

HEX-LAYER: LAYERED ALL-HEX MESH GENERATION ON THIN SECTION SOLIDS VIA CHORDAL SURFACE TRANSFORMATION

W R Quadros and Kenji Shimada*

Carnegie Mellon University

ABSTRACT

This paper proposes chordal surface transform for representation and discretization of thin section solids, such as automobile bodies, plastic injection mold components and sheet metal parts. A multiple-layered all-hex mesh with a high aspect ratio is a typical requirement for mold flow simulation of thin section objects. The chordal surface transform reduces the problem of 3D hex meshing to 2D quad meshing on the chordal surface. The chordal surface is generated by cutting a tet mesh of the input CAD model at its mid plane. Radius function and curvature of the chordal surface are used to provide sizing function for quad meshing. Two-way mapping between the chordal surface and the boundary is used to sweep the quad elements from the chordal surface onto the boundary, resulting in a layered all-hex mesh. The algorithm has been tested on industrial models, whose chordal surface is 2-manifold. The graphical results of the chordal surface and the multiple-layered all-hex mesh are presented along with the quality measures. The results show geometrically adaptive high aspect ratio all-hex mesh, whose average scaled Jacobean, is close to 1.0.

Keywords: quadrilateral meshing, chordal axis transform, medial surface, and anisotropic hexahedral mesh generation.

1. INTRODUCTION

This paper proposes a new method for the representation and discretization of thin section objects such as automobile bodies, plastic injection mold components, sheet metal parts, paint coating and rubber sheets, into an assemblage of hexahedral finite elements. Thin section solids have smaller wall thickness and relatively large surface area, as shown in Figure 1. In other words, the ratio of area to perimeter of the cross section is small. Finite element meshes are used in the Finite Element Method (FEM), a versatile and powerful numerical procedure to analyze complex structures and continua in many scientific and engineering fields. The accuracy of the results obtained by FEM greatly depends on the quality of the finite element mesh; so there is a great demand for the generation of high quality hex mesh with fewer elements, so that accurate results can be obtained in less time. The Hex-Layer aims at solving this problem for specific type of geometries, called thin section solids.

Thin section objects can be best represented by Medial Axis Transform (MAT) [2], which is the locus of the center of the maximal sphere as it rolls inside a solid, along with the associated radius function. Like mid surface abstraction, which has the required property of MAT, here an approximation of the mid surface (called the “chordal surface” [9]), is proposed as the skeleton of 3D objects. The skeleton model has numerous applications in such diverse areas as computer vision, robot path planning, evaluation of moulds and dies, feature recognition, medical diagnostics and mesh generation [10, 22].

The Hex-Layer uses the chordal surface to reduce the problem of 3D hex meshing to 2D quad meshing on the chordal surface. Quad elements are then swept using two-way mapping between the chordal surface and the boundary of the model to generate the layered all-hex mesh. The results of the implementation of the chordal surface, and the hex mesh generation of objects whose chordal surface is 2-manifold, is presented. The extension of the Hex-Layer to deal with general solids is addressed in the conclusion.

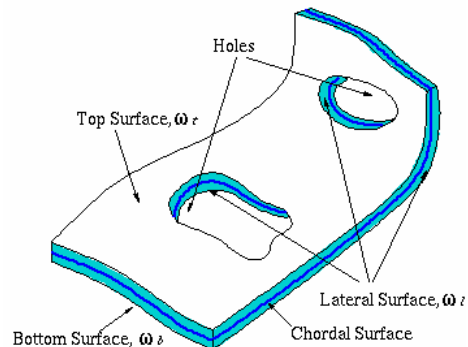


Figure 1. Thin section object with holes

2. LITERATURE REVIEW

From the literature, hex meshing can be broadly classified into the direct method and the indirect method [3]. Direct methods generate a hex mesh directly from a solid. Indirect methods first subdivide a solid into a tet mesh; the tet mesh is then converted to a hex mesh. One approach is to subdivide

* Correspondence to: Kenji Shimada

The Department of Mechanical Engineering, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213, Email: Shimada@cmu.edu

each tet into four hexahedral elements [19]. Even though this approach generates all-hex mesh, the quality will be poor. Recently, Hexhoop [18] was proposed to solve the problem of converting a hex-dominant mesh to an all-hex mesh, using templates. However, combining multiple tets to form a hex has proved to be a difficult task. In the indirect methods the quality of the final mesh greatly depends on the input tet mesh.

Some of the major algorithms belonging to the direct method are, the advancing front methods [1, 20], mapped meshing methods, grid based approaches [14, 13], skeleton based methods, and optimal node placement and connectivity methods [4, 15]. Only the algorithms that are closely related with the Hex-Layer are discussed here.

Sweeping is a type of mapped meshing method applicable specifically to geometries whose end faces contain topologically equivalent quad mesh. Quad mesh is then swept along a direction specified by a curve. Layers of hexahedral elements are formed at regular intervals with the same topology as that of a quadrilateral mesh. This technique can be generalized to mesh certain classes of volumes by defining so-called ‘multiple source’ and ‘multiple target surfaces’, and using Boolean operations [7, 6]. An all-hex meshing algorithm, called ‘the Graft Tool’ [11] has been proposed for multi-directional swept volumes. Gambit uses a multi-axis cooper algorithm [8], which provides a toolbox for identification of sub domains with a definite axis, and it is meshed using a cooper tool. Thin section solids, such as automobile bodies and injection mould components, are usually not composed of sub domains with definite axis, and identifying multiple source and multiple target surfaces may be difficult.

The medial surface, along with the radius function, is called the Medial Axis Transform (MAT). It is a mathematically well defined 2D representation of a 3D solid. Medial surface methods [10] decompose the complex geometry into simpler sub domains; the sub domains are classified based on the topology. Templates are then used to mesh the sub domains. Linear programming is used to ensure conformity between the meshes of the sub domains. Recently MAT was used to subdivide a complex object into only one type of sub-domain, called “tracks” [23], in 2D domain and rectangular columns [12] in 3D space. In Sampl’s method, the medial surface obtained is first meshed into a quad-dominant mesh, containing triangles and quadrilaterals, using an advancing front based algorithm (Paving). The mesh on the medial surface is then extruded on both sides of the medial surface until it intersects the boundary of the object to obtain 3D mesh. The output mesh contains non-hex elements at the concave edges and vertices. Although the medial surface is mathematically well defined, its generation is computationally very expensive and contains extra surfaces which may not be of use for engineering applications. The quality of the mesh at the boundary of the thin section objects can be improved by using the chordal or mid surface.

The Hex-Layer uses a chordal surface, which is generated by cutting a tet mesh of the solid at its mid-plane; therefore it can be classified under indirect methods. But, the quality of the final mesh is independent of the tet mesh. Because the medial surface is the region of natural termination of the

opposite fronts of the advancing front method, it is implied that the Hex-Layer avoids expensive interference checks at every layer, with a one time cost of generating the chordal surface.

3. PROBLEM DEFINITION

In this paper, a thin section solid whose chordal surface is 2-manifold is considered. Many industrial models, made from an injection molding process and sheet metal works, fall in this category. Figure 1 shows a thin section solid with top, bottom, and lateral surface. Note that Hex-Layer does not require the explicit identification of top, bottom, and lateral surface as user input.

Consider a solid $\Omega \subset \mathcal{R}^3$, in which the boundary $\partial\Omega$ consists of a top surface ω_t , bottom surface ω_b , and lateral surface ω_l . Ω is said to be a thin section solid if $A(\omega_t), A(\omega_b) \gg A(\omega_l)$, where $A(\omega_i)$ is the area of ω_i . The chordal surface of Ω is denoted by $C(\Omega)$ and is 2-manifold, if for every point $\mathbf{x} \in C(\Omega)$, the neighborhood of \mathbf{x} denoted by $N(\mathbf{x})$ is topologically equivalent to a 2-disk $D^2 = \{\mathbf{y} \in \mathcal{R}^2 \mid \|\mathbf{y} - \mathbf{y}_0\| < 1\}$.

