Identifying and Meshing Thin Sections of 3-d Curved Domains

Luzhong Yin, Xiaojuan Luo, Mark S. Shephard

Scientific Computation Research Center, Rensselaer Polytechnic Institute, Troy, NY 12180.
lyin@scorec.rpi.edu; xluo@scorec.rpi.edu; shephard@scorec.rpi.edu

Summary: Realization of the full benefits of variable p-version finite elements requires the careful construction of prismatic elements in thin sections. This paper presents a procedure to automatically isolate the thin sections using the points on an approximate medial surface computed by an octree-based algorithm. Using the pairs of triangles associated with medial surface (MS) points, in conjunction with adjacency, classification and distance information, sets of surface triangles that are on opposite face patches in thin sections are identified. Mesh modifications are then executed to match the surface triangulations on the opposite face patches such that prismatic elements can be generated without diagonal edges through the thickness directions.
Keywords: thin sections, medial surface, prismatic elements

1. Introduction

Historically, the methods used to analyze thin sections involved applying deformation assumptions to the 3-D elasticity equations allowing the problem dimensionality to be reduced [1]. The application of such methods requires a reduced dimensional domain model. Application of these methods requires the identification of the thin sections and then the application of model dimension reduction on those portions of the domain [2]. Handling the interconnection between two-dimensional reduced elements to fully three-dimensional solid elements is another source of difficulty [3].

Since the assumptions corresponding to those deformation models are equivalent to allowing only low order deformation modes in the thickness direction, an alternative is to apply full three-dimensional model discretized with p-version finite elements with low polynomial order through the thickness [4, 5, 6]. Tetrahedral meshes cannot effectively be used to implement the appropriate low order deformation modes through the thickness due to the presence of the through the thickness diagonals. Therefore a mesh that contains a single element through the thickness without through the thickness diagonals is needed.

The automatic generation of meshes for general 3-D domains with such elements (prism of hexahedra) in the thin sections is not a straightforward process, particularly in the case where adaptive p-version finite element methods are applied that will require large curved elements of high polynomial order in the other

directions. This paper reports on the status of efforts to development mesh generation procedures aimed at producing the desired p-version curved finite elements for models containing thin sections.

A key challenge to properly mesh the thin sections in general geometrical models is to identify and isolate thin sections from the rest of the domain. Most of efforts on meshing a thin section use a priori information on the thin section [7]. One approach identified as appropriate for identifying thin sections is the medial surface [2, 8]. The medial surface is the locus of the center of a maximum sphere rolls around the interior of the model. It is the set of interior points that is equidistant to more than one points on the model boundary. The medial surface can provide the following information on the region's geometry and topology [9]:

- Indication of local feature size (or local thickness) by the distance from a medial surface point to its closest boundary points
- Information on 'opposite' boundary points by relating the closest boundary points to a medial surface point.

The medial surface has been used to partition the geometry model into easily meshable subregions by several authors to generate volume mesh, for example [9, 10, 11]. Other mesh-related applications of medial surface include construction of three-dimensional anisotropic geometric metrics for geometric adaptation [12]. In the present procedure we determine and use a limited number of points on an approximation medial surface to identify the thin sections and generate prismatic layer mesh through thickness.

Section 2 presents the criteria to identify pairs of opposite triangles on the thin sections based on points on the medial surface for a classified surface triangulation of the model and gives an octree-based algorithm for their determination. Section 3 discusses the procedure that given those pairs of triangles determines any missing thin section triangles and isolates the thin sections. Section 4 considers the procedures for then meshing the thin sections. A general volume mesh generator is applied to fill the remaining domains for p-version adaptive analysis. The geometric approximation required by the p-version finite elements is achieved by applying a curving procedure in [13, 14]. Section 5 presents an example to show the benefits of above meshing procedure in p-version analysis.

1.1 Nomenclature

M_i^d the i-th mesh entity of dimension d, d=0 for vertex, d=1 for an edge, d=2 for a face.

$\overline{M}_i^d$ the closure of the i-th mesh entity of dimension d = 1 or 2.

G_i^d the i-th entity of dimension d in geometry model.

$[$ classification symbol used to indicate the associated of one or more mesh entities with an entity in geometry model [15].

O_i the i-th octant.

O_i^d the i-th octant entity of dimension d, d=0 for octant vertex, d=1 for octant edge.

E_i^0 the i-th medial surface point.

$p(\bullet)$ single closest point from entity $\bullet$ of dimension 0 to surface mesh.

$p_i(\bullet)$ the i-th closest point from entity $\bullet$ of dimension 0 to surface mesh.

2. Determination of Medial Surface Points and Associated Triangle Pairs

2.1 Criteria to Define Thin Section Triangle Pairs

The definition of a thin section is closely related to size of and order of the elements in the mesh. The geometric characteristic for a thin section is the dimension through the thickness is far less than the "in-plane" dimensions. We identify the thin sections using a surface triangulation of the model. The basic idea is to find pairs of triangles on "opposite model faces" that are close to each other relative to their size, thus indicating they are within a thin section.

A point on the medial surface can provide the local thickness [9] by the distance to its closest boundary points and 'opposite' boundary points by relating the closest boundary points. Therefore, the concerned pair of the opposite triangles can be defined based on a medial surface point as follows.

A pair of triangles M_i^2 and $M_{i'}^2$ is candidate *thin section triangle pair* if there exist a pair of closest boundary points P_1 and P_2 from a medial surface point E_i^0, such that $P_1 \subset \overline{M}_i^2$ and $P_2 \subset \overline{M}_{i'}^2$, where the P_1 and P_2 have following properties:

(1) The ratio of thickness (defined as the diameter of the maximum inscribed sphere associated with E_i^0) to the average size of M_i^2 and $M_{i'}^2$ is smaller than a default value, for example 1/3 of the average edge length of the element.

