Available online at www.sciencedirect.com
SCIENCE DIRECT®
Comput. Methods Appl. Mech. Engrg. 192 (2003) 2775–2787
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

# Simultaneous untangling and smoothing of tetrahedral meshes

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

*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*

Received 5 November 2002; received in revised form 29 January 2003

**Abstract**

The quality improvement in mesh optimisation techniques that preserve its connectivity are obtained by an iterative process in which each node of the mesh is moved to a new position that minimises a certain objective function. The objective function is derived from some quality measure of the *local submesh*, that is, the set of tetrahedra connected to the adjustable or *free node*. Although these objective functions are suitable to improve the quality of a mesh in which there are non-*inverted* elements, they are not when the mesh is tangled. This is due to the fact that usual objective functions are not defined on all $\mathbb{R}^3$ and they present several discontinuities and local minima that prevent the use of conventional optimisation procedures. Otherwise, when the mesh is tangled, there are local submeshes for which the free node is out of the *feasible region*, or this does not exist. In this paper we propose the substitution of objective functions having barriers by modified versions that are defined and regular on all $\mathbb{R}^3$. With these modifications, the optimisation process is also directly applicable to meshes with inverted elements, making a previous untangling procedure unnecessary. This simultaneous procedure allows to reduce the number of iterations for reaching a prescribed quality. To illustrate the effectiveness of our approach, we present several applications where it can be seen that our results clearly improve those obtained by other authors.


*Keywords:* Mesh smoothing; Mesh untangling; Tetrahedral mesh generation; Adaptive meshes; Finite elements

## 1. Introduction

In finite element simulation the mesh quality is a crucial aspect for good numerical behaviour of the method. In a first stage, some automatic 3-D mesh generator constructs meshes with poor quality and in special cases, for example when node movement is required, inverted elements may appear. So, it is necessary to develop a procedure that optimises the pre-existing mesh. This process must be able to smooth and untangle the mesh.

The most usual techniques to improve the quality of a *valid* mesh, that is, one that does not have inverted elements, are based upon local smoothing. In short, these techniques consist of finding the new

* Corresponding author.
*E-mail address:* rafa@dma.ulpgc.es (R. Montenegro).

0045-7825/03/$ - see front matter © 2003 Published by Elsevier Science B.V.
doi:10.1016/S0045-7825(03)00299-8

positions that the mesh nodes must hold, in such a way that they optimise an objective function. Such a function is based on a certain measurement of the quality of the *local submesh*, $N(v)$, formed by the set of tetrahedra connected to the *free node* $v$. As it is a local optimisation process, we can not guarantee that the final mesh is globally optimum. Nevertheless, after repeating this process several times for all the nodes of the current mesh, quite satisfactory results can be achieved. Usually, objective functions are appropriate to improve the quality of a valid mesh, but they do not work properly when there are inverted elements. This is because they present singularities (barriers) when any tetrahedron of $N(v)$ changes the sign of its Jacobian determinant. To avoid this problem we can proceed as Freitag and co-workers in [9,11,12], where an optimisation method consisting of two stages is proposed. In the first one, the possible inverted elements are untangled by an algorithm that maximises their negative Jacobian determinants [11]; in the second, the resulting mesh from the first stage is smoothed using another objective function based on a quality metric of the tetrahedra of $N(v)$ [12]. Two of these objective functions are presented in Section 2. After the untangling procedure, the mesh has a very poor quality because the technique has no motivation to create good-quality elements. Moreover, the quality of the mesh could decrease during the untangling process. This effect may be clearly seen if we apply this process to a valid mesh. As remarked in [9], it is not possible to apply a gradient-based algorithm to optimise the objective function because it is not continuous all over $\mathbb{R}^3$, making it necessary to use other non-standard approaches to overcome the problem. Tinoco et al. have instead proposed in [22,23] a displacement of barriers introducing global parameters, calculated in terms of area, in order to obtain convex structured grids for two-dimensional domains.

In Section 3 we propose an alternative to these procedures, such that the untangling and smoothing are carried out in the same stage. So, we obtain meshes with better quality just after untangling than those algorithms based on maximising the minimum Jacobian determinant and, thus, the number of smoothing iterations for reaching a given quality is smaller. For this purpose, we use a suitable modification of the objective function in such a way that it is regular all over $\mathbb{R}^3$. When a feasible region (subset of $\mathbb{R}^3$ where $v$ could be placed, being $N(v)$ a valid submesh) exists, the minima of the original and modified objective functions are very close and, when this region does not exist, the minimum of the modified objective function is located in such a way that it tends to untangle $N(v)$. The latter occurs, for example, when the fixed boundary of $N(v)$ is tangled. With this approach, we can use any standard and efficient unconstrained optimisation method to find the minimum of the modified objective function, see for example [2]. In Section 4 we analyse the main steps to implement conventional optimisation algorithms.

In this work we have applied the proposed modification to two different objective functions derived from each *algebraic mesh quality metrics* studied in [13], but it would also be possible to apply it to other objective functions which have barriers like those presented in [14]. The results for a test problem and a practical application are shown in Section 5. Finally, conclusions and future research are presented in Section 6.

## 2. Objective functions

Several *tetrahedron shape measures* [5] could be used to construct an objective function. Nevertheless those obtained by algebraic operations are specially indicated for our purpose because they can be computed very efficiently. The above mentioned algebraic mesh quality metrics and the corresponding objective functions are shown in this section.

