MÉTODOS COMPUTACIONAIS EM ENGENHARIA
Lisboa, 31 de Maio - 2 de Junho, 2004

OPTIMIZATION OF SURFACE MESHES BY PROJECTIONS ON THE PLANE

J.M. Escobar[*] R. Montenegro, G. Montero, E. Rodríguez and J. M. González-Yuste[**]

[*] Departamento de Señales y Comunicaciones
e-mail: jescobar@dsc.ulpgc.es

[**] Instituto de Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería (IUSIANI),
Universidad de Las Palmas de Gran Canaria,
web http://www.iusiani.ulpgc.es

Resumen. *This paper presents a new procedure to improve the quality of triangular meshes defined on surfaces. The improvement is got by an iterative process in which each node of the mesh is moved to a new position that minimizes certain objective function. This objective function is derived from quality measures of the local mesh (the set of triangles connected to the adjustable or free node). If we allow the free node move on the surface without imposing any restrictions, only guided by the improvement of the quality, it can occur that the optimization procedure constructs 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 keeps the free node inside of 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 in which the surface mesh can be projected performing a valid mesh, that is, without inverted elements.*

Palabras clave: mesh generation, mesh smoothing, surface meshes, surface mesh optimization, finite element, adaptive meshes.

1. INTRODUCTION

Although there are many works about the optimization techniques for 2-D or 3-D meshes, the number of papers that deal with the problem of surface mesh optimization is limited. The quality of the surface mesh heavily affect to the numerical behavior of 3-D finite element simulation as it is in this mesh where the boundary conditions are imposed. Moreover, the possible improvement of a 3-D mesh is conditioned by the quality of its surface mesh, so it is very important to develop a technique that allows us to optimize this last. In this work we present a procedure to smooth meshes defined on surfaces. The smoothing technique is based on a vertex repositioning directed by the minimization of an appropriated objective function. The construction of the objective function is done in the framework of theory of *algebraic quality measures* developed in [2]. For 2D or 3D meshes the quality improvement is obtained by an iterative process in which each node of the mesh is moved to a new position that minimizes the objective function [1]. This function is derived from a quality measure of the *local mesh*, that is, the set of triangles connected to the *free node*.

We have chosen, as a starting point, a 2D objective function that presents a barrier in the 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 avoids 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 2D or 3D meshes, but not to surface meshes. To overcome this problem, the local mesh, $M(p)$, sited on a surface Σ, is orthogonally projected on a plane P (if this exists) in such a way that it performs a valid local mesh $N(q)$. Here p is the free node on Σ and q is its projection on P. The optimization of $M(p)$ is got by the appropriated optimization of $N(q)$. To do this we search *ideal* triangles in $N(q)$ that become equilateral in $M(p)$. In general, when the local mesh $M(p)$ is on a curved surface, each triangle is placed on a different plane and it is not possible to define a feasible region. Indeed, it is not clear the concept of valid mesh in this case. This lack of meaning motivates that we assume $M(p)$ as *acceptable* respect to P if $N(q)$ is valid. Note that the feasible region is always perfectly defined in $N(q)$. To construct the objective function in $N(q)$, it is first necessary to define the objective function in $M(p)$ and, afterwards, to establish the connection between them. A crucial aspect for this construction is to keep the barrier of the 2D 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. We develops this approximation in section 2.2.

The optimization of $N(q)$ becomes an iterative process of two-dimensional problems. The optimal solutions of each two-dimensional problem form a sequence $\{\mathbf{x}^k\}$ of points belonging to P. We have checked in may numerical test that $\{\mathbf{x}^k\}$ is always a convergent sequence. We will show an example of this convergence in section 3.1. 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 was smooth. Indeed, the surface determined by the linear interpolation of the initial mesh can be 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 is moved, the optimization process check the distance between the center of the triangles of $M(p)$ and the original surface. If this distance is greater than certain threshold, the movement of the node is aborted and its previous position is stored. Several examples and applications presented in section 3 show how this technique is capable of improving the quality of surface meshes.

