32

Medial Surface Generation and Refinement

G. Renner and I.A. Stroud
Computer and Automation Research Institute, Hungarian Academy of Sciences, Budapest

Abstract

Generating the MAT of a three-dimensional solid is a complicated and time consuming task. This paper presents methods of improving efficiency and reliability, first using detailed analysis of geometric cases to determine critical points and secondly using multiple start points for MAT surface calculation. Because of the requirement that the original object be planar the MAT surface is an approximation to the true MAT but it is possible to refine the MAT in a post-processing step to improve its quality.

Keywords

3D MAT, Medial Surface

1 INTRODUCTION

The medial surface (Medial Axis Transform surface or simply MAT) as the skeleton of an object reflects all its topological features in a very compact form. Together with the related radius function it also provides complete geometric information about the solid. Thus the MAT of an object is a higher level abstraction which is unique and invertible. All these characteristics make the MAT a potential alternative representation for geometric modelling in CAD systems. While in the case of the very common boundary representation it is difficult to gain an overall view of an object because the representation method is essentially local, the MAT provides global information. An additional advantage is that the dimensionality of the MAT is lower than that of the original object. This makes the MAT practical for applications in the fields of feature recognition, path planning, robot motion planning, finite element analysis, etc.

There are several theoretical investigations for MAT representations. The topological properties of the MAT are discussed by Wolter (1992). Brandt (1991) (1992) and Chiang (1992) analyse the mathematical properties of the MAT for 2D regions. The three dimensional case is discussed by Nackmann and Pizer (1985).

Several algorithms have been published to compute the MAT of a planar region with different types of boundary (e.g. Montanari (1969), Preparata (1977), Srinivasan and Nackman (1987), Gursoy (1989)). Some of them may be instructive for the 3D case, however such an algorithm cannot be derived from them.

M. J. Pratt et al. (eds.), *Product Modeling for Computer Integrated Design and Manufacture*

There exist discrete type algorithms for the computation of the MAT which can be used for 2D and 3D objects. Such algorithms are e.g. the octree based calculation by Lavender et al. (1992), the numeric simulation of a wave propagation by Scott et al. (1989) or the digital thinning process by Sudhalkar et al. (1993). Brandt (1992), Yu et al. (1991), Sheehy et al. (1995) make a discretisation of the boundary of the object, then apply a discrete- point Voronoi calculation for the point set to obtain the points of the MAT.

There are currently three main continuous MAT calculation methods i.e. where neither the volume nor its boundary are discretised. Hoffmann (1994) published a method for the computation of the MAT of a CSG object. The other two, by Sherbrooke et al. (1993) and by Reddy and Turkiyyah (1995) calculate the MAT surface of boundary models. This paper describes developments of the algorithm which was originally developed by Reddy and Turkiyyah. The Reddy/Turkiyyah algorithm bears some resemblance to that of Sherbrooke, Patrikalakis and Brisson (1993), but ignores the geometry of the MAT surface edges and concentrates on constructing the topological arrangement. Like Sherbrooke et al. the algorithm starts by finding a "seam end" as start point, then finds the corresponding junction point and grows the MAT surface incrementally from this. The main difference is that it does not march along the MAT edges determining the geometry but jumps straight to the junction points, which means that the geometry of the MAT edges derived by this method is only approximate.

2 MAT OF A SOLID OBJECT

2.1 Basic properties of the MAT

Before discussing the algorithm for calculation of the MAT of a solid object, some basic definitions should be noted.

The maximal inscribed sphere of an object is a sphere which is contained in the object but which is not a proper subset of (i.e. completely contained in) any other sphere inside the object. The medial axis (MA) or skeleton of a solid object is the locus of the centre points of all maximal spheres inside the object. Some authors take the limit points of the above locus in order to make the medial axis closed and compact if the object is bounded. Because the MA of a 3D object is a structure containing basically faces it is also called the "Medial Surface". The radius function of the MA is a continuous function at the points of the MA and is equal to the radius of the corresponding maximal sphere. The MA together with the radius function is the Medial Axis Transform (MAT) of the object.

