

Surface remeshing by local hermite diffuse interpolation

Alain Rassineux, Pierre Villon, Jean-Michel Savignat, Olivier Stab

► **To cite this version:**

Alain Rassineux, Pierre Villon, Jean-Michel Savignat, Olivier Stab. Surface remeshing by local hermite diffuse interpolation. International Journal for Numerical Methods in Engineering, Wiley, 2000, Unstructured Mesh Generation, 49 (1-2), pp.31-49. <10.1002/1097-0207(20000910/20)49:1/23.0.CO;2-6>. <hal-00655134>

HAL Id: hal-00655134

<https://hal-mines-paristech.archives-ouvertes.fr/hal-00655134>

Submitted on 12 Oct 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Surface remeshing by local hermite diffuse interpolation

A. Rassineux^{1,*,\dagger}, P. Villon¹, J.-M. Savignat² and O. Stab²

¹*Université de Technologie de Compiègne, LG2MS, U.R.A. CNRS 1505 BP529 F-60205
Compiègne Cedex, France*

²*CGES, Ecole Nationale Supérieure des Mines de Paris, 35, rue Saint-Honoré, 77305 Fontainebleau, France*

We propose a method to build a three-dimensional adapted surface mesh with respect to a mesh size map driven by surface curvature. The data needed to optimize the mesh have been reduced to an initial mesh. The building of a local geometrical model but continuous over the whole domain is based on a local Hermite diffuse interpolation calculated from the nodes of the initial mesh and from the normal vectors to the surface. The optimization procedures involve extracting from the surface mesh sets of triangles sharing the same node or the same edge and then remeshing the outer contour to a higher criterion (size or shape). These procedures may be used in order to refine or coarsen the mesh but also in a final step to enhance the shape quality of the elements. Examples demonstrate the ability of the method to create adapted meshes of complex surfaces while meeting high-quality standards and a good respect of the geometrical surface.

KEY WORDS: unstructured mesh generation; mesh adaptation; mesh optimization; moving least squares; diffuse approximation

1. INTRODUCTION

The accuracy of the finite element method is highly dependent on the density of the mesh. In order to control the quality of the analysis, error estimation procedures can be applied to provide information on optimal mesh density [1–5]. The process involved in mesh adaptation techniques is iterative. Once an initial coarse mesh is created, a first solution is obtained, and an error sensitivity analysis can be performed in order to calculate the density of the optimal mesh. When the densities of the optimal mesh have been computed, a new mesh which respects the prescribed density is generated. The aim of such a procedure is to improve the accuracy of the numerical solution while reducing the number of elements. The decision of refining the mesh is based on whether the element error exceeds a threshold specified by the user.

The generation of a three-dimensional surface mesh [6, 7] of an object with an imposed density while meeting ‘acceptable’ shape quality requirements is a difficult and time-consuming task.

*Correspondence to: A. Rassineux, Université de Technologie de Compiègne, LG2MS, U.R.A. CNRS 1505 BP529 F-60205 Compiègne Cedex, France.

^{\dagger} E-mail: alain.rassineux@utc.fr

Without pretending to be exhaustive, we have classified different methods by which a CAD mesh can be adapted. Surface mesh adaptation techniques can be classified into two different categories according to the use or the non-use of the parametric space.

The first group of techniques makes use of the parametric surfaces provided by a modeler. The surface of the object is composed of a set of patches provided by a CAD environment. The most frequently used method to generate a surface mesh consists of using a mapping technique between the two-dimensional parameter space of each patch and another parameter space suitable for meshing. The difficulties of the mapping from parametric to real space have been fully described in a paper of De Cougny and Shephard [8]. The authors propose a method in which the triangulation of the patches is achieved in the real space in order to obtain good mesh size while maintaining surface mesh validity requirements. Solutions to overcome the difficult problems coming from the periodicity or possible degeneracy of parametric space are fully discussed. Once the mesh is obtained, a post-treatment is achieved to enhance the quality of the mesh but also to check its validity through overlapping and intersection tests. The authors show that element size can be controlled in the parametric space provided by the modeler. Since new nodes are calculated on the geometry itself, the geometrical features of the object being obeyed is guaranteed. In a different spirit, Noël *et al.* [9] propose an approach to adapt free form surface mesh in a CAD environment. The surface domain is described by B-spline or Bezier trimmed patches [10]. To overcome the problems occurring with patch-by-patch techniques, nodes can slide on a patch or jump from a patch to another one. The basic idea relies on an analogy between a finite element mesh and the equilibrium position of a network of branches and the position of nodes is controlled through the solution of the equilibrium equations of the bar network. The approach first presented in a two-dimensional context has been extended to free form surfaces.

The same distinction in the use or the non-use of the parametric space can be made when the mesh is created by advancing front technique directly on the surface instead of using a surface mapping method. The underlying parametric representation of the surface patches can be used with a local patch metric derived from the first fundamental form in order to create better quality elements. In this spirit, Tristano *et al.* [11] propose an original automatic mesh generator for 3D parametric surfaces using the advancing front method with a Riemannian surface in order to overcome problems coming from the parametric space. The creation of an element from the data of a segment on the boundary is fully discussed. George and Borouchaki [12] also propose to use a Riemannian metric in a Delaunay context.

Techniques of the second type are based on the polyhedral representation of the object by the mesh itself and therefore do not use the parameter space. These techniques can also be gathered into two groups. Techniques of the first group work on the mesh itself and no secondary geometrical model is built to achieve the adaptation. The basic idea of these techniques is to use a high-density mesh on which simplification tools are performed. However, the mesh must allow to capture the smallest details of the geometry and the sudden changes of curvature in the model. The criterion used to remove a node from the triangulation is often based on the distance between the node and a plane calculated by least-squares fit from the data of the neighbouring nodes. Véron and Léon [13] propose a node selection criterion based on an easy approximation of the Gaussian curvature. Thereafter, nodes are sorted according to their probability to be removed from the initial mesh. Among these techniques, a recent paper of Hattangady [14] presents also a coarsening technique of mesh models for representation of rigid objects in finite element analysis. The basic idea is to reduce significantly the time spent by the solver by representing the mesh model of the rigid objects with as few elements as possible. The procedures developed are

used as a filter before running the analysis. Mesh coarsening is computed in order to eliminate all unnecessary nodes and elements. Graded meshes can be obtained but since no new nodes are added, the initial mesh size must equal the size of the smallest feature to be maintained. The decimation process is based on edge collapsing, edge swapping and planar Laplacian smoothing.

By opposition to an advancing front technique which makes use of the local parametric space, Lau and Lo [15] proposed a technique based on surface normals and tangents evaluation in which the surface is meshed directly by an advancing front technique without regard to the parametric representation of the underlying geometry. However, surface projections must be performed to locate the new nodes on the surface and intersection tests are required to ensure the mesh validity.