(2) The angle formed by the outward normal to M_i^2 and $M_{i'}^2$ is close to π.

The situation of the medial surface point defined by conditions of (1) and (2) is shown in Fig. 1.
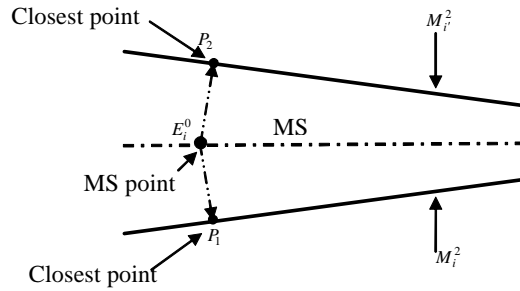
Fig. 1. A thin section triangle pair identified by a medial surface point E_i^0

A candidate thin section triangle pair is further processed to ensure that all points on their closures meet those conditions. The key step to identify the thin section

triangle pairs is to calculate the points on the medial surface of the classified sur-
face triangulation. We use octree to calculate the medial surface points with the
goal of identifying most, but not all, triangles in the thin sections.

2.2 Octree-Based Algorithm to Compute the Medial Surface Points

The medial surface points are calculated for a classified surface triangulation. The
classification information of the surface triangulation is used to ignore the medial
surface branches of the triangulated model that do not exist in the smooth curved
model. That is, the two closest points of a medial surface point on two adjacent
triangles that are classified on one C^1 continuous model face will be ignored in the
calculations. From the property that a closed geometrical model has a closed set of
medial surfaces [16], a medial octant tracing algorithm was constructed. In this al-
gorithm, medial octants are defined as octants that intersect medial surface as O_1
and O_2 shown in Fig. 2. The steps of the tracing algorithm are as follows.

- Construct octree by inserting surface mesh entities into boundary octants.
- Determine an octant with an edge that intersects the medial surface.
- Resolve all intersections of that octant edge by a traversal algorithm.
- Continue the traversal on the other edges until all intersections are resolved.
- Move to neighboring octants of the intersection points to process their other oc-
 tant edge/medial surface intersections.

The above tracing procedure can be illustrated by Fig. 2. Suppose by the traversal
algorithm discussed below, we resolve the intersections shown as E_1^0, E_2^0 and E_3^0
on the edges of O_1. After that we move to the next medial octant O_2 adjacent to
E_3^0 to process the other intersection points on all the edges of O_2. The procedure
will repeat until no new neighbor medial octant can be found.
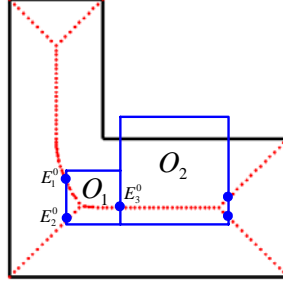
Fig. 2. Move from one resolved medial octant O_1 to the next octant O_2 adjacent to
a known medial surface point E_3^0 to calculate other new intersection locations.

   To control the medial octant size, before calculating intersections, recursively
subdivide the neighbor medial octants to be no more than one level different. Fur-
ther subdivide the medial octant to the same order of the size of surface triangles
that are closest to the octant vertices.

The goal of the algorithm is to determine the intersection between an octant edge and the medial surface. An octant edge can have multiple intersections. To determine those intersections, we employ the relationship among Voronoi regions and medial surface [8] for polyhedron. The medial surface of a convex polyhedron is identical to its Voronoi diagram which is defined as the boundaries of the Voronoi regions. If a polyhedron is non-convex, the Voronoi diagram is a superset of medial surface. In the presence of concave edges whose interior dihedral angles $w_i > \pi$, the Voronoi region associated with each of concave edges is a particular subregion bounded by a portion of the medial surface and planes perpendicular to the boundary planes at the concave edge and intersect the medial surface. In this case, the difference between the medial surface and Voronoi diagram is that medial surface does not include planes of the Voronoi diagram incident at the concave edges, shown as the dash lines in Fig. 3, where the dot lines is medial surface.
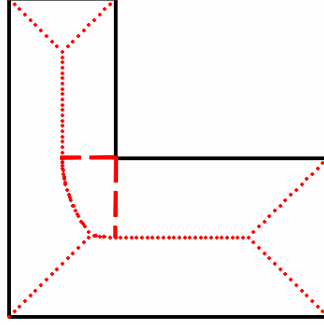
Fig. 3. Relationship among Voronoi regions, Voronoi diagram and medial surface.

Taking into account non-convex region cases, an octant edge with bounding vertices closest to two different surface entities (i.e. in two different Voronoi regions associated with the two entities) that are not adjacent to one re-entrant edge or corners, intersects the medial surface. The closest point $p(O_1^0)$ determines the Voronoi region that the vertex O_1^0 is in. That is, if $p(O_1^0)$ is found on a surface entity, then O_1^0 is in the Voronoi region associated with that surface entity. The current octree is employed to determine the closest point information. The procedure to resolve the intersections is as follows:

   Assume there is just one intersection on the edge bounded by O_1^0 and O_2^0. Using the equidistant condition to the boundary entities whose associated Voronoi regions O_1^0 and O_2^0 are in, the location of the intersection can be found for the parameter t in the interpolation formulation

$$\mathbf{E}_i^0 = (1-t)\mathbf{O}_1^0 + t\mathbf{O}_2^0 \tag{1}$$

where the bold letters denotes the location vectors at the corresponding vertices. Note a Voronoi region in a polyhedron could be associated with face, concave edge or concave corner. For the case that two Voronoi regions are associated with two faces as shown in Fig. 4, t is given as

$$t = \frac{\left(\mathbf{O}_2^0 - \mathbf{O}_1^0\right) \cdot \mathbf{n}_2 + d_2 - d_1}{\left(\mathbf{O}_1^0 - \mathbf{O}_2^0\right) \cdot \mathbf{n}_1 + \left(\mathbf{O}_2^0 - \mathbf{O}_1^0\right) \cdot \mathbf{n}_2} \tag{2}$$

where d_1 and d_2 are distances from O_1^0 and O_2^0 to their closest points on boundary, $\mathbf{n}_1$ and $\mathbf{n}_2$ are unit vectors from O_1^0 and O_2^0 to their closest points, respectively. Note that Eqn (2) is also valid when O_1^0 and O_2^0 are on the boundary, which leads to d_1 and d_2 equal to zero, and $\mathbf{n}_1$ and $\mathbf{n}_2$ are in the outward normal directions to the faces where O_1^0 and O_2^0 are on.
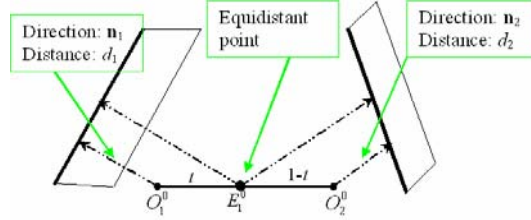
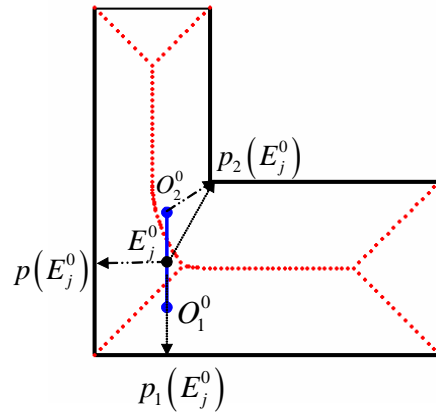
Fig. 4. Medial surface point calculation

If $p\left(O_1^0\right)$ or $p\left(O_2^0\right)$ is on concave edge or concave vertex, the equidistant condition leads to a quadratic equation to solve the parameter t in Eqn (1).

After getting the assumed intersection in Eqn (1), we request its closest points on the boundary. If multiple closest points are returned there is a single intersection. If a single closest point is returned there are multiple intersections, in which case subdivide the edge at that location and repeat until the intersections are resolved.

The efficiency of the above algorithm is illustrating using Fig. 5. Given the octant edge whose bounding vertices are O_1^0 and O_2^0, we obtain the closest points on the boundary to O_1^0 and O_2^0 and an assumed medial surface intersection point E_j^0 is found. The closest point to E_j^0 is a single point $p\left(E_j^0\right)$, thus indicating the point is not on the medial surface. After subdivide the edge at E_j^0, we obtain the correct intersection points, one on each sub-edge bounded by $\left(O_1^0, E_j^0\right)$ and $\left(E_j^0, O_2^0\right)$.

It is noted the tracing algorithm can start from a convex model edge whose interior dihedral angle is $w_i < \pi$, or by a medial surface point calculated on a ray in the direction of the normal to a surface triangle to the interior of model by the above edge intersection algorithm. Operations to calculate medial surface points are not applied to octant edges external to the model.

Fig. 6 shows a result of the above algorithm for an example. Fig. 6(b) shows the medial octants that are traversed by the above tracing algorithm. The medial octant sizes are refined to match surface mesh size.

Fig. 5. Multiple intersection on the edge bounded by O_1^0 and O_2^0
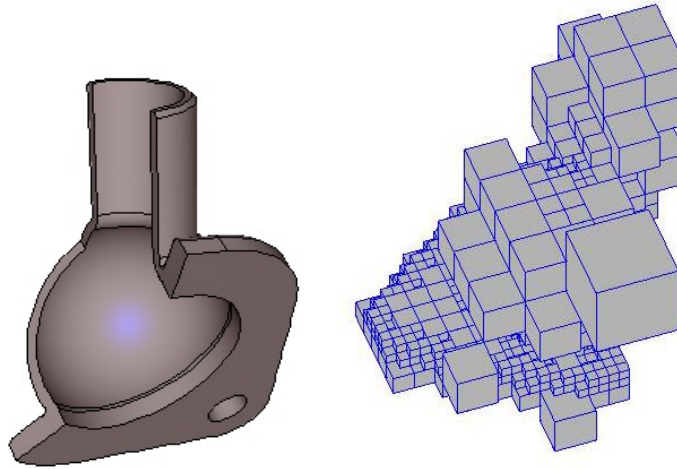
Fig. 6. (a) Model; (b) traversed medial octants.

3 Defining Thin Sections

The medial surface point calculations provide a set of unorganized and incomplete thin section triangle pairs. We organize the thin section triangles in the sets using knowledge of which model faces they are classified on. After the initial sets are

constructed, the missing thin section triangles are determined. The procedure has three steps:

- Collect starting triangle sets using classification information.
- Complete the triangle sets to define thin section surface patches
- Construct the loops for each thin section surface patch and match the loops on the boundary of opposite thin section patches.