The spatial position and size of a sphere is totally defined by four scalar values (three for the coordinates of the centre point and one for the radius). Thus a vertex of the MAT is determined by four constraints, i.e. four touching points where the sphere with centre point at the vertex touches the boundary of the object (e.g. point K in Figure 1a). If the object does not have curved faces this also means four distinct boundary elements. If the number of touching points is three then one degree in the determination of the maximal sphere is left, i.e. the sphere may move along a curve making an edge of the MAT structure (KL). In the case of two touching points (two boundary elements involved) a surface in the MAT is obtained (KLMN).

These are, however, the basic regular cases for the topological elements of the MAT of a solid object. Degeneracies may occur, for example a block with square cross-section shown in Figure 1b. Here the vertices of the MAT (points K and L in Figure 1b) are determined by five elements (each by an end-surface and a dependent set - a set of elements one of which is redundant - of four side faces) and the

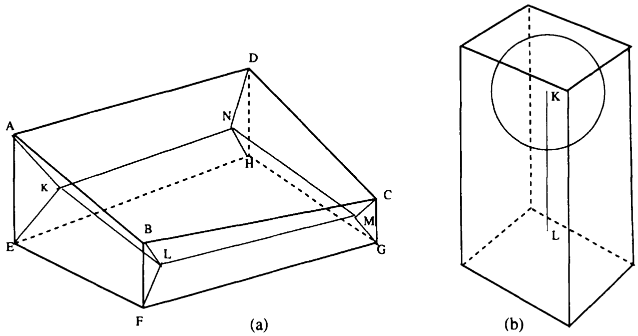


Figure 1 Medial surface (MAT) elements

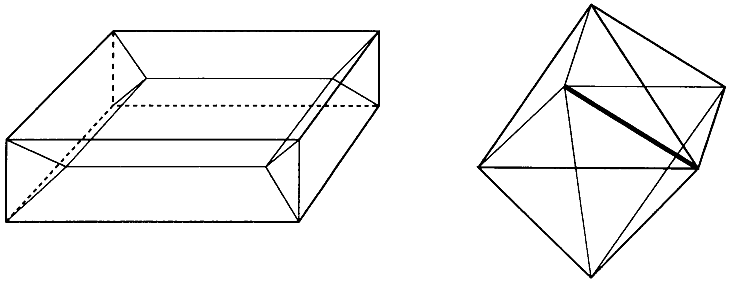
edge of the MAT is determined by the dependent set of the four side faces. For a cube, the MAT consists of a single vertex defined by a dependent set of six elements.

The relevant boundary elements for calculation of the MAT are all faces, concave edges and concave vertices (i.e. where two or more concave edges meet). It is clear that a sphere with non-zero radius inside cannot touch a convex edge, nor a vertex where only convex edges meet. Similar considerations apply for vertices where only one concave edge is present.

2.2 Tetrahedral structure

The above basic topological structure of the MAT can be represented by a tetrahedral structure, see Figure 2. The four 'vertices' or corner elements of the tetrahedron are simply references to the elements which define the critical point. Tetrahedra can be "complete" (where four elements defining a critical point have been found) or "incomplete" (where no suitable fourth element was found). Each complete tetrahedron corresponds to a 'critical point' or vertex of the MAT and has four associated 'triangular' facets. The triangular facets correspond to every combination of the four elements associated by the tetrahedron. Each set of three elements defines an edge of the MAT, some of which connect to other critical points, other edges are the so-called 'wings' of the MAT running out to the boundary of the object. The incomplete tetrahedra define the end-points of the wings. The edges of the tetrahedra are called here "struts", and can be classified as "internal" (surrounded by complete tetrahedra) or external (where at least two - only one is not possible - adjacent tetrahedra are incomplete).