2. CONSTRUCTION OF THE OBJECTIVE FUNCTION

As it is shown in [1], [3], and [4] we can derive optimization functions from the *algebraic quality measures* of the triangles belonging to the local mesh. Suppose that we have a triangular mesh defined in a two-dimensional space. Let t be an element in the physical space whose vertices are given by $\mathbf{x}_k = (x_k, y_k)^T \in \mathbb{R}^2$, $k = 0, 1, 2$ and 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 $A = (\mathbf{x}_1 - \mathbf{x}_0, \mathbf{x}_2 - \mathbf{x}_0)$. Let now t_I be an *ideal* triangle (not necessarily equilateral) and let W_I be its Jacobian matrix; then, we define the weighted Jacobian matrix as $S = AW_I^{-1}$. This weighted matrix 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 $|S| = \sqrt{tr\left(S^T S\right)}$, is specially indicated because it is easily computable. Thus, it is shown in [2] that $q_\eta = \frac{3\sigma^{\frac{2}{3}}}{|S|^2}$ is an algebraic quality measure of t , where $\sigma = \det(S)$. The maximum value of q_η is the unity and it is reached when $S = \mu\Theta$, where μ is a nonnegative scalar and $\Theta \in SO(2)$, where $SO(2)$ is the set of all 2×2 orthogonal matrices with determinant 1 (the rotations group). Then, the Jacobian matrix satisfies $A = \mu\Theta W_I$, which means that optimal value of q_η is reached when A is a scale change and a rotation of the Jacobian matrix associated to the ideal triangle t_I. In other words, the triangle that maximizes q_η is similar to t_I. We can derive an objective function from this quality measure. Thus, let $\mathbf{x} = (x, y)^T$ be the position of the free node, and let S_m be the weighted Jacobian matrix of the m-th triangle of the local mesh. The objective function associated to m-th triangle is $\eta_m = \frac{|S_m|^2}{2\sigma_m^{\frac{2}{3}}}$, and the corresponding objective function for the local mesh is the n-norm of $(\eta_1, \eta_2, \ldots, \eta_M)$, that is,

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

where M is the number of triangles in the local mesh. 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 $\mathcal{H}$ defined as $\mathcal{H} = \bigcap_{m=1}^{M} H_m$ where H_m are the half-planes defined by $\sigma \geq 0$, $\mathbf{x} \in \mathbb{R}^2$. We say that a triangle is *inverted* if $\sigma < 0$. The objective function (1) presents a barrier in 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 we should establish beforehand what an inverted element is. We will deal with these questions in next subsection.

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 was a valid mesh $N(q)$. Moreover, suppose that we define the axes in such a way that 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 name $N(q)$ 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 search *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 $\xi_k = (x_k, y_k, z_k)^T \in \mathbb{R}^3$, $k = 0, 1, 2$ and t_R be the reference triangle in P (see Fig. 1). If we choose ξ_0 as the translation vector, the affine map that takes t_R to τ is $\xi = A_\pi \mathbf{u} + \xi_0$, 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, let consider that $t \in N(q)$ is the orthogonal projection of τ on P, then, its vertices are $\mathbf{x}_k = (x_k, y_k)^T \in \mathbb{R}^2$, $k = 0, 1, 2$. Taking $\mathbf{x}_0$ as translation vector, the affine map that takes t_R to t is $\mathbf{x} = A_P \mathbf{u} + \mathbf{x}_0$, and A_P 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 matrix of the affine map that takes t to τ 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 ξ_0. We have to find the *ideal* triangle $t_I \subset P$ such that it was mapped by T into an equilateral one, $\tau_E \subset \pi$.

