INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING

Int. J. Numer. Meth. Engng 2000; **00**:1–6 *Prepared using* nmeauth.cls *[Version: 2002/09/18 v2.02]*

An Algebraic Method for Smoothing Surface Triangulations on a Local Parametric Space

J. M. Escobar*, G. Montero, R. Montenegro and E. Rodríguez

University Institute of Intelligent Systems and Numerical Applications in Engineering,

University of Las Palmas de Gran Canaria,

Campus Universitario de Tafira, 35017 Las Palmas de Gran Canaria, Spain.

SUMMARY

This paper presents a new procedure to improve the quality of triangular meshes defined on surfaces. The improvement is obtained by an iterative process in which each node of the mesh is moved to a new position that minimizes a certain objective function. This objective function is derived from algebraic quality measures of the local mesh (the set of triangles connected to the adjustable or *free node*). If we allow the free node to move on the surface without imposing any restriction, only guided by the improvement of the quality, the optimization procedure can construct a high-quality local mesh, but with this node in an *unacceptable* position. To avoid this problem the optimization is done in the *parametric mesh*, where the presence of barriers in the objective function maintains the free node inside the *feasible region*. In this way, the original problem on the surface is transformed into a two-dimensional one on the *parametric space*. In our case, the parametric space is a plane, chosen in terms

*Correspondence to: University Institute for Intelligent Systems and Numerical Applications in Engineering,

University of Las Palmas de Gran Canaria, Campus Universitario de Tafira, 35017 Las Palmas de Gran Canaria, Spain.

Contract/grant sponsor: Spanish Government and FEDER; contract/grant number: REN2001-0925-C03-02/CLI, CGL2004-06171-C03-02/CLI

Received

Revised

of the local mesh, in such a way that this mesh can be optimally projected performing a *valid* mesh, that is, without *inverted* elements. Several examples and applications presented in this work show how this technique is capable to improve the quality of triangular surface meshes.

KEY WORDS: Surface mesh smoothing; Surface triangulations; Algebraic quality measures; Adaptive finite elements

1. INTRODUCTION

The element quality of a mesh heavily affects the numerical convergence of the FEM computational scheme (see for example [1]). In particular, the quality of the surface mesh of a 3-D domain has an important role when boundary conditions involve the evaluation of integrals on boundary faces. Moreover, the improvement of a 3-D mesh is conditioned by the quality of its boundary discretization. Special attention must be taken into account when a mesh generator, which starts from a boundary triangulation, is used. This is the case of the advancing front method. It can be thus deduced that it is essential to have a technique that allows us to optimize surface meshes.

In this work we present an algebraic method to smooth meshes defined on surfaces. The smoothing technique is based on a vertex repositioning directed by the minimization of an suitable objective function. The construction of the objective function is done in the framework of theory of *algebraic quality measures* developed by Knupp in [2]. For 2-D or 3-D meshes the quality improvement can be obtained by an iterative process in which each node of the mesh is moved to a new position that minimizes the objective function [3]. This function is derived from a quality measure of the *local mesh* determined by the set of triangles connected to the adjustable or *free node*.

We have chosen, as a starting point in section 2, a 2-D objective function that presents a barrier at the

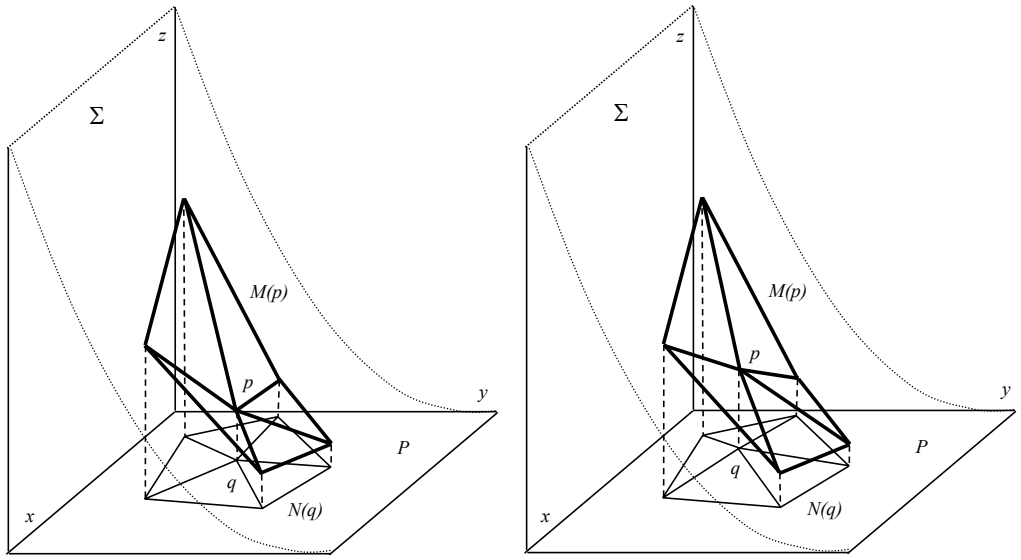


Figure 1. Illustration of a local mesh $M(p)$ on a surface Σ and its projection $N(q)$ on a plane P (a). Optimized local meshes after relocation of the free node (b).

boundary of the *feasible region* (set of points where the free node could be placed to get a *valid* local mesh, that is, without *inverted elements*). This barrier has an important role because it does not allow the optimization algorithm to create a tangled mesh when it starts with a valid one. Nevertheless, objective functions constructed by algebraic quality measures are only directly applicable to inner nodes of 2-D or 3-D meshes, but not to its boundary nodes. To overcome this problem, the local mesh, $M(p)$, sited on a surface Σ, is orthogonally projected on a plane P (the existence and search for this plane will be discuss in section 3) in such a way that it obtains a valid local mesh $N(q)$, see Figure 1(a). Therefore, it can be said that $M(p)$ is *geometrically conforming* with respect to P [4]. Here p is the free node on Σ and q is its projection on P. The optimization of $M(p)$ is obtained by the appropriated optimization of $N(q)$. To do this we try to get *ideal* triangles in $N(q)$ that become equilateral in $M(p)$, see Figure

1(b). In general, when the local mesh $M(p)$ is on a surface, each triangle is placed on a different plane and it is not possible to define a feasible region on Σ. Nevertheless, this region is perfectly defined in $N(q)$ as it is analyzed in section 2.1.

To construct the objective function in $N(q)$, it is first necessary to define the objective function in $M(p)$ and, afterward, to establish the connection between them. A crucial aspect for this construction is to keep the barrier of the 2-D objective function. This is done with a suitable approximation in the process that transforms the original problem on Σ into an entirely two-dimensional one on P. We develop this approximation in section 2.2. The barrier has a beneficial effect on keeping $N(q)$ valid if initially it satisfies this requirement. In other words, the barrier prevents creation of $N(q)$ tangled mesh if it is valid before the optimization.

The optimization of $N(q)$ becomes a two-dimensional iterative process. The optimal solutions of each two-dimensional problem form a sequence $\left\{\mathbf{x}^k\right\}$ of points belonging to P. We have checked in many numerical test that $\left\{\mathbf{x}^k\right\}$ is always a convergent sequence. We will show an example of this convergence in section 4.1. It is important to underline that this iterative process only takes into account the position of the free node in a discrete set of points, the points on Σ corresponding to $\left\{\mathbf{x}^k\right\}$ and, therefore, it is not necessary that the surface is smooth. Indeed, the surface determined by the piecewise linear interpolation of the initial mesh is used as a reference to define the geometry of the domain.

If the node movement only responds to an improvement of the quality of the mesh, it can happen that the optimized mesh loses details of the original surface, specially when this has sharp edges or vertices. To avoid this problem, every time the free node p is moved on Σ, the optimization process checks the distance between the centroid of the triangles of $M(p)$ and the underlying surface (the true surface, if it is known, or the piece-wise linear interpolation, if it is not). When this distance is greater than certain threshold value, the movement of the node is aborted and its previous position is stored. Details

about this strategy are presented in section 2.2. Although this criterion does not assure that the maximal distance between the original and optimized meshes exceeds a given tolerance threshold, this is a good and easily computable approximation. Following this procedure, it is possible to measure such distance in more significant points of the triangles and, thus, to obtain a more accurate approximation.

There are several alternatives to the previous method. For example, Garimella et al. [5] develop a method to optimize meshes in which the nodes of the optimized mesh are kept close to the original positions by imposing the Jacobians of the current and original meshes to also be close. Frey et al. [6] get a control of the gap between the mesh and the surface by modifying the element-size (subdividing the longest edges and collapsing the shortest ones) in terms of an approximation of the smallest principal curvatures radius associated with the nodes. Rassineux et al. [7] also use the smallest principal curvatures radius to estimate the element-size compatible with a prescribed gap error. They construct a geometrical model by using the Hermite diffuse interpolation in which local operations like edge swapping, node removing, edge splitting, etc. are made to adapt the mesh size and shape. More accurate approaches, that have taken into account the directional behavior of the surface, have been considered in by Vigo [8] and, recently, by Frey in [9].

Other applications of our proposed optimization technique are presented in section 4.2 and 4.3.

2. CONSTRUCTION OF THE OBJECTIVE FUNCTION

As it is shown in [3], [10], and [11] we can derive optimization functions from *algebraic quality measures* of the elements belonging to a local mesh. Let us consider a triangular mesh defined in $\mathbb{R}^2$ and let t be an triangle in the physical space whose vertices are given by $\mathbf{x}_k = (x_k, y_k)^T \in \mathbb{R}^2$, $k = 0, 1, 2$. First, we are going to introduce an algebraic quality measure for t. Let t_R be the reference triangle with vertices $\mathbf{u}_0 = (0,0)^T$, $\mathbf{u}_1 = (1,0)^T$, and $\mathbf{u}_2 = (0,1)^T$. If we choose $\mathbf{x}_0$ as the translation

vector, the affine map that takes t_R to t is $\mathbf{x} = A\mathbf{u} + \mathbf{x}_0$, where A is the Jacobian matrix of the affine map referenced to node $\mathbf{x}_0$, given by $A = (\mathbf{x}_1 - \mathbf{x}_0, \mathbf{x}_2 - \mathbf{x}_0)$. We will denote this type of affine map as $t_R \stackrel{A}{\rightarrow} t$. Let now t_I be an *ideal* triangle (not necessarily equilateral) whose vertices are $\mathbf{w}_k \in \mathbb{R}^2$, $(k = 0, 1, 2)$ and let $W_I = (\mathbf{w}_1 - \mathbf{w}_0, \mathbf{w}_2 - \mathbf{w}_0)$ be the Jacobian matrix, referenced to node $\mathbf{w}_0$, of the affine map $t_R \stackrel{W_I}{\rightarrow} t_I$; then, we define $S = AW_I^{-1}$ as the weighted Jacobian matrix of the affine map $t_I \stackrel{S}{\rightarrow} t$. In the particular case that t_I was the equilateral triangle t_E, the Jacobian matrix $W_I = W_E$ will be defined by $\mathbf{w}_0 = (0, 0)^T$, $\mathbf{w}_1 = (1, 0)^T$ and $\mathbf{w}_2 = (1/2, \sqrt{3}/2)^T$.

