Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 203 (2017) 362–374

Procedia Engineering

www.elsevier.com/locate/procedia

26th International Meshing Roundtable, IMR26, 18-21 September 2017, Barcelona, Spain

An augmented Lagrangian formulation to impose boundary conditions for distortion based mesh moving and curving

Eloi Ruiz-Gironés[a], Abel Gargallo-Peiró[a], Josep Sarrate[b], Xevi Roca[a,*]

[a]*Computer Applications in Science and Engineering, Barcelona Supercomputing Center, 08034 Barcelona, Spain*
[b]*Laboratori de Càlcul Numèric (LaCàN), Departament d'Enginyeria Civil i Ambiental, Universitat Politècnica de Catalunya (UPC), Campus Nord UPC, 08034 Barcelona, Spain.*

Abstract

We formulate a mesh morphing technique as mesh distortion minimization problem constrained to weakly satisfy the imposed displacement of the boundary nodes. The method is devised to penalize the appearance of inverted elements during the optimization process. Accordingly, we have not equipped the method with untangling capabilities. To solve the constrained minimization problem, we apply the augmented Lagrangian technique to incorporate the boundary condition in the objective function using the Lagrangian multipliers and a penalty parameter. We have applied the proposed formulation to mesh moving and mesh curving problems. The results show that the method has the ability to deal with large displacements for 2D and 3D meshes with non-uniform sizing, and mesh curving of highly stretched 2D high-order meshes.

Peer-review under responsibility of the scientific committee of the 26th International Meshing Roundtable.

Keywords: Mesh morphing; mesh moving; mesh curving; smoothing; boundary displacements; augmented Lagrangian

1. Introduction

In several applications such as large deformations and high-order mesh curving techniques, it is needed to deform an initial mesh in order to adapt it to a new configuration determined by a boundary displacement. Usually, it is necessary to recover a valid mesh in which the position of the boundary nodes is the prescribed one. There are different techniques to morph [1] a valid mesh, based on solid mechanics analogies [2–4], optimization-based methods [5–12], and solution of PDE's [13–15].

One critical step is how to impose the boundary displacement condition for the mesh morphing problem. When the mesh is subject to large displacements, the techniques to perform the mesh morphing process may not converge if inverted elements are originated during the procedure. In particular, the mesh morphing technique may require the capability of untangling to recover from invalid configurations and obtain a final valid mesh. For instance, in mesh distortion minimization methods [5,6,9,10], it is necessary to use a regularized distortion measure [6,16] to remove the singularities of the objective function. In [11,12], it is introduced a moving log-barrier to repair the inverted elements.

* Corresponding author
E-mail address: xevi.roca@bsc.es

1877-7058
Peer-review under responsibility of the scientific committee of the 26th International Meshing Roundtable.
10.1016/j.proeng.2017.09.820

In [3,4] the authors propose to incrementally move the mesh boundary. Other methods solve a non-linear PDE [13,14] in which the initial condition of the boundary nodes may impact the performance of the non-linear solver. Note that these different treatments arise because of the imposition of the boundary condition.

In this work, we propose to incorporate the boundary condition into a minimization problem. To this end, we pose a constrained minimization problem in which we optimize the mesh distortion, constrained to the boundary condition. To solve this non-linear minimization problem, we apply the augmented Lagrangian method [17], in which the boundary condition is incorporated into the target function by means of the Lagrangian multipliers and a penalty parameter. Then, we solve a series of minimization problems with increasing penalty parameter and better approximation of the Lagrangian multipliers. The algorithm finishes when the boundary constraint is satisfied within a prescribed tolerance.

The proposed method has several advantages. The mesh boundary is not fixed during the optimization process since it is driven by the penalty parameter and the Lagrange multipliers. Thus, similarly to the incremental moving methods, the boundary condition is only satisfied at the end of the optimization process. However, it is worth to point out that in our method we do not impose the trajectory of the boundary nodes during the deformation process, since it is automatically computed by the augmented Lagrangian method. Moreover, since the mesh is valid during the whole minimization, it is not necessary to feature untangling capabilities. We propose to use a global non-linear solver, in which the nodes of the mesh are moved at the same time. To this end, we use a backtracking line-search method [17] in which the advancing direction is computing using Newton's method and the step length is selected according to the Wolfe condition. We point out that to perform Newton's method we use the analytical derivatives of the objective function.

The rest of the paper is structured as follows. In Section 2, we review the existing literature related to the presented work. In Section 3, we formulate the constrained minimization problem and detail the implemented augmented Lagrangian method. In Section 4 ,we present several examples to show the features of the proposed method. Finally, in Section 5, we detail the conclusions and the future work.

2. Related work

In some applications, mesh untangling and smoothing can be required to repair invalid elements and improve the overall mesh quality when applying a mesh morphing technique. In this cases, the boundary nodes are displaced to a new position, and it is necessary to recover a new mesh composed of valid elements. There are different formulations to define a mesh morphing technique. For instance, in the solid mechanics approach, the mesh is moved using linear or non-linear elasticity analogies [2–4]. Other approaches consist on minimizing an objective function that measures a quantity of interest of the mesh, such as the mesh distortion [5,6,18], the minimum Jacobian of the iso-parametric mapping [11], or by formulating a variational minimization problem [14]. Finally, it is possible to define the mesh morphing process in terms of a PDE [13]. In all these approaches, it is necessary to fix the position of the boundary nodes in order to satisfy the prescribed mesh morphing displacement. Usually, it is introduced as a boundary condition for the mesh morphing problem. In the most common approach, the position of the boundary nodes is fixed, and then, the location of inner nodes is computed accordingly. However, when the boundary nodes are displaced in the initial step, invalid elements may appear, and they could hinder the convergence of the mesh morphing method, specially when complex geometries or non-uniform element sizes are present.

In reference [19], the authors propose to solve a constrained minimization problem in which the constraint is the final position of the boundary nodes. To this end, they apply a penalty method to solve a series of optimization problems to enforce the constraint. Other approaches first project the boundary nodes to the CAD model, and then pose an unconstrained minimization problem in which the boundary nodes can slide along the geometric entity they belong to [10,11,20]. In this case, the imposition of the boundary condition is more flexible than prescribing a fixed nodal value. The only requisite is that the boundary nodes are located on the required geometric entity. In [12], the authors combine a mesh curving technique with a geometric accuracy measure into a single functional. The boundary nodes are free to slide along the geometric entities taking into account the geometric accuracy.

In reference [21], the authors introduce a geometric accuracy measure, and it is later combined with a mesh curving technique in [22]. In this approach, all the nodes of the mesh are free to move in the space, while the boundary nodes take into account the geometric error. In the proposed optimization process, the geometric accuracy term is treated as a

boundary condition of the mesh curving problem by means of a penalty method. In this approach, the boundary nodes are not restricted to move along the geometric model. Instead, the boundary mesh approximates the CAD model in a weak sense.

References [3,13,23] propose an incremental displacement of the boundary nodes form the initial position to the target one. In this manner, inverted elements may not appear, and the optimization process is more robust. To this end, the optimization method tries a full step to the target position. If the optimization process fails, the step is divided until the optimization converges. This approach is iterated until the final position of the nodes is reached.