On the other hand, the Jacobian matrix of the affine map restricted to V_π that takes the reference triangle $\tau_R \subset V_\pi$ to τ_E is

$$W_E = \begin{pmatrix} 1 & 1/2 \\ 0 & \sqrt{3}/2 \end{pmatrix} \tag{5}$$

The factorization of A_π as a product of an orthogonal matrix Q and an upper triangular R with $[R]_{ii} > 0$, yields $A_\pi = QR$. Taken into account that the columns of the 3×2 matrix Q define an orthonormal basis that spans V_π, we can see R as the 2×2 Jacobian matrix of the affine map that takes τ_R to τ (see Fig. 1). Then,

$$QW_E = A_\pi R^{-1} W_E \tag{6}$$

is the Jacobian matrix of τ_E given in the canonical basis of $\mathbb{R}^3$. The weighted Jacobian matrix in V_π is

$$S = RW_E^{-1} \tag{7}$$

The Jacobian matrix, W_I, associated to the *ideal* triangle in P is calculated by imposing the condition

$$TW_I = A_\pi R^{-1} W_E \tag{8}$$

Substituting T, given by (4), we obtain

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

so the *ideal-weighted* Jacobian matrix, defined on P and given by $S_I = A_P W_I^{-1}$, results

$$S_I = A_P W_E^{-1} R A_P^{-1} \tag{10}$$

and, taken into account (7), yields

$$S_I = A_P W_E^{-1} S W_E A_P^{-1} = A_P W_E^{-1} S \left(A_P W_E^{-1} \right)^{-1} = S_E S S_E^{-1} \tag{11}$$

where $S_E = A_P W_E^{-1}$ is the *equilateral-weighted* Jacobian matrix. Finally, from (11), we obtain the next similarity transformation of S

$$S = S_E^{-1} S_I S_E \tag{12}$$

2.2. Optimization of the parametric mesh

We could use S, given in (7), to construct the objective function and solve the optimization problem. Nevertheless this procedure has important disadvantages. In general, when the local mesh $M(p)$ is on a curved surface, each triangle is sited on a different plane, and 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 that make $\det(S) \neq 0$ for all the triangles of $M(p)$ are valid but, maybe, they are not *acceptable*. Thus, it can happen that the optimized mesh, $M(\overline{p})$, was valid but its corresponding parametric mesh, $N(q)$, was not. We will consider this situation as *unacceptable*. As example, in Fig.2(a) is shown a mesh with three triangles, that we suppose sited on a curved surface, for example, a sphere. The optimal position for the free node (in

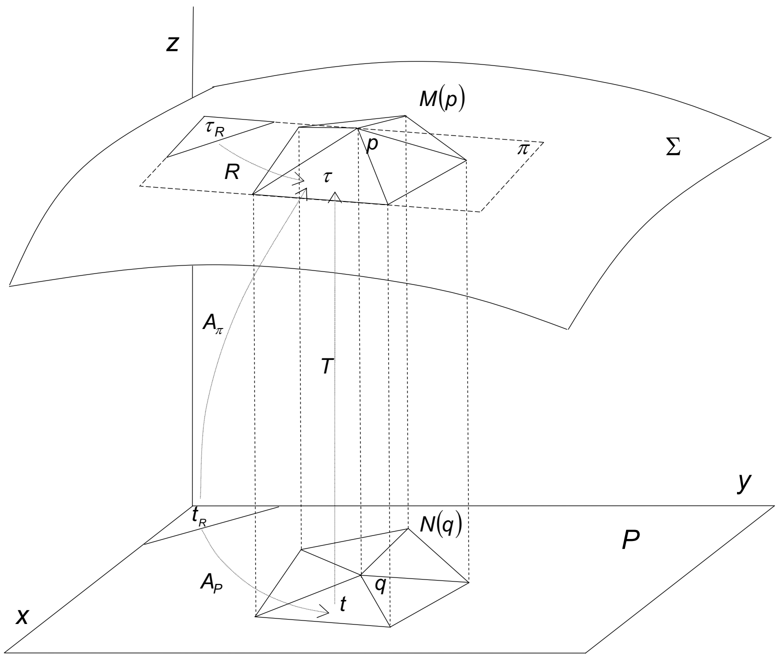


Figura 1. Local surface mesh and its associated parametric mesh.

white) is shown in Fig.2(b). There are not inverted triangles because the nodes are placed in different z-coordinates. Note that, although this new position is optimal in the relative to the shape of the triangles, it is not acceptable for many purposes as, for example, to construct a 3-D mesh from it.

