Automatic quadrilateral and hexahedral finite element mesh generation : review of existing methods

Valérie Mounoury — Olivier Stab

Centre CAO-Robotique
Ecoles des Mines de Paris
60 bd Saint-Michel
75272 Paris cedex 06

ABSTRACT. *In this paper, automatic quadrilateral and hexahedral mesh generation methods are reviewed. A classification is proposed and each method is explained. The first class deals with conversion techniques. In the second one, generation methods are studied. Decompositions into topologically simple subdomains are often used. Most of the methods deal with two dimensional mesh generation techniques and with their possible extension to three dimensional domains. A few recent techniques which produce hexahedral elements are presented. The industrial use of certain methods is mentioned.*

RÉSUMÉ. *Cet article propose un état de l'art en matière de maillage automatique en quadrilatères et en hexaèdres. Chaque méthode est classée et les algorithmes sont présentés dans leurs grandes lignes. On distingue d'une part, les méthodes dites de conversion et d'autre part, celles dites de génération. Dans cette dernière classe, les techniques de décomposition en primitives topologiques simples sont les plus fréquentes. La plupart des méthodes sont initialement présentées en 2D et leur extension possible au cas tridimensionnel est finalement discutée. Peu de méthodes produisent effectivement des maillages hexaédriques. L'utilisation en milieu industriel de telle ou telle méthode est signalée.*

KEY WORDS : *automatic mesh generation, conversion, node connection, grids, quadtrees, iterative and recursive decomposition, feature recognition, mesh templates, mapping techniques, transitional meshing.*
MOTS-CLÉS : *génération automatique de maillages, conversion, connexion de noeuds, grilles et arbres quaternaires, décomposition itérative et récursive, reconnaissance des particularités topologiques, modèles de maillage, techniques de transformation, maillages de transition.*

1. Introduction

In recent years, researchers' attention has been focused on automatic finite element mesh generation, which is considered as a bottleneck in the finite element method. Most of the works deal with the discretization in triangles and tetrahedra. These elements are easier to be generated on complex 2D and 3D geometries and large meshes are created at a very low CPU cost. However, some finite element analysts still prefer quadrilateral and hexahedral elements due to a better results quality, using an equal number of degrees of freedom ([RAM 94][VIN 94]). This paper reviews automatic meshing techniques that generate such elements.

The mesh generation methods are classified according to their input data. Two classes are defined :

— conversion algorithms, working on domains already meshed into triangular or tetrahedral elements ;

— generation algorithms for which input data are a boundary representation or a constructive solid geometry tree of the domain.

Whereas semi-automatic techniques are as frequently used for hexahedral meshing as for quadrilateral meshing, very few automatic methods have been proposed for three dimensional domains. 2D techniques are the mostly reviewed here. Three techniques converting triangular meshes into quadrilateral ones ([LIU 90][HEI 83][JOH 91]) are presented. Generation methods are mostly used ; most of them are based on the decomposition into topologically simple regions ([ZHA 92][GUR 92][YEU 73][RAZ 89]) which can be meshed by different kinds of classical mapping techniques ([CHI 91][WEL 88][BAL 85][HOY 92]) or by other techniques recently developed ([CHE 89]).

2. Conversion methods

During the past twenty years, many works were done to automate mesh generation on any domain. The most efficient algorithms produce exclusively triangular 2D meshes. So some researchers have decided to use this capabilities to develop new quadrilateral element meshing methods.

These methods are based on the conversion of triangular meshes into quadrilateral ones. The easiest conversion method consists in dividing each triangle into three quadrilaterals (in 3D, each tetrahedron into four hexahedra), but the resulting elements are not well-shaped because of small angles at existing nodes shared by more than four triangles (in 3D, more than eight tetrahedra), and large angles at new nodes ([HO 88]). Conversion techniques deal only with combination algorithms.

The problem is to combine the most of triangular elements while ensuring a good quality for the quadrilaterals. The conversion process can be divided into three main steps :

— combination of triangular elements,
— elimination of remaining elements,
— improvement of the element shape.

The algorithms differ in the combination and elimination strategy which is based on three essential points :

— the element processing order on which depends the number of remaining triangles,

— the criteria to combine triangles so that quadrilateral elements can be well-shaped,

— the operators used to eliminate the remaining triangles.

These points are gone over for three conversion algorithms developed by Heighway (arbitrary combination), Johnston (frontal method) and Liu (rules system).

2.1. *Arbitrary combination*

Element processing order : Heighway ([HEI 83]) defines a priority for each triangle : the less candidates it has, the higher the priority is. A candidate is a neighbour that forms a quadrilateral with no reflex interior angle. Beginning with the elements presenting the smallest number of candidates reduces the number of remaining triangles.

Criteria for the combination : if the base element has more than one candidate then the choice is arbitrary.

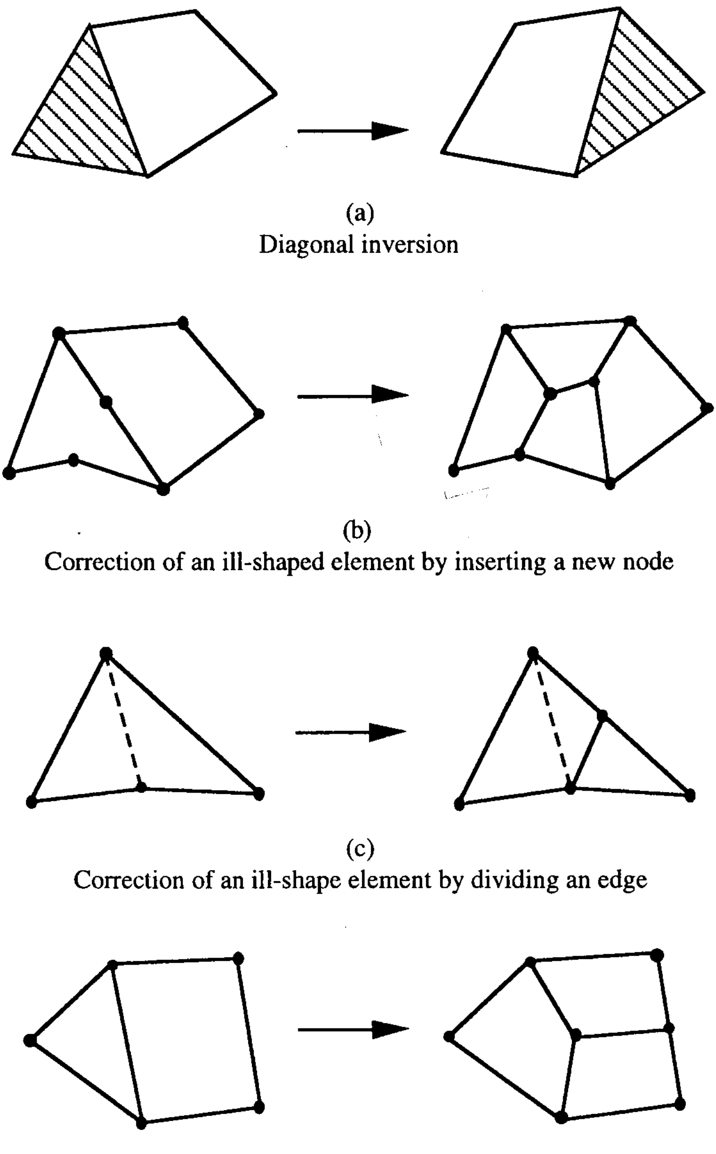
Operators for the elimination : each remaining element can "walk" towards another one across the mesh by diagonal inversion (see Figure 1(a)). Both are paired to form a quadrilateral. As no new node is generated, the conversion is complete only when the number of elements in the initial mesh is even. Ill-shaped quadrilaterals can be formed by the "walking" process. The correction consists in adding a node in an adjacent quadrilateral (see Figure 1(b)). Finally, the element shape is improved by a node replacement process.

2.2. *Frontal method*

Element processing order : Johnston's algorithm ([JOH 91]) uses the advancing front technique ; the boundary edges form the initial front. The element priority is defined from the number of candidates, like Heighway, and also from its position with regard to the front : the triangles sharing edges with the front have a higher priority, so the conversion proceeds from the boundaries towards the interior of the domain.

Criteria for the combination : triangles with the highest priority are considered. If there are several such elements, the one that forms the best-shaped quadrilateral with the base element is chosen. If some elements have bad interior angles, they are corrected by nodal geometry changes (see Figure 1(c)).

Operators for the elimination : isolated triangles are combined with adjacent quadrilaterals by the operator shown in Figure 1(d) ; so the conversion is complete without any special condition on the initial mesh.