The MAT calculation method in effect produces the 'dual' of the MAT which is represented as a connected set of tetrahedra, described above. It should be emphasised that the tetrahedral structure has purely logical or organisational meaning, and has to be converted to produce the MAT structure.



MAT components

6 bounding entities
4 vertices
12 edges
1 loop

Tetrahedral structure

6 vertices
4 tetrahedra
12 triangles
1 internal strut

Figure 2 Tetrahedral structure

3 BASIC METHOD FOR CALCULATION OF THE MAT

Our algorithm works with objects with planar geometry, so the first step is to facet the object to approximate the curved faces with planar facets and the curved edges by several straight edges.

Some simplifications were also introduced into the final MAT. The MAT of a planar polyhedral model may be rather complicated. At the same time it is satisfactory for many purposes to compute an approximate MAT which correctly reflects the structure of the precise MAT. For this reason we concentrate on correct computation of the medial surface vertices and their topological connections which are derived from the terahedral structure described above. In our solution MAT vertices are connected by straight lines rather than the exact curves. Also, closed loops of linked MAT edges are not represented geometrically as surfaces. All this means that we determine an exact topology but only an approximate geometry. However, the MAT topology and geometry can be refined in a post-processing step (see section 6).

3.1 MAT calculation steps

Our algorithm is an extension of the Reddy/Turkiyyah algorithm which copes with general objects with holes by looking for multiple pieces of MAT surface. Another development, described in section 5, was to produce multiple start points. This means, effectively, that the algorithm works in parallel and is faster, but hasn't changed the basic algorithm.

The basic algorithm can be summarised as follows:

1. Make a list of all relevant units in the object. A unit is either a face, a concave edge or a vertex where two or more concave edges meet.
2. Determine a starting point, in the original algorithm a convex vertex where at least three faces

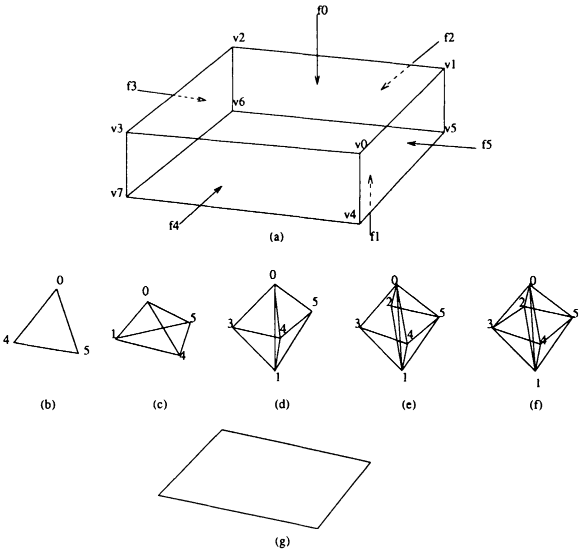


Figure 3 MAT calculation for a block

meet along convex edges. These first three elements form a 'seed triangle' which is placed on a list of triangles to be processed

3. Pick a triangle from the list to be processed. If none exists, look for unused elements to make a new seed triangle, if all elements have been used, stop. Find a set of fourth elements which together with the three defined by the triangle bound spheres inside the object. If no candidate fourth elements are found, repeat this step.
4. Sort the list of possible elements in order of increasing distance of the sphere centre from the previous critical point.
5. Discard any candidates where other elements intersect the sphere, i.e. the distance to some element is less than the radius of the sphere bounded by the three seed elements and the candidate fourth element.
6. If any fourth elements remain, pick the closest one and form a new tetrahedron. If any triangles already exist then merge the existing triangle into the new tetrahedron, if possible. Add any new triangles bounding the tetrahedron to the list of triangles to be processed.
7. Repeat from Step 2.

The tetrahedral structure is built up during the process of calculating critical points. The actual MAT surface is the 'dual' of this structure.