Moreover, the direct optimization of $M(p)$ would require the imposition of the constraint $\xi \in \Sigma$, which would complicates its resolution.

For all these reasons, we will approach the problem in a different way. We will use an approximate version of the similarity transformation given in (12) that avoids these conflicts.

Consider that, for example, we choose $\mathbf{x}_0$ as free node , that is, $\mathbf{x} = \mathbf{x}_0$, then, the free node on the surface is $\xi = (x, y, f(x,y))^T = \xi_0$. Note that $S_E = A_P W_E^{-1}$ depends on $\mathbf{x}$ through A_P and $S_I = A_P W_I^{-1}$ depends on ξ, due to $W_I = A_P R^{-1} W_E$, and R is function of ξ. Thus, we have $S_E(\mathbf{x})$ and $S_I(\xi)$. The approximate problem consists on keeping inalterable the ideal element, W_I, in each step of the optimization process. To do this we fix $W_I(\xi)$ to its initial value, $W_I^0 = W_I(\xi^0)$, where ξ^0 is the initial position of ξ. Thus, $S_I^0(\mathbf{x}) = A_P(\mathbf{x})(W_I^0)^{-1}$ and, the associated similarity transformation of S, yields

$$S^0(\mathbf{x}) = S_E^{-1}(\mathbf{x})\, S_I^0(\mathbf{x})\, S_E(\mathbf{x}) \tag{13}$$

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 $t \subset P$

$$\eta = \frac{|S^0|^2}{2\left(\sigma^0\right)^{\frac{2}{3}}} \tag{14}$$

where $\sigma^0 = \det(S^0)$.

Note that the optimization of the local mesh is a two-dimensional problem without constraints, defined on $N(q)$, and, therefore, it can be solved with a low computational cost. Furthermore, if we write W_I^0 as $A_P^0(R^0)^{-1}W_E$, where $A_P^0 = A_P(\mathbf{x}_0)$ and $R^0 = R(\xi_0)$, it is easy to show that S^0 can be simplified as

$$S^0(\mathbf{x}) = R^0\left(A_P^0\right)^{-1} S_E(\mathbf{x}) \tag{15}$$

In fact, this is the expression used to construct the objective function.

Let now analyze the behavior of the objective function when the free node crosses the boundary of the feasible region. If we write $\alpha_P = \det(A_P)$, $\alpha_P^0 = \det(A_P^0)$, $\rho^0 = \det(R^0)$, $\omega_E = \det(W_E)$ and take into account (15), we can span σ^0 as $\left(\alpha_P^0\rho^0\right)^{-1}\alpha_P\omega_E$. Note that α_P^0, ρ^0 and ω_E are constants, so η has a singularity when $\alpha_P = 0$, that is, when $\mathbf{x}$ is on the boundary of the feasible region. This singularity determines a barrier in the objective function that prevents the optimization algorithm to take the free node outside this region. This barrier does not appear if we use the exact weighted Jacobian matrix S, given in (7), due to $\det(R) = R_{11}R_{22} > 0$.