Let $T$ be a tetrahedral element in the physical space whose vertices are given by $\mathbf{x}_k = (x_k, y_k, z_k)^{\mathrm{T}} \in \mathbb{R}^3$, $k = 0, 1, 2, 3$ and $T_{\mathrm{R}}$ be the reference tetrahedron with vertices $\mathbf{u}_0 = (0, 0, 0)^{\mathrm{T}}$, $\mathbf{u}_1 = (1, 0, 0)^{\mathrm{T}}$, $\mathbf{u}_2 = (0, 1, 0)^{\mathrm{T}}$ and $\mathbf{u}_3 = (0, 0, 1)^{\mathrm{T}}$. If we choose $\mathbf{x}_0$ as the translation vector, the affine map that takes $T_{\mathrm{R}}$ to $T$ is $\mathbf{x} = A\mathbf{u} + \mathbf{x}_0$, where $A$ is the Jacobian matrix of the affine map referenced to node $\mathbf{x}_0$, and expressed as

$$A = \begin{pmatrix} x_1 - x_0 & x_2 - x_0 & x_3 - x_0 \\ y_1 - y_0 & y_2 - y_0 & y_3 - y_0 \\ z_1 - z_0 & z_2 - z_0 & z_3 - z_0 \end{pmatrix}. \tag{1}$$

Let now $T_I$ be an equilateral tetrahedron with all its edges of length one and vertices located at $\mathbf{v}_0 = (0,0,0)^T$, $\mathbf{v}_1 = (1,0,0)^T$, $\mathbf{v}_2 = (1/2, \sqrt{3}/2, 0)^T$, $\mathbf{v}_3 = (1/2, \sqrt{3}/6, \sqrt{2}/\sqrt{3})^T$. Let $\mathbf{v} = W\mathbf{u}$ be the linear map that takes $T_R$ to $T_I$. Its Jacobian matrix is

$$W = \begin{pmatrix} 1 & 1/2 & 1/2 \\ 0 & \sqrt{3}/2 & \sqrt{3}/6 \\ 0 & 0 & \sqrt{2}/\sqrt{3} \end{pmatrix}. \tag{2}$$

The affine map that takes $T_I$ to $T$ is given by $\mathbf{x} = AW^{-1}\mathbf{v} + \mathbf{x}_0$, and its Jacobian matrix is $S = AW^{-1}$. This weighted matrix $S$ is independent of the node chosen as reference; it is said to be *node invariant* [13]. 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 $|S| = \sqrt{\mathrm{tr}(S^T S)}$, is specially indicated because it is easily computable. Thus, it is shown in [13] that both $q_\eta(S) = (3\sigma^{(2/3)})/|S|^2$ and $q_\kappa(S) = 3/(|S||S^{-1}|)$ are algebraic quality measures of $T$, where $\sigma = \det(S)$. The maximum value of these quality measures is the unity and it corresponds to equilateral tetrahedron. On the other hand, any flat tetrahedron has quality measure zero. We can derive optimisation functions from these quality measures. Thus, let $\mathbf{x} = (x, y, z)^T$ be the free node position of $v$, and let $S_m$ be the weighted Jacobian matrix of the $m$th tetrahedron of $N(v)$. We define the objective function of $\mathbf{x}$, associated to an $m$th tetrahedron as

$$\eta_m = \frac{|S_m|^2}{3\sigma_m^{(2/3)}} \tag{3}$$

or

$$\kappa_m = \frac{|S_m||S_m^{-1}|}{3} = \frac{|S_m||\Sigma_m|}{3\sigma_m}, \tag{4}$$

where $\Sigma_m = \sigma_m S_m^{-1}$ is the adjoint matrix of $S_m$. Then, the corresponding objective functions for $N(v)$ can be constructed by using the $p$-norm of $(\eta_1, \eta_2, \ldots, \eta_M)$ or $(\kappa_1, \kappa_2, \ldots, \kappa_M)$ as

$$|K_\eta|_p(\mathbf{x}) = \left[ \sum_{m=1}^{M} \eta_m^p(\mathbf{x}) \right]^{(1/p)} \tag{5}$$

or

$$|K_\kappa|_p(\mathbf{x}) = \left[ \sum_{m=1}^{M} \kappa_m^p(\mathbf{x}) \right]^{(1/p)}, \tag{6}$$

where $M$ is the number of tetrahedra in $N(v)$. The objective function $|K_\eta|_1$ was deduced and used in [1] for smoothing and adapting of 2-D meshes. The same function was introduced in [4], for both 2- and 3-D mesh smoothing, as a result of a force-directed method. Functions $|K_\kappa|_p$ are proposed and widely analysed in [9,12]. Finally, both type of functions, among others, are studied and compared in [14]. Note that the above objective functions can be only used for smoothing valid meshes, that is, $\sigma_m > 0$, $\forall m = 1, \ldots, M$.

Although these optimisation functions are smooth in those points where $N(v)$ is a valid submesh, they became discontinuous when the volume of any tetrahedron of $N(v)$ goes to zero. It is due to the fact that $|K_\eta|_p$ and $|K_\kappa|_p$ approach infinity when $\sigma_m$ tends to zero and its numerators are bounded below. In fact, it is possible to prove that $|S_m|$ and $|\Sigma_m|$ reach their minima, with strictly positive values, when $v$ is placed in the

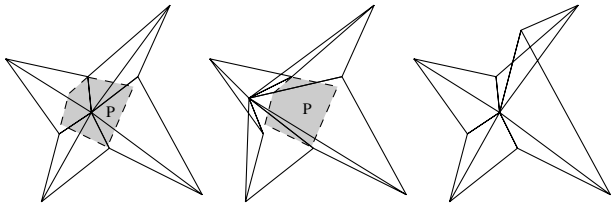



Fig. 1. (a) A valid mesh with the free node inside the feasible region. (b) Non-valid mesh with the free node outside the feasible region. (c) Non-valid mesh with fixed boundary tangled.

geometric centre of the fixed face of the $m$th tetrahedron. The positions where $v$ must be located to get $N(v)$ to be valid, i.e., the feasible region, is the interior of the polyhedral set $P$ defined as

$$P = \bigcap_{m=1}^{M} H_m, \tag{7}$$

where $H_m$ are the half-spaces defined by $\sigma_m(\mathbf{x}) \geqslant 0$ (the shaded region of Fig. 1(a)). This set can occasionally be empty, for example, when the fixed boundary of $N(v)$ is tangled (Fig. 1(c)). In this situation, functions $|K_\eta|_p$ and $|K_\kappa|_p$ stop being useful as optimisation functions. On the other hand, when the feasible region exists, that is $\operatorname{int} P \neq \emptyset$, the objective functions tend to infinity as $v$ approaches the boundary of $P$. Due to these singularities, a barrier is formed which avoids reaching the appropriate minimum by using gradient-based algorithms, when these start from a free node outside the feasible region (see Fig. 1(b)). In other words, with these algorithms we can not optimise a tangled mesh $N(v)$ with the above objective functions, even if the feasible region exists.