3.1 Collect Starting Thin Section Triangle Sets

Given medial surface point E_i^0, introduce

$$\left| E_i^0 \right|_* = \begin{cases} 1 & thin \\ 0 & not\ thin \end{cases} \tag{3}$$

to indicate when E_i^0 defines a thin section triangle pair. $\left| E_i^0 \right|_* = 1$ indicates the medial surface point is associated with thin triangle pair and $\left| E_i^0 \right|_* = 0$ means they are not part of a thin section, see Section 2.1. Denote the triangle pair as

$$\left[E_i^0 \right]_* = \left\{ M_k^2, M_{k'}^2 \right\}. \tag{4}$$

With the above notations, a starting thin section triangle set is defined as

$$\hat{G}_j^2 = \left\{ M_i^2 \middle| M_i^2 \left[\ G_j^2 \text{ and } M_i^2 \in \left[E_l^0 \right]_* \text{ and } \left| E_l^0 \right|_* = 1 \right\} \tag{5}$$

Note that each $\hat{G}_j^2$ is uniquely associated with model face G_j^2. For this unique association, the identity tags of "opposite" sets for $\hat{G}_j^2$ can be recorded during the construction of $\hat{G}_j^2$. Generally, $\hat{G}_j^2$ may have one or more opposite sets denoted as $opp(\hat{G}_j^2)$. A simple example in Fig. 7 shows $\hat{G}_1^2 = \left\{ M_a^2, M_b^2 \right\}$, $opp(\hat{G}_1^2) = \left\{ \hat{G}_3^2 \right\}$; $\hat{G}_2^2 = \left\{ M_c^2, M_d^2, M_e^2 \right\}$, $opp(\hat{G}_2^2) = \left\{ \hat{G}_3^2 \right\}$ and $\hat{G}_3^2 = \left\{ M_{a'}^2, M_{b'}^2, M_{c'}^2, M_{d'}^2, M_{e'}^2 \right\}$, $opp(\hat{G}_3^2) = \left\{ \hat{G}_1^2, \hat{G}_2^2 \right\}$.
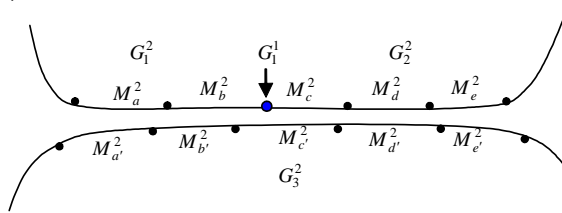

Fig. 7. An example to demonstrate the starting triangle sets

Note sets at this point may have to be later split or merge to represent a thin section surface patch.

3.2 Determining the Missing Thin Section Triangles

The majority of thin section triangles are identified by the medial surface points in the tracing algorithm but some are missed, see Figure 8 (a), where the dark shaded triangle faces are the identified thin section triangles.
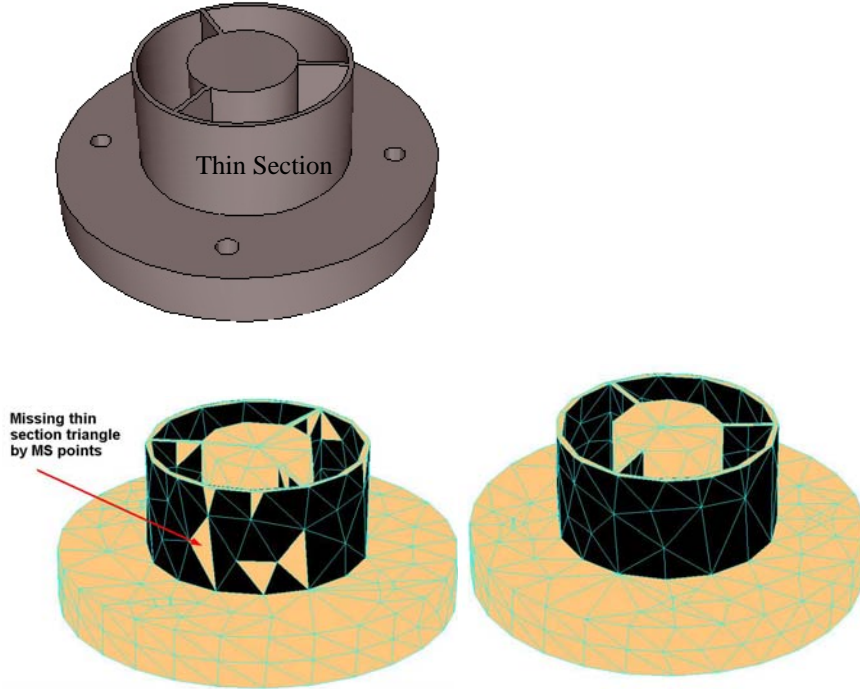

Fig. 8. (a) Geometry model with thin section; (b) Thin section triangles obtained by medial surface points; (c) The completed thin section surface patches.

To determine whether a missing triangle M_e^2 on G_j^2 belongs to $\hat{G}_j^2$, local thickness h_e at M_e^2 is compared with the local thickness h_i at an edge adjacent triangle M_i^2 that is inside $\hat{G}_j^2$. If $|h_e - h_i|/h_i$ is smaller than a default value, place M_e^2 in $\hat{G}_j^2$. The local thickness h_e at M_e^2 is obtained by searching for the closest point on the triangles classified on the model faces that are known to be opposite to G_j^2. The triangle $M_{e'}^2$ that the closest point is on is defined as opposite triangle to M_e^2. Also place $M_{e'}^2$ in the set that is opposite to $\hat{G}_j^2$ if it is not there already. Note, $M_{e'}^2$ must be in the neighborhood of $M_{i'}^2$ which is opposite to M_i^2. The search is a local operation. Fig 8 shows an example before (Fig 8 (b)) and after completing the thin section triangle sets (Fig. 8(c)). In Fig 8(b), the faces in black are determined by medial surface points and those in gray on the thin sections are missing surface triangles. Fig. 8(c) shows all the thin section triangles are recovered shown

in black. Note the boundary edges of $\hat{G}_j^2$ can also be identified as those used by only one triangle in the set. We denote the set of boundary edges of $\hat{G}_j^2$ as $\partial \hat{G}_j^2$.

