in the solution that can arise from a piecewise linear representation of the curved domain boundaries [16, 17, 18, 19, 20]. An implicit assumption in high-order methods is that the geometry is represented with sufficient accuracy to enable high convergence rates to be realized.

Despite the attractive features of high-order methods, their adoption for practical applications has been hampered by technical challenges such as the development of robust implementations and the

---

\*Correspondence to: X. Roca, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA.

generation of suitable 3D curved meshes. In this paper, we focus our attention on the generation of high-order meshes for general geometries that are suitable for high-order finite element analyses. To that end, the generated meshes must satisfy two requirements. On one hand, each tetrahedral element must be the image of a valid straight-sided reference element through an element-wise invertible mapping. On the other hand, the boundary elements must be curved to ensure that the error introduced in the solution by the inexact approximation of the geometry is smaller than the solution discretization error.

High-order meshes are usually generated using *a posteriori* approaches [17, 21, 22, 23, 24, 25] whereby the final mesh is obtained by modifying an initial linear mesh. Starting from an initial linear mesh that approximates the domain geometry, the first step consists of adding high-order nodes, either by enriching the linear tetrahedra uniformly or according to a suitable distribution aimed at minimizing interpolation errors [4, 26]. Next, the boundary nodes are projected onto exact curved boundary. Finally, the mesh is smoothed and untangled to remove the invalid (folded) and low-quality (distorted) elements that might be created during the curving of the mesh boundary. This last step is indeed the most critical to ensure the quality of the generated meshes. When the grid deformation dictated by the projection of the boundary nodes onto the boundary of the domain is small, existing approaches are effective at producing suitable high-order meshes, but they often fail for more extreme cases where the required deformation is larger.

The main goal of this work is to use an *a posteriori* approach to generate 3D curved high-order meshes in which the boundary nodes are on CAD surfaces. To this end, we propose a robust smoothing and untangling method that converts an initial linear mesh to the desired curved high-order mesh. Specifically, the main contribution of this work is the formulation of a minimization problem for a regularized measure of the distortion of a non-linear mapping in which the decision variables are the coordinates of the interior mesh nodes. For each element, we consider the mapping between the initial linear tetrahedron and the curved element in the final mesh. The mapping distortion measure is minimum when the element shape (but not necessarily its size) in the final and initial configurations is preserved. In addition, the mapping distortion measure is regularized so that it is well defined for both valid (unfolded) elements, where the mapping between the linear and curved elements is one-to-one, and invalid (folded), elements where the mapping becomes singular and the determinant of the transformation Jacobian becomes non-positive. For folded elements, the value of the distortion measure is a large positive value which forces the minimization process to untangle the mesh. Once all the elements are untangled, the regularization can be switched off. In that case, the resulting objective function tends to infinity if the Jacobian determinant tends to zero. This form of the objective function prevents untangled (valid) configurations from becoming tangled (invalid) during the optimization process.

The proposed formulation has the following advantages: first, it is capable of transforming an invalid curved high-order mesh to a valid mesh, thanks to the use of the regularized objective function; second, it ensures that initially valid configurations remain valid after transformation; third, it can deal with polynomials of any degree; fourth, it can handle curved boundaries; and fifth, geometrical mesh features such as element shape, anisotropy are preserved.

We present several examples that demonstrate the advantages of the proposed method. Specifically, we show that the proposed approach is able to untangle consistently invalid initial elements for approximations up to degree ten and large deformations of the initial mesh boundaries. Moreover, we show that some features of the initial mesh such as non-uniform element size, and anisotropy (boundary layers) are preserved. Finally, we demonstrate our approach with some examples in which the geometry is defined by a 3D CAD model.

The remainder of the paper is organized as follows. In Section 2, we review previous literature relevant to this work. In Section 3, we present the proposed optimization formulation. In Section 4, we describe the regularization of the distortion measure to untangle invalid meshes. In Section 5, we describe the steps required to generate a curved high-order mesh using the proposed optimization method. Finally, in Section 6, we present several examples to illustrate the advantages and application of the proposed method.

## 2. RELATED WORK

A *posteriori* methods for high-order mesh generation can be divided in three main categories depending on the technique used to curve the mesh and match the domain boundary. The first group of methods [21, 22, 23] curve the mesh boundary, identify mesh entities that produce invalid elements, and eliminate the invalid elements by a combination of local mesh refinements, edge and face swaps, and node relocations. The second group of methods deform an initial linear mesh composed of straight-sided elements using a linear [27, 28, 29] or a nonlinear [25] elasticity analogy. The displacements of the boundary nodes are prescribed to lie on the curved boundary and an elasticity problem is solved using a high-order continuous Galerkin method. The final group of methods [30, 31, 32, 33, 34, 35], solve an optimization problem for the node locations according to an objective function that attempts to guarantee the accurate representation of the domain boundary and the element quality.

The main challenge of a *posteriori* mesh deformation methods is to repair the invalid elements that may arise from displacing the boundary nodes of the initial straight-sided mesh. In some cases, these elements may be excessively distorted, having an adverse effect on the accuracy of the finite element solutions, or may even be invalid in which case the mapping between the master element and the curved deformed element is not invertible. A first requirement to repair invalid elements is an algorithm to detect them and to quantify the level of distortion. For high-order elements, several approaches have been proposed to detect the validity of the mapping [36, 17, 21, 22, 23, 24, 37, 38, 39], and to define suitable quality measures [40, 41, 42, 43, 44, 45, 31, 32, 33, 35].

The mesh repairing can be accomplished by means of topological changes and node relocation. In general, node relocation is necessary, since some interior high-order nodes close to the boundary may end up outside the domain after the boundary nodes are displaced. The node relocation can be formulated as a global problem involving the coordinates of all the interior nodes or as a local problem involving only one or several nodes at a time. The benefit of the global formulation is that the number of iterations to solve the problem is smaller than with a local formulation. However, local formulations tend to be easier to implement and require less computer memory. For linear elements, several local [46, 47, 48] and global [49, 50, 51, 52, 53, 54, 55] approaches have been proposed to improve the mesh quality. For high-order meshes, similar local [32] and global relocation methods [31, 33, 34, 35] have also been developed. It is worth noting that global methods can be implemented locally if a suitable iterative method can be devised. The approach adopted here, borrows from the strategy presented in [55] for surface mesh generation where the node relocation problem is formulated as a global problem but solved using a non-linear Gauss-Seidel iteration.

When the starting mesh in the optimization process is valid, a strategy to prevent elements from becoming invalid is to use an objective function which penalizes the appearance of small Jacobian determinants. This approach has been used in [48, 55, 56] for linear elements and in [25, 32, 34, 33, 35] for high-order elements. A more challenging problem occurs when the starting mesh already contains invalid elements. In such cases, one can either use a log-barrier method [34, 54] or can define a regularized distortion quality measure which is defined for all element configurations and has very large values when the element is invalid. One such regularized distortion measure is presented in [32, 33] for high-order planar and surface elements and used to generate curved meshes. This technique is based on the extension to high-order elements of the definition of quality and distortion measure for linear elements presented in [57, 58].

In this work, we extend the approach proposed in [32, 33] to high-order tetrahedral meshes using the untangling approach proposed in [48, 55]. Further, we propose an expression to evaluate the untangling parameter introduced in [48, 56], eliminating the need for a user-defined input.

## 3. FORMULATION OF THE MESH OPTIMIZATION

In this section, we present a new formulation to solve the mesh deformation problem, which is based on the optimization of a measure of the mesh distortion. To this end, we first review a distortion measure for linear elements. This measure can be interpreted as a point-wise measure of

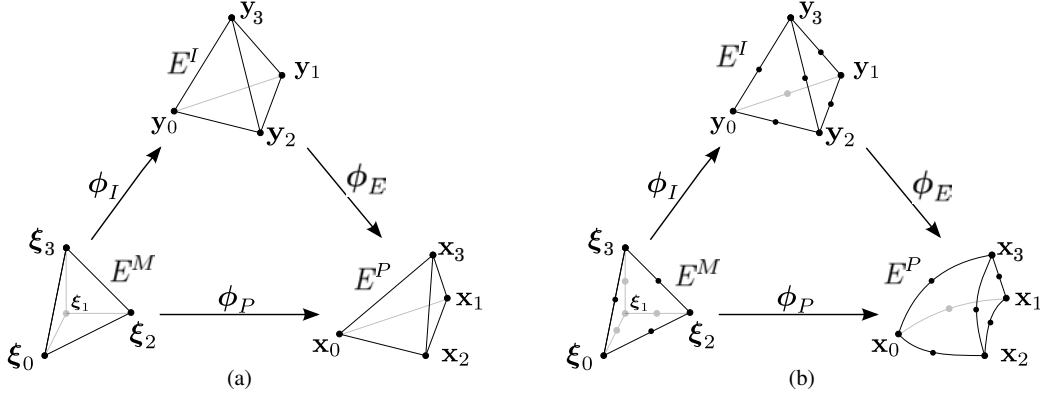


Figure 1. Mappings between the master, initial and physical elements: (a) linear case, and (b) high-order case.

the distortion of a deformation map between an ideal domain and the physical one. The continuous version of the curving problem is stated as finding a deformation map that features an ideal point-wise distortion. Then, we write down the discrete version of the problem where the ideal domain is a straight-sided mesh and the physical domain is a mesh with known curved boundaries. Finally, we pose a global and non-linear optimization problem to find the curved mesh that provides the ideal point-wise distortion in the least-squares sense.

### 3.1. Distortion measures for linear elements

We start by reviewing the distortion measures for linear elements presented in [57, 58]. Consider a linear tetrahedron in the initial configuration,  $E^I$ , having the desired shape and size and the corresponding linear tetrahedron in the physical space,  $E^P$ . To measure the deviation between these two elements, we consider the unique affine mapping,  $\phi_E$ , from  $E^I$  to  $E^P$ , see Figure 1(a). The mapping  $\phi_E$ , can be easily expressed in terms of two additional mapping,  $\phi_P$  and  $\phi_I$ , between the master element  $E^M$  and the initial,  $E^I$ , and physical element,  $E^P$ , respectively. Thus,  $\phi_E$  is determined by the composition

$$\phi_E : E^I \xrightarrow{\phi_I^{-1}} E^M \xrightarrow{\phi_P} E^P.$$

Since  $\phi_E$  is affine, its Jacobian,  $\mathbf{D}\phi_E$ , is constant. The element shape distortion measure [57] is given by,

$$\eta(\mathbf{D}\phi_E) = \frac{\|\mathbf{D}\phi_E\|^2}{d |\sigma|^{2/d}}, \quad (1)$$

where  $d = 3$  is the spatial dimension,  $\|\cdot\|$  is the Frobenius norm, and  $\sigma = \det(\mathbf{D}\phi_E)$ . This distortion measure quantifies the deviation of the shape of the physical tetrahedron  $E^P$  with respect to the shape of the initial tetrahedron,  $E^I$ . We note that this distortion measure is invariant to translation and rotations and equals 1 when the two elements only differ by a scale factor, and tends to  $\infty$  as  $E^P$  becomes degenerate. When working with linear meshes, it is convenient to work with the quality measure

$$q = \frac{1}{\eta}, \quad (2)$$

which takes values in the interval  $[0, 1]$ .