Among the techniques which do not use parametric space, the main idea of the second group consists of using the mesh to create a geometrical underlying model. Borouchaki [16] uses a method introduced first by Walton and Meek [17] for constructing a global composite surface from a triangular polyhedral representation of a surface, each triangle representing an individual patch. The iterative algorithm to enhance the mesh is based on edge splitting, edge collapsing, edge swapping and nodal shifting. Such approaches allow not only mesh coarsening but also mesh refinement.

In order to quantify the difference between the geometry and the mesh during the adaptation process, authors have proposed geometrical error estimators. To measure the accuracy at which the mesh can describe what would be a smooth exact geometry of the deformed sheet under the assumption of membrane kinematics, Bonet [5] uses a geometric error estimator. The error is based on an estimated strain tensor between discrete and approximate geometries. The basic idea is that many processes involving deformation of thin sheets and materials like metals or polymers take place under incompressible conditions. The in-plane strain components of the strain tensor would result in a change of the shell thickness. The value of the error is given by the L2 norm of the strain tensor as the integral over the volume of the second invariant of the strain tensor. In order to capture the variations of curvature, De Cougny and Shephard [8] use a simple, purely geometrical criterion. For a given radius of curvature R and a given error ε the maximum allowable arc length is approximated by the chord length $L = 8R\varepsilon/(1 + 4\varepsilon^2)$.

The method we propose in this paper belongs to the second group of techniques since no use is made of the mapping provided by a CAD modeler. The geometrical support is build by a weighted moving least squares [18] approximation method on a local window denoted as diffuse approximation [19, 20]. In our case, interpolating weights have been chosen. The objective is to determine a local surface equation using the nodes of the initial mesh and the normal vectors to the surface calculated from the mesh. The determination of an interpolating local surface equation, however continuous over the domain, enables us to locate nodes on the surface with respect to the curvature during a refinement process. It also allows us to control the coarsening of the mesh. We believe that the continuity, the accuracy and the low computational cost of our geometrical diffuse model are the main advantages of our technique.

The method is also used to compute the principal curvatures on the surface. In order to demonstrate how well the geometry can be represented with our diffuse model and only in this purpose, a geometric error estimator has been used. The results show that the method can be easily coupled with any estimator during a finite element analysis. The respect of the geometry has been successfully validated on usual primitives such as planes, cylinders, cones, spheres and torus. No assumption is made on the surface model. The surface may be closed or not and contain

inner loops. Edges may be multiply connected and non-manifold cases can be handled without restriction.

The approach can be recommended for instance in metal forming techniques involving the deformation of initially flat thin sheets of metal [21] into dies of complex shapes or in the blowing of thin sheet in polymer industry. In large boundary motion problems, the element size along Lagrangian surfaces can induce problems of accuracy and the technique can be used to overcome the basic shortcomings of the mesh distortion and element entanglement encountered in the Lagrangian formulation. We also believe that the local representation of the surface can be used to improve the efficiency and the accuracy of the local search procedures in contact searching and treatment of the contact.

2. BUILDING OF THE HERMITE DIFFUSE MODEL

This chapter introduces some basic notions about metric properties of surfaces [10]. The form of surface suited to the local diffuse interpolation is denoted as Monge patch [22]. A classification of nodes and edges is also presented. The determination of the local surface by Hermite diffuse interpolation is fully discussed.

2.1. Curvature of a Monge patch

A Monge patch is a patch of the form

$$X: U \rightarrow R^3$$

$$X(x, y) = (x, y, z(x, y))$$

where U is an open set in R^2 and $z: U \rightarrow R$ is a differentiable function.

The maximum and minimum of the normal curvature at a given point on a surface are called the principal curvatures. The Gaussian curvature of a regular surface is given by the product of the two principal curvatures.

For a Monge patch, the Gaussian curvature is given by

$$K = \frac{1}{R_1} \frac{1}{R_2} \frac{(\partial^2 z / \partial x^2)(\partial^2 z / \partial y^2) - (\partial^2 z / \partial x \partial y)^2}{(1 + (\partial z / \partial x)^2 + (\partial z / \partial y)^2)^2}$$

and the mean curvature is

$$H = \frac{1}{R_1} \frac{1}{R_2} \frac{(1 + (\partial z / \partial y)^2) \partial^2 z / \partial x^2 - 2(\partial z / \partial x)(\partial z / \partial y) \partial^2 z / \partial x \partial y + (1 + (\partial z / \partial x)^2) \partial^2 z / \partial y^2}{(1 + (\partial z / \partial x)^2 + (\partial z / \partial y)^2)^{3/2}}$$

The principal curvatures can be easily derived from the above relationships.

2.2. Definitions. Classification of nodes and edges

A triangular element of the mesh is denoted as a face. By analogy on superconvergent patch recovery method (SPR) as proposed by Zienkiewicz and Zhu [3, 4], we shall denote as patch the

set of elements whose nodes play a role in the diffuse interpolation scheme. The central face on which we decide to calculate the surface equation is denoted as the reference face.

The neighbourhood of a node can be defined as the set of nodes which share an edge with this central node. This set is constituted by the nodes belonging to one level of element around the node.

A classification of the nodes and the edges is performed. Figure 1 illustrate the different node and edge types. The detection of features is based on an angle criteria. We are aware that the robustness of the approach depends upon the angle threshold and upon the tolerance chosen, especially for complex models. In this paper, we have chosen not to give much emphasis to this problem. However, we are working on a technique to detect intersection lines between geometrical features using the diffuse model. Nevertheless, sharp edges and singular nodes which have not been detected or which must appear on the new mesh can be specified interactively. Furthermore, as far as the mesh is driven by a criterion which measures the accuracy with which the mesh describes the geometry (this point is detailed in Section 3), we experienced that the method proved to produce a reasonable estimate even when a sharp ridge or a singular node was not identified as far as the angle threshold is less than 45° .

Edges are classified into three sets: boundary edges, sharp edges and interior edges. Edges belonging to a single face are boundary edges. As shown in Figure 1(a), if the cosine between normal vectors of two adjacent faces is less than a limit specified by the user, the edge is denoted as a sharp edge. Other edges are interior edges.

Nodes are gathered into 4 types: boundary nodes, nodes belonging to sharp edges, singular nodes and interior nodes. A node is denoted as singular (top of a cone for instance) when its angular defect is greater than a threshold specified by the user. The angular defect at an interior node of the surface mesh is defined as the absolute value of $2\pi - \text{sum of interior angles of faces sharing the node}$ (Figure 1(b)).

A node shared by two sharp edges or by two boundary edges and at which the cosine of the angle between the two oriented edges is less than a threshold given by the user is also considered as a singular node (Figure 1(c)).