Given a solid Ω whose $C(\Omega)$ is 2-manifold, generate:

1. Chordal surface $C(\Omega)$ and the mapping $\pi_t : C(\Omega) \rightarrow \omega_t$ and $\pi_b : C(\Omega) \rightarrow \omega_b$
2. Uniform isotropic single/multi-layered all-hex mesh
3. Uniform anisotropic single/multi-layered all-hex mesh
4. Geometry adaptive isotropic single/multi-layered all-hex mesh
5. Geometry adaptive anisotropic single/multi-layered all-hex mesh

4. OVERVIEW OF PROPOSED METHOD

The main purpose of the Hex-Layer is to reduce the 3D hex meshing problem to a 2D quad meshing, followed by sweeping the quad elements. This is accomplished by generating a chordal surface, which is proposed as a skeleton for representing thin section objects, and by using two-way mapping to sweep the quad mesh lying on the chordal surface onto the top and the bottom surface. Figure 2 shows the steps involved in generating multi-layered hex mesh of the input CAD model.

The tet mesh of the CAD model is generated using a bubble packing algorithm. The bubble packing algorithm automatically places the nodes only on the top and bottom surface, without any interior nodes or nodes on the lateral surface. This is accomplished by setting the sphere radius more than half the maximum wall thickness. This bypasses the relatively difficult task of identifying the top and the bottom surface of the CAD model. The Delaunay method is used to connect the nodes, to generate a good quality tet mesh [16], as shown in Figure 2(b). Tet element (Figure 3), which has at least one node on both the top and the bottom surface

of the solid, is considered to be a valid tet for our algorithm; otherwise it is called the “flat” element. Unlike previous indirect methods, tet mesh is not directly used to generate hex mesh, rather it is used to generate the chordal surface. Therefore the final hex mesh quality does not greatly depend on the tet mesh generated.

The advantage of using the chordal surface rather than the medial surface is that the generation of the chordal surface is computationally less expensive. The chordal surface is generated by cutting the tet elements at its mid section, as shown in Figure 2(c), and is discussed in detail in Section 5. Once the chordal surface becomes available, the 3D hex meshing reduces to 2D quad meshing [5] on the chordal surface. Either the uniform or the geometry adaptive quad mesh (based on the radius function and the curvature of the chordal surface) is generated, as shown in Figure 2(d), and is briefly discussed in Section 6. The quality and size of the quad mesh determines the final quality and size of the hex elements. Note that if the quad mesh is generated on the top or the bottom surface, rather than on the chordal surface, the quality of the hex mesh after sweeping will be poor near the bottom or the top surface respectively.

The final step is to lay the desired number of hex layers (Figure 2(e)) by locally sweeping every quad element on both sides of the chordal surface. Sweeping of the quad element is accomplished by placing hex nodes along the normal direction at every quad node and connecting them in order, as explained in Section 7

The major contributions of this paper are the generation of a chordal surface and layered hex meshing by local sweeping of 2D quad elements lying on the chordal surface. Section 8 shows the results obtained by using the Hex-Layer on two industrial objects whose chordal surface is 2-manifold. The extension of the Hex-Layer to handle objects with a non-manifold chordal surface is highlighted in the conclusion.

5. CHORDAL SURFACE GENERATION

Prasad L. [9, 17] first proposed chordal axis transform (CAT) for the morphological analysis of 2D planar shapes. Like the medial surface, extraction of the chordal surface of 3D objects would be of importance in many diverse areas such as computer vision, robot path planning, feature recognition and medical diagnostics. This paper proposes the chordal surface for all-hex meshing of thin section solids.

This section provides the mathematical definition based on graph theory, and its geometric construction from input tet mesh. Let the top surface of the thin walled solid be represented by a simple graph, G , consisting of a nonempty set $V(G)$ of vertices and a set $E(G)$ of edges. Similarly, the bottom surface is represented by a simple graph, H , consisting of a vertex set $V(H)$ and an edge set $E(H)$. And let E_I be the set of edges of the tet mesh so that $\forall e(u, v) \in E_I, u \in V(G)$ and $v \in V(H)$.

Definition: The chordal surface, a composition of simple graphs G and H , is a simple graph represented as $G[H]$, with vertex set $V(G) \times V(H)$, in which vertex (u, v) is adjacent to vertex (u', v') if and only if either: (1) $uu' \in E(G)$ and $v = v'$

or (2) $u = u'$ and $vv' \in E(H)$, where, $e(u, v), e(u', v') \in E_I$.

Figure 3 shows all possible types of tets in a tet mesh of a solid, whose chordal surface is 2-manifold. Case 4-0 and Case 0-4 tets are considered to be topologically invalid for chordal surface generation, as all the four vertices lie on the same surface.

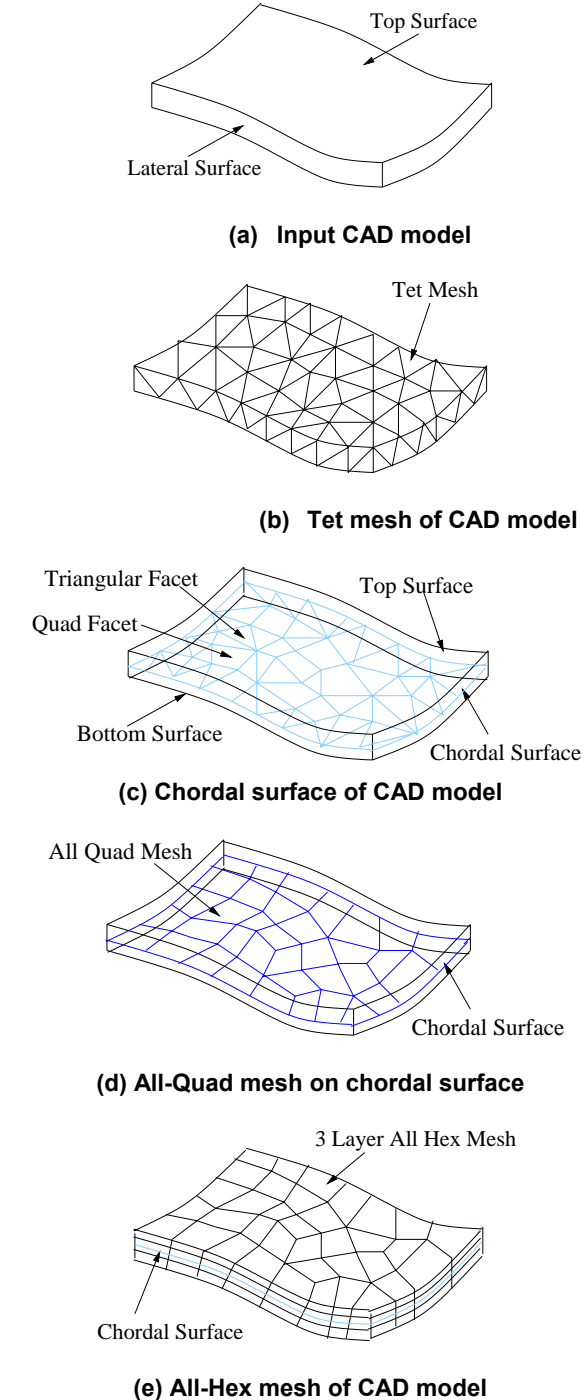


Figure 2. Overview of the Hex-Layer algorithm

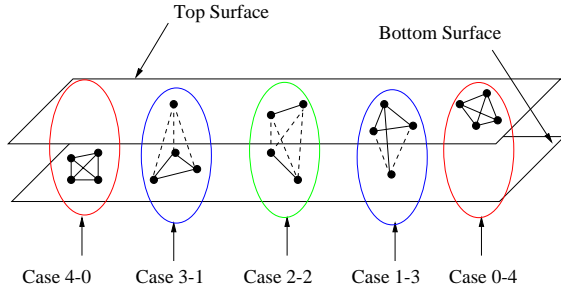


Figure 3. Possible types of tet elements

But geometrically, such flat tets may exist in a high curvature region and are removed during the processing of the tets, which is discussed in Section 5.4. Cases 3-1 and 1-3 are similar and contain three internal edges connecting the top and bottom surfaces. Cutting internal edges results in a triangular chordal surface facet. Case 2-2 contains four internal edges, which form the quadrilateral chordal surface facet. Therefore the chordal surface consists of only triangular and quadrilateral facets.