To extend the former distortion (quality) framework to high-order elements, we consider that the physical element,  $E^P$ , is a nodal high-order tetrahedron of degree  $p$  determined by  $n_p$  nodes with coordinates  $\mathbf{x}_i \in \mathbb{R}^3$ , for  $i = 1, \dots, n_p$ . The ideal element,  $E^I$ , corresponds to a nodal high-order

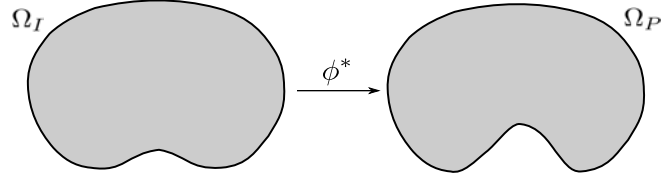


Figure 2. Mapping between the initial and physical domains.

tetrahedron with the desired shape, size and it is characterized by a target node distribution, see Figure 1(b). Similarly to the linear case, we measure the distortion in terms of a unique mapping  $\phi_E$ , from  $E^I$  to  $E^P$ . To this end, the master element,  $E^M$ , is characterized by a node distribution  $\xi_i \in \mathbb{R}^3$ , for  $i = 1, \dots, n_p$ , that provides a quasi-optimal Lebesgue constant [4]. Again, the mapping from each ideal element,  $E^I$ , to the corresponding physical element,  $E^P$ , can be expressed as  $\phi_E = \phi_P \circ \phi_I^{-1}$ . However,  $\phi_E$  is no longer affine, since  $\phi_I$  and  $\phi_P$  are not, in general, affine mappings. Therefore, the Jacobian  $\mathbf{D}\phi_E$  is, in general, also not constant. Note that if we consider the basis  $\{N_i\}_{i=1, \dots, n_p}$  of nodal shape functions (Lagrange interpolation) of degree  $p$  for the given point distribution of the master element, we have that

$$\mathbf{x} = \phi_E(\mathbf{y}) = \sum_{i=1}^{n_p} \mathbf{x}_i N_i(\phi_I^{-1}(\mathbf{y})) \quad (3)$$

and therefore,  $\phi_E$  is explicitly expressed in terms of the element nodal coordinates  $\mathbf{x}_i$ , for  $i = 1, \dots, n_p$ .

### 3.2. Curving: globally defined smooth mapping

Given an initial domain  $\Omega_I \subset \mathbb{R}^d$ , and a physical domain  $\Omega_P \subset \mathbb{R}^d$ , we want to characterize  $\Omega_P$  in terms of  $\Omega_I$ . We assume that the physical domain can be defined as the image of a diffeomorphism  $\phi \in \mathcal{C}^1(\Omega_I, \Omega_P)$ , see Figure 2. To determine the desired diffeomorphism  $\phi$ , we need a distortion measure for non-linear mappings. To this end, given a distortion measure for linear elements  $\eta$ , we define the distortion measure of  $\phi$  at a point  $\mathbf{y} \in \Omega_I$  as,

$$\mathbf{M}\phi(\mathbf{y}) := \eta(\mathbf{D}\phi(\mathbf{y})). \quad (4)$$

We note that the Jacobian  $\mathbf{D}\phi$  at a given point  $\mathbf{y}$  is a linear map and therefore, the distortion measure  $\mathbf{M}$  is well defined.

Given the initial domain  $\Omega_I$  and the boundary of the physical domain,  $\partial\Omega_P$ , the continuous problem is that of finding a diffeomorphism  $\phi^*$  between  $\Omega_I$  and  $\Omega_P$  such that the distortion measure  $\mathbf{M}$  is ideal. That is,

$$\begin{aligned} \mathbf{M}\phi^* &= 1, & \text{in } \Omega_I, \\ \phi^* &= g, & \text{on } \partial\Omega_I, \end{aligned}$$

where  $g$  is the mapping between  $\partial\Omega_I$  and  $\partial\Omega_P$ .

### 3.3. Curving: element-wise defined smooth mapping

We consider that the domain  $\Omega_I$  is approximated by the mesh  $\mathcal{M}_I$  composed by the union of the elements  $E_e^I$ , for  $e = 1, \dots, n_E$ . In addition, we consider that the prescribed curved boundary  $\partial\Omega_P$  is approximated by a surface mesh  $\partial\mathcal{M}_P$  determined by a mapping  $g_h$  from  $\partial\mathcal{M}_I$  to  $\partial\mathcal{M}_P$ . In this setting, we seek an optimal mapping  $\phi_h^*$  from  $\mathcal{M}_I$  to  $\mathcal{M}_P$ , see Figure 3, such that for all  $E^I$  in  $\mathcal{M}_I$ , it has an ideal distortion measure. That is,

$$\mathbf{M}\phi_h^* = 1, \quad \text{in } \mathcal{M}_I, \quad (5)$$

$$\phi_h^* = g_h, \quad \text{on } \partial\mathcal{M}_I. \quad (6)$$

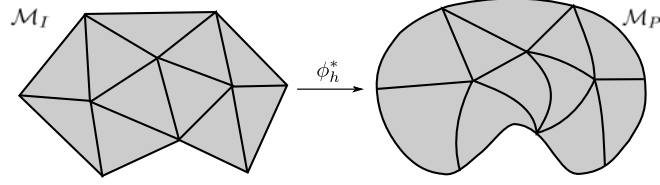


Figure 3. Mapping between the initial and physical meshes.

Since we want to obtain a conformal nodal high-order mesh, we seek for the mapping  $\phi_h^*$ , see Figure 3 in the space of vector functions

$$\mathcal{U} := \{\mathbf{u} \in [C^0(\mathcal{M}_I)]^d \mid \mathbf{u}|_{E^I} \in [\mathcal{P}^p(E^I)]^d, \forall E^I \in \mathcal{M}_I\},$$

where  $\mathcal{P}^p(E^I)$  is the space of polynomials of degree  $p$  on the element  $E^I$ .

For a given mesh  $\mathcal{M}_I$  and a boundary configuration  $g_h$ , a mapping  $\phi_h$  such that Equation (5) and Equation (6) are verified may be, in general, not achievable. To address this issue, we impose the optimality condition in a least-squares sense. That is, we seek  $\phi_h^*$  in  $\mathcal{U}_D$  such that

$$\phi_h^* = \underset{\phi_h \in \mathcal{U}_D}{\operatorname{argmin}} \|\mathbb{M}\phi_h - 1\|_{\mathcal{M}_I}^2, \quad (7)$$

where

$$\mathcal{U}_D := \{\phi_h \in \mathcal{U} \mid (\mathbb{M}\phi_h - 1) \in \mathcal{L}^2(\mathcal{M}_I), \text{ and } \phi_h = g_h \text{ on } \partial\mathcal{M}_I\}.$$

In Equation (7) we define the norms

$$\|f\|_{\mathcal{M}_I} := \sqrt{\langle f, f \rangle_{\mathcal{M}_I}}, \quad (8)$$

$$\|f\|_{E^I} := \sqrt{\langle f, f \rangle_{E^I}}, \quad (9)$$

and the inner product for scalar functions  $f$  and  $g$  in  $\mathcal{M}_I$

$$\langle f, g \rangle_{\mathcal{M}_I} := \sum_{e=1}^{n_E} \langle f|_{E_e^I}, g|_{E_e^I} \rangle_{E_e^I}, \quad (10)$$

$$\langle f, g \rangle_{E^I} := \int_{E^I} f(\mathbf{y}) g(\mathbf{y}) d\mathbf{y}. \quad (11)$$

Once  $\phi_h^*$  is found, each element in the physical domain can be obtained as

$$E_e^P = \phi_h^*(E_e^I),$$

and therefore, the desired physical mesh  $\mathcal{M}_P$  is composed by the union of the elements  $E_e^P$ , for  $e = 1, \dots, n_E$ .

### 3.4. Curving: nodal high-order mesh optimization

The minimization problem stated in Equation (7) can be rewritten in terms of elemental contributions. In particular, according to Equation (10) and Equation (8), we have that

$$\|\mathbb{M}\phi_h - 1\|_{\mathcal{M}_I}^2 = \sum_{e=1}^{n_E} \|\mathbb{M}\phi_{h|_{E_e^P}} - 1\|_{E_e^I}^2 = \sum_{e=1}^{n_E} \|\mathbb{M}\phi_{E_e} - 1\|_{E_e^I}^2,$$

where  $\phi_{h|_{E_e^P}}$  is the mapping  $\phi_{E_e}$  between  $E_e^I$  and its physical element  $E_e^P$ . Therefore, we seek  $\phi_h^*$  in  $\mathcal{U}_D$  such that :

$$\phi_h^* = \underset{\phi_h \in \mathcal{U}_D}{\operatorname{argmin}} \|\mathbb{M}\phi_h - 1\|_{\mathcal{M}_I}^2 = \underset{\phi_h \in \mathcal{U}_D}{\operatorname{argmin}} \sum_{e=1}^{n_E} \|\mathbb{M}\phi_{E_e} - 1\|_{E_e^I}^2. \quad (12)$$

In particular, for nodal high-order elements, and according to Equation (3),  $\phi_{E_e}$  depends on the coordinates of the  $n_p$  element nodes. Thus, the distortion at a point  $\mathbf{y}$  in  $E_e^I$  in Equation (12) can be written as:

$$\mathbf{M}\phi_{E_e}(\mathbf{y}) = \mathbf{M}\phi_{E_e}(\mathbf{y}; \mathbf{x}_{e,1}, \dots, \mathbf{x}_{e,n_p})$$

where the pairs  $(e, j)$  in  $\mathbf{x}_{e,j}$  denotes the local  $j$ -th node of element  $e$ . Thus, for nodal high-order elements, determining  $\phi_h^*$  in Equation (12), is equivalent to determining the coordinates of the nodes of the high-order mesh. Moreover, the element contribution to the objective function only depends on the nodes of that element.

We reorder the coordinates of the nodes,  $\mathbf{x}_i$ , in such a manner that  $i = 1, \dots, n_F$  are the indexes corresponding to the free (interior) nodes, and  $i = n_F + 1, \dots, n_N$  correspond to the fixed nodes (nodes on the CAD surfaces). Note that the coordinates of the fixed nodes are determined by the function  $g_h$ , and can be obtained using the technique presented in [33, 55]. Defining

$$f(\mathbf{x}_1, \dots, \mathbf{x}_{n_F}; \mathbf{x}_{n_F+1}, \dots, \mathbf{x}_{n_N}) := \frac{1}{2} \|\mathbf{M}\phi_h - 1\|_{\mathcal{M}_I}^2, \quad (13)$$

we can formulate the mesh optimization problem as finding  $\{\mathbf{x}_1^*, \dots, \mathbf{x}_{n_F}^*\} \subset \mathbb{R}^3$  such that:

$$\{\mathbf{x}_1^*, \dots, \mathbf{x}_{n_F}^*\} = \underset{\mathbf{x}_1, \dots, \mathbf{x}_{n_F} \in \mathbb{R}^3}{\operatorname{argmin}} f(\mathbf{x}_1, \dots, \mathbf{x}_{n_F}; \mathbf{x}_{n_F+1}, \dots, \mathbf{x}_{n_N}), \quad (14)$$

where  $\mathbf{x}_i = g_h(\mathbf{y}_i)$  for  $i = n_F + 1, \dots, n_N$ . In Appendix A, we detail our approach to solve the global minimization problem stated in Equation (14).