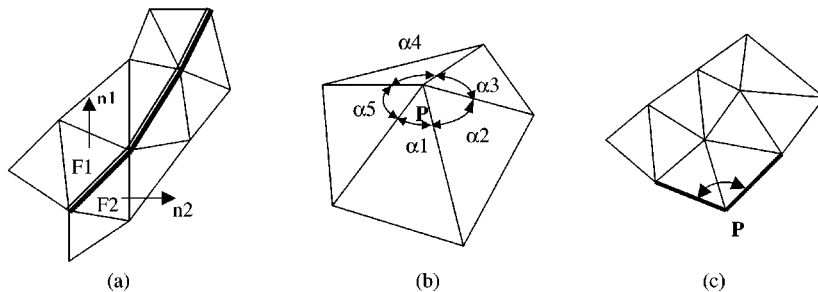


Figure 1. Node and edge classification: (a) detection of sharp edges. The cosine between normal vectors $n1, n2$ of adjacent faces $F1$ and $F2$ is less than a limit specified by the user; (b) detection of a singular node. The angular defect at P is $|2\pi - (\alpha1 + \alpha2 + \alpha3 + \alpha4 + \alpha5)|$; (c) detection of a singular node on a boarder edge. Node P is shared by two boundary edges at which the cosine between the two orientated edges is less than a threshold given by the user.

2.3. Choice of the surface parameters

We want to determine a local surface equation from the data of a set of nodes and from the average normal vectors evaluated at each node. In order to compute a local equation, we suppose that at each node, we can find a neighbourhood on which the surface can be interpolated by a Monge patch of equation $z = f(x, y)$ as shown in Figure 2.

An important issue regarding implementation of the method is the local co-ordinate system with which the surface equation is expressed. We noticed that the choice of the projection plane (plane of the reference face or least-squares fit plane from the nodal set) has no noticeable influence on the result provided that normal vectors of elements belonging to the same patch are close enough. A least-squares fit method has been chosen.

In order to construct a surface equation on which we plan to evaluate curvatures, we have decided to evaluate the surface equation through a second-order equation. The polynomial is defined over each patch of element attached to a reference face. The surface equation can be expressed as

$$z = f(x, y) = \langle 1, x, y, x^2, xy, y^2 \rangle \alpha = p^T \alpha$$

where α is a 6 coefficient vector.

In order to recover a bilinear surface equation over the patch, there must be at least 6 nodes in the patch. During the adaptation process, nodes are likely to be created, slid or even removed. As a consequence, whenever a node must be projected on the diffuse model, a patch (or a reference face) must be determined. For this reason, we decided to choose a diffuse interpolation neighbourhood based on a face model. However, an approach based on nodal patch evaluation can also be carried out.

The selection of the nodal interpolation set and the computation of the normal vector to the surface are now discussed.

2.4. Neighbourhood determination

As shown in Figure 3(a), nodes selected for the diffuse interpolation model must belong to the ring of elements sharing at least one node with the reference face. This set represented in

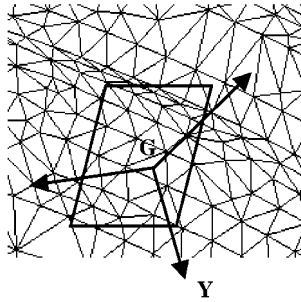


Figure 2. Determination of a local Monge patch $z = f(x, y)$. A local set of axis (G, X, Y, Z) is calculated at each face. G is the center of gravity of the face. Plane (X, Y) can be determined by least-squares fit from the nodal set of neighbouring nodes.

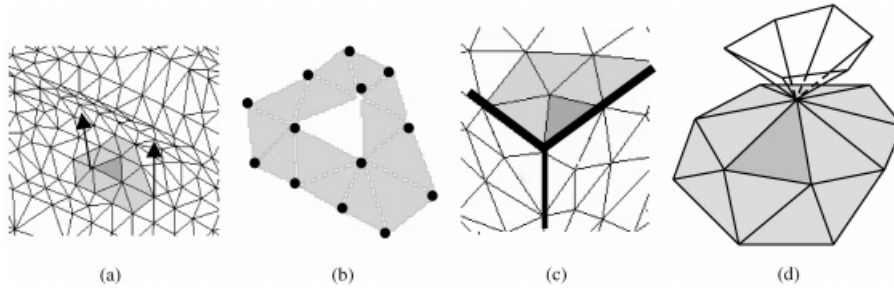


Figure 3. Determination of the interpolation set: (a) ring of elements sharing at least one node with the reference face; (b) set of neighboring nodes; (c) the reference face is on the top of a cube. Lateral faces are rejected from the patch; (d) non-manifold topology. Elements belonging to the upper surface are rejected from the patch associated to reference face F .

Figure 3(b) is composed of the neighbourhood of the nodes which constitute the reference face. Elements containing a sharp edge are rejected from the patch except if the edge belongs to the boundary of the patch. On our example represented in Figure 3(c), lateral faces are rejected from the patch.

If an upper and lower surface share a common internal vertex (Figure 3(d)), the curvature at the shared vertex cannot be uniquely represented. In order to overcome the problem of non-manifold topology, the ring of elements selected for the diffuse interpolation patch are determined by adjacent progression around the reference face. As shown in our example (Figure 3(d)), this element selection procedure guarantees that upper elements are rejected from the set and such non-manifold topologies can be handled.

The computation of normal vector to the surface at singular node is detailed thereafter.

2.5. Computation of normal vectors

The normal vector to a surface node is calculated as the average of all normal vectors of faces shared by the node weighted by the area of each face. However, the computation of the normal vector at sharp edge nodes or at singular nodes requires a special treatment. We recall that normal vector computation is used to determine a local Monge patch equation associated to a reference face. Therefore, the set of elements used to calculate the normal vector to a node is restricted to the triangles attached to the patch at which the diffuse interpolation is computed.

On Figure 4(a), we want to determine the Monge patch of reference triangle $F1$ located on the upper face of a cube and the Monge patch of face $F2$ located on a lateral side of the cube. Sharp edges delimit 3 different areas, the upper face and two lateral faces. The set of interpolation elements attached to $F1$ (respectively $F2$) is restricted to elements belonging to the upper face of the cube (respectively lateral face). Figure 4(b) illustrates that the computation of the normal vector at a singular node or at a node belonging to a sharp edge may change according to the reference face at which the interpolation is calculated whenever areas of different normal vector meet.

We note that a node or an element can belong to different patches. Also, whenever a node is projected on the diffuse model, different patches can be used as far as the validity

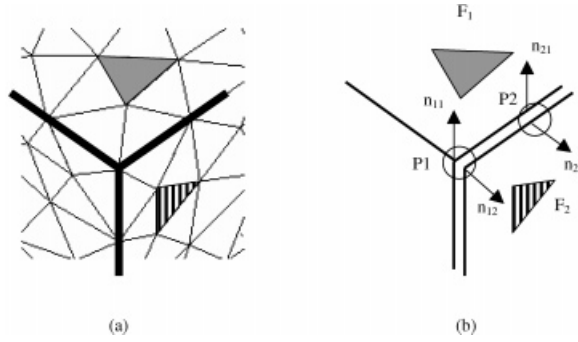


Figure 4. Computation of normal vectors to the surface mesh: (a) the mesh represents the vertices of a cube. Monge patches associated with faces F_1 and F_2 must be determined; (b) determination of normal vectors at P_1 and P_2 . n_{11} and n_{21} (respectively n_{12} and n_{22}) are associated with the diffuse interpolation at face F_1 (respectively F_2).

of the interpolation area is not only restricted to the boundary of the triangular element. This is made possible only because the diffuse interpolation is continuous from one element to a neighbouring element (provided that the adjacent elements are not separated by a sharp edge).