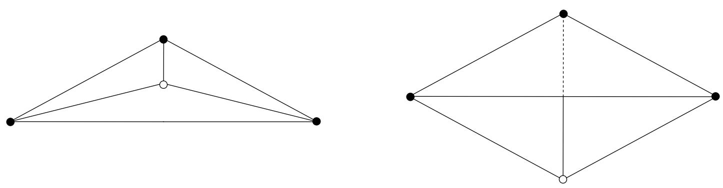
Now, we are going to see how the map $T^0 = T(\xi^{\mathbf{0}})$ transforms the *ideal* triangle on P into an equilateral one on Σ. Thus, consider the function given in (14) and suppose that its minimum value is reached at $\overline{\mathbf{x}}^0$, then, the weighted Jacobian matrix yields, $S^0\left(\overline{\mathbf{x}}^0\right) = R^0\left(A_P^0\right)^{-1} S_E\left(\overline{\mathbf{x}}^0\right) = \mu\Theta$, where $\mu \geq 0$ and $\Theta \in SO(2)$. We deduce that $S_E\left(\overline{\mathbf{x}}^0\right) = A_P\left(\overline{\mathbf{x}}^0\right)W_E^{-1} = \mu A_P^0\left(R^0\right)^{-1}\Theta$ and, then, the optimal value of the Jacobian matrix is $A_P\left(\overline{\mathbf{x}}^0\right) = \mu A_P^0\left(R^0\right)^{-1}\Theta W_E$. Note that $\mu\Theta W_E$ represents a scale change and a rotation of the equilateral triangle and, then, it is also the Jacobian matrix, W_E', of other equilateral triangle. By applying $T^0 = A_\pi^0\left(A_P^0\right)^{-1}$ to $A_P\left(\overline{\mathbf{x}}^0\right)$, where $A_\pi^0 = A_\pi(\xi^{\mathbf{0}})$, results, $T^0 A_P\left(\overline{\mathbf{x}}^0\right) = A_\pi^0\left(R^0\right)^{-1} W_E'$. Now, taken into account (6) and writing $Q^0 = Q(\xi^{\mathbf{0}})$, we obtain

$$T^0 A_P\left(\overline{\mathbf{x}}^0\right) = Q^0 W_E' \tag{16}$$

where Q^0W_E' is the Jacobian matrix of an equilateral triangle $\tau_E' \in M(p)$ given in the canonical basis of $\mathbb{R}^3$.

In short, the triangle $t \subset P$ defined by A_P in the optimal point $\overline{\mathbf{x}} \in P$ is transformed by T^0 into on equilateral triangle belonging to $M(p)$.

Due to T and Q are evaluated in the pre-optimized point, $\xi^{\mathbf{0}}$, the previous property does not mean that $A_P\left(\overline{\mathbf{x}}^0\right)$ is the Jacobian matrix of the "optimal" ideal triangle, $\bar{t}_I$. This triangle must satisfies $T\left(\overline{\xi}^0\right) A_P\left(\overline{\mathbf{x}}^0\right) = Q\left(\overline{\xi}^0\right) W_E'$, but in general $\overline{\xi}^0 \neq \xi^0$. To get a better approximation to $\bar{t}_I$ the point ξ^0 is replaced by $\overline{\xi}^0$ in the factorization (15) and, then, the new objective function is constructed and optimized taking $\mathbf{x}^1 = \overline{\mathbf{x}}^0$ as starting point. This local process is repeated obtaining a sequence $\left\{\mathbf{x}^k\right\}$ of optimal points. Of course, if we have a local mesh instead of a unique triangle, the objective function will be $|K_\eta|_n$. We have experimentally verified in numerous tests that $\{\mathbf{x}^k\}$ converges when the function $f(x, y)$ that defines Σ is continuous.



(a) (b)

Figura 2. Mesh sited on a curved surface (a), and the *not acceptable* optimized mesh (b)

3. EXAMPLES

Two test problems are considered in this section. In the first example we show the behavior of the algorithm in a local mesh formed by six triangles. In the second one we analyze the effects of the smoothing in two meshes, one regular and other refined, constructed on a surface with abrupt gradients.

3.1. Test problem 1

To understand the way in which the optimization algorithm works we choose a simple mesh with six triangles placed on the surface given by the function $f(x, y) = \frac{5}{4}\left(x^2 + (y-1)^2\right)$. The projection of this mesh on the plane $z = 0$ forms another mesh with all the triangles equilateral. The positions of the fixed nodes on $z = 0$ are $\mathbf{x}_1 = (0, -1)^T$, $\mathbf{x}_2 = (\frac{\sqrt{3}}{2}, -\frac{1}{2})^T$, $\mathbf{x}_3 = (\frac{\sqrt{3}}{2}, \frac{1}{2})^T$, $\mathbf{x}_4 = (0, 1)^T$, $\mathbf{x}_5 = (-\frac{\sqrt{3}}{2}, \frac{1}{2})^T$, and $\mathbf{x}_6 = (-\frac{\sqrt{3}}{2}, -\frac{1}{2})^T$, and the initial node position for the free node is $\mathbf{x}_0 = (0, 0)^T$. The frontal view of the initial surface mesh is shown in Fig. 3(a), and its projection on $z = 0$ in 3(c). The corresponding meshes after three steps of the optimization algorithm are shown in 3(b) and 3(d). In these figures it can be seen how the algorithm locates the free node in a position on the plane $z = 0$ that makes the triangles of the surface mesh as equilateral as possible. We have used the exact representation of the surface to calculate the z-coordinate, but similar results are obtained using the approximate surface defined by the linear interpolation of the initial mesh.