3.3 Construct the Boundary Loops on Thin Section Surface Patches

To complete the definition of the thin section surface patches opposite each other, the loops on the boundary of surface patches have to be matched. The process can lead to the need to split surface patches. Figure 9 (a) shows an example that has three thin section surface patches, where $\hat{G}_3^2$ is opposite to $\hat{G}_1^2$ and $\hat{G}_2^2$ with each loop on each of the sets. In this case, splitting $\hat{G}_3^2$ to form two loops on $\hat{G}_3^2$ to match the loops on $\hat{G}_1^2$ and $\hat{G}_2^2$ is needed as shown in Fig. 9 (b). Note the loops in $\hat{G}_1^2$ and $\hat{G}_2^2$ cannot be merged to form one loop since the model edge must be used in the volume mesh generation.
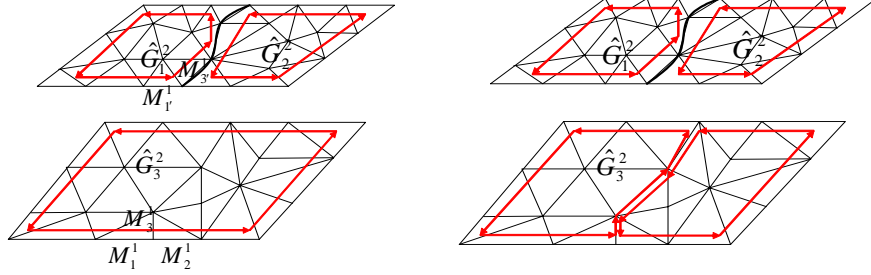
Fig. 9. (a) Loops on thin section surface patches (b) Opposite loops

The procedure to create the loop L_k and its opposite loop $L_{k'}$ can be started from any $M_f^1 \in \partial \hat{G}_j^2$ which looks for its opposite edge $M_{f'}^1 \in \partial \hat{G}_{j'}^2$ through the opposite triangles in following steps:

- If $M_f^1 \in \partial \hat{G}_j^2$ can find its opposite edge $M_{f'}^1 \in \partial \hat{G}_{j'}^2$, the procedure continues to process the next edge $M_d^1 \in \partial \hat{G}_j^2$ adjacent to M_f^1. For example, in Fig. 9(a), the edge $M_{1'}^1 \in \partial \hat{G}_1^2$ is found opposite to $M_1^1 \in \partial \hat{G}_3^2$. The next edge $M_2^1 \in \partial \hat{G}_3^2$ adjacent to M_1^1 is processed.

- If $M_d^1 \in \partial \hat{G}_j^2$ cannot find its opposite edge on $\partial \hat{G}_{j'}^2$, then from the edge $M_{d'}^1 \in \partial \hat{G}_{j'}^2$ adjacent to $M_{f'}^1 \in \partial \hat{G}_{j'}^2$, its opposite edge is found from the interior edges of $\hat{G}_j^2$ adjacent to M_f^1. Fig. 9 shows a such case, where $M_2^1 \in \partial \hat{G}_3^2$ cannot find an opposite edge on $\partial \hat{G}_1^2$, therefore, from $M_{3'}^1 \in \partial \hat{G}_1^2$ adjacent to $M_{1'}^1$, the in-

terior edge M_3^1 is found opposite to it. The procedure leads to split thin section surface patches from Section 3.2 as shown in Fig. 9(b).

In the above procedure, the interior edge refers to the edge not belonging to $\partial \hat{G}_j^2$ but on the closure of the triangles in $\hat{G}_j^2$.

4. Meshing Thin Sections

The thin section information obtained from Section 3 is characterized by a pair of opposite thin section surface patches with paired closed opposite loops as boundaries. To generate structured prismatic elements without long diagonal edges through the thickness the opposite triangulation sets for thin section surface patch must be topologically matched and be geometrically similar.

4.1 Overall Algorithm

The overall procedure to mesh each thin section consists of the following steps:
- Apply local mesh modifications to match the thin section boundaries
- Delete the surface triangulation of one triangle set
- Copying the remaining triangulation to its opposite model face
- Connect the matched triangulations to form prismatic elements

4.2 Boundary Matching

The procedure to match the boundaries of the triangle sets for a thin section can be divided into two continuous operations. First, apply split or collapse operations to modify the mesh topology to ensure the mesh edges in each paired opposite loops are one-to-one matched. Second, the desired target locations for the vertices in the loops are computed and local mesh modifications as in [13] are applied to incrementally move the vertices towards to the target locations.

4.2.1 Topological Matching

For each pair of opposite loops, the process begins to traverse one loop through vertex adjacency information from one selected starting mesh edge and vertex to match the topological configuration to its opposite loop. Split and/or collapse operations are used to keep a loop L_i one-to-one matching with its opposite loop $L_{i'}$. The procedure starts from a vertex with lowest dimensional classification.