2.6. Hermite diffuse interpolation

The diffuse approximation method has been introduced by Nayroles *et al.* [19] and recently discussed by Breitkopf *et al.* [20]. The diffuse approximation is equivalent to the standard moving least squares (MLS) introduced by Lancaster [18] based on polynomial approximation but is more general in the sense that additional constraints may be easily introduced. Several approaches are based on the MLS method providing smooth approximation of data across irregular patterns of nodes. Meshless methods have been fully presented in an exhaustive survey provided by Belytschko *et al.* [23].

The MLS method is local: at any arbitrary evaluation point x , only the closest nodes x_i are taken into account. The influence of a node x_i is governed by decreasing weight functions $w_i = w(\|x_i - x\|)$ which vanish outside the domain of influence of the node.

In a general case, the MLS approximation does not interpolate data. This property is achieved by introducing singular weight functions w which take an infinite value at the node. The new weights may be obtained by scaling the original weight functions in a way to give a unit value at a node $w_i(x_i) = 1$ and then by applying the following substitution:

$$[w(x_i, x)] \rightarrow \left[\frac{w(x_i, x)}{1 - w(x_i, x)} \right]$$

The diffuse interpolation method only differs from the diffuse approximation in the choice of the weighting functions. In the case of the diffuse interpolation, interpolating weights are chosen. The

diffuse approximation in its standard form has been transformed in order to take into account derivative entities provided from the surface equation.

We experienced that the criterion based on normal colinearity has a major influence on the resolution of the diffuse method and therefore enables a more accurate computation of the curvature than a fitting method based on a single set of node (as diffuse interpolation does if normal vectors are not taken into account).

Assuming that two adjacent patches from the geometry model are connected with respect to tangency rules, our technique guarantees that the interpolation is continuous over the whole surface. Patches are not only calculated for strictly interior faces but also for elements which have a boundary edge or a sharp edge. Patches based on such faces are determined with a fewer number of nodes and the curvature computation may fail when the number of nodes is too low. In such case, the order of the surface equation is decreased and a standard diffuse interpolation can be applied.

The diffuse approximation leads to the minimization of a criterion $J_x(\alpha)$ composed of two terms $J_{1x}(\alpha)$ and $J_{2x}(\alpha)$.

All entities described below are expressed in the local co-ordinate system associated with each patch. We consider a set of n interpolation nodes.

Let $M_i(x_i, y_i, z_i)$ be a node belonging to the set.

The criterion suggesting the interpolating condition can be written as

$$J_{1x}(\alpha) = \sum_{i=1}^{i=n} w(x_i, x) \times (p^T(x_i - x)\alpha - z_i)^2$$

where w_i are interpolating functions associated to each node M_i of the set.

The computation of the weights is obtained as follows:

The diffuse interpolation is calculated at a point x located on the reference face as shown in Figure 5(a). For each node x_i belonging to the patch, its influence on the evaluation at node x is determined by setting to 1 the value affected to x_i . The value of nodes connected to x_i by one edge is set at $1/2$ (first level of nodes). The value at other nodes is set at 0. Finally, the contribution

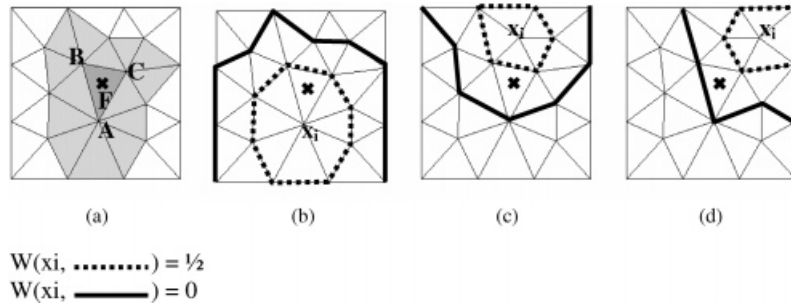


Figure 5. Interpolating weight functions $w(x_i, x)$: (a) the diffuse interpolation is calculated at a point x ; (b) x_i belongs to reference face F , $(w_a, w_b, w_c) = (1, \frac{1}{2}, \frac{1}{2})$; (c) Node x_i is adjacent to F , $(w_a, w_b, w_c) = (0, \frac{1}{2}, \frac{1}{2})$; (d) Node x_i is linked by one edge to F , $(w_a, w_b, w_c) = (0, 0, \frac{1}{2})$.

inside the element is given by the shape functions of the triangle. Weights are normalized in a final step. $w(w_i, x)$ functions have been represented in Figures 5(b)–5(d).

Our computation of the weights is based on the topology of the mesh only and the best solution would be to include the edge lengths according to the geodesical distance in the computation process. A solution could consist of computing a first diffuse model based on our simplified weight computation. Then, the edge lengths could be calculated with the help of the metric provided by the diffuse model. We first used a procedure in which edge lengths were calculated as euclidian distances on the initial mesh. The results show that no important changes can be observed as far as the initial mesh provides a reasonable description of the geometry. Moreover, we noticed that the interpolation condition reduces the influence of the distance between the evaluation point x and the contributing nodes x_i . The main role of the weights is to guarantee the continuity of the model over the whole domain. Furthermore, we experienced that when edge lengths were taken into account, the cost of the evaluation of the diffuse interpolation was higher than the mesh adaptation process.

In order to obtain a low-cost evaluation of the diffuse interpolation, we implemented specific data structures. We experienced that the cost of the diffuse interpolation was less than 10 per cent of the mesh adaptation process even for larger models.

The normal vector to the surface mesh is defined as $\mathbf{n} = n_i \mathbf{x} + n_j \mathbf{y} + \mathbf{z}$.

The normal vector to the Diffuse Monge patch is given by

$$\mathbf{n}_{\text{diff}} = \frac{\partial M}{\partial x} \times \frac{\partial M}{\partial y} = -\frac{\partial P}{\partial x} \mathbf{x} - \frac{\partial P}{\partial y} \mathbf{y} + \mathbf{z}$$

It can be concluded that the criterion based on the colinearity of normal vectors is

$$J_{2x}(\alpha) = \sum_{i=1}^{i=n} w(x_i, x) \times \left\{ \left(\frac{\partial p^T(x_i - x)}{\partial x} \alpha - n_{ix} \right)^2 + \left(\frac{\partial p^T(x_i - x)}{\partial y} \alpha - n_{iy} \right)^2 \right\}$$

Finally, the final criterion which take into account both the condition can be expressed as follows:

$$J_x(\alpha) = (1 - t) \times J_{1x}(\alpha) + t \times J_{2x}(\alpha) \quad \text{where } t \in [0,1]$$