#### 4. MESH UNTANGLING

The minimization problem presented in Equation (14) is based on a Jacobian-based distortion measure presented in Equation (1). This distortion measure becomes infinity when the high-order element is degenerated (when  $\sigma = \det(\mathbf{D}\phi_E) = 0$ ). This feature precludes its use in an untangling procedure. In order to address this issue, we replace the distortion measure in Equation (1) by a regularized form.

##### 4.1. Regularization of the distortion measure

To incorporate the untangling capability to the optimization method, we use the regularization proposed in [48] and replace  $\sigma$  in Equation (1) by

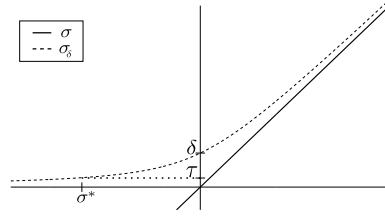
$$\sigma_\delta(\sigma) = \frac{1}{2} \left( \sigma + \sqrt{\sigma^2 + 4\delta^2} \right), \quad (15)$$

where  $\delta$  is a positive element-wise parameter and its selection is discussed in Section 4.2. The regularized Jacobian,  $\sigma_\delta(\sigma)$ , is a monotonically increasing function of  $\sigma$ , such that  $\sigma_\delta(0) = \delta$ , and tends to 0 when  $\sigma$  tends to  $-\infty$ , see Figure 4.

Using the regularized Jacobian, the modified shape distortion measure becomes,

$$\eta_\delta(\mathbf{D}\phi_E) = \frac{\|\mathbf{D}\phi_E\|^2}{3|\sigma_\delta|^{2/3}}. \quad (16)$$

The use of the regularized Jacobian removes the vertical asymptote at  $\sigma = 0$ , and therefore, allows the optimization procedure to recover from the invalid configurations. Moreover, for small values of  $\delta$ , the minimum of  $\eta_\delta$  is close to the minimum of the original shape distortion measure  $\eta$ . We note that the element-wise parameter  $\delta$  is only set to non-zero values when an invalid mesh configuration is considered. Once all the elements are valid,  $\delta$  can be set to zero for all the elements. To analyze the validity of a mesh, we set  $\delta = 0$ , replacing  $\sigma_\delta$  in Equation (16) by  $\sigma_0 = (\sigma + |\sigma|)/2$ . In this way, for invalid meshes,  $\sigma$  becomes negative and,  $\eta_\delta = \infty$  and  $q = 0$ .

Figure 4. Representation of  $\sigma_\delta(\sigma)$ .

#### 4.2. Selection of the regularization parameter

The regularization parameter  $\delta$  needs to satisfy the following criteria. On the one hand,  $\delta$  has to be large enough to ensure that  $\delta^2$  is significant compared to  $\sigma^2$ . On the other hand, it has to be small enough to ensure that the minimum of the modified mesh distortion function is not too far from the original minimum.

In order to simplify the computation of the derivatives of the distortion measure we select a constant value of  $\delta$  for each element. In particular, we determine  $\delta$  taking into account the determinant of the Jacobian of the straight-sided initial element ( $\det \phi_I$ ) and only use the regularized distortion measure when the element is invalid ( $\sigma \leq 0$ ).

Let  $\sigma^* = -\det \phi_I$  be a reference (negative) value of the determinant of that element. Then, we determine  $\delta$  by ensuring that  $\sigma_\delta(\sigma^*)$  is always positive. Since  $\sigma_\delta$  is a strictly increasing function, then we guarantee that  $\sigma_\delta > 0$  for  $\sigma > \sigma^*$ . In particular, we impose

$$\sigma_\delta(\sigma^*) = \frac{1}{2} \left( \sigma^* + \sqrt{(\sigma^*)^2 + 4\delta^2} \right) = \tau > 0,$$

where  $\tau$  is a given tolerance. Hence,

$$\delta(\sigma^*) = \frac{1}{2} \sqrt{(2\tau + |\sigma^*|)^2 - (\sigma^*)^2} = \sqrt{\tau^2 + \tau |\sigma^*|}. \quad (17)$$

The parameter  $\tau$  should be small compared to  $\sigma$ . We propose to select  $\tau$  according to

$$\tau = \alpha |\sigma^*|, \quad (18)$$

where  $\alpha = 10^{-3}$  has been found to work well in practice and used for all the presented examples. The final expression for  $\delta$  becomes,

$$\delta(\sigma^*) = |\sigma^*| \sqrt{\alpha^2 + \alpha}. \quad (19)$$

which implies that (see Figure 4):

$$\begin{aligned} \sigma_{\delta(\sigma^*)}(0) &= \delta(\sigma^*), \\ \sigma_{\delta(\sigma^*)}(\sigma^*) &= \tau. \end{aligned}$$

In order to illustrate the behavior of the regularized distortion measure presented in Equation (15) we consider a triangular element with two fixed nodes,  $\mathbf{x}_0 = (0, 0.5)$  and  $\mathbf{x}_1 = (0, -0.5)$ , and examine the mesh distortion measure as the third node  $\mathbf{x}_2(x) = (x, 0)$  is moved along the  $x$ -axis, see Figure 5. In addition, we select the equilateral triangle as the initial element and  $\delta = 0.05$  for the regularized distortion measure. Figure 6(a) shows the shape distortion measure, Equation (1), when node  $\mathbf{x}_2$  moves from  $x = -5$  to  $x = 5$ . Note that only  $0 < x < \infty$  correspond to valid configurations. For  $x \rightarrow \pm\infty$  the distortion measure tends to infinity since the triangle tends to a degenerated configuration. The minimum values ( $\eta = 1$ ) are achieved at  $x = \pm\sqrt{3}/2$  where the triangle is equilateral (ideal). However, only  $x = \sqrt{3}/2$  corresponds to a valid configuration. At the local minimum  $x = -\sqrt{3}/2$  the element is inverted and has negative area. Moreover, at  $x = 0$ , when the triangle has null area, the shape distortion measure has a vertical asymptote. Figure 6(b)



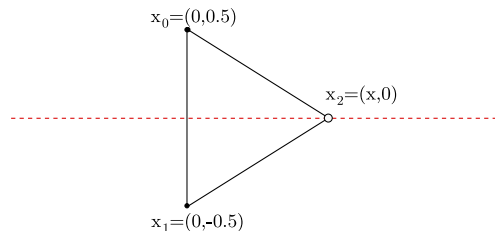
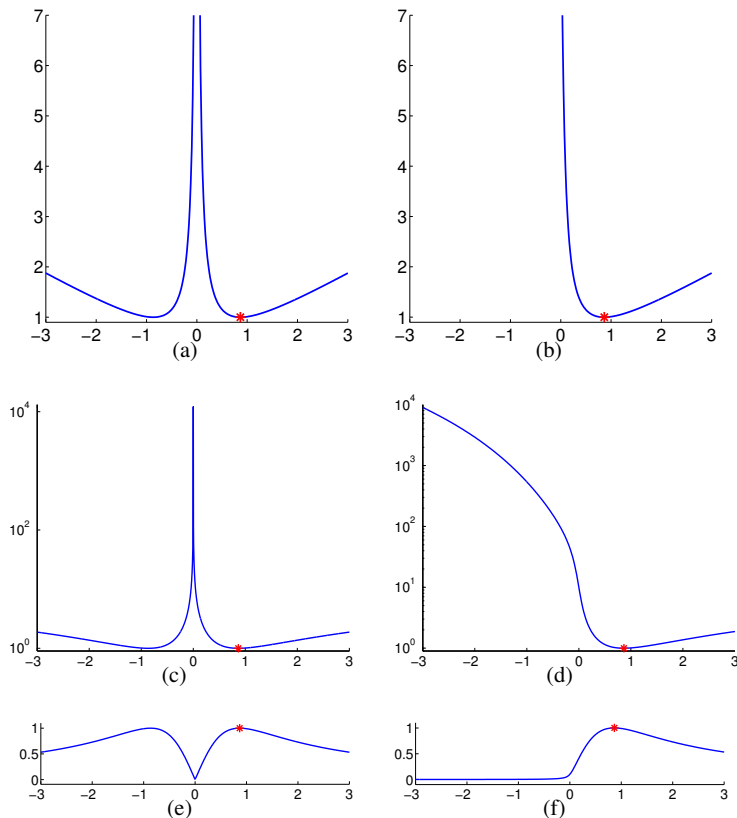
Figure 5. Triangle with  $x_2$  moving on a segment.

Figure 6. Shape distortion and quality measures for the linear triangle test when node  $x_2$  moves from  $x = -5$  to  $x = 5$ : (a) distortion, (b) regularized distortion, (c) distortion in logarithmic scale, (d) regularized distortion in logarithmic scale, (e) quality, and (f) regularized quality. The ideal configuration of the triangle is plotted with a red dot.

shows the value of the regularized shape distortion measure, Equation (16), when node  $x_2$  moves from  $x = -5$  to  $x = 5$ . To show that the asymptote has been removed, we plot in Figure 6(c) and 6(d) the distortion measure using a logarithmic scale. The regularization removes the vertical asymptote at  $x = 0$ , and results in a continuous and differentiable function with a single minimum near  $x = \sqrt{3}/2$  (the ideal configuration). Figure 6(e) and 6(f) plot the shape quality measure and the regularized shape quality measure, respectively. Note that, according to Equation (2), the quality measures are computed as the inverse of the corresponding distortion measure. Finally, Figure 7 shows the distortion and quality measures when the free node  $x_2$  moves in  $\mathbb{R}^2$ . Specifically, Figure 7(a) displays the shape distortion measure, Equation (1), and Figure 7(b) displays the regularized shape distortion measure, Equation (16). When the regularized distortion measure is used, the local

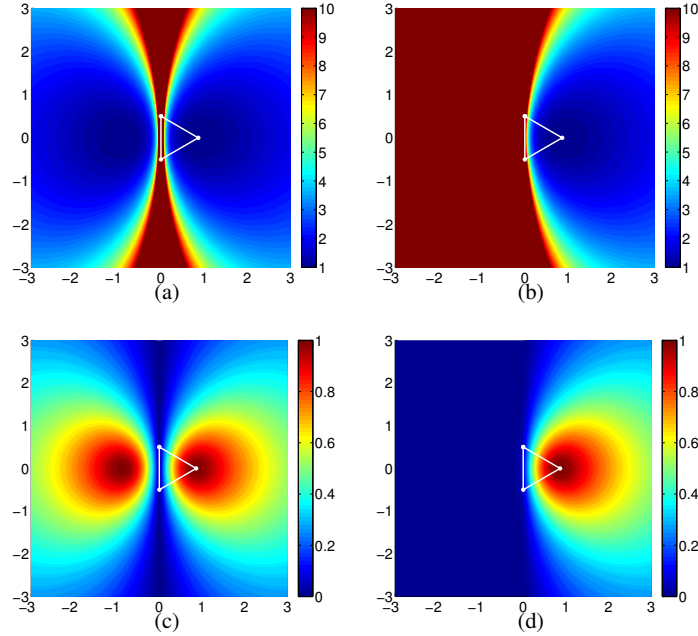


Figure 7. Shape distortion and quality measures for the linear triangle test when node  $x_2$  moves in a quadrilateral: (a) distortion, (b) regularized distortion, (c) quality, and (d) regularized quality. The ideal configuration of the triangle is also shown (white).