The weighted matrix S is independent of the node chosen as reference; it is said to be *node invariant* [2]. We can use matrix norms, determinant or trace of S to construct algebraic quality measures of t. For example, the Frobenius norm of S, defined by $|S| = \sqrt{tr\,(S^T S)}$, is specially indicated because it is easily computable. Thus, it is shown in [2] that $q_\eta = \frac{2\sigma}{|S|^2}$ is an algebraic quality measure of t , where $\sigma = \det(S)$. The maximum value of q_η is unity and it is reached when $S = \mu\Theta$, where μ is a nonnegative scalar and Θ is a 2×2 orthogonal matrix with determinant 1 (a rotation matrix). So, when $q_\eta = 1$ the Jacobian matrix is $A = \mu\Theta W_I$ and, therefore, the triangle t is a scale change and a rotation of the ideal triangle t_I. In other words, the triangle t that maximizes q_η is similar to t_I. We shall use this quality measure to construct an objective function.

Let $\mathbf{x} = (x, y)^T$ be the position vector of the free node, and let S_m be the weighted Jacobian matrix of the m-th triangle of a valid local mesh of M triangles. The objective function associated to m-th triangle is $\eta_m = \frac{|S_m|^2}{2\sigma_m}$, and the corresponding objective function for the local mesh is the n-norm of $(\eta_1, \eta_2, \ldots, \eta_M)$, i.e .,

$$|K_\eta|_n (\mathbf{x}) = \left[\sum_{m=1}^{M} \eta_m^n (\mathbf{x}) \right]^{\frac{1}{n}} \tag{1}$$

In this context the feasible region is defined as the set of points where the free node must be located to get the local mesh to be valid. More concretely, the feasible region is the interior of the polygonal set

$\mathcal{H} = \bigcap_{m=1}^{M} H_m$, where H_m are the half-planes defined by $\sigma_m(\mathbf{x}) \geq 0$. We say that a triangle is *inverted* if $\sigma < 0$. The objective function (1) presents a barrier at the boundary of the feasible region that avoids the optimization algorithm to create a tangled mesh when it starts with a valid one.

Previous considerations and definitions are only directly applicable for 2-D (or 3-D) meshes, but some of them must be properly adapted when the meshes are located on an arbitrary surface. For example, the concept of valid mesh is not clear in this situation because neither is the concept of inverted element. We will deal with these questions in next subsections.

2.1. Relation between the surface mesh and the parametric mesh

Suppose that for each local mesh $M(p)$ placed on the surface Σ, that is, with all its nodes on Σ, it is possible to find a plane P such that the orthogonal projection of $M(p)$ on P is a valid mesh $N(q)$. Moreover, suppose that we define the axes in such a way that the x, y-plane coincide with P. If, in the feasible region of $N(q)$, it is possible to define the surface Σ by the parametrization $\mathbf{s}(x, y) = (x, y, f(x, y))$, where f is a continuous function, then, we can optimize $M(p)$ by an appropriate optimization of $N(q)$. We will refer to $N(q)$ as the *parametric mesh*. The basic idea consists on finding the position $\overline{q}$ in the feasible region of $N(q)$ that makes $M(p)$ be an optimum local mesh. To do this, we define *ideal* elements in $N(q)$ that become equilateral in $M(p)$. Let $\tau \in M(p)$ be a triangular element on Σ whose vertices are given by $\mathbf{y}_k = (x_k, y_k, z_k)^T$, $(k = 0, 1, 2)$ and t_R be the reference triangle in P (see Figure 2). If we choose $\mathbf{y}_0$ as the translation vector, the affine map $t_R \overset{A_\pi}{\rightarrow} \tau$ is $\mathbf{y} = A_\pi \mathbf{u} + \mathbf{y}_0$, where A_π is its Jacobian matrix, given by

$$A_\pi = \begin{pmatrix} x_1 - x_0 & x_2 - x_0 \\ y_1 - y_0 & y_2 - y_0 \\ z_1 - z_0 & z_2 - z_0 \end{pmatrix} \tag{2}$$

Now, consider that $t \in N(q)$ is the orthogonal projection of τ on P. Then, the vertices of t are $\mathbf{x}_k = \Pi \mathbf{y}_k = (x_k, y_k)^T$, $(k = 0, 1, 2)$, where $\Pi = (\mathbf{e}_1, \mathbf{e}_2)^T$ is 2×3 matrix of the affine map $\tau \stackrel{\Pi}{\rightarrow} t$, being $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ the canonical basis in $\mathbb{R}^3$ (the associated projector from $\mathbb{R}^3$ to P, considered as a subspace of $\mathbb{R}^3$, is $\Pi^T \Pi$). Taking $\mathbf{x}_0$ as translation vector, the affine map $t_R \stackrel{A_P}{\rightarrow} t$ is $\mathbf{x} = A_P \mathbf{u} + \mathbf{x}_0$, where $A_P = \Pi A_\pi$ is its Jacobian matrix

$$A_P = \begin{pmatrix} x_1 - x_0 & x_2 - x_0 \\ y_1 - y_0 & y_2 - y_0 \end{pmatrix} \tag{3}$$

Therefore, the 3×2 matrix of the affine map $t \stackrel{T}{\rightarrow} \tau$ is

$$T = A_\pi A_P^{-1} \tag{4}$$

Let V_π be the subspace spanned by the column vectors of A_π and let π be the plane defined by V_π and the point $\mathbf{y}_0$. Our goal is to find the *ideal* triangle $t_I \subset P$, moving q on P, such that t_I is mapped by T into an equilateral one, $\tau_E \subset \pi$. Note that each triangle of $M(p)$ defines a different ideal triangle on P, because the plane π changes with the considered triangle.

Because $\text{rank}(A_\pi) = \text{rank}(A_P) = 2$, there exists a unique factorization $A_\pi = QR$, where Q is an orthogonal matrix and R is an upper triangular one with $[R]_{ii} > 0$ $(i = 1, 2)$. The columns of the 3×2 matrix Q define an orthonormal basis $\{\mathbf{q}_1, \mathbf{q}_2\}$ that spans V_π, so we can see Q as the matrix of the affine map $t_R \stackrel{Q}{\rightarrow} \tau_R$ and R as the 2×2 Jacobian matrix of the affine map $\tau_R \stackrel{R}{\rightarrow} \tau$ (see Figure 2). As $t_R \stackrel{W_E}{\rightarrow} t_E$ and Q is an orthogonal matrix that keeps the angles and norms of the vectors, then $t_E \stackrel{Q}{\rightarrow} \tau_E$ and, therefore

$$QW_E = A_\pi R^{-1} W_E \tag{5}$$

is the 3×2 Jacobian matrix of affine map $t_R \stackrel{QW_E}{\rightarrow} \tau_E$. On the other hand, we define on the plane π

$$S = RW_E^{-1} \tag{6}$$

as the 2×2 weighted Jacobian matrix of the affine map that transforms the equilateral triangle into the physical one, that is, $\tau_E \overset{S}{\rightarrow} \tau$.

We have chosen as ideal triangle in π the equilateral one ($\tau_I = \tau_E$), then, the Jacobian matrix W_I of the affine map $t_R \overset{W_I}{\rightarrow} t_I$ is calculated by imposing the condition $TW_I = QW_E$, because $t_R \overset{TW_I}{\rightarrow} \tau_I$ and $t_R \overset{QW_E}{\rightarrow} \tau_E$. Taking into account (5), this yields

$$TW_I = A_\pi R^{-1} W_E \tag{7}$$

and, from (4), we obtain

$$W_I = A_P R^{-1} W_E \tag{8}$$

so we define on P the *ideal-weighted* Jacobian matrix of the affine map $t_I \overset{S_I}{\rightarrow} t$ as $S_I = A_P W_I^{-1}$. From (8) it results

$$S_I = A_P W_E^{-1} R A_P^{-1} \tag{9}$$

and, from (6)

$$S_I = A_P W_E^{-1} S W_E A_P^{-1} = A_P W_E^{-1} S \left(A_P W_E^{-1}\right)^{-1} = S_E S S_E^{-1} \tag{10}$$

where $S_E = A_P W_E^{-1}$ is the *equilateral-weighted* Jacobian matrix of the affine map $t_E \overset{S_E}{\rightarrow} t$. Finally, from (10), we obtain the next similarity transformation.

$$S = S_E^{-1} S_I S_E \tag{11}$$

Therefore, it can be said that the matrices S and S_I are *similar.*

2.2. Optimization on the parametric space

S, as it is defined in (6), might be used to construct the objective function and, then, solve the optimization problem. Nevertheless, this procedure has important disadvantages. First, the optimization

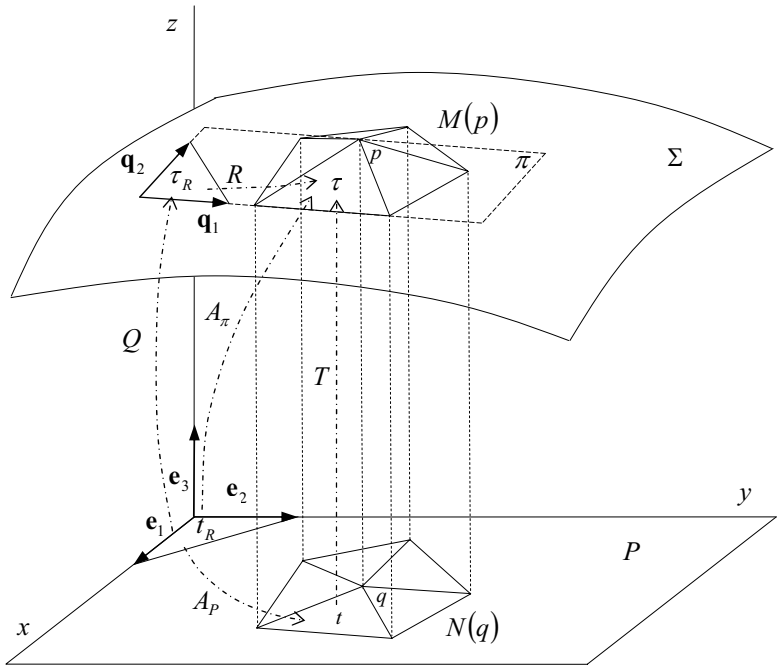


Figure 2. Local surface mesh $M(p)$ and its associated parametric mesh $N(q)$.

of $M(p)$, working on the true surface, would require the imposition of the constraint $p \in \Sigma$. It would complicate the resolution of the problem because, in many cases, Σ is not defined by a smooth function. Moreover, when the local mesh $M(p)$ is on a curved surface, each triangle is sited on a different plane and the objective function, constructed from S, lacks barriers. It is impossible to define a feasible region in the same way as it was done at the beginning of this section. Indeed, all the positions of the free node, except those that make $\det(S) = 0$ for any triangle, produce correct triangulations of $M(p)$. However, for many purposes as, for example, to construct a 3-D mesh from the surface triangulation, there are unacceptable positions of the free node. As example, in Figure 3(a) is shown a surface mesh formed by the triangles ABp, BCp and ACp, whose vertices A, B, C and p are sited on a sphere. The feasible region of the parametric mesh $N(q)$, resultant to project $M(p)$ into the plane P, is the triangle ABC.