(a)
Diagonal inversion

(b)
Correction of an ill-shaped element by inserting a new node

(c)
Correction of an ill-shape element by dividing an edge

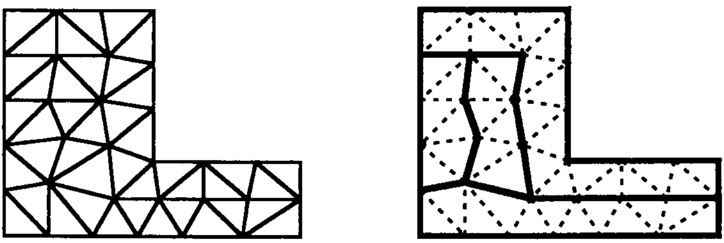
(d)
Combination of a triangle with an adjacent quadrilateral

Figure 1. *The operators used in the conversion methods*

2.3. *Rules system*

Element processing order : the initial step of Liu's algorithm ([LIU 90]) is the construction of element layers in the triangular mesh, which reduces the search space. A layer is constructed by grouping triangles with a maximum of two edges connected to other elements in the same layer (Figure 2). Afterwards, the layer processing order does not matter. The number of triangles in a layer must be even for a complete conversion ; if not, a triangle is divided.

Criteria for the combination : the system considers four triangular elements within a layer at a time and through a set of rules (now 27) chooses the best adapted pattern. If the first two triangles form an ill-shaped quadrilateral, Liu lists 11 patterns covering all cases of topological and geometrical configurations to mesh the six-sided polygon defined by the four triangles. Figure 3 displays some examples. Poor quality elements can occasionally be generated when using type 10 or pairing two remaining elements in a layer. Flat angles are eliminated by node addition or elimination.



(a)
initial mesh

(b)
one of the possible layer constructions

Figure 2. *An example of layer construction*

Operators for the elimination : before the combination, a triangle is divided in layers with an odd number of elements. As the rules always combine two or four elements, no triangle remains and the conversion is complete.

The major problem of these methods is that the quality of a converted mesh highly depends on the shape of the initial elements. Good triangular meshes do not necessarily provide good quadrilateral meshes. Rassineux ([RAS 92]) has suggested adapting the triangular meshes to the quadrilateral element generation with a more suitable point distribution : points are created on an orthogonal grid ; the triangles are not equilateral but the combination of two elements produces a perfect square.

These methods may be extended to three dimensional domains. More combinations must be tested, since one hexahedron is formed by five or six tetrahedra.

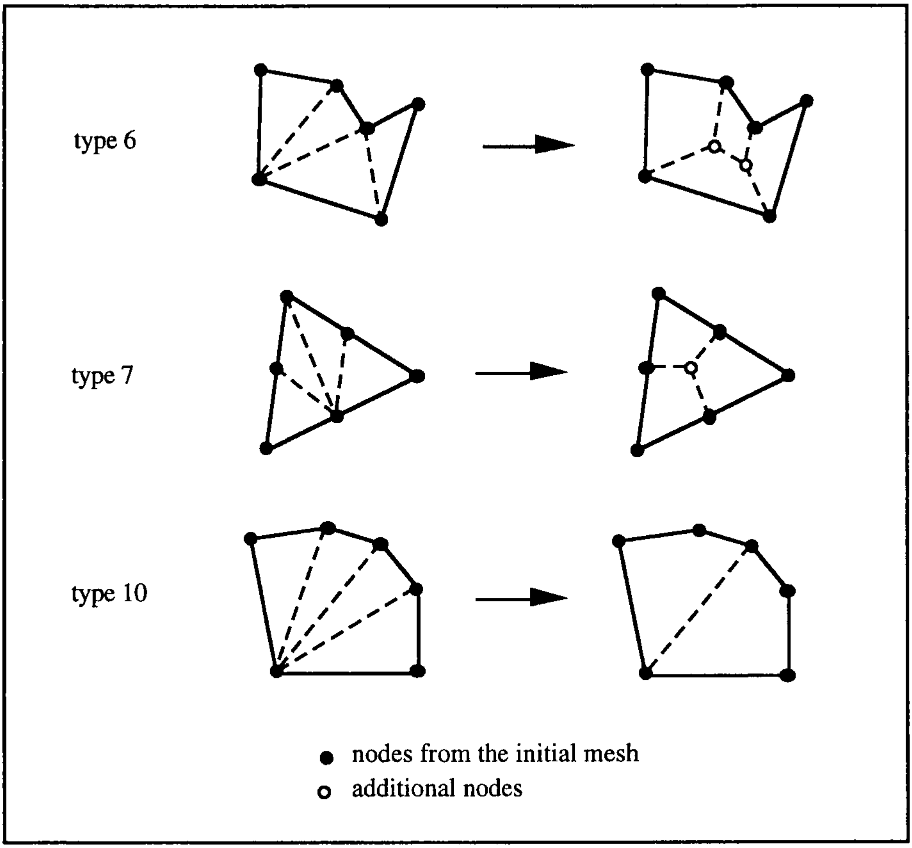


Figure 3. *Some examples of Liu's mesh templates*

3. Generation methods

Ho-Le's classification ([HO 88]) is adopted for the quadrilateral mesh generation review : as a mesh consists of a set of nodes and a set of elements, the basis for this classification is the temporal order of the creation of these sets. So there are four mesh generator classes :

— generators which create the nodes first : the node connection approach,

— generators which create the mesh topology first : the grid-based approach,

— generators which create the nodes and the elements simultaneously : the geometric decomposition,

— generators which adapt mesh templates : the decomposition into primitives.

As the latter class is most often used, it is covered exclusively in Section 4.

3.1. *Node connection approach*

This mesh generation approach consists of two main steps : point generation and mesh construction over these points. Lee ([LEE 83]) has proposed an algorithm for the quadrilateral element generation.

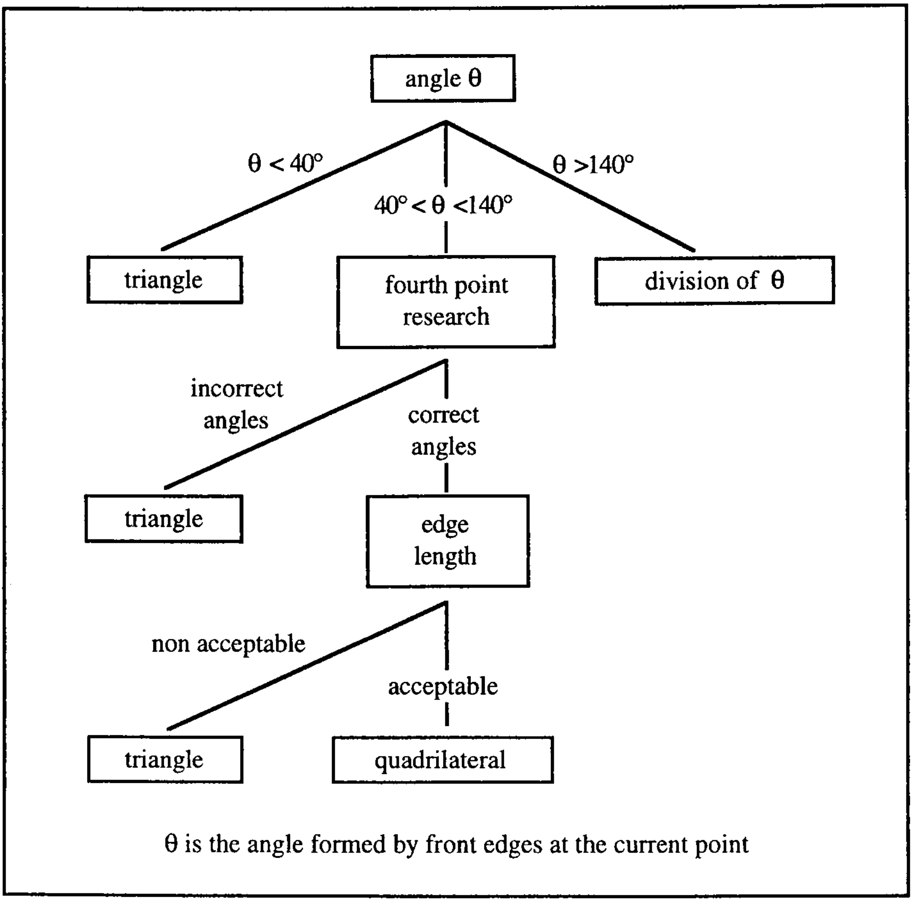


Figure 4. *Simplified decision tree*

The points generated on the domain must be well-distributed. Lee's algorithm works on a CSG representation of the domain. Distribution templates are used ; they are adapted to quadrilateral elements generation and take into account the topology of each primitive. The mesh construction algorithm is based on the concept of the front. The initial front is empty. Each new element is generated ahead of the front, leaning on one or more front edges (if some already exist). The search of the fourth point is based on angle tests. If any front point satisfies the conditions, the algorithm looks for a new point near the ideal position. When the creation of a well-shaped quadrilateral fails, triangles are formed. A simplified decision tree is presented in Figure 4.

For the extension to hexahedral meshing, Lee observes that the decision tree will be more complicated. No algorithm for 3D node connection is published yet.

3.2. *Grid-based approach*

These meshing methods are based on a simple observation : with an adjustment to the domain boundaries, a grid may become a mesh. The algorithms differ in the kind of grid and the adjustment technique.