For each mesh edge M_i^1 in the closed loop L_i, retrieve the attached edges $\{M_j^1\}$ in the opposite closed loop $L_{i'}$. Let

$$| M_j^1 |^* = \begin{cases} 1, assigned \\ 0, unassign \end{cases} \tag{4}$$

be the operation to determine whether the mesh edge M_j^1 already associates with one mesh edge in loop L_i. If only one mesh edge M_j^1 is attached and $| M_j^1 |^* = 0$,

let M_j^1 associate with M_i^1 and update $|M_j^1|^*=1$ and continue to next mesh edge M_{i+1}^1. Otherwise, either split or collapse is applied to produce or eliminate mesh edges to maintain the one-to-one matching. As examples, Figure 10 shows that a split operation is applied on mesh edge M_1^1 to produce one more edge M_2^1 to obtain the matched pair edges (M_1^1, M_1^1) and (M_2^1, M_2^1) then update $|M_2^1|^*=1$. Considering the mesh vertices will be moved in the next step, the split operation applied currently does not snap the new introduced vertices to the model boundary.
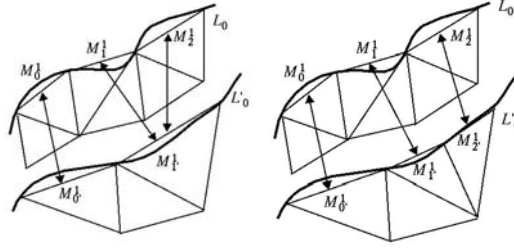
Fig. 10. Split operation to assign one-to-one match for mesh edge M_2^1

4.2.2 Target Location for Vertices in the Opposite Model Face

To achieve the geometrical similarity between the two triangle sets for a thin section, each vertex M_i^0 in one triangle set need to compute an appropriate location for its matched vertex M_i^0 on the opposite model face. The target location for M_i^0 is obtained by first computing the closest point P_i^0 on the mesh face M_i^2 classified on the opposite model face to M_i^0, and then, projecting P_i^0 to the model face by the model parameters of P_i^0 determined by the interpolation of P_i^0 on the triangle face M_i^2.

4.2.3 Incrementally Move Vertices on the Thin Section Boundary

The movement of the mesh vertices on the thin section boundary loop L_i can cause the surface mesh to become invalid. Fig 11 shows an example where two triangle faces marked as shaded in Fig 11(b) become invalid because of moving vertices M_2^0 to its target locations. This problem is avoided by applying the following procedure [13]:

- Put all of the vertices with attached target locations into a list
- Traverse the list and deal with one vertex at each step
- If the vertex moves to the target location without causing any problem, move it and remove it from the list. Otherwise, apply local mesh modifications to correct the invalid elements. Remove the vertex from the list (See reference [13]).
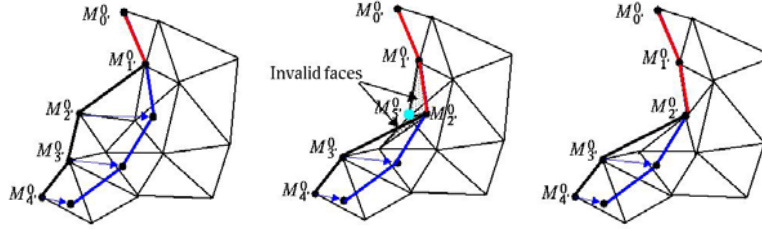
Fig. 11. (a) Move M_0^0 and M_1^0. (b) Move M_2^0, invalid faces marked as shaded. (c) Collapse M_5^0 to M_2^0.

4.3 Surface Triangulation Matching

The surface triangulation matching between the two triangle sets for a thin section is achieved through the triangulation deletion of one triangle set and the copying of the remaining triangle set to the opposite model face. The key technique is the location computation of the copied vertices on the opposite model face. This is accomplished by the interpolation strategy discussed in Section 4.2

4.4 Volume Mesh Generation

With the topologically and geometrically matched surface triangulation for thin sections, the volume mesh generation procedure constructs prismatic elements by directly connecting each paired triangles classified on the two opposite model faces. Because the generalized volume mesh generator requires the exposure mesh faces to be triangles, one layer of pyramid elements are added neighboring to the interior quadrilateral faces of the structured prismatic elements.
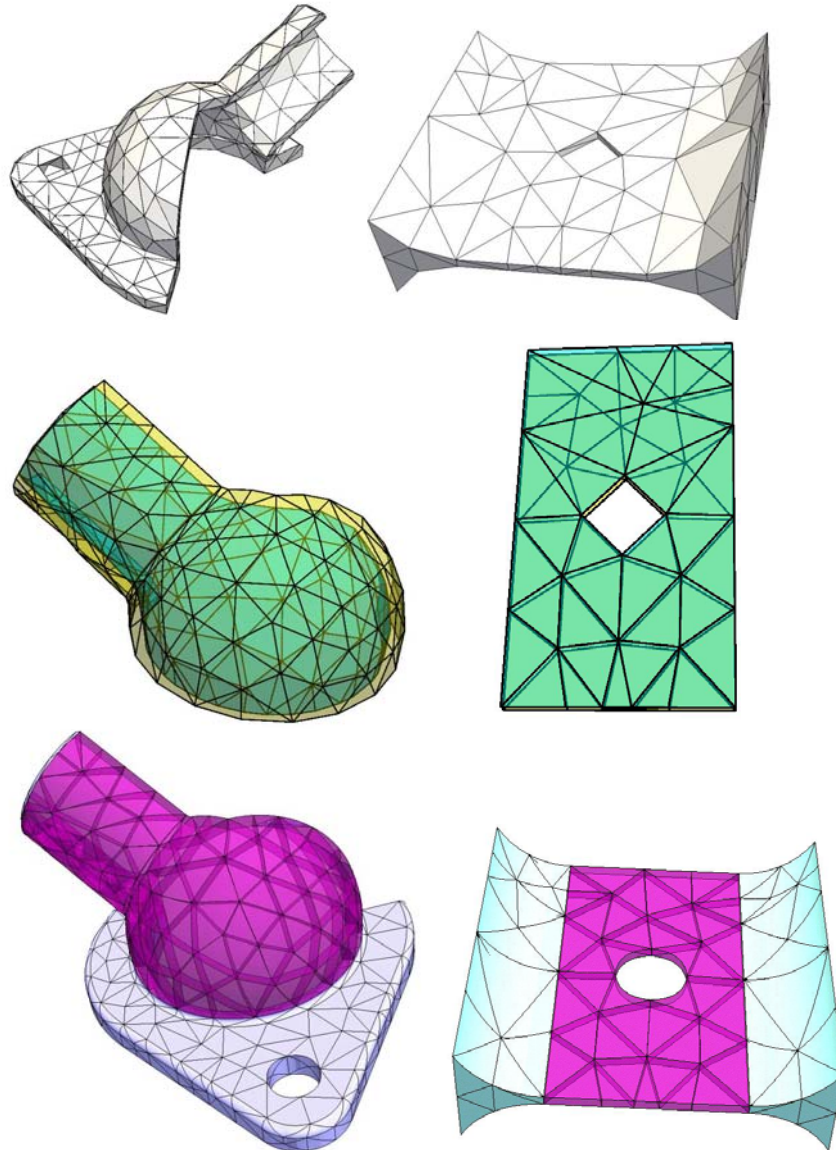
4.5 Examples