If the optimization is done only attending to the shape of the triangles, we could obtain a non valid mesh $N(q)$ as it is shown in Figure 3(c). This would be the resultant smoothed mesh if no restriction, except $p \in \Sigma$, is imposed. Note that, in this case, q is outside the feasible region and, then, $N(q)$ is tangled. We shall consider $M(p)$ unacceptable as a local surface triangulation. Finally, in Figure 3(b) is shown the resulting position obtained when the optimization process only allows movement of node q inside the feasible region. A particular case of this example is discussed in section 4.1.

To overcome these difficulties we propose to carry out the optimization of $M(p)$ in an indirect way, working on $N(q)$. With this approach the movement of the free node will be restricted to the feasible region of $N(q)$, which avoids construction of unacceptable surface triangulations. It all will be carried out using an approximate version of the similarity transformation given in (11).

Let us consider that $\mathbf{x} = (x, y)^T$ is the position vector of the free node q, sited on the plane P. If we suppose that Σ is parametrized by $\mathbf{s}(x, y) = (x, y, f(x, y))$, then, the position of the free node p on the surface is given by $\mathbf{y} = (x, y, f(x, y))^T = (\mathbf{x}, f(\mathbf{x}))^T$.

Note that $S_E = A_P W_E^{-1}$ only depends on $\mathbf{x}$ because W_E is constant and A_P is a function of $\mathbf{x}$. Besides, $S_I = A_P W_I^{-1}$ depends on $\mathbf{y}$, due to $W_I = A_P R^{-1} W_E$, and R is a function of $\mathbf{y}$. Thus, $S_I(\mathbf{y})$ depends on the parametrization of the surface, but $S_E(\mathbf{x})$ does not. We shall optimize the local mesh $M(p)$ by an iterative procedure maintaining constant $W_I(\mathbf{y})$ in each step. To do this, at the first step, we fix $W_I(\mathbf{y})$ to its initial value, $W_I^0 = W_I(\mathbf{y}^0)$, where $\mathbf{y}^0$ is given by the initial position of p. So, if we define $S_I^0(\mathbf{x}) = A_P(\mathbf{x})(W_I^0)^{-1}$, we approximate the similarity transformation (11) as

$$S^0(\mathbf{x}) = S_E^{-1}(\mathbf{x})\, S_I^0(\mathbf{x})\, S_E(\mathbf{x}) \tag{12}$$

Now, the construction of the objective function is carried out in a standard way, but using S^0 instead of S. Following the same procedure, pointed out at the beginning of this section, we obtain the objective

function for a given triangle $\tau \subset \pi$

$$\eta^0(\mathbf{x}) = \frac{\left|S^0(\mathbf{x})\right|^2}{2\sigma^0(\mathbf{x})} \tag{13}$$

where $\sigma^0(\mathbf{x}) = \det(S^0(\mathbf{x}))$.

With this approach the optimization of the local mesh $M(p)$ is transformed into a two-dimensional problem without constraints, defined on $N(q)$, and, therefore, it can be solved with low computational cost. Furthermore, if we write W_I^0 as $A_P^0 (R^0)^{-1} W_E$, where $A_P^0 = A_P\left(\mathbf{x}^0\right)$ and $R^0 = R\left(\mathbf{y}^0\right)$, it is straightforward to show that S^0 can be simplified as

$$S^0(\mathbf{x}) = R^0\left(A_P^0\right)^{-1} S_E(\mathbf{x}) \tag{14}$$

In fact, this is the expression used to construct our objective function for the local mesh

$$\left|K_\eta^0\right|_n(\mathbf{x}) = \left[\sum_{m=1}^{M}\left(\eta_m^0\right)^n(\mathbf{x})\right]^{\frac{1}{n}} \tag{15}$$

Let now analyze the behavior of the objective function when the free node crosses the boundary of the feasible region. If we denote $\alpha_P = \det(A_P)$, $\alpha_P^0 = \det\left(A_P^0\right)$, $\rho^0 = \det\left(R^0\right)$, $\omega_E = \det(W_E)$ and taking into account (14), we can write $\sigma^0 = \rho^0\left(\alpha_P^0\right)^{-1}\alpha_P \omega_E^{-1}$. Note that ρ^0, α_P^0, and ω_E are constants, so η^0 has a singularity when $\alpha_P = 0$, that is, when q is placed on the boundary of the feasible region of $N(q)$. This singularity determines a barrier in the objective function that prevents the optimization algorithm from taking the free node outside this region. This barrier does not appear if we use the exact weighted Jacobian matrix S, given in (6), due to $\det(R) = R_{11}R_{22} > 0$.

Suppose that $\mathbf{x}^1 = \overline{\mathbf{x}}^0$ is the minimizing point of (15). As this objective function has been constructed by keeping $\mathbf{y}$ in its initial position, $\mathbf{y}^0$, then $\mathbf{x}^1$ is only the first approximation to our problem. This result is improved updating the objective function at $\mathbf{y}^1 = (\mathbf{x}^1, f(\mathbf{x}^1))^T$ and, then, computing the new minimizing position, $\mathbf{x}^2 = \overline{\mathbf{x}}^1$. This local optimization process is repeated,

obtaining a sequence $\left\{\mathbf{x}^k\right\}$ of optimal points. The Algorithm 1 summarize the previous local mesh optimization procedure. Due to the regularity of the objective function $\left|K_\eta^k\right|_n(\mathbf{x})$ inside the feasible region, any standard and efficient unconstrained optimization method can be used. In particular, we have applied BFGS method [12] to minimize this objective function.

At present, we have not found a general theoretical proof about the convergence of Algorithm 1, but we have experimentally verified its convergence in numerous tests involving continuous functions to define the surface Σ. As was to be expected, the convergence rate of Algorithm 1 depends on the grade of oscillations of reference surface in the local region around the free node. In concrete, if surface Σ is a plane, then the ideal triangle t_I does not depends on the position of the free node. Therefore, the matrix W_I associated to the affine map $t_R \overset{W_I}{\rightarrow} t_I$ is constant. This implies that $S_I^k(\mathbf{x}) = A_P(\mathbf{x})(W_I^k)^{-1} = A_P(\mathbf{x})(W_I^0)^{-1} = S_I^0(\mathbf{x})$ and, taking into account equation (12), $S^k(\mathbf{x}) = S^0(\mathbf{x})$. In consequence, Algorithm 1 converges in one iteration since the objective functions satisfy $\left|K_\eta^k\right|_n(\mathbf{x}) = \left|K_\eta^0\right|_n(\mathbf{x})$ for any k.

To prevent exceptional cases in which convergence is not achieved, we have fixed a maximum number of iterations in our code for terminating the Algorithm 1. In section 4.2 we will show results about convergence for a test problem.

Let us consider P as an optimal projection plane (this aspect will be discussed in next section). In order to prevent a loss of the details of the original geometry, our optimization algorithm evaluates the difference of heights ($[\Delta z]$) between the centroid of the triangles of $M(p)$ and the reference surface, every time a new position $\mathbf{x}^k$ is calculated. If this distance exceeds certain threshold, $\Delta(p)$, the movement of the node is aborted and the previous position is stored. This threshold $\Delta(p)$ is established attending to the size of the elements of $M(p)$. In concrete, the algorithm evaluates the average distance between the free node and the nodes connected to it, and takes $\Delta(p)$ as percentage of this distance.

Algorithm 1 Local optimization

Let $\varepsilon > 0$ be the stopping criterion;

Let $\mathbf{y}^0 = (\mathbf{x}^0, f(\mathbf{x}^0))^T$ be the initial position of the free node on Σ;

$S^0(\mathbf{x})$ and $\eta^0(\mathbf{x})$ are constructed by (14) and (13), respectively, for all triangles of $M(p)$;

$\left|K_\eta^0\right|_n(\mathbf{x})$ is constructed by (15) using $\eta^0(\mathbf{x})$;

Let $\overline{K}^0 = \left|K_\eta^0\right|_n\left(\overline{\mathbf{x}}^0\right) = \min_{\mathbf{x}\in\mathcal{H}}[\left|K_\eta^0\right|_n(\mathbf{x})]$, where $\mathcal{H}$ is the feasible region of $N(q)$;

$\mathbf{x}^1 = \overline{\mathbf{x}}^0$;

$\mathbf{y}^1 = (\mathbf{x}^1, f(\mathbf{x}^1))^T$;

$k = 1$;

$convergence = false$;

while $(convergence = false)$ **do**

$S^k(\mathbf{x})$ and $\eta^k(\mathbf{x})$ are constructed by (14) and (13), respectively, for all triangles of $M(p)$, fixing

$\mathbf{y} = \mathbf{y}^k$;

$\left|K_\eta^k\right|_n(\mathbf{x})$ is constructed by (15) using $\eta^k(\mathbf{x})$;

Let $\overline{K}^k = \left|K_\eta^k\right|_n\left(\overline{\mathbf{x}}^k\right) = \min_{\mathbf{x}\in\mathcal{H}}[\left|K_\eta^k\right|_n(\mathbf{x})]$;

if $(\left|\frac{\overline{K}^k - \overline{K}^{k-1}}{\overline{K}^k}\right| < \varepsilon)$ **then**

$convergence = true$;

end if

$\mathbf{x}^{k+1} = \overline{\mathbf{x}}^k$;

$\mathbf{y}^{k+1} = (\mathbf{x}^{k+1}, f(\mathbf{x}^{k+1}))^T$;

$k = k + 1$;

end while

The optimal solution is $\mathbf{x}^k$;

Another possibility is to fix $\Delta(p)$ as a constant for all local meshes. In the particular case in which we have an explicit representation of the surface by a function $f(x, y)$, $\Delta(p)$ can be established as a percentage of the maximum difference of heights between the original surface and the initial mesh.