5.1. Finding Interior Triangles and Adjacency List of Tet Mesh

A triangular face of any tet element is said to be interior if it is shared by two tet elements. Each of the four triangles of every tet is tested by checking the three vertices of the triangle with four vertices of every other tet. If the three vertices of a triangle are present in another tet, it is an interior triangle, and the adjacency list of both the tets incident on the interior triangle is updated. This intermediate adjacency list reduces the computation cost. If only one or none of the vertices of the triangle are present in another tet, then that tet is not checked for three other triangles. The adjacency list establishes connectivity between the tets and is used in the breadth first traversal of tet mesh.

5.2. Finding External Edges and Marking Boundary Edges

Once the interior triangles are known, the rest of the triangles of the tet mesh form a set of connected exterior triangles. The topological requirement for the boundary of any valid solid is that the outer surface mesh should be 2-manifold, so every external edge should be shared by two external triangles. Boundary edges are those external edges whose dihedral angle is less than the predefined threshold value, which in our case is set to 125 degrees.

5.3. Classification of Tets Based on Number of External Triangles and External Edges.

The number of exterior triangles and exterior edges of a tet is used as the criteria to classify the tets and is tabulated in Table 1. This classification determines the order of processing the tets while generating the chordal surface, which is explained in Section 5.4. In Table 1, ‘X’ indicates

that the tet cannot exist; ‘Y’ indicates that the tet can exist; and ‘D’ indicates a degenerate case. The Existence of degenerate tets is possible when either the top or the bottom surface becomes a line, which is not discussed in this paper.

The classification shown in Table 1 is valid for a tet mesh of a thin section solid, whose chordal surface is 2-manifold, the tet mesh contains nodes only on the top or the bottom surface and does not contain flat tets. A tet with four exterior triangles forms an isolated tet and cannot exist with any combination of external edges. A tet with three exterior triangles exists at a corner of the solid, as shown in Figure 5(a). A tet with two exterior triangles contains at least five external edges, as shown in Figures 4(a) and 5(b). A tet with one exterior triangle will have a minimum of three external edges. Figure 4 (a) shows the tet with three external edges and Figure 4(b) shows a tet with four and five external edges. A tet which is totally covered by four adjacent tets has zero exterior triangles and should contain at least two external edges (Figure 3). As the internal edges (Figure 4(c)), are exposed to the external surface, the number of external edges increases up to six, as shown in Figure 5(c).

Table 1 Possible types of simple and complex tets

External tri/edges	0	1	2	3	4	5	6
0	X	X	Y	Y	Y	Y	Y
1	X	X	X	Y	Y	Y	D
2	X	X	X	X	X	Y	D
3	X	X	X	X	X	X	Y
4	X	X	X	X	X	X	X

5.4. Processing Simple Tets and Incremental Processing of Complex Tets

Chordal surface generation is accomplished in two steps: (1) processing simple tets and (2) incremental processing of complex tets. The tets which exist at the interior of the domain are cut without ambiguity to form a major part of the chordal surface. These tets are called simple tets (Figure 4). If it is not possible to decide the cutting direction with the known information, such as the exterior triangle, exterior edges and boundary information, then those tets are called “complex tets” (Figure 5), and their processing/cutting is postponed. Complex tets usually exist at the boundary, joints and other intricate parts. The complex tets adjacent to the available section of the chordal surface are processed first. This incremental approach of processing the complex tets is performed over the iterations till all the complex tets are processed. The iteration stops, either when all the complex tets are processed, or if some undesirable tets exist (which does not fall in the classification given in Table 1). The existence of such undesirable tets is an indication of the presence of the invalid flat tets (Case 0-4 and 4-0 in Figure 3 and Figure 4). Flat tets cover internal tets, thus giving rise to undesirable tets; this is overcome by removing the flat tets. The adjacent tets are updated by labeling the internal shared

triangle and edges as exterior. This processing of simple and complex tets is repeated for all the tets.

Figure 4 and Figure 5 shows all the simple and complex tets with their chordal surface facets. Solid lines are external edges and dashed lines are internal edges (not hidden lines). The cycle formed by three solid lines, by default indicates the external triangle of a tet element. Chordal surface facets are indicated by shaded polygon. In Figures 4 and 5 the chordal vertex v of an edge with end vertices v_t and v_b lying on the top and the bottom surface respectively is given by $v = (v_t + v_b)/2$. A brief description of cutting the simple and the complex tets is given below.

Simple tets:

Following are six simple tets, those can be cut without ambiguity and which do not require the chordal surface information of the adjacent tets.

TWO_EXT_TRI_FIVE_EXT_EDGE: This tet (Figure 4(a)) exists at the boundary and contains three out of four vertices on the boundary. A chordal surface facet is obtained by cutting the edges which are incident on a vertex which lies on the boundary, but not on the common edge shared by the two external triangles. If all the four vertices are lying on the boundary as shown in Figure 5(b), then it is difficult to determine the cutting direction, and it is treated as a complex tet.

ONE_EXT_TRI_THREE_EXT_EDGE: This tet (Figure 4(a)) exists away from the boundary. Cutting the internal edges gives a triangular chordal surface facet.

ONE_EXT_TRI_FOUR_EXT_EDGE: Figure 4(b) shows that if all the three vertices of the exterior triangle lie on the boundary, and the fourth vertex does not incident on the boundary, then the chordal surface facet is quadrilateral. If the fourth vertex is also on the boundary, then the tet is treated as a complex tet as shown in Figure 5(b). If all the three vertices of the exterior triangle do not lie on the boundary, then the edges incident on the fourth vertex are cut to obtain the triangular chordal surface facet, as shown in the Figure 4(b).

ONE_EXT_TRI_FIVE_EXT_EDGE: This tet exists at the corner as shown in Figure 4(b). Note that there is a cycle of three external edges forming an interior triangle. The three edges incident on a vertex that is not incident on the external triangle, are cut to form a triangular chordal surface facet.

ZERO_EXT_TRI_TWO_EXT_EDGE: This tet is shown in the left side of Figure 4(c). Cutting the internal edges results in a quadrilateral chordal surface facet.

ZERO_EXT_TRI_THREE_EXT_EDGE: This tet (Figure 4(c)) also can be cut without ambiguity to form a quadrilateral chordal surface facet. If more than three edges are external, as shown in Figure 5(c) and 5(d), they are treated as complex tets.

Complex tets:

The complex tets listed below are processed in a particular

order. The tets which touch the chordal surface are processed before the tets which are far away from the chordal surface.