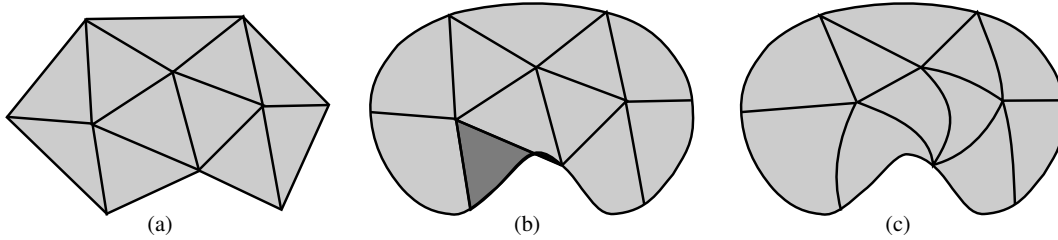


Figure 8. High-order mesh generation procedure: (a) ideal high-order mesh  $\mathcal{M}_I$ , (b) initial curved mesh  $\phi_h^0(\mathcal{M}_I)$ , and (c) final optimized high-order mesh  $\phi_h^*(\mathcal{M}_I)$ . Invalid elements are colored in dark gray.

minimum outside the feasible region is removed and the function is continuous and differentiable. Figures 7(c) and 7(d) present the corresponding quality measures.

## 5. CURVED HIGH-ORDER MESH GENERATION PROCESS

The *a posteriori* high-order mesh generation process is composed by three steps: generating the ideal mesh; curving the mesh boundary; and curving the mesh volume. Figure 8 illustrates these three main steps in 2D.

### 5.1. Generating the ideal mesh

A starting linear mesh can be generated using any established mesh generation procedure that provides control over the size and shape of the generated elements. Note that these mesh characteristics will be inherited by the final high-order mesh. That is, in terms of topology and element size and shape, we assume that the initial linear mesh is optimal. Once the linear mesh is

generated, we add additional nodes to increase its polynomial degree, see Figure 8(a), to that of the final mesh. This straight-sided high-order mesh becomes the ideal mesh  $\mathcal{M}_I$ .

### 5.2. Curving the mesh boundary

The second step of the proposed method is to curve the faces of the high-order straight-sided mesh that lie on the boundary of the geometry. This will allow generating high-quality elements near the boundaries. To generate a valid and high-quality high-order surface mesh,  $g_h$ , we use the technique presented in [33, 55], which is based on an optimization method to generate high-order meshes with the nodes on the CAD surfaces. Generating a valid surface mesh is of the major importance for ensuring the robustness of the proposed procedure. Next, we curve the boundary faces of the straight-sided to match the surface mesh determined by  $g_h$  and set the resulting high-order mesh to be the initial configuration,  $\phi_h^0(\mathcal{M}_I)$ , for our optimization procedure. It is important to point out that this initial configuration may contain invalid (tangled) elements on the boundary, the dark gray element in Figure 8(b) illustrates this situation in 2D. In Section 6.3, we show several configurations where invalid tetrahedra appear close to the curved features of the geometry.

### 5.3. Curving the mesh volume

Finally, we smooth and untangle the curved and high-order mesh using the optimization procedure proposed in Section 3.3. Specifically, we solve the minimization problem stated in Equation (7) starting from the initial curved mesh  $\phi_h^0(\mathcal{M}_I)$ . Figure 8(c) shows the optimized mesh,  $\phi_h^*(\mathcal{M}_I)$ , in the 2D case. Note that in this example the regularized mesh distortion measure of Equation (16) was used to explicitly penalize those elements that have negative values of the Jacobian determinant.

## 6. RESULTS

In this section, we present several examples to illustrate the main features of the proposed approach. In all figures, the meshes are colored according to the point-wise quality, i.e. the inverse of the distortion, see Equation (2) and Equation (4). It is also convenient to quantify the distortion and quality of an element. We follow [32, 35], and define the element distortion as,

$$\eta_E := \frac{\|\mathbf{M}\phi_E\|_{E^I}}{\|\mathbf{1}\|_{E^I}},$$

which results in an element quality given by  $q_E := 1/\eta_E$ . For each example, we present a table summarizing the element quality statistics. Specifically, we show the minimum quality, and the number of tangled elements ( $\#inv$ ) before the smoothing and untangling process is initiated. We highlight that in all cases, the optimized mesh increases the minimum and it does not include any inverted element. Since the element distortion (quality) measure is defined as a normalized average, we have that the maximum (minimum) value of the point-wise distortion (quality) measure on that element is bigger (smaller) than the value of the element distortion (quality) measure. Accordingly, the figures present information on the quality with more resolution and restrictiveness than the quality statistics tables. Nevertheless, an invalid element is always detected in the tables since it has a null (divergent) value of the element quality (distortion).

All the results have been obtained on a MacBook Pro with one dual-core Intel Core i7 CPU, with a clock frequency of 3.0 GHz, and a total memory of 16 GBytes. As a proof of concept, a mesh optimizer has been fully developed in MATLAB without using any additional toolbox. The MATLAB prototyping code is sequential (one execution thread), corresponds to the implementation of the method presented in this work, and uses the non-linear Gauss Seidel procedure detailed in Appendix A. Although the code is not optimized, is not parallel, and is not compiled, we provide an indication of its performance to illustrate the cost of the mesh optimization. To this end, we consider a unit-length 3D box as the initial geometry and three fully tangled meshes of interpolation degree four composed by 100 (1289), 1051 (12431), and 10067 (112615) elements (nodes). To

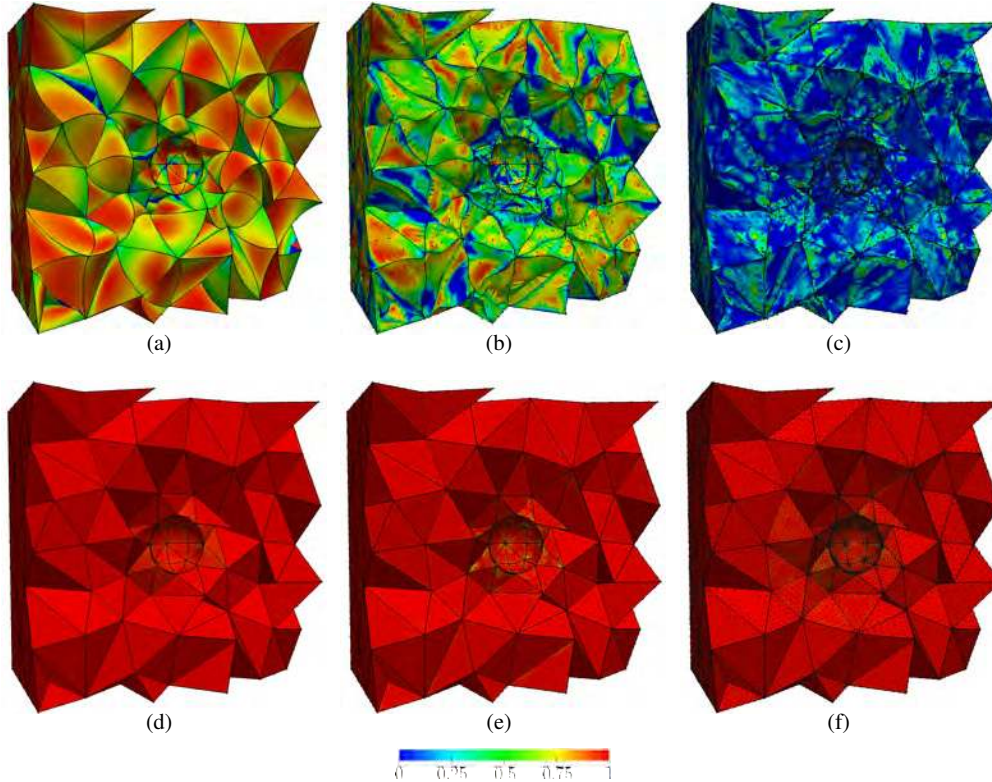


Figure 9. Initial and smoothed curved high-order meshes on a cube with a spherical cavity. Polynomial degrees: (a,d) 2, (b,e) 5 and (c,f) 10.

fully untangle and smooth the four meshes, the interpreted optimizer has needed 1.55, 4.3, and 22.4 minutes, respectively. It is important to point out that for bigger meshes the code seems to perform more efficiently. The explanation is that the function call overhead arising from the MATLAB interpreted code is more relevant for smaller meshes than for bigger meshes. Specifically, the code runs in chunks of nodes the same optimization operations (vectorization) to remove the overhead of interpreted loops. Smaller meshes benefit less from this vectorization since the amount of work is smaller in the resulting chunks of nodes and therefore, the function call overhead is relatively more significant.

### 6.1. Robustness for high polynomial degrees

To illustrate the ability of our approach to untangle meshes which may contain a large number of inverted elements, we consider a cube of side length 5 with an spheric cavity of diameter 1 placed at the center of the cube. The initial isotropic linear mesh consists of 1441 elements. Next, we increase the polynomial degree of the elements to degrees 2, 5 and 10. We then curve the boundary faces to match the geometry. In order to test our approach, we randomly perturb the interior nodes obtaining meshes that have 373, 1333 and 1439 inverted elements, respectively (see Table I). Finally, we apply our optimization procedure to obtain valid meshes for all cases. Figure 9 shows the initial meshes containing invalid elements and the final high-quality meshes for the different polynomial degrees. In Table I we summarize the element quality statistics for the initial and final meshes.

In the next example, we illustrate the robustness of our approach to deal with large boundary deformations and non-convex geometries. We consider the valid curved mesh of polynomial degree five of the previous example and displace the sphere 1.5 units towards the right boundary. The

Table I. Quality statistics for the meshes shown in Figure 9. The meshes are composed by 1441 elements, and 2327, 32382 and 249387 nodes, for polynomial degrees 2,5, and 10, respectively.