Two example models with thin sections are given in Figure 12. The first row shows the input surface triangulation for the two models. The second row shows just the isolated opposite thin section surface patches after loop construction and loop matching. As can be seen from the figures the two surface meshes are not yet matched. The bottom row shows the final meshes where the thin section meshes have been matched, the prismatic thin section elements created, volume mesh completed and the mesh properly curved to the domain boundary.

5. p-Version Analysis Example for Model with Thin Sections

In this section, a structural part with thin sections shown in Figure 13 is analyzed using p-version method. Due to the symmetry of the problem, only one half of domain is analyzed by assigning properly symmetric boundary conditions. For this particular model, there are two portions of the domain can be regarded as thin

sections with thickness 0.5 (marked in Figure 13) determined by the dimensions of the model.

Figure 12: Examples meshes for models with thin sections: Surface triangulations (top). Thin sections (middle). Curved meshes with prismatic elements (bottom).
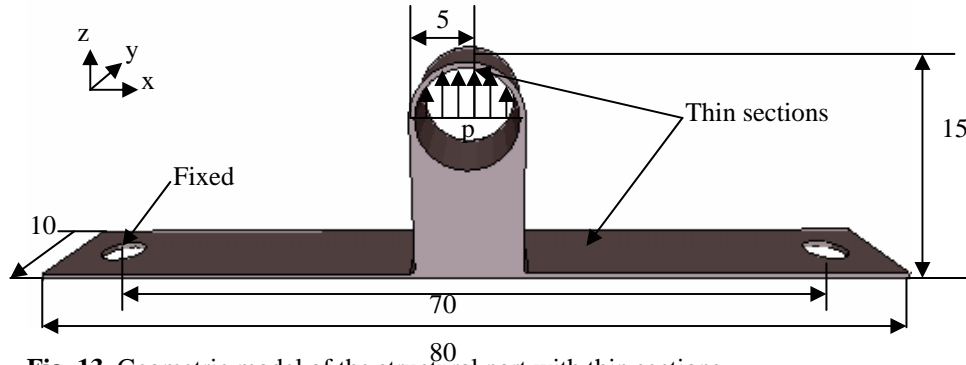
Fig. 13. Geometric model of the structural part with thin sections

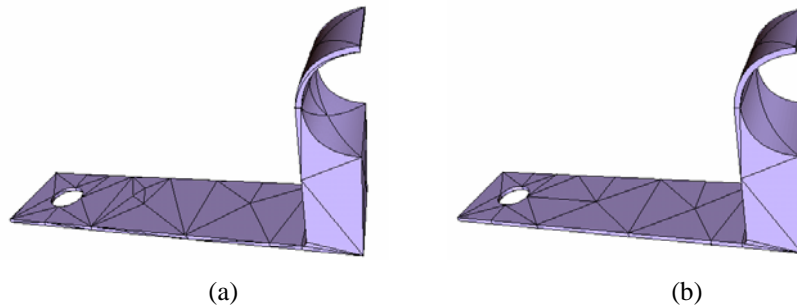(a)   (b)

Fig. 14. Two meshes for one quarter of the domain

Figure 14 shows two curved meshes for the one half of the domain. The mesh in Figure 14(a) is generated by the automatic tetrahedral volume mesh generators and the mesh in Figure 14(b) is generated using the procedure presented in this paper that has mixed topological elements. The two thin section structures are meshed with prismatic elements without diagonal edge through the thickness directions in Figure 14(b) comparing to the all tetrahedral element mesh. Table 1 presents the summary of these two meshes and the statistic indicates that number of elements for mixed topological mesh has been reduced almost 50% comparing to the all tetrahedral mesh.

Table 1. The comparision between the two meshes

	Regions			Faces		Edges	Vertices
All tetraheral	131			328		264	67
Mixed topology	66			197		194	63
	Tet	Prism	Pyramid	Tri	Quad		
	38	24	4	149	48		

The material is assumed to be linearly elastic with Young's modulus E = 200,000PA and Poission's ratio v =0.33. The quadratic z direction pressure

$p = 1. - (x/4.5)^2$ is applied on the inner circular surface that will cause the bending of the thin sections. The problem is analyzed using:

- Uniform p-version method that varies the polynomial from p=2 to p=8.
- Adaptive p-version method that enriches the polynomial independently at each coordinate directions of the elements.