## 3. Modified objective functions

We propose a modification in the previous objective functions (5) and (6), so that the barrier associated with their singularities will be eliminated and the new functions will be smooth all over $\mathbb{R}^3$. An essential requirement is that the minima of the original and modified functions are nearly identical when $\operatorname{int} P \neq \emptyset$. Our modification consists of substituting $\sigma$ in (5) or (6) by the positive and increasing function

$$h(\sigma) = \tfrac{1}{2}(\sigma + \sqrt{\sigma^2 + 4\delta^2}) \tag{8}$$

being the parameter $\delta = h(0)$. We represent in Fig. 2 the function $h(\sigma)$.

Thus, the new objective functions here proposed are given by

$$|K_\eta^*|_p(\mathbf{x}) = \left[\sum_{m=1}^{M} (\eta_m^*)^p(\mathbf{x})\right]^{(1/p)} \tag{9}$$

and

$$|K_\kappa^*|_p(\mathbf{x}) = \left[\sum_{m=1}^{M} (\kappa_m^*)^p(\mathbf{x})\right]^{(1/p)}, \tag{10}$$

where

$$\eta_m^* = \frac{|S_m|^2}{3h(\sigma_m)^{(2/3)}} \tag{11}$$

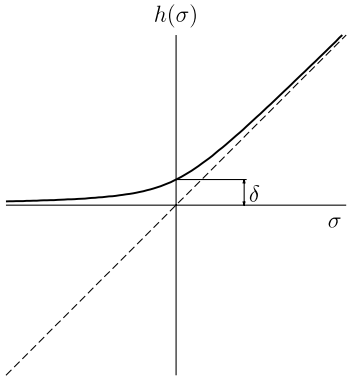



Fig. 2. Representation of function $h(\sigma)$.

and

$$\kappa_m^* = \frac{|S_m||\Sigma_m|}{3h(\sigma_m)} \tag{12}$$

are the modified objective functions for the $m$th tetrahedron.

The behaviour of $h(\sigma)$ in function of $\delta$ parameter is such that, $\lim_{\delta\to 0} h(\sigma) = \sigma$, $\forall\sigma \geqslant 0$ and $\lim_{\delta\to 0} h(\sigma) = 0$, $\forall\sigma \leqslant 0$. Thus, if $\operatorname{int} P \neq \emptyset$, then $\forall \mathbf{x} \in \operatorname{int} P$ we have $\sigma_m(\mathbf{x}) > 0$, for $m = 1, 2, \ldots, M$ and, as smaller values of $\delta$ are chosen, $h(\sigma_m)$ behaves very much as $\sigma_m$, so that, original objective functions and their corresponding modified versions are very close in the feasible region. Particularly, in the feasible region, as $\delta \to 0$, functions $|K_\eta^*|_p$ and $|K_\kappa^*|_p$ converge pointwise to $|K_\eta|_p$ and $|K_\kappa|_p$, respectively. Besides, by considering that $\forall\sigma > 0$, $\lim_{\delta\to 0} h'(\sigma) = 1$ and $\lim_{\delta\to 0} h^{(n)}(\sigma) = 0$, for $n \geqslant 2$, it is easy to prove that the derivatives of these objective functions verify the same property of convergence. As a result of these considerations, it may be concluded that the positions of $v$ that minimise original and modified objective functions are nearly identical when $\delta$ is *small*. Actually, the value of $\delta$ is selected in terms of point $v$ under consideration, making it as small as possible and in such a way that the evaluation of the minimum of modified functions does not present any computational problem.

In particular, let $\gamma$ be the *machine epsilon* ($0 < \gamma \ll 1$), then, to avoid divisions by zero when computing the modified objective functions (11) and (12), we have to impose $h(\sigma) \geqslant \gamma$ for all the tetrahedra of $N(v)$. As $h(\sigma)$ is an increasing function, the worst case corresponds to $\sigma = \sigma_{\min}$, where $\sigma_{\min} = \min_{m=1,\ldots,M}(\sigma_m)$. It can be proved that for the following selection of $\delta$,

$$\delta \geqslant \delta_{\min} = \begin{cases} \sqrt{\gamma(\gamma - \sigma_{\min})} & \text{if } \sigma_{\min} < \gamma \\ 0 & \text{if } \sigma_{\min} \geqslant \gamma \end{cases} \tag{13}$$

the condition $h(\sigma) \geqslant \gamma$ is assured. In practice, we use a modified value of $\gamma$ by introducing a security factor. Note that when the free node is *clearly* inside of a feasible region ($\sigma_{\min} \geqslant \gamma$), our modified objective functions coincide with the original ones.

Suppose that $\operatorname{int} P = \emptyset$, then the original objective functions, $|K_\eta|_p$ and $|K_\kappa|_p$, are not suitable for our purpose because they are not correctly defined. Nevertheless, modified functions are well defined and tend to solve the tangle. We can reason it from a qualitative point of view by considering that the dominant terms in $|K_\eta^*|_p$ or $|K_\kappa^*|_p$ are those associated to the tetrahedra with more negative values of $\sigma$ and, therefore, the minimisation of these terms imply the increase of these values. It must be remarked that $h(\sigma)$ is an increasing function and $|K_\eta^*|_p$ and $|K_\kappa^*|_p$ tend to $\infty$ when the volume of any tetrahedron of $N(v)$ tends to $-\infty$, since $\lim_{\sigma\to-\infty} h(\sigma) = 0$.

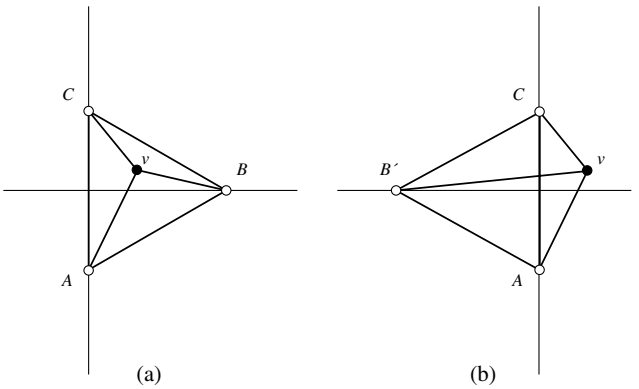



Fig. 3. Test example: (a) valid mesh and (b) tangled mesh.