Mesh	Figure	Min	#inv
Initial	9(a)	0.00	373
Smoothed	9(d)	0.96	0
Initial	9(b)	0.00	1333
Smoothed	9(e)	0.97	0
Initial	9(c)	0.00	1439
Smoothed	9(f)	0.96	0

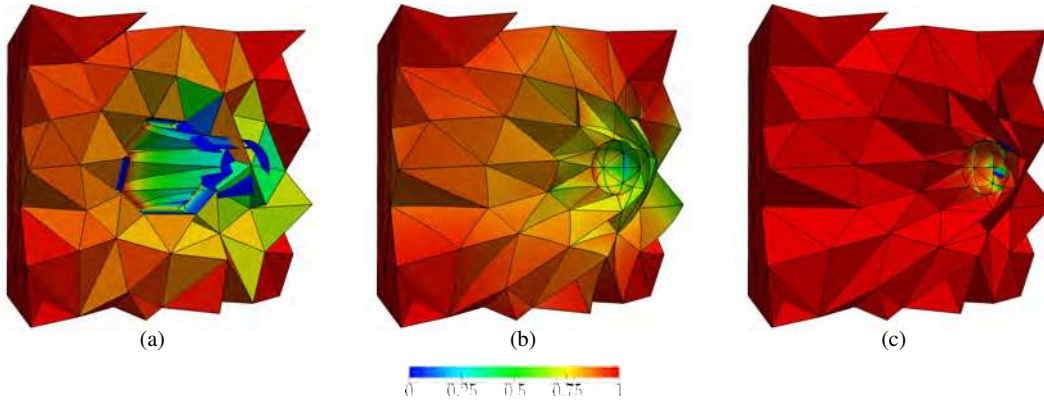


Figure 10. Meshes of polynomial degree 5 on a cube with a spherical cavity. (a) Initial curved mesh. (b) Mesh optimized with our objective function. (c) Mesh optimized with an objective function with constant Hessian.

Table II. Quality statistics for the meshes of polynomial degree 5 shown in Figure 10. The meshes are composed of 1441 elements and 32382 nodes .

Formulation	Figure	Min	#inv
Initial curved mesh	10(a)	0.00	134
Non-linear	10(b)	0.15	0
Linear (starting from 10(a))	-	0.00	110
Linear (starting from 10(b))	10(c)	0.00	15

resulting mesh contains 134 invalid elements, see Figure 10(a). Using the proposed method, we are able to untangle this inverted configuration obtaining a valid mesh, see Figure 10(b). We compare our approach with a linear method obtained by imposing, for each element, that the Jacobian of the physical configuration is equal to the Jacobian of the initial configuration in a least-squares sense. Hence, the objective function in that case is:

$$f(\mathbf{x}_1, \dots, \mathbf{x}_{n_F}; \mathbf{x}_{n_F+1}, \dots, \mathbf{x}_{n_N}) = \|\mathbf{D}\phi_h - \mathbf{Id}\|^2. \quad (20)$$

This simpler formulation fails to generate a valid mesh when it starts from the tangled configuration presented in Figure 10(a). The fact that the linear approach is unable to handle inverted elements is further illustrated by the fact that starting from the valid mesh obtained by our non-linear method, see Figure 10(b), the linear method generates a invalid mesh that contains 15 inverted elements, see Figure 10(c). The mesh quality statistics for this example considering the different objective functions in the optimization process are summarized in Table II. We note that our proposed nonlinear least-squares approach combined with the ability to handle inverted elements is the only approach capable of consistently producing valid meshes.

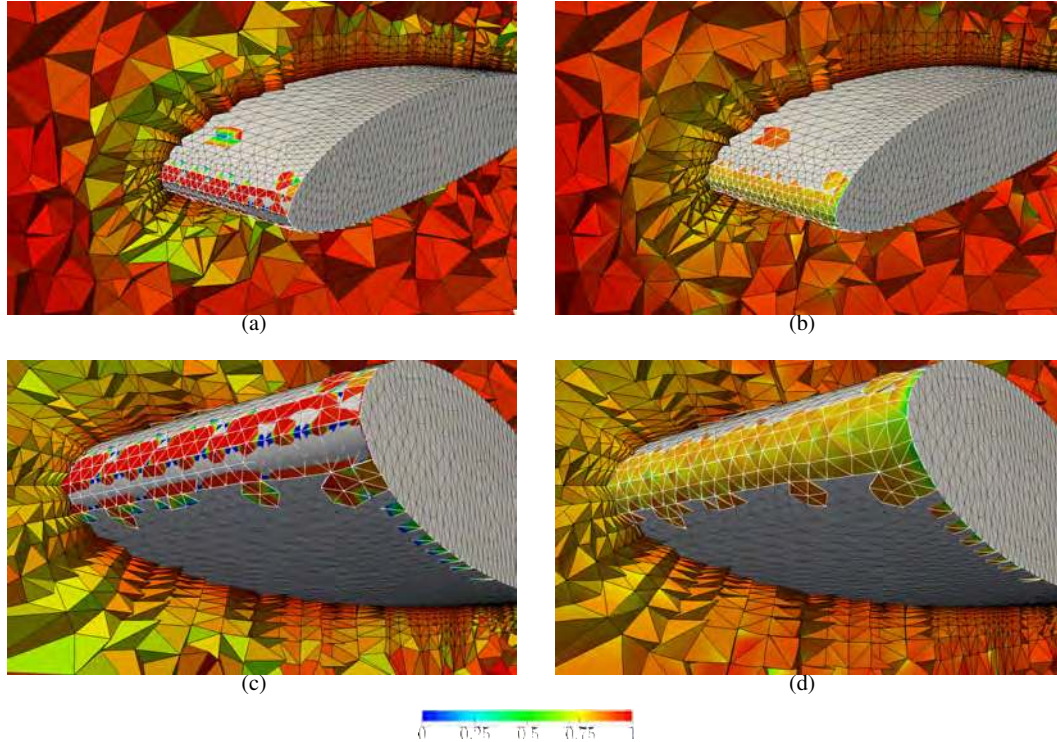


Figure 11. Tetrahedral meshes of polynomial degree 2 with boundary layer around a SD7003 airfoil. Initial curved high-order mesh (a) overview and (c) zoom. Final smoothed high-order mesh: (b) overview and (d) zoom. In these two figures we highlight with white edges the inverted high-order elements that appear in the initial curved high-order mesh.

Table III. Quality statistics of a mesh of polynomial degree 2 for a SD7003 airfoil. The mesh is composed by 177208 elements and 244346 nodes.

Mesh	Figure	Min	#inv
Initial	11(a)	0.00	1096
Smoothed	11(b)	0.29	0

The objective of the comparison of our non-linear method with a standard linear least-squares method is to highlight that it is required to account for the Jacobian of the iso-parametric mapping in a non-linear manner. Thus, it is possible to penalize the inversion of previously valid elements. On the contrary, in the linear case, the optimal position can correspond to a configuration composed by invalid elements even when the initial approximation is valid. This is a well-known issue of standard smoothers, such as Laplacian based methods. In practice, invalid elements usually appear next to non-convex geometries and where large deformations are imposed, as in the presented example. Notice that Laplacian smoothing can be interpreted as an iterative method that minimizes an energy functional that has a constant Hessian, such as the linear method presented in this example.

### 6.2. Preservation of the features of the initial mesh including boundary layers

In this example, we illustrate how our approach preserves the features of the linear mesh such as a stretched boundary layer mesh, see [59]. We consider a linear boundary layer mesh for a SD7003 airfoil. We increase the order of the polynomial degree to 2. Next, we curve the boundary faces to match the geometry, see Figure 11(a). In Table III we observe that this initial mesh contains 1096 tangled elements, all of them adjacent to the airfoil. Figure 11(c) shows a zoom of the inverted

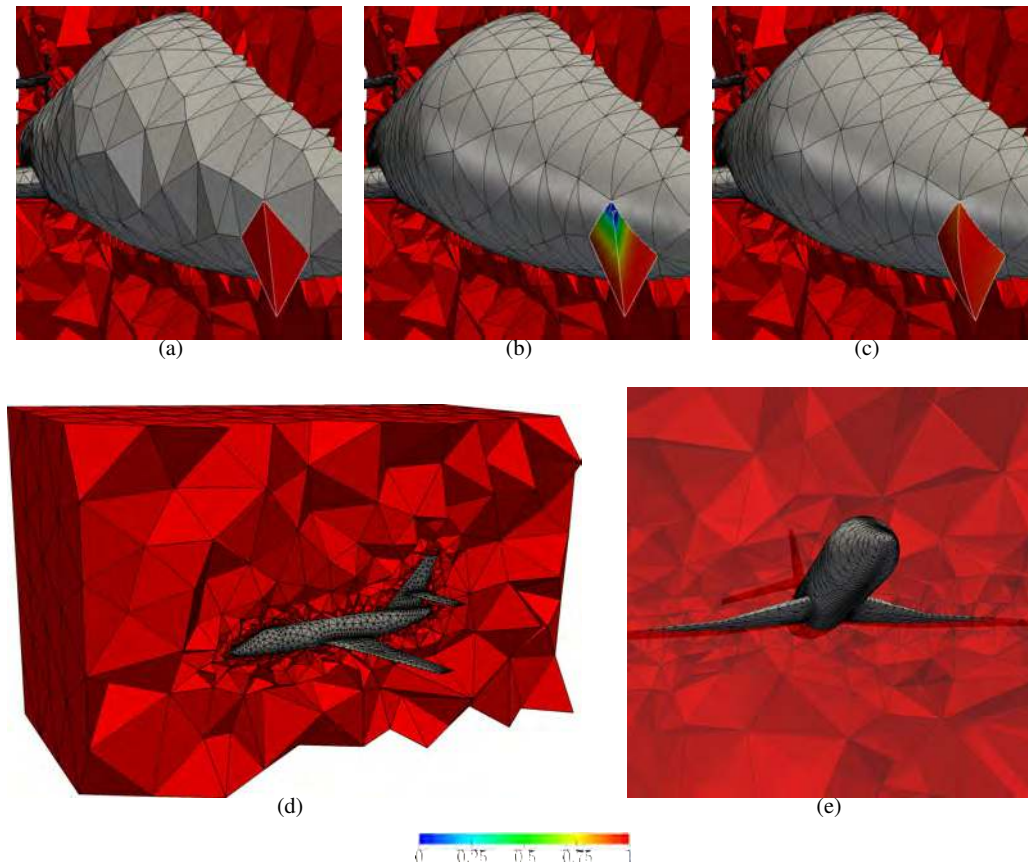


Figure 12. Tetrahedral meshes of polynomial degree 4 around a simplified Falcon aircraft. Detail of the: (a) ideal straight-sided mesh, (b) initial curved mesh, and (c) smoothed mesh. (d,e) General views of the smoothed mesh. In figures (a,b,c) two inverted high-order elements that appear when the mesh is curved are shown with the edges in white.

Table IV. Quality statistics of a mesh of polynomial degree 4 for a simplified Falcon aircraft. The mesh is composed by 27511 elements and 317854 nodes.