The above criterion can be rewritten as $J_x(\alpha) = (P^T \alpha - r)^T W (P^T \alpha - r)$ where $P^T = [P(x_1 - x), P(x_2 - x), \dots, P(x_n - x)]$ and $r(r_1, r_2, \dots, r_n)$ with

$$r_i = (z_i, n_{ix}, n_{iy})^T \quad \text{and} \quad P(x_i - x) = \begin{bmatrix} 1 & x_i - x & y_i - y & (x_i - x)^2 & (x_i - x)(y_i - y) & (y_i - y)^2 \\ 0 & 1 & 0 & 2(x_i - x) & y_i - y & 0 \\ 0 & 0 & 1 & 0 & x_i - x & 2(y_i - y) \end{bmatrix}$$

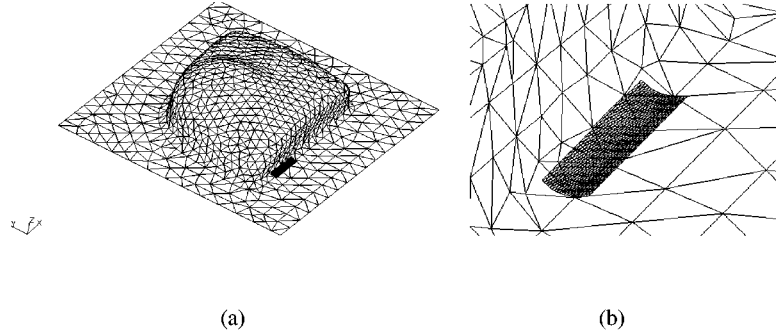


Figure 6. Representation of connecting fillets by hermite diffuse interpolation: (a) mesh of a Twingo cup; (b) representation of several adjacent Monge patches along a connecting fillet area.

and W denotes the diagonal by blocks matrix of the weights given by

$$W = \begin{bmatrix} w(x_1, x) \begin{bmatrix} 1-t & 0 & 0 \\ 0 & t & 0 \\ 0 & 0 & t \end{bmatrix} & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & w(x_n, x) \begin{bmatrix} 1-t & 0 & 0 \\ 0 & t & 0 \\ 0 & 0 & t \end{bmatrix} \end{bmatrix}$$

The minimization of the quadratic error criterion $J_x(\alpha)$ leads to the 6×6 system

$$(PWP^T)\alpha = PWr$$

The system is solved for each patch at the centre of gravity of the reference face.

In Figures 6, points have been created from different patches on a fillet area.

3. CONTROL OF THE MESH SIZE

In order to measure the accuracy with which the mesh describes the geometry, an error estimator inspired from Bonet's work [5] and proposed by Rodriguez-Villa [24] has been used. The estimator provides a relationship between the length L of an edge of the mesh and the smallest principal curvatures radius of model R . The error ε is estimated with the following formula:

$$\varepsilon = \xi^2 \sqrt{1 + \frac{1}{(1 - \xi^2/4)^2}} \quad \text{with } \xi = \frac{L}{R}$$

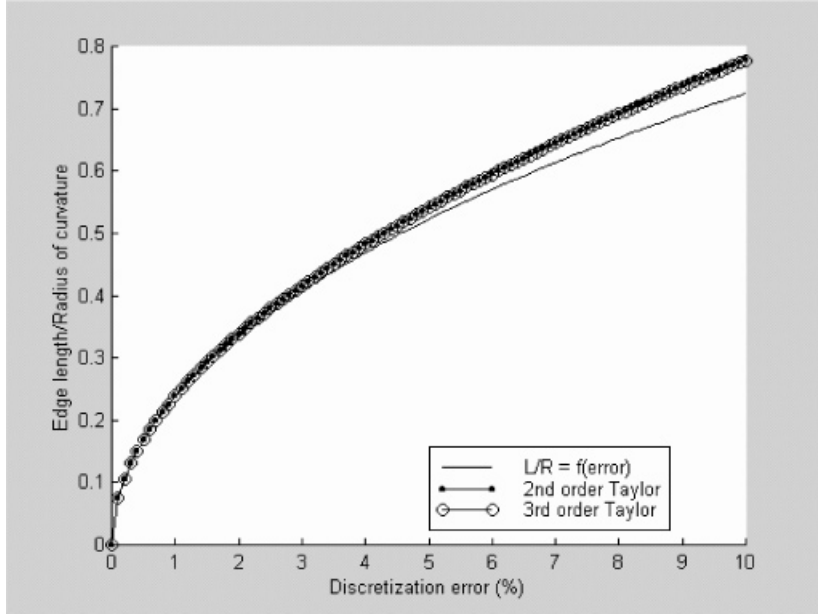


Figure 7. Diagram ε (error) = f (Length/Radius of curvature).

Once the error is chosen, the maximum allowable length must be determined. The previous equation in which L is unknown has been solved. In order to guarantee a fast computation of the error, we estimated the maximum allowable length by a Taylor expansion. A study (Figure 7) shows that if the error value does not exceed 10 per cent our estimation remains valid and that the approximation can be limited to the second order. The relationship between the error ε and the length L is given by

$$L = \sqrt{2 \times \varepsilon} \times (1.68179 - 0.594604 \times \varepsilon)R$$

4. ADAPTATION PROCESS

The different steps of the adaptation process are described. The adaptation technique involves local modification of the mesh in order to satisfy an *a priori* given error [1–5]. These techniques are based on optimum mesh size evaluated from an error estimator. Interior edges can be connected to any number of faces. No special restriction is made concerning the number of faces connected to an interior edge. In order to clarify our explanations, we shall consider that interior edges are connected to two faces exactly.

4.1. Classification of vertices and edges

Mesh vertices and edges are classified. Boundary edges are identified. Sharp edges, singular interior and boundary nodes are determined with respect to angles between neighbouring edges or elements. Values of these parameters are set *a priori* by the user.

4.2. *Creation of contour lines and inner curves*

In order to allow node evaluation on contour lines, boundary and sharp edges are chained to create a network of boundary and inner curves (namely cubic splines). Singular nodes are located at the extremities of the curves. When a closed loop is detected, two curves of equal length are created to overcome the problems encountered with one-to-one parameter space.

4.3. *Creation of a secondary diffuse interpolation model*

The mesh is modified since the process is iterative. The original mesh, denoted as reference mesh, is used as background mesh in order to locate new nodes on the original surface. The reference mesh is also used to determine the value of the criterion (in our case, the curvature). In order to optimize the projection of the nodes and the query on the reference mesh for criterion evaluation, a diffuse interpolation model associated to the modified mesh is updated at each step of the process. The evaluation of nodes during mesh enrichment or mesh coarsening are first made on this model and projected thereafter. Intensive use is made of the data structures developed for 3D mesh generation efficiency [25, 26]. In particular, entities belonging to the reference mesh are stored in an octree [25].

4.4. *Node and criterion evaluation*

The surface mesh is orientated and the Hermite diffuse interpolation is built from the initial triangulation. A list of edges is constituted. Every edge from the list is examined to determine whether the edge must be kept, split or removed from the triangulation in order to satisfy the criterion. The decision is made by evaluating the error criterion at the middle of the edge.

The computation of the error criterion depends on the edge type: boundary edges, sharp edges, or interior edges.