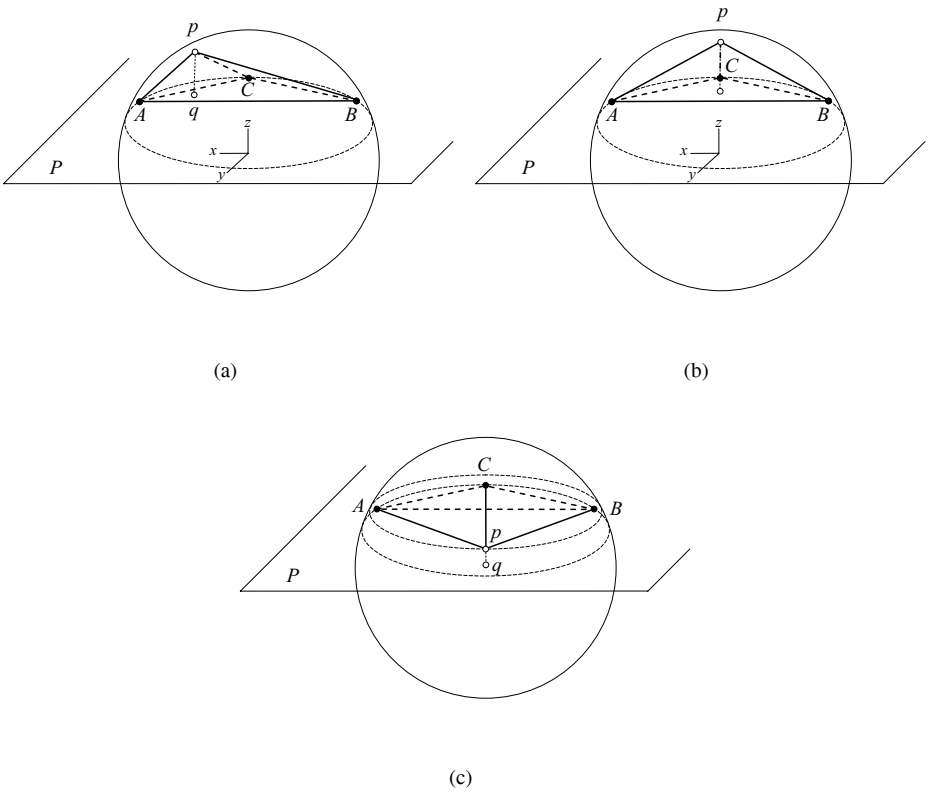


Figure 3. Initial surface mesh formed by the triangles ABp, BCp and ACp sited on a sphere (a). Optimized triangulation with a valid parametric mesh $N(q)$ (b). Optimized triangulation with a tangled parametric mesh (c).

3. SEARCH FOR THE OPTIMAL PROJECTION PLANE

The former procedure needs a plane in which the local mesh, $M(p)$, is projected conforming a valid mesh, $N(q)$. If this plane exists it is not unique, because a small rotation of the coordinate system

produces another valid projection plane, that is, another plane in which $N(q)$ is valid. We have observed that the number of iterations required by the Algorithm 1 depends on the chosen plane. In general, this number is less if the plane is well *faced* to $M(p)$. We have to find the rotation of reference system x, y, z such that the new x', y'-plane, P', is optimal with respect to a suitable criterion.

We will denote $N(q')$ as the projection of $M(p)$ onto P' and t' the projection of the physical triangle $\tau \in M(p)$ onto P'. Let $A'_P = (\mathbf{x}'_1 - \mathbf{x}'_0, \mathbf{x}'_2 - \mathbf{x}'_0)$ be the matrix associated with the affine map that takes the reference element defined on P' to t', then, the area of t' is given by $\frac{1}{2}|\alpha'_P|$ where $\alpha'_P = \det(A'_P)$.

Our goal is to find a coordinate system rotation such that $\sum_{m=1}^{M} \alpha'_{P_m}$ is maximum satisfying the constraints $\alpha'_{P_m} = \det\left(A'_{P_m}\right) > 0$ for all the triangles of $N(q')$, that is, $m = 1, ..., M$. Note that, to do possible the observance of the constraints, it is necessary that all triangles of $M(p)$ have the same orientation in the original mesh. In [13] a method to determine a projection plane is considered but without the enforcement of these constraints. We remark that in our problem it is essential to maintain the constraints in order to get $N(q')$ valid.

According to Euler's rotation theorem, any rotation may be described using three angles. The so-called *x-convention* is the most common definition. In this convention, the rotation is given by Euler angles (ϕ, θ, ψ), where the first rotation is by an angle $\phi \in [0, 2\pi]$ about the z-axis, the second is by an angle $\theta \in [0, \pi]$ about the x-axis, and the third is by an angle $\psi \in [0, 2\pi]$ about the z-axis (again).

Let $\Phi(\phi, \theta, \psi)$ be the Euler's rotation matrix such that $\mathbf{y}' = \Phi\mathbf{y}$, then, the Jacobian matrix $A_\pi = (\mathbf{y}_1 - \mathbf{y}_0, \mathbf{y}_2 - \mathbf{y}_0)$ associated to the triangle τ of $M(p)$, defined in (2), can be spanned on the rotated coordinate system as $A'_\pi = (\mathbf{y}'_1 - \mathbf{y}'_0, \mathbf{y}'_2 - \mathbf{y}'_0) = \Phi A_\pi$. Thus, the Jacobian matrix A'_P is written as $A'_P = \Pi A'_\pi = \Pi\Phi A_\pi$. With these considerations it is easy to prove that the value of α'_P is

$$\alpha'_P = \det(\Pi\Phi A_\pi) = m_1 \sin(\phi)\sin(\theta) + m_2 \sin(\theta)\cos(\phi) + m_3 \cos(\theta) \tag{16}$$

where m_i is the minor obtained by deleting the i-th row of A_π. Note that equation (16) only depends

on ϕ and θ angles, as was to be expected.

Although the above maximization problem can be solved taken into account the constraints, we propose an unconstrained approach.

Let us consider, as a first attempt, the objective function $\sum_{m=1}^{M} (\alpha'_{P_m})^{-1}(\phi, \theta)$. The minimization of this function tends to maximize the values of α'_{P_m} and, due to the barrier that appears when $\alpha'_{P_m} = 0$ for some triangle of $N(q')$, the values of α'_{P_m} are maintained positive if the minimization algorithm starts at an interior point, that is, a point (ϕ_0, θ_0) belonging to the set Ψ of angles (ϕ, θ) such that $\alpha'_{P_m}(\phi, \theta) > 0$ for $(m = 1, ..., M)$. On the other hand, if any $\alpha'_{P_m} < 0$ the barrier prevents reaching the required minimum. In next paragraph we propose a method to find an interior point (ϕ_0, θ_0) of Ψ to be used as a starting point in the minimization algorithm.

Let $G = [\mathbf{g}_m]$ be the $3 \times M$ matrix formed by the vectors, $\mathbf{g}_m$, normal to the triangles of $M(p)$. A solution of the inequality system (if it exists) $G^T \mathbf{g} > \mathbf{0}$ provides a direction, defined by vector $\mathbf{g}$, such that all the triangles of $M(p)$ can be projected on a plane, normal to the unitary vector $\mathbf{n} = \frac{\mathbf{g}}{\|\mathbf{g}\|}$, so that $\alpha'_{P_m} > 0$ for $(m = 1, ..., M)$. Then, it only remains to find the angles ϕ_0 and θ_0 in which the coordinate system needs to be rotated to get the z' axis to point in the direction of $\mathbf{n}$. More precisely, the angles ϕ_0 and θ_0 are the solution of the equation $\Phi^T(\phi_0, \theta_0, 0)\, \mathbf{e}_3 = \mathbf{n}$, where $\mathbf{e}_3 = (0, 0, 1)^T$. If the inequality system has no solution, then, there is no valid projection plane for this local mesh, against the premise done in section 2.1. In this case, the local mesh optimization procedure maintain the free node p at its initial position.

We have observed that the previous objective function has computational difficulties as the optimization algorithms use discrete steps to search for the optimal point. A step leading outside the region Ψ may indicate a decrease in the value of the objective function and take to a false solution. To overcome this problem we propose a modification of the objective function in such a way that it will be

regular all over $\mathbb{R}^3$ and its barrier will be "smoothed". The modification consists of substituting ${\alpha'}_{P_m}$ by $h(\alpha_{P_m})$, where $h(\alpha)$ is the positive and increasing function given by

$$h(\alpha) = \frac{1}{2}(\alpha + \sqrt{\alpha^2 + 4\delta^2}) \tag{17}$$

being the parameter $\delta = h(0)$. The behavior of $h(\alpha)$ in function of δ parameter is such that, $\lim_{\delta \to 0} h(\alpha) = \alpha$, $\forall \alpha \geq 0$ and $\lim_{\delta \to 0} h(\alpha) = 0$, $\forall \alpha \leq 0$. The characteristics of h function and its application in the context of mesh untangling and smoothing have been studied in [14]. Thus, the proposed objective function for searching the projection plane is

$$\Omega(\phi, \theta) = \sum_{m=1}^{M} \frac{1}{h(\alpha'_{P_m}(\phi, \theta))} \tag{18}$$

A crucial property is that the angles that minimize the original and modified objective functions are nearly identical when δ is *small*. Details about the determination of δ value for 3-D triangulations can be found in [14]. Note that $h(\alpha)$ is an increasing function that makes Ω tend to ∞ when the area of any triangle of $N(q')$ tends to $-\infty$, since $\lim_{\alpha \to -\infty} h(\alpha) = 0$. This last property makes the function Ω continues increasing when the sign of ${\alpha'}_{P_m}$ changes from positive to negative and making it very difficult to leave Ψ.