For a simple 100x70x50 block (shown in Figure 3) the algorithm works as follows.

The relevant elements to define the MAT surface are the boundary faces, labelled f0...f5. A convex vertex (e.g. vertex v0) is chosen as the start point and three (here there are only three) faces at the vertex are chosen as defining the seed triangle, here faces 0, 4 and 5, Figure 3b. The triangle is placed on the list to be processed. This triangle is then selected as a potential base for a new tetrahedron and a fourth element sought. Each of the other three elements, faces 1, 2 and 3, are combined with faces 0, 4 and 5 to find the critical point, defined logically in the datastructure as a tetrahedron. Face 1 is chosen to make the tetrahedron (Figure 3c) and the three new triangular triangles [0,1,4], [1,4,5] and [0,1,5] are placed on the list to be processed. The next triangle, [0,1,4] builds a tetrahedron with face 3 (Figure 3d) giving the new triangles [0,3,4], [1,3,4] and [0,1,3] which are also placed on the list. The next triangle, [1,4,5] is selected. Here no fourth element is found. Triangle [0,1,5] builds a tetrahedron with face 2 (Figure 3e) giving new triangles [0,2,5], [1,2,5] and [0,1,2]. Triangles [0,3,4] and [1,3,4] do not build tetrahedra. Triangle [0,1,3] builds a tetrahedron with face 2 (Figure 3f), forming new triangles [0,2,3] and [1,2,3]. The third triangle, [0,1,2] already exists so tetrahedron 2 and tetrahedron 4 are connected through this triangle. Triangles [0,2,5] and [1,2,5] do not build tetrahedra. When triangle [0,1,2] is encountered on the list it is noted that it has already been incorporated into a second tetrahedron so it is not processed. No other triangle builds a tetrahedron. The incomplete tetrahedra, i.e. where no fourth element is found, are preserved as they generate the so-called wings of the MAT.

After processing the following complete tetrahedra have been defined:

- **Tetrahedron 0: [f0,f1,f4,f5] critical point position (25,-10,0)**
- **Tetrahedron 1: [f0,f1,f4,f3] critical point position (-25,-20,0)**
- **Tetrahedron 2: [f0,f1,f2,f5] critical point position (25,10,0)**
- **Tetrahedron 3: [f0,f1,f2,f3] critical point position (-25,10,0)**

And the following incomplete tetrahedra and their corresponding original object vertices have been defined:

Tetrahedron 4: [f0,f4,f5,-1] vertex 0	**Tetrahedron 8: [f0,f2,f3,-1] vertex 2**
Tetrahedron 5: [f1,f4,f5,-1] vertex 4	**Tetrahedron 9: [f1,f2,f3,-1] vertex 6**
Tetrahedron 6: [f0,f2,f5,-1] vertex 1	**Tetrahedron 10: [f0,f3,f4,-1] vertex 3**
Tetrahedron 7: [f1,f2,f5,-1] vertex 5	**Tetrahedron 11: [f1,f3,f4,-1] vertex 7**

3.2 Geometric calculations

It can be seen from the basic algorithm that geometric calculation of the MAT involves the determination of spheres touching four arbitrary boundary elements. The centre points of such spheres will be the vertices of the MAT structure. Each of the boundary elements face and edge of the object is a finite portion of the corresponding infinite surface or curve. For geometric calculations we use the infinite entities (here straight line or plane) and check whether the touching points are on the finite bounding parts of the elements.

Relations between the centre point **C**, the radius r of the sphere and the geometric parameters of each type of touching element can be summarised as follows:

1. For a sphere touching a plane with normal **n** and point **P**:

$$(\mathbf{C} - \mathbf{P})\mathbf{n} - r = 0$$

2. For a sphere touching an edge with direction vector e and point **P**:

$$(\mathbf{C} - \mathbf{P})^2 - [(\mathbf{C} - \mathbf{P})\mathbf{e}]^2 - r = 0$$

3. For a sphere passing through a vertex **P**:

$$(\mathbf{C} - \mathbf{P})^2 - r^2 = 0$$

When performing the above algorithm, after four boundary elements have been selected a system of equations is assembled. This is a set of equations of the above types corresponding to each of the four selected boundary elements. Solution of the system of equations provides the four unknowns, i.e. the three coordinates of the sphere centre and the sphere radius r.