3.2.1. *Regular grids*

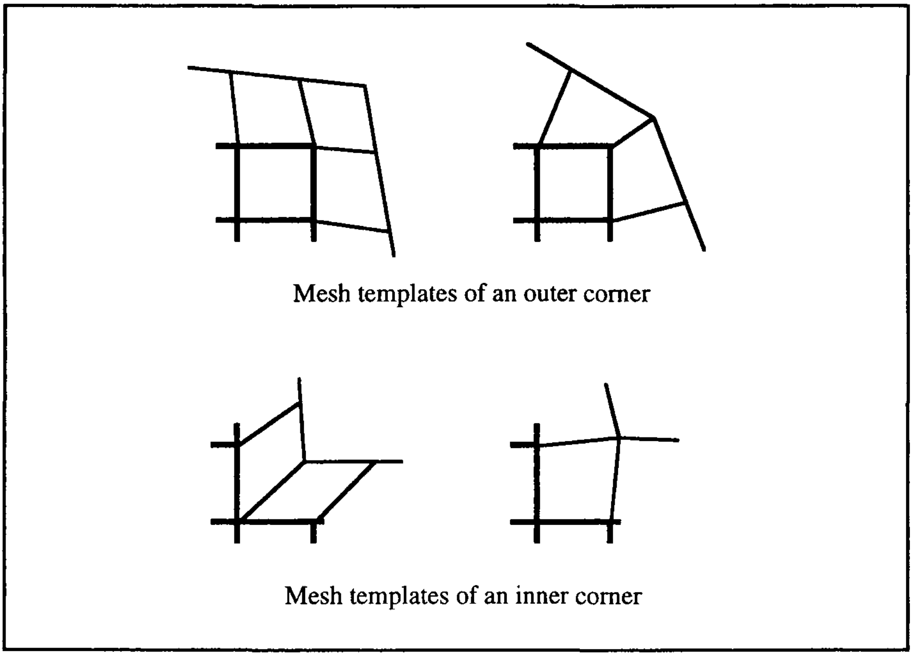


Figure 5. *Mesh templates of the corners*

A regular grid can be superimposed on the domain to get a uniform mesh. Whereas some grid cells are interior or exterior, others intersect the boundaries. Schneiders ([SCH 92]) only keeps interior ones. The remaining area between the domain boundaries and the interior cells is meshed in two steps :

— templates which minimize simultaneously edge and angle sizes are chosen at the "corners" (see Figure 5),

— areas between two "corners" are regularly subdivided.

Although this type of meshing guarantees an all-quadrilateral mesh with excellent interior elements, only uniform meshes can be generated.

From an initial regular grid, Garcia ([GAR 91]) generates a mesh using a grid adjustment algorithm. A function of heterogeneity, which represents geometrical and physical characteristics, is defined for each element. The method consists of an iterative process which applies forces in order to reduce the size of the heterogeneous elements. The deformations are calculated by two ways : in the DEGA (Discret Elastic Grid Adjustment) method, the mesh is considered as an elastic grid ; in the FEMA (Finite Element Mesh Adjustment) method, the elements are continuous domains with elastic laws of behaviour.

Garcia has also studied this method for three dimensional meshing. In this case, an additional processing ensures the flatness of the element faces after the node displacements. The experimentation is done only on two dimensional geometries. For instance, the method has been used in reservoir simulation ([GAR 92]) and Figure 6 shows a geological interpretation of an oil reservoir digitized into rock-types.

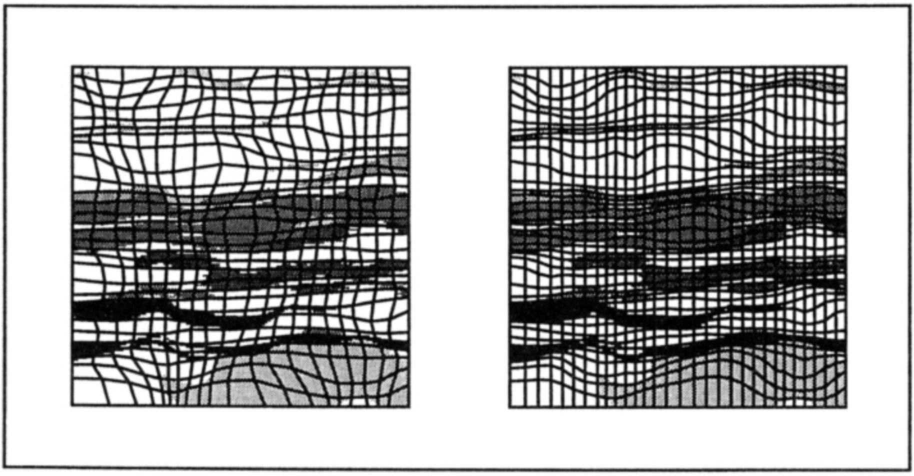


Figure 6. *Oil reservoir meshed with the DEGA method (different parameters)*

3.2.2. *Quadtrees*

A quadtree is based on a recursive spatial decomposition that produce a set of disjoint squares called quadrants. This base structure was modified by Shephard and Yerry ([SHE 83]) to suit mesh generation :

— internal quadrants were subdivided according to the mesh control information,

— boundary quadrants were cut to approximate the domain boundary,

— the quadtree was refined until only one tree level exists between adjacent quadrants.

Templates were used to mesh interior quadrants and boundary quadrant aswell. This caused the limitation of the geometric representation of the latters. Baehmann ([BAE 87]) uses the same structure but brings some improvements to the original algorithm :

— For an accurate representation of the domain geometry, real intersections are used to cut boundary quadrants. Polygons with narrow corners or tight geometries may be produced. They are processed by dividing the polygons and eliminating the short boundary segments.

— As the boundary quadrants can assume an infinite number of shapes (due to real intersections), an algorithm based on iterative decomposition is used to generate the elements. Quadrilateral elements are removed from the polygons until only polygons with four, five or six nodes remain. Templates can then be used to complete the mesh.

A smooth procedure is performed on the quadrants before the meshing process and on the elements after the process termination. Baehmann's algorithm produces all-quadrilateral meshes with better-shaped elements compared to the initial Yerry and Shephard's mesh generator.

3.3. *Geometric decomposition*

These methods generate simultaneously the nodal geometry and the mesh topology. They are based on recursive or iterative decomposition.

3.3.1. *Recursive decomposition*

Triquamesh ([SLU 82][JAI 86]) is one of the few mesh generators using this method. The user supplies the boundary of the domain and a value at all geometric points corresponding to the element size. The following steps describe the mesh generation process :

— nodes are generated on the curves of the boundary according to the user given values ;

— concave regions are subdivided using the best splitting lines until only convex subregions remain ;

— into convex subregions, layers of elements are cut off along a straight line as long as values of the angles are between 40 and 140 degrees. Remaining areas are divided into new convex subregions using the best splitting lines ; and so on until all subregions contain either four or six nodes ;

— convex subregions with six nodes are meshed with either two, three, four, or five quads depending on their configuration.

The core of this method is the determination of the best splitting lines. A splitting line always connects two existing nodes. The first one is chosen to be the most concave point. The second one is selected with respect to some geometrical criteria :

— the splitting line should be as short as possible,

— the angles formed by the splitting line at the boundary should be nearly multiples of 90 degrees,

— the number of nodes generated on the splitting line should be minimum while respecting the neighbouring node density.

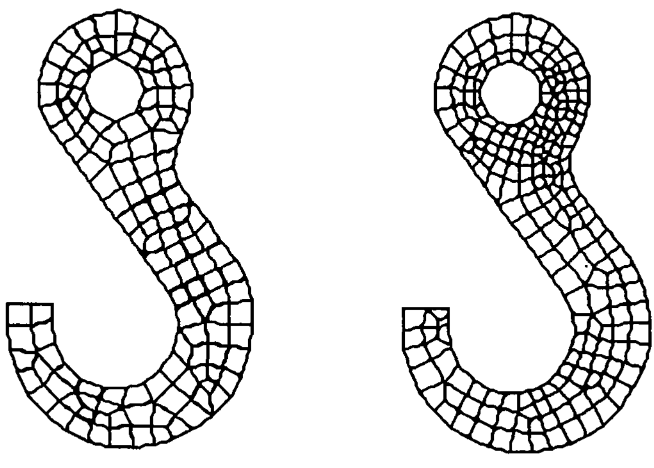
Triquamesh is the mesh generator available in the I-DEAS software by SDRC. Figure 7 shows the exemple of the crane-hook (found in [SLU 82]), meshed with I-DEAS version VI. The first mesh (Figure 7(a)) is uniform, the second one (Figure 7(b)) is locally refined by a factor 2.

Talbert and Parkinson ([TAL 90]) developped a mesher based on the same algorithm with some modifications :

— to improve the accuracy of the finite element analysis, a layer of very well-shaped elements are generated by offsetting the boundary ;

— refinement coefficients control the nodes placement in transition regions by specifying the fraction of a splitting line that must be discretized by equally spaced nodes.