4. EXAMPLES AND APPLICATIONS

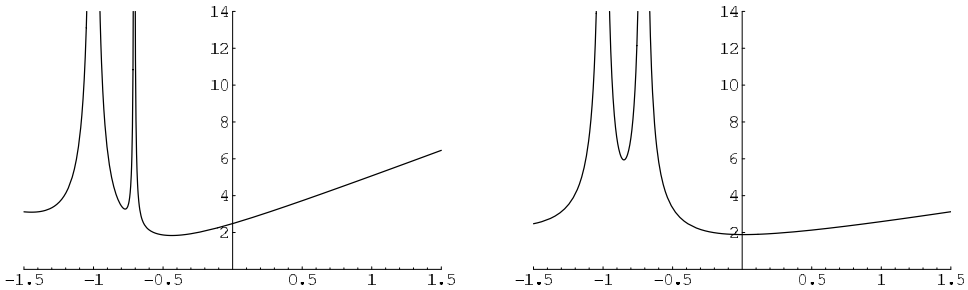
This section is divided in four subsections. In the former two subsections we analyze test problems in order to show the behavior of the Algorithm 1. In the third subsection we consider an orographic surface defined by a terrain. Finally, in the last subsection, the proposed technique is applied to smooth the mesh of a scanned object. The norm chosen for the objective function (1) in all these applications have been $n = 2$.

4.1. Test problem 1

The first test problem is a practical example based on the situation shown in Figure 3 and commented in section 2.2. This example shows the differences between our proposed objective functions $\left|K_\eta^k\right|_2$ and the *exact version*, $|K_\eta|_2$, constructed by S^k (14) and S (6), respectively. Let us suppose that the mesh nodes of Figure 3 are sited on a sphere with radium 4 and centered in $(0,0,0)$. The positions of nodes are $A(\frac{1}{\sqrt{2}},-\frac{1}{\sqrt{2}},\sqrt{15})$, $B(-\frac{1}{\sqrt{2}},-\frac{1}{\sqrt{2}},\sqrt{15})$ and $C(0,-1,\sqrt{15})$, and the initial position of the free node is $p(0,-0.717,3.935)$. Firstly, this mesh is optimized by using the iterative procedure presented in Algorithm 1 with the approximate objective function $\left|K_\eta^k\right|_2$ and, after, by using the direct minimization of $|K_\eta|_2$. Thus, in Figure 4 you can see the transversal cuts of $\left|K_\eta^0\right|_2$ and $\left|K_\eta^3\right|_2$ through the y,z-plane. In concrete, Figure 4(a) shows the initial objective function $\left|K_\eta^0\right|_2$ in terms of y and Figure 4(b) shows the similar representation of the objective function at the third iteration of Algorithm 1, $\left|K_\eta^3\right|_2$. We present in Figure 5(a) the exact objective function $|K_\eta|_2$ for this test problem. The presence of barriers at $y=-1$ and $y=-\frac{1}{\sqrt{2}}$ in the functions of Figures 4(a) and (b) maintains the free node inside the feasible region. Nevertheless, it does not necessarily occur if we minimize the exact objective function (see Figure 5(a)) due to the barrier at $y=-\frac{1}{\sqrt{2}}$ does not exist. In present case, the minimizing position of the exact objective function is $p(0,0.025,4.0)$, that is, outside the feasible region. For other initial position of the free node the minimizing point could be inner to the feasible region, for example, if we start at $p(0,-0.75,3.929)$ we obtain the minimum at $p(0,-0.848,3.909)$. On the other hand, Algorithm 1 converges to $p(0,-0.850,3.909)$, therefore, inside of the feasible region. The same minimum is reached if it start at any other position of the free node, whenever this one is inner to the feasible region. We remark that, in this last situation, minimizing points obtained by exact and approximated objective functions are near the same.

Finally, in Figure 5(b) we show the behavior of exact objective function for a similar test to former

one, but taking the radium of the sphere $r = 1.5$. The positions of nodes are now $A(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, \frac{\sqrt{5}}{2})$, $B(-\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, \frac{\sqrt{5}}{2})$ and $C(0, -1, \frac{\sqrt{5}}{2})$, and the initial position of the free node is $p(0, -0.717, 1.317)$. In this case there is not any local minimum inner to the feasible region and, then, any initial position of the free node leads to minimizing point $p(0, -0.061, 1.499)$ external to the feasible region. Nevertheless, our algorithm always converges to a minimizing point $p(0, -0.825, 1.252)$, sited inside of the feasible region.

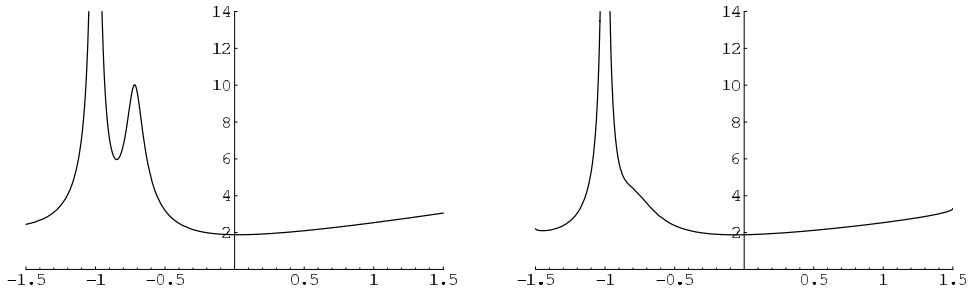


(a) (b)

Figure 4. Transversal cut of $\left|K_\eta^0\right|_2$ through the y, z-plane in function of y for the mesh of Figure 3 formed by nodes A, B, C and p sited on a sphere of radium $r = 4$ (a). The objective function after three iterations, $\left|K_\eta^3\right|_2$, for the previous mesh (b).

4.2. Test problem 2

To understand other aspects of the behavior of Algorithm 1, now we choose a simple mesh with six triangles placed on the paraboloid given by $f(x, y) = \frac{5}{4}(x^2 + (y-1)^2)$. The projection of this mesh on the plane $z = 0$ forms a mesh with all the triangles equilateral. The positions of the fixed nodes on the paraboloid are $\mathbf{y}_1 = (0, -1, 5)^T$, $\mathbf{y}_2 = (\frac{\sqrt{3}}{2}, -\frac{1}{2}, \frac{15}{4})^T$, $\mathbf{y}_3 = (\frac{\sqrt{3}}{2}, \frac{1}{2}, \frac{5}{4})^T$, $\mathbf{y}_4 = (0, 1, 0)^T$, $\mathbf{y}_5 = (-\frac{\sqrt{3}}{2}, \frac{1}{2}, \frac{5}{4})^T$, and $\mathbf{y}_6 = (-\frac{\sqrt{3}}{2}, -\frac{1}{2}, \frac{15}{4})^T$, and the initial position of the free node is $\mathbf{y}_0 = (0, 0, \frac{5}{4})^T$. At the bottom of Figure 6(a) it is presented the projection of the mesh

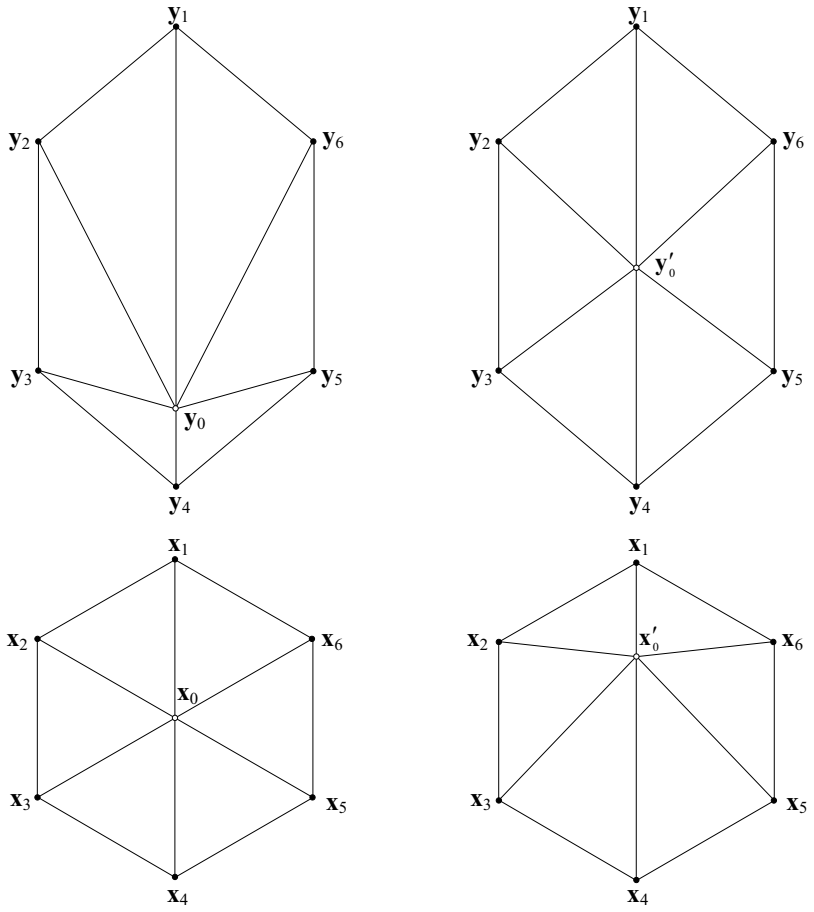


(a) (b)

Figure 5. Exact objective function for a mesh sited on a $r = 1.5$ sphere (b) Transversal cut by y, z-plane of the exact objective function $|K_\eta|_2$ as a function of y, for the mesh of Figure 3 formed by nodes A, B, C and p sited on spheres of radium $r = 4$ (a) and $r = 1.5$ (b).

on plane $z = 0$ and, at the top, the projected mesh on the tangent plane to the paraboloid at $\mathbf{y}_0$. The corresponding meshes after three steps of Algorithm 1 are shown in 6(b). Although the quality of the mesh on plane $z = 0$ has decreased after the process, the mesh quality on the paraboloid has increased. In concrete, the minimum mesh quality on the paraboloid goes from 0.476 to 0.600 and the average quality goes from 0.642 to 0.668. In this application we have used the exact representation of the surface to calculate the z-coordinate of the free node at each step of Algorithm 1, but very close results are obtained using the approximate surface defined by the linear interpolation of the initial mesh.