THREE_EXT_TRI_SIX_EXT_EDGE: These tets exist at the corner, as shown in the Figure 5(a). When any two edges are touched by the chordal surface, then the tet can be processed.

TWO_EXT_TRI_FIVE_EXT_EDGE and **ONE_EXT_TRI_FOUR_EXT_EDGE:** The external edge incident on the chordal surface determines the cutting direction (Figure 5(b)).

ZERO_EXT_TRI_FOUR_EXT_EDGE and **ZERO_EXT_TRI_FIVE_EXT_EDGE** and **ZERO_EXT_TRI_SIX_EXT_EDGE:** Figure 5(c) and 5(d) shows tets with no external triangle. These tets are processed using existing chordal surface information.

Invalid flat tets:

TWO_EXT_TRI_FIVE_EXT_EDGE_FLAT: This is a flat tet that contains no vertex on the boundary except the ones incident on the common external edge (Case 4-0 and 0-4 in Figure 3 and Figure 4(d)).

5.5. Find Average Normal and Radius at Every Chordal Vertex

After the generation of chordal surface, the radius function at every chordal vertex is calculated to establish the two-way mapping between the chordal surface and the boundary. As the chordal surface facets are computed locally by processing every tet element, the facets' normal vector is adjusted by reversing the vertex list to maintain consistent orientation.

The normal vector of every facet is calculated before the average unit normal vector at the chordal vertex is calculated. As explained in Section 3, the chordal surface facet consists of either a triangular or a quadrilateral facet.

Let v_j : j^{th} vertex of i^{th} chordal facet

n_i : Unit normal vector of i^{th} chordal facet

Normal vector of triangular facet: Because the chordal vertices are always coplanar, the cross product of any two adjacent edges gives the direction of the normal.

$$e_1 = v_1 - v_2$$

$$e_2 = v_3 - v_2$$

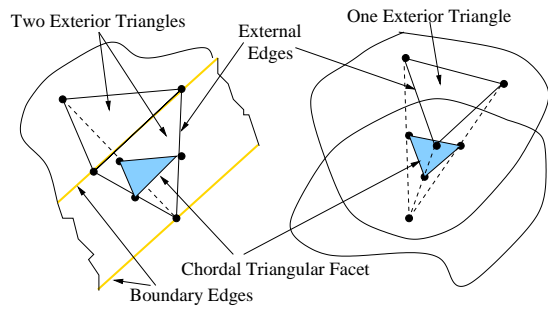
where e_1 and e_2 are edges of triangular facet.

Normal vector of quadrilateral facet: The chordal vertices of a quadrilateral chordal surface facet may not be coplanar and the normal vector is calculated as below

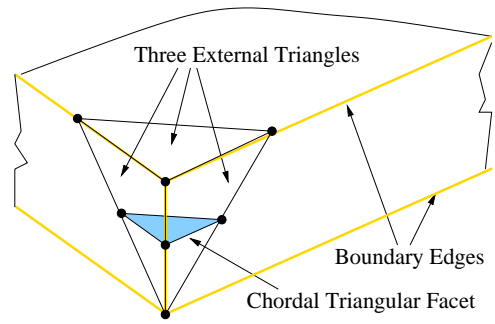
$$e_1 = v_4 - v_2$$

$$e_2 = v_3 - v_1$$

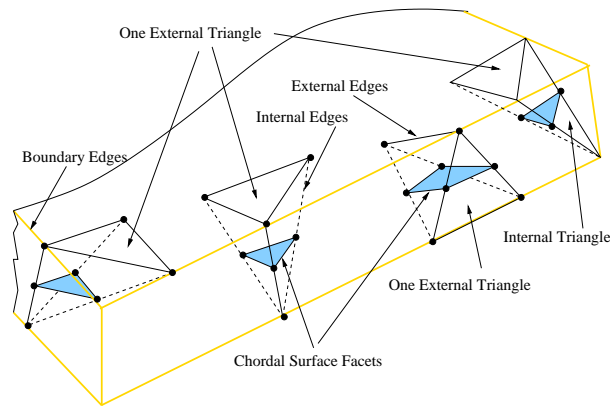
where e_1 and e_2 are diagonals of quadrilateral facet. The normal vector of both triangular and quadrilateral facets is computed as given in Equation 1:



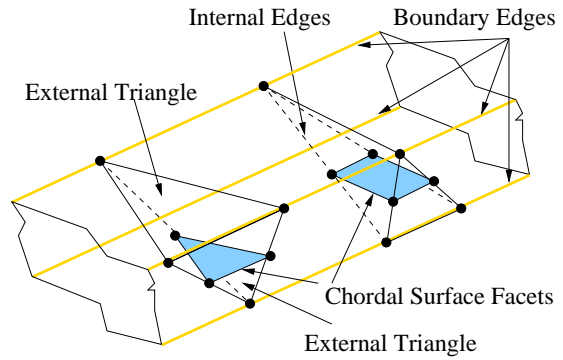
(a) TWO_EXT_TRI_FIVE_EXT_EDGE and ONE_EXT_TRI_THREE_EXT_EDGE



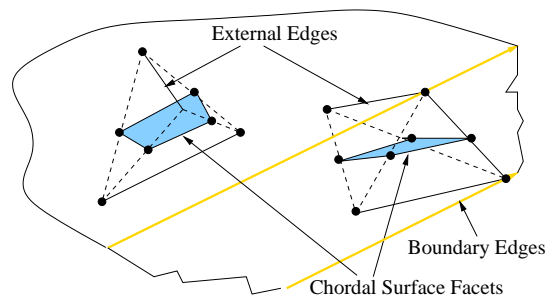
(a) THREE_EXT_TRI_SIX_EXT_EDGE



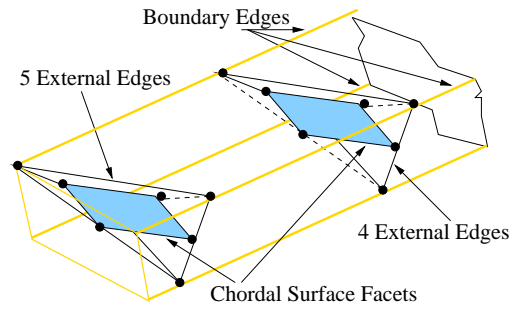
(b) ONE_EXT_TRI_FOUR_EXT_EDGE and ONE_EXT_TRI_FIVE_EXT_EDGE



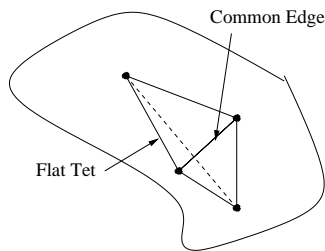
(b) TWO_EXT_TRI_FIVE_EXT_EDGE and ONE_EXT_TRI_FOUR_EXT_EDGE



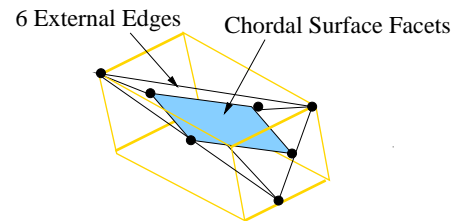
(c) ZERO_EXT_TRI_TWO_EXT_EDGE and ZERO_EXT_TRI_THREE_EXT_EDGE



(c) ZERO_EXT_TRI_FIVE_EXT_EDGE and ZERO_EXT_TRI_FOUR_EXT_EDGE



(d) TWO_EXT_TRI_FIVE_EXT_EDGE_FLAT



(d) ZERO_EXT_TRI_SIX_EXT_EDGE

Figure 4. Simple and flat tets

Figure 5. Complex tets

$$\mathbf{n}_i = \frac{\mathbf{e}_2 \times \mathbf{e}_1}{\|\mathbf{e}_2 \times \mathbf{e}_1\|} \quad (1)$$

The unit average normal vector ' $\bar{\mathbf{n}}$ ' at a vertex ' \mathbf{v} ', which is incident on ' k ' chordal surface facets (Figure 6), is given by Equation 2:

$$\bar{\mathbf{N}} = \frac{\sum_{i=1}^k \mathbf{n}_i}{k}$$

$$\bar{\mathbf{n}} = \frac{\bar{\mathbf{N}}}{\|\bar{\mathbf{N}}\|} \quad (2)$$

At every chordal surface vertex, only one common interior edge passes through the vertex, and there can be many tets incident on the common edge. From Section 5.4 we know that the chordal vertex is the mid point of the common edge. The radius / depth ' r ' at the vertex ' \mathbf{v} ', is computed as shown in Figure 6 and is given by Equation 3.

$$r = \bar{\mathbf{n}} \cdot \mathbf{l} \quad (3)$$

where $\mathbf{l} = \mathbf{v}_t - \mathbf{v}$ which is half the length of the common edge.