In conclusion, by using the modified objective functions, we can untangle the mesh and, at the same time, improve its quality.

For a better understanding of the behaviour of the objective functions and their modifications, we propose the following test example. Let us consider a simple 2-D mesh formed by three triangles, $vBC$, $vCA$ and $vAB$, as shown in Fig. 3(a), where we have fixed $A(0,-1)$, $B(\sqrt{3},0)$, $C(0,1)$ and $v(x,y)$ is the free node. In this case, the feasible region is the interior of the equilateral triangle $ABC$. In Fig. 4(a) we show $|K_\eta|_2$ (solid line) and $|K_\eta^*|_2$ for $\delta = 0.1$, 0.2, 0.3 (dashed lines) as a function of $x$ for a fixed value $y = 0$ (the $y$-coordinate of the optimal solution). A similar plot is shown in Fig. 5(a) for $|K_\kappa|_2$ (solid line) and $|K_\kappa^*|_2$ (dashed lines). We can see that original objective functions present several local minima and discontinuities, opposite to the modified ones. Besides, the original functions reach their absolute minima outside the feasible region. Vertical asymptotes in original objective functions correspond to positions of the free node for which $\sigma = 0$ for any tetrahedra of the local mesh. As might be expected, the optimal solution for the two modified function results in $v(\sqrt{3}/3,0)$. The original and modified functions are nearly identical in the proximity of this point, as can be seen in Figs. 4(a) and 5(a). In addition, note that the effect of increasing the value of $\delta$ in the modified functions is only significant far from the optimal location.

Let us now consider the tangled mesh obtained by changing the position of point $B(\sqrt{3},0)$ to $B'(-\sqrt{3},0)$ (see Fig. 3(b)). Here, the mesh is constituted by the triangles $vB'C$, $vCA$ and $vAB'$, where $vB'C$ and $vAB'$ are inverted. The feasible region does not exist in this new situation. The graphics of functions $|K_\eta|_2$ and $|K_\eta^*|_2$ are represented in Fig. 4(b). Similarly, $|K_\kappa|_2$ and $|K_\kappa^*|_2$ are shown in Fig. 5(b). Although the mesh cannot be

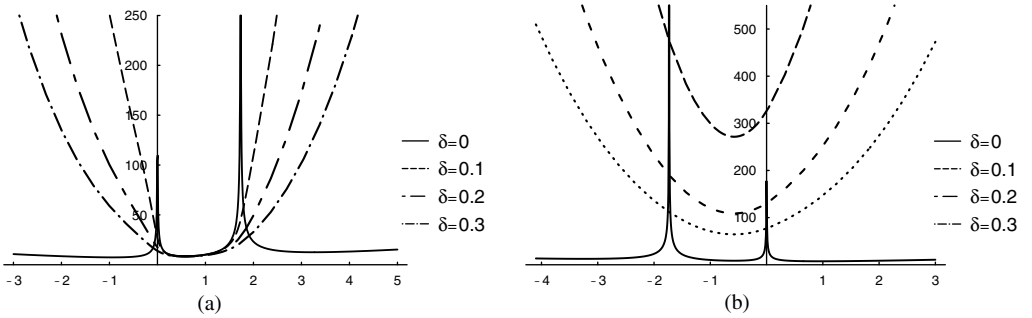



Fig. 4. (a) Transversal cut of $|K_\eta|_2$ (solid line, $\delta = 0$) and $|K_\eta^*|_2$ (dashed lines, $\delta = 0.1$, 0.2, 0.3) for the test example represented in Fig. 3(a); (b) the same objective functions for the tangled mesh of Fig. 3(b).

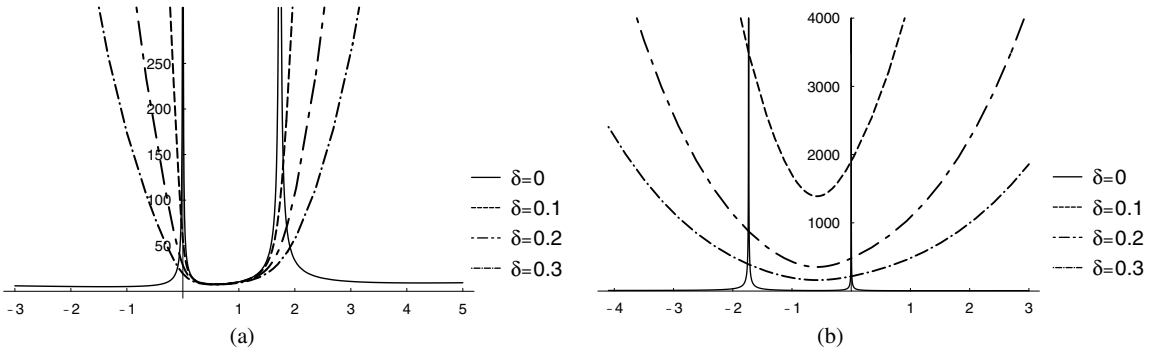



Fig. 5. (a) Transversal cut of $|K_\kappa|_2$ (solid line, $\delta = 0$) and $|K^*_\kappa|_2$ (dashed lines, $\delta = 0.1$, 0.2, 0.3) for the test example represented in Fig. 3(a); (b) the same objective functions for the tangled mesh of Fig. 3(b).

untangled, we get $v(-\sqrt{3}/3, 0)$ as the optimal position of the free node by using our modified objective functions. For this position the three triangles are "equally inverted" (same negative values of $\sigma$). In this particular example the same result could be achieved by maximising the minimum value of $\sigma$ in the mesh, as proposed in [11].

