

Parametric surface meshing using a combined advancing-front generalized Delaunay approach

Houman Borouchaki^{1,2,*}, Patrick Laug^{2,†} and Paul-Louis George^{2,§}

¹ *UTT, GSM-LASMIS, Université de Technologie de Troyes, BP 2060, 10010 Troyes Cedex, France*

² *INRIA, Gamma project, Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex, France*

SUMMARY

An indirect method for meshing parametric surfaces conforming to a user-specifiable size map is presented. First, from this size specification, a Riemannian metric is defined so that the desired mesh is one with unit length edges with respect to the related Riemannian space (the so-called ‘unit mesh’). Then, based on the intrinsic properties of the surface, the Riemannian structure is induced into the parametric space. Finally, a unit mesh is generated completely inside the parametric space such that it conforms to the metric of the induced Riemannian structure. This mesh is constructed using a combined advancing-front—Delaunay approach applied within a Riemannian context. The proposed method can be applied to mesh composite parametric surfaces. Several examples illustrate the efficiency of our approach. Copyright © 2000 John Wiley & Sons, Ltd.

KEY WORDS: parametric surface; surface meshing; anisotropic mesh; Riemannian metric; advancing front method; Delaunay method; adaptive meshing

1. INTRODUCTION

Surface meshing is of the utmost importance 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. 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–CAM modelers. In this paper, we make some remarks about a method suitable for generating a constrained mesh of a parametric patch. The constraint consists of a size map (which prescribes a size for every direction) associated with mesh vertices and also a shape quality associated with mesh elements. In fact, the aim is to construct a mesh that conforms to the specifications included in a given size map and such that its elements are as regular as possible.

*Correspondence to: Houman Borouchaki, UTT, GSM-LASMIS, Université de Technologie de Troyes, BP 2060, 10010 Troyes Cedex, France

†E-mail: houman.borouchaki@univ-troyes.fr

‡E-mail: patrick.laug@inria.fr

§E-mail: paul-louis.george@inria.fr

There are, essentially, two approaches to meshing parametric surfaces: direct and indirect. In the direct approach, the mesh is generated over the surface directly in \mathbb{R}^3 . Among the direct approaches we can cite the octree-based method [1], the advancing-front-based method [2] and the paving-based method [3]. The octree-based method consists of, first, generating a set of boxes, whose sizes conform to a given size map, then finding the intersections of each box with the surface and finally meshing each part of surface delimited by these contours. The resulting mesh contains the edges constituting the intersection contours. Small edges (with respect to the size map) may be generated by the above construction, as there is no control on the intersection of the boxes with the surface. An optimization stage is then necessary to remove these small edges. The advancing-front-based method consists of generating the surface mesh, layer by layer, starting from its contour. Thus, a front is constituted by the boundary of a layer and is moved through the interior of the surface when it contains more than three edges. In fact, at each iteration, a front edge is selected and an optimal point (which forms an optimal element with this edge with respect to a given size map) is generated near the surface which is then mapped onto the surface. The resulting point on the surface is then adjusted iteratively. The difficulty of such a method lies in the management of front collisions. The paving-based method works like an advancing-front-based method but at each iteration, a front vertex is selected and all the elements sharing this vertex are first generated in the tangent plane of the surface at this vertex and then mapped onto the surface. Compared with the advancing-front-based approach, this method converges more quickly as the local behaviour of the surface in the vicinity of a point upon it is well defined, whereas this behaviour is not well defined in the vicinity of an edge.

The indirect approach consists of meshing the parametric domain and mapping the resulting mesh onto the surface. It is conceptually straightforward as a two-dimensional mesh is generated in the parametric domain and thus it is expected to be faster than the direct approach. The problem with these methods 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 [4–7]. 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 [8–12].

For all these methods, 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 the two following properties hold:

- (1) all mesh elements are close to the surface,
- (2) 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 greatest 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. Note that if a given size map is specified, the above two properties can be locally violated. In fact, it is more useful to find

a compromise between the geometric approximation of the surface and a conforming size map surface mesh.

In this paper, we propose an indirect method based on the Riemannian metric. The general problem of parametric surface meshing is reduced to a problem of two-dimensional anisotropic meshing for which an advancing-front Delaunay coupled approach applied in an anisotropic context is proposed.

The method can take into account the geometric accuracy of the resulting meshes and can be applied in the context of adaptive computation.

We begin (Section 2) with a brief presentation of the method by introducing the concept of a metric. We describe (Section 3) the concept of the induced metric in parametric space so as to establish the equivalence between a surface mesh conforming to a given size map and a uniform mesh of size unity in parametric space with respect to the associated induced metric. In particular (Section 4), we show that a mesh satisfying the two geometric properties (a geometric mesh) conforms to a special size map, called a geometric size map. A compromise solution to generate a geometric mesh conforming to a given size map is also given. The advancing-front Delaunay coupled approach to constructing a uniform mesh of size unity with respect to a metric is then described (Section 5). Various application test examples are provided (Section 6) to illustrate the capabilities of the proposed method. Finally (Section 7), we give some indications to extend the results to general composite parametric surfaces.

2. A BRIEF PRESENTATION OF THE METHOD

Let Σ be a parametric surface defined by

$$\sigma : \Omega \rightarrow \Sigma, \quad (u, v) \mapsto \sigma(u, v)$$

Ω being a domain of \mathbb{R}^2 , and σ a continuous function of class C^2 . We assume that Ω is closed and bounded as is Σ . Let \mathcal{H}_3 be a size map associated to Σ . This means that at each point P of Σ , a length size $h(P)$ is defined which indicates the ideal mesh size in every direction. The problem that we face is to construct a mesh that conforms to the size map \mathcal{H}_3 . In other words, if V is a vertex of this mesh, the mesh size at the vicinity of V must be $h(V)$. We denote such a mesh by $\mathcal{T}(\Sigma, \mathcal{H}_3)$. The aim of the indirect approach is to construct a mesh in the parametric domain Ω so as to obtain $\mathcal{T}(\Sigma, \mathcal{H}_3)$ after mapping onto the surface.

Using the size map \mathcal{H}_3 , we can define a new metric on Ω (the actual metric being the identity \mathcal{I}_3). Indeed, by normalizing to unity the ideal mesh size, we can specify a Riemannian structure on Σ so that the ideal mesh conforming to the size map \mathcal{H}_3 is a uniform mesh of size unity (i.e. a mesh with unit length edges) with respect to the Riemannian structure. To this end, we redefine locally the metric \mathcal{M}_h at each point P of Σ as

$$\mathcal{M}_h(P) = \frac{1}{h^2(P)} \mathcal{I}_3$$

If PX is an ideal mesh edge sharing P , the Euclidean length size $L_{\mathcal{M}_h(P)}(PX)$ of PX with respect to metric $\mathcal{M}_h(P)$ is defined as

$$L_{\mathcal{M}_h(P)}(PX) = \sqrt{\overrightarrow{PX} \mathcal{M}_h(P) \overrightarrow{PX}}$$

and we deduce

$$L_{\mathcal{M}_h(P)}(PX) = 1 \Leftrightarrow L_{I_3}(PX) = h(P)$$

where $L_{I_3}(PX)$ is the usual Euclidean length size of edge PX . Thus, the problem of constructing a mesh conforming to a size map is reduced to a problem of constructing a uniform mesh of size unity with respect to some new metric, formally we have

$$\mathcal{T}(\Sigma, \mathcal{H}_3) = \mathcal{T}_{\mathcal{M}_h}(\Sigma)$$

where $\mathcal{T}_{\mathcal{M}_h}(\Sigma)$ is the uniform mesh of size unity of Σ with respect to the metric \mathcal{M}_h which is called a unit \mathcal{M}_h -mesh of Σ . In the next section, we show that this metric can be induced on the parametric space. In fact, we can define a metric $\tilde{\mathcal{M}}_h$ in Ω so as the mapping of the unit $\tilde{\mathcal{M}}_h$ -mesh of Ω onto Σ is the unit \mathcal{M}_h -mesh of Σ

$$\mathcal{T}_{\mathcal{M}_h}(\Sigma) = \sigma(\mathcal{T}_{\tilde{\mathcal{M}}_h}(\Omega))$$

Notice that the metric \mathcal{M}_h is isotropic while the metric $\tilde{\mathcal{M}}_h$ (as we will see hereafter) is generally anisotropic.

Now, the problem is reduced to constructing a unit $\tilde{\mathcal{M}}_h$ -mesh of Ω in \mathbb{R}^2 which is in fact a general problem of two-dimensional anisotropic meshing. In Section 4, we propose an algorithm to construct such a mesh.

A mesh conforming to a given size map can violate considerably the accurate (geometric) approximation of the surface. The latter can be quantified using the two properties mentioned in the introduction. In Section 3, we show that we can define a size map called a geometric size map, for a given geometric tolerance, so that a mesh conforming to this size map is a geometric mesh which is referred to as a Θ -mesh. Therefore, to obtain a geometric mesh conforming to a specified size map, we must bound this latter by an adequate geometric size map.

To summarize, in order to construct a mesh conforming to a given size map \mathcal{H}_3 , we proceed as follows:

- (1) Compute the geometric size map \mathcal{G}_3 on Σ related to a Θ -mesh for a given geometric tolerance, bound possibly the specified size map \mathcal{H}_3 by the geometric size map \mathcal{G}_3 ,
- (2) Define the new metric \mathcal{M}_h on Σ so as the mesh conforming to the size map \mathcal{H}_3 is a unit \mathcal{M}_h -mesh of Σ ,
- (3) Specify the induced metric $\tilde{\mathcal{M}}_h$ on Ω so as the unit \mathcal{M}_h -mesh of Σ is obtained by the mapping of unit $\tilde{\mathcal{M}}_h$ -mesh of Ω ,
- (4) Construct the unit $\tilde{\mathcal{M}}_h$ -mesh of Ω ,
- (5) Map this unit mesh onto the surface.