After computing the discrete radius/depth function at every chordal vertex, the continuous radius/depth function over the entire chordal surface can be obtained by interpolating over the domain, as discussed in Section 7.1. Note that the radius function of MAT is perpendicular to the boundary, and the current unit average normal vector computed is perpendicular to the chordal surface. For general solids, sweeping along the radius direction of MAT is recommended, as it gives almost orthogonal hex elements at the boundary [21]. The current approach is best suited for thin section solids, as it gives almost the same quality mesh in all the layers, which is discussed in Section 7.3.

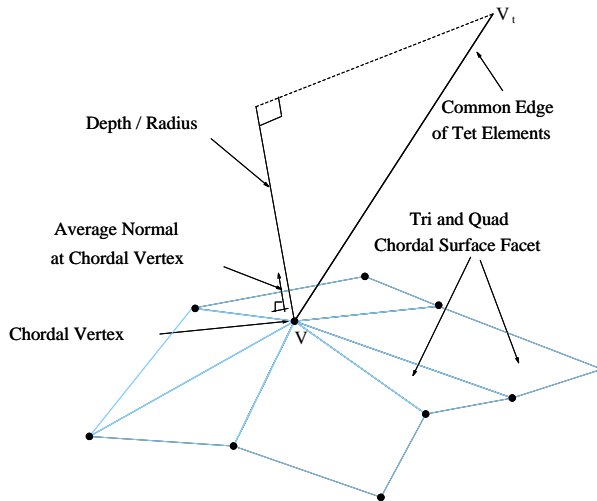


Figure 6. Normal and radius/depth at a chordal vertex

6. QUAD MESHING ON A CHORDAL SURFACE

The chordal surface reduces the 3D problem of hex meshing to 2D quad meshing. The size and quality of the final hexahedral mesh is controlled by the quad elements on the chordal surface. Either isotropic quad elements can be generated, or the radius function and curvature of the chordal surface can be used to adapt the size of the quad elements. The bubble packing algorithm is used here [5] to generate high quality quad-dominant mesh on multiple connected surfaces in 3D space, as shown in Figures 12(d) and 13(d). Templates are used to convert the quad-dominant mesh to all-quad mesh.

6.1. Sizing Function for Quad Meshing

For geometry adaptive quad meshing (Figure 12(d)), the sizing function is set directly proportional to the radius function (Figure 7). This would result in geometry adaptive hex meshing, as shown in Figure 12(e).

The curvature of the chordal surface is another parameter which decides the size of quad elements. Fine elements are obtained in the high curvature region, as shown in Figure 12(e).

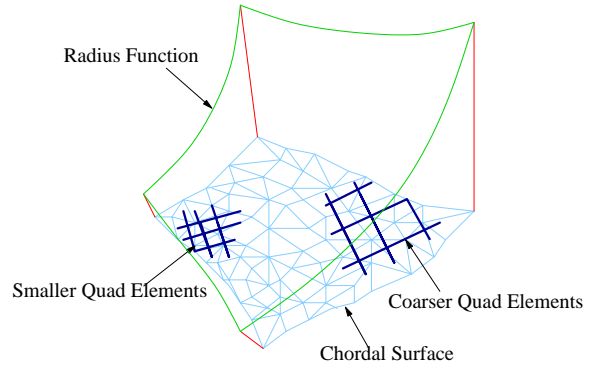


Figure 7. Sizing based on radius function

6.2. Converting Quad Dominant Mesh to All Quad Mesh using Templates

As the quad dominant mesh contains only triangles and quadrilaterals, simple templates are used to convert the quad dominant mesh to an all quad mesh. A quad element can be converted into four quad elements by inserting a node at the centroid of the quad element and connecting it with the mid point of each of the edges of the quad element. A similar process converts the triangle into three triangular elements. The initial quad dominant mesh is made coarser as the number of elements in the all quad mesh becomes larger, after the use of templates. Note that the templates can be avoided by using a suitable all-quad meshing algorithm.

7. ALL-HEX MESHING USING QUAD MESH

The sweeping of quad elements to result in layered hex mesh, is further reduced to sweeping quad nodes to place the hex nodes and to establish connectivity between the hex nodes. The coordinates of quad nodes lying on the chordal surface gives the source position vector for sweeping. The sweeping direction and magnitude at the quad node are interpolated from the average unit normal vector and radius values at the chordal vertices. The number of hex nodes placed by sweeping the quad node depends on the number of hex layers needed. Hex nodes are then connected by using the connectivity of quad nodes at the quad elements. The following section gives the details of hex meshing using quad mesh.

7.1. Interpolate Normal and Radius at a Quad Node

A quad node can lie on a vertex, on an edge, or on a facet of the chordal surface. If a quad node lies on a chordal vertex, then the unit average normal computed using Equation 2 and radius computed using Equation 3 at the chordal vertex determines the sweeping direction and magnitude. If a quad node lies on an edge of the chordal surface, then the normal and radius is calculated by linearly interpolating the unit average normal and the radius at the end chordal vertices, as given below.

Let \mathbf{v}_i and \mathbf{v}_j be the end vertices of an edge of the chordal surface on which a quad node \mathbf{q} lies. Let $\bar{\mathbf{n}}_i$, $\bar{\mathbf{n}}_j$ be the average unit normal and r_i, r_j be the depth at \mathbf{v}_i and \mathbf{v}_j respectively.

The sweeping direction $\bar{\mathbf{n}}$ and magnitude r at \mathbf{q} are given by Equations 4 and 5.

$$\bar{\mathbf{n}} = \bar{\mathbf{n}}_i(1-u) + \bar{\mathbf{n}}_j \quad (4)$$

$$r = r_i(1-u) + r_j u \quad (5)$$

where parameter $u = \frac{\mathbf{q} - \mathbf{v}_i}{\mathbf{v}_j - \mathbf{v}_i}$.

If a quad node lies inside a facet of the chordal surface, then the normal and the radius at the quad node are interpolated using the chordal vertices of the facet.

Let A be the area of the triangular facet. Quad node \mathbf{q} partitions the triangle into three sub-triangles whose areas are A_1, A_2 and A_3 , as shown in Figure 8. The normal $\bar{\mathbf{n}}$ and depth ' r ' at \mathbf{q} are given by Equations 6 and 7.

$$r = \sum_{i=1}^{i=3} w_i r_i \quad (6)$$

$$\bar{\mathbf{n}} = \sum_{i=1}^{i=3} w_i \bar{\mathbf{n}}_i \quad (7)$$

where weight $w_i = \frac{A_i}{A}$

When a quad node lies on a quadrilateral facet of the chordal surface, then inverse bilinear interpolation is used to find the normal and the depth at the quad node.