4.4.1. Evaluation on boundary and sharp edges. If the edge belongs to the boundary of the surface or to an inner curve (sharp edges), the middle of the edge is evaluated using the parameter curve of both nodes of the edge. Then a query to the reference mesh is made in order to determine the closest face to the node. In the case of a boundary edge, only one reference face can be found. Otherwise, the sharp edge is shared by two faces. The evaluation face is the one which has the smallest radius of curvature.

4.4.2. Evaluation on interior edges. The last case occurs when the edge belongs to the interior of the surface and is not a sharp edge. The edge is shared by two faces. The middle node is evaluated using the updated diffuse interpolation model. The node is thereafter projected on the reference surface. In that case, the two radii of curvature on the surface are similar, and any face can be chosen for evaluation. However, the decision is made to the patch which has been determined with the greatest number of nodes.

4.5. *Mesh optimization procedures*

We decided to adapt the mesh by using a local remeshing strategy. Similar procedures have been described by many authors [13, 14, 24, 25] and only a brief description of the technique is given. We experienced that the local transformation approach is numerically robust.

Mesh optimization procedures based on optimum mesh size and shape are carried out in an iterative process. The method involves extracting sub-meshes from the surface mesh. Sub-meshes are constituted of triangles sharing the same node or the same edge. The contour of the set of triangles is then remeshed to a higher criterion size or shape. The process is repeated as long as the mesh can be enhanced. We observe that our heuristic approach leads to a monotone convergence. However, no theoretical termination proof can be given.

The procedure are known as edge swapping, vertex removing, edge removing, edge splitting and edge collapsing. In the case of edge collapsing, elements sharing at least one node of the edge are selected.

The method consists of finding all triangulations of a convex polygon composed of a given number of nodes. The number of nodes has been limited to 8 for efficiency requirements. The criteria of all different triangles which can be created from all triangulations are calculated. The different mesh combinations do not depend on the position of the nodes, and are therefore pre-programmed. In a context of tetrahedron mesh optimization, Rassineux [25] uses a local advancing front technique when the shell is not convex. A similar approach has been used. The contour is triangulated in the plane associated to the local Monge patch. In that case, there is no limitation of the number of faces constituting the contour.

A local transformation only changes a set of triangles in a fixed area and since the transformation is performed if the sub-shell is valid and can be meshed to a higher criterion, no invalid triangulation can occur. Furthermore, different criteria can be used for optimization of the mesh.

If an edge is at least twice as long as the value determined by the criterion, the edge is split considering the point in the middle of the edge in the parameter space.

If an edge is at least twice as short as the value determined by the criterion, edge collapsing or node removing procedures can be applied.

Our edge swapping strategy consists of improving the quality of a pair of triangles with respect to a 3D criterion which combines both a shape quality criterion and the geometrical error estimator ε defined in Section 3.

The shape quality criterion for a triangle is defined as follows:

$$Qe(T) = \frac{\alpha \times \rho}{L}$$

where L is the longest edge length of the element and ρ , the ratio of the radius of the circle inscribed in the element. A coefficient α is applied so that the higher criterion (equilateral element) is set at 1.

Edge swapping is performed when both shape quality and geometrical error criteria can be improved. However, no sharp edge can be removed from the triangulation.

In a final step, a simple nodal repositioning technique is achieved. For each node, the polygon constituted by the edges which surround the node is determined. The method consists of creating a set of nodes in the vicinity of the node (with respect to the local Monge patch of the surface). Then, the node which provides the best quality is chosen. The cost of the method is higher than a Laplacian smoothing technique on the Monge patch but provides better results on curved patches.

4.6. Mesh validity

Singular nodes are not removed. Edges on contour lines and inner curves must be preserved.

4.7. Mesh size smoothing procedure

A mesh is obtained after the adaptation process. Even if this mesh respects a size map provided by the criterion, it has to be enhanced in order to guarantee a smooth gradation of the mesh over the whole surface. We propose a simple method to remedy the mesh distortions by smoothing the size map of the mesh.

Once all curvatures have been determined on each face, the corresponding allowable edge length is computed. Our smoothing strategy consists of comparing the size assigned to a face with its neighbouring values. The maximum gradation ratio between two adjacent faces is set at a value between 1 and 2. This also allows us to control the gradation of the mesh. We compared our strategy with an approach in which the mesh was enhanced in a final step. We experienced that better results were obtained if the size map was smoothed *a priori*.

5. EXAMPLES

The shape quality criterion of a triangle has been defined in Section 4.5. In order to validate our method, we have decided in a first step to adapt meshes used for numerical sheet forming simulation. These examples provide a significant overview of the difficulties encountered with sudden changes of curvature (especially when connecting fillets are represented) and demonstrate the ability of the hermite diffuse approximation technique to respect the geometry and to capture the variations of curvature. In a second step, we have adapted complex shape models comprising a large number of sharp edges. Highly graded meshes can be obtained while meeting good shape

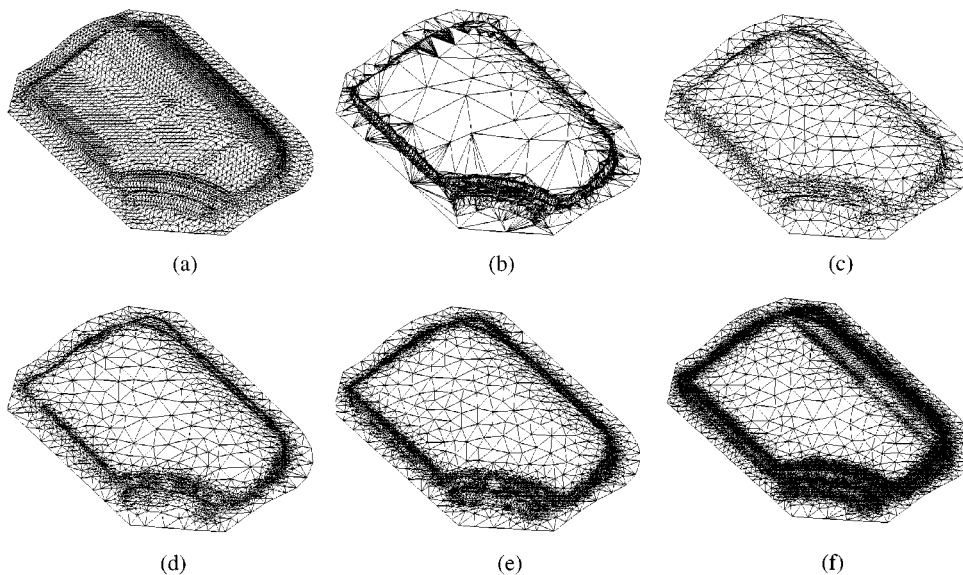


Figure 8. Front door of a car (Numisheet99 benchmark): (a) initial mesh; (b) mesh at $\varepsilon = 0.01$. No size smoothing procedure has been applied; (c) mesh at $\varepsilon = 0.05$; (d) mesh at $\varepsilon = 0.02$; (e) mesh at $\varepsilon = 0.01$; (f) mesh at $\varepsilon = 0.001$.

quality requirements. The CPU time of the process on a PC Pentium II 333 MHz is less than 1 min for models composed of 10 000 elements. We noticed that a majority of the CPU time is spent in the remeshing process and especially in projection operations on the reference mesh which explains that the CPU time depends on the number of elements on the initial mesh.