To check the convergence of the local process we choose the former application first with an exact representation of the surface, and then by using the linear interpolation. The logarithm of the relative error used as stopping criterium in Algorithm 1, $\log\left(\left|\frac{\overline{K}^k - \overline{K}^{k-1}}{\overline{K}^k}\right|\right)$, is represented in terms of the number of iterations for both cases in Figure 7. We have observed a similar behavior in our experimental test.



(a) (b)

Figure 6. Representations of the mesh constructed on the paraboloid $f(x,y) = \frac{5}{4}(x^2 + (y-1)^2)$. At the bottom of figure (a) is shown the projection on $z = 0$ for the initial position of free node (in white). At the top, it is represented the projection on the tangent plane to the paraboloid at $\mathbf{y}_0$ (an approximate frontal view). In figure (b) the same projections are shown with the free node at the final position after three iterations of Algorithm 1.

4.3. Application to orographic surfaces

The numerical simulation of environmental problems, as wind field adjustment [16] or air pollution modeling, require 3-D meshes adapted to complex terrains [15]. The aim is to create a tetrahedral

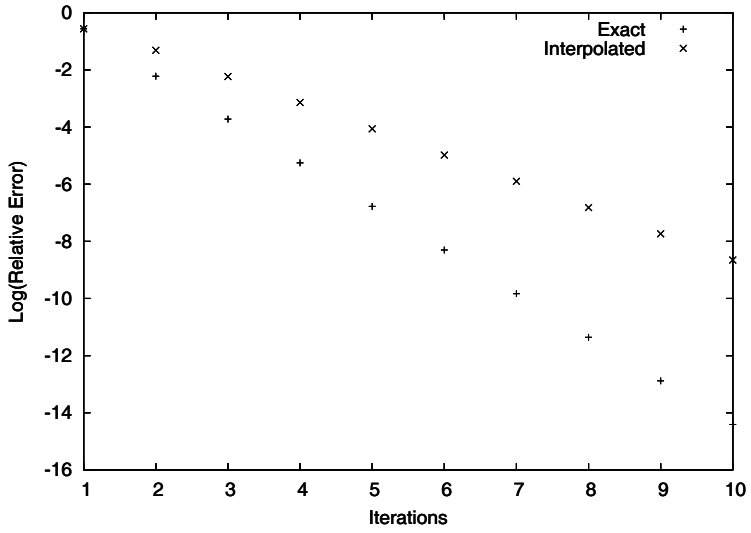


Figure 7. Convergence of Algorithm 1 for the test problem 2. Logarithm of the relative error in terms of the number of iterations.

mesh of a region bounded in its lower part by the terrain and in its upper part by a horizontal plane. To do this we make a 3-D Delaunay triangulation of a previously established distribution of points, whose density increases with the complexity of the orography. The point generation in the domain is done over different layers defined from the terrain to the upper part of the domain. The adaptive position of nodes in the terrain surface is automatically determined by applying a 2-D refinement/derefinement algorithm of nested meshes. To avoid conforming problems between mesh and orography, the tetrahedral mesh will be designed with the help of an auxiliary parallelepiped, in such a way that every terrain node is projected on its lower plane. Once the 3-D Delaunay triangulation of the set of points has been constructed on the parallelepiped, points are replaced on their real positions keeping the mesh topology [15]. In this last stage there can be occasional low quality elements, or even inverted elements. In particular, the triangles of the terrain, that initially was rectangle on the lower plane of the auxiliary parallelepiped, result deformed and the quality of the surface mesh decrease. Therefore, it is necessary

to apply untangling and smoothing procedures. Here, we propose to solve the problem in two stages. In the first one the surface mesh is optimized and, in the second one, the 3-D mesh is untangled and smoothed maintaining fixed the surface nodes. This last problem already has been solved in [14], therefore, it only remain to undertake the first task.

The surface defined by the terrain is very irregular, so it is difficult to find a smooth function that properly interpolates this surface. For this reason, we have used a linear interpolation on the initial mesh to obtain a reference surface. The particular way in which the surface mesh has been constructed make it possible to choose the same parametric plane P for all the local meshes, but in order to compare results, we have used two strategies in this application: using optimal local projection planes and fixing the $z = 0$ plane. The results are shown below.

We have applied our smoothing technique to a terrain triangulation of NW of Gran Canaria island. The dimensions of the domain are 16.5×9.5 *km* and it has been discretized by using a digitalization of the terrain with a resolution of 25×25 *m* and a maximum error of 5 *m* in height. In Figure 8 it is shown the relief of this part of the island. A projection on $z = 0$ plane of the original mesh, obtained by our mesh generator [15], is shown in Figure 9(a). Note that, due to existence of many ravines and cliffs, it was necessary to refine several times in neighborhood of these areas in order to capture the real orography. The average quality of this original mesh, measured with the condition number metric [3], is 0.86. The corresponding smoothed mesh after four stages of the optimization procedure, using optimal projection planes, is shown in Figure 9(b). Its average quality has been increased until 0.90 (the same result is obtained by using the $z = 0$ parametric plane). This improvement is not very great since the initial quality is close to the maximum quality, 1. The average quality of the worst 200 triangles has been increased from 0.55 until 0.66 (0.59 using the $z = 0$ parametric plane). The minimum quality have changed from 0.37 until 0.39 (0.37 with $z = 0$ parametric plane). As it was expected, the use of

optimal projection planes improves the results.

We have allow a 10% over the average distance between the free node and the nodes connected to it to establish the local threshold $\Delta(p)$. With this strategy, the maximum gap detected between the reference surface and the optimized one has been 9.4 *m* (respect to z' direction) for the application obtained by choosing oriented parametric planes and 10.9 *m* for the application with fixed $z = 0$ parametric plane.

4.4. Application to a scanned object

We have applied the optimization technique to a mesh obtained from the Large Geometric Model Archives at Georgia Institute of Technology. Now, the projection plane is chosen in terms of the local mesh to be analyzed. The mesh (see Figure 10) has 96966 triangles and 48485 nodes. The value of the average quality is 0.82 (measured with the quality metric based on the condition number [3]). The optimized mesh is shown in Figure 11. Note the poor quality of the original mesh in several parts of the neck of the horse in Figure 12(a). After five iterations of our optimization procedure the mesh has been smoothed, as it can be seen in Figure 12(b). To show that our surface smoothing procedure prevents a loss of sharp features, we present a detail of the horse's ears for the initial and optimized meshes in Figure 13. Its average quality has been increased to 0.91. A more significant data is the increment in the average quality of the worst 1000 triangles: its initial value is 0.17 and the final value is 0.63. The quality curves for the initial and optimized meshes are shown in Figure 14. These curves are obtained by sorting the elements in increasing order of its quality, $q(e)$.

The stopping criteria for Algorithm 1 used in this application has been $\varepsilon = 10^{-2}$, and the most frequent number of iteration to achieve the convergence has been one. In general, we have observed that the computational cost for finding a locally optimal plane has been less than the one for optimizing

the local mesh with Algorithm 1.

Choosing the same strategy and tolerance than in the previous example for evaluating the local threshold $\Delta(p)$, we have obtained, by using Gauss's Theorem, the volume of the horse before and after smoothing, resulting 0.0002634 and 0.0002633, respectively. That means a decrease in the volume of 0.038%.

5. CONCLUSIONS AND FUTURE RESEARCH

We have developed an algebraic method to optimize triangulations defined on surfaces. Its main characteristic is that the original problem is transformed into a fully two-dimensional sequence of approximate problems on the parametric space. This characteristic allows the optimization algorithm to deals with surfaces that only need to be continuous. Moreover, the barrier exhibited by the objective function in the parametric space prevents the algorithm to construct unacceptable meshes. This would not be assured if working on the real mesh. Indeed, our procedure can be used to optimize the boundary of a 3-D mesh, although in this case, the node movement on the surface can produce a tangle in the 3-D mesh, making necessary a subsequent untangling and smoothing procedure.

We have also developed a procedure to find an optimal projection plane (our parametric space) based on the minimization of a suitable objective function. We have observed in many test that the number of iterations carried out for Algorithm 1 is less as the projection plane is better faced to the local mesh $M(p)$, so the correct choice of this plane plays a relevant role. Moreover, we have also observed in problems in which it is possible to choose a unique projection plane (as those defined by orographic surfaces) that the resulting mesh, after a fixed number of iterations, has better quality when we work with optimal parametric planes.

The optimization process includes a control on the gap between the optimized mesh and the reference

surface that avoids losing details of the original geometry. In this work we have used a piecewise linear interpolation to define the reference surface when the true surface is not known, but it would be also possible to use a more regular interpolation, for example, the proposed in [7]. Likewise, it would be possible to introduce a more sophisticated stopping criterion for the gap control that takes into account the curvature of the surface [6], [7], [8], [9].

In the present work we have only considered a sole objective function obtained from an isotropic and area independent algebraic quality metric. Nevertheless, the framework that establishes the *algebraic quality measures* [2] provides us the possibility to construct anisotropic and area sensitive objective functions by using a suitable metric.

ACKNOWLEDGEMENTS

This work has been supported by the Spanish Government and FEDER, grant contracts: REN2001-0925-C03-02/CLI and CGL2004-06171-C03-02/CLI.