Mesh	Figure	Min	#inv
Straigh-sided	12(a)	1.00	0
Initial	12(b)	0.00	12
Smoothed	12(c)	0.31	0

elements in the lower surface close to the leading edge. Figure 11(b) presents the final high-order mesh obtained by using the straight-sided high-order mesh as the initial configuration in our optimization process. All the inverted elements have been untangled and the final mesh presents a minimum quality of 0.29, see Table III. Figure 11(d) shows that the inverted elements in the leading edge have been untangled.

### 6.3. Generation of curved high-order tetrahedral meshes from CAD models

We consider the CAD definition of a simplified Falcon aircraft and a mesh of degree 4 with valid and high-quality elements on the exterior domain. The straight-sided high-order mesh is shown in Figure 12(a). We choose this mesh as the ideal mesh, since we want to preserve the mesh features (shape, size, anisotropy...) of the linear mesh. We curve the boundary faces by relocating the nodes

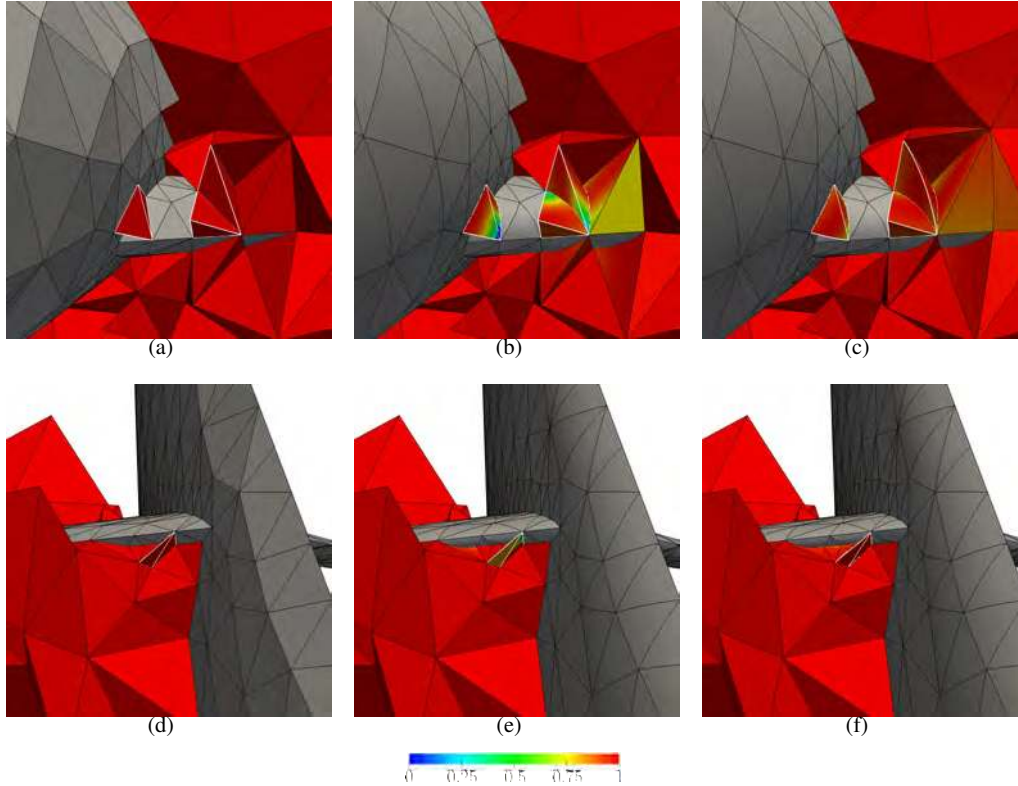


Figure 13. Tetrahedral meshes of polynomial degree 4 around a simplified Falcon aircraft. Detail of the mesh where inverted elements can be observed: (a,d) straight-sided mesh, (b,e) initial curved mesh, and (c,f) smoothed mesh. Some inverted high-order elements that appear when the mesh is curved (b,d) are shown with white edges.

Table V. Quality statistics of the mesh of polynomial degree 4 for a simplified Falcon aircraft used in the numerical simulation. The mesh is composed by 64992 elements and 96258 nodes.

Mesh	Figure	Min	#inv
Initial	-	0.00	11
Smoothed	14	0.85	0

on the CAD surface [33, 55]. This process leads to 12 invalid elements. Figure 12(b) presents a detail of the initial curved mesh near the nose and Figure 12(c) shows a detail of the optimized mesh near the nose including the two inverted elements highlighted in Figure 12(b). We highlight that the final mesh is composed by valid and high-quality elements. Additional details of the different meshes are shown in Figure 13.

#### 6.4. Inviscid flow solution on a curved and high-order tetrahedral mesh

We perform a flow simulation using a high-order mesh for the simplified Falcon geometry of the previous example. We consider an inviscid flow with a free stream Mach number of 0.6 ( $M_\infty$ ) and an angle of attack of  $0^\circ$  ( $\alpha$ ). The mesh quality statistics of the initial and optimized meshes are shown in Table V. Then, we use this mesh to obtain an approximation of the steady state solution of the Euler equations. This approximation is obtained with a 3D and parallel solver [60] using the hybridized discontinuous Galerkin (HDG) method [61, 62, 63, 64]. Both the steady state solution and the mesh are represented by element-wise polynomials of degree 4. The mesh is composed by 64992 elements and 129984 faces. Each element (face) of polynomial degree four contains 35



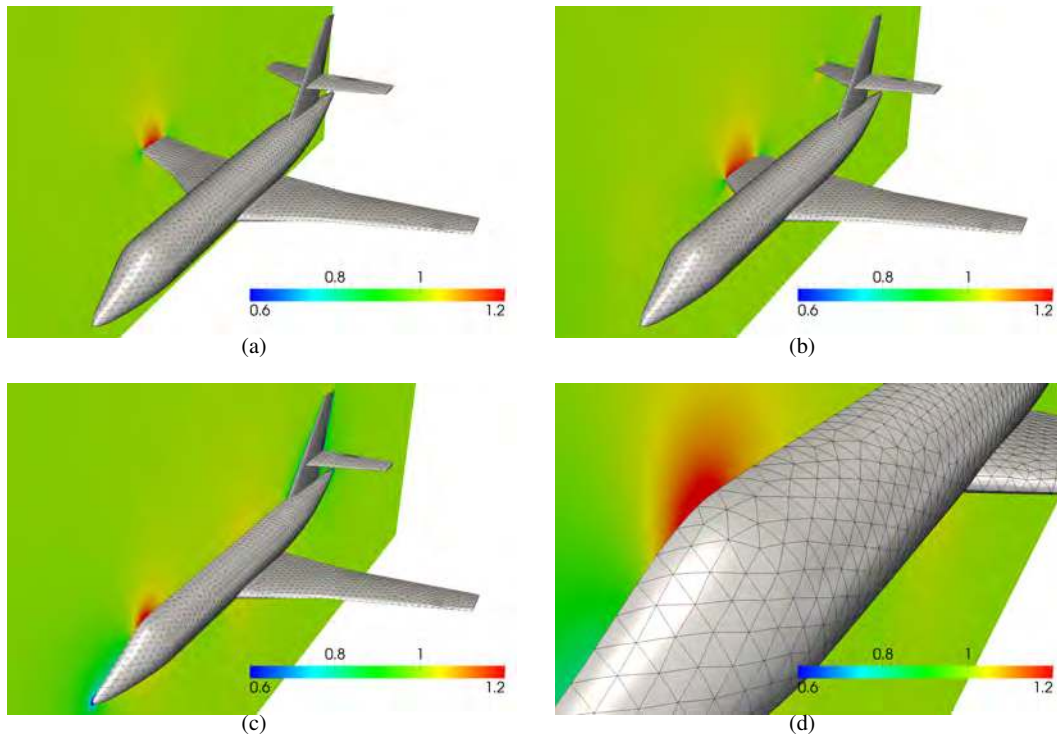


Figure 14. Curved mesh and inviscid flow around a simplified Falcon aircraft for polynomial degree four,  $\alpha = 0^\circ$ , and  $M_\infty = 0.6$ . Distribution of the magnitude of the velocity on different cut planes: (a,b,c) general view, and (d) detailed view.

(21) points. Furthermore, the solution of the compressible Euler equations has 5 components per point that correspond to the conserved quantities, namely: the density, the momentum, and the energy. This results in an HDG solution having a total of 14103264 degrees of freedom (DOFs) where: 11373600 DOFs are on the elements, and 2729664 DOFs are on the faces. To reduce the computation time, the mesh is partitioned in 128 sub-domains and accordingly, the HDG solver is run on 128 cores. In Figure 14, we present the magnitude of the velocity of the steady state of the inviscid flow, and a detail of the curved and high-order mesh on the aircraft surface. We point out that the curved elements are required to obtain a steady state solution of the Euler equations. Note that if the initial linear mesh is converted to a high-order mesh of degree four, but composed by straight-sided tetrahedra, the HDG solver does not converge to a steady state solution. This is in full agreement with the results reported in [16], where they show that piece-wise linear approximations of the curved surfaces lead to artificially polluted solutions of the Euler equations. On the contrary, the meshes generated with the proposed method allow the convergence of the HDG solver.

## 7. CONCLUDING REMARKS

We have presented an *a posteriori* method for generating curved, high-order conformal tetrahedral meshes from CAD models. The main driver for the proposed method is to obtain valid meshes for high-order finite element analyses. We remark that for the proposed smoothing and untangling method to work it is necessary that the initial tetrahedral mesh (determining the ideal mesh), and the curved and high-order surface mesh (determining the boundary mapping) are valid meshes. The untangling procedure requires that one scalar parameter be specified to control the amount of regularization in case of inverted elements. To this end, we have proposed a practical approach to chose this scalar parameter.

The robustness of the smoothing and untangling method emerges from the properties of the proposed optimization, namely: to repair invalid curved meshes (untangling), to transform valid configurations to valid configurations (consistency), to deal with any polynomial degree (high-order), and to preserve the geometrical features of the initial mesh (shape, stretching and size). To test the robustness of the untangling capability, we have considered several examples. Specifically, we have shown that the method untangles meshes which contain large number of invalid initial elements for: approximations up to degree 10, large deformations of the curved boundaries, concave boundaries, and highly stretched boundary layer elements. Finally, we have shown one example that demonstrates that the generated meshes are valid and therefore, allow obtaining converged solutions with an unstructured, high-order HDG inviscid flow solver.

Several aspects of the proposed method have to be investigated and improved in the near future. First, we have planned to implement the method in parallel to deal with larger meshes. In addition, we will perform a study on the cost of the optimization. This study will analyze the number of iterations required to converge, the scalability of the solver in terms of the number of elements, and the speed-up obtained with the parallel implementation. Second, note that the proposed mesh curving method tries to minimize the distortion and therefore, to maximize a measure of the mesh quality. In this sense, it could be interesting to perform a study on the influence of the quality of the curved meshes on different aspects of the HDG solver: non-linear solver robustness, mesh convergence rate, and accuracy of several outputs of interest. Finally, in this work we did not perform any viscous flow simulation using the boundary layer capability developed. To address this issue, in future works we will perform viscous flow simulations of practical interest on meshes that feature properly curved boundary layers obtained with the proposed method.