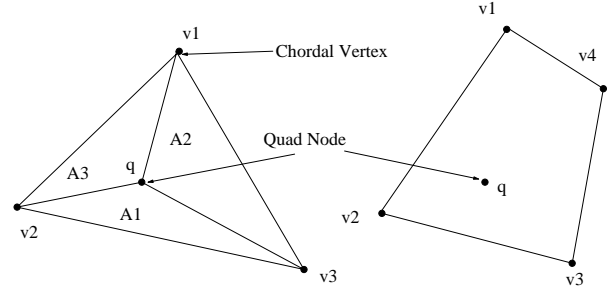


Figure 8. Interpolation of depth and normal at a quad node lying on a chordal surface facet

7.2. Placing Hex Nodes along the Normal Direction at a Quad Node

After finding the sweeping direction and magnitude at every quad node, the quad nodes are swept to place hex nodes at regular intervals, based on the number of hex layers needed. When the number of layers is even, a hex node coincides with the quad node present on the chordal surface, if the number of layers is odd, no agreement occurs. Figure 9 shows the placement of hex nodes for both even and odd numbers of hex layers ' m '.

For even numbers of hex layers, hex node \mathbf{h}_i is given by

$$\mathbf{h}_i = \mathbf{q} + \bar{\mathbf{n}}(s \cdot i), \quad 0 \leq i \leq m/2$$

$$\mathbf{h}_i = \mathbf{q} - \bar{\mathbf{n}}(s \cdot i), \quad 0 < i \leq m/2$$

For odd numbers of hex layers, hex node \mathbf{h}_i is given by

$$\mathbf{h}_i = \mathbf{q} + \bar{\mathbf{n}} \cdot s \cdot (i + 0.5), \quad 0 \leq i \leq (m-1)/2$$

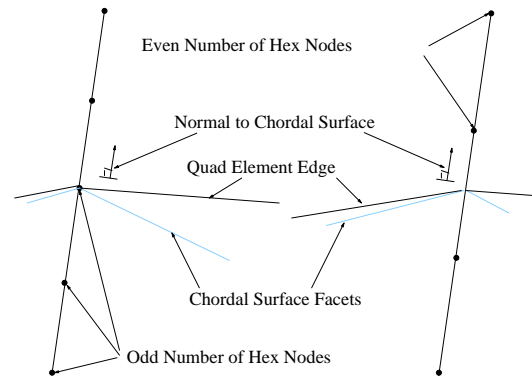


Figure 9. Placing hex nodes at a quad node

$$\mathbf{h}_i = \mathbf{q} - \bar{\mathbf{n}} \cdot s \cdot (i + 0.5), \quad 0 \leq i \leq (m-1)/2$$

where $\bar{\mathbf{n}}$ is average unit normal at \mathbf{q} and step size $s = (2 \times r) / m$.

7.3. Building Hex Elements at Quad Element

Connecting hex nodes to form hex elements is dictated by the connectivity of quad nodes in the underlying quad mesh on the chordal surface. Connectivity between the hex elements of layered mesh is obtained by the adjacency list of quad elements. Figure 10 shows even and odd number of hex elements at a quad element.

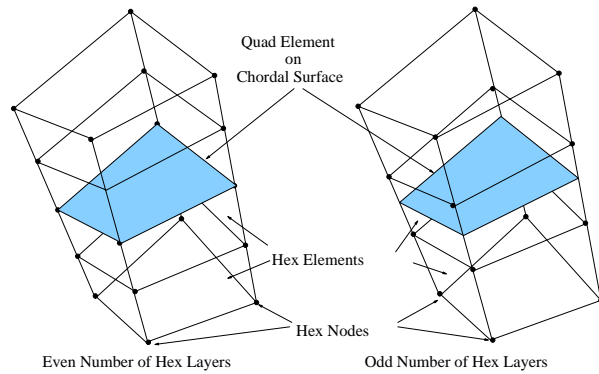


Figure 10. Layers of hex elements at quad element

8. RESULTS AND DISCUSSION

This section presents the results obtained by Hex-Layer on the objects whose chordal surface is 2-manifold and the results for two industrial parts are shown in Figures 12 and 13. The Hex-Layer algorithm is implemented in VC++ using OpenGL for graphics display.

Figures 12 and 13 show the input CAD model, tet mesh, chordal surface, quad dominant mesh and layered hex mesh of a plastic injection molding component (a car mirror casing), and a metal bracket with holes. The size of the hex elements shown in Figures 12 and 13 depend mainly on the size of the quad elements on the chordal surface. Radius function and curvature of chordal surface is used to adapt the size of the quad elements to geometric features, as shown in Figure 12(d). Figure 12(f) shows the exploded view of six-layer anisotropic all-hex mesh.

Hex-Layer generates high quality, almost cuboidal shape hex elements with the desired number of hex layers. Figure 11 shows the distribution of elements based on the aspect ratio. On the average, about 80% of the elements attain the desired aspect ratio and the remaining elements deviate only marginally from the desired value. Figure 11 shows the distribution of elements based on the scaled Jacobean. Note that the scaled Jacobean of most of the hex elements is near 1.0. The graph with a plot symbol shows the aspect ratio and scaled Jacobean of the hex elements of the geometry adaptive anisotropic two-layer all-hex mesh, shown in Figure 12,

which contains 9602 elements. The graph without a plot symbol shows the aspect ratio and scaled Jacobean of hex elements of uniform anisotropic six-layer all-hex mesh, shown in Figure 13, which contains 8664 hex elements.

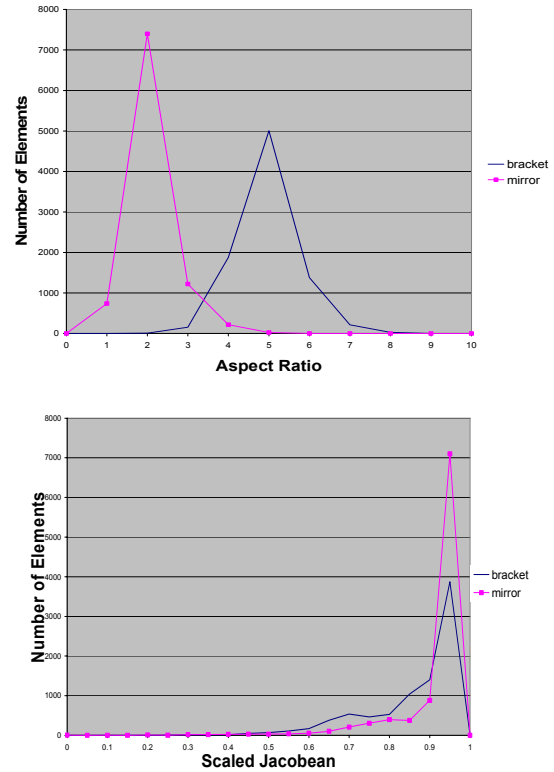


Figure 11. Quality metrics of all-hex mesh

9. CONCLUSION

A new method has been proposed for the generation of the chordal surface, which is used as a skeleton to represent the 3D thin section solids. A new hex meshing algorithm for the generation of multi-layered all-hex mesh of the thin section solids, whose chordal surface is 2-manifold, is proposed. The algorithm as such need not be restricted to these objects. Work is underway to handle the objects whose chordal surface is non-manifold. More general classification of the tets for objects whose chordal surface is non-manifold, and sweeping of quad elements along the tracks [21] (radii of maximal sphere), will make the algorithm more general.