In incremental stepping methods [1,3,4] invalid intermediate boundary configurations might appear hampering the capability to recover a valid volume mesh. This is not the case with our method, since the boundary conditions are weakly imposed and we penalize the appearance of inverted elements during the optimization process. To avoid the appearance of invalid boundary configurations it is also possible to constrain the boundary nodes to move on top of the CAD entities and incorporate untangling capabilities [10,12,24]. On the contrary, in this work the boundary nodes are not constrained to be on top of the boundaries and the we do not need untangling capabilities. There are other methods, proposed for linear meshes, where the boundary condition is incorporated in the minimization functional [7,15,19]. However, these approaches do not use an augmented Lagrangian formulation to incrementally impose the boundary conditions as we propose here. Furthermore, we have checked our method not only for large displacements but for non-uniform sizing, highly stretched elements, and curved high-order meshes.

3. Formulation of the problem

In this section, we first review the point-wise distortion measure used in this work. Following, we detail the new technique that we use to prescribe a displacement of the mesh boundary. The goal is to obtain a valid mesh with optimal quality featuring no inverted elements during the deformation process.

3.1. Point-wise shape distortion measure

In this work, we use the point-wise shape distortion measure, introduced in [18,25]. It is defined in terms of a mapping $\boldsymbol{\phi}$ that transforms a given ideal mesh, $\mathcal{M}_I$, to a physical mesh, $\mathcal{M}_P$, using the shape distortion measure introduced in [5,26] as

$$\eta(\mathbf{D}\boldsymbol{\phi}(\mathbf{y})) = \frac{|\mathbf{D}\boldsymbol{\phi}(\mathbf{y})|^2}{n\sigma(\mathbf{D}\boldsymbol{\phi}(\mathbf{y}))^{2/n}}, \tag{1}$$

where $\mathbf{y}$ is a point in the ideal mesh, $\mathbf{D}\boldsymbol{\phi}$ is the Jacobian of $\boldsymbol{\phi}$, $\sigma(\cdot)$ is the determinant, n is the space dimension and $|\cdot| = \sqrt{(\cdot,\cdot)}$ is the Frobenius norm of matrices. The point-wise distortion measure is equal to one when the mapping $\boldsymbol{\phi}$ is locally a rotation, a translation or a scaling, and it is equal to infinity when the mapping is degenerated.

The point-wise distortion measure presents asymptotes when $\sigma(\mathbf{D}\boldsymbol{\phi}(\mathbf{y})) = 0$. This prevents its use in a continuous optimization procedure. To overcome this drawback, in [6,16] a regularization of Equation (1) is introduced as

$$\eta_\delta(\mathbf{D}\boldsymbol{\phi}(\mathbf{y})) = \frac{|\mathbf{D}\boldsymbol{\phi}(\mathbf{y})|^2}{n\sigma_\delta(\mathbf{D}\boldsymbol{\phi}(\mathbf{y}))^{2/n}}, \tag{2}$$

where

$$\sigma_\delta(\mathbf{D}\boldsymbol{\phi}(\mathbf{y})) = \frac{1}{2}\left(\sigma + \sqrt{\sigma^2 + 4\delta^2}\right), \tag{3}$$

and the value of the δ parameter depends on the required amount of regularization. For a further analysis on the selection of δ for high-order meshes, see [18,27–29]. In the case of linear meshes, the δ parameter can be chosen according to [6,30–32]. From now on, we can set $\delta = 0$ since we want to detect if a given element goes from a valid to an invalid configuration. However, we do not need untangling capabilities. Degenerated or inverted elements are detected since they feature an infinite value of distortion [33].

3.2. Optimization framework

Given an initial ideal mesh, $\mathcal{M}_I$, we want to characterize a morphed mesh, $\mathcal{M}_P$, in terms of a diffeomorphism $\boldsymbol{\phi}^*$. This diffeomorphism has to satisfy the morphing boundary condition, and should present optimal point-wise distortion. That is,

$$\begin{aligned} M\boldsymbol{\phi}^* &= 1, && \forall \mathbf{y} \in \mathcal{M}_I, \\ \boldsymbol{\phi}^* &= \mathbf{g}_D, && \forall \mathbf{y} \in \partial\mathcal{M}_I, \end{aligned} \tag{4}$$

where $\mathbf{g}_D$ is a known and fixed Dirichlet boundary condition in $\partial\mathcal{M}_I$ and

$$M\boldsymbol{\phi}^*(\mathbf{y}) = \eta_\delta(\mathbf{D}\boldsymbol{\phi}^*(\mathbf{y}))$$

is the point-wise distortion measure, Eq. (1), of the mapping $\boldsymbol{\phi}^*$ between $\mathcal{M}_I$ and $\mathcal{M}_P$ at each point $\mathbf{y} \in \mathcal{M}_I$.

Note that for a given initial mesh, $\mathcal{M}_I$, and boundary condition, $\mathbf{g}_D$, it may not exist a mapping $\boldsymbol{\phi}^*$ such that Equation (4) is verified. For this reason, we impose the optimality condition in a least-squares sense by means of a constrained minimization problem:

$$\begin{aligned} &\min_{\boldsymbol{\phi}} \|M\boldsymbol{\phi} - 1\|^2, \\ &\text{constrained to:} \\ &\boldsymbol{\phi} = \mathbf{g}_D \qquad \forall \mathbf{y} \in \partial\mathcal{M}_I, \end{aligned} \tag{5}$$

where

$$\|M\boldsymbol{\phi} - 1\|^2 = \sum_{e_I \in \mathcal{M}_I} \int_{e_I} (M\boldsymbol{\phi}(\mathbf{y}) - 1)^2 \, \mathrm{d}\mathbf{y},$$

denoting by e_I the elements of the ideal mesh $\mathcal{M}_I$, herein named ideal elements.

Equation (5) defines a constrained minimization problem in which we minimize the point-wise distortion of the mesh while the boundary condition is treated as a constraint. To solve this constrained minimization problem, we use the augmented Lagrangian technique, see [17]. The key idea of the augmented Lagrangian method is to solve a series of unconstrained minimization problems until the constraint is satisfied within a tolerance. The augmented Lagrangian functional associated with problem (5) is

$$E_{\lambda,\mu}(\boldsymbol{\phi}) = \|M\boldsymbol{\phi} - 1\|^2 - (\boldsymbol{\lambda}, \mathbf{c}_D)_{\partial\mathcal{M}_I} + \frac{1}{2}\mu^2 \|\mathbf{c}_D\|^2_{\partial\mathcal{M}_I}, \tag{6}$$

where $\mathbf{c}_D = (\boldsymbol{\phi} - \mathbf{g}_D)/\|1\|_{\partial\mathcal{M}_I}$ is the boundary constraint scaled by the measure of the boundary, μ is a penalty parameter and $\boldsymbol{\lambda}$ are the Lagrange multipliers associated to the constraint $\mathbf{c}_D$.

The scalar product $(\cdot,\cdot)_{\partial\mathcal{M}_I}$ and its associated norm, $\|\cdot\|_{\partial\mathcal{M}_I}$, are defined as

$$(\boldsymbol{\lambda}, \mathbf{c}_D)_{\partial\mathcal{M}_I} = \sum_{f_I \in \partial\mathcal{M}_I} \int_{f_I} \frac{1}{h} \boldsymbol{\lambda} \cdot \mathbf{c}_D \, \mathrm{d}\Gamma, \qquad \|\mathbf{c}_D\|^2_{\partial\mathcal{M}_I} = \sum_{f_I \in \partial\mathcal{M}_I} \int_{f_I} \frac{1}{h} \mathbf{c}_D \cdot \mathbf{c}_D \, \mathrm{d}\Gamma,$$

being f_I the elemental faces that are located at the boundary of the mesh, and h a measure of the boundary face size. By taking into account the element size of the boundary elements in the boundary integrals, we balance the distortion and the constraint contributions in the augmented Lagrangian functional.

Note that $\boldsymbol{\phi}$ is expressed in terms of the physical nodes as follows

$$\boldsymbol{\phi} = \sum_{i=1}^{n_N} \mathbf{x}_i N_i,$$

where n_N is the number of nodes in the mesh, and $\{N_i\}_{i=1,\dots,n_N}$ is a Lagrangian basis of element-wise polynomial shape functions continuous at the element interfaces. Thus, the augmented Lagrangian functional only depends on the position of the physical nodes:

$$E_{\lambda,\mu}(\boldsymbol{\phi}) = E_{\lambda,\mu}(\mathbf{x}_1, \dots, \mathbf{x}_n).$$

Algorithm 1 Augmented Lagrangian method.

Input: Mesh $\mathcal{M}_I$, ω^* ,ε^*, μ_0, ω_0, ε_0
Output: Mesh $\mathcal{M}_P$
1: **function** meshOptimization
2: $\boldsymbol{\phi} \leftarrow \mathbf{Id}$
3: $\boldsymbol{\lambda}_0 \leftarrow \mathbf{0}$
4: $m_0 \leftarrow 10$
5: $m_1 \leftarrow m_0$, $\mu_1 \leftarrow \mu_0$
6: $\omega_1 \leftarrow \omega_0$, $\varepsilon_1 \leftarrow \varepsilon_0$
7: **while** $\|\mathbf{c}_D\|^2_{\partial\mathcal{M}_I} > \varepsilon^*$ **and** $\|\nabla E_{\lambda_k,\mu_k}(\boldsymbol{\phi})\| > \omega^*$ **do**
8: optimizeFunction($E_{\lambda_k,\mu_k}(\boldsymbol{\phi})$, ω_k)
9: **if** $\|\mathbf{c}_D\|^2_{\partial\mathcal{M}_I} \leq \varepsilon_k$ **then**
10: $\boldsymbol{\lambda}_{k+1} \leftarrow \boldsymbol{\lambda}_k - \mu_k \mathbf{c}_D$
11: $m_{k+1} \leftarrow m_k$
12: $\mu_{k+1} \leftarrow \mu_k$
13: $\varepsilon_{k+1} \leftarrow \varepsilon_k / m_{k+1}^{0.9}$
14: $\omega_{k+1} \leftarrow \omega_k / \mu_{k+1}$
15: **else**
16: $\boldsymbol{\lambda}_{k+1} \leftarrow \boldsymbol{\lambda}_k$
17: $m_{k+1} \leftarrow 100 m_k$
18: $\mu_{k+1} \leftarrow \mu_0 / (m_0/m_{k+1})$
19: $\varepsilon_{k+1} \leftarrow \varepsilon_0 / (m_0/m_{k+1})^{0.1}$
20: $\omega_{k+1} \leftarrow \omega_0 / (m_0/m_{k+1})$
21: **end if**
22: **end while**
23: $\mathcal{M}_P \leftarrow \boldsymbol{\phi}\big(\mathcal{M}_I\big)$
24: **end function**

Moreover, we approximate the Lagrange multipliers, $\boldsymbol{\lambda}$, in a similar manner as

$$\boldsymbol{\lambda} = \sum_{j=1}^{m_N} \boldsymbol{\lambda}_j N_j^b,$$

where m_N is the number of boundary nodes, and $\{N_j^b\}_{j=1,\dots,m_N}$ is a Lagrangian basis of element-wise polynomial shape functions continuous at the element interfaces, defined at the boundary of the mesh. In our application, we use the same polynomial degree to define the physical mesh and the Lagrange multipliers.

Algorithm 1 details the proposed implementation of the augmented Lagrangian method adapted for mesh smoothing with prescribed boundary displacement.

The input of the algorithm is an initial mesh, $\mathcal{M}_I$, the tolerances for optimization of the non-linear objective function and the boundary condition, ω^* and ε^*, respectively, and the parameters of the augmented Lagrangian method. The algorithm stops when a solution is found that satisfies

$$\|\mathbf{c}_D\|^2_{\partial\mathcal{M}_I} < \varepsilon^*, \qquad \|\nabla E_{\lambda_k,\mu_k}(\boldsymbol{\phi})\| < \omega^*.$$

We initialize $\boldsymbol{\phi}$ to the identity mapping, **Id**. That is, we start the optimization process using the initial mesh. Note that the identity mapping is optimal with respect to the distortion measure. However, it does not satisfy the boundary constraint. Then, we initialize the multipliers in Line 3, the penalty parameter in Line 5, and the tolerances to check the evolution of the non-linear problem and the constraint in Line 6. Lines 7–22 define the main loop of the augmented Lagrangian method. This loop is performed until the stopping criteria are satisfied. In Line 8 we optimize Functional (6) using the current $\boldsymbol{\lambda}$ and μ parameters. The stopping criterion of this non-linear solver is $\|\nabla E_{\lambda_k,\mu_k}(\boldsymbol{\phi})\| < \omega_k$. Then, we update the values of $\boldsymbol{\lambda}$ and μ according to the current value of the constraint $\mathbf{c}_D$. If the constraint norm is small enough, Line 9, we update the values of $\boldsymbol{\lambda}$, keep the value of μ, and tighten the tolerances. On the contrary, if the constraint norm is too high, Line 15, we increase the value of μ, keep the current value of $\boldsymbol{\lambda}$, and tighten the tolerances accordingly.

Note that the only unknowns of Functional (6) are the nodal coordinates, while $\boldsymbol{\lambda}$ and μ are parameters that are selected according to the augmented Lagrangian method, see Lines 9–21. Thus, we minimize Functional (6) with respect to the position of the mesh nodes, given the penalty parameter μ, and the Lagrange multipliers of the boundary nodes $\boldsymbol{\lambda}_1, \dots, \boldsymbol{\lambda}_{m_N}$.

To optimize each non-linear problem, we use a backtracking line-search method in which the advancing direction is selected using Newton's method, and the step length is set according to Wolfe condition, see [17] for further details. The linear systems related to the Newton's method iterations are solved using the GMRES iterative method preconditioned using an incomplete LU decomposition without any levels of fill-in.

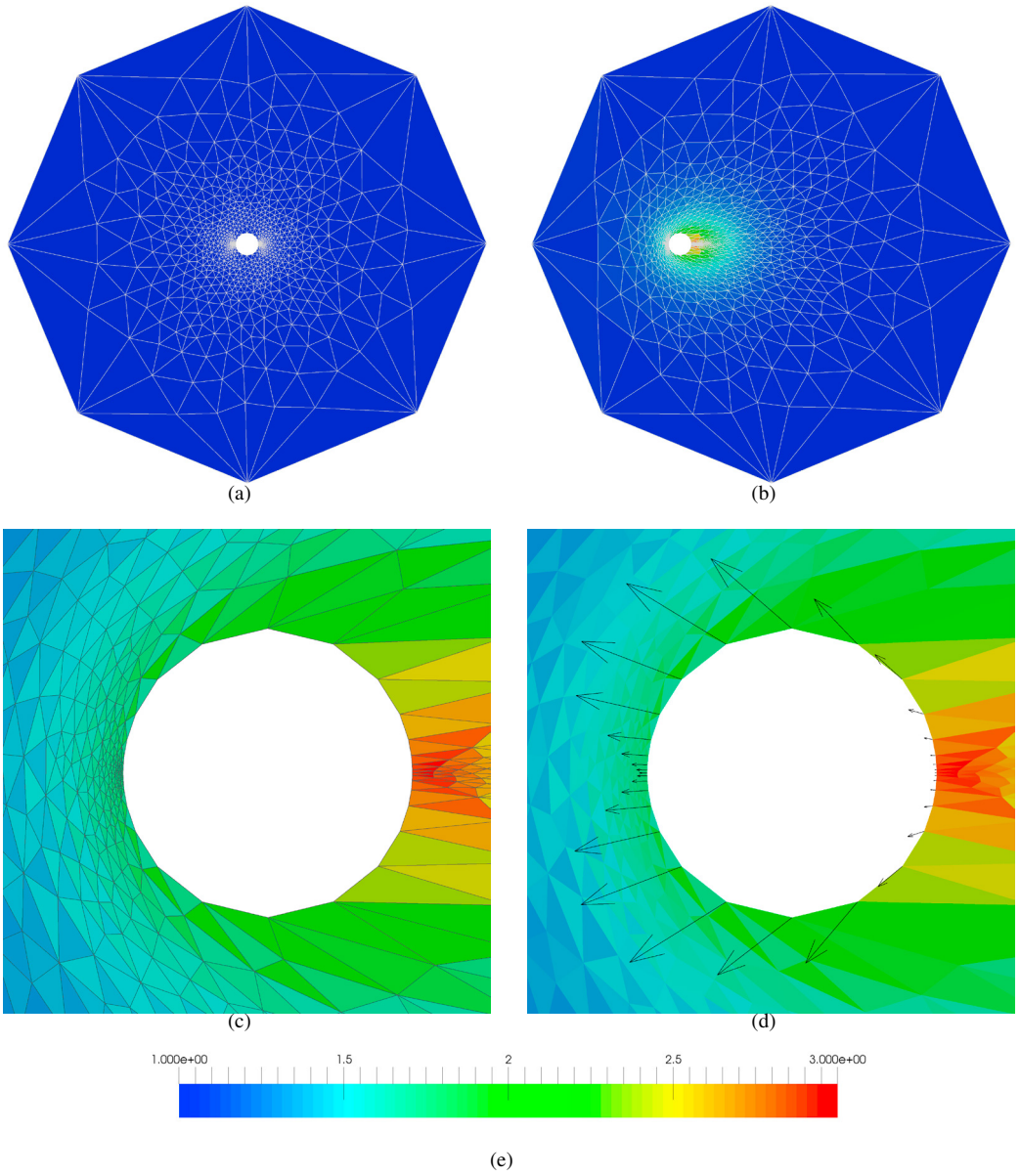

Fig. 1. Non-uniform size mesh colored by distortion: (a) initial, (b) optimized, (c) detail, (d) Lagrange multipliers on the boundary points, and (e) elemental distortion color legend.

Our method is devised to favor that at each optimization step, a valid mesh is deformed to a valid mesh. To this end, two main ingredients have been considered. First, Functional (6) penalizes inverted elements by taking an infinite value. Second, we have incorporated a backtracking line search to Newton's method. Thus, if a Newton full step was deforming a valid mesh to an invalid mesh, the backtracking line-search would decrease the step length to recover a valid mesh.

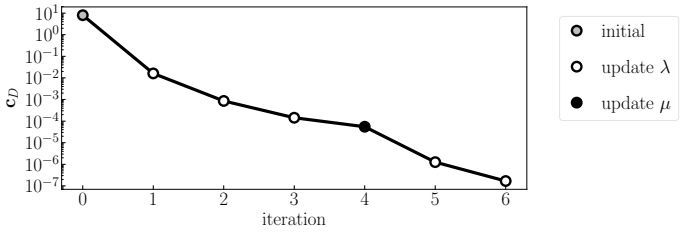

Fig. 2. Evolution of the constraint norm over the augmented Lagrangian iterations for the mesh moving of a circle.

4. Examples

This section presents several examples to show the capabilities of the presented mesh morphing method. Specifically, we show two examples of large displacements for two- and three-dimensional geometries, and one example of high-order mesh curving. Finally, in Table 2, we present for each example the evolution of the relative mesh quality with respect to the initial mesh during the augmented Lagrangian optimization process, defined as $q = \frac{\|1\|}{\|M\boldsymbol{\phi}\|}$. At the first iteration, the mesh is optimal with respect to the mesh quality. Note that in all the cases, in the next iterations, the optimization process adapts the boundary mesh without hampering the mesh quality.

4.1. Displacement of an inner boundary: non-uniform size mesh

The challenge for this example is to deal with a large displacement of a non-uniform size mesh where we move a circular inner boundary, see Figure 1(a). The inner and outer radius are 1 and 21 units, respectively, and the applied displacement of the inner circle is 8 units. The mesh is composed of 1642 linear triangles and 841 nodes. The element size is not uniform and ranges from 0.2 to 0.02 in the inner circle, and it is constant to 16.5 at the outer boundary. Figure 1(b) shows the final mesh after applying the proposed augmented Lagrangian solver. The mesh is composed of valid elements, and the maximum distortion is around 2.1 (minimum quality of 1/2.1 over a maximum of 1). Moreover, the constraint norm at the final iteration is of the order of 10^{-7}. While the proposed method does not locate the nodes exactly at the constrained position, the user can control the constraint norm by selecting the required constraint tolerance.

Figure 1(c) presents a detailed view of the mesh around the inner circle. The elements around the circle are stretched in order to accommodate the displacement of the circular boundary. In this case, the maximum distortion is located at the rear part of the circle. Finally, Figure 1(d) shows the Lagrange multipliers obtained using the proposed augmented Lagrangian method. The Lagrange multipliers can be interpreted as the *force* required to locate the nodes at the final position while keeping a valid mesh.

Figure 2 shows the evolution of the constraint norm in logarithmic scale over the iterations of the proposed augmented Lagrangian solver. That is, we show the accuracy of the boundary representation during the optimization process. We plot using white dots the iterations where the augmented Lagrangian updates the Lagrangian multiplier, and with black dots when the penalty parameter is updated. In this case, the constraint norm decreases in every iteration, and only six iterations of the augmented Lagrangian method are necessary to obtain a valid mesh that satisfies the boundary condition.

We perform a scaling analysis in order to determine the execution times of the proposed augmented Lagrangian optimizer. To this end, we solve the linear systems with a sparse direct solver and we perform several levels of refinement to the initial mesh shown in 1(a). In each refinement, we divide all the triangles of the mesh into four triangles. For each refinement level, we compute the execution time, the mesh quality and the number of augmented Lagrangian iterations, see Table 1. Note that at each step, the number of elements is multiplied by four, according to the refinement rule, while the number of nodes is roughly multiplied by four. It is worth to notice that the mesh quality and the number of augmented Lagrangian iterations is almost constant in all the executions. Finally, in order to compare the execution times of each case, we show them in Figure 1(a). Note that as the number of elements and nodes increase, the execution time also increases. This is because we have to build and solve a larger sparse linear system with more unknowns.

Table 1.

elements	nodes	time (s)	mesh quality	augmented Lagrangian iterations
1642	841	1.11	0.936	6
6568	3324	3.56	0.937	5
26272	13216	14.29	0.937	6
105088	52704	101.94	0.937	6
420352	210816	546.66	0.937	6

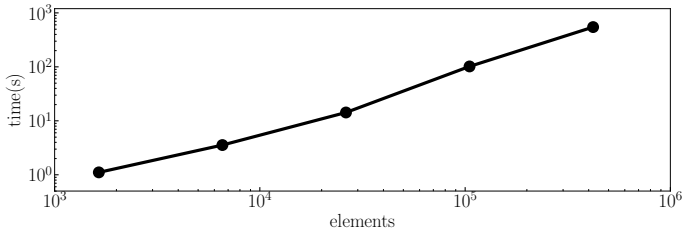

Fig. 3. Execution times for morphing the refined meshes of the circle.

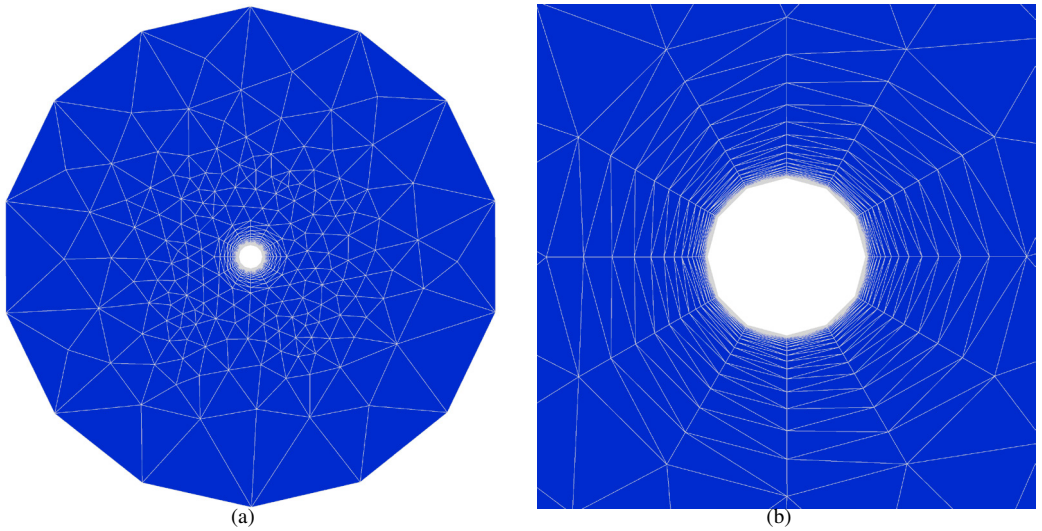

Fig. 4. Initial linear mesh with a boundary layer around the inner circle: (a) full view, (b) detailed view.

4.2. Mesh curving of a wall boundary: highly stretched triangles

In this example the main difficulty is to curve a high-order mesh of polynomial degree four with highly stretched elements. The initial mesh is generated for a domain determined by an exterior and internal circumference centered at the origin and with radius 21 and 1, respectively, see Figure 4(a). The mesh is composed by 697 nodes and 1368 elements and features a boundary layer next to the inner circumference. The boundary layer, see Figure 4(b), is composed of 42 levels determined by a growing ratio of 1.3, a wall size of 10^{-5}, and a tangential size of 0.524. Note that this determines a maximum element stretching ratio around $1 : 50000$.

The proposed constrained minimization process is applied to obtain a valid curved mesh composed of high quality elements. Figure 5(a) shows the curved boundary elements and the Lagrange multipliers associated with the constraint. Note that the Lagrange multipliers are approximated using continuous element-wise polynomials of degree four. That is, we use the same polynomial degree of the mesh to approximate the Lagrange multipliers.

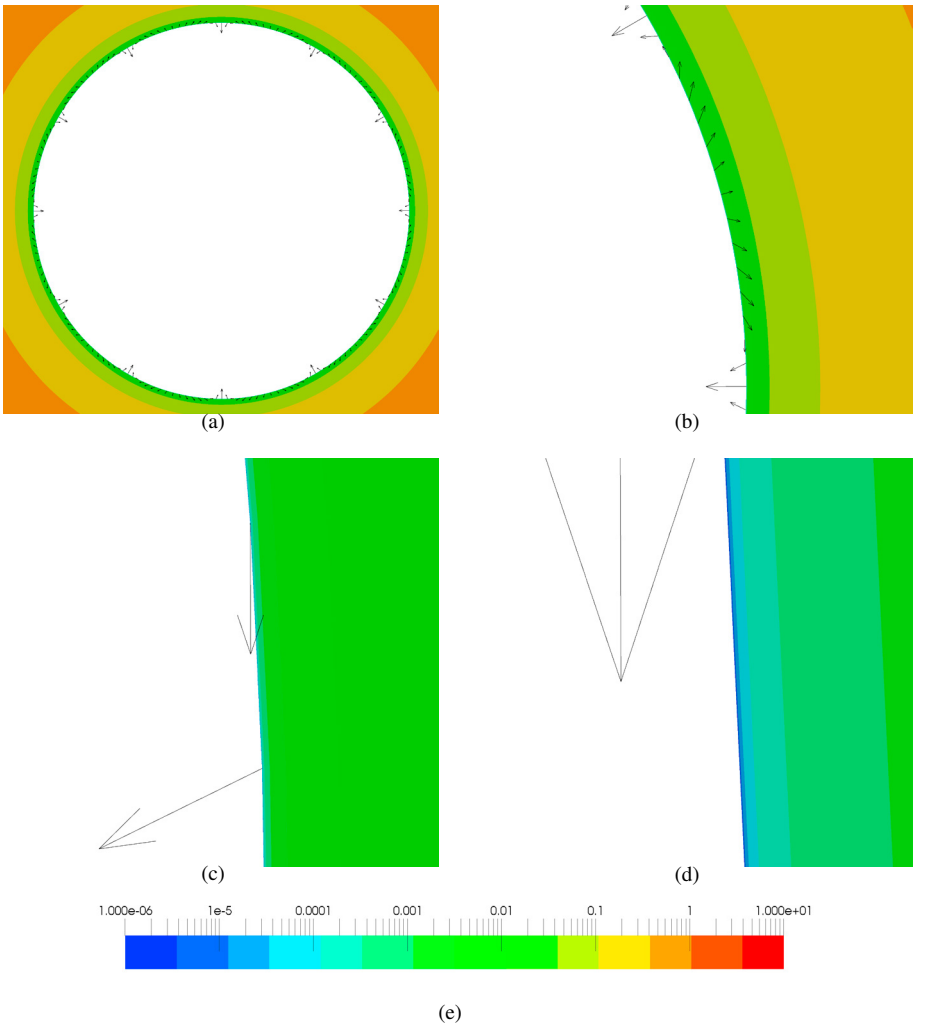

Fig. 5. Final curved boundary layer mesh of polynomial degree 4 and Lagrange multipliers. The mesh is colored by distance to the wall: (a) initial, (b) optimized, (c) Lagrange multipliers on the boundary nodes, (d) detail, and (e) color legend in logarithmic scale for the distance to the wall.

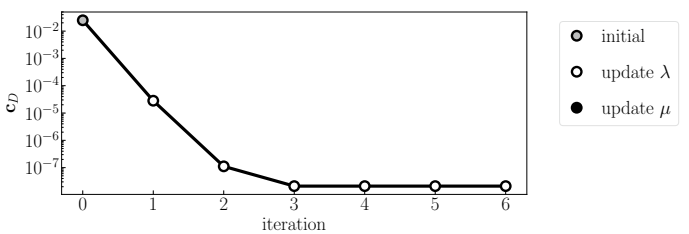

Fig. 6. Evolution of the constraint norm over the augmented Lagrangian iterations for the mesh curving of a circle.

Figures 5(a) to 5(d) present the mesh colored according to the wall distance in logarithmic scale. We highlight that the wall distance is still preserved by the final curved high-order mesh. Note that the wall distance is correctly reproduced near the inner circle and therefore, the boundary layer elements have been correctly curved. To optimize this mesh, the execution time of the augmented Lagrangian solver is 11.13 seconds.

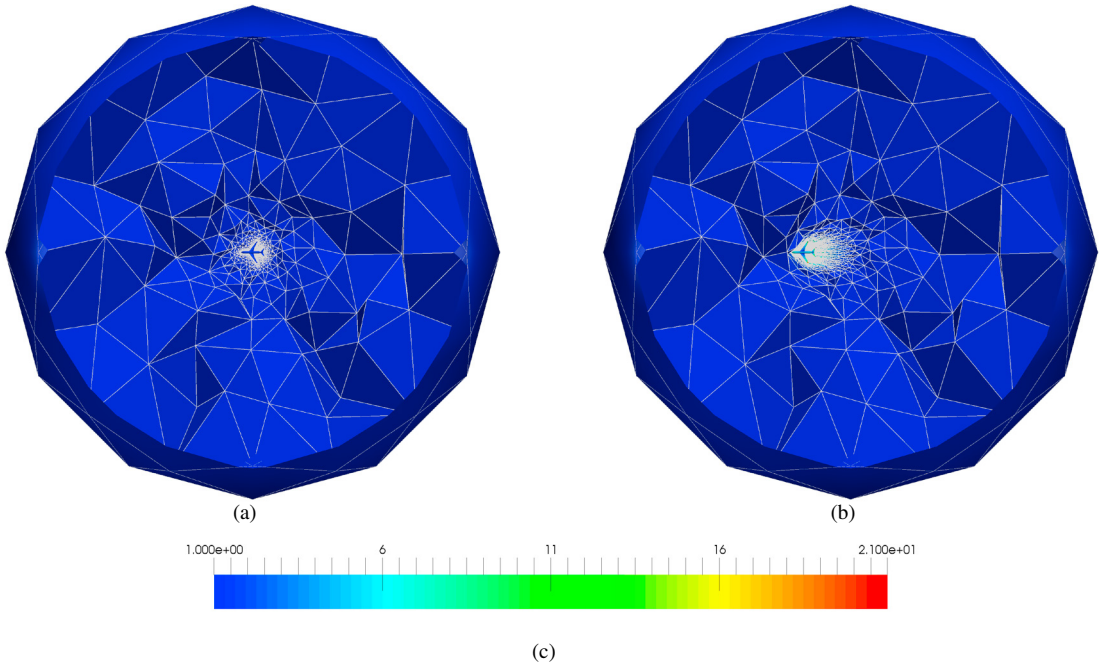

Fig. 7. Non-uniform size mesh of an aircraft colored by distortion: (a) initial, (b) optimized, and (c) distortion color legend.

The evolution of the constraint norm over the iterations of the augmented Lagrangian method is shown in Figure 6. The augmented Lagrangian method converges in six iterations to obtain the optimal mesh. Similarly to the previous example, we depict with black dots the iterations when the penalty parameter is updated, and with white dots when the Lagrangian multipliers are updated. The norm of the constraint decreases at each iteration of the solver, since we have obtained a good approximation of the Lagrange multipliers. At the final iteration, the constraint norm is of the order of 10^{-7}.

4.3. Displacement of an aircraft: non-uniform size tetrahedra over a complex geometry

The objective of this example is to show that the proposed method is able to handle large displacements of complex geometries without generating inverted elements. To this end, we generate a mesh for the exterior domain of an aircraft corresponding to the geometry provided by the Drag Prediction Workshop, see Figures 7(a) and 8(a). The mesh contains 20192 nodes and 97868 linear tetrahedra. The length of the aircraft is two units, and we apply a displacement of four units. Figure 7(b) shows a general view of the final mesh, and Figure 8(b) shows a detailed view of the mesh around the aircraft.

Figures 9(a) and 9(b) show the distribution of the Lagrange multipliers magnitude at the aircraft. Note that the magnitude of the Lagrange multipliers varies between 10^{-4} and 10^{5}, thus there are nine orders of magnitude between the minimum and the maximum magnitude.

Finally, in Figure 10 we show the evolution of the constraint norm over the augmented Lagrangian iterations. The full minimization process using the augmented Lagrangian method has taken nine iterations. The initial constraint norm is equal to four, and in the first iteration it is reduced to 10^{-5}. Then, in the following iterations, the optimization process further reduces the norm of the constraint up to 10^{-6}.

5. Conclusions

In this work we have shown a method to gradually impose the boundary displacement condition in a mesh morphing technique. The proposed method is formulated in terms of a constrained minimization problem. Specifically,

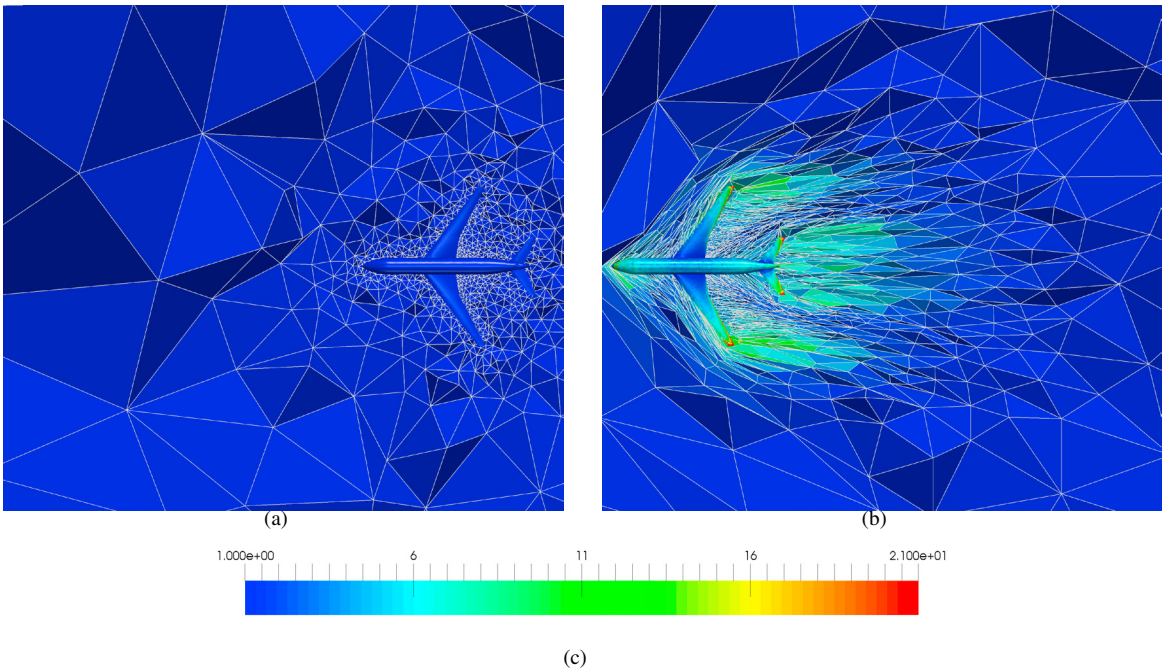

Fig. 8. Detail of the aircraft mesh colored by distortion: (a) initial, (b) optimized, and (c) distortion color legend.

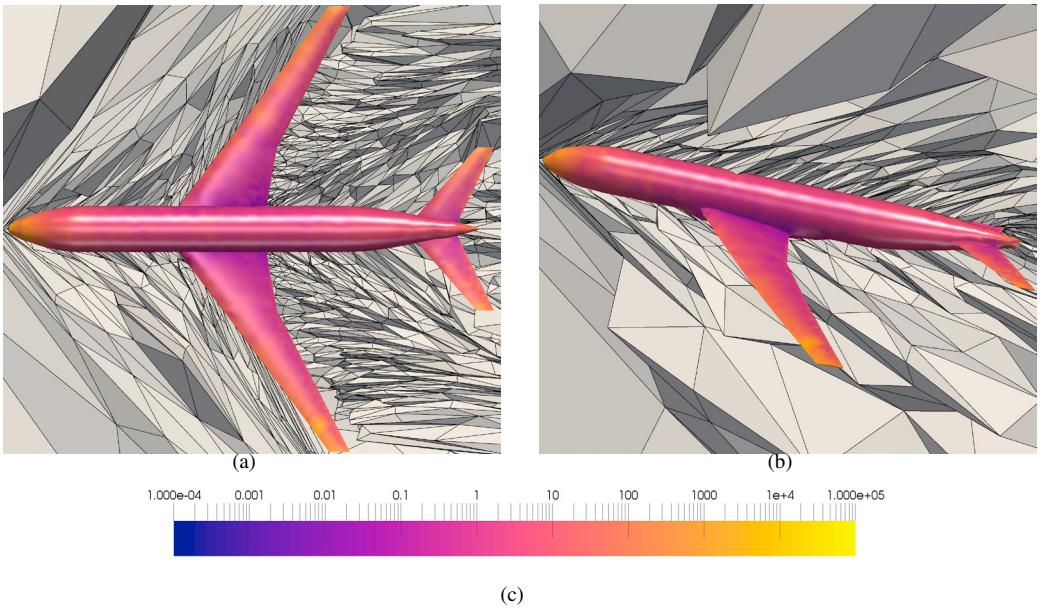

Fig. 9. Detail of the optimized non-uniform size mesh of an aircraft and surface colored by Lagrange multiplier magnitude: (a) top view, (b) perspective view, (c) and logarithmic scale color legend for the Lagrange multiplier magnitude.

we minimize the mesh distortion, constrained to the position of the boundary nodes. To solve the constrained minimization problem, we use the augmented Lagrangian method. Thus, we obtain an approximation of the Lagrangian multipliers associated to the boundary constraint, and we approximate them using polynomials of the same degree as the element-wise representation of physical mesh. We show that the mesh morphing technique does not produce

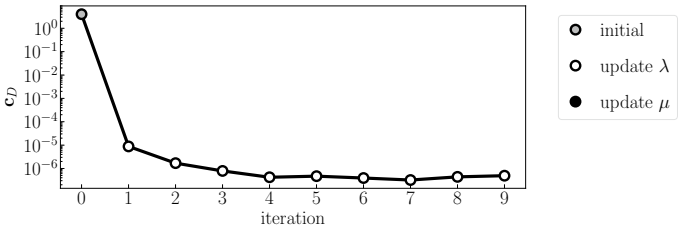

Fig. 10. Evolution of the constraint norm over the augmented Lagrangian iterations for the mesh moving of an aircraft.

Table 2. Evolution of the mesh quality during the iterations of the augmented Lagrangian optimization process, and the execution time for each example.

example case	relative mesh quality										time (s)
circle displacement	1.0	0.937	0.936	0.936	0.936	0.936	0.936				0.73
mesh curving	1.0	0.999	0.999	0.999	0.999	0.999	0.999				26.9
aircraft displacement	1.0	0.998	0.998	0.998	0.998	0.998	0.998	0.998	0.998	0.998	476.25

inverted elements at any stage of the process and thus, the optimization does not need to feature untangling capabilities. We have applied the proposed formulation to mesh moving and mesh curving problems. The proposed method is able to handle large displacements for 2D and 3D meshes with non-uniform sizing. Moreover, it can also handle the curving of highly stretched high-order meshes.

The differences in the evolution of the augmented Lagrangian methodology are originated by the different nature of the proposed examples. In general terms larger displacements might induce more updates of the constraint penalty. This is only the case of the first example which features the largest imposed displacement. In the other examples the boundary constraint is fulfilled in the first iterations and thus, the process is mainly driven to update the Lagrange multipliers.

We have shown in the examples that the constraint is reproduced with high accuracy, even though the nodes are not exactly placed on the target position. Nevertheless, it is possible to *project* the nodes on the target position, although the error of reproducing the constraint would increase. That is, interpolative conditions feature zero error at the interpolation nodes, and non-zero residual at intermediate points.

To solve the non-linear problem of each augmented Lagrangian solver, we use a backtracking line-search method in which the advancing direction is computed using Newton's method, and the step length using the Wolfe conditions. Note the we are solving the full non-linear problem at each iteration and therefore, we solve a sparse linear system at each Newton iteration. In the near future, we consider to parallelize the code in order to improve the computational efficiency of the proposed method, and be able to apply it to more computationally demanding mesh moving and curving problems.

Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme under grant agreement No 715546. The work of the corresponding author has been partially supported by the Spanish Ministerio de Economía y Competitividad under the personal grant agreement RYC- 2015-01633. The work of the third author has been supported by the Spanish Ministerio de Economía y Competitividad under grant agreement CTM2014-55014-C3-3-R.

References

[1] Matthew L Staten, Steven J Owen, Suzanne M Shontz, Andrew G Salinger, and Todd S Coffey. A comparison of mesh morphing methods for 3d shape optimization. In *Proceedings of the 20th international meshing roundtable*, pages 293–311. Springer, 2011.

[2] S. Sherwin and J. Peiró. Mesh generation in curvilinear domains using high-order elements. *Int. J. Numer. Meth. Eng.*, 53(1):207–223, 2002.

[3] P.-O. Persson and J. Peraire. Curved mesh generation and mesh refinement using lagrangian solid mechanics. In *Proc. 47th AIAA*, 2009.

[4] Z. Xie, R. Sevilla, O. Hassan, and K. Morgan. The generation of arbitrary order curved meshes for 3D finite element analysis. *Comput. Mech.*, 51:361–374, 2012.
[5] P. M. Knupp. Algebraic mesh quality metrics. *SIAM J. Numer. Anal.*, 23(1):193–218, 2001.
[6] J. M. Escobar, E. Rodríguez, R. Montenegro, G. Montero, and J. M. González-Yuste. Simultaneous untangling and smoothing of tetrahedral meshes. *Comput. Meth. Appl. Mech. Eng.*, 192(25):2775–2787, 2003.
[7] Christian Schüller, Ladislav Kavan, Daniele Panozzo, and Olga Sorkine-Hornung. Locally injective mappings. In *Computer Graphics Forum*, volume 32, pages 125–135. Wiley Online Library, 2013.
[8] Shahar Z Kovalsky, Noam Aigerman, Ronen Basri, and Yaron Lipman. Large-scale bounded distortion mappings. *ACM Trans. Graph.*, 34(6):191–1, 2015.
[9] A. Gargallo-Peiró, X. Roca, J. Peraire, and J. Sarrate. Distortion and quality measures for validating and generating high-order tetrahedral meshes. *Engineering with Computers*, 31(3):423–437, 2015.
[10] E. Ruiz-Gironés, X. Roca, and J. Sarrate. High-order mesh curving by distortion minimization with boundary nodes free to slide on a 3D CAD representation. *Computer-Aided Design*, 72:52–64, 2016.
[11] T. Toulorge, C. Geuzaine, J.-F. Remacle, and Jonathan Lambrechts. Robust untangling of curvilinear meshes. *J. Comput. Phys.*, 254:8 – 26, 2013.
[12] T. Toulorge, J. Lambrechts, and J.F. Remacle. Optimizing the geometrical accuracy of curvilinear meshes. *Journal of Computational Physics*, 2016.
[13] M. Fortunato and P.E. Persson. High-order unstructured curved mesh generation using the winslow equations. *Journal of Computational Physics*, 307:1–14, 2016.
[14] D. Moxey, D. Ekelschot, Ü. Keskin, S.J. Sherwin, and J. Peiró. High-order curvilinear meshing using a thermo-elastic analogy. *Computer-Aided Design*, 72:130–139, 2016.
[15] Michael Rabinovich, Roi Poranne, Daniele Panozzo, and Olga Sorkine-Hornung. Scalable locally injective mappings. *ACM Transactions on Graphics (TOG)*, 36(2):16, 2017.
[16] V.A. Garanzha and I.E. Kaporin. Regularization of the barrier variational method. *Computational mathematics and mathematical physics*, 39(9):1426–1440, 1999.
[17] J. Nocedal and S. Wright. *Numerical optimization*. Springer Verlag, 1999.
[18] A. Gargallo-Peiró, X. Roca, J. Peraire, and 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):342–363, 2015.
[19] A. Kelly, L. Kaczmarczyk, and C.J. Pearce. Mesh Improvement Methodology for 3D Volumes with Non-Planar Surfaces. In *Proc. 21st Int. Meshing Roundtable*, 2011.
[20] L Liu, Y Zhang, T J R Hughes, M A Scott, and T W Sederberg. Volumetric t-spline construction using boolean operations. *Engineering with Computers*, pages 1–15, 2013.
[21] E. Ruiz-Gironés, J. Sarrate, and X. Roca. Defining an $\mathcal{L}_2$-disparity measure to check and improve the geometric accuracy of non-interpolating curved high-order meshes. *Procedia Engineering*, 124:122–134, 2015.
[22] E. Ruiz-Gironés, J. Sarrate, and X. Roca. Generation of curved high-order meshes with optimal quality and geometric accuracy. *Procedia Engineering*, 163:315–327, 2016.
[23] Roman Poya, Ruben Sevilla, and Antonio J Gil. A unified approach for a posteriori high-order curved mesh generation using solid mechanics. *Computational Mechanics*, 58(3):457–490, 2016.
[24] L. Liu, C. L. Tai, Z. Ji, and G. Wang. Non-iterative approach for global mesh optimization. *Comput. Aided Design*, 39(9):772–782, 2007.
[25] X. Roca, A. Gargallo-Peiró, and J. Sarrate. Defining quality measures for high-order planar triangles and curved mesh generation. In *Proc. 20th Int. Meshing Roundtable*, pages 365–383. Springer International Publishing, 2012.
[26] P. M. Knupp. Algebraic mesh quality metrics for unstructured initial meshes. *Finite Elem. Anal. Des.*, 39(3):217–241, 2003.
[27] A. Gargallo-Peiró. *Validation and generation of curved meshes for high-order unstructured methods*. PhD thesis, Universitat Politècnica de Catalunya, 2014.
[28] A. Gargallo-Peiró, X. Roca, J. Peraire, and J. Sarrate. Inserting curved boundary layers for viscous flow simulation with high-order tetrahedra. In *Research Notes, 22nd Int. Meshing Roundtable*. Springer International Publishing, 2013.
[29] A. Gargallo-Peiró, X. Roca, J. Peraire, and J. Sarrate. A distortion measure to validate and generate curved high-order meshes on cad surfaces with independence of parameterization. *International Journal for Numerical Methods in Engineering*, 2015.
[30] J. M. Escobar, G. Montero, R. Montenegro, and E. Rodríguez. An algebraic method for smoothing surface triangulations on a local parametric space. *Int. J. Numer. Meth. Eng.*, 66(4):740–760, 2006.
[31] Eloi Ruiz-Gironés, Xevi Roca, Josep Sarrate, Rafael Montenegro, and José María Escobar. Simultaneous untangling and smoothing of quadrilateral and hexahedral meshes using an object-oriented framework. *Advances in Engineering Software*, 80:12–24, 2015.
[32] A. Gargallo-Peiró, X. Roca, and J. Sarrate. A surface mesh smoothing and untangling method independent of the CAD parameterization. *Comput. Mech.*, 53(4):587–609, 2014.
[33] A. Gargallo-Peiró, X. Roca, J. Peraire, and J. Sarrate. Distortion and quality measures for validating and generating high-order tetrahedral meshes. *Eng. Comput.*, 31:423–437, 2015.