The implicit element residual posterior error estimator [17] is applied to evaluate the solution accuracy in relative error in energy norm e_r as

$$e_r = \sqrt{\frac{\|u - u_X\|}{\|u\|}}\% = \sqrt{\frac{\|e\|}{\|u\|}}\% = \sqrt{\frac{\sum_k \eta_k}{\|u\|}}\% \tag{5}$$

where u, u_X are the exact solution and finite element solution of the problem. $e = u - u_X$ defines the error and η_k is the elemental error estimator [17]. $\|u\|$ is the exact strain energy. Since the true exact value is not known, we adopt the value 53.7761 computed from StressCheck [18] on a mesh with 3081 tetrahedral regions at polynomial order 8. Table 2 presents the analysis results with respect to the number of degree of freedom for uniform p-version method and Table 3 for adaptive p-version method.

Table 2: Uniform p-version analysis results

p-order	All tetrahedral mesh		Mixed topology mesh	
	Dof	e_r	Dof	e_r
2	890	60.42	710	47.74
3	2566	28.86	1699	15.39
4	5586	17.78	3380	12.20
5	10343	11.61	5939	7.39
6	17230	7.76	9574	4.13
7	26640	5.56	14483	2.16
8	38966	4.15	20864	1.57

Table 3: Adaptive p-version analysis results

Iteration step	All tetrahedral mesh		Mixed topology mesh	
	Dof	e_r	Dof	e_r
1	890	60.42	710	47.74
2	2635	24.02	1925	14.35
3	2837	22.08	2145	10.46
4	3371	17.06	3536	8.87
5	4129	11.19	5767	4.36
6	4432	9.02	6768	1.97

Results show that,

- In case of the uniform p-version method, the mixed topology mesh can obtain more accurate solution (1.57%) than the all tetrahedral mesh (4.15%) with only 54% of the degrees of freedom.
- In case of the adaptive p-version method for the all tetrahedral mesh, the adaptive analysis stops after 6 iteration steps with an unsatisfied solution accuracy level (9.02%) because some elements reached the polynomial limitation p=8.
- For the mixed topology mesh, the adaptive p-version method uses only 47% of the degrees of freedom than used for uniform p-version method to achieve an solution accuracy that is slightly better. The highest polynomial order at the last step is p=7 for the elements next to the fixed cylinder hole.

6. Closing Remarks

The paper presented a procedure to automatically isolate and mesh thin sections of 3-D solid models with prismatic elements for directional p-version finite element analysis. Key ingredients of the procedure are:

- Construction of an octant tracing algorithm to calculate a limited number of medial surface points to define the thin section triangle pairs,
- A strategy to organize the thin section triangle pairs to define thin section face patches that are opposite to each other,
- A procedure to generate prismatic volume mesh by copying one side of surface mesh to the other side and connecting the corresponding opposite vertices.

Uniform p-version analysis results clearly demonstrated that the mixed topology mesh generated by the presented procedure for model with thin section can obtain more accurate solution substantially fewer degree of freedom than the mesh with all tetrahedral elements.

7. Acknowledgement

References

1. J.N. Reddy, Theory and analysis of elastic plates. 1999, Taylor & Francies.
2. R.J. Donaghy, C.G Armstrong, M.A. Price (2000) Dimensional Reduction of surface models for analysis. Eng. with Computers 16: 24-35.
3. RL Actis, SP Engelstad, BA Szabo (2002) Computational requirements in design and design certification, Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, pp. 1379-1389.

4. Duster, H. Broker, E. Rank (2001) The p-version of the finite element method for three dimensional curved thin walled structures, Int. J. Numer. Methods Engrg, 52: 673-703.
5. C.A. Duarte, I. Babuska (2002) Mesh-independent p-orthotropic enrichment using the generalized finite element method, Int. J. Numer. Methods Engrg, 55: 1477-1492.
6. S. Dey, M.S. Shephard, J. E. Flaherty (1997) Geometry representation issues associated with p-version finite element computation, Comput. Methods. Appl. Mechanics. Engrg., 150: 39-55.
7. C.K. Lee, Q.X. Xu (2005) A new automatic adaptive 3D solid mesh generation scheme for thin-walled structures. Inter. J. Numer. Meth. Engng 62: 1519-1558.
8. N. M. Patrikalakis, H. N. Gursoy (1990) Shape interrogation by medial axis transform, ASME Advances in Design Automation, 16th design automation conf., Chicago, DE-vol 23, 77-88.
9. P. Sampl (2001) Medial axis construction in three dimensions and its application to mesh generation. Engineering with Computers 17: 234-248.
10. M. A. Price, C. G. Armstrong, M. A. Sabin (1995) Hexahedral mesh generation by medial surface subdivision: I. Solids with convex edges, Int. J. Numer. Meth. Eng, 38, 3335-3359
11. M. A. Price, C. G. Armstrong (1997) Hexahedral mesh generation by medial surface subdivision: part II. Solids with flat and concave edges, Int. J. Numer. Meth. Eng, 40, 111-136
12. K-F Tchon, M. Khachan, F. Fuibault, R. Camarero (2005) Three-dimensional anisotropic geometric metrics based on local domain curvature and thickness, Computer-Aided Design 37: 173-187
13. X. Li, M.S. Shephard, M.W. Beall (2003) Accounting for curved domains in mesh adaptation, Int. J. Numer. Methods Engrg, 58: 247-276.
14. X.J. Luo, M.S. Shephard, R.M. Obara, R. Nastasia, M.W. Beall (2004) "Automatic p-version mesh generation for curved domains", Engrg. Comput. 20: 265-285.
15. M.W. Beall and M.S. Shephard (1997) A general topology-based mesh data structure, Inter. J. Numer. Meth. Engng 40: 1573-1596.
16. A. Lieutier (2004) Any open bounded subset of r^n has the same homotopy type as its medial axis, Computer-Aided Design 36: 1029-1046.
17. M. Ainsworth and J.T. Oden (2000) A posteriori error estimation in finite element analysis, A wiley-interscience publication.
18. StressCheck User's Manual (2005)