3. INDUCED METRIC MAP IN THE PARAMETRIC SPACE

In this section we establish the relation between the isotropic metric \mathcal{M}_h of \mathbb{R}^3 specified in Σ and the anisotropic metric $\tilde{\mathcal{M}}_h$ of \mathbb{R}^2 specified in Ω so that the relation $\mathcal{T}_{\mathcal{M}_h}(\Sigma) = \sigma(\mathcal{T}_{\tilde{\mathcal{M}}_h}(\Omega))$ is satisfied. For this, first, we recall the usual Euclidean length formula of a curved segment of \mathbb{R}^3 , then we extend this notion to the case where a generalized metric is specified in Σ .

3.1. Usual Euclidean length of a curved segment

Let Γ be a curved segment of Σ defined by a continuous function of class C^2 , $\gamma(t) \in \mathbb{R}^3$, where $t \in [a, b]$. Denoting the usual Euclidean norm by $\|\cdot\|$, the usual Euclidean length $L_{\mathcal{J}_3}(\Gamma)$ of Γ is given by

$$L_{\mathcal{J}_3}(\Gamma) = \int_a^b \|\gamma'(t)\| dt = \int_a^b \sqrt{{}^t\gamma'(t)\gamma'(t)} dt$$

As Γ is plotted on Σ , there is a function $\omega(t) \in \Omega$, where $t \in [a, b]$, such that $\gamma = \sigma \circ \omega$. We have $\gamma'(t) = \sigma'(\omega(t))\omega'(t)$, where $\sigma'(\omega(t))$ is the 3×2 matrix defined as

$$\sigma'(\omega(t)) = (\sigma'_u(\omega(t)) \ \sigma'_v(\omega(t)))$$

Thus, we obtain

$${}^t\gamma'(t)\gamma'(t) = {}^t\omega'(t){}^t\sigma'(\omega(t))\sigma'(\omega(t))\omega'(t)$$

but we have

$${}^t\sigma'(\omega(t))\sigma'(\omega(t)) = \begin{pmatrix} {}^t\sigma'_u(\omega(t)) \\ {}^t\sigma'_v(\omega(t)) \end{pmatrix} (\sigma'_u(\omega(t)) \ \sigma'_v(\omega(t)))$$

or

$${}^t\sigma'(\omega(t))\sigma'(\omega(t)) = \mathcal{M}_\sigma(\omega(t))$$

where \mathcal{M}_σ is a 2×2 matrix which characterizes the local intrinsic metric of Σ at point $\gamma(t)$ and is defined as

$$\mathcal{M}_\sigma(\omega(t)) = \begin{pmatrix} {}^t\sigma'_u(\omega(t))\sigma'_u(\omega(t)) & {}^t\sigma'_u(\omega(t))\sigma'_v(\omega(t)) \\ {}^t\sigma'_v(\omega(t))\sigma'_u(\omega(t)) & {}^t\sigma'_v(\omega(t))\sigma'_v(\omega(t)) \end{pmatrix}$$

We can deduce that

$$L_{\mathcal{J}_3}(\Gamma) = \int_a^b \sqrt{{}^t\omega'(t)\mathcal{M}_\sigma(\omega(t))\omega'(t)} dt$$

The above formula has an interesting interpretation. The Euclidean length of the curved segment $\omega(t)$ plotted in Ω depends on the Euclidean norm of $\omega'(t)$, while the Euclidean length of the curved segment $\gamma(t) = \sigma(\omega(t))$ (image of $\omega(t)$ on Σ) depends on the ‘Riemannian norm’ of $\omega'(t)$ with respect to the local intrinsic metric of Σ .

In particular (according to the mesh generation purpose), if $\omega(t)$ is a line segment AB of Ω , we have $\omega(t) = A + t\overrightarrow{AB}$, $\omega'(t) = \overrightarrow{AB}$, and

$$L_{\mathcal{J}_3}(\sigma(AB)) = \int_0^1 \sqrt{{}^t\overrightarrow{AB}\mathcal{M}_\sigma(A + t\overrightarrow{AB})\overrightarrow{AB}} dt$$

The above formula allows us to compute the length of the curved segment on Σ which is the image of an edge plotted on Ω . If the new metric \mathcal{M}_σ is given in Ω , we have

$$L_{\mathcal{M}_\sigma}(AB) = \int_0^1 \sqrt{{}^t\overrightarrow{AB}\mathcal{M}_\sigma(A + t\overrightarrow{AB})\overrightarrow{AB}} dt = L_{\mathcal{J}_3}(\sigma(AB))$$

Now, let us give the geometrical interpretation of the local metric \mathcal{M}_σ . Let o be a point of Ω , $O = \sigma(o)$ its image on Σ and $\mathcal{M}_\sigma(o)$ be the local intrinsic metric of Σ at O . Let ε be an arbitrary real value, then the locus of the points x of Ω such that

$${}^t\overrightarrow{ox} \mathcal{M}_\sigma(O) \overrightarrow{ox} = \varepsilon^2$$

is an ellipse, centred at o , denoted by $\mathcal{E}(o, \varepsilon)$. In the \mathbb{R}^2 space provided with the Euclidean metric defined by $\mathcal{M}_\sigma(o)$, $\mathcal{E}(o, \varepsilon)$ is a circle centred at o whose radius is ε . Assuming that $\mathcal{E}(o, \varepsilon) \subset \Omega$, for every $x \in \mathcal{E}(o, \varepsilon)$ we consider the curve $\Gamma_X = \sigma(ox)$ plotted on Σ , image by σ of the segment ox of Ω . Then the usual Euclidean length of Γ_X is given by

$$L_{\mathcal{J}_3}(\Gamma_X) = \int_0^1 \sqrt{{}^t\overrightarrow{ox} \mathcal{M}_\sigma(o + t\overrightarrow{ox}) \overrightarrow{ox}} dt$$

For a sufficiently small ε , irrespective of the parameter t , we have

$$\mathcal{M}_\sigma(o + t\overrightarrow{ox}) \approx \mathcal{M}_\sigma(o)$$

and then

$$L_{\mathcal{J}_3}(\Gamma_X) = \sqrt{{}^t\overrightarrow{ox} \mathcal{M}_\sigma(o) \overrightarrow{ox}} = \varepsilon$$

Thus, the locus of the curves, plotted on Σ , of origin O and length ε (for ε small enough) is the image by σ of the ellipse $\mathcal{E}(o, \varepsilon)$ plotted on Ω . The above computation indicates the metric deformation from surface Σ to its parametric domain Ω .

3.2. Generalized length of a curved segment

Let us consider the case where a general isotropic metric \mathcal{M}_h of \mathbb{R}^3 is specified on Σ . Recall that, if P is a point of Σ

$$\mathcal{M}_h(P) = \frac{1}{h^2(P)} \mathcal{J}_3$$

where $h(P)$ is the mesh size associated to P which is specified by the size map \mathcal{H}_3 . In fact, the metric \mathcal{M}_h defines a Riemannian structure over Σ . Provided with this Riemannian structure, the new (Riemannian) length $L_{\mathcal{M}_h}(\Gamma)$ of Γ is given by

$$L_{\mathcal{M}_h}(\Gamma) = \int_a^b \sqrt{{}^t\gamma'(t) \mathcal{M}_h(\gamma(t)) \gamma'(t)} dt$$

which can be written as

$$L_{\mathcal{M}_h}(\Gamma) = \int_a^b \sqrt{{}^t\omega'(t) \tilde{\mathcal{M}}_h(\omega(t)) \omega'(t)} dt$$

where

$$\tilde{\mathcal{M}}_h(\omega(t)) = {}^t\sigma'(\omega(t)) \mathcal{M}_h(\gamma(t)) \sigma'(\omega(t))$$

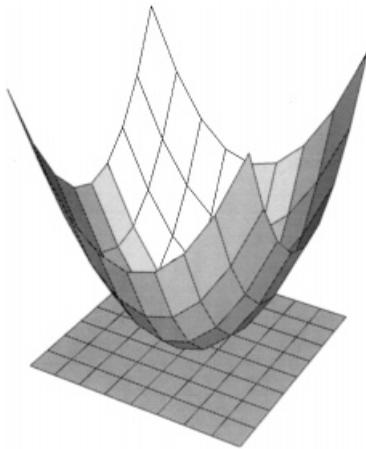


Figure 1. A quadratic surface.

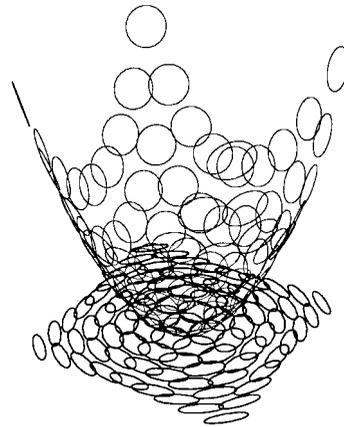


Figure 2. Metric deformation.

or

$$\tilde{\mathcal{M}}_h(\omega(t)) = \frac{1}{h^2(\gamma(t))} \mathcal{M}_\sigma(\omega(t))$$

In this case, if $\omega(t)$ is a line segment AB of Ω , we obtain

$$L_{\tilde{\mathcal{M}}_h}(\sigma(AB)) = \int_0^1 \sqrt{\vec{tAB} \cdot \tilde{\mathcal{M}}_h(A + tAB) \vec{tAB}} dt$$

The above formula allows us to compute the generalized length of the curved segment on Σ which is the image of an edge plotted on Ω . Let us define the new metric $\tilde{\mathcal{M}}_h$ in Ω . We have

$$L_{\tilde{\mathcal{M}}_h}(AB) = \int_0^1 \sqrt{\vec{tAB} \cdot \tilde{\mathcal{M}}_h(A + tAB) \vec{tAB}} dt$$

and thus we obtain

$$L_{\tilde{\mathcal{M}}_h}(\sigma(AB)) = L_{\tilde{\mathcal{M}}_h}(AB)$$

Figure 1 shows a mapping of a quad uniform grid of a square onto the quadratic surface defined by $z = (x^2 + y^2)/2$. As we can see, the surface mesh is not uniform due to the local intrinsic metric deformation. Figure 2 shows this deformation: the circles on the tangent planes of the surface defining a uniform metric \mathcal{M}_h on the surface correspond to the ellipses (on the parametric space) defining a non-uniform metric $\tilde{\mathcal{M}}_h$ in the parametric space.