Control Data and Aerospatiale propose a complete family of tools called ICEM CFD/CAE for analysis pre-processing. ICEM QUAD, their unstructured surface mesh generator is the one developped by Talbert and Parkinson ([BER 94][RAN 92]). Figure 8 shows a mesh from ICEM version 3.2.



(a) (b)

Figure 7. *Meshes of a crane-hook by the Triquamesh algorithm*

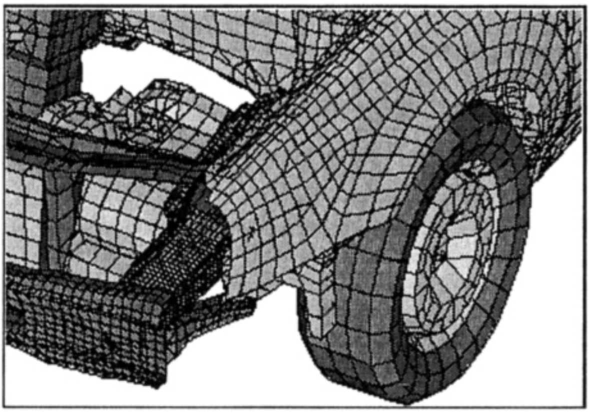


Figure 8. *a Golf meshed in ICEM by Talbert and Parkinson's algorithm*

Triquamesh has extended the principle of recursive decomposition to 3D geometries. A loop is constructed around the volume by connecting edges of neighbouring faces with concave angles. This loop is generally non-planar but can be meshed by the same algorithm as the boundary faces. This 3D algorithm does not allow hexahedral elements generation : a subregion with a boundary meshed in all-quadrilateral elements is not always divisible into hexahedra without topological modifications on the boundary mesh, which then introduce incompatibilities between adjacent subregions.

3.3.2. *Iterative decomposition*

These methods use the advancing front technique. The algorithms consist of the following steps :

— if it is required, cut lines transform multiply connected regions into simply connected and/or convex regions ;

— nodes are generated on boundary segments and cut lines according to the user given densities. The created edges form the initial front ;

— an iterative process generates one element or a row of elements at once. The front is updated in each loop.

Three algorithms are reviewed. They differ especially in the procedure of the element generation.

In the algorithm proposed by Zhu and Zienkiewicz ([ZHU 91]), regions must be simply connected with an even number of nodes on the boundary. A quadrilateral element is generated in four steps :

— a front segment AB is chosen ;

— according to the density, a first triangle ABC is created. C is either a front node or a new node. The front is updated. If C is a front node, the front is subdivided into two subfronts ;

— a second triangle ACD or CBD is created. If there are two subfronts, the second triangle is formed in the loop containing an odd number of edges. If D introduces two new subfronts, they should contain an even number of edges ;

— the two triangles are combined to form a quadrilateral element.

As the number of boundary nodes is even, this procedure ensures that the domain will be meshed into all-quadrilateral elements. The generation process does not take into consideration the element shape. A smoothing process and topological operators are used to enhance the mesh quality.

In Sezer and Zeid's algorithm ([SEZ 91]), the iterative decomposition process applies to simply connected convex regions. Rows of elements are generated by connecting front nodes with offsetting nodes. The latter are determined by a ray-based free-form offsetting :

— rays in the X-direction are generated on the domain ;

— the intersections of the rays with the front edges determine a set of imaginary nodes. News nodes are created by offsetting these ones along the rays ;

— some new nodes are eliminated to respect the density and avoid bad elements ;

— each front node is then connected to the nearest offsetting node to form either triangular or quadrilateral elements.

When the element generation is complete, triangles and quads are divided to produce an all-quadrilateral mesh.

Blaker's algorithm ([BLA 91]) generates rows of quadrilateral elements in multiply connected regions. The paving operation proceeds as follows :

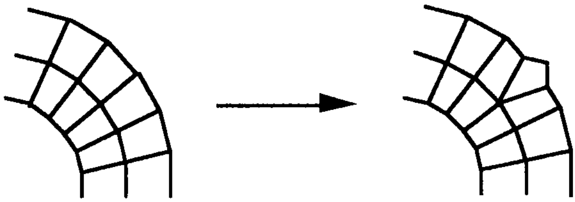
— the begining and ending nodes of a row are chosen according to their interior angle (the angle between two consecutive front edges) to form the most regular shapes ;

— each node of the selected row produces 1, 3 or 5 projected nodes depending on whether front angle is near 180, 270 or 360 degrees. The distance of the projection is chosen so that new quadrilaterals can be well-shaped ;

— nodes are connected to form new elements ;

— as layers of elements begin to coincide (small angles) or to overlap, they are connected together by a seaming process.

To avoid expansion and contraction of the element size, due respectively to concave or convex curves of the boundary, 'wedges' and 'tucks' are added in the mesh (Figure 9). A mesh smoothing guarantees well-shaped elements after each step of the paving process.



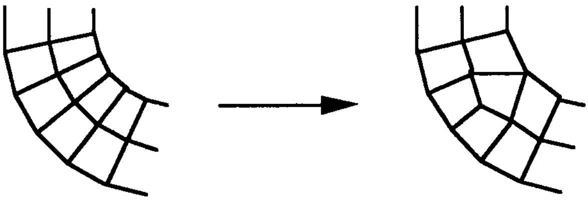
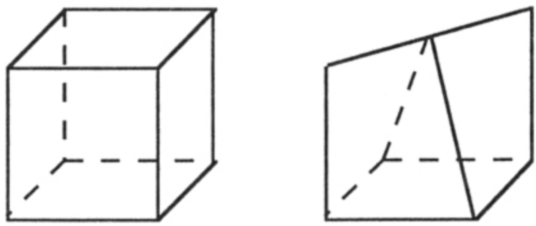


Figure 9. *Cor-rection of element size expansion and contraction*

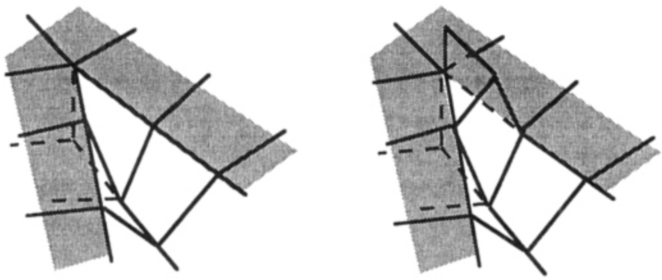
PSA uses Stresslab+, the Computer Vision tool Stresslab in which Blaker's 2D paving algorithm has been added for internal custom.

Blaker has extended his method to mesh 3D domains ([BLA 93]). The quadrilateral discretization of the boundary faces forms the initial front. Iteratively, a portion of the current front is projected to generate new hexahedral elements. 'Seams' and 'wedges' are used to correct unacceptable configurations. When two faces sharing one or more edges are very close, they are seamed. In the other cases, elements with five faces, called wedges, are used. For a valid hexahedral mesh, a wedge must be resolved by forward or backward propagation until it terminates either in the current front or in the domain boundary. Propagations terminating in the current front are always preferred, as they leave the exterior boundary intact. Figure 10 shows a configuration where a wedge is needed and the mechanism of forward propagation.

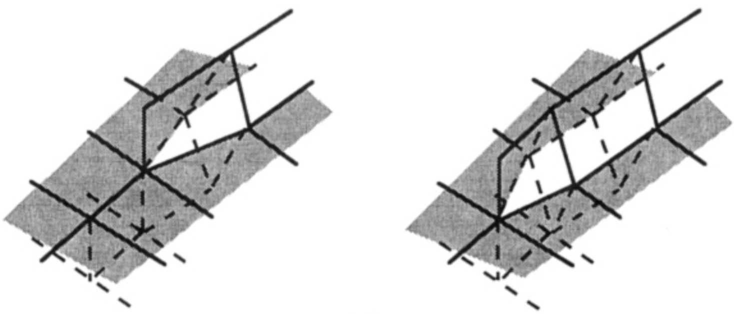
The requirements or problems of the convergence are not discussed. According to the author, simple geometries were successfully meshed but no example is given in the paper. Seams and wedges are especially introduced as powerful tools which may allow great progress in hexahedral meshing.



(a)
A wedge is created by collapsing a face of an hexahedral element



(b)
A configuration where a wedge insertion is needed



(c)
Forward propagation of a wedge

Figure 10. *Wedge utilisation in Blaker's hexahedral mesher*

4. Decomposition into primitives

Among all automatic mesh generation methods, those which use a decomposition into primitives and primitive templates are the most appreciated due to the very well-shaped quadrilateral and hexahedral element generation.

This approach requires the decomposition of the domain into topologically simple subdomains. Quadrilateral and hexahedral primitives are usually used. During the last years, others primitives have been introduced and numerous algorithms have been proposed to automate the decomposition step.

4.1. *Decomposition techniques*

<u>Rule 7</u>: The subdivision line should divide the inner angle at the starting point as equally as possible; an angle less than 45° or larger than 135° is not acceptable.

<u>Rule 8</u>: The dying point of the subdivision line, if it lies on a side of the polygon, should divide the side into two segments as equally as possible.