REFERENCES

1. Berzins M. Mesh quality: A function of geometry, error estimates or both? *Engineering with Computers* 1999; **15**(3):236-247. DOI: 10.1007/s003660050019
2. Knupp PM. Algebraic mesh quality metrics. *SIAM Journal on Scientific Computing* 2001; **23**(1):193-218. DOI: 10.1137/S1064827500371499
3. Freitag LA, Knupp PM. Tetrahedral mesh improvement via optimization of the element condition number. *International Journal for Numerical Methods in Engineering* 2002; **53**(6):1377-1391. DOI: 10.1002/nme.341
4. Frey PJ, Borouchaki H. Surface mesh quality evaluation. *International Journal for Numerical Methods in Engineering* 1999; **45**(1):101-118. DOI: 10.1002/(SICI)1097-0207(19990510)45:1<101::AID-NME582>3.0.CO;2-4
5. Garimella RV, Shaskov MJ, Knupp PM. Triangular and quadrilateral surface mesh quality optimization using

local parametrization. *Computer Method in Applied Mechanics and Engineering* 2004; **193**(9-11):913-928. DOI: 10.1016/j.cma.2003.08.04

6. Frey PJ, Borouchaki H. Geometric surface mesh optimization. *Computing and Visualization in Science* 1998; **1**(3):113-121. DOI: 10.1007/s007910050011
7. Rassineux A, Villon P, Savignat JM, Stab O. Surface remeshing by local Hermite diffuse interpolation. *International Journal for Numerical Methods in Engineering* 2000; **49**(1-2):31-49. DOI: 10.1002/1097-0207(20000910-20)49:1-2<31::AID-NME921>3.0.CO;2-6
8. Vigo M, Pla N, Brunet P. Directional adaptive surface triangulation. *Computer Aided Geometric Design* 1999; **16**(2):107-126. DOI: 10.1016/S0167-8396(98)00040-5
9. Frey PJ, Borouchaki H. Surface meshing using a geometric error estimate. *International Journal for Numerical Methods in Engineering* 2003; **58**(2):227-245. DOI: 10.1002/nme.766
10. Knupp PM. Achieving finite element mesh quality via optimization of the Jacobian matrix norm and associated quantities. Part I - A framework for surface mesh optimization. *International Journal for Numerical Methods in Engineering* 2000; **48**(3):401-420. DOI: 10.1002/(SICI)1097-0207(20000530)48:3<401::AID-NME880>3.0.CO;2-D
11. Knupp PM. Achieving finite element mesh quality via optimization of the Jacobian matrix norm and associated quantities. Part II - A frame work for volume mesh optimization and the condition number of the Jacobian matrix. *International Journal for Numerical Methods in Engineering* 2000; **48**(8):1165-1185. DOI: 10.1002/(SICI)1097-0207(20000720)48:8<1165::AID-NME940>3.0.CO;2-Y
12. Bazaraa MS, Sherali HD, Shetty CM. *Nonlinear Programing: Theory and Algorithms*. Wiley: New York, 1993. ISBN: 0-471-55793-5
13. Rassineux A, Britkopf P, Villon P. Simultaneous surface and tetrahedron mesh adaption using mesh-free techniques. *International Journal for Numerical Methods in Engineering* 2003; **57**(3):371-389. DOI: 10.1002/nme.681
14. Escobar JM, Rodríguez E, Montenegro R, Montero G, González-Yuste JM. Simultaneous untangling and smoothing of tetrahedral meshes. *Computer Method in Applied Mechanics and Engineering* 2003; **192**(25):2775-2787. DOI: 10.1016/S0045-7825(03)00299-8
15. Montenegro R, Montero G, Escobar JM, Rodríguez E, González-Yuste JM. Tetrahedral mesh generation for environmental problems over complex terrains. *Lecture Notes in Computer Science* 2002; **2329**:335-344.
16. Montero G, Rodríguez E, Montenegro R, Escobar JM, González-Yuste JM. Genetic algorithms for an improved parameter estimation with local refinement of tetrahedral meshes in a wind model. *Advances in Engineering Software 2005;* **36**(1):3-10. DOI: 10.1016/j.advengsoft.2004.03.011

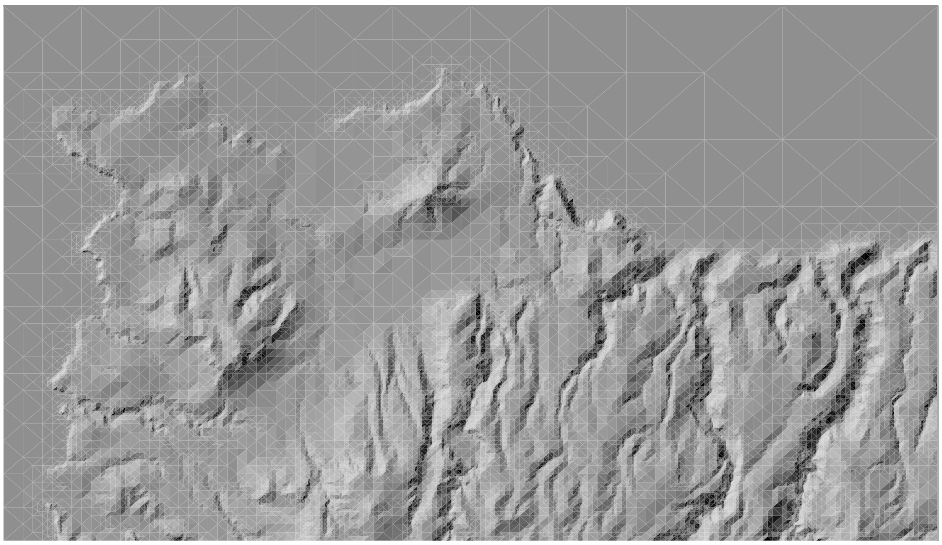
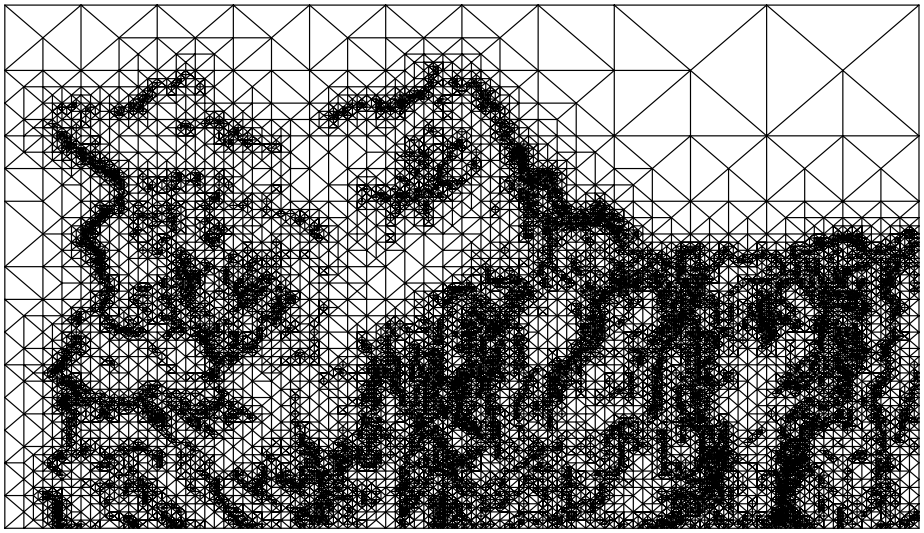
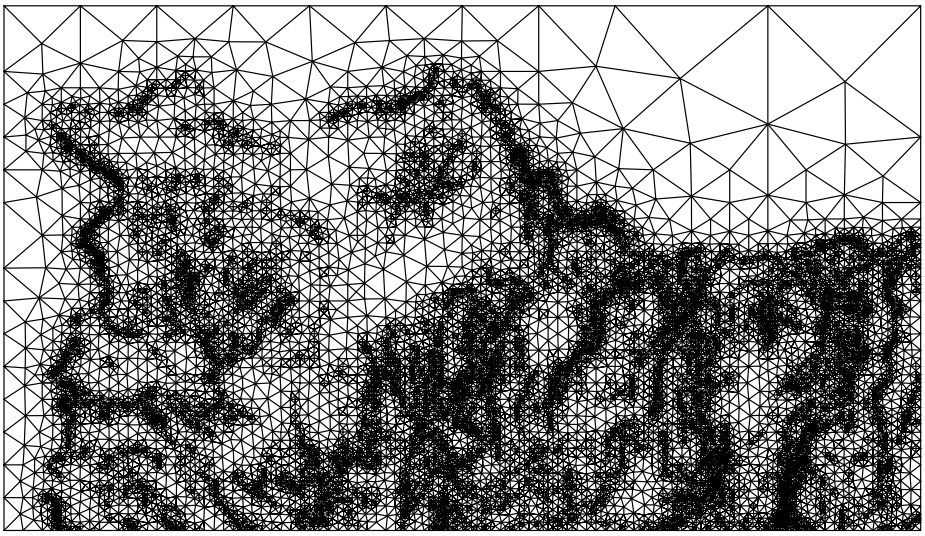


Figure 8. Relief of the NW of Gran Canaria island.



(a)



(b)

Figure 9. Original mesh of the NW of Gran Canaria island (a), and the smoothed mesh after four stages of our optimization procedure (b).

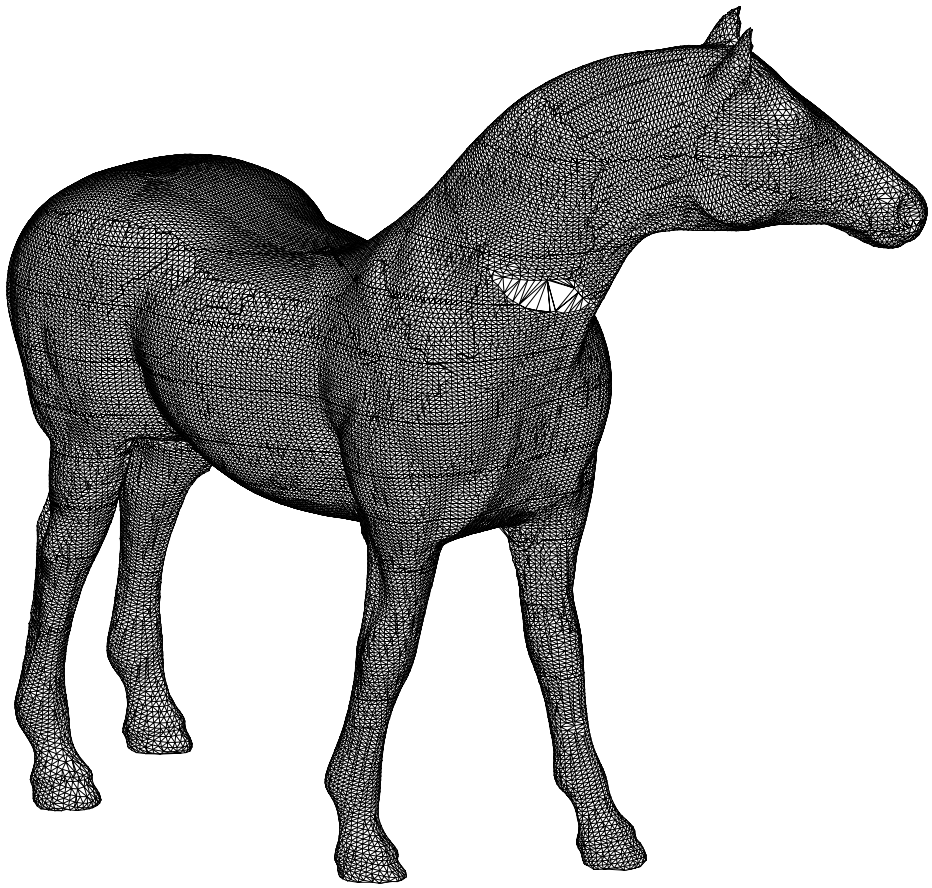


Figure 10. Original mesh of the horse from the Large Geometric Model Archives at Georgia Institute of Technology.