Let us come back to the problem of mesh generation, the last equation shows that mapping of unit $\tilde{\mathcal{M}}_h$ -mesh of Ω onto the surface Σ gives the unit \mathcal{M}_h -mesh of Σ .

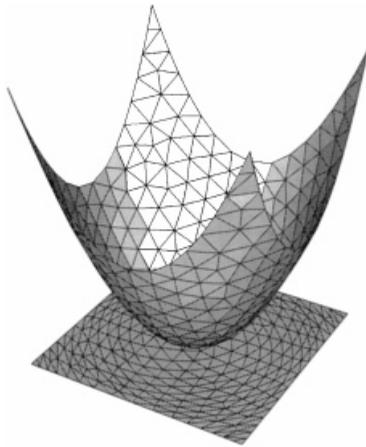


Figure 3. Unit meshes.

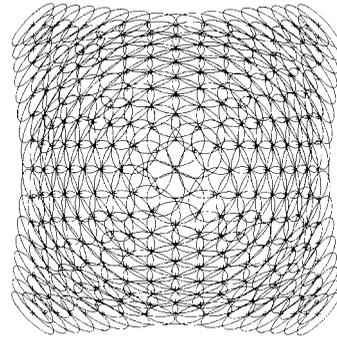


Figure 4. Metric on parametric space.

Figure 3 shows the mapping of a unit $\tilde{\mathcal{M}}_h$ -mesh of the parametric space onto the unit \mathcal{M}_h -mesh of the surface while Figure 4 illustrates the metric at each vertex of the unit $\tilde{\mathcal{M}}_h$ -mesh of the parametric space. Notice that an ideal unit mesh is such that these metrics meet the mesh vertices.

Remark. If a general anisotropic metric map \mathcal{M}_3 is specified on Σ , we can apply the same reasoning. In this case, we have

$$\tilde{\mathcal{M}}_h(\omega(t)) = {}^t\sigma'(\omega(t)) \cdot \mathcal{M}_3(\gamma(t)) \cdot \sigma'(\omega(t))$$

and we cannot express explicitly the metric $\tilde{\mathcal{M}}_h(\omega(t))$ in terms of the local intrinsic metric $\mathcal{M}_\sigma(\omega(t))$ of surface Σ .

4. GEOMETRIC MESH

In the previous section, we showed how to control the size of a mesh of the surface Σ indirectly by generating an adequate mesh in its parametric space. Recall that a mesh of a surface whose element vertices belong to the surface is suitable if all mesh elements are close to the surface and if every mesh element is close to the tangent plane related to its vertices. The first property is obvious while the second ensures some regularity for the mesh. A mesh satisfying the above properties is called a geometric mesh. A mesh conforming a given size map is not necessarily a geometric mesh. The problem that we face is to determine whether there is a size map such that a mesh conforming to this map is a geometric mesh. In other words, for each point of the surface, we must determine whether there is a ‘geometric’ mesh size such that all elements sharing this point and conforming to this size verify the two geometric properties. To this end, first we examine the case of a ‘free’ point of the surface and then, the case of a point belonging to a given curve plotted on the surface (and in particular the case of boundary points).

4.1. Free points

The free points belong only to the surface, so the key idea is to use the local behaviour of σ to determine the geometric size map [13]. For this, consider a point $X_0 = \sigma(x_0)$ of Σ , where $x_0 = (u_0, v_0)$, for $x = (u, v)$ close enough to x_0 , $\sigma(x)$ can be approximated by the quadratic function σ_{q,x_0} with

$$\sigma_{q,x_0}(x) = X_0 + \sigma'_u(x_0)u + \sigma'_v(x_0)v + \frac{1}{2}(\sigma''_{uu}(x_0)u^2 + 2\sigma''_{uv}(x_0)uv + \sigma''_{vv}(x_0)v^2)$$

which defines a quadratic surface Σ_{q,X_0} . Thus, the problem is reduced to defining a geometric size mesh at the vicinity of X_0 belonging to a quadratic surface.

The tangent plane of Σ at X_0 is spanned by the two vectors $\sigma'_u(x_0)$ and $\sigma'_v(x_0)$, its unit normal vector $\nu(X_0)$ is then defined as

$$\nu(X_0) = \frac{\sigma'_u(x_0) \times \sigma'_v(x_0)}{\|\sigma'_u(x_0) \times \sigma'_v(x_0)\|}$$

To analyse the behaviour of the quadratic surface Σ_{q,X_0} at the vicinity of x_0 , we consider the trace of Σ_{q,X_0} on the plane $\pi(X_0, \tau, \nu)$ spanned by the two orthogonal unit vectors

$$\tau(X) = \frac{\sigma'_u(x_0)u + \sigma'_v(x_0)v}{\|\sigma'_u(x_0)u + \sigma'_v(x_0)v\|} \quad \text{and} \quad \nu(X_0)$$

which is a normal section of Σ passing through $\tau(X)$. From this, we can deduce the ideal mesh size in the direction $\tau(X)$. The trace of Σ_{q,X_0} on $\pi(X_0, \tau, \nu)$ is the curve $\zeta(x)\tau(X) + \eta(x)\nu(X_0)$ passing through X_0 (origin of plane $\pi(X_0, \tau, \nu)$) defined by

$$\zeta(x) = {}^t\sigma_{q,x_0}(x)\tau(X) \quad \eta(x) = {}^t\sigma_{q,x_0}(x)\nu(X_0)$$

or

$$\zeta(x) = \|\sigma'_u(x_0)u + \sigma'_v(x_0)v\| + O(\|x\|^2)$$

$$\eta(x) = \frac{1}{2}({}^t\sigma''_{uu}(x_0)\nu(X_0)u^2 + 2{}^t\sigma''_{uv}(x_0)\nu(X_0)uv + {}^t\sigma''_{vv}(x_0)\nu(X_0)v^2)$$

Thus we have

$$\zeta^2(x) = \|\sigma'_u(x_0)u + \sigma'_v(x_0)v\|^2 + o(\|x\|^2) = {}^t x \mathcal{M}_\sigma(X_0) x + o(\|x\|^2)$$

which gives

$$\frac{\zeta^2(x)}{{}^t x \mathcal{M}_\sigma(X_0) x} = 1 + o(1)$$

Let us denote by $\mathcal{N}_\sigma(X_0)$ the 2×2 matrix

$$\begin{pmatrix} {}^t\sigma''_{uu}(x_0)\nu(X_0) & {}^t\sigma''_{uv}(x_0)\nu(X_0) \\ {}^t\sigma''_{uv}(x_0)\nu(X_0) & {}^t\sigma''_{vv}(x_0)\nu(X_0) \end{pmatrix}$$

then $\eta(x)$ can be expressed as

$$\eta(x) = \frac{1}{2} {}^t x \mathcal{N}_\sigma(X_0) x$$

Since $o(\eta) = o(\|x\|^2) = o(\zeta^2)$, we have

$$\eta = \frac{1}{2}\kappa(x)\zeta^2 + o(\zeta^2)$$

where

$$\kappa(x) = \frac{{}^t x \cdot \mathcal{N}_\sigma(X_0)x}{{}^t x \cdot \mathcal{M}_\sigma(X_0)x}$$

The above equation shows that the curvature of curve $\zeta(x)\tau(X) + \eta(x)v(X_0)$ at origin X_0 is $\kappa(x)$ and its principal unit normal coincides with $v(X_0)$. This implies (as we will see below) that the ideal mesh size in the direction $\tau(X)$ must be proportional to $|\rho(x)|$ (with a small factor of proportionality), where $\rho(x) = 1/\kappa(x)$ is the radius of curvature at X_0 . To find the ideal mesh size in every direction, we must determine the minimum of $|\rho(x)|$ (thus the maximum of $|\kappa(x)|$, quotient of two quadratic forms). For this, we can apply the simultaneous reduction of the matrices $\mathcal{N}_\sigma(X_0)$ and $\mathcal{M}_\sigma(X_0)$ which provides an orthonormal basis $(W_1(X_0), W_2(X_0))$ of tangent plane of Σ at X_0 for which the two above matrices are diagonal. Indeed, this basis is constituted by the unit eigenvectors of the matrix $\mathcal{M}_\sigma^{-1}(X_0)\mathcal{N}_\sigma(X_0)$ (note that $\mathcal{M}_\sigma(X_0)$ represents a metric and thus is invertible). This matrix is called the local geometric matrix of Σ at X_0 and is denoted by $\mathcal{G}_\sigma(X_0)$. In fact, in the basis $(W_1(X_0), W_2(X_0))$, $\kappa(x)$ is expressed as

$$\kappa(x) = \frac{\kappa_1(X_0)u^2 + \kappa_2(X_0)v^2}{u^2 + v^2}$$

where $\kappa_1(X_0)$ (resp. $\kappa_2(X_0)$) is the eigenvalue corresponding to $W_1(X_0)$ (resp. $W_2(X_0)$). We have

$$|\kappa(x)| \leq \frac{|\kappa_1(X_0)|u^2 + |\kappa_2(X_0)|v^2}{u^2 + v^2} \leq \max(|\kappa_1(X_0)|, |\kappa_2(X_0)|)$$

and therefore

$$|\rho(x)| \leq \min(|\rho_1(X_0)|, |\rho_2(X_0)|)$$

where $\rho_1(X_0) = 1/\kappa_1(X_0)$ (resp. $\rho_2(X_0) = 1/\kappa_2(X_0)$) is the radius of curvature in the direction $W_1(X_0)$ (resp. $W_2(X_0)$). To summarize, the ideal edge size (in any direction) sharing a point X_0 of Σ must be proportional to the maximum of the absolute value of the eigenvalues of the local geometric matrix $\mathcal{G}_\sigma(X_0)$ of Σ at X_0 which is indeed the absolute value of the minimal radius of curvature in any direction. The latter is denoted as $\rho(X_0)$.