As can be seen, two of the above equations are quadratic, consequently a highly nonlinear system of equations must generally be solved. We may simply assemble the four equations of the above types and use a general purpose nonlinear equation solver. This way is followed by Reddy and Turkiyyah [ReTu95].

Non-linear system solving may cause many problems. First, a good starting point must be given for the iterative solution. The iteration may be slow, or divergence may even occur. In addition to this we need all real solutions in order to select the geometrically relevant one. All these aspects make the use of general nonlinear solvers a cumbersome, unstable and slow process.

Investigating the above equations shows that the systems of equations for a sphere bounded by four planes is linear. Moreover, other conbinations of the equations can be found where the solution can be derived analytically without solving a nonlinear system of equations.

We have carefully analysed the above systems of equations from analytic and from geometric points of view. It can be concluded that eleven of the possible fifteen combinations can be solved by analytic tools. In the remaining four cases specific singular geometric situations can be detected when again analytic solution is possible. Thus the use of a nonlinear system solver can in this way be strongly repressed. This results in a stable and fast procedure for handling a crucial geometric problem in medial surface computation.

3.3 MAT construction from the tetrahedral structure

When the tetrahedral structure is built the tetrahedra, triangular facets and struts are linked into simple lists. To construct the MAT the list of tetrahedra is scanned and vertices created for each (if wing edges are not required then vertices are only created for complete tetrahedra). The triangle list defines the edges of the MAT. Each triangle lies between two tetrahedra and creating the edges involves linking the vertices corresponding to these tetrahedra which were created in the first step of the conversion process. If wings are not required then only complete tetrahedral vertices are connected. Finally the struts are used to define faces in the MAT. Each strut is surrounded by an ordered list of triangles which defines the order of the edges in the MAT face loops.

The next step is to merge associated face loops. This is done by comparing struts. If two struts have the same start and end elements then the corresponding edge loops are checked to see if one lies inside the other. If they do, then the loops are merged into a single face. Finally zero-length edges are removed.

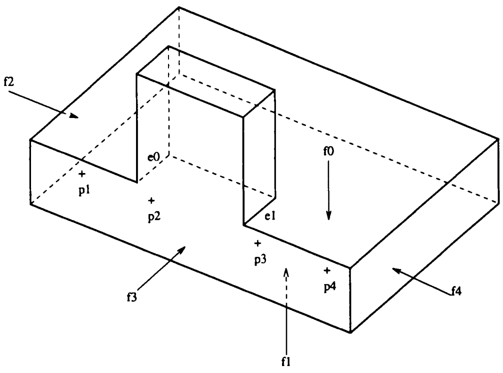


Figure 4 Multiple choice fourth elements

4 SPECIAL CASES

4.1 Multiple fourth element choices

The Reddy/Turkiyyah algorithm assumes that only one fourth element exists which builds a tetrahedron on any facet. Unfortunately there are some occasions on which this condition is violated. In Figure 4 the points p1-p4 and corresponding tetrahedra are:

Point p1: [f0,f1,f3,f2] Point p2: [f0,f1,f3,e0] Point p3: [f0,f1,f3,e1] Point p4: [f0,f1,f3,f4]

All these tetrahedra have a common triangle: [f0,f1,f3], so care has to be taken to sort out such problems. Tetrahedra with common facets form pairs and so the critical points corresponding to these tetrahedra can be sorted by parameter value along the curve defined by the three corner elements of the common triangle in a manner analogous to that of sorting intersection points along an intersection curve in Boolean set operations.