The first example (Figure 8) is a front door of a car (Numisheet99 Benchmark). Some connecting fillets have been represented roughly on the initial mesh which explains the presence of planar areas close to highly curved zones. This example show, as we expected, that the accuracy of the interpolation greatly depends on the initial mesh. We can see in Figure 8(b) the mesh when no size smoothing procedure is applied.

This example shows that our approach can be applied in order to get a new mesh which preserves with accuracy the geometrical features of the model while reducing the number of elements. The examples also demonstrate the geometrical criterion ability to capture curvature variations. The results of the remeshing process at different geometrical error criteria are presented in Table I.

Table I. Front door of a car (Numisheet99 benchmark).
Mesh statistics.

| Numisheet | Nele | Qmin | CPU(s) |
|-----------------------|--------|------|--------|
| Initial mesh | 8813 | 0.25 | |
| $\varepsilon = 0.05$ | 2936 | 0.48 | 4 |
| $\varepsilon = 0.02$ | 6800 | 0.49 | 9 |
| $\varepsilon = 0.01$ | 13 074 | 0.43 | 21 |
| $\varepsilon = 0.001$ | 31 444 | 0.43 | 69 |

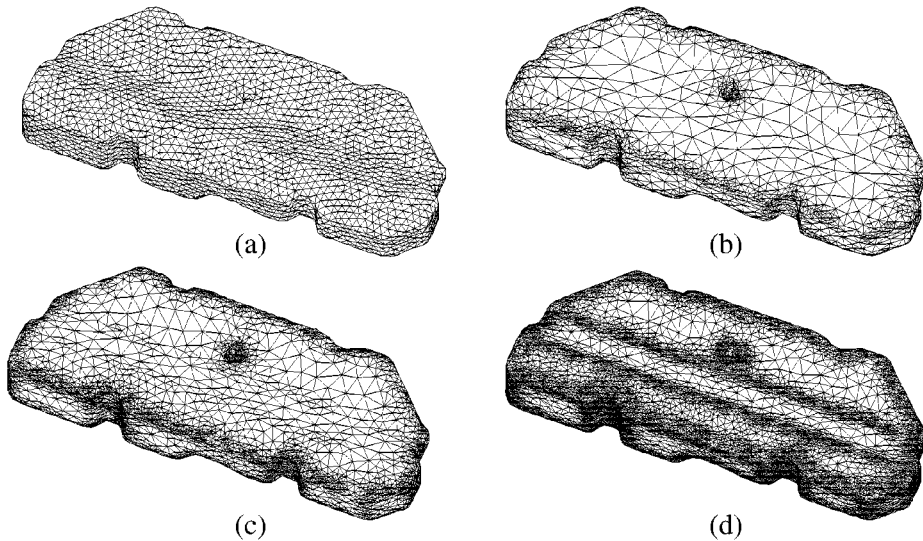


Figure 9. Tank: (a) initial mesh; (b) mesh at $\varepsilon = 0.1$; (c) mesh at $\varepsilon = 0.01$; (d) mesh at $\varepsilon = 0.001$.

This example (Figure 9) shows that a good quality mesh can be obtained even when planar surfaces meet highly curved areas (especially in the area close to the cylindrical hole) and also provides a good illustration of the simple, however efficient, size smoothing procedure. Results of the adaptation process are provided in Table II.

Figure 10 show the mesh of a polymetallic mass (Hajar mines in Morocco). This example provide a significant overview of the main difficulties that can be encountered: high curvature, sharp edges, complex shapes, high number of elements on the initial mesh. We remark that the distribution of elements of the initial mesh (Figure 10(a)) do not follow the curvature of the model and this example demonstrates the efficiency of both enrichment and decimation processes. The results of the remeshing processes are given in Table III.

The last example is a steering gearbox (courtesy of Peugeot SA). The size of the initial mesh shown in Figure 11(a) is almost regular. The example combines the following difficulties: high curvature in the area of connecting fillets and complex shapes due to the presence of many sharp edges. Mesh statistics are provided in Table IV.

Table II. Tank mesh statistics.

| Tank | Nele | Qmin | CPU(s) |
|-----------------------|--------|------|--------|
| Initial mesh | 5990 | 0.80 | |
| $\varepsilon = 0.1$ | 5941 | 0.57 | 5 |
| $\varepsilon = 0.01$ | 13 562 | 0.47 | 11 |
| $\varepsilon = 0.001$ | 31 488 | 0.63 | 34 |

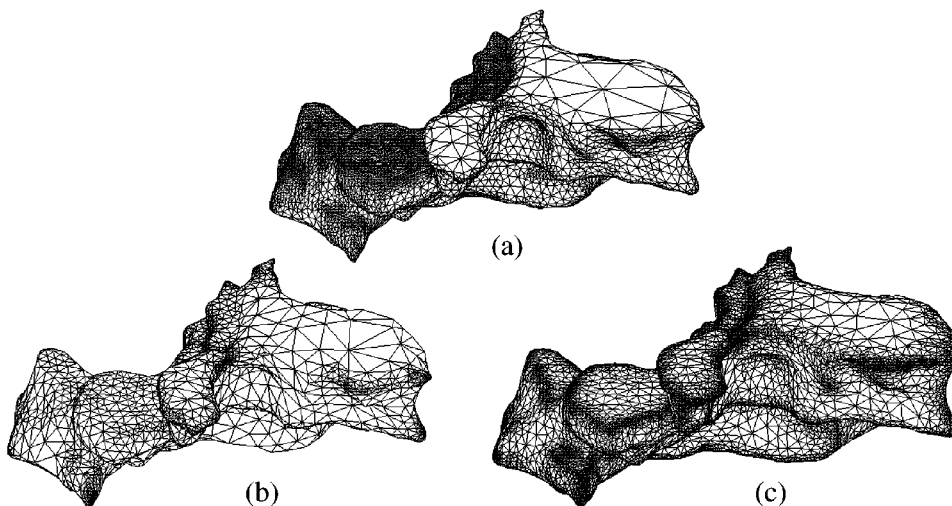


Figure 10. Polymetallic mass (Hajar mines): (a) initial mesh; (b) mesh at $\varepsilon = 0.1$; (c) mesh at $\varepsilon = 0.01$.