Now we will show that to discretize a curve (knowing its curvature) by controlling the gap between the curve and its discretization, it is sufficient to consider a local mesh size proportional to the radius of curvature. Let X_0 be a point of the curve, $\tau(X_0)$ its unit tangent vector, $v(X_0)$ its principal unit normal and $\rho(X_0)$ its radius of curvature. If X is a point of the curve close enough to X_0 , we have

$$\|CX\| = |\rho(X_0)|$$

where C is the centre of curvature at X_0 defined by $X_0C = \rho(X_0)v(X_0)$. This means that X belongs to the circle of radius $|\rho(X_0)|$ and of centre C . The problem reduces to the discretization of a circle. Let θ be the angle between $\tau(X_0)$ and edge $\overrightarrow{X_0X}$ and $\delta(X_0)$ the gap between the edge X_0X and the circular arc X_0X , we have

$$\varepsilon = |\rho(X_0)|\delta(X_0) = 1 - \cos \theta$$

where ε represents the relative gap between the edge X_0X and the corresponding circular arc. Thus, locally we can control this gap in terms of the cosine of the angle between the edge (of the discretization) and the tangent of the curve. The length of the edge X_0X is then

$$\|X_0X\| = k(\theta)|\rho(X_0)|$$

with $\kappa(\theta) = 2 \sin \theta \simeq 2\sqrt{2\varepsilon}$.

Let us consider again point X_0 of surface Σ and let $\nu(X_0)$ be the unit normal of Σ at X_0 . The above discussion shows that for every edge X_0X verifying $\|X_0X\| = 2 \sin \theta |\rho(X_0)|$ (given θ), the angle between edge X_0X and the tangent plane of Σ at X_0 (characterized by $\nu(X_0)$) is bounded by θ and the relative gap between this edge and the surface is less than $1 - \cos \theta$. Thus, if K , a mesh element sharing X_0 , conforms to mesh size $2 \sin \theta |\rho(X_0)|$, the gap between K and the surface is bounded. This fact ensures the first property for a geometric mesh. In Reference [14] we show that if in addition K is equilateral then the angular gap between K and the tangent plane of Σ at X_0 is bounded (second geometric property).

4.2. Tied points

A tied point of the surface is a point belonging to one or several curved segments plotted on the surface. In this case two geometric mesh sizes must be considered: surface mesh size and curve mesh size. The first mesh size is discussed in the previous section. The second must ensure the geometric accuracy of the discretization of the curved segments. Let us denote by Γ_i these curved segments. Thus, for each Γ_i , the curve mesh size must be proportional to the absolute value of the radius of curvature of Γ_i . To ensure the first geometric property, it is obvious that the mesh size must be the minimum of the surface mesh size and the curve mesh sizes. Again, we can easily establish that for an equilateral element the second geometric property is also verified.

4.3. Θ -mesh

Let us denote by P_f (resp. P_t) the free points (resp. tied points) of the surface. Let $\Gamma_i(P_t)$ be the curved segments containing tied point P_t . Consider, for a given θ , the isotropic size map $\mathcal{G}_3(\theta)$ for which the mesh size h_g is defined as follows

$$\begin{aligned} h_g(P_f) &= 2 \sin \theta |\rho(P_f)| \\ h_g(P_t) &= 2 \sin \theta \min \left(|\rho_\Sigma(P_t)|, \min_i |\rho_{\Gamma_i(P_t)}(P_t)| \right) \end{aligned}$$

where ρ_Σ denotes the curvature with respect to the surface and ρ_{Γ_i} , the curvature with respect to the curve. From the previous results, we can assert that a mesh conforming to the size map $\mathcal{G}_3(\theta)$ is a geometric mesh which is called a Θ -mesh. The geometric accuracy of a Θ -mesh is dependent on θ . Note that some limitation must be considered for angle θ to ensure the validity of the approximation (see Reference [14] for more details).

Let us consider the case where a given size map \mathcal{H}_3 is specified. To obtain a Θ -mesh conforming to size map \mathcal{H}_3 , it is sufficient to limit the size given in \mathcal{H}_3 by the size given in $\mathcal{G}_3(\theta)$ for some specified θ depending on the desired geometric accuracy.

Figure 5 shows, for $\theta = 8^\circ$, the circles on the tangent planes of the surface defining a geometric metric $\mathcal{G}_3(8^\circ)$ on the surface correspond to the ellipses (on the parametric space) of the induced metric $\tilde{\mathcal{G}}_3(8^\circ)$ in the parametric space.

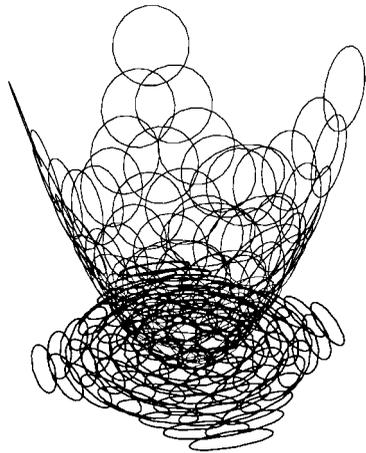


Figure 5. Geometric metric.

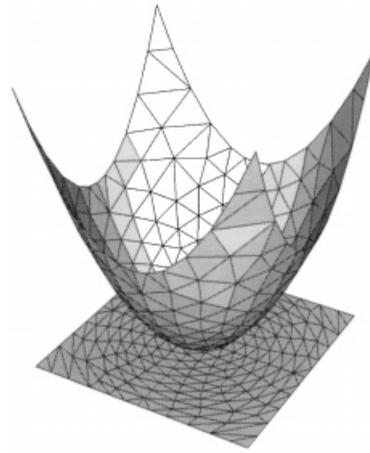
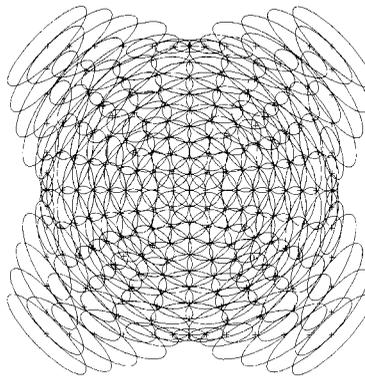
Figure 6. Θ -mesh.

Figure 7. Metric on parametric space.

Figure 6 shows the mapping of a unit geometric $\mathcal{S}_3(8^\circ)$ -mesh of the parametric space onto a Θ -mesh of the surface while Figure 7 illustrates the metric at each vertex of the unit geometric mesh of the parametric space.

5. A METHOD FOR GENERATING A UNIT MESH

In this section, we propose a method to construct a unit mesh of a domain Ω of \mathbb{R}^2 (defined from its contour) with respect to a Riemannian structure provided in Ω . It consists of meshing Ω in such a way that every edge in the resulting mesh is of unit length size (the Riemannian structure is used to determine the length of every edge plotted in Ω). This structure is defined from a metric associated with the points of Ω . The metric at point P of Ω is defined by a symmetric positive

definite 2×2 matrix, denoted as $\mathcal{M}_2(P)$ and given by

$$\mathcal{M}_2(P) = \begin{pmatrix} a(P) & b(P) \\ b(P) & c(P) \end{pmatrix}$$

where $a(P) > 0$ and $a(P)c(P) - b^2(P) > 0$. The aim is thus to obtain a unit mesh with respect to the field \mathcal{M}_2 , i.e. such that every mesh edge PX connected to P satisfies at best the relationship

$$\overrightarrow{PX} \mathcal{M}_2(P) \overrightarrow{PX} = 1$$

The meshing process of Ω , with respect to the control space, includes the following two stages:

- (1) the generation of the unit mesh of the boundary $b(\Omega)$ of Ω and
- (2) the generation of the unit mesh of Ω using the unit mesh of its boundary as input data.

Afterwards, we give the main features of our unit meshing algorithm. For more details see References [15, 16].

5.1. Unit mesh of the boundary

We assume that a mathematical model defines the geometry $b(\Omega)$ of the domain. This means that the boundary $b(\Omega)$ of Ω is made up of a set of curved segments ω_i of Ω where each ω_i is defined by

$$\omega_i : [a_i, b_i] \rightarrow \mathbb{R}^2, \quad t \mapsto \omega_i(t)$$

where $\omega_i(t)$ is a continuous function of class C^2 . The meshing problem then reduces to discretizing a given curved segment ω of Ω with

$$\omega : [a, b] \rightarrow \mathbb{R}^2, \quad t \mapsto \omega(t)$$

again, $\omega(t)$ being a continuous function of class C^2 .

Recall that the length of ω with respect to the Riemannian structure \mathcal{M}_2 is given by

$$L(\omega) = \int_a^b \sqrt{{}^t\omega'(t) \cdot \mathcal{M}_2(\omega(t)) \omega'(t)} dt$$

To discretize ω by edges with unit length sizes, first we compute the nearest integer n to $L(\omega)$ (ω must be discretized with n edges), then we determine the real values t_i , $1 \leq i \leq n-1$ ($t_0 = a$ and $t_n = b$) so that

$$\frac{L(\omega)}{n} = \int_{t_i}^{t_{i+1}} \sqrt{{}^t\omega'(T) \cdot \mathcal{M}_2(\omega(t)) \omega'(t)} dt$$

Finally the discretization of ω is constituted by the union of line segments $\omega(t_i)\omega(t_{i+1})$.

Actually, two methods can be envisaged to determine n and the t_i 's: direct and indirect methods. In the direct method, we assume that we can access to the first derivative $\omega'(t)$ of $\omega(t)$ and we use a special quadrature formula to approximate $L(\omega)$ (thus n) and also to compute the t_i 's. The idea is first to use adaptive Simpson quadrature formula to subdivide ω into a set of curved segments of length size smaller than a given threshold value δ (for example equal to 0.1), then to approximate the length size of ω by the sum of the length sizes of these curved segments. This subdivision also

allows us to determine the t_i 's using a simple rule of 3. The indirect method [17] consists of, first, approximating the curved segment ω with a polyline segment, then applying the same approach as in the direct method. The polyline segment makes it possible to avoid computing $\omega'(t)$.