4.2 Regular objects and 4+ element critical points

The MAT calculation method does not work well for regular objects. For the object shown in Figure 3 the MAT vertices are defined by four elements and the edges by three elements. For the block in Figure 1 with square section there are multiple (4) coincident MAT points at each end, so the end vertices are defined by 5 elements and the edge by four elements. For a cube multiple coincident vertices are defined by different combinations of the six faces. The MAT surface consists of a single vertex defined by six elements.

Similar problems occur even with objects which appear irregular. At vertices where there are two or more adjacent concave edges meet then there are at least five entities which define the same MAT point. In this case the vertex, the two adjacent edges and their common face form a dependent set which determines a critical point with a fifth element. Multihedral nodes and multisided facets need to be included in the data structure to represent the MAT dual structure properly. Tetrahedra represent the minimum nodes but are not always suitable for decomposing more complicated cases. It is certainly possible to calculate some multisided facets as an initial step but it is not always easy to determine general n-sided internal nodes.

5 MULTIPLE START POINTS

It is clear from the example given of MAT surface generation for a block that many triangles are encountered which do not lead to complete tetrahedra because no fourth element is found. In the example of the cube, above, four triangles have a tetrahedron on both sides, leaving eight 'terminal' triangles where no fourth element is found. These are the incomplete tetrahedra listed in the example. Finding multiple start points involves determining the 'external' triangles first. For the block example, above (Figure 3), this works as follows:

All external triangles are produced, i.e. [0,2,3], [1,2,3], [0,2,5], [1,2,5], [0,3,4], [1,3,4], [0,4,5], [1,4,5] (in some order) and placed on the list to be processed. [0,2,3] is picked from the list, and face 1 is found to define a sphere giving tetrahedron [0,1,2,3]. Triangles [0,1,2] and [0,1,3] are new, but triangle [1,2,3] has already been defined in the initial step. The next triangle on the list is [1,2,3], but since it has already been incorporated into a tetrahedron it is ignored. The next triangle is [0,2,5] and face 1 is again found as the fourth element of a tetrahedron, [0,1,2,5]. Triangle [0,1,5] is new but triangle [1,2,5] was defined in the initial step and triangle [0,1,2] was defined from the first tetrahedron (so the tetrahedra are joined through this triangle. The next triangle [1,2,5] has been processed as triangle [0,3,4] is processed and face 1 found as the fourth element giving tetrahedron [0,1,3,4]. Triangle [0,1,4] is new but triangle [1,3,4] was defined in the initial step and triangle [0,1,3] was defined by the initial tetrahedron. The final triangle to be processed is triangle [0,4,5] building tetrahedron [0,1,4,5]. Here all triangles already exist, triangle [1,4,5] in the initial step, triangle [0,1,5] from the second tetrahedron and triangle [0,1,4] from the third tetrahedron. All other triangles on the list are found to have been processed so the construction terminates.

6 MAT REFINEMENT

MAT surface refinement is the process of adjusting the MAT surface produced from a planar polyhedron to the MAT surface of the original object. Refinement depends on being able to relate topological elements in the facetted model back to their corresponding elements in the original model. Faces always derive from faces, edges can derive from edges in the original model, or from faces if they were produced in the facetting process. Vertices can derive from vertices, or from edges if the edge were curved and had to be subdivided, or from faces if they were produced when facetting the face.

Consider the curved block shown in Figure 5. The original model and its faces is shown in Figure 5a, and the corresponding facetted model in Figure 5b. The faces and edges used to produce the MAT surface are labelled. The correspondences between elements in the facetted model and elements in the original model are:

Facetted model:	f0	f1	f2	f3	f4	f5	f6	f7	f8	f9	e0	e1
Original model:	F0	F1	F2	F3	F4	F4	F4	F5	F5	F5	F4	F4

Only e0 is mentioned here because only this edge is used in the construction of the MAT surface.