#### ACKNOWLEDGEMENTS

The first author was sponsored by CUR from DIUE of the *Generalitat de Catalunya* and the European Social Fund (grants FI-DGR'11 and BE-DGR'12), and by the Ferran Sunyer i Balaguer Foundation (grant FSB'13).

#### A. NUMERICAL OPTIMIZATION

To solve the optimization problem in Equation (14), we can choose a global or a local approach. For small problems solving the global system resulting from assembling the contributions of all the elements works well. If the problem is too large and we want to reduce the memory requirements, we choose a non-linear iterative method that: exploits the locality of the problem, avoids solving large linear systems, and is well suited for parallelization (by coloring the mesh nodes). Specifically, we use a non-linear iterative Gauss-Seidel method determined by the iteration

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^k - \alpha_i^k [\nabla_{ii}^2 f(\mathbf{w}_i^k)]^{-1} \nabla_i f(\mathbf{w}_i^k) \quad i = 1, \dots, n_F, \quad (21)$$

where  $\alpha_i^k$  is the step length, and

$$\mathbf{w}_i^k = (\mathbf{x}_1^{k+1}, \dots, \mathbf{x}_{i-1}^{k+1}, \mathbf{x}_i^k, \mathbf{x}_{i+1}^k, \dots, \mathbf{x}_{n_F}^k; \mathbf{x}_{n_F+1}^0, \dots, \mathbf{x}_{n_N}^0)$$

is the vector of updated node locations for the  $i - 1$  first nodes. Note that  $\nabla_i$  and  $\nabla_{ii}^2$  denote the gradient and the Hessian with respect to the coordinates  $\mathbf{x}_i$  of node  $i$ .

To implement this iterative non-linear solver, we have to compute the gradient  $\nabla_i f$ , the Hessian  $\nabla_{ii}^2 f$ , and the step length,  $\alpha_i^k$ . According to Equation (13) and taking into account Equation (8), the computation of the gradient

$$\nabla_i f(\mathbf{x}_1, \dots, \mathbf{x}_{n_F}; \mathbf{x}_{n_F+1}, \dots, \mathbf{x}_{n_N}) = \frac{\partial}{\partial \mathbf{x}_i} \sum_{e=1}^{n_E} \|\mathbf{M}\phi_{E_e} - 1\|_{E_e^I}^2 = \sum_{e=1}^{n_E} \frac{\partial}{\partial \mathbf{x}_i} \int_{E_e^I} (\mathbf{M}\phi_{E_e}(\mathbf{y}) - 1)^2 d\mathbf{y},$$

**Algorithm 1** Backtracking Line Search

---

```

1: function BACKLINESEARCH(Vector  $\mathbf{w}_i^k$ , Vector  $\mathbf{p}_i^k$ )
2:   Set  $\alpha > 0$ ,  $\rho \in (0, 1)$ ,  $c \in (0, 1)$ ;
3:    $\mathbf{w}_i^\alpha \leftarrow \mathbf{w}_i^k + (\mathbf{0}, \dots, \mathbf{0}, \alpha \mathbf{p}_i^k, \mathbf{0}, \dots, \mathbf{0})$ ;
4:   while  $f(\mathbf{w}_i^\alpha) > f(\mathbf{w}_i^k) + c\alpha[\nabla_i f(\mathbf{w}_i^k)]^T \mathbf{p}_i^k$  do
5:      $\alpha \leftarrow \rho\alpha$ ;
6:      $\mathbf{w}_i^\alpha \leftarrow \mathbf{w}_i^k + (\mathbf{0}, \dots, \mathbf{0}, \alpha \mathbf{p}_i^k, \mathbf{0}, \dots, \mathbf{0})$ ;
7:   end while
8:   return  $\alpha$ ;
9: end function

```

---

can be simplified. That is,  $\int_{E_e^I} (\eta(\mathbf{D}\phi_{E_e}(\mathbf{y}) - 1)^2 \mathrm{d}\mathbf{y})$  only depends on the coordinates of the nodes of the element  $E_e^P$ . Therefore, we have that

$$\frac{\partial}{\partial \mathbf{x}_i} \int_{E_e^I} (\mathbf{M}\phi_{E_e}(\mathbf{y}) - 1)^2 \mathrm{d}\mathbf{y} = \frac{\partial}{\partial \mathbf{x}_i} \int_{E_e^I} (\eta(\mathbf{D}\phi_{E_e}(\mathbf{y})) - 1)^2 \mathrm{d}\mathbf{y} = 0$$

for all the elements that do not contain the node  $i$ . Thus, the gradient can be evaluated as

$$\nabla_i f(\mathbf{x}_1, \dots, \mathbf{x}_{n_F}; \mathbf{x}_{n_F+1}, \dots, \mathbf{x}_{n_N}) = \sum_{e \sim i} \frac{\partial}{\partial \mathbf{x}_i} \int_{E_e^I} (\mathbf{M}\phi_{E_e}(\mathbf{y}) - 1)^2 \mathrm{d}\mathbf{y},$$

where  $e \sim i$  denotes that the summation is performed only for the elements that contain the node  $i$ . Therefore, if we define

$$\hat{f}(\mathbf{x}_i) := \sum_{e \sim i} \int_{E_e^I} (\mathbf{M}\phi_{E_e}(\mathbf{y}) - 1)^2 \mathrm{d}\mathbf{y} = \sum_{e \sim i} \|\mathbf{M}\phi_{E_e} - 1\|_{E_e^I}^2, \quad (22)$$

we have that

$$\nabla_i f(\mathbf{x}_1, \dots, \mathbf{x}_{n_F}; \mathbf{x}_{n_F+1}, \dots, \mathbf{x}_{n_N}) = \nabla_i \hat{f}(\mathbf{x}_i). \quad (23)$$

Moreover, the Hessian can be computed as

$$\nabla_{ii}^2 f(\mathbf{x}_1, \dots, \mathbf{x}_{n_F}; \mathbf{x}_{n_F+1}, \dots, \mathbf{x}_{n_N}) = \nabla_{ii}^2 \hat{f}(\mathbf{x}_i). \quad (24)$$

Finally, we have to compute the step length  $\alpha_i^k$ . To this end, we use the Backtracking Line Search algorithm [65] detailed in Algorithm 1, where we set:  $\alpha = 1$ ,  $\rho = 0.5$  and  $c = 10^{-4}$ . Note that in this algorithm, we have to evaluate the global objective function  $f$  and its gradient to check the sufficient decrease condition in Line 4. From Equation (23), the sufficient decrease condition is equivalent to

$$f(\mathbf{w}_i^\alpha) > f(\mathbf{w}_i^k) + c\alpha[\nabla_i \hat{f}(\mathbf{x}_i^k)]^T \mathbf{p}_i^k,$$

where  $\mathbf{w}_i^\alpha$  is defined in Line 3 of Algorithm 1. Moreover, we have that

$$f(\mathbf{w}_i^\alpha) - f(\mathbf{w}_i^k) = \hat{f}(\mathbf{x}_i^\alpha) - \hat{f}(\mathbf{x}_i^k),$$

since the contributions of the elements that do not depend on the free node are mutually cancelled, being  $\mathbf{x}_i^\alpha = \mathbf{x}_i^k + \alpha \mathbf{p}_i^k$ . Therefore, the sufficient decrease condition is equivalent to

$$\hat{f}(\mathbf{x}_i^\alpha) > \hat{f}(\mathbf{x}_i^k) + c\alpha[\nabla_i \hat{f}(\mathbf{x}_i^k)]^T \mathbf{p}_i^k. \quad (25)$$

Taking into account Equation (23), (24), and (25), we observe that we only need to compute the gradients, the Hessian, and the value of the local function  $\hat{f}$  introduced in Equation (22).

In our implementation, we exploit the computational reduction associated with the evaluation of the function  $\hat{f}$ . To this end, we denote by  $\mathcal{M}_\mathbf{x}$  the elements that contain a free node  $\mathbf{x}$ . The set of elements  $\mathcal{M}_\mathbf{x}$  is referred as the *submesh* associated with node  $\mathbf{x}$ .

*Remark 1*

Let  $\mathbf{x}_i^k$  be the coordinates of node  $i$  at step  $k$ , and let  $\mathcal{M}_{\mathbf{x}_i^k}$  be the corresponding associated submesh composed by  $m_i$  elements. We say that

$$\hat{f}(\mathbf{x}_i) = \sum_{e \sim i} \|\mathbb{M}\phi_{E_e} - 1\|_{E_e^I}^2 = \sum_{e \sim i} \int_{E_e^I} (\mathbb{M}\phi_{E_e}(\mathbf{y}) - 1)^2 d\mathbf{y} = \sum_{e=1}^{m_i} \int_{E_e^I} (\mathbb{M}\phi_{E_e}(\mathbf{y}) - 1)^2 d\mathbf{y}$$

is a local merit function that measures the *deviation respect an ideal configuration of the submesh distortion associated with  $\mathbf{x}_i$* . According to this merit function, and to Equations (23), (24), and (25), we can implement the iteration  $k + 1$  for node  $i$  of the proposed non-linear method, Equation (21), as

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^k - \alpha_i^k [\nabla_{ii}^2 \hat{f}(\mathbf{x}_i^k)]^{-1} \nabla_i \hat{f}(\mathbf{x}_i^k) \quad i = 1, \dots, n_F.$$