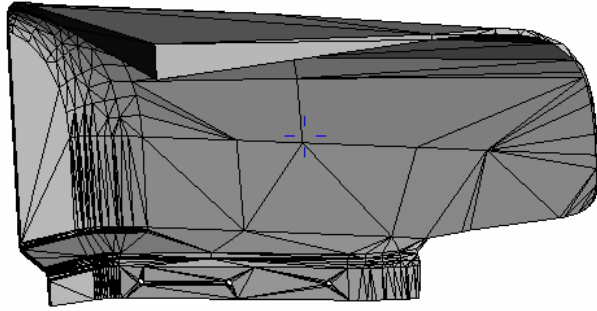
ACKNOWLEDGMENT

We thank Dr. Soji Yamakawa, for generating the tet mesh of the input CAD model and quad mesh on the chordal surface.

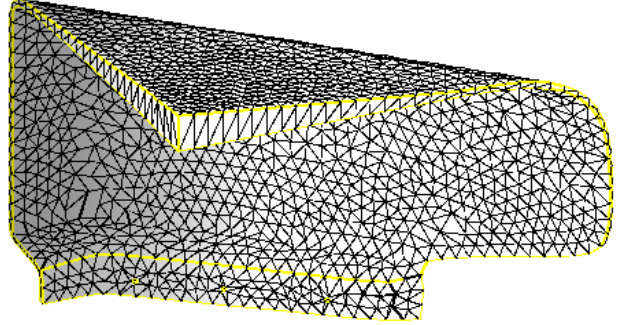
This material is based in part on work supported under a NSF CAREER Award (No. 9985288).

REFERENCES

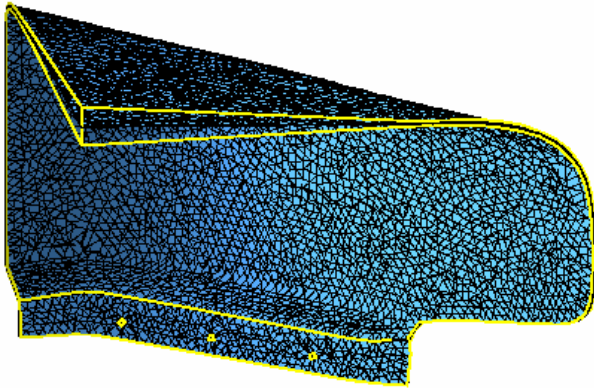
- [1] S. A. Canann, "Plastering and Optismoothing: New Approaches to Automated, 3D Hexahedral Mesh Generation and Mesh Smoothing", *Ph.D. Dissertation, Brigham Young University, Provo, UT*, (1991).
- [2] H. Blum, "A transformation for extracting new descriptors of shape", , *Models for the Perception of Speech and Visual Form, Cambridge, MA, The MIT Press*, pp. 326-380, (1967).
- [3] S. J. Owen, "A Survey of Unstructured Mesh Generation Technology", <http://www.andrew.cmu.edu/user/sowen/survey/index.html>.
- [4] K. Shimada, "Physically Based Mesh Generation: Automated Triangulation of Surfaces and Volumes via Bubble Packing", *Ph.D. Dissertation, Massachusetts Institute of Technology, Cambridge, MA*, (1993).
- [5] K. Shimada, J. H. Liao and T. Itoh, "Quadrilateral Meshing with Directionality Control through the Packing of Square Cells", *Proc. 7th International Meshing Roundtable*, pp. 61-75, (1998).
- [6] S. B. a. D. W. Lai, "Automated hexahedral mesh generation by generalized multiple source to multiple target sweeping", *Int. Journal for Numerical Methods in Engineering*, 49, pp. 261-275, (2000).
- [7] M. L. Staten, Scott A. Canann and S. J. Owen, "BMSWEEP: Locating Interior Nodes during Sweeping", *7th International Meshing Roundtable*, (1998).
- [8] K. a. T. B. Miyoshi, "Hexahedral Mesh Generation Using Multi-Axis Cooper Algorithm", *Proceedings, 9th International Meshing Roundtable*, pp. 89-97, (2000).
- [9] L. Prasad, "Morphological Analysis of Shapes", <http://cnls.lanl.gov/Highlights/1997-07/>, (1997).
- [10] M. A. Price and C. G. Armstrong, "Hexahedral Mesh Generation by Medial Surface Subdivision: Part II", *International Journal for Numerical Methods in Engineering*, 40, pp. 111-136, (1997).
- [11] S. R. Jankovich, S. E. Benzley, J. F. Shepherd and S. A. Mitchell, "The Graft Tool: an all-hexahedral transition algorithm for creating a multi-directional swept volume mesh", *Proc. 8th International Meshing Roundtable*, pp. 387-392, (1999).
- [12] P. Sampl, "Semi-Structured Mesh Generation Based on Medial Axis", *9th International Meshing Roundtable*, (2000).
- [13] R. Schneider, "An Algorithm for the Generation of Hexahedral Element Meshes Based On An Octree Technique", *Proceedings, 6th International Meshing Roundtable*, pp. Abstract195-196, (1997).
- [14] R. Schneider, "A Grid-Based Algorithm for the Generation of Hexahedral Element Meshes", *Engineering With Computers*, 12, pp. 168-177, (1996).
- [15] Soji Yamakawa and K. Shimada, "Hex-dominant Mesh Generation with Directionality Control via Packing Rectangular Solid Cells", *to appear in the proceedings of Geometric Modeling and Processing*, (2002).
- [16] Soji Yamakawa and K. Shimada, "High Quality Anisotropic Tetrahedral Mesh Generation via Ellipsoidal Bubble Packing", *Proc. 9th International Meshing Roundtable*, pp. 263-273, (2000).
- [17] Soji Yamakawa and K. Shimada, "Quad-Layer: Layered Quadrilateral Meshing of Narrow Two-Dimensional Domain by Bubble Packing and Chordal Axis Transformation", *Proceedings of ASME Design Automation Conference*, (2001).
- [18] Soji Yamakawa and K. Shimada, "Hexhoop: Modular Templates For Converting A Hex-Dominant Mesh To An All-Hex Mesh", *Proceedings, 10th International Meshing Roundtable*, pp. 235-246, (2001).
- [19] Takeo Taniguchi, Tomoaki Goda, Harald Kasper and W. Zielke, "Hexahedral Mesh Generation of Complex Composite Domain", *5th International Conference on Grid Generation in Computational Field Simulations, Mississippi State University*, pp. 699-707, (1996).
- [20] Ted D. Blacker and R. J. Myers, "Seams and Wedges in Plastering: A 3D Hexahedral Mesh Generation Algorithm", *Engineering With Computers*, 2, pp. 83-93, (1993).
- [21] W.R. Quadros, "Laytracks: An Automatic Quad Mesh Generator Based On MAT", *M.Sc Thesis Department of Mechanical Engineering, Indian Institute of Science, Bangalore, INDIA*, (2001).
- [22] W.R. Quadros, K. Ramaswami, F.B. Prinz and B. Gurumoorthy, "Automated Geometry Adaptive Quadrilateral Mesh Generation using MAT", *Proceedings of ASME DETC*, (2001).
- [23] K. R. W.R. Quadros, F.B. Prinz, B. Gurumoorthy, "LayTracks: A New Approach to Automated Quadrilateral Mesh Generation Using Medial Axis Transform", *9th International Meshing Roundtable*, (2000).