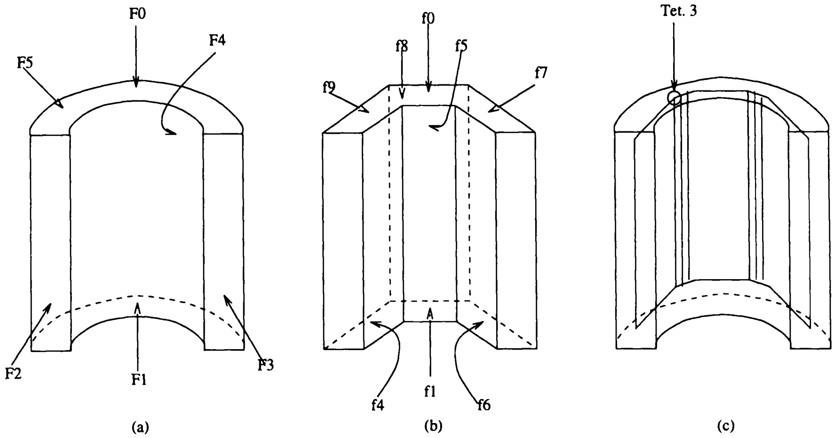


Figure 5 Identification of artefact tetrahedra

Considering one of the tetrahedra adjacent to the concave edge, tetrahedron 3 in figure 5c, the elements and facets of this are:

Tetrahedron 3: f0,f4,f9,e0 Corresponds to F0,F4,F5,F4

triangle 1: f0,f4,f7 (corresp. F0,F4,F5) connects to tetrahedron 1

triangle 2: f0,f4,e0 (corresp. F0,F4,F4) produces wing edge

triangle 3: f0,f7,e0 (corresp. F0,F5,F4) connects to tetrahedron 5

triangle 4: f4,f7,e0 (corresp. F4,F5,F4) connects to tetrahedron 4

Here it is clear the edges from triangle 2 and triangle 4 are products of the facetting, the wing edge from triangle 2 is unnecessary and the edge from triangle 4 is internal to a face in the 'true' MAT surface. The edges from triangles 1 and 3 are real edges, but since tetrahedron 3 is a degenerate terahedron they should be merged.

There are special cases, though, where care has to be taken not to remove essential elements so an extra check has to be included. In a sphere, for example, the central point is defined many times, by the method, from combinations of four triangles which all derive from the same original surface.

There are two parts to the refinement process, geometrical refinement and topological cleansing.

Geometrical refinement

The geometry of the MAT produced by the algorithm is only an approximation to the geometry of the MAT of the original object, so a process of geometrical refinement has to be undertaken to obtain the correct geometry. The facetted object MAT geometry can be used as an approximation which is refined by relaxation. The approximate MAT provides good starting points because the facetting makes the MAT geometry almost planar.

Topological cleansing

This is simply the process of removing redundant edges and vertices from the MAT model. The first process is to remove redundant edges from the facetted MAT surface by traversing all edges and checking the correspondences as outlined above. The next step is to delete the unnecessary nodes or vertices from the MAT by deleting those which correspond to tetrahedra dependent on only three elements of the original.

Space does not allow more detail to be presented here. Instead this topic will be dealt with elsewhere.