In order 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. In Fig. 4 is shown the the relative error, $\log\left(\left|\frac{\overline{K}^{k+1} - \overline{K}^k}{\overline{K}^k}\right|\right)$, in terms of the number of iterations for both cases, where $\overline{K}^k$ be the optimal value of the objective function in the k-th iteration.

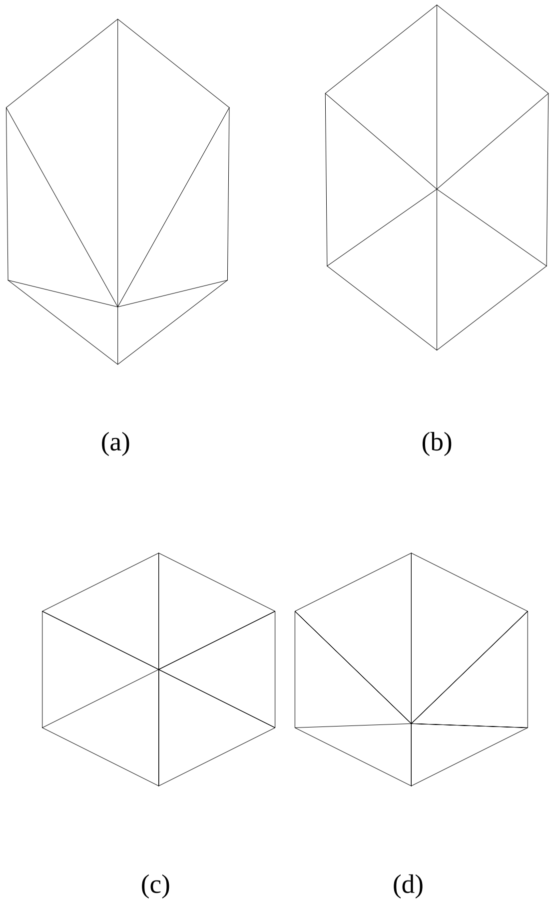


Figura 3. Frontal view of the initial surface mesh (a) and its projection on the plane (c). Frontal view of the optimized surface mesh (b) and its projection (d)

We have observed a similar behavior in all the examples treated until now. Sometimes, the number of iterations required to reach a reasonable relative error ($\simeq 10^{-2}$) is clearly greater ($\simeq 30$) than the needed in this example but, anyway, the algorithm always converges.

3.2. Test problem 2

In this example we consider a more complex surface, defined on the unit square by a function $f(x, y)$, with two maxima, two minima and one saddle point. Two meshes have been constructed on this surface, one regular and the other refined. In this application the same parametric plane P can be used for optimizing all the local meshes.

In Fig. 5(a) and 5(c) are shown the surface mesh and the parametric mesh, respectively. The optimized versions are shown in 5(b) and 5(c). On the other hand, in Fig. 6(a) is shown the surface mesh with refinement in region with hight gradients. Its corresponding parametric mesh