## 4. Optimisation of the modified objective functions

Conventional optimisation algorithms, such as steepest descent or Newton, require the evaluation of the gradient and, in some cases, the Hessian of the objective function. So, in this Section we calculate the first and second derivatives of $\eta^*$ and $\kappa^*$ with respect to arbitrary parameters $\alpha$ and $\beta$ that represent any of the $x$, $y$ and $z$ coordinates of the free node. In order to obtain these derivatives we will consider the inner product of two matrices, $R$ and $S$, as

$$(R, S) = \mathrm{tr}(R^{\mathrm{T}} S), \tag{14}$$

so, the Frobenius norm of $S$ is $|S| = \sqrt{(S, S)}$. If we denote $\partial_\alpha$ as the partial derivative operator respect to $\alpha$, such that $\partial_\alpha R = [\partial_\alpha r_{ij}]$ for an $n \times n$ $R = [r_{ij}]$ matrix, $1 \leqslant i, j \leqslant n$, it is possible to show that the derivative of $\eta^*$ is

$$\partial_\alpha \eta^* = 2\eta^* \left[ \frac{(\partial_\alpha S, S)}{|S|^2} - \frac{\partial_\alpha \sigma}{3\sqrt{\sigma^2 + 4\delta^2}} \right]. \tag{15}$$

For $\kappa^*$ we have

$$\partial_\alpha \kappa^* = \kappa^* \left[ \frac{(\partial_\alpha S, S)}{|S|^2} + \frac{(\partial_\alpha \Sigma, \Sigma)}{|\Sigma|^2} - \frac{\partial_\alpha \sigma}{\sqrt{\sigma^2 + 4\delta^2}} \right]. \tag{16}$$

To obtain the second derivatives, we must take into account that $S$, $\Sigma$ and $\sigma$ are linear functions of $x$, $y$, $z$, and thus $\partial_{\alpha\beta} S$, $\partial_{\alpha\beta} \Sigma$ and $\partial_{\alpha\beta} \sigma$ are zero. With these considerations we have

$$\partial_{\alpha\beta} \eta^* = \frac{\partial_\alpha \eta^* \partial_\beta \eta^*}{\eta^*} + 2\eta^* \left[ \frac{(\partial_\alpha S, \partial_\beta S)}{|S|^2} - \frac{2(\partial_\alpha S, S)(\partial_\beta S, S)}{|S|^4} + \frac{\sigma \partial_\alpha \sigma \partial_\beta \sigma}{3(\sigma^2 + 4\delta^2)^{(3/2)}} \right], \tag{17}$$

and for $\kappa^*$ we have

$$\partial_{\alpha\beta}\kappa^* = \frac{\partial_\alpha\kappa^*\,\partial_\beta\kappa^*}{\kappa^*} + \kappa^*\left[\frac{(\partial_\alpha S,\partial_\beta S)}{|S|^2} + \frac{(\partial_\alpha \Sigma,\partial_\beta \Sigma)}{|\Sigma|^2} - \frac{2(\partial_\alpha S,S)(\partial_\beta S,S)}{|S|^4} - \frac{2(\partial_\alpha \Sigma,\Sigma)(\partial_\beta \Sigma,\Sigma)}{|\Sigma|^4} + \frac{\sigma\,\partial_\alpha\sigma\,\partial_\beta\sigma}{(\sigma^2+4\delta^2)^{(3/2)}}\right]. \tag{18}$$

Eqs. (17) and (18) can be simplified since $(\partial_\alpha S,\partial_\alpha S) = 3/2$, $(\partial_\alpha S,\partial_\beta S) = 0$, if $\alpha \neq \beta$.

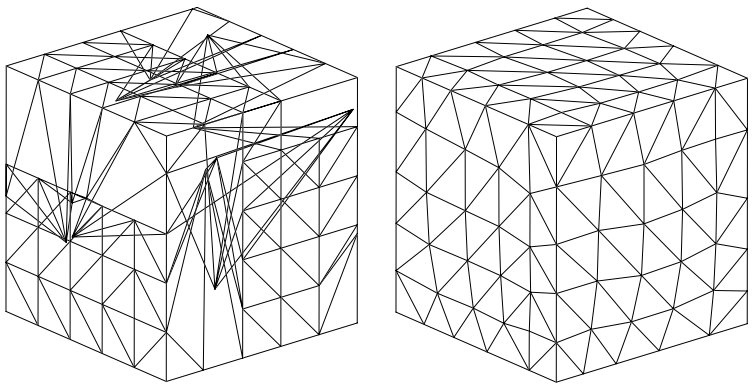
If $\delta$ satisfies Eq. (13), the derivatives of our modified objective functions, $|K^*_\eta|_p$ and $|K^*_\kappa|_p$, can be easily evaluated through the previous expressions without computational problems, since $\gamma \leqslant h(\sigma) \leqslant \sqrt{\sigma^2+4\delta^2}$.

## 5. Applications

To check the effectiveness of the proposed techniques we first consider a regular mesh of a unit cube with 750 tetrahedra and 216 nodes. Initially, the nodes were uniformly distributed, but 22% of them are replaced in random position to get a tangled mesh with valid connectivity. The random redistribution is done in such a way that the nodes originally belonging to an edge or a face of the cube are forced to stay in the same edge or face. In order to do so, some coordinates are fixed and the others take values in the interval [0,1]. The fixed coordinates also remain inalterable in the optimisation process. Thus, for these nodes, the objective function has only one or two variables. The initial tangled mesh, shown in Fig. 6(a), has 146 inverted tetrahedra and an average quality measure of $\overline{q}_\kappa = 0.461$ (the average quality of the regular mesh is 0.749). Here we have chosen as quality measure $q_\kappa = 1/\kappa$ for valid tetrahedra and $q_\kappa = 0$ for inverted ones. The result after eight sweeps of the mesh optimisation process with $|K^*_\eta|_2$ is shown in Fig. 6(b). In this case, the steepest descent algorithm was used for the optimisation of the objective functions.

In Table 1 we present the evolution in the number of inverted tetrahedra, $N_{\text{inv}}$, and the average quality measure, $\overline{q}_\kappa$, of the unit cube mesh in terms of the number of iterations. For this test we have taken four objective functions: $|K^*_\eta|_1$, $|K^*_\eta|_2$, $|K^*_\kappa|_1$ and $|K^*_\kappa|_2$.

In order to compare the behaviour of our SUS algorithm (simultaneous untangling and smoothing) with Opt-MS developed by Freitag (see [8]), we consider the initial unit cube with the same uniform distribution of 216 nodes. This mesh is tangled applying three different random redistributions of its inner nodes, such that we obtain three initial tangled meshes with $N_{\text{inv}} = 34$, 122 and 153 inverted tetrahedra and $\overline{q}_\kappa = 0.653$,



(a) (b)

Fig. 6. (a) The tangled mesh of the unit cube and (b) the resulting mesh after eight steps of the optimisation process.