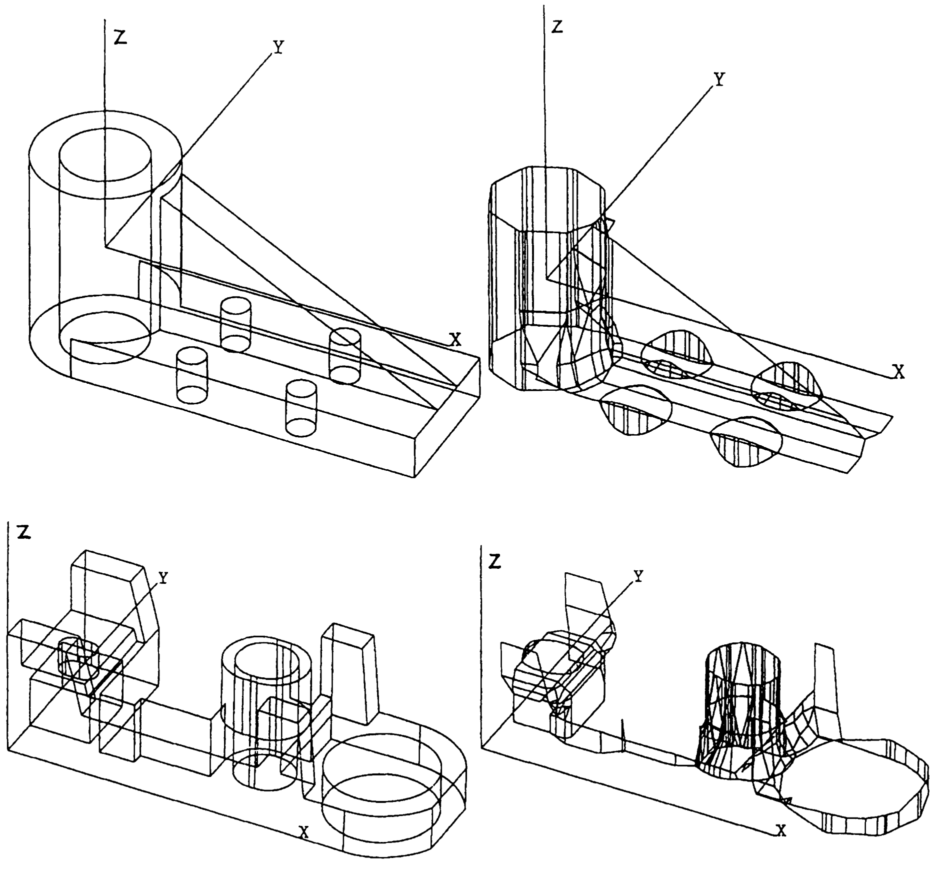
7 CONCLUSIONS

The methods for improving efficiency and reliability presented here have been tested on many examples, including complicated examples, see Figure 6. Generally the results seem good, but the problems caused when more than four elements define a critical point remain to be solved. The approximate MAT produced is good enough for many purposes but also offers a suitable starting point for refinement, as outlined here.

8 ACKNOWLEDGEMENTS

The authors would like particularly to thank Tigyi Attila for his dissertation work on geometric case analysis supervised by Gabor Renner. We would also like to thank the other members of the Geometric Modelling Laboratory for their support.

The research described in this paper was supported by Copernicus grant RECCAD 94-1068 and by the National Science Foundation of the Hungarian Academy of Sciences grants no. 16420, 16502.



Original object MAT surface

Figure 6 MAT examples

REFERENCES

Brandt, J. W. (1991), "Theory and Applications of the Skeleton Representation of Continuous Shapes", Ph.D.Dissertation, University of California, Davis.

Brandt, J. W. (1992), "Describing a solid with the three- dimensional skeleton", in Proceedings of The International Society for Optical Engineering volume 1830, Curves and Surfaces in Computer Vision and Graphics, ed. J.D.Warren, pp 258-269, SPIE, Boston, Mass.

Chiang, C.-S. (1992), "The Euclidean Distance Transform", Ph.D. dissertation, Purdue University.

Gursoy, H. N. (1989), "Shape Interrogation by Medial Axis Transform for Automated Analysis", Ph.D. dissertation, M.I.T.

Hoffmann, C. M. (1994), "How to construct the skeleton of CSG objects", Proc. IMA conference The Mathematics of Surfaces IV, ed. Bowyer, pp. 421-437, Oxford University Press.

Lavender, D., Bowyer, A., Davenport, J., Wallis, A., Woodwark, J. R. (1992), "Voronoi diagrams of set-theoretic solid models", IEEE Computer Graphics