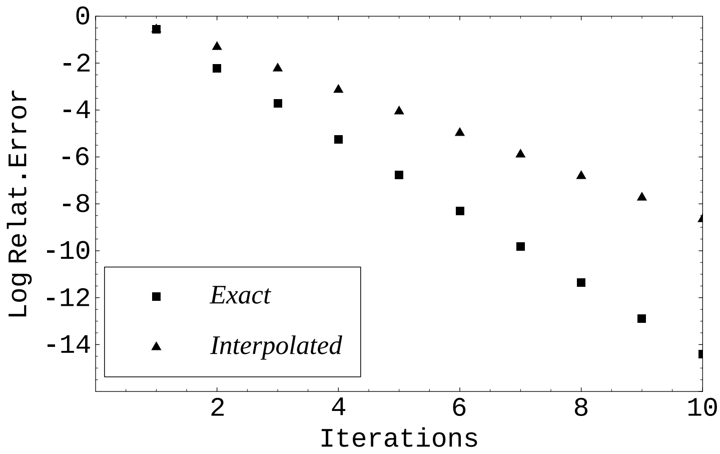


Figura 4. Convergence of the local process. Logarithm of the relative error in terms of the number of iterations.

is shown in Fig. 6(c). The optimized meshes are shown in Fig. 6(b) and (6d).

In order to prevent a loss of the details of the original surface, every time the free node is moved, the optimization algorithm evaluates the distance between the triangles of $M(p)$ and the surface. This distance is given by the difference of heights between the center of the triangle of the present mesh and its corresponding point on the surface. If this distance is greater than certain threshold, the movement of the node is aborted and its previous position is stored. The threshold is established by analyzing the maximum distance between the initial mesh and the surface. An alternative to this control is to use the method of the reference Jacobian developed in [5].

In this example the improvement in the average quality of the mesh is not very significant because the initial mesh is good. The main effect of the optimization is produced on the triangles with worst quality.

4. CONCLUSIONS

We have developed a method to optimize meshes defined on surfaces. Its main characteristic is that the original problem is transformed into a fully two-dimensional one on the parametric space. This allows the optimization algorithm to deal 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 working on the real mesh.

This procedure can be used to optimize the boundary of a 3-D mesh. Note that the node movement on the surface can produce a 3-D tangled mesh and, in this case, we have to use a untangling and smoothing procedure [6].

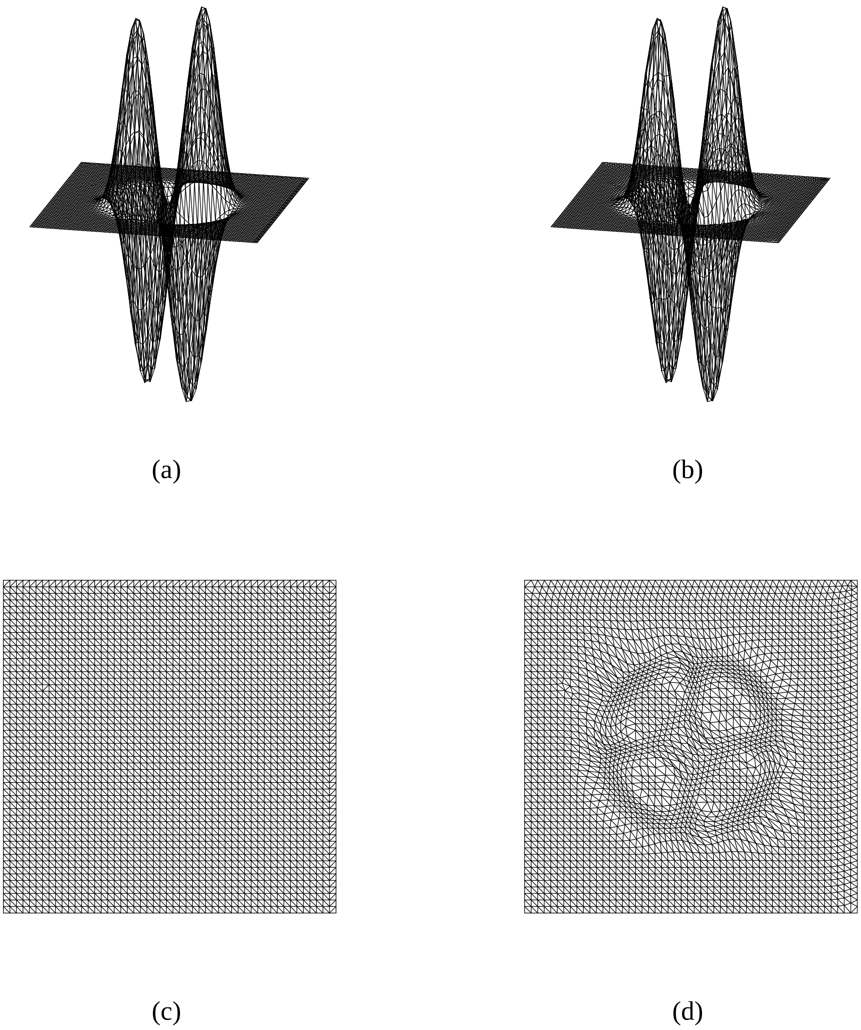
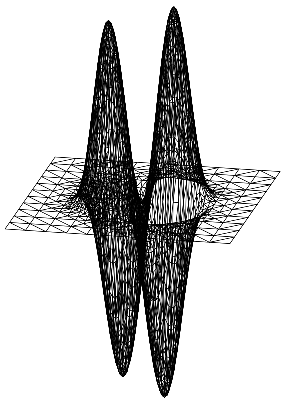
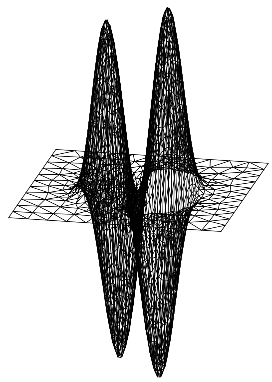