5.2. Unit mesh of the domain

The unit mesh of the boundary of Ω provides a set of *constrained edges* having as endpoints a set of points, denoted as $\mathcal{S}(\Omega)$. At first, an initial constrained empty mesh of Ω is generated, whose sole vertices are the members of $\mathcal{S}(\Omega)$, respecting the constrained edges. A new mesh is then obtained by inserting iteratively field points inside this mesh, and optimized so as to obtain the unit mesh of the domain. The mesh is initialized by the initial constrained empty mesh. At each iteration, a set of edges of the current mesh of Ω , constituting a front, is selected and the field points

- (1) are generated with respect to front edges so as to form unit triangles and
- (2) are inserted into the current mesh via the *constrained* Delaunay kernel applied in a Riemannian context.

This process is repeated for as long as the current mesh of Ω is modified. Afterwards, we review the so-called unit advancing-front point placement strategy, the generalized constrained Delaunay kernel and the optimization procedure.

5.2.1. Unit advancing-front point placement strategy. An edge of the current mesh of Ω is selected as front edge if it separates a unit triangle from a non-unit one. A unit triangle is a triangle whose edges are of unit length size. A careful attention must be paid to ensure the 'correct' filling of the domain Ω . In fact, a non-unit triangle can be considered as a unit one although the unit length edge criterion has failed (the reader is referred to Reference [15] for more details). The 'optimal' point with respect to an edge front is defined, in the same side as the non-unit triangle (with respect to the edge), so as to form a unit triangle with the edge front. In practice, to define the optimal point P with respect to an edge front AB we can proceed as follows:

- (1) Compute P_a (resp. P_b) so that the triangle ABP_a (resp. ABP_b) is equilateral with respect to the metric $\mathcal{M}_2(A)$ (resp. $\mathcal{M}_2(B)$) and set $P = (P_a + P_b)/2$
- (2) Adjust iteratively the position of P by the following:
 - (i) Set $P_a = A + \frac{\vec{AP}}{L(AP)}$ (resp. $P_b = B + \frac{\vec{BP}}{L(BP)}$), where $L(AP)$ (resp. $L(BP)$) is the length of edge AP (resp. BP) with respect to the Riemannian metric \mathcal{M}_2

$$L(AP) = \int_0^1 \sqrt{{}^tAP \tilde{\mathcal{M}}_2(A + tAP) AP} dt$$

$$\left(\text{resp. } L(BP) = \int_0^1 \sqrt{{}^tBP \tilde{\mathcal{M}}_2(B + tBP) BP} dt \right)$$

- (ii) Set $P = (P_a + P_b)/2$.

Again, the length sizes $L(AP)$ and $L(BP)$ are approximated using an adaptive Simpson quadrature formula. The optimal point P is generated if it belongs to Ω and if it is not too close to an existing previously generated point or vertex.

5.2.2. *Generalized constrained Delaunay kernel.* The classical constrained Delaunay kernel is a procedure resulting in the insertion of one internal point in a (Delaunay) triangulation, based on an *Euclidean proximity criterion*. Formally speaking, the constrained Delaunay kernel can be written as [18, 19]

$$T = T - C(P) + B(P)$$

where $C(P)$ is the cavity associated with point P and $B(P)$ is the triangulation of $C(P)$ enclosing P as a vertex, T denoting the current Delaunay mesh. The cavity is constructed using a proximity criterion, written as

$$\{K, K \in T, P \in \text{Disc}(K) \text{ and } P \text{ visible from any vertex of } K\}$$

where $\text{Disc}(K)$ is the circumdisc of K .

The generalization of this procedure consists in redefining the cavity $C(P)$ in a Riemannian context [20]. Therefore, we define first the *Delaunay measure* $\alpha_{\mathcal{M}_2}$ associated with the pair (P, K) , with respect to a given metric \mathcal{M}_2

$$\alpha_{\mathcal{M}_2}(P, K) = \left[\frac{d(O_K, P)}{r_K} \right]_{\mathcal{M}_2}$$

where O_K (resp. r_K) is the centre (resp. radius) of the circumdisc of K and $[*]_{\mathcal{M}_2}$ indicates that the quantity $*$ is evaluated in the Euclidean space characterized by the metric \mathcal{M}_2 . The usual proximity criterion, $P \in \text{Disc}(K)$, is expressed as $\alpha_{\mathcal{I}_2}(P, K) < 1$, where \mathcal{I}_2 is the identity metric. The cavity $C(P)$ is then redefined as

$$C(P) = C_1(P) \cup C_2(P)$$

with

$$C_1(P) = \{K, K \in T, K \text{ including } P\}$$

and

$$C_2(P) = \left\{ K, K \in T, \exists K' \in C(P), K \text{ adjacent to } K' \right. \\ \left. \alpha_{\mathcal{M}(P)}(P, K) + \sum_V \alpha_{\mathcal{M}(V)}(P, K) < 4, V \text{ vertex of } K \right. \\ \left. P \text{ visible from the vertices of } K \right\}$$

Hence, the region $C(P)$ is constructed by adjacency from the elements of $C_1(P)$. With this definition, we can deduce that the generalized cavity is star-shaped with respect to P and the triangulation of $B(P)$ is then valid.

5.2.3. *Optimization procedure.* As an approximate Delaunay kernel is used to improve the vertices connection, the above proposed method provides only an approximate unit mesh of the domain Ω . Thus, to improve the shape quality of the resulting mesh, two processes can be used, diagonal swapping and relocation of the internal points. In fact, we assume that the domain Ω is already filled

with an adequate number of field points. The target is to obtain at best equilateral triangles with respect to the Riemannian metric \mathcal{M}_2 . The optimization procedure consists in applying iteratively diagonal swapping followed by point relocation. Subsequently, we recall the edge length quality and the extended element shape quality and we review the two above optimization procedures.

Edge length quality. Let AB be a mesh edge. The edge length quality Q_1 of AB with respect to a Riemannian metric \mathcal{M}_2 can be defined as

$$Q_1(AB) = \begin{cases} L_{\mathcal{M}_2}(AB) & \text{if } L_{\mathcal{M}_2}(AB) \leq 1 \\ \frac{1}{L_{\mathcal{M}_2}(AB)} & \text{if } L_{\mathcal{M}_2}(AB) > 1 \end{cases}$$

With this measure, we have $0 \leq Q_1(K) \leq 1$ and a unit edge has a length quality 1. This edge length quality can qualify the conformity of the mesh to the given Riemannian metric.

The edge length quality of a mesh \mathcal{T} can be defined as

$$Q_1(\mathcal{T}) = \left(\frac{1}{|\mathcal{T}|} \sum_{e \in \mathcal{T}} Q_1(e), \min_{e \in \mathcal{T}} Q_1(e) \right)$$

where e is an edge of mesh \mathcal{T} and $|\mathcal{T}|$ the number of elements of mesh \mathcal{T} . These two quantities measure the average edge length quality and the min edge length quality of the mesh.

Element shape quality. Let $K = P_1P_2P_3$ be a triangle, in the usual Euclidean space, a possible definition of its shape quality is, following Reference [21]

$$Q_s(K) = c \frac{|\text{Det}(\overrightarrow{P_1P_2}, \overrightarrow{P_1P_3})|}{\sum_{1 \leq j < k \leq 3} \|\overrightarrow{P_jP_k}\|^2}$$

where $\text{Det}(\overrightarrow{P_1P_2}, \overrightarrow{P_1P_3})$ is the determinant of the matrix whose columns are $\overrightarrow{P_1P_2}$ and $\overrightarrow{P_1P_3}$ (i.e. twice the surface of triangle K) while $\|\overrightarrow{P_jP_k}\|$ is the length of edge P_jP_k of K and $c = 2\sqrt{3}$ is a normalization factor such that an equilateral triangle shape quality is 1. Then $0 \leq Q_s(K) \leq 1$ and a well-shaped triangle has a quality close to 1 while an ill-shaped triangle has a quality close to 0.

In a Riemannian space (related to a metric \mathcal{M}_2), the shape quality of a triangle K can be defined as

$$Q_s(K) = \min_{1 \leq i \leq 3} Q_s^i(K)$$

where $Q_s^i(K)$ is the triangle shape quality in the Euclidean space characterized by metric $\mathcal{M}_2(P_i)$ at vertex P_i of K . To compute the quantity $Q_s^i(K)$, we just have to transform the Euclidean space associated with the metric $\mathcal{M}_2(P_i)$ specified at point P_i into the usual Euclidean space. Then we consider the quality of the so transformed triangle K^i , i.e.

$$Q_s^i(K) = Q_s(K^i)$$

In Reference [8] it is shown that

$$Q_s^i(K) = c \frac{|\sqrt{\text{Det}(\mathcal{M}_2(P_i))} \text{Det}(\overrightarrow{P_1P_2}, \overrightarrow{P_1P_3})|}{\sum_{1 \leq j < k \leq 3} \overrightarrow{P_jP_k} \mathcal{M}_2(P_i) \overrightarrow{P_jP_k}}$$

Again, the element shape quality of a mesh \mathcal{T} can be defined as

$$Q_s(\mathcal{T}) = \left(\frac{1}{|\mathcal{T}|} \sum_{K \in \mathcal{T}} Q_s(K), \min_{K \in \mathcal{T}} Q_s(K) \right)$$

where K is an element of mesh \mathcal{T} . These two quantities measure the average and min element shape quality of the mesh.

Diagonal swapping. Diagonal swapping is a way to improve the mesh quality (edge length or element shape) by topological modification. This technique also makes it possible to suppress an edge when this is possible. Let e be an edge in the mesh. We denote by the *shell* of e the set of triangles sharing e . The quality of a shell is that of its worst element. The diagonal swapping is then applied if the resulting mesh is of quality better than that of the initial shell.

Each edge e associated with a ratio β_e denoting the quality improvement factor when the diagonal swapping is applied to e . With the aim of optimizing the mesh quality, diagonal swapping is applied iteratively following the variation of β_e . Initially, the ratio of improvement is fixed to a value $\beta_e > 1$ (in practice, $\beta_e = 2$ is advised) then factor β_e is decreased to 1. In this way, the most significant diagonal swappings are done first. This pseudo-sorting procedure for diagonal swapping is due to the Riemannian structure provided in the domain and can be ignored in a classical mesh optimization (with respect to a Euclidean structure).