<u>Rule 9</u>: At the dying point the two inner angles formed by the subdivision line and the two original sides should be as equal as possible; an angle less than 45° or larger than 135° is not acceptable.

Figure 11. *Examples of decomposition rules*

Two of the proposed algorithms are based on a geometric decomposition. Blaker's algorithm ([BLA 88]) removes various primitives (triangles, circles, semi-circles and transional quadrilaterals) iteratively until the initial region is exhaustively decomposed. Zhang's process ([ZHA 92]) divides a domain into quadrilateral primitives using a recursive decomposition. The selection of the splitting lines are controled through rules, which consider geometrical and topological criteria. Some Zhang's rules are given in Figure 11.

Other methods have the advantage of considering any natural axis of the form to mesh. Gursoy's algorithm ([GUR 92]) is based on the medial axis. This method was first used for feature recognition in computer vision ([BLU 73] [PAV 77]). In this technique, every interior point is associated with the nearest point of the domain boundary. The points that can be associated with several boundary points, form the medial axis (see Figure 12). Using this skeleton, the domain is divided into topologically simple subdomains. Adding some edges, the resulting subdomains are either three or four-sided subdomains.

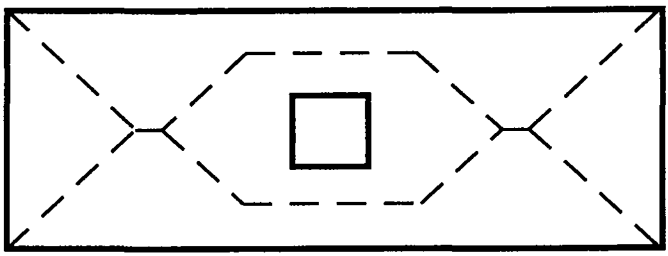


Figure 12. *Medial axis of a polygonal domain*

Yeung's method ([YEU 73]) divides a domain which is defined by a simply connected boundary into meshable regions. For multiply connected regions, some auxiliary cuts are introduced from each inside boundary to the outside boundary. From an initial segment, coarse quadrilaterals are iteratively formed using only the boundary segments. If the number of segments is odd, the last element is a triangle.

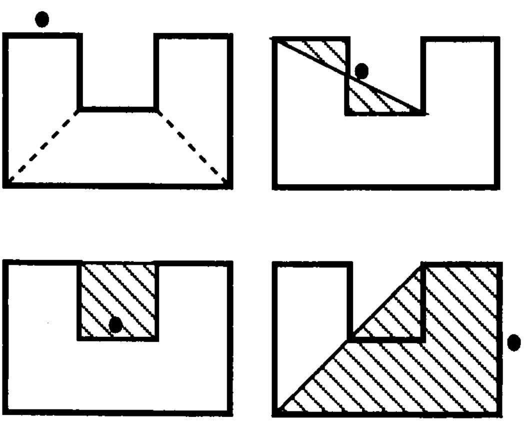


Figure 13. *Decomposition of a domain by Yeung's method*

For an N-sided boundary, there are N possible initial segments and therefore N possible decompositions (Figure 13). The N decompositions have to be valued in order to select the best one. Evaluation functions eliminate reflex angles, detect overlapping regions, and value the shape of the quadrilateral primitives. As no overlapping is allowed, it is not sure that such a decomposition exists for any domain.

The method proposed by Razdan ([RAZ 89]) is the only one which works on three dimensional domains. The algorithm is based on geometrical and topological feature recognition. This technique is also used for automatic simplification of forms ([ANS 88] [FLO 89] [COR 91]).

Feature recognition is heuristic and uses pattern recognition rules. The algorithm works on a boundary representation. Faces are classified according to their number of concave edges. Local configurations are analysed through a set of rules and identified as holes, slots, pockets or protusions. The isolation step divides the object into subdomains, each of them containing one feature ; the procedure is accomplished through bounding box expansion. The last step uses decomposition operators to divide each subdomain according to the feature type into convex volumes which have four, five or six faces. Figure 14 illustrated the three steps process.

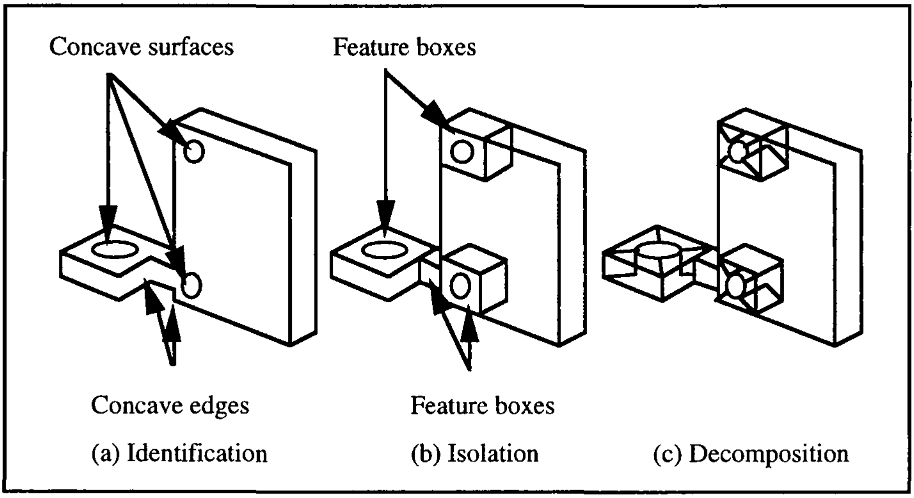
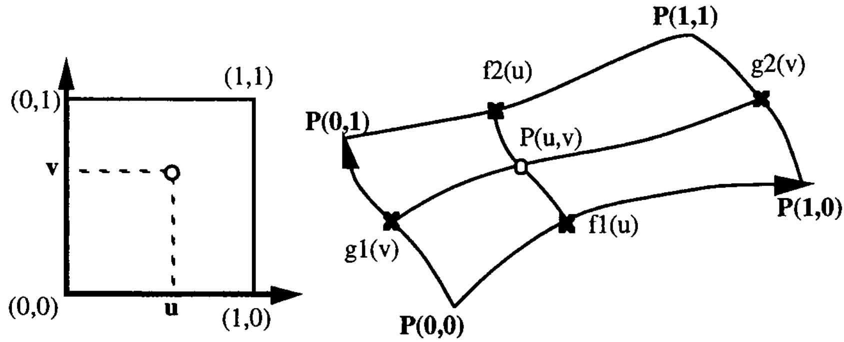


Figure 14. *Feature based object decomposition*

The main difficulty of this method lies in the exact rule writing. The algorithm still has some limits : for example, intersecting features are not recognized, so their decomposition is left to the user.

4.2. *Primitive meshing*

Blaker uses various primitives as triangles, circles, semi-circles. But the other decomposition algorithms provide quadrilateral primitives. Most of the primitive meshing methods are based on mathematical transformations that allow to map a mesh template of a unit square (or a unit cube in 3D) onto a region with the same topology.



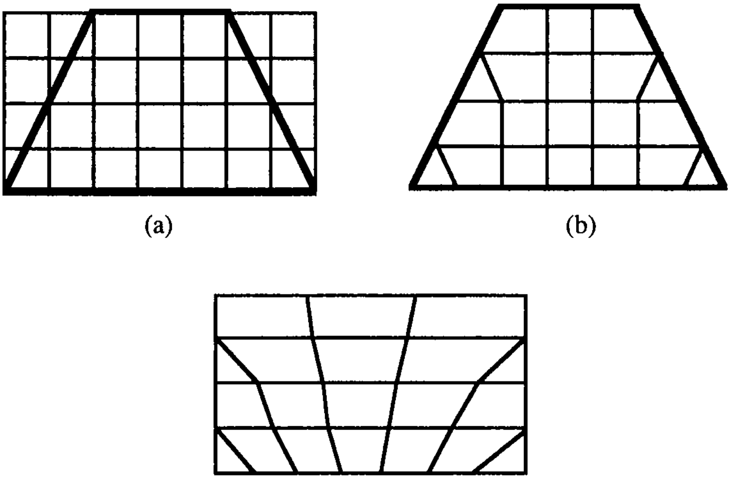
$$P(u,v) = (1-v)\, f1(u) + v\, f2(u) + (1-u)\, g1(v) + u\, g2(v) - (1-u)(1-v)\, P(0,0) - (1-u)\, v\, P(0,1) - (1-v)\, u\, P(1,0) - u\, v\, P(1,1)$$

Figure 15. *Transfinite mapping method*

Isoparametric and transfinite mappings are classical techniques. Whereas the isoparametric mapping approximates the boundary curves with linear, quadratic or cubic polynomials ([ZIE 71]), the transfinite mapping uses their analytic definitions ([HAL 76]). The mesh is achieved by intersecting constant co-ordinate curves in u and v directions (Figure 15).

Transfinite mapping method has the advantage of generating nodes on the exact boundary curves. Moreover, a special case, called discrete transfinite mapping ([HAB 81]), allows to represent a boundary curve as a finite set of points located on the curve. So, some complex curves which have no accurate mathematical description can be used.