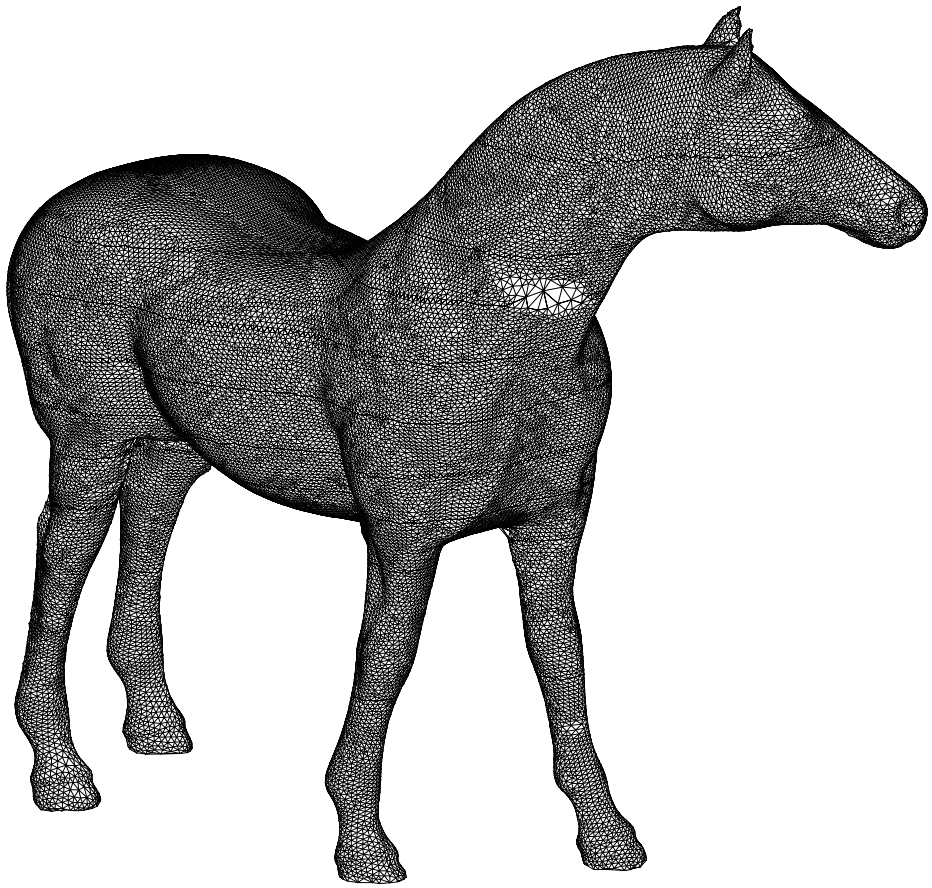
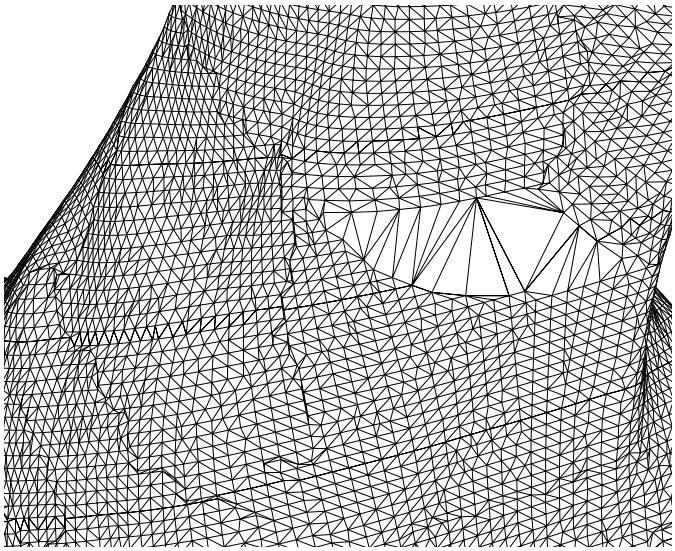
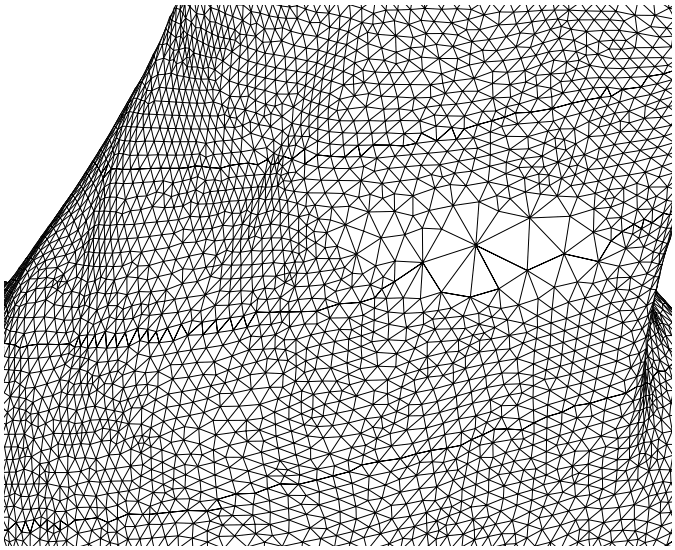


Figure 11. Optimized mesh of the horse after five iterations of our procedure.

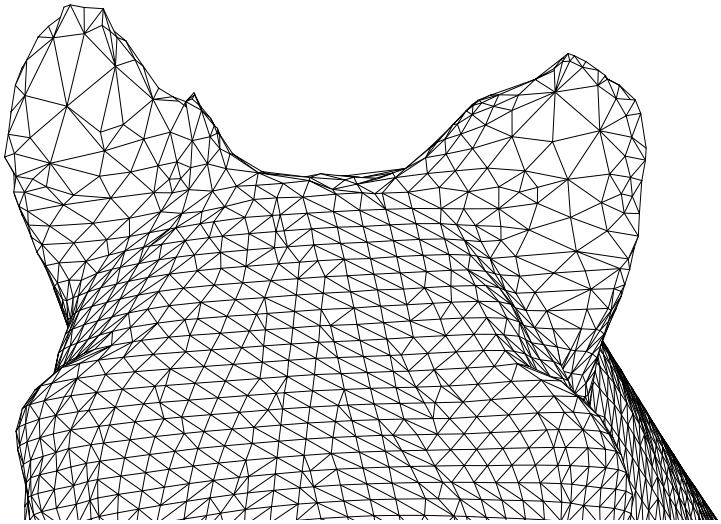


(a)

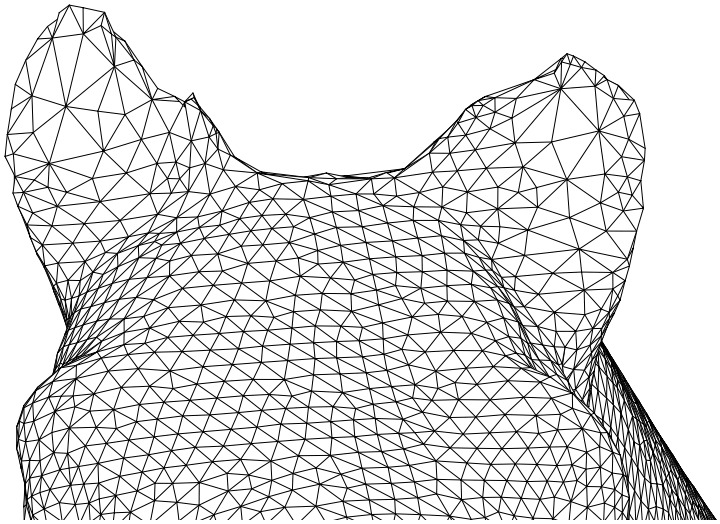


(b)

Figure 12. Detail of the neck of the horse for the initial (a) and optimized (b) meshes.



(a)



(b)

Figure 13. Detail of the horse's ears for the initial (a) and optimized (b) meshes.

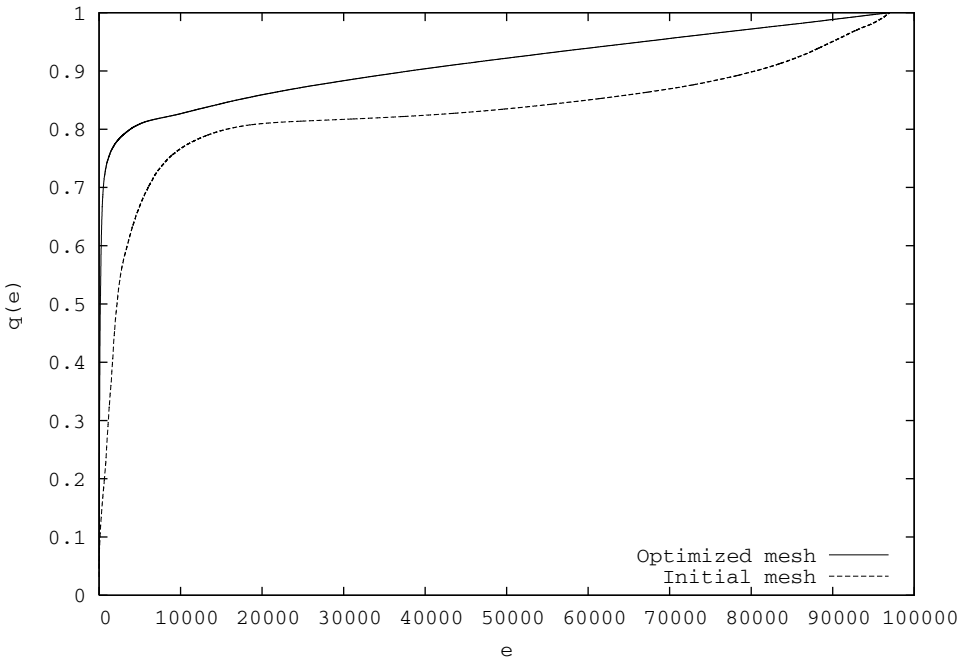


Figure 14. Quality curves of the initial horse mesh and the resulting smoothed mesh after five steps of our optimization process. Function $q(e)$ is the quality measure of the triangle e.