Moving the points. Let P be an internal point in the mesh and (K_i) be the ball of P (the set of elements with P as vertex), this process consists in moving P to enhance the quality of the ball (i.e. that of its worst element). Two procedures have been developed, one leading to achieve unit length for the edges (and thus improve the edge length quality), the other to obtain optimal elements (to improve the element shape quality).

Let (P_i) be the vertex in (K_i) other than P . Each point P_i is associated with an optimal point (P_i^*) such that

$$\overrightarrow{P_i P_i^*} = \frac{1}{L_{\mathcal{M}_2}(P_i P)} \overrightarrow{P_i P}$$

so that $L_{\mathcal{M}_2}(P_i P_i^*) = 1$ holds. This process consists in moving point P step by step towards the centroid of the points (P_i^*) if the shape quality of set (K_i) is enhanced. This process leads to establishing unit length for the edges emanating from P .

Let (e_i) be the edges opposite vertex P for the triangles (K_i) where $K_i = [P, e_i]$. With each edge e_i , is associated the optimal point P_i^* such that triangle $K_i^* = [P_i^*, e_i]$ enjoys the best possible shape quality $Q_s(K_i^*)$. Let C be the centroid of the P_i^* s, then point P is moved step by step towards C while the variation of the quality is controlled. This process leads to establishing optimality in terms of triangle shape with respect to the Riemannian structure. To obtain point P_i^* , it is possible to consider the centroid of the optimal points associated with e_i , each of them being evaluated in the metric specified at the vertices of triangle K_i .

5.2.4. Some implementation issues. In this short section, we would like to make some remarks about the main difference between a general two-dimensional anisotropic meshing and a parametric domain meshing. Recall that the length of segment AB of Ω provided the metric $\tilde{\mathcal{M}}_h$ is

$$L_{\tilde{\mathcal{M}}_h}(AB) = \int_0^1 \sqrt{\overrightarrow{tAB} \tilde{\mathcal{M}}_h(A + t\overrightarrow{AB}) \overrightarrow{AB}} dt$$

To compute this length size, we usually use an approximative quadrature formula. To do so, we have to know, for a given point (u, v) of Ω , the metric $\tilde{\mathcal{M}}_h(u, v)$. In fact

$$\tilde{\mathcal{M}}_h(u, v) = \frac{1}{h^2(\sigma(u, v))} \mathcal{M}_\sigma(u, v)$$

where $\mathcal{M}_\sigma(u, v)$ is the local intrinsic metric of Σ at point $\sigma(u, v)$ which is dependent on the first partial derivatives σ'_u and σ'_v of σ . Thus an external procedure (depending on the definition of Σ) must be used which returns the two vectors σ'_u and σ'_v .

Moreover, if a geometric mesh is required, we also have to know the local geometric matrix \mathcal{G}_σ of Σ which is dependent on the second partial derivatives σ''_{uu} , σ''_{uv} and σ''_{vv} of σ . In this case, the external procedure must also return these derivatives.

6. APPLICATION EXAMPLES

The proposed method is implemented in the BLSURF [22] software package which includes a general anisotropic mesh generator. To show the efficiency of our approach, three test examples are given. The first example consists of an analytical parametric patch, the second is a discrete one, and the last is applicable to finite element computation. For some generated meshes we will give the number of elements and the required CPU time in seconds (HP700/99 MHz). To analyse the quality of the generated meshes, we will give also the element shape quality and the edge length quality with respect to the related Riemannian structure as defined in the previous section.

6.1. Klein bottle

We consider the surface (the Klein bottle [23]) defined by

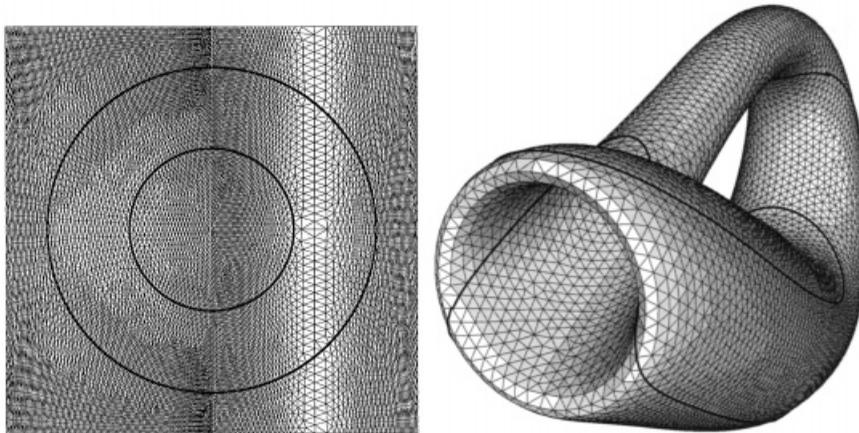
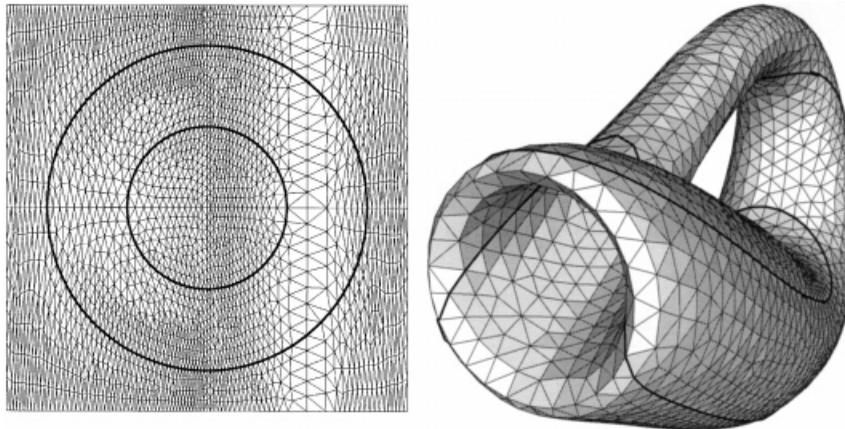
$$\begin{aligned} x &= \begin{cases} 6 \cos u(1 + \sin u) + r \cos u \cos v & \text{if } 0 \leq u \leq \pi \\ 6 \cos u(1 + \sin u) - r \cos v & \text{if } \pi \leq u \leq 2\pi \end{cases} \\ y &= \begin{cases} 16 \sin u + r \sin u \cos v & \text{if } 0 \leq u \leq \pi \\ 16 \sin u & \text{if } \pi \leq u \leq 2\pi \end{cases} \\ z &= r \sin v \end{aligned}$$

where $r = 4 - 2 \cos u$ over the square $0 \leq u, v \leq 2\pi$. In addition, we specify in the parametric domain two curved segments, namely two circles of origin (π, π) and radii 0.3 and 0.6.

Figure 8 (right-hand side) shows a uniform mesh with size 0.5 in \mathbb{R}^3 which represents the mapping of the mesh of its parametric domain (left-hand side). Another uniform mesh with size 1.0 is also given in Figure 9. A Θ -mesh related to $\theta = 8^\circ$ is illustrated by Figure 10 (right-hand side) in \mathbb{R}^3 and (left-hand side) in its parametric domain. The characteristics of these meshes are shown in Table I ($k05$, $k10$ and Θ -8).

Now we consider the case where a size map is specified by a continuous function $h(u, v)$ over the parametric domain defined by

$$h(u, v) = 10 \left| \sqrt{u^2 + v^2} - 0.1 \arctan \frac{v}{u} \right| + \varepsilon$$

Figure 8. Uniform mesh $h = 0.5$ ($k05$).Figure 9. Uniform mesh $h = 1.0$ ($k10$).

which imposes a small size along a given curved segment plotted on parametric domain with a size expansion elsewhere. Figure 11 shows a mesh conforming to this size map. Again, this size map is bounded with a geometric size map corresponding to a Θ -mesh related to $\theta = 14^\circ$. The resulting mesh is shown in Figure 12. Table I gives the characteristics of these meshes (spir and spirg).

6.2. Bust of Victor Hugo

In this section, we focus on the surface meshing problem of a real object from a set of sampled points obtained using the three-dimensional digitalization system '3D Videolaser' ([24]). Because of its functionalities, the data have a rectangular topology. This means that the sampled points are distributed over a network of meridians and parallels of the object surface. In particular, for objects

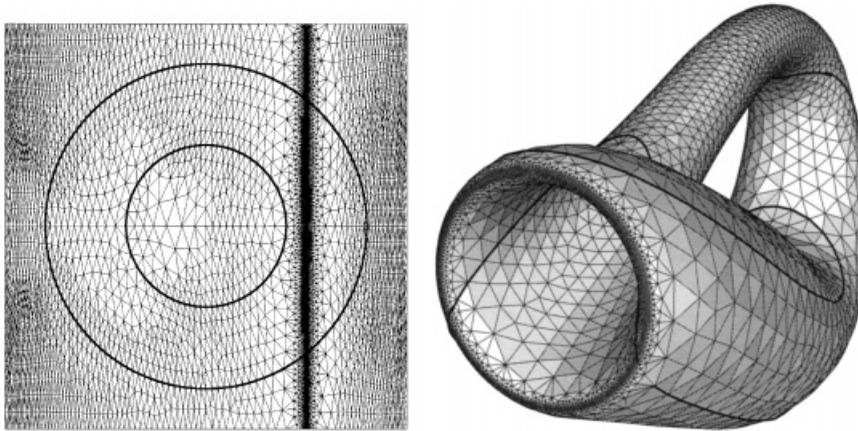
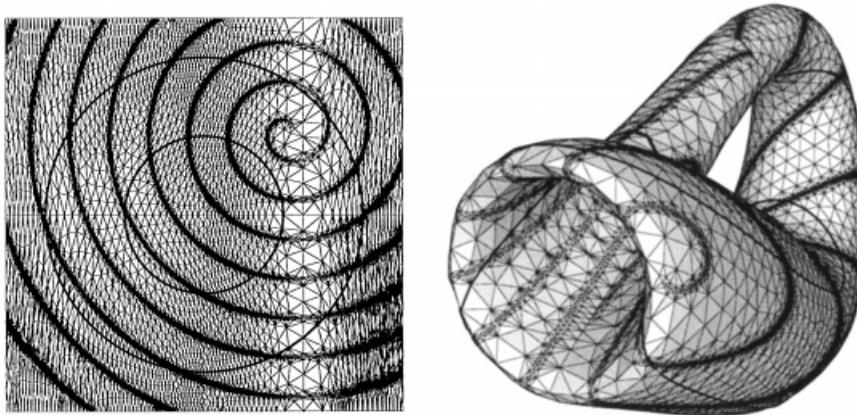
Figure 10. Θ -mesh $8^\circ(\Theta-8)$.