These mapping methods impose an important restriction on the mesh topology : two opposite sides of a quadrilateral primitive must have the same number of nodes. Therefore, non-uniform mesh generation is not possible using these techniques. To generate transitional mesh, other methods, some of them based on classical mapping techniques, were developped.


(a) (b) (c)

(a) Superimposition of a super-element on the initial mesh
(b) Super-element mesh after the elimination process
(c) Final transitional mesh of the domain

Figure 16. *Chinnaswamy's method*

Chinnaswamy ([CHI 91]) uses an inverse transfinite mapping. A super-element, superimposed on a regular mesh of the domain, is mapped by an inverse transfinite mapping on a unit square. The external elements are removed ; those which intersect the square boundary are deformed or cut. Afterwards, the mesh is mapped again on the region. Figure 16 shows an example.

The technique described by Wellford ([WEL 88]) also transforms a regular mesh into a transitional one. Two polynomial equations, called node definition functions, where zeros correspond to the nodes of the initial mesh for the first one, and to the nodes of the transitional mesh for the second one, are determined. The "sweeping" function is defined as the weighted sum of these two equations. The limiting values 0 and 1 of the weighting factor are respectively associated with the initial and final node definition functions. The weighting factor is slowly increased to sweep the nodes from their initial position to their final position ; some of them may be driven out of the domain (Figure 17).

The latter two methods allow transitions in the two co-ordinate directions with no restriction on the element densities. The resulting meshes contain some triangles.

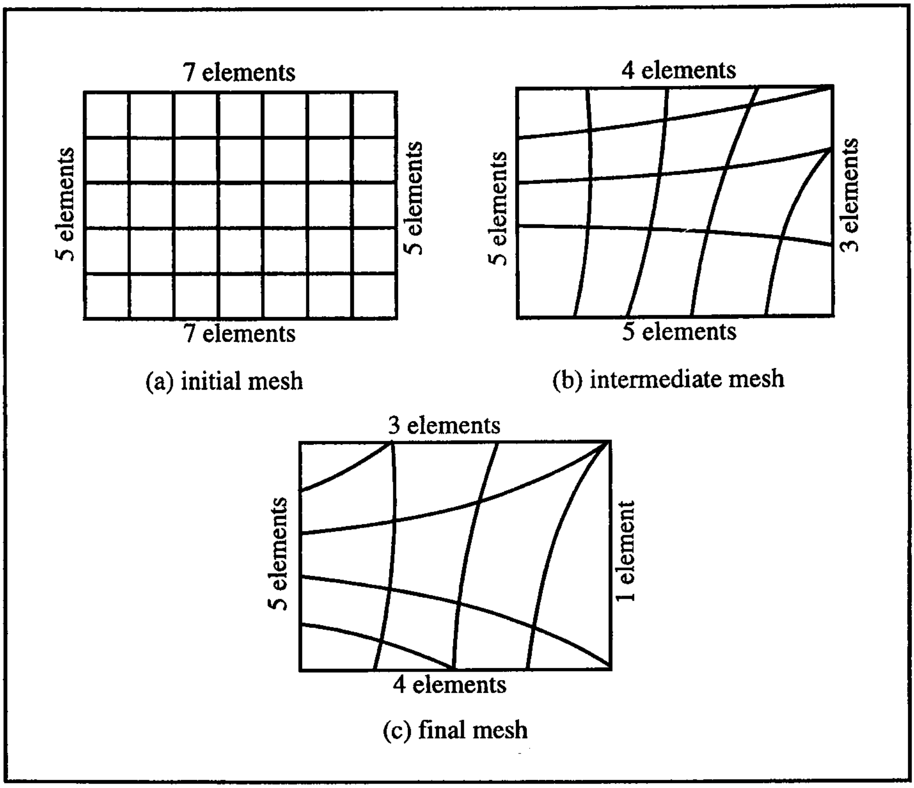


Figure 17. *Transitional mesh by sweeping function*

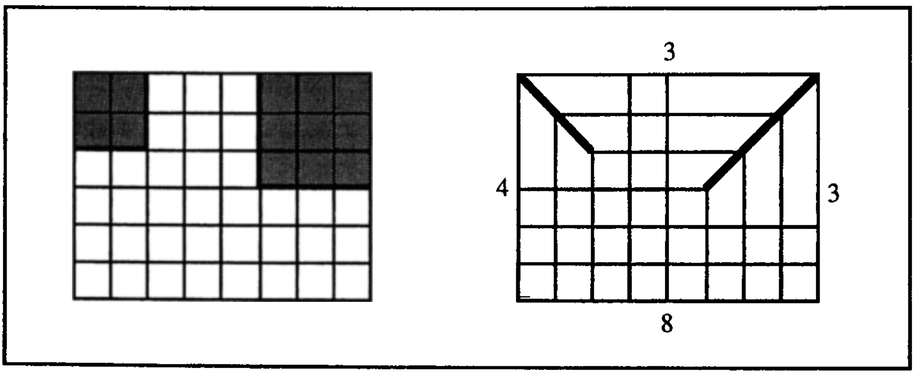


Figure 18. *Hoyte's technique*

Hoyte ([HOY 92]) has developpped a very simple technique to generate transitional meshes, in all-quadrilateral elements, on a topologically square super-element. The only restriction is the even sum of the boundary divisions. Quadratic segments of a grid are cut away from each corner ; the new sides are "glued" together to re-establish a four-sided polygon (see Figure 18). Parameters for the grid size and the quadratic segment size are calculated by solving equations defined from the user given number of elements for each side.

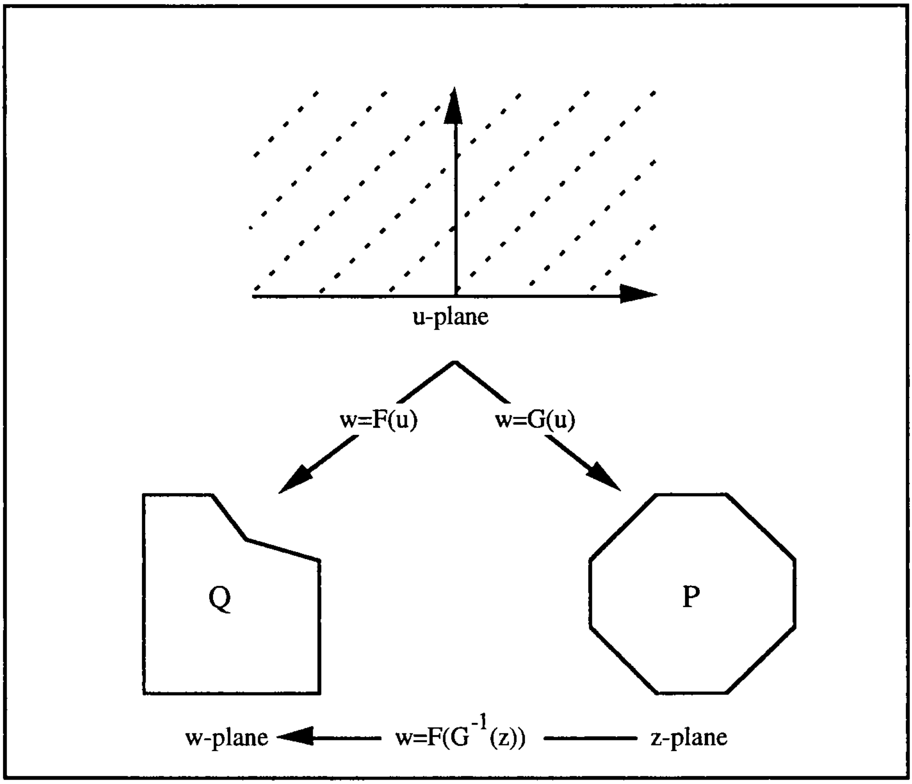
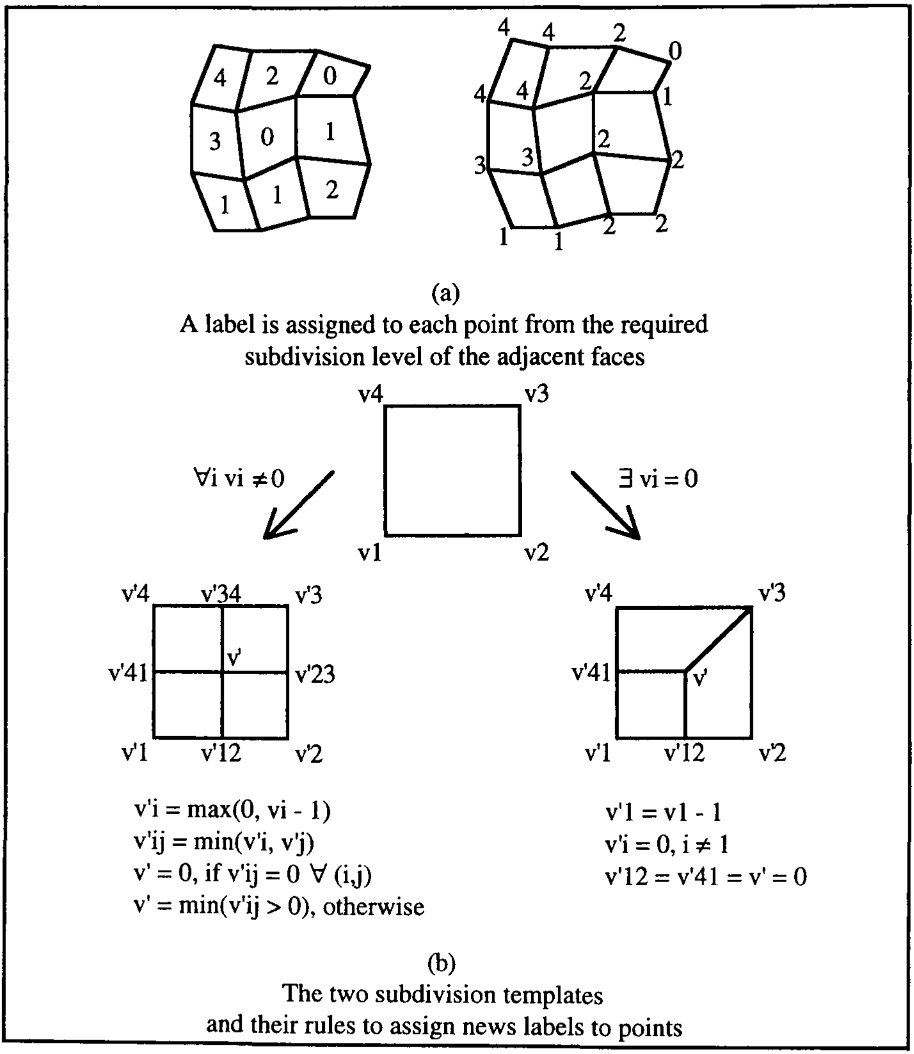