Table 1
Number of inverted tetrahedra and average quality in terms of the number of iterations of the mesh optimisation process for the cube test (216 nodes and 750 tetrahedra)

| Iteration | Objective function | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | $\lvert K_\eta^* \rvert_1$ | | $\lvert K_\eta^* \rvert_2$ | | $\lvert K_\kappa^* \rvert_1$ | | $\lvert K_\kappa^* \rvert_2$ | |
| | $N_{\text{inv}}$ | $\overline{q}_\kappa$ | $N_{\text{inv}}$ | $\overline{q}_\kappa$ | $N_{\text{inv}}$ | $\overline{q}_\kappa$ | $N_{\text{inv}}$ | $\overline{q}_\kappa$ |
| 0 | 146 | 0.461 | 146 | 0.461 | 146 | 0.461 | 146 | 0.461 |
| 1 | 61 | 0.629 | 11 | 0.704 | 89 | 0.598 | 86 | 0.636 |
| 2 | 2 | 0.731 | 0 | 0.799 | 18 | 0.691 | 0 | 0.753 |
| 3 | 0 | 0.827 | 0 | 0.826 | 1 | 0.813 | 0 | 0.828 |
| 4 | 0 | 0.850 | 0 | 0.836 | 0 | 0.832 | 0 | 0.847 |
| 5 | 0 | 0.857 | 0 | 0.841 | 0 | 0.849 | 0 | 0.855 |
| 6 | 0 | 0.860 | 0 | 0.843 | 0 | 0.856 | 0 | 0.858 |
| 7 | 0 | 0.861 | 0 | 0.845 | 0 | 0.859 | 0 | 0.860 |
| 8 | 0 | 0.862 | 0 | 0.845 | 0 | 0.861 | 0 | 0.860 |

Table 2
Comparative results using Opt-MS and simultaneous untangling and smoothing techniques for the unit cube test, using five smoothing steps after $I_{\text{unt}}$ untangling iterations

| Initial mesh | | | | Opt-MS | | | SUS | | |
|---|---|---|---|---|---|---|---|---|---|
| $N_{\text{nod}}$ | $N_{\text{tet}}$ | $N_{\text{inv}}$ | $\overline{q}_\kappa$ | $I_{\text{unt}}$ | $q_\kappa^{\min}$ | $\overline{q}_\kappa$ | $I_{\text{unt}}$ | $q_\kappa^{\min}$ | $\overline{q}_\kappa$ |
| 216 | 750 | 34 | 0.653 | 3 | 0.224 | 0.779 | 1 | 0.518 | 0.846 |
| | | 122 | 0.467 | 3 | 0.162 | 0.761 | 1 | 0.517 | 0.847 |
| | | 153 | 0.460 | 3 | 0.148 | 0.787 | 1 | 0.516 | 0.847 |
| 10,648 | 55,566 | 1890 | 0.701 | 7 | 0.0035 | 0.691 | 3 | 0.455 | 0.794 |
| | | 7221 | 0.565 | 8 | 0.000025 | 0.572 | 4 | 0.456 | 0.793 |

0.467 and 0.460 average quality measure, respectively. SUS algorithm untangles these meshes in one single iteration, while Opt-MS package does it after three iterations (see Table 2). Besides, using five smoothing steps after these untangling iterations, we get better values of minimum and average quality measure, $q_\kappa^{\min}$ and $\overline{q}_\kappa$, in all cases. Note that SUS algorithm always obtains final meshes with similar values of quality.

To complicate this test problem, we now consider two tangled meshes of the unit cube including 10,648 nodes and 55,566 tetrahedra, with $N_{\text{inv}} = 1890$ and 7221 inverted tetrahedra, respectively. Last two rows of Table 2 contains the results obtained by Opt-MS and SUS algorithms. Note that the differences between both procedures are emphatized in these applications.

We have also used these untangling and optimisation techniques to construct 3-D meshes adapted to complex surfaces as those defined by irregular terrains [15] and [16]. The aim is to create a tetrahedral 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 [6], 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 [7]. 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. In this last stage there can be occasional low quality elements, or even inverted elements, thus making it necessary to apply any untangling and optimisation procedures.

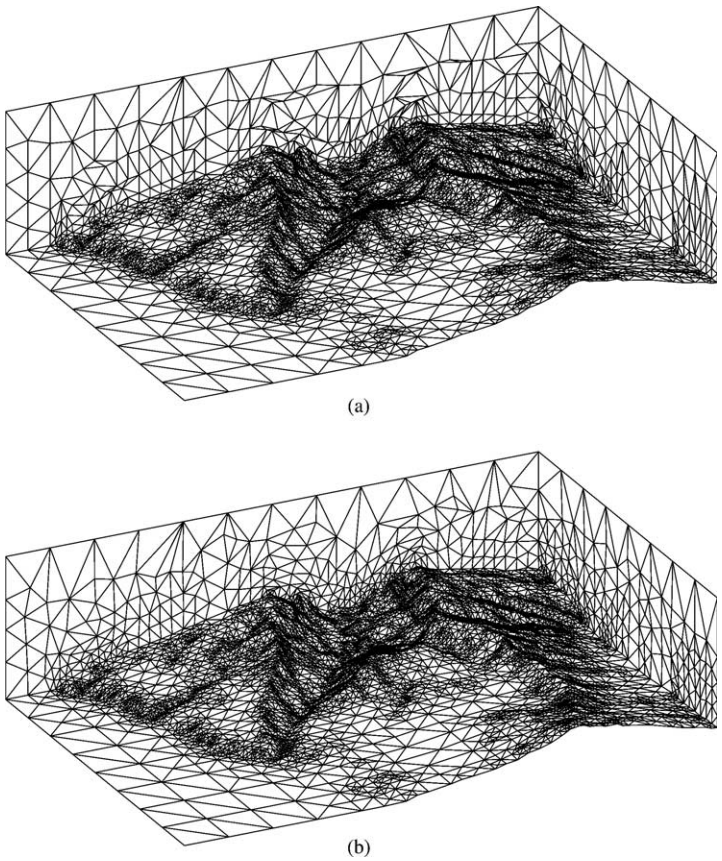



Fig. 7. Rectangular area of Isla de La Palma (Canary Islands): (a) initial mesh with 574 inverted tetrahedra and (b) resulting valid mesh after 10 steps of the optimisation process.