Table III. Polymetallic mass (Hajar mines). Mesh statistics.

| Hajar | Nele | Qmin | CPU(s) |
|----------------------|--------|----------|--------|
| Initial mesh | 17 010 | 0.278273 | |
| $\varepsilon = 0.1$ | 6264 | 0.34 | 17 |
| $\varepsilon = 0.01$ | 31 166 | 0.39 | 85 |

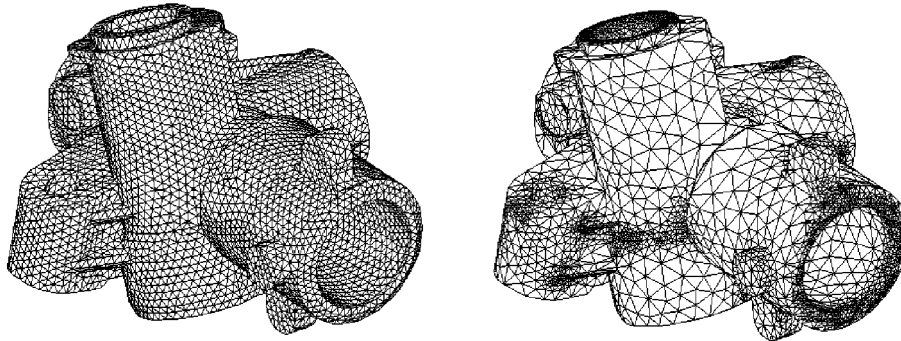


Figure 11. Steering gearbox (courtesy of Peugeot SA): (a) initial mesh; (b) mesh at $\varepsilon = 0.01$.

Table IV. Steering gearbox (courtesy of Peugeot SA). Mesh statistics.

| PSA | Nele | Qmin | CPU |
|----------------------|--------|------|-----|
| Initial mesh | 23 634 | 0.09 | |
| $\varepsilon = 0.01$ | 37 004 | 0.12 | 126 |

6. CONCLUSION

We have presented a method to build a three-dimensional adapted surface mesh with respect to a mesh size map driven by surface curvature from the only data of an initial mesh. The results show that the method can be easily coupled with any estimator during a finite element procedure and demonstrate the ability of the Hermite diffuse approximation technique to respect the geometry. The approach can be used in the context of a 3D adaptive process to modify the boundary mesh with respect to a size map provided as a result of *a posteriori* error estimator and we plan to couple the surface mesh generator to a 3D mesh generator which respects a size criterion [26]. We are also using the high-resolution curvature analysis provided by Hermite diffuse approximation to geometrical feature recognition. Results already show that the basic surfaces (plane, cone, torus, spheres) can be identified.

REFERENCES

1. Beckers P, Zhong HG. Mesh adaptation for 2D stress analysis. *2nd International Conference on Computational Structures Technology*, Athens Greece, August 30–September 1, published in *Advances in Post Processing for Finite Element Technology*. Civil Comp Ltd: Edinburgh, Scotland, B.H.V, Edn. Topping and Papadrakakis, 1994; 47–59.
2. Zienkiewicz OC, Zhu JZ. A simple error estimator and adaptive procedure for practical engineering analysis. *International Journal for Numerical Methods in Engineering* 1987; **24**(2):337–357.
3. Zienkiewicz OC, Zhu JZ. The superconvergent patch recovery and a posteriori error estimates. Part 1: the recovery technique. *International Journal for Numerical Methods in Engineering* 1992; **33**(7):1331–1364.
4. Zienkiewicz OC, Zhu JZ. The superconvergent patch recovery and a posteriori error estimates. Part 2: error estimates and adaptivity. *International Journal for Numerical Methods in Engineering* 1992; **33**(7):1365–1382.
5. Bonet J. Error estimators and enrichment procedures for the finite element analysis of thin sheet large deformation processes. *International Journal for Numerical Methods in Engineering* 1994; **37**(9):1573–1591.
6. Chew LP. Guaranteed-quality mesh generation for curved surfaces. *Proceedings of the 9th Annual Symposium on Computational Geometry*. ACM: New York, 1993; 274–280.
7. Aftosmis MJ. Automatic generation of CFD-ready surface triangulations from CAD geometry. *AIAA Paper 97-0775*, Jan 99.
8. De Cougny HL, Shephard MS. Surface meshing using vertex insertion, *5th International Meshing Roundtable*, Sandia National Laboratories, October 1996; 243–256.
9. Noel F, Leon JC, Trompette P. A new approach to free-form surface mesh control in a CAD environment. *International Journal for Numerical Methods in Engineering* 1995; **38**(18):3121–3142.
10. Léon J-C. *Modélisation et construction de surfaces pour la CFAO*. Edn. Hermes; Paris, 1991.
11. Tristano JR, Owen SJ, Canann SA. Advancing front surface mesh generation in parametric space using a Riemannian surface definition. *7th International Meshing Roundtable*, Dearborn, MI 1998.
12. George PL, Borouchaki H. *Delaunay Triangulation and Meshing Application to Finite Elements*. Edn. Hermes: Paris, 1998.
13. Véron P, Léon J-C. Static polyhedron simplification using error measurements. *Computer Aided Design* 1997; **4**:287–298.
14. Hattangady N. Coarsening of mesh models for representation of rigid objects in finite element analysis. *International Journal for Numerical Methods in Engineering* 1999; **44**:313–326.
15. Lau TS, Lo SH. Finite element mesh generation over analytical surfaces. *Computers and Structures* 1996; **59**(2):301–309.
16. Borouchaki H. Geometric surface mesh. *IDMME98*, 1998; 343–350.
17. Walton DJ, Meek DS. A triangular G1 Bézier patch from boundary curves. *Computer-Aided Design* 1996; **28**:113–123.
18. Lancaster P, Salkauskas K. Surfaces Generated by moving least squares methods. *Mathematics of Computation* 1981; **155**(37):141–158.
19. Nayroles B, Touzot G, Villon P. 'L' approximation diffuse. *Comptes Rendus de l'Académie des Sciences Paris II* 1991; **313**:293–29.
20. Breitkopf P, Touzot G, Villon P. Explicit form and efficient computation of MLS shape functions and their derivatives. *International Journal for Numerical Methods in Engineering*, in press.
21. Guo YQ, Batoz JL, Bouabdalah S, Mercier F, Naceur H. Some recent developments of the simplified inverse approach for sheet metal forming analysis. *Advances in Computational Structures Technology*. Edn. Topping: Edinburgh, Scotland, 1998; 333–342.
22. Gray A. *A Monge Patch, Modern Differential Geometry of Curves and Surfaces with Mathematica* (2nd edn). CRC Press: Boca Raton, FL, 1997; 398–401.
23. Belytschko T, Krongauz Y, Organ D, Fleming M, Krysl P. Meshless methods: an overview and recent developments. *Computer Methods in Applied Mechanics and Engineering, special issue on Meshless Methods* 1996; **139**:3–47.
24. Villa R. Etude théorique et expérimentale de l'extrusion-soufflage de corps creux en polymères. *Doctor Thesis*, ENSMP, 1997.
25. Rassinoux A. Generation and optimization of tetrahedral meshes by advancing front technique. *International Journal for Numerical Methods in Engineering* 1998; **41**:651–674.
26. Löhner R. Some useful data structure for generation of unstructured grid. *Applied Numerical Methods* 1988; **3**:123–135.
27. Liszka TJ, Duarte CAM, Tworzydło WW. Hp-Meshless cloud method. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**:263–288.