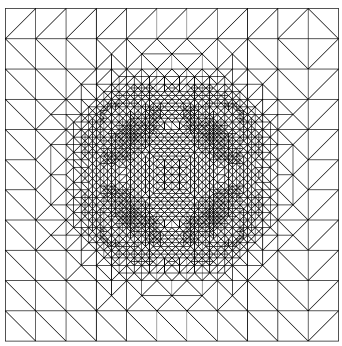
Figura 5. Lateral view of the initial surface mesh (a), optimized surface mesh (b), initial parametric mesh (c) and optimized parametric mesh (d)



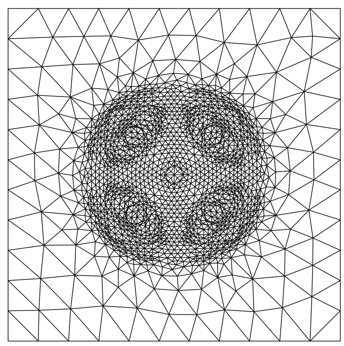
(a)



(b)



(c)



(d)

Figura 6. Initial surface mesh with refinement (a), its optimized surface mesh (b), refined parametric mesh (c) and its associated optimized parametric mesh (d)

REFERENCIAS

[1] L. A. Freitag and P.M. Knupp, Tetrahedral Mesh Improvement via Optimization of the Element Condition Number, *Nonlinear Analysis, Theory, Methods and Applications*, **53**, 1377-1391 (2002).

[2] P.M. Knupp, Algebraic Mesh Quality Metrics, *SIAM J. Sci. Comput.*, **23**, 193-218 (2001).

[3] P.M. Knupp, Achieving Finite Element Mesh Quality via Optimization of the Jacobian Matrix Norm and Associated Quantities. Part I - A Framework for Surface Mesh Optimization, *Int. J. Numer. Meth. Eng.*, **48**, 401-420 (2000).

[4] P.M. Knupp, 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., *Int. J. Numer. Meth. Eng.*, **48**, 1165-1185 (2000).

[5] R.V. Garimella, M.J. Shaskov and P.M. Knupp, Triangular and Quadrilateral Surface Mesh Quality Optimization using Local Parametrization, *Comp. Meth. Appl. Mech. Eng.*, **193**, 913-928 (2004).

[6] J.M. Escobar, E. Rodríguez, R. Montenegro, G. Montero and J.M. González-Yuste, Simultaneous Untangling and Smoothing of Tetrahedral Meshes, *Comp. Meth. Appl. Mech. Eng.*, **192**, 2775-2787 (2003).