## References

1. Szabo B, Babuška I. *Finite Element Analysis*. John Wiley & Sons New York, 1991.
2. Schwab C. *p- and hp-finite element methods: Theory and applications in solid and fluid mechanics*. Clarendon Press Oxford, 1998.
3. Deville MO, Fischer PF, Mund EH. *High-order methods for incompressible fluid flow*, vol. 9. Cambridge University Press, 2002.
4. Hesthaven J, Warburton T. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*. Texts in Applied Mathematics, Springer, 2007.
5. Karniadakis G, Sherwin S. *Spectral/hp element methods for computational fluid dynamics*. Oxford University Press, 2013.
6. Vos PE, Sherwin S, Kirby R. From  $h$  to  $p$  efficiently: implementing finite and spectral/hp element methods to achieve optimal performance for low- and high-order discretisations. *J. Comput. Phys.* 2010; **229**(13):5161–5181.
7. Cantwell C, Sherwin S, Kirby R, Kelly P. From  $h$  to  $p$  efficiently: strategy selection for operator evaluation on hexahedral and tetrahedral elements. *Comput. Fluids* 2011; **43**(1):23–28.
8. Cantwell C, Sherwin S, Kirby R, Kelly P. From  $h$  to  $p$  efficiently: selecting the optimal spectral/hp discretisation in three dimensions. *Math. Model. Nat. Phenom.* 2011; **6**(3):84–96.
9. Löhner R. Error and work estimates for high-order elements. *Int. J. Numer. Methods Fluids* 2011; **67**(12):2184–2188.
10. Yano M, *et al.*. An optimization framework for adaptive higher-order discretizations of partial differential equations on anisotropic simplex meshes. PhD Thesis, Massachusetts Institute of Technology 2012.
11. Kirby R, Sherwin S, Cockburn B. To CG or to HDG: a comparative study. *J. Sci. Comput.* 2012; **51**(1):183–212.
12. Huerta A, Roca X, Angeloski A, Peraire J. Are high-order and hybridizable discontinuous Galerkin methods competitive? *Oberwolfach Reports* 2012; **9**(1):485 – 487.
13. Huerta A, Angeloski A, Roca X, Peraire J. Efficiency of high-order elements for continuous and discontinuous Galerkin methods. *Int. J. Numer. Methods Eng.* 2013; **96**:529–560, doi:10.1002/nme.4547.
14. Löhner R. Improved error and work estimates for high-order elements. *Int. J. Numer. Methods Fluids* 2013; **72**:1207–1218.
15. Wang Z, Fidkowski K, Abgrall R, Bassi F, Caraeni D, Cary A, Deconinck H, Hartmann R, Hillewaert K, Huynh H, *et al.*. High-order cfd methods: current status and perspective. *International Journal for Numerical Methods in Fluids* 2013; **72**(8):811–845.
16. Bassi F, Rebay S. High-order accurate discontinuous finite element solution of the 2D Euler equations. *J. Comput. Phys.* 1997; **138**(2):251–285.
17. Dey S, Shephard MS, Flaherty JE. Geometry representation issues associated with p-version finite element computations. *Comput. Meth. Appl. M.* 1997; **150**(1–4):39–55.
18. Luo X, Shephard MS, Remacle JF. The influence of geometric approximation on the accuracy of higher order methods. *8th International Conference on Numerical Grid Generation in Computational Field Simulations*, 2002.
19. Xue D, Demkowicz L. Control of geometry induced error in hp finite element (FE) simulations. I. Evaluation of FE error for curvilinear geometries. *Internat. J. Numer. Anal. Model.* 2005; **2**(3):283–300.
20. Sevilla R, Fernández-Méndez S, Huerta A. NURBS-Enhanced Finite Element Method (NEFEM): a seamless bridge between CAD and FEM. *Arch. Comput. Methods Engrg.* 2011; **18**(4):441–484.
21. Dey S, O’Bara R, Shephard MS. Curvilinear mesh generation in 3D. *Comput. Aided Design* 2001; **33**:199–209.
22. Luo X, Shephard MS, Remacle JF, O’Bara R, Beall M, Szabó B, Actis R. P-version mesh generation issues. *Proc. 11th Int. Meshing Roundtable*, Citeseer, 2002; 343–354.
23. Luo X, Shephard MS, O’Bara R, Nastasia R, Beall M. Automatic p-version mesh generation for curved domains. *Eng. Comput.* 2004; **20**(3):273–285.
24. Shephard MS, Flaherty JE, Jansen K, Li X, Luo X, Chevaugeon N, Remacle JF, Beall M, O’Bara R. Adaptive mesh generation for curved domains. *Appl. Numer. Math.* 2005; **52**(2-3):251–271.
25. Persson PO, Peraire J. Curved mesh generation and mesh refinement using lagrangian solid mechanics. *Proc. 47th AIAA*, 2009.
26. Warburton T. An explicit construction of interpolation nodes on the simplex. *J. Eng. Math.* 2006; **56**(3):247–262.
27. Nielsen EJ, Anderson WK. Recent improvements in aerodynamic design optimization on unstructured meshes. *AIAA journal* 2002; **40**(6):1155–1163.

28. Oliver T. A high-order, adaptive, discontinuous Galerkin finite element method for the Reynolds-averaged Navier-Stokes equations. PhD Thesis, Massachusetts Institute of Technology 2008.
29. Xie Z, Sevilla R, Hassan O, Morgan K. The generation of arbitrary order curved meshes for 3D finite element analysis. *Comput. Mech.* 2012; **51**:361–374.
30. Sherwin S, Peiró J. Mesh generation in curvilinear domains using high-order elements. *Int. J. Numer. Meth. Eng.* 2002; **53**(1):207–223.
31. George PL, Borouchaki H. Construction of tetrahedral meshes of degree two. *Int J Numer Meth Eng* 2012; **90**(9):1156–1182.
32. Roca X, Gargallo-Peiró A, Sarrate J. Defining quality measures for high-order planar triangles and curved mesh generation. *Proc. 20th Int. Meshing Roundtable*. Springer International Publishing, 2012; 365–383.
33. Gargallo-Peiró A, Roca X, Peraire J, Sarrate J. Defining quality measures for mesh optimization on parameterized CAD surfaces. *Proc. 21st Int. Meshing Roundtable*. Springer International Publishing, 2013; 85–102.
34. Toulorge T, Geuzaine C, Remacle JF, Lambrechts J. Robust untangling of curvilinear meshes. *Journal of Computational Physics* 2013; **254**:8 – 26.
35. Gargallo-Peiró A, Roca X, Peraire J, Sarrate J. Defining quality measures for validation and generation of high-order tetrahedral meshes. *Proc. 22nd Int. Meshing Roundtable*. Springer International Publishing, 2014; 109–126.
36. Mitchell A, Phillips G, Wachspress E. Forbidden shapes in the finite element method. *IMA J. Appl. Math.* 1971; **8**(2):260.
37. Field D. Algorithms for determining invertible two-and three-dimensional quadratic isoparametric finite element transformations. *Int. J. Numer. Meth. Eng.* 1983; **19**(6):789–802.
38. Baart M, Mulder E. A note on invertible two-dimensional quadratic finite element transformations. *Commun. Appl. Numer. M.* 1987; **3**(6):535–539.
39. Johnen A, Remacle JF, Geuzaine C. Geometrical validity of curvilinear finite elements. *J. Comput. Phys.* 2013; **233**:359 – 372.
40. Salem A, Canann S, Saigal S. Robust distortion metric for quadratic triangular 2D finite elements. *Appl. Mech. Div. ASME.* 1997; **220**:73–80.
41. Salem A, Canann S, Saigal S. Mid-node admissible spaces for quadratic triangular arbitrarily curved 2D finite elements. *Int. J. Numer. Meth. Eng.* 2001; **50**(2):253–272.
42. Yuan K, Huang Y, Pian T. Inverse mapping and distortion measures for quadrilaterals with curved boundaries. *Int. J. Numer. Meth. Eng.* 1994; **37**(5):861–875.
43. Knupp PM. Label-invariant mesh quality metrics. *Proc. 18th Int. Meshing Roundtable*, Salt Lake City, 2009; 139–155.
44. Branets L, Carey G. Extension of a mesh quality metric for elements with a curved boundary edge or surface. *J. Comput. Inf. Sci. Eng.* 2005; **5**(4):302–308.
45. Salem A, Saigal S, Canann S. Mid-node admissible space for 3D quadratic tetrahedral finite elements. *Eng. Comput.* 2001; **17**(1):39–54.
46. Freitag LA, Plassmann P. Local optimization-based simplicial mesh untangling and improvement. *Int. J. Numer. Methods Engrg.* 2000; **49**:109–125.
47. Freitag LA, Knupp PM. Tetrahedral mesh improvement via optimization of the element condition number. *Int. J. Numer. Methods Engrg.* 2002; **53**:1377–1391.
48. Escobar JM, Rodríguez E, Montenegro R, Montero G, González-Yuste JM. Simultaneous untangling and smoothing of tetrahedral meshes. *Comput. Methods Appl. Mech. Engrg.* 2003; **192**(25):2775–2787.
49. Jiao X, Wang D, Zha H. Simple and effective variational optimization of surface and volume triangulations. *Engrg. Comput.* 2011; **27**:81–94.
50. Knupp PM. Hexahedral and tetrahedral mesh untangling. *Engrg. Comput.* 2001; **17**(3):261–268.
51. Knupp PM. A method for hexahedral mesh shape optimization. *Int. J. Numer. Methods Eng.* 2003; **58**(2):319–332.
52. López EJ, Nigro NM, Storti MA. Simultaneous untangling and smoothing of moving grids. *Int. J. Numer. Methods Engrg.* 2008; **76**(7):994–1019.
53. Garimella R, Shashkov M, Knupp PM. Triangular and quadrilateral surface mesh quality optimization using local parameterization. *Comput. Methods Appl. Mech. Engrg.* 2004; **193**(9–11):913–928.
54. Sastry S, Shontz S, Vavasis S. A log-barrier method for mesh quality improvement and untangling. *Engineering with Computers* 2012; doi:10.1007/s00366-012-0294-6.
55. Gargallo-Peiró A, Roca X, Sarrate J. A surface mesh smoothing and untangling method independent of the CAD parameterization. *Computational Mechanics* 2014; **53**(4):587–609, doi:10.1007/s00466-013-0920-1.
56. Sastry S, Shontz S, Vavasis S. A log-barrier method for mesh quality improvement. *Proc. 20th Int. Meshing Roundtable*. Springer International Publishing, 2012; 329–346.
57. Knupp PM. Algebraic mesh quality metrics. *SIAM J. Numer. Anal.* 2001; **23**(1):193–218.
58. Knupp PM. Algebraic mesh quality metrics for unstructured initial meshes. *Finite Elem. Anal. Des.* 2003; **39**(3):217–241.
59. Gargallo-Peiró A, Roca X, Peraire J, Sarrate J. Inserting curved boundary layers for viscous flow simulation with high-order tetrahedra. *Research Notes, 22nd Int. Meshing Roundtable*. Springer International Publishing, 2013.
60. Roca X, Nguyen N, Peraire J. Scalable parallelization of the hybridized discontinuous Galerkin method for compressible flow. *43rd AIAA Fluid Dynamics Conference and Exhibit*, 2013.
61. Cockburn B, Dong B, Guzmán J, Restelli M, Sacco R. A hybridizable discontinuous galerkin method for steady-state convection-diffusion-reaction problems. *SIAM J. Sci. Comput.* 2009; **31**:3827.
62. Nguyen N, Peraire J, Cockburn B. An implicit high-order hybridizable discontinuous Galerkin method for linear convection-diffusion equations. *J. Comput. Phys.* 2009; **228**:3232–3254.
63. Nguyen N, Peraire J, Cockburn B. An implicit high-order hybridizable discontinuous Galerkin method for nonlinear convection-diffusion equations. *J. Comput. Phys.* 2009; **228**:8841–8855.
64. Peraire J, Nguyen N, Cockburn B. A hybridizable discontinuous Galerkin method for the compressible Euler and Navier-Stokes equations. *Proceedings of the 48th AIAA Aerospace Sciences Meeting and Exhibit*, 2010; AIAA

Paper 2010–363.

65. Nocedal J, Wright S. *Numerical optimization*. Springer verlag, 1999.