As a practical application of our mesh generator and the optimisation procedure we have taken under consideration a rectangular area in Isla de La Palma (Canary Islands) of $22.8 \times 15.6$ km, where extreme heights vary from 0 to 2279 m. The upper boundary of the domain has been placed at $h = 6$ km. To define the topography we had a digitalisation of the area where heights were defined over a grid with a spacing step of 200 m in directions $x$ and $y$. Starting from a uniform 2-D mesh $\tau_1$ of the rectangular area with a size of elements about $2 \times 2$ km, six global refinements were made using Rivara 4-T algorithm [20]. Once the data were interpolated on this refined mesh, the derefinement algorithm developed in [7] and [19] with a derefinement parameter of $\varepsilon = 25$ m was used. Thus, the adapted mesh nears the terrain surface with an error less than that value. The node distribution of $\tau_1$ is the one considered on the upper boundary of the domain.

The meshes have 81,068 tetrahedra and 16,504 nodes, with a maximum valence of 36 (see Fig. 7). The initial tangled mesh has 574 inverted tetrahedra with an average quality measure $\overline{q}_\kappa = 0.626$ (see Fig. 8). The node distribution is hardly modified after 10 steps of the optimisation process using $|K_\eta^*|_1$. All the inverted elements disappear in the first step of this process and average quality measure increases to $\overline{q}_\kappa = 0.706$. This measure tends to stagnate quickly; in the fifth step we obtain $\overline{q}_\kappa = 0.732$ and in the 10th step $\overline{q}_\kappa = 0.734$. After this optimisation process, the worst quality measure of the optimised mesh tetrahedra is 0.112. We remark that the number of parameters necessary to define the resulting mesh is quite

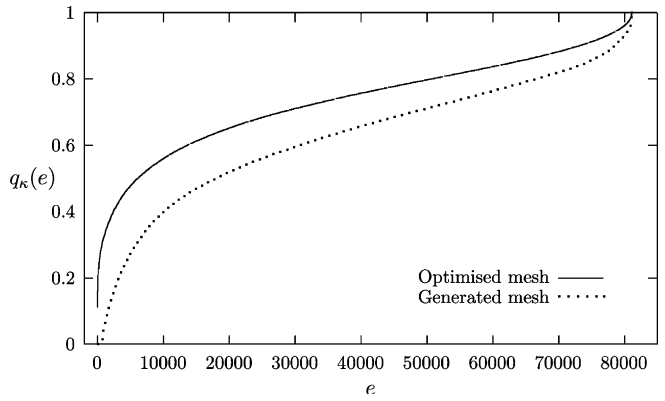



Fig. 8. Quality curves of the generated mesh and the resulting mesh after 10 steps of the optimisation process. Function $q_\kappa(e)$ is a quality measure for tetrahedron $e$.

Table 3
Comparative results using simultaneous untangling and smoothing techniques for La Palma test, using five smoothing steps after $I_{\text{unt}}$ untangling iterations

| Initial mesh | | | | SUS | | |
|---|---|---|---|---|---|---|
| $N_{\text{nod}}$ | $N_{\text{tet}}$ | $N_{\text{inv}}$ | $\overline{q}_\kappa$ | $I_{\text{unt}}$ | $q_\kappa^{\min}$ | $\overline{q}_\kappa$ |
| 16,504 | 81,068 | 431 | 0.723 | 2 | 0.112 | 0.735 |
| | | 3441 | 0.667 | 3 | 0.112 | 0.735 |
| | | 10,051 | 0.547 | 4 | 0.118 | 0.734 |

low, as well as the computational cost. In fact, the complexity of 2-D refinement/derefinement algorithm is linear [19]. Besides, in experimental results we have approximately obtained a linear complexity in function of the number of points for our algorithm of 3-D Delaunay triangulation [6]. In the present application only a few seconds of CPU time on an XEON were necessary to construct the mesh before its optimisation. The complexity of each step of the mesh optimisation process is also linear. In practice, also a few seconds of CPU time were necessary to obtain the optimised mesh applying 10 steps of this latter procedure and using BFGS method [2] to minimise the objective function.

Finally, Table 3 shows the results of SUS algorithm for three new tangled meshes of La Palma problem. Starting from the mesh represented in Fig. 7(b), a percentage of its inner nodes has been randomly moved a bounded distance, $h/3$, around the initial positions, obtaining $N_{\text{inv}} = 431$, 3441 and 10,051 inverted tetrahedra, respectively. We observe that only a few iterations are necessary for untangling. Moreover, the quality of these meshes after five smoothing steps are nearly identical. In general, the final quality of a mesh, after a sufficient number of iterations of the optimisation process, depends more on the mesh topology than on the initial state of tangling. In fact, it always exists a limit of quality defined by the given connectivities of the nodes of the mesh.

## 6. Conclusions and future research

In this paper we present a way to avoid the singularities of common objective functions used to optimise tetrahedral meshes. To do so, we propose a modification of these functions in such a way that it makes them regular all over $\mathbb{R}^3$. Thus, the modified objective functions can be used to smooth and untangle the

mesh simultaneously. With this approach we reduce the number of iterations for reaching a prescribed quality of the mesh, as it can be seen in the applications. We observe that, even, the number of iterations for untangling is less than the one required by the procedure that maximises the minimum Jacobian determinant. Besides, the mesh obtained with our algorithm, just after untangling, has clearly better quality. The regularity shown by the modified objective functions allows the use of standard optimisation algorithms as steepest descent, conjugate gradient, Newton, etc. In principle, a similar modification could be also applicable to other objective functions having the same behaviour as those studied here. These techniques can be implemented in a parallel algorithm, as reported in [10], in order to reduce the computational time of the process.

In our applications we have observed that the behaviour is similar in both types of objective functions, $|K_\eta^*|_p$ and $|K_\kappa^*|_p$, for several values of $p$. Nevertheless, due to its simplicity, $|K_\eta^*|_1$ is more appropriate from a computational point of view.