Figure 19. *Conformal mapping*

Hoyte has studied this method to mesh tridimensional blocks. Raws of elements are removed along the block edges. The author presents five operators, which are the configurations where gluing process is possible. Some of these operators may be used simultaneously. Two adjacents blocks, independently meshed with these operators, have the same discretization on the common edges, but not necessarily the same mesh topology on the common face. Blocks can not then be connected together. This reduces the interest of the 3D method.

Other techniques have been proposed to generate transitional meshes without any initial regular mesh. The conformal mapping is more general than the other mapping methods : polygons may have more than four sides. Brown ([BRO 81]) presented a special technique using the Schwarz-Christoffel transformation : a complex half-plane is associated with the interior of an arbitrary polygon. Two transformations are defined : F, which transforms the half-plane onto the region to mesh, and G, which transforms the half-plane onto a simple or master mesh polygon. The transformation parameters are calculated by numerical methods. The master mesh is mapped onto the region by applying FoG^{-1} (Figure 19). Baldwin and Schreyer ([BAL 85]) have improved the method for the quadrilateral generation and the density specification.



(a)
A label is assigned to each point from the required subdivision level of the adjacent faces

(b)
The two subdivision templates and their rules to assign news labels to points

Figure 20. *Subdivision process in Cheng's decomposition*

The technique proposed by Cheng ([CHE 89]) allows transitional mesh generation on a quadrilateral primitive network. The principle is similar to the recursive spatial decomposition. The user supplies a subdivision level to each primitive. A label is then assigned to each point (the maximal subdivision level of the faces sharing this point) to control the subdivision process (Figure 20(a)). Primitives are recursively divided using two subdivision templates : one creates four new subquadrants, the other creates only three. Figure 20(b) presents the two templates and the rules to assign new labels to points. The second template is applied when only one point of the quadrant has a non-null label. To ensure the conformity of the mesh, edges with the same point label assignment must always be divided in the same way. Edges with two null point labels may be divided or not. To avoid the division of such edges, quadrants with (0,0,i,j) assignments and all assignments obtained by permutation of these values must not occur. A preprocessing phase modifies the initial set of point label assignments to eliminate non-admissible configurations.

This technique allows parallel processings, while ensuring the conformity and the smoothness of the resultant mesh. The quality of the mesh will be acceptable with a regular quadrilateral network only : interior vertices would rather be shared by four regions, as the subdivision templates do not take into consideration the quadrant shape.

A lot of methods are now available to generate transitional quadrilateral meshes. Very few authors tackle the 3D problem. No general transition model is known for hexahedral meshing.

5. Conclusion

In this paper, automatic quadrilateral and hexahedral generation methods are reviewed. Very few techniques have been proposed for the three dimensional meshing. Most of the two dimensional methods seem to have a natural extension to the hexahedral element generation. But, in fact, moving on to three dimensional cases complicates the problems which yet appear in 2D and often induces new ones.

Methods based on the mesh conversion or on the node connection approach have not been tested on 3D domains because of the very high number of possible combinations.

Some geometrical problems appear with hexahedra. Whereas criteria for the triangular, quadrilateral and tetrahedral element generation only consist of angle and edge measurements, hexahedral elements have to be checked for warped faces. This additional criterion has to be considered in the generation and smoothing processes to limit the deviation of the element faces from planar faces.

The most important difficuty lies in the topological aspect of 3D meshing. The recursive decomposition methods have not been developped for hexahedral generation because of the impossible configurations in mesh topology. Some local operators are available to modify the topology of 2D meshes, but topological operations on 3D meshes are often difficult and induce propagations in the whole meshes.

Blaker has recently proposed a new hexahedral mesh generator, but conclusions can not be drawn because of the lack of information about experimentation on simple and complex geometries.

The methods that use a decomposition into primitives and primitive templates are the most frequently used. Primitives templates allows to control the local topology of the mesh, but templates available for 3D primitives do not allow any transition between low density and high density regions. Some solutions were proposed to generate transitional meshes on quadrilateral domain, but topological problems avoid a natural extension to hexahedral elements.

With Razdan's method, the decomposition into primitives becomes identified to feature recognition. But, his algorithm has still important limits. For many years, researchers have been studying a more "intelligent" decomposition. For exemple, Alagar suggests in [ALA 90] to use geometrical, topological and physical information stored in a semantic CSG tree. Holt ([HOL 86]) proposes the formation of a knowledge base containing theorical and practical information about finite element analysis. In both methods, adding semantic allows the systems to deduce some characteristics of the geometry (axis of symetry, prismaticity, ...), to choose the element type and size, to define the primitive processing order. So, mesh generators tend to become expert systems, which determine themselves the best meshing strategy.

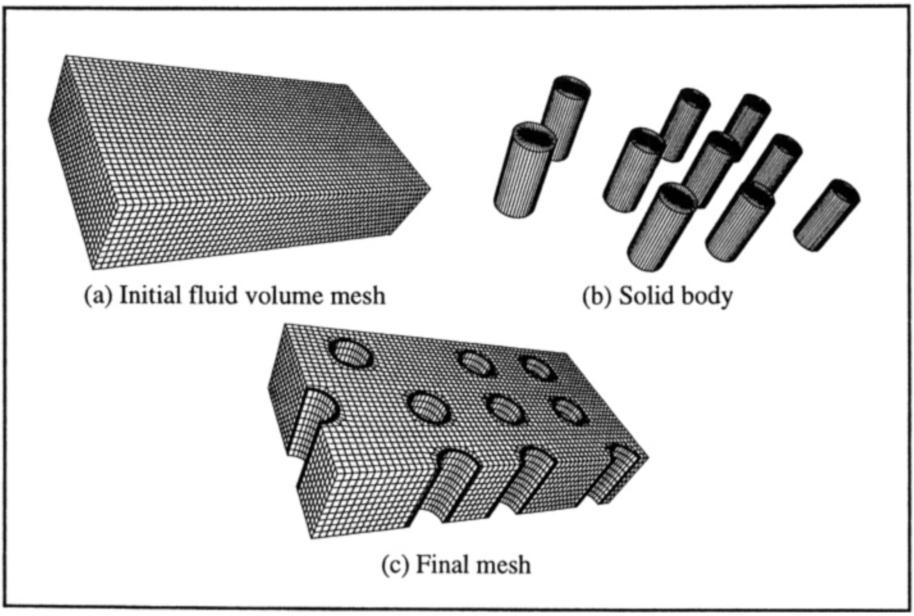


Figure 21. *PSI mesh tool using boolean difference*

Because of the lack of a general solution at the present time, industrialists still developp tools dedicated to specific applications. For example, Pam System International (PSI) proposes a tool, based on the boolean difference, to mesh a fluid volume intersected by a solid body like shown in Figure 21 ([PSI 90]).

References

[ALA 90] ALAGAR V.S., BUI T.D., PERIYASAMY K., "Semantic CSG trees for finite element analysis", *Computer-aided Design,* vol. 22, n° 4, p. 194-198 (1990).

[ANS 88] ANSALDI S., FALCIDIENO B., "The problem of form feature classification and recognition in CAD/CAM", *NATO ASI series,* vol. F40, *Theoretical foundations of computer graphics and CAD, Earnshaw R.A. ed., Springer-Verlag*, p. 899- 918 (1988).