Figure 11. Spiral conforming mesh (spir).

Table I. Mesh characteristics (Klein bottle).

mesh	nv	ne	Q_1	Q_s	tcpu
$k05$	8510	17 020	0.58–1.01	0.62–0.98	12
$k10$	2077	4154	0.63–1.02	0.55–0.97	4
$\Theta-8$	4155	8310	0.44–1.01	0.47–0.94	19
Spir	14 154	28 308	0.27–1.01	0.22–0.86	53
Spirg	14 821	29 642	0.27–1.00	0.23–0.86	100

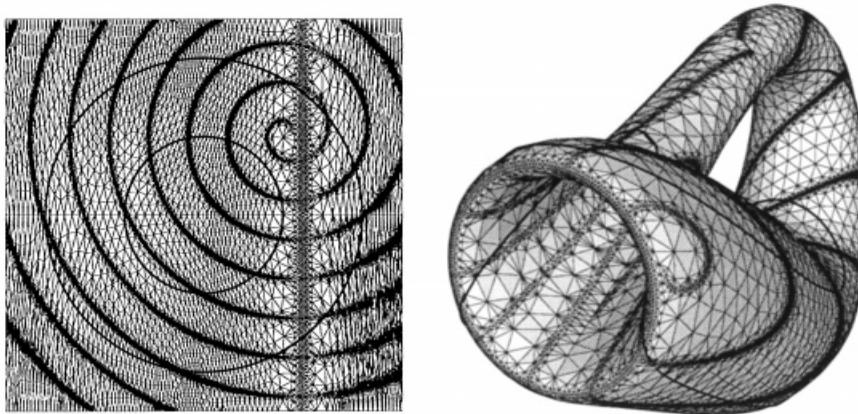


Figure 12. Geometric spiral conforming mesh (spirig).

having a cylindrical geometry, the data (sampled points of the surface) are a matricial structure whose columns and lines correspond respectively to meridians and parallels. Each element of this matricial structure represents the distance between this element (which is a point of the surface) and the rotational axis of the object.

This system allows us to define a cylindrical surface, ‘in a discrete manner’, by

$$x = r(\theta, z) \cos(\theta)$$

$$y = r(\theta, z) \sin(\theta)$$

$$z = z$$

where the parametric space is the rectangle

$$0 \leq \theta < 2\pi \quad \text{and} \quad 0 \leq z \leq d$$

which is called the CAD grid. Indeed, the function $r(\theta, z)$ is known only for a finite number of points of the rectangle, more precisely on the uniform grid

$$\theta_j = j\Delta\theta, \quad 0 \leq i \leq n_c$$

$$z_i = i\Delta z, \quad 0 \leq j \leq n_l$$

where n_c (resp. n_l) is the number of columns (resp. lines) of the grid and $\Delta\theta$ (resp. Δz) is the sampling step according to θ (resp. z).

To mesh such a surface, by considering the rectangular parametric space, we must ensure that the discretizations of contours $\theta = 0$ and $\theta = 2\pi$ are identical. A generated point (θ, z) is mapped onto the surface using an interpolating scheme over the uniform CAD grid. As a final remark, note that in this case, the external procedure to evaluate the quantities $r'_\theta(\theta, z)$, $r'_z(\theta, z)$, $r''_{\theta^2}(\theta, z)$, $r''_{z^2}(\theta, z)$ and $r''_{\theta z}(\theta, z)$ uses a finite difference method applied to the CAD grid.

Figure 13 shows the mapping of the 69 by 120 CAD grid of a bust of Victor Hugo. Figures 14 and 15 show two uniform meshes corresponding, respectively, to some mesh size

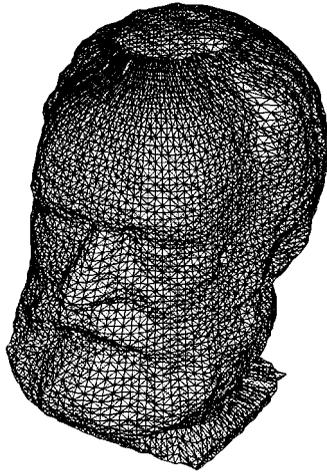
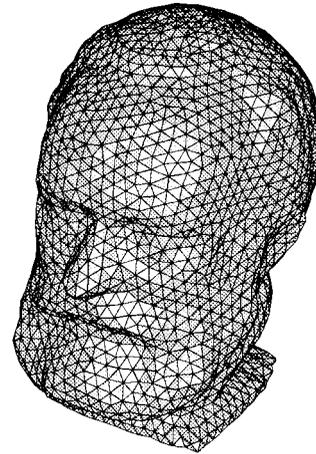
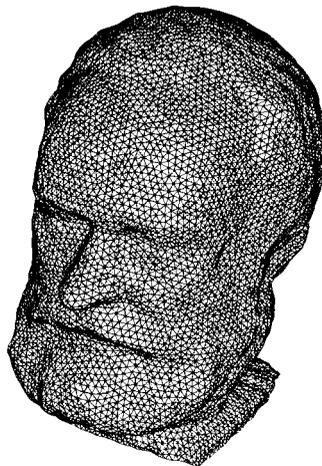
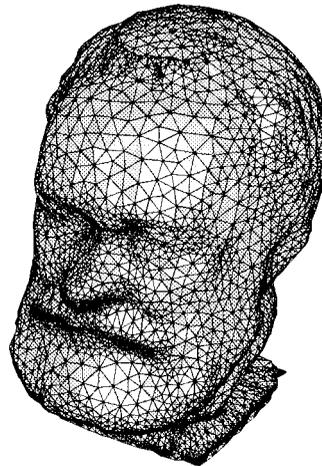


Figure 13. Initial CAD mesh of Hugo.

Figure 14. Uniform mesh h .Figure 15. Uniform mesh $h/2$.Figure 16. Θ -mesh 8° .

h and $h/2$. Figure 16 corresponds to a Θ -mesh related to $\theta = 8^\circ$. Figures 17–19 represent the above meshes in parametric space. Table II gives the characteristics of these meshes.

These results show that the two uniform meshes are of good quality, while the geometric mesh contains small or big edges with respect to the size specification. This fact is due to the locally abrupt variation of the curvature on the surface. We can avoid this by controlling the underlying mesh gradation. In fact, the geometric size map must be modified according to a specified mesh gradation.

To end this section, on a light note, we present a mesh of a bust of Victor Hugo conforming to a size map having the shape of a pair of spectacles (cf. Figure 20).

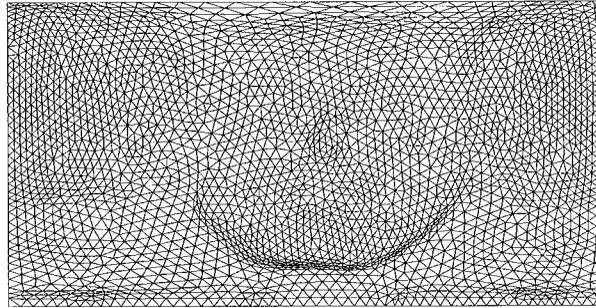
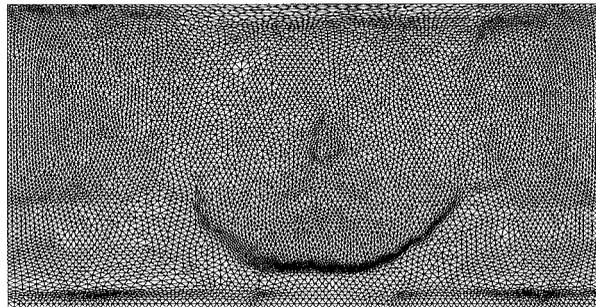
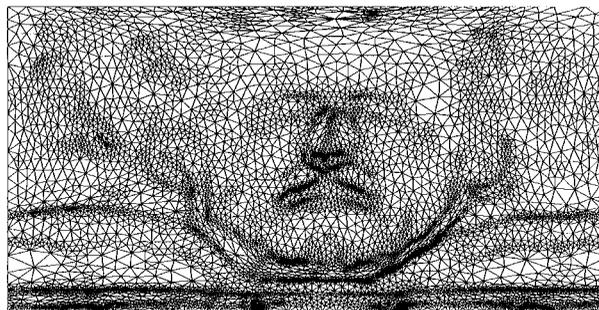
Figure 17. Uniform mesh h in parametric space.Figure 18. Uniform mesh $h/2$ in parametric space.Figure 19. Θ -mesh 8° in parametric space.

Table II. Mesh characteristics (bust of Victor Hugo).

mesh	nv	ne	Q_1	Q_s	tcpu
h	2513	4866	0.59–1.01	0.60–0.95	12
$h/2$	9837	19357	0.53–1.02	0.61–0.97	30
Θ -8	7100	13981	0.30–1.05	0.22–0.90	22

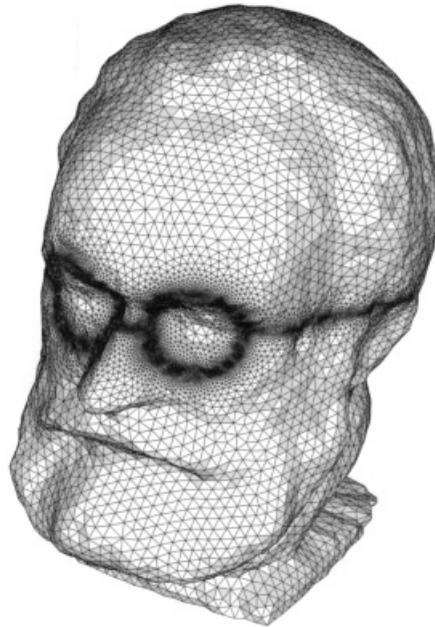


Figure 20. Another view of Victor Hugo!