We have efficiently used these techniques in the generation of 3-D meshes adapted to complex surfaces [15,16], and in other applications [17]. They could also be applied to any other process that implies node movement, such as mesh smoothing for free surface methods and the construction of solution-adapted meshes. The objective functions used in this work are based on quality measures that only take into account geometrical criteria. Nevertheless, we could also use some error indicator which allows the optimisation of the pre-existing mesh attending to the numerical solution of the considered problem.

A promising field of study would combine the 3-D refinement/derefinement of nested meshes with node movement, where the ideas presented here could be introduced. Good recent results have been obtained in [3,18,21] using these techniques, for determining the shape and size of the elements in anisotropic problems.

## Acknowledgements

This work has been supported by the Spanish Government's Science and Technology Department (Ministerio de Ciencia y Tecnología), grant contract: REN2001-0925-C03-02/CLI. The authors are also grateful to Dr. David Shea for his editorial assistance and to the referees for their comments and suggestions that resulted in an improved version of this paper.

## References

[1] R.E. Bank, R.K. Smith, Mesh smoothing using a posteriori error estimates, SIAM J. Numer. Anal. 34 (1997) 979–997.
[2] M.S. Bazaraa, H.D. Sherali, C.M. Shetty, Nonlinear Programing: Theory and Algorithms, John Wiley and Sons, Inc., New York, 1993.
[3] G.C. Buscaglia, E.A. Dari, Anisotropic mesh optimization and its applications in adaptivity, Int. J. Numer. Methods Engrg. 40 (1997) 4119–4136.
[4] H.N. Djidjev, Force-directed methods for smoothing unstructured triangular and tetrahedral meshes, Department of Computer Science, University of Warwick, Coventry, UK, 2000. Available in http://www.andrew.cmu.edu/user/sowen/topics/new.html.
[5] J. Dompierre, P. Labbé, F. Guibault, R. Camarero, Proposal of benchmarks for 3D unstructured tetrahedral mesh optimization, in: Proc. 7th International Meshing Roundtable, Dearborn, MI, 1998, Sandia Report SAND 98-2250, Sandia National Laboratories, Albuquerque, NM, 1998, pp. 459–478.
[6] J.M. Escobar, R. Montenegro, Several aspect of the three-dimensional Delaunay triangulation, Adv. Engrg. Soft. 27 (1/2) (1996) 27–39.
[7] L. Ferragut, R. Montenegro, A. Plaza, Efficient refinement/derefinement algorithm of nested meshes to solve evolution problems, Commun. Numer. Methods Engrg. 10 (1994) 403–412.
[8] L.A. Freitag, Users manual for Opt-MS: local methods for simplicial mesh smoothing and untangling, Technical Report ANL/MCS-TM-239, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 1999.

[9] L.A. Freitag, P.M. Knupp, Tetrahedral element shape optimization via the Jacobian determinant and condition number, in: Proc. 8th International Meshing Roundtable, Lake Tahoe, CA, 1999, Sandia Report SAND 99-2288, Sandia National Laboratories, Albuquerque, NM, 1999, pp. 247–258.
[10] L.A. Freitag, M. Jones, P. Plassmann, A parallel algorithm for mesh smoothing, SIAM J. Sci. Comput. 20 (2000) 2023–2040.
[11] L.A. Freitag, P. Plassmann, Local optimization-based simplicial mesh untangling and improvement, Int. J. Numer. Methods Engrg. 49 (2000) 109–125.
[12] L.A. Freitag, P.M. Knupp, Tetrahedral mesh improvement via optimization of the element condition number, Int. J. Numer. Methods Engrg. 53 (2002) 1377–1391.
[13] P.M. Knupp, Algebraic mesh quality metrics, SIAM J. Sci. Comput. 23 (2001) 193–218.
[14] 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. Methods Engrg. 48 (2000) 1165–1185.
[15] R. Montenegro, G. Montero, J.M. Escobar, E. Rodríguez, J.M. González-Yuste, Tetrahedral mesh generation for environmental problems over complex terrains, in: Proc. ICCS'2002, Lecture Notes in Computer Science, vol. 232, Springer, Berlin, 2002, pp. 335–344.
[16] R. Montenegro, G. Montero, J.M. Escobar, E. Rodríguez, Efficient strategies for adaptive 3-D mesh generation over complex orography, Neural Parallel Scientific Comput. 10 (1) (2002) 57–76.
[17] E. Rodríguez, G. Montero, R. Montenegro, J.M. Escobar, J.M. González-Yuste, Parameter estimation in a three-dimensional wind field model using genetic algorithms, in: Proc. ICCS'2002, Lecture Notes in Computer Science, vol. 2329, Springer, Berlin, 2002, pp. 950–959.
[18] C.C. Pain, A.P. Umpleby, C.R.E. de Oliveira, A.J.H. Goddard, Tetrahedral mesh optimization for steady-state and transient finite element calculations, Comput. Methods Appl. Mech. Engrg. 190 (2001) 3771–3796.
[19] A. Plaza, R. Montenegro, F. Ferragut, An improved derefinement algorithm of nested meshes, Adv. Engrg. Soft. 27 (1/2) (1996) 51–57.
[20] M.C. Rivara, A grid generator based on 4-triangles conforming mesh-refinement algorithms, Int. J. Numer. Methods Engrg. 24 (1987) 1343–1354.
[21] A. Tam, D. Ait-Ali-Yahia, M.P. Robichaud, A. Moore, V. Kozel, W.G. Habashi, Anisotropic mesh adaptation for 3-D flows on structured and unstructured grids, Comput. Methods Appl. Mech. Engrg. 189 (2000) 1205–1230.
[22] J.G. Tinoco-Ruiz, P. Barrera-Sánchez, Area functionals in plane grid generation, in: M. Cross et al. (Eds.), Proc. 6th Conference in Numerical Grid Generation in Computational Field Simulations, International Society of Grid Generation, Greenwich, UK, 1998, pp. 293–302.
[23] J.G. Tinoco-Ruiz, P. Barrera-Sánchez, Area control in generating smooth and convex grids over general plane regions, J. Comput. Appl. Math. 103 (1) (1999) 19–32.