[BAE 87] BAEHMANN P.L., WITTCHEN S.L., SHEPHARD M.S., GRICE K., YERRY M.A., "Robust, geometrically based, automatic 2-D mesh generation", *Int. J. Numer. Methods Eng.,* vol. 24, p. 1043-1078 (1987).

[BAL 85] BALDWIN K.H., SHREYER H.L., "Automatic generation of quadrilateral elements by a conformal mapping", *Eng. Comput.,* vol. 2, p. 187-194 (1985).

[BER 94] BERTIN D., CASTIES C., LORDON J., "A new automatic grid generation environment for CFD applications", *Proc. of the 4th Intern. Conf. on Numerical Grid Generation in Computational Fluid Dynamics and Related Fields, Swansy, Wales* (1994).

[BLA 88] BLAKER T.D., "Knowledge system approach to automated two-dimensional quadrilateral mesh generation", *Proc. Computers in Eng., San Francisco CA, ASME, New York,* p. 153-162 (1988).

[BLA 91] BLAKER T.D., "Paving : a new approach to automated quadrilateral mesh generation", *Int. J. Numer. Methods Eng.,* vol. 32, p. 811-847 (1991).

[BLA 93] BLAKER T.D., MEYERS R.J., " Seams and wedges in plastering : a 3D hexahedral mesh genration algorithm", *Eng. Comput.,* vol. 9, p. 83-93 (1993).

[BLU 73] BLUM H., "Biological shape and visual science", *J. theoretical biology 38,* p. 205-287 (1973).

[BRO 81] BROWN P.R., "A non-interactive method for the automatic generation of finite element meshes using the Schwarz-Christoffel transformation", *Computer Methods in Applied Mech. and Eng.,* vol. 25, p. 101-126 (1981).

[CHE 89] CHENG F., JAROMCZYK J.W., LIN J.R., CHANG S.S., LU J.Y., "A parallel mesh generation algorithm based on the vertex label assignment scheme", *Int. J. Numer. Methods Eng.,* vol. 28, p. 1429-1448 (1989).

[CHI 91] CHINNASWAMY C., AMADEI B., ILLANGASEKARE T.H., "A new method for finite element transitional mesh generation",*Int. J. Numer. Methods Eng.,* vol. 31, p. 1253-1270 (1991).

[COR 91] CORNEY J., CLARK D.E.R., "A feature recognition algorithm for multiply connected depressions and protusions in 21/2D objects", *Proc. Symp. on Solid Modeling Foundation and CAD/CAM Appl., Texas* (1991).

[FLO 89] DE FLORIANI L., "Feature extraction from boundary models of three dimensional objects", *IEEE Trans. on Pattern Analysis and Machine Intelligence,* vol. 11, n° 8, p. 785-798 (1989).

[GAR 91] GARCIA M., Génération automatique de maillages appliquée aux domaines complexes du sous-sol, Thèse, Techniques et économie de l'entreprise minière, Ecole Supérieure des Mines de Paris (1991).

[GAR 92] GARCIA M. and al., "Automatic grid generation for modeling reservoir heterogeneities", *SPE Reservoir Engineering,* p. 278-284 (1992).

[GUR 92] GÜRSOY H.N., PATRIKALAKIS N.M., "An automatic coarse and fine surface mesh generation scheme based on medial axis transform : part I Algorithms", *Eng. Comput.,* vol. 8, p. 121-137 (1992).

[HAB 81] HABER R., SHEPHARD M.S., ABEL J.F., GALLAGHER R.H., GREENBERG D.P., "A general 2-D, graphical finite element preprocessor utilizing discrete transfinite mappings", *Int. J. Numer. Methods Eng.,* vol. 17, p. 1015-1044 (1981).

[HAL 76] HALL C.A., "Transfinite interpolation and applications to engineering problems", *in Laws and Shaney (eds) Theory of approximation, Academic Press*, p. 308-331 (1976).

[HEI 83] HEIGHWAY E.A., "A mesh generator for automatically subdividing irregular polygons into quadrilaterals", *IEEE Trans. Magn.*, vol. MAG-19, n° 6, p. 2535-2538 (1983).

[HO 88] HO-LE K., "Finite element mesh generation methods : a review and classification", *Computer-aided design*, vol. 20, p. 27-38 (1988).

[HOL 86] HOLT R.H., NARAYANA L., "Adding intelligence to finite element modeling", *Expert Systems in Government Symp., Mc Lean VA, USA*, p. 326-336 (1986).

[HOY 92] HOYTE J., "The cut&glue mesh generation algorithm", *Eng. Comput.*, vol. 8, p. 51-58 (1992).

[JAI 86] JAIN A., "Modern methods for automatic finite element mesh generation", *The american society of civil engineers*, p. 19-28 (1986).

[JOH 91] JOHNSTON B.P., SULLIVAN J.M., KWASNIK A., "Automatic conversion of triangular finite element meshes to quadrilateral elements", *Int. J. Numer. Methods Eng.*, vol. 31, p. 67-84 (1991).

[LEE 83] LEE Y.T., Automatic finite element mesh generation based on constructive solid modelling, Thesis, Depart. Mech. Eng., University of Leeds (1983).

[LIU 90] LIU Y.C., EL MARAGHY H.A., ZHANG K.F., "An expert system for forming quadrilateral elements", *Eng. Comput.*, vol. 7, p. 249-257 (1990).

[PAV 77] PAVLIDIS T., *Structural pattern recognition*, New-York, Springler-Verlag (1977).

[PSI 90] PSI (Pam System International), PAM-GEN, Optimesh, User's manual Version 1.0 (1990).

[RAM 94] RAMSAY A., "Comparison of HEX and TET meshes", *Bench Mark, june 1994*, p. 11-13 (1994).

[RAN 92] RANOUX G., "Création rapide de modèles d'analyse dans un monde d'ingénierie simultanée", *Conf. SIA (Société des Ingénieurs de l'Automobile)* (1992).

[RAS 92] RASSINEUX A., Automatic mesh generator of planar domains, Rapport de recherche, CSI-Framasoft, (1992).

[RAZ 89] RAZDAN A., HENDERSON M.R., CHAVEZ P.F., ERICKSON P.A., "Feature based object decomposition for finite element meshing", *The visual Computer*, vol. 5, p. 291-303 (1989).

[SCH 92] SCHNEIDERS R., OBERSHELP W., KOPP R., BECKER M., "New and effective remeshing scheme for the simulation of metal forming processes", *Eng. Comput.*, vol. 8, p. 163-176 (1992).

[SEZ 91] SEZER L., ZEID I., "Automatic quadrilateral/triangular free-form mesh generation for planar regions", *Int. J. Numer. Methods Eng.*, vol. 32, p. 1441-1483 (1991).

[SHE 83] SHEPHARD M.S., YERRY M.A., "A modified quadtree approach to finite element mesh generation", *IEEE Comput. Graph. & Appl.*, p. 39-46 (1983).

[SLU 82] SLUITER M.L.C., HANSEN D.C., "A general purpose automatic mesh generator for shell and finite elements", *in Hulbert (ed) Comput. in Eng.*, vol. 3, *Book n°G00217, ASME*, p. 29-34 (1982).

[TAL 90] TALBERT J.A., PARKINSON A.R., "Development of an automatic, two-dimensional finite element mesh generator using quadrilateral elements and Bezier curve boundary definition", *Int. J. Numer. Methods Eng.*, vol. 29, p. 1551-1567 (1990).

[VIN 94] VINSON S., IRVING S., "Can I use an auto-tet mesh? A comparison of a hexahedral traditional mapped mesh against a series of automatically produced tetrahedral meshed models for a doubly curved geometry example", *Bench Mark, june 1994*, p. 15-19 (1994).

[WEL 88] WEELFORD L.C., GORMAN M.R., "A finite element transitional mesh generation procedure using sweeping functions", *Int. J. Numer. Methods Eng.*, vol. 26, p. 2623-2643 (1988).

[YEU 73] YEUNG S.F., HSU M.B., "A mesh generation method based on set theory", *Computers & Structures*, vol. 3, p. 1063-1077 (197).

[ZHA 92] ZHANG D., CHENG G., GU Y., "Automatic generation of quadrilateral mapping elements and applicability of shape optimisation", *Computers & Structures,* vol. 45, n° 4, p. 697-705 (1992).

[ZHU 91] ZHU J.Z., ZIENKIEWICZ O.C., HINTON E., WU J., "A new approach to the development of automatic quadrilateral mesh generation", *Int. J. Numer. Methods Eng.,* vol. 32, p. 849-866 (1991).

[ZIE 71] ZIENKIEWICZ O.C., PHILLIPS D.V., "An automatic mesh generation scheme for plane and curved surfaces by isoparametric co-ordinates", *Int. J. Numer. Methods Eng.,* vol. 3, p. 519-528 (1971).

Article reçu le 2 mars 1994.
Version révisée le 3 août 1994.