6.3. Columbia shuttle

The last example concerns the Columbia shuttle which can be defined using NURBS patches. Figures 21 and 22 show, respectively, a uniform mesh and a geometric mesh of the shuttle.

7. CONCLUSION

An indirect method based on the metric concept for meshing parametric surfaces conforming to a given size map has been introduced. The problem has been reduced to generating a two-dimensional anisotropic mesh in the parametric domain. To this end, a combined advancing-front-Delaunay algorithm applied in a Riemannian context is proposed. The approach presented emphasized the geometric approximation of the surface. Several examples have been illustrated and many other confidential industrial examples have been tested, showing the pertinence of the method.

The proposed approach can be easily extended to composite parametric surfaces meshing. The main difference lies in the discretization of interface curves which represent the common boundary of several patches. In fact, these interfaces must be discretized directly in \mathbb{R}^3 in contrast to the case of one unique patch. Indeed, it provides a unique discretization of contours is related parametric spaces so as to ensure the conformity of the global surface mesh. Therefore, to discretize the interface curves we should know the inverse of the mapping functions defining the surface. To avoid this, we can use an approximation of interface curves in \mathbb{R}^3 and also in parametric spaces by polyline segments. The following example (Figures 23 and 24) shows two different meshes of the famous Utah Teapot which is constituted by 32 patches.

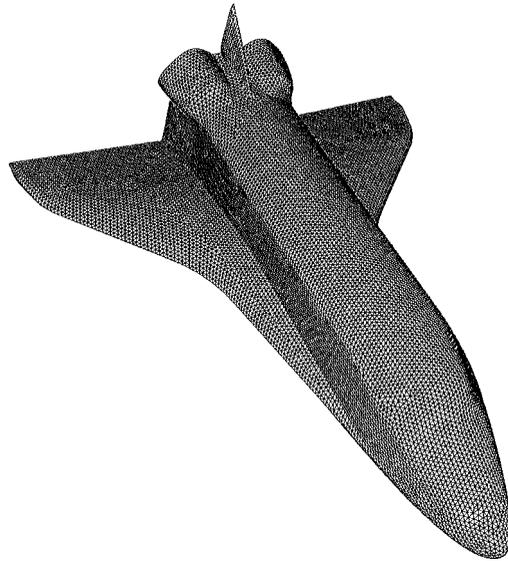


Figure 21. Uniform mesh.

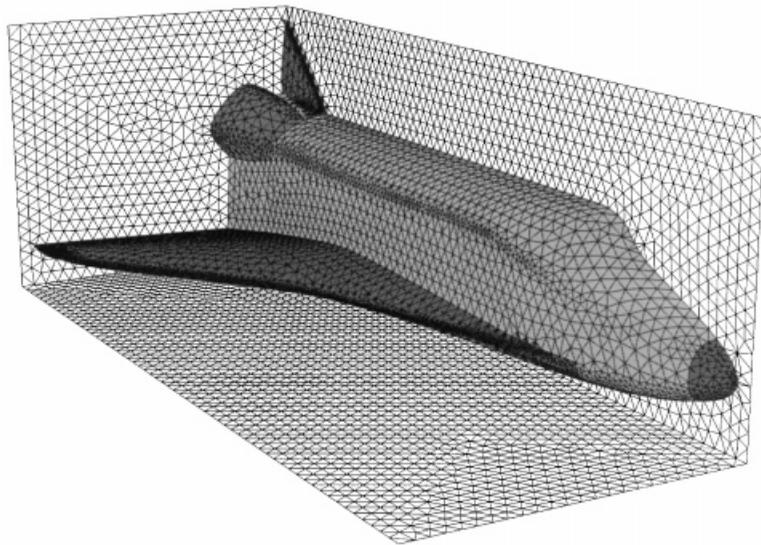


Figure 22. Geometric mesh.

The proposed method can be applied in an adaptive computation. In this case, the specified size map is not continuous. In fact, the mesh size is given (for example via an *a posteriori* error estimate) at vertices of a surface mesh which has been used for computation. Let us denote this mesh as a background mesh. By interpolating the size over the background mesh we obtain a

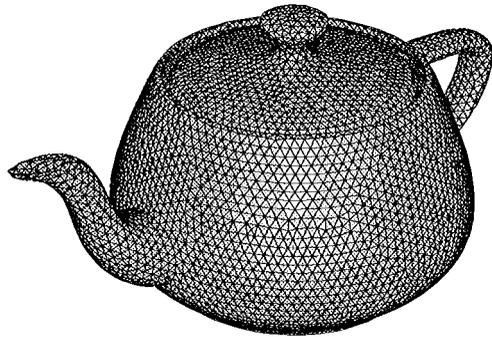


Figure 23. Uniform mesh.

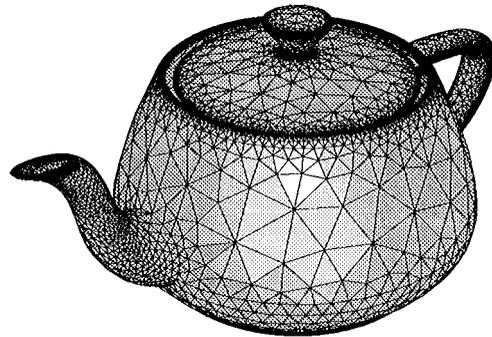


Figure 24. Geometric mesh.

continuous size map and we can use the proposed method to generate a conforming mesh. This procedure is iterated until the *a posteriori* size specification is reached.

Finally, we mention the generalization of the method to:

- (1) the generation of meshes conforming to a general anisotropic metric map, and in particular geometric anisotropic meshes,
- (2) the generation of meshes constituted by quadratic elements.

REFERENCES

1. Shephard MS, Georges MK. Automatic three-dimensional mesh generation technique by the finite octree technique. *International Journal for Numerical Methods in Engineering* 1991; **32**:709–749.
2. Löhner R. Regridding surface triangulations. *Journal of Computational Physics* 1996; **126**(6):1–10.
3. Hartmann E. A marching method for the triangulation of surfaces. *The Visual Computer* 1998; **14**:95–108.
4. Dolenc A, Mäkelä I. Optimized triangulation of parametric surfaces. *Mathematics of Surfaces IV* 1990.
5. Filip D, Magedson R, Markot R. Surface algorithm using bounds on derivatives. *Computer-Aided Geometric Design* 1986; **3**:295–311.
6. Piegel LA, Richard AM. Tessellating trimmed NURBS surfaces. *Computer-Aided Design* 1995; **27**(1):16–26.
7. Sheng X, Hirsch BE. Triangulation of trimmed surfaces in parametric space. *Computer-Aided Design* 1992; **24**(8):437–444.
8. Borouchaki H, George PL, Hecht F, Laug P, Saltel E. Delaunay mesh generation governed by metric specifications. Part I: algorithms. *Finite Elements in Analysis and Design* 1997; **25**(1):61–83.
9. Borouchaki H, George PL, Mohammadi B. Delaunay mesh generation governed by metric specifications. Part II: Applications. *Finite Elements in Analysis and Design* 1997; **25**(1):85–109.
10. Bossen FJ, Heckbert PS. A pliant method for anisotropic mesh generation. *Fifth International Meshing Roundtable '96 Proceedings*, 1996; 375–390.
11. Shimada K. Anisotropic triangular meshing of parametric surfaces via close packing of ellipsoidal bubbles. *Sixth International Meshing Roundtable '97 Proceedings*, 1997; 63–74.
12. Tristano JR, Owen SJ, Canann SA. Advancing front surface mesh generation in parametric space using a Riemannian surface definition. *Seventh International Meshing Roundtable '98 Proceedings*, 1998; 429–445.
13. Lelong-Ferrand J, Arnaudès JM. *Cours de Mathématiques, Tome 3, Géométrie et Cinématique*. Dunod Université, Bordas: Paris, 1977.
14. Borouchaki H. *Curve and Surface Meshing*, to appear.
15. Borouchaki H. Quality meshes. In *Traité de Mécanique et Ingénierie des Matériaux, Méthodes Numériques, Maillage*. Hermès: Paris, to appear.
16. George PL, Borouchaki H. *Delaunay Triangulation and Meshing. Application to Finite Element Methods*. Hermès: Paris, 1998.
17. Laug P, Borouchaki H, George PL. Discrétisation adaptative des courbes. *Revue Internationale de CFAO et d'Informatique Graphique* 1996; **11**(6):617–634.

18. Watson DF. Computing the n-dimensional Delaunay tessellation with application to Voronoi polytopes. *The Computer Journal* 1981; **24**:167–172.
19. Borouchaki H, George PL. Aspects of 2D Delaunay mesh generation. *International Journal for Numerical Methods in Engineering* 1997; **40**:1957–1975.
20. Borouchaki H, George PL. Triangulation de Delaunay et métrique riemannienne. Applications aux maillages éléments finis. *Revue Européenne des Éléments Finis* 1996; **5**(3):323–340.
21. Lo SH. Automatic mesh generation and adaptation by using contours. *International Journal for Numerical Methods in Engineering* 1991; **31**:689–707.
22. Laug P, Borouchaki H. BLSURF—Mailleur de surfaces composées de carreaux paramétrés—Manuel d'utilisation. *Rapport Technique INRIA RT-0232*, Juin 1999.
23. Bourke P. Klein bottle, <http://www.mhri.edu.au/~pdb/geometry/klein>, October 1996.
24. Schmitt F, Maître H, Clainchard A, Lopez-Krahn J. Acquisition and representation of real object surface data. *SPIE Proceedings of Biostereometrics'85 Conference*, vol. 602. Cannes, France, 2–6 December, 1985.