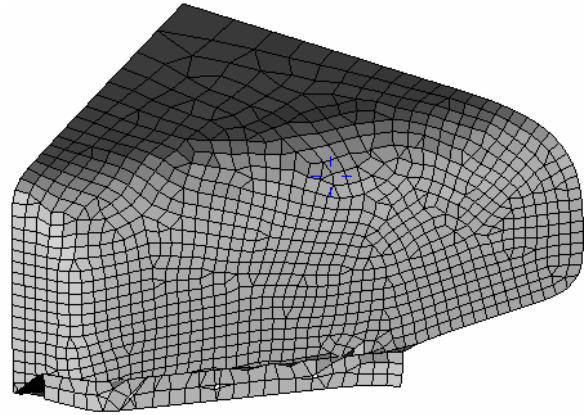
(a) Input CAD model



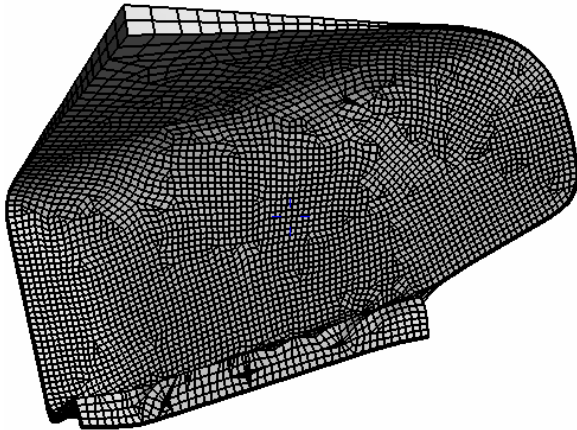
(b) Tet mesh and boundary edges



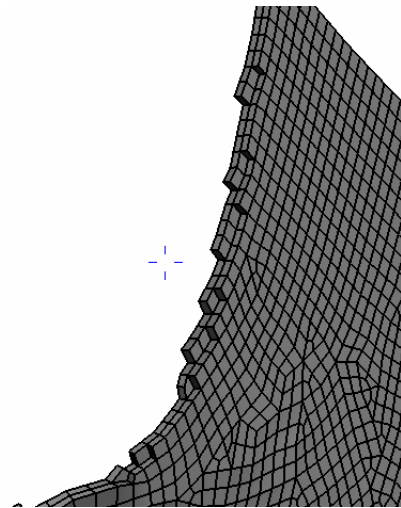
(c) Chordal surface of CAD model



(d) Quad-dominant mesh on chordal surface

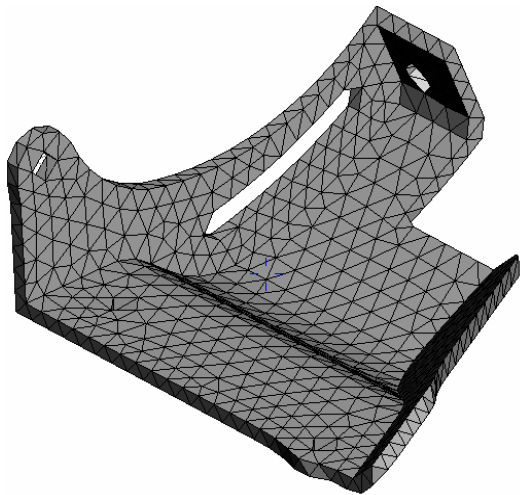


(e) Two-layer geometry adaptive all-hex mesh

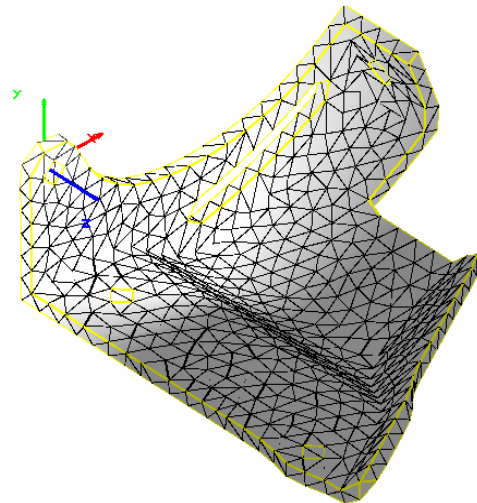


(f) Exploded view of 2-layer all-hex mesh

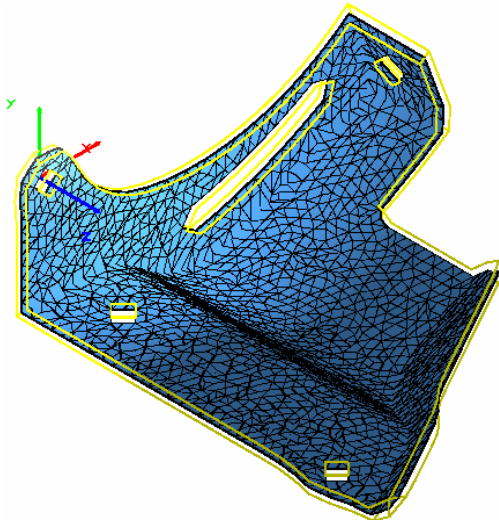
Figure 12. Plastic injection mold component (a car mirror casing)



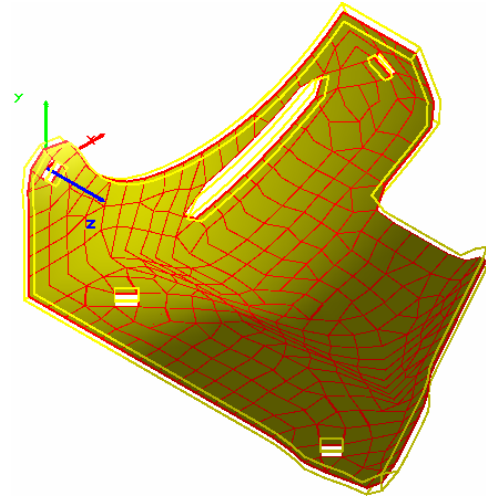
(a) Input CAD model



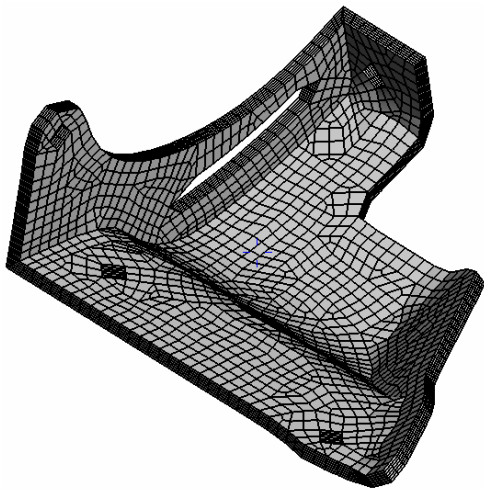
(b) Tet mesh and boundary edges



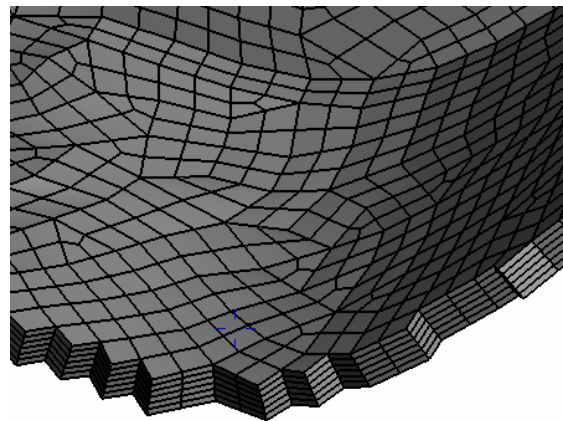
(c) Chordal surface of CAD model



(d) Quad-dominant mesh on chordal surface



(e) Six-layer uniform anisotropic all-hex mesh



(f) Exploded view of six-layer all-hex mesh

Figure 13. Thin wall metal bracket with holes