High quality geometric meshing of CAD surfaces

P. Laug[1] and H. Borouchaki[2]

[1] INRIA Paris-Rocquencourt, France, patrick.laug@inria.fr
[2] UTT, Troyes, France, houman.borouchaki@utt.fr

Summary. A wide range of surfaces can be defined by means of composite parametric surfaces as is the case for most CAD modelers. There are, essentially, two approaches to meshing parametric surfaces: direct and indirect. Popular direct methods include the octree-based method, the advancing-front-based method and the paving-based method working directly in the tridimensional space. The indirect approach consists in meshing the parametric domain and mapping the resulting mesh onto the surface. Using the latter approach, we propose a general "geometry accurate" mesh generation scheme using geometric isotropic or anisotropic metrics. In addition, we introduce a new methodology to control the mesh gradation for these geometric meshes in order to obtain finite element geometric meshes. Application examples are given to show the pertinence of our approach.

Keywords: parametric surface meshing, curve discretization, anisotropic meshing, mesh gradation, geometric meshes.

1 Introduction

Surface meshing is involved in many numerical fields which include the finite element method. It is a necessary step when one wants to construct the mesh of a solid domain in three dimensions. Generally, isotropic meshes are used in solid mechanics while anisotropic meshes are preferred in CFD (computational fluid dynamics) as directional fields must be captured. A wide range of surfaces can be defined by means of composite parametric surfaces. Most of the surfaces are approximated by polynomial or rational parametric patches as is the case for most CAD modelers. In this case, the indirect approach (consisting in meshing the parametric domain and mapping the resulting mesh onto the surface) is conceptually straightforward as a planar mesh is generated in the parametric domain. In this paper, we are interested in generating geometry preserving meshes called geometric meshes for finite element computation.

Despite its simplicity, the problem with the indirect approach is the generation of a mesh which conforms to the metric of the surface. Historically, people were initially interested in surface visualization using this indirect approach. In fact, they aimed to minimize the error in the polyhedral approximation of the surface indirectly in the parametric space without paying attention to the quality of the resulting mesh [1, 2, 3, 4]. The mesh in the parametric surface is usually anisotropic, due to the metric deformation from the surface to its parametric domain. Thus, for people in finite element computation, the problem is reduced to the generation of an anisotropic mesh in the parametric domain. To this end, various algorithms are proposed [5, 6, 7, 8]. In addition, one can control explicitly the accuracy of a generated element with respect to the geometry of the surface if careful attention is paid. Indeed, a mesh of a parametric patch whose element vertices belong to the surface is "geometrically" suitable if all mesh elements are close to the surface and if every mesh element is close to the tangent planes related to its vertices. A mesh satisfying these properties is called a geometric mesh. The first property allows us to bound the gap between the elements and the surface. This gap measures the largest distance between an element (any point of the element) and the surface. The second property ensures that the surface is locally of order G^1 in terms of continuity. To obtain this, the angular gap between the element and the tangent plane at its vertices must be bounded. These properties result in the definition of a mesh metric map depending of surface curvatures called geometric metrics and the goal is to generate a unit mesh (all elements are of unit size with respect to the geometric metrics).

We propose a general scheme of an indirect approach for generating isotropic and anisotropic geometric meshes of a surface constituted by a conformal assembly of parametric patches, based on the concept of metric. The different steps of the scheme are detailed and, in particular, the definition of the geometric metric at each point of the surface (internal to a patch, belonging to an interface or boundary curve, or extremity of such a curve) as well as its corresponding induced metric in parametric domains.

Isotropic or anisotropic geometric metrics can locally produce significant size variations (internal to a patch or across interface curves) and can even be discontinuous along the interface curves. The larger the rate of the mesh size variation, the worse is the shape quality of the resulting mesh. To control this size variation, various methodologies based on metric reduction have been proposed [9] in the case of a continuous isotropic metric. We introduce a novel iterative mesh gradation approach for discontinuous metrics. The approach uses a particular metric reduction procedure in order to ensure the convergence of the gradation process. In particular, we show that in the worst case the anisotropic discontinuous geometric metric map is reduced to a isotropic continuous geometric metric map for which the gradation is controlled.

In Section 2, we introduce and detail the general scheme for meshing composite parametric surfaces. The new mesh gradation control is developed in Section 3. Several application examples are provided in Section 4 to illustrate the capabilities of the proposed method. Finally, in the last section, we conclude with a few words about the prospects.

2 Methodology

A surface Σ composed of parametric patches is defined by a collection of surface patches Σ_i fitted together in a "conforming" manner (see equation (6) below) and verifying:

$$\Sigma = \bigcup_i \Sigma_i \,, \quad \Sigma_i = \sigma_i(\Omega_i) \tag{1}$$

where Ω_i is a domain of $\mathbb{R}^2$ (parametric domain) and σ_i is a C^1 continuous application:

$$\sigma_i : \Omega_i \subset \mathbb{R}^2 \to \Sigma_i \subset \mathbb{R}^3 \,, \quad \begin{pmatrix} u \\ v \end{pmatrix} \mapsto \sigma_i(u,v) \in \mathbb{R}^3 \tag{2}$$

Each domain Ω_i is defined by its contour, closed and non self-intersecting, constituted by a collection of contiguous curve segments γ_{ij} in $\mathbb{R}^2$:

$$\overline{\Omega_i} = \bigcup_j \gamma_{ij} \,, \quad \gamma_{ij} = \omega_{ij}([a_{ij}, b_{ij}]) \tag{3}$$

with

$$\omega_{ij} : [a_{ij}, b_{ij}] \subset \mathbb{R} \to \gamma_{ij} \subset \mathbb{R}^2 \,, \quad t \mapsto \omega_{ij}(t) \in \mathbb{R}^2 \tag{4}$$

thus verifying

$$\gamma_{ij} \cap \gamma_{ik} \quad = \quad \emptyset \quad \text{or} \quad e_{il} \tag{5}$$

where $\emptyset$ denotes the empty set and e_{il} a common extremity of curve segments γ_{ij} and γ_{ik}.

Surface Σ is conforming if and only if:

$$\Sigma_i \cap \Sigma_j \quad = \quad \emptyset \quad \text{or} \quad \bigcup_k E_{ij,k} \quad \text{or} \quad \bigcup_k \Gamma_{ij,k} \tag{6}$$

where $\exists\, l, m$ such that $E_{ij,k} = \sigma_i(e_{il}) = \sigma_j(e_{jm})$
and $\exists\, l, m$ such that $\Gamma_{ij,k} = \sigma_i(\gamma_{il}) = \sigma_j(\gamma_{jm})$.

Therefore, $\Gamma_{ij,k}$ is a boundary curve segment shared by Σ_i and Σ_j, image of two boundary curve segments γ_{il} of Ω_i and γ_{jm} of Ω_j. Thus, by considering common curve segments only once, we obtain:

$$\bigcup_i \overline{\Sigma_i} \quad = \quad \bigcup_j \Gamma_j \tag{7}$$

where

$$\Gamma_j \cap \Gamma_k \quad = \quad \emptyset \quad \text{or} \quad E_{jk} \tag{8}$$

and there exists a set of indices (i, k) such that each Γ_j equals $\sigma_i(\gamma_{ik})$.

We suppose in the following that Σ is conforming (see [10] for setting the conformity of any surface). The generation of a mesh of Σ following an indirect approach is given by the following general scheme:

1. Specification of a size map or metric map associated with points of Σ.
2. Discretization of each Γ_j.
3. Transfer of the discretization of each Γ_j onto corresponding segments γ_{ik}.
4. Mesh generation of each Ω_i from the discretization of its boundary (obtained in the previous step).
5. Mapping the mesh of each Ω_i onto Σ_i.
6. Construction of the mesh of Σ from meshes of Σ_i.

These different steps are detailed in the following (for further information, see references cited in each step):

2.1 Size map or metric map

Within a classical framework, mainly two categories of size maps or metric maps can be considered. *The first category* concerns uniform meshes with a given constant size h or a given constant metric $\mathcal{M} = \frac{1}{h^2}\,\mathcal{I}_3$ (the size specification results in a given metric and a mesh complying with this size is a mesh whose edge length equals unity in this metric). The advantage of this kind of meshing is that it provides, in general, equilateral meshes. On the other hand, it cannot guarantee a good representation of the geometry of the domain for a given size. *The second category* concerns meshes referred to as geometric, adapted to the geometry of the patches composing the surface. To define the size or the metric at a given point of the surface, three cases are discussed hereafter: internal point, interface or boundary point and extremity point.

Internal point. An internal point P is a point belonging to the interior of a patch Σ_i. In an isotropic framework, it can be demonstrated that locally the geometric size at P must be proportional to the minimal radius of curvature $\rho_1(P)$ of patch Σ_i [11]:

$$\mathcal{M}_{iso}(\Sigma_i, P) = \frac{1}{h_1^2(P)}\,\mathcal{I}_3 \quad \text{with} \quad h_1(P) = \lambda_1\,\rho_1(P) \tag{9}$$

where $\lambda_1 = 2\,\sin\theta$, θ being the maximum angle between an element and tangent planes to the surface, or equivalently $\lambda_1 = 2\,\sqrt{\varepsilon\,(2-\varepsilon)}$, ε being the maximum relative distance between an element and the surface. In an anisotropic framework, the metric can also be deduced from the principal radii of curvature ($\rho_1(P) < \rho_2(P)$) and the principal directions of curvature (defined by two orthogonal unit vectors $\overrightarrow{v_1}(P)$ and $\overrightarrow{v_2}(P)$) of patch i [12]:

$$\mathcal{M}_{aniso}(\Sigma_i, P) = \begin{pmatrix} \overrightarrow{v_1}(P) & \overrightarrow{v_2}(P) \end{pmatrix} \begin{pmatrix} \frac{1}{h_1^2(P)} & 0 \\ 0 & \frac{1}{h_2^2(P)} \end{pmatrix} \begin{pmatrix} \overrightarrow{v_1}(P)^T \\ \overrightarrow{v_2}(P)^T \end{pmatrix} \tag{10}$$

with $h_1(P) = \lambda_1\,\rho_1(P)$ and $h_2(P) = \lambda_2\,\rho_2(P)$, where λ_1 can be defined again by $\lambda_1 = 2\,\sqrt{\varepsilon\,(2-\varepsilon)}$ and λ_2 is a smaller coefficient given by $\lambda_2 = 2\,\sqrt{\varepsilon\,\frac{\rho_1}{\rho_2}\,(2-\varepsilon\,\frac{\rho_1}{\rho_2})}$. The above anisotropic geometric metric is degenerate since the size is not defined in the direction orthogonal to the plane containing $\overrightarrow{v_1}$ and $\overrightarrow{v_2}$. In order to obtain a well-defined metric consistent with the isotropic case, we redefine the anisotropic geometric as:

$$\mathcal{M}_{aniso}(\Sigma_i, P) = \left(\overrightarrow{v_1}(P)\ \overrightarrow{v_2}(P)\ \overrightarrow{n}(P) \right) \begin{pmatrix} \frac{1}{h_1^2(P)} & 0 & 0 \\ 0 & \frac{1}{h_2^2(P)} & 0 \\ 0 & 0 & \frac{1}{h_1^2(P)} \end{pmatrix} \begin{pmatrix} \overrightarrow{v_1}(P)^T \\ \overrightarrow{v_2}(P)^T \\ \overrightarrow{n}(P)^T \end{pmatrix} \tag{11}$$

where $\overrightarrow{n}(P)$ is the unit normal to the surface at the considered point. In practice, the sizes in the above metrics are bounded by specified minimal and maximal size values and thus these metrics are always well defined.

The defined geometric metrics allows us to bound by a specified threshold the angular deviation θ of each element with respect to the tangent planes at its vertices. The Hausdorff distance between each element and the surface can be expressed by these angular deviations. To bound this distance by a threshold value, it is sufficient to consider the related angular deviation and thus the corresponding geometric metric. In this case, the angular deviation θ depends on the considered vertex.

Interface or boundary point. An interface or boundary point C is a point belonging to the interior of a curve segment Γ_j. For an interface point, curve Γ_j is shared by at least two patches while for an boundary point, curve Γ_j belongs to only one patch. Let us denote by $\{\Sigma_{ij}\}$ the set of patches containing Γ_j. The geometric size at C depends on the geometric size of each Σ_{ij} and also the geometric size of curve Γ_j. If $\rho(C)$ is the radius of curvature of curve Γ_j at C, the geometric size of curve Γ_j is defined by:

$$\mathcal{M}(\Gamma_j, C) = \frac{1}{h^2(C)}\, \mathcal{I}_3 \quad \text{with} \quad h(C) = \lambda_1\, \rho(C)\,. \tag{12}$$

Hence, at an interface or boundary point C, several geometric metrics are defined ($\mathcal{M}_{iso}(\Sigma_{ij}, C)$ or $\mathcal{M}_{aniso}(\Sigma_{ij}, C)$ and $\mathcal{M}(\Gamma_j, C)$).

Extremity point. An extremity point E is a common extremity of a set of curves $\{\Gamma_j\}$. Each $\{\Gamma_j\}$ belongs to a set of patches $\{\Sigma_{ij}\}$. Therefore, the geometric size at E depends on the geometric size of each curve Γ_j and the geometric size of corresponding patches Σ_{ij}. Similarly, at an extremity point E, several geometric metrics ($\mathcal{M}_{iso}(\Sigma_{ij}, E)$ or $\mathcal{M}_{aniso}(\Sigma_{ij}, E)$ for all i, j such that Σ_{ij} contains a curve $\{\Gamma_j\}$ with E as extremity and $\mathcal{M}(\Gamma_j, E)$ for all j such that E is an extremity of Γ_j) are defined.

Remark: size variation. The problem with this kind of meshing (geometric meshing) is that it can produce a very important variation of the size according to the variation of curvature. The shape quality of the elements largely depends of the size variation underlying in the metric field. To remedy this, it is sufficient to modify the metric field according to the desired size variation. To control the latter, methods of size smoothing or mesh gradation control can be considered. This issue is detailed in the next section.

2.2 Discretization of curve segments Γ_j

The discretization of each curve segment consists in subdividing the curve by curve segments of unit length with respect to a specified isotropic metric function. For each point C of a curve, this metric length is obtained regarding the metric at the point C in the direction of the tangent to the curve. In the geometric case, as mentioned above, several metrics are defined ($\mathcal{M}_{iso}(\Sigma_{ij}, C)$ or $\mathcal{M}_{aniso}(\Sigma_{ij}, C)$ on adjacent patches, and $\mathcal{M}(\Gamma_j, C)$ on the curve). Thus the "metric length" at C is the minimum length specified by these metrics in the direction of the tangent at C to the curve. To compute the length of a curve segment with respect to a metric, a polyline approximating the curve is constructed and the length of this polyline is calculated (this length computation allows us to subdivide the curve by segments of unit length).

2.3 Inverse mapping of the discretization of Γ_j in parametric domains

The discretization of Γ_j is defined by a set of vertices ordered by their curvilinear abscissae. This discretization is mapped back to the corresponding curve segments γ_{ik} in parametric domains. The discretization of all curve segments γ in the parametric domains being well defined, the corresponding metrics in parametric domains must now be provided. These bidimensional metrics will be calculated from metrics in the tridimensional space that are defined in the following.

For an interface or boundary point C of a curve segment Γ_j belonging to a given patch Σ_{ij}, the metric $\mathcal{M}_{iso}(\Sigma_{ij}, C)$ or $\mathcal{M}_{aniso}(\Sigma_{ij}, C)$ is shrunk to fit the metric length at C in the direction of the tangent to the curve giving the new geometric metric $\overline{\mathcal{M}}_{iso}(\Sigma_{ij}, C)$ or $\overline{\mathcal{M}}_{aniso}(\Sigma_{ij}, C)$. For an extremity point E of a patch Σ_{ij}, the same procedure is applied considering each interface or boundary curve Γ_j of Σ_{ij} such that E is an extremity of Γ_j leading to different geometric metrics $\overline{\mathcal{M}}_{iso}(\Sigma_{ij}, E)$ or $\overline{\mathcal{M}}_{aniso}(\Sigma_{ij}, E)$ and we consider the new geometric metric at E the metric $\overline{\overline{\mathcal{M}}}_{iso}(\Sigma_{ij}, E)$ or $\overline{\overline{\mathcal{M}}}_{aniso}(\Sigma_{ij}, E)$ giving the smallest size along the tangent direction at each curve Γ_j. Thus for an extremity point E, the geometric metric with respect to a patch Σ_{ij} is such that the minimal metric length at E is satisfied.

As an ilustration, Fig. 1 (left) shows an interface curve Γ shared by two patches Σ_1 and Σ_2. Using the previous notations, Γ is in fact equal to a curve Γ_j, and Σ_1 (resp. Σ_2) is equal to a patch $\Sigma_{i_1 j}$ (resp. $\Sigma_{i_2 j}$. As explained in section 2.2, the metric length at a point C belonging to the interior of Γ is the minimum length specified by the three metrics $\mathcal{M}_1 = \mathcal{M}_{aniso}(\Sigma_{i_1 j}, C)$, $\mathcal{M}_2 = \mathcal{M}_{aniso}(\Sigma_{i_2 j}, C)$ and $\mathcal{M}(\Gamma_j, C)$ in the direction of the tangent τ at C to the curve. In this example, the minimum length l_{min} is given by the latter metric. Consequently, the shrunk metrics $\overline{\mathcal{M}}_1 = \overline{\mathcal{M}}_{aniso}(\Sigma_{i_1 j}, C)$ and $\overline{\mathcal{M}}_2 = \overline{\mathcal{M}}_{aniso}(\Sigma_{i_2 j}, C)$ are represented by ellipsoids centered at C and passing through a same point of τ at a distance l_{min} of C.

On Fig. 1 (right), an extremity E is shared by two curves $\Gamma_1 = \Gamma_{j_1}$ and $\Gamma_2 = \Gamma_{j_2}$ at the boundary of patch $\Sigma = \Sigma_{ij_1} = \Sigma_{ij_2}$. The previous process gives one point on tangent τ_1 to Γ_1 and a second point on tangent τ_2 to Γ_2, and the corresponding metrics $\overline{\mathcal{M}}_1 = \overline{\mathcal{M}}_{aniso}(\Sigma_{ij_1}, E)$ and $\overline{\mathcal{M}}_2 = \overline{\mathcal{M}}_{aniso}(\Sigma_{ij_2}, E)$. In this new example, the minimal metric length is given by the second metric and thus the geometric metric at E with respect to patch Σ is $\overline{\overline{\mathcal{M}}} = \overline{\mathcal{M}}_2$ or $\overline{\overline{\mathcal{M}}}_{aniso}(\Sigma, E) = \overline{\mathcal{M}}_{aniso}(\Sigma_{ij_2}, E)$.

Fig. 1. Left: anisotropic metrics at an interface point C belonging to the interior of a curve segment Γ shared by two patches Σ_1 and Σ_2. Right: anisotropic metrics at a common extremity E of two curves Γ_1 and Γ_2 bounding a patch Σ.

2.4 Mesh generation of domains Ω_i

We use an indirect method for meshing general parametric surfaces conforming to a pre-specified metric map $\mathcal{M}_3$ (for more details, see [11]). Let Σ be such a surface parameterized by:

$$\sigma : \Omega \longrightarrow \Sigma, \qquad (u, v) \longmapsto \sigma(u, v)\,, \tag{13}$$

where Ω denotes the parametric domain. The Riemannian metric specification $\mathcal{M}_3$ gives the unit measure in any direction. In the geometric case this metric is defined as:

- internal point: $\mathcal{M}_{iso}(\Sigma_i)$ or $\mathcal{M}_{aniso}(\Sigma_i)$.
- interface or boundary point: $\overline{\mathcal{M}}_{iso}(\Sigma_i)$ or $\overline{\mathcal{M}}_{aniso}(\Sigma_i)$.
- extremity point: $\overline{\overline{\mathcal{M}}}_{iso}(\Sigma_i)$ or $\overline{\overline{\mathcal{M}}}_{aniso}(\Sigma_i)$.

The goal is to generate a mesh of Σ such that the edge lengths are equal to one with respect to the related Riemannian space (such meshes being referred to as "unit" meshes). Based on the intrinsic properties of the surface, namely the first fundamental form:

$$\mathcal{M}_\sigma = \begin{pmatrix} \sigma_u^T\,\sigma_u & \sigma_u^T\,\sigma_v \\ \sigma_v^T\,\sigma_u & \sigma_v^T\,\sigma_v \end{pmatrix} , \tag{14}$$

the Riemannian structure $\mathcal{M}_3$ is induced into the parametric space as follows:

$$\widetilde{\mathcal{M}}_2 = \begin{pmatrix} \sigma_u^T \\ \sigma_v^T \end{pmatrix} \mathcal{M}_3 \begin{pmatrix} \sigma_u & \sigma_v \end{pmatrix} . \tag{15}$$

The above equation is the product of three matrices respectively of order 2×3, 3×3 and 3×2, resulting in a metric of order 2×2 in the parametric domain.

Even if the metric specification $\mathcal{M}_3$ is isotropic, the induced metric in parametric space is in general anisotropic, due to the variation of the tangent plane along the surface. Finally, a unit mesh is generated completely inside the parametric space such that it conforms to the induced metric $\widetilde{\mathcal{M}}_2$. This mesh is constructed using a combined advancing-front – Delaunay approach applied within a Riemannian context: the field points are defined after an advancing front method and are connected using a generalized Delaunay type method.

This method is efficient if the metric $\mathcal{M}_\sigma$ of the first fundamental form of the surface is well defined and its variation is bounded. If this is not the case, one can consider the metric in the vicinity of the degenerated points.

2.5 Mapping back the mesh of each Ω_i onto Σ_i

The mesh of each Σ_i is constituted by vertices, images by σ_i of the vertices of the mesh of Ω_i, keeping the same connectivity. This methodology is functional if the tangent plane metric does not involve strong variations (i.e., the image of an edge of the mesh of the parametric domain is close to the straight segment joining the images of its extremities).

2.6 Construction of the mesh of Σ from meshes of Σ_i

The global mesh of Σ is obtained by gathering all the meshes of patches Σ_i. In this process, vertices of the discretizations of the boundary curves must not be duplicated.

3 Mesh gradation

The metric $\mathcal{M}_3$ can locally produce important size variations, in particular in the present context of geometric mesh generation. These size variations entail a generation of elements having a poor shape quality. To remedy this, metric $\mathcal{M}_3$ can be modified while accounting for the size constraints at best and while controlling the underlying gradation, which measures the size variation in the vicinity of a vertex [9]. The general scheme of the mesh gradation methodology has several steps:

1. Generation of the initial geometric mesh, controlled by the geometric metrics detailed in the previous section.
2. Computation of the geometric metrics at the vertices of this mesh.
3. Modification of these metrics in order to bound the gradation by a specified threshold c_{goal}.
4. Generation of an adapted mesh controlled by the modified metrics.
5. Repetition of the three steps 2–4, once or several times.

The purpose of the step repetition is to accurately capture the surface geometry, and in practise it is applied only once. In the following, for each case of isotropic or anisotropic geometric metrics, the step 3 is detailed and the corresponding algorithm is given.

3.1 Isotropic geometric metrics

As indicated in section 2.3, at the end of step 2 of the general scheme, the geometric metric at each vertex of the mesh is defined by:

- If the vertex is a point P belonging to the interior of a patch Σ_i, its metric is unique and defined by $\mathcal{M}_{iso}(\Sigma_i, P)$.
- If the vertex is a point C belonging to the interior of a curve segment Γ_j, several metrics $\overline{\mathcal{M}}_{iso}(\Sigma_{ij}, C)$ are defined. However, in the case of isotropic geometric metrics, these metrics are identical for all patches $\Sigma_{i,j}$ because all these isotropic metrics give the same length in the direction of the tangent to Γ_j. This common metric is then denoted by $\overline{\mathcal{M}}_{iso}(\Sigma_{*j}, C)$.
- If the vertex is a point E being a common extremity of a set of curves $\{\Gamma_j\}$, several metrics $\overline{\mathcal{M}}_{iso}(\Sigma_{*j}, E)$ corresponding to every $\{\Gamma_j\}$ are defined. Among these metrics, there exists a metric denoted by $\overline{\mathcal{M}}_{iso}(\Sigma_{**}, E)$ which gives the smallest length in all directions. The latter metric is taken into account in the gradation control methodology.

The modification of the geometric metrics consists in locally modifying these metrics by considering the size variation on each edge of the mesh. For each edge, the modification includes two successive steps – the calculation of the shock and, if necessary, a metric update – which are detailed below.

Calculation of the shock. Let PQ be an edge, and let $\mathcal{M}(P)$ and $\mathcal{M}(Q)$ be the metrics at its extremities. If $h(P)$ and $h(Q)$ respectively represent the sizes specified by these metrics (in all directions and in particular in the direction of vector $\overrightarrow{PQ}$), let us assume without loss of generality that $h(P) \leq h(Q)$. The H-shock (or more simply the shock) $c(PQ)$ related to the edge PQ is the value:

$$c(PQ) = \left(\frac{h(Q)}{h(P)}\right)^{1/l(PQ)} \tag{16}$$

where $l(PQ)$ is the length of edge PQ in a metric interpolating the size given by the two extremity metrics $\mathcal{M}(P)$ and $\mathcal{M}(Q)$ in direction $\overrightarrow{PQ}$:

$$l(PQ) = ||\overrightarrow{PQ}|| \int_0^1 \frac{1}{h(P + t\,\overrightarrow{PQ})}\, dt \tag{17}$$

Metric update. If the shock $c(PQ)$ is greater than the given threshold c_{goal}, then the size $h(Q)$ is multiplied by η, or equivalently the metric $\mathcal{M}(Q)$ is divided by η^2, where η is a size reduction factor given by:

$$\eta = \left(\frac{c_{goal}}{c(PQ)}\right)^{l(PQ)} \quad < \quad 1 \tag{18}$$

General algorithm. Using the above notations, the gradation algorithm for isotropic metrics can be written in simplified pseudo-code as shown on Fig. 2. Its inputs are the mesh, the geometric metrics $\mathcal{M}$ at the mesh vertices, and the threshold c_{goal}. The outer loop runs until $c_{max} \leq c_{goal}$, where c_{max} is the maximum shock on all the edges. Consequently, in output, metrics are modified so that the gradation is bounded by the given threshold c_{goal}.

```
Input: mesh, M_iso, c_goal
Repeat {
    c_max = 0
    For each edge PQ of the mesh {
        Compute c(PQ), the shock on PQ
        If (c(PQ) > c_goal) update M_iso(Q)
        c_max = max(c_max, c(PQ))
    }
} until (c_max ≤ c_goal)
Output: M_iso,gra = M_iso
```

Fig. 2. Gradation algorithm in the isotropic case.

3.2 Anisotropic geometric metrics

In the anisotropic case, the geometric metrics at each vertex of the mesh are defined as follows:

- If the vertex is a point P belonging to the interior of a patch Σ_i, its metric is unique and defined by $\mathcal{M}_{aniso}(\Sigma_i, P)$.
- If the vertex is a point C belonging to the interior of a curve segment Γ_j, several metrics $\overline{\mathcal{M}}_{aniso}(\Sigma_{ij}, C)$ are defined. Indeed, the anisotropic metric is discontinuous at C.
- If the vertex is a point E being a common extremity of a set of curves $\{\Gamma_j\}$, several metrics $\overline{\mathcal{M}}_{aniso}(\Sigma_{ij}, E)$ corresponding to every $\{\Gamma_j\}$ are defined. In addition, for all patch Σ_{ij} containing E, the metrics $\mathcal{M}_{aniso}(\Sigma_{ij}, E)$ are also considered. Notice also that the metric is discontinuous at E.

Before we give the general algorithm for the gradation of these anisotropic metrics, several points must be clarified concerning the calculation of the shock and the metric update.

Calculation of the shock. Let PQ be an edge of a mesh of a patch Σ_i. For each extremity, for instance point P, are defined a metric $\mathcal{M}(P)$ and a direction $\vec{v}(P)$ as follows:

- If PQ is an internal edge of the mesh, $\vec{v}(P) = \overrightarrow{PQ}$ and there are three possibilities for metric $\mathcal{M}(P)$: if P belongs to the interior of Σ_i, $\mathcal{M}(P) = \mathcal{M}_{aniso}(\Sigma_i, P)$; if P belongs to the interior of a curve Γ_j, $\mathcal{M}(P) = \overline{\mathcal{M}}_{aniso}(\Sigma_{ij}, P)$; if P is an extremity, $\mathcal{M}(P) = \mathcal{M}_{aniso}(\Sigma_i, P)$ independently from any curve Γ_j.
- Otherwise, PQ belongs to the discretization of a curve Γ_j. Direction $\vec{v}(P)$ is given by the tangent to Γ_j at P (see section 2.3) and metric $\mathcal{M}(P)$ is defined by $\overline{\mathcal{M}}_{aniso}(\Sigma_{ij}, P)$.

Denoting by $h(P)$ the size specified by metric $\mathcal{M}(P)$ in direction $\vec{v}(P)$, sizes $h(P)$ and $h(Q)$ are defined at both extremities of PQ and the shock $c(PQ)$ is calculated like in the isotropic case using equations (16) and (17).

Metric update. If the shock $c(PQ)$ is greater than the given threshold c_{goal}, a metric update is necessary. The procedure detailed in section 3.1 for the isotropic case is rather straightforward: assuming that $h(P) < h(Q)$, the metric associated with Q is divided by a factor η^2. In the anisotropic case, this procedure is more complicated, as explained in the following.

Firstly, the metric map is discontinuous along interface curves and thus, as mentioned above, several metrics are associated with a given point. The metric to be updated for a point P is $\mathcal{M}(P)$ whose definition is given above, depending on edge PQ (internal or not) and on point P (internal to a patch, internal to a curve, or extremity). Careful attention must be paid if $\mathcal{M}(P) = \overline{\mathcal{M}}_{aniso}(\Sigma_{ij}, P)$, a metric defined on a curve Γ_j. In this case, the updated metric gives a new metric length in the direction of the tangent to Γ_j at P, and all the patches sharing Γ_j must be updated so that their local metrics give the same metric length at P (as in section 2.3).

A second point is that a simple homothetic reduction of a metric $\mathcal{M}(P)$ does not guarantee the convergence of the gradation process. Indeed, a reduction on one patch implies other reductions on adjacent patches, which may imply a reduction on the first patch, resulting in an endless loop. To avoid this, the key idea is to run beforehand an isotropic gradation defining an isotropic metric at each vertex. The latter is used as a lower limit for the anisotropic gradation. This methodology guarantees the convergence of the process and ensures that a smaller number of elements is generated in the anisotropic case. More precisely, let h_{lim} be the size limit given by the isotropic metric, let $h_1 < h_2$ be the sizes along the principal axes of a metric $\mathcal{M}$, and let η be the size reduction factor. To reduce metric $\mathcal{M}$ with a factor $\eta < 1$, first a homothetic reduction replaces h_1 by $h'_1 = \eta\, h_1$ and h_2 by $h'_2 = \eta\, h_2$. However, if $h'_1 < h_{lim}$, h'_1 is set to h_{lim} and h'_2 is computed so that the size in direction $\overrightarrow{PQ}$ is $h'(P) = \eta\, h(P)$. This procedure is illustrated on Fig. 3. Metric $\mathcal{M}$ is represented by the outer ellipse and h_{lim} is the radius of the inner circle. If η is near 1 then a homothetic reduction is made, but if η becomes smaller the metric becomes "more isotropic". The prior isotropic gradation guarantees that it is never necessary to go below the size limit h_{lim}.

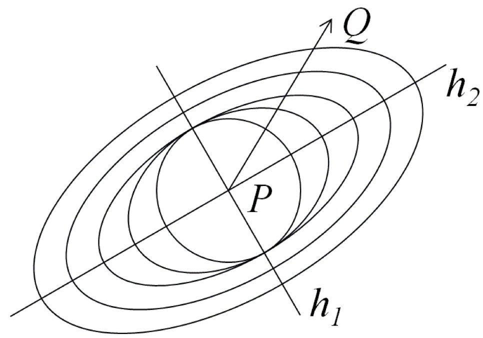

Fig. 3. Reduction of a metric complying with a lower bound h_{lim}.

Thirdly, a problem may occur during the anisotropic gradation process. If a shock $c(PQ) > c_{goal}$ is detected on an edge PQ such that $h(P) < h(Q)$, a metric update at Q may be impossible because the size limit is reached: $h_1 = h_2 = h_{lim}$. This may happen because the edges are not analyzed in the same order in the isotropic and anisotropic gradations. In this case, it is still possible to update the metric at the other extremity P; an iterative procedure finds a new reduction factor $\eta_P < 1$ such that the shock on edge PQ is less than c_{goal}, and metric $\mathcal{M}(P)$ is reduced with this factor η_P.

General algorithm. The gradation algorithm in the anisotropic case is written in simplified pseudo-code on Fig. 4 with inputs and outputs similar to the isotropic case.

4 Application examples

The methodology is implemented in the BLSURF software package [13]. To illustrate our approach for high quality geometric meshing, two examples of CAD surfaces are presented in this section, which represent respectively a crank and a propeller. In each example, the input is an IGES file read by Open Cascade platform and the surface meshes are generated by BLSURF.

Crank. In this first example, we consider a simplified crank. Fig. 5 shows a representation of the CAD model and three isotropic geometric meshes, while

Input: mesh, $\mathcal{M}_{aniso}$, c_{goal}
Run the gradation algorithm in the isotropic case, giving $\mathcal{M}_{iso,gra}$
Repeat {
    $c_{max,1} = 0$
    For each patch Σ_i of the mesh {
        Repeat {
            $c_{max,2} = 0$
            For each edge PQ of patch Σ_i {
                Compute $c(PQ)$, the shock on PQ
                If $(c(PQ) > c_{goal})$ update $\mathcal{M}_{aniso}(Q)$ or else $\mathcal{M}_{aniso}(P)$
                $c_{max,2} = \max(c_{max,2},\, c(PQ))$
            }
            $c_{max,1} = \max(c_{max,1},\, c_{max,2})$
        } until $(c_{max,2} \leq c_{goal})$
    }
} until $(c_{max,1} \leq c_{goal})$
Adjust $\mathcal{M}_{aniso}(E)$, the metrics at the extremities
Output: $\mathcal{M}_{aniso,gra} = \mathcal{M}_{aniso}$

Fig. 4. Gradation algorithm in the anisotropic case.

Fig. 6 shows four anisotropic geometric meshes. Some corresponding statistics are displayed in Table 1. The CAD surface is made up of 12 patches. For all these geometric meshes, an angle $\theta = 2$ degrees is specified, where θ is the maximum angle between each triangle and the underlying tangent planes to the surface (see section 2.1).

The first geometric mesh of Fig. 5 is isotropic, without gradation. As pointed out at the end of section 2.1, elongated triangles can be noticed because of important variations of the surface curvature. To remedy this, a gradation of 2.5 is applied on the metric field, showing an improvement of the shape quality. With a smaller threshold of 1.5, the mesh triangles become almost equilateral. The relative number of elements for these three isotropic meshes is respectively 1.000, 1.118 and 1.287.

To reduce the number of elements with the same geometric accuracy, anisotropic geometric meshes are built. The first mesh of Fig. 6, without gradation, contains less than 9.9 % of the corresponding number of triangles in the isotropic case. Again, a gradation 2.5 (resp. 1.5) has been applied to produce the second (resp. third) mesh. Finally, an even better shape quality can obtained by limiting the aspect ratio of the metrics. The fourth mesh shown is generated with a threshold of 2.5 for the metric aspect ratio. Compared with an isotropic mesh with a same gradation, the relative number of elements is respectively 0.099, 0.157, 0.240 and 0.491 (the latter having a limited anisotropy).

For the last example with 12,424 triangles, the total CPU time on a Dell Precision mobile workstation M6400 at 2.53 GHz is 0.811 seconds. This includes the input of the CAD file, the setting of the topology, the generation of the initial mesh and the two adapted meshes (cf. section 3) and the output of the mesh file. In the first adaptation, the isotropic gradation runs 5 iterations and the anisotropic gradation 2 more iterations, and in the second adaptation the number of iterations is $8+3$, which are executed in a negligible time.

mesh	gradation	aspect ratio	vertices	triangles
iso	∞	∞	9826	19656
iso	2.5	∞	10988	21980
iso	1.5	∞	12644	25292
aniso	∞	∞	969	1942
aniso	2.5	∞	1726	3456
aniso	1.5	∞	3030	6064
aniso	1.5	2.5	6210	12424

Table 1. Crank: meshing statistics.

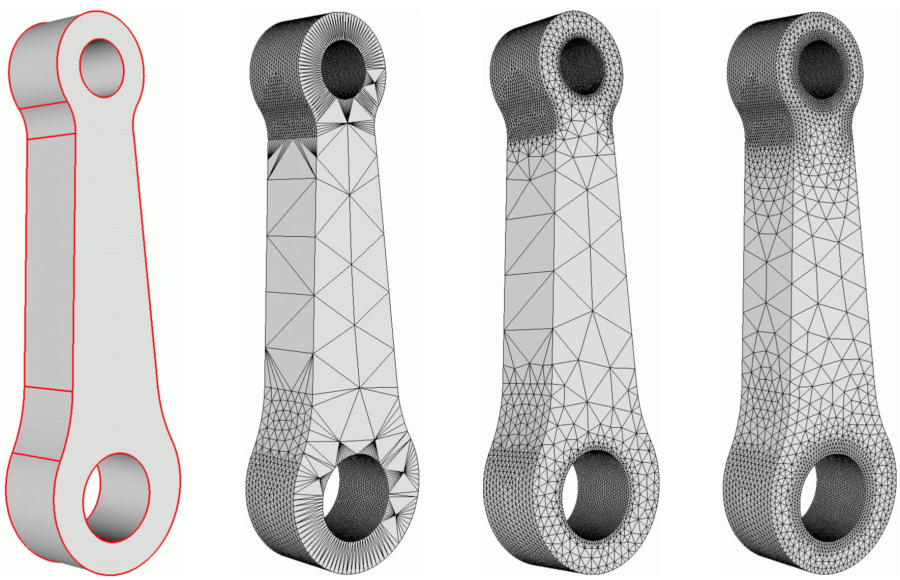

Fig. 5. Crank, from left to right: CAD model, isotropic meshes with gradation ∞, 2.5, and finally 1.5.

Propeller. In this second example, we consider a submarine propeller. Similarly to the previous section, isotropic and anisotropic meshes are shown on Fig. 7. Close up pictures are shown on 8 and meshing statistics are displayed

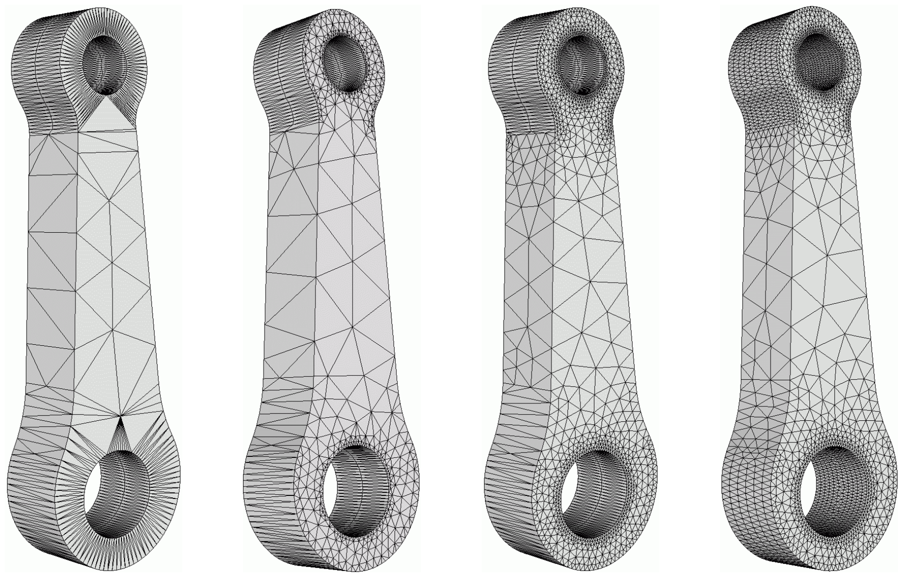

Fig. 6. Crank, from left to right: anisotropic meshes with gradation ∞, 2.5, 1.5, and finally same gradation 1.5 with aspect ratio 2.5.

in Table 2. The CAD surface is made up of 10 patches, one for each of the 7 blades and 3 for the boss. The angle specified for all these geometric meshes is now $\theta = 4$ degrees.

Fig. 7 (top left) shows an isotropic mesh of the propeller without gradation, resulting in a poor shape quality due to the varying curvatures of the curves and surfaces. This quality is improved with a gradation of 2.5 and triangles are almost equilateral with a gradation of 1.5. The relative number of elements for these three isotropic meshes is respectively 1.000, 1.006 and 1.131, showing a lesser increase than in the previous example. Anisotropic meshes with gradation ∞, 2.5, 1.5, and finally same gradation 1.5 with aspect ratio 2.5 are also generated. Compared with an isotropic mesh with a same gradation, the relative number of elements is respectively 0.402, 0.448, 0.644, 2.033. This ratio is higher than in the previous example (and can even be greater than one) because the default value of h_{min}, the minimum size of the element edges, is 100 times smaller in the anisotropic case than in the isotropic case. This minimum is reached in this example because of the high curvatures near the leading edge of each blade. Therefore, the geometric accuracy is much better in the anisotropic case, with a comparable number of elements.

For the last example with 211,745 triangles, the total time for I/O and meshing is 24.289 seconds. The first gradation runs $6 + 3$ iterations and the second $16 + 3$ iterations with a negligible execution time.

mesh	gradation	aspect ratio	vertices	triangles
iso	∞	∞	46093	92055
iso	2.5	∞	47150	92606
iso	1.5	∞	52149	104159
aniso	∞	∞	18585	37028
aniso	2.5	∞	21656	41508
aniso	1.5	∞	33600	67053
aniso	1.5	2.5	107168	211745

Table 2. Propeller: meshing statistics.

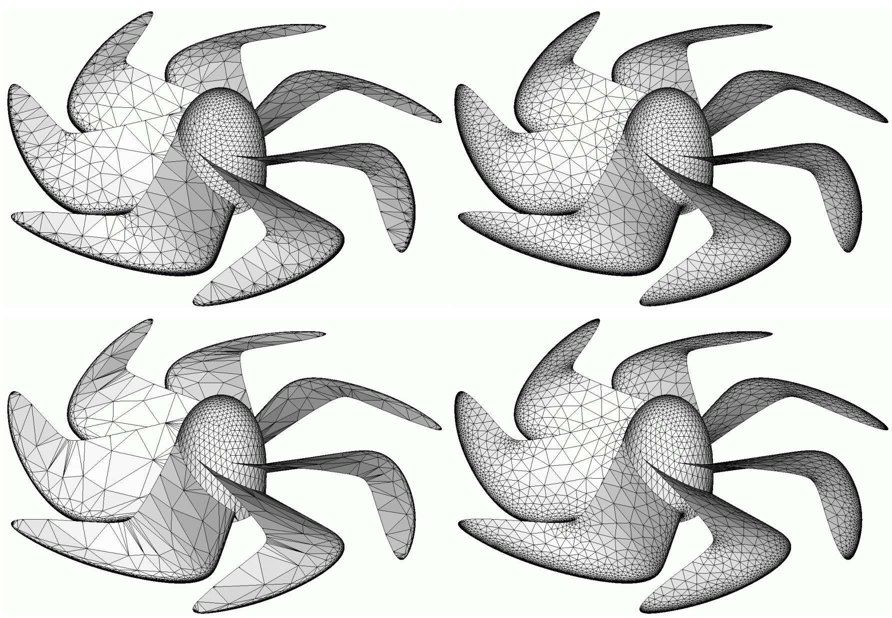

Fig. 7. Propeller: isotropic (top) and anisotropic (bottom) meshes without gradation and with gradation 1.5.

5 Conclusion

The general scheme of an indirect approach for meshing a surface constituted by a conformal assembly of parametric patches has been introduced and each step of the general scheme has been detailed. Emphasis has been placed on the geometric mesh generation, based on continuous isotropic and discontinuous anisotropic geometric metrics. In addition, a new mesh gradation control strategy for discontinuous anisotropic geometric metrics has been proposed. This strategy can be applied to control the gradation for volume meshing from a 3D continuous metric map. The proposed methodology has been applied to numerical examples, showing its efficiency.

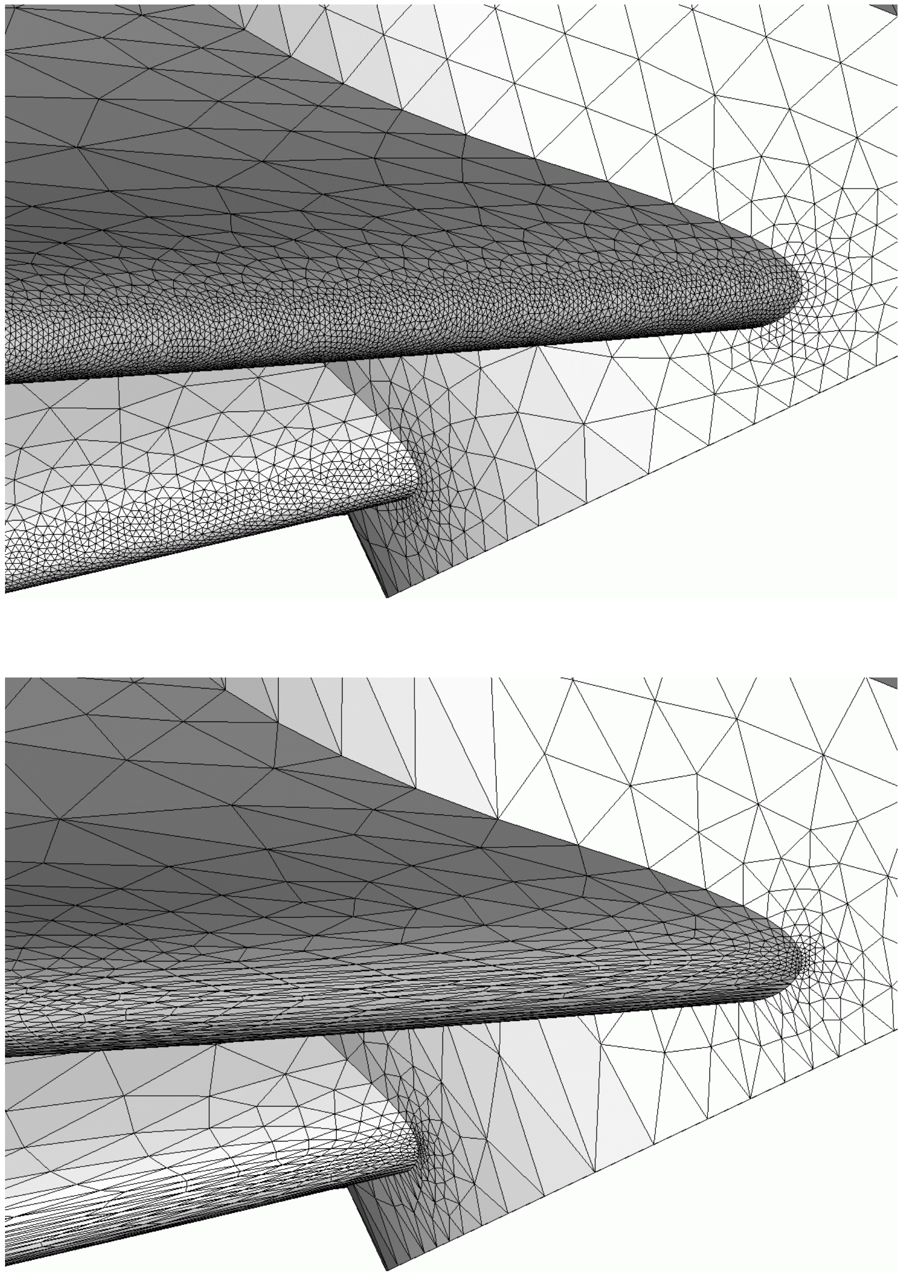

Fig. 8. Propeller: close up views of geometric meshes with gradation 1.5, either isotropic or anisotropic.

Future works include the parallelization of the geometric meshing process for large complex geometry, the parallelization of the mesh gradation strategy and the patch independent anisotropic geometric meshing.

References

1. Dolenc A, Mäkelä I (1990) Optimized triangulation of parametric surfaces. Mathematics of Surfaces IV
2. Filip D, Magedson R, Markot R (1986) Surface algorithm using bounds on derivatives. Comput.-Aided Geom. Des. 3:295–311
3. Piegl LA, Richard AM (1995) Tessellating trimmed NURBS surfaces. Computer Aided Design 27(1):16–26
4. Sheng X, Hirsch BE (1992) Triangulation of trimmed surfaces in parametric space. Computer Aided Design 24(8):437–444
5. Borouchaki H, George PL, Hecht F, Laug P, Saltel E (1997) Delaunay mesh generation governed by metric specifications – Part I: Algorithms. Finite Elements in Analysis and Design (FEAD) 25(1–2):61–83
6. Bossen FJ, Heckbert PS (1996) A pliant method for anisotropic mesh generation. 5th International Meshing Roundtable ‘96 Proceedings 375–390
7. Shimada K (1997) Anisotropic triangular meshing of parametric surfaces *via* close packing of ellipsoidal bubbles. 6th International Meshing Roundtable ‘97 Proceedings 63–74
8. Tristano JR, Owen SJ, Canann SA (1998) Advancing front surface mesh generation in parametric space using a Riemannian surface definition. 7th International Meshing Roundtable ‘98 Proceedings 429–445
9. Borouchaki H, Hecht F, Frey P (1998) Mesh gradation control. Int J Numer Meth Engng 43:1143–1165
10. Renaut E (2009) Reconstruction de la topologie et génération de maillages de surfaces composées de carreaux paramétrés. Thèse de doctorat UTT/INRIA soutenue le 15 décembre 2009
11. Borouchaki H, Laug P, George PL (2000) Parametric surface meshing using a combined advancing-front – generalized-Delaunay approach. Int J Numer Meth Engng 49(1–2):233–259
12. Laug P, Borouchaki H (2003) Interpolating and meshing 3-D surface grids. Int J Numer Meth Engng 58(2):209–225
13. Laug P, Borouchaki H (1999) BLSURF – Mesh generator for composite parametric surfaces – User’s manual. INRIA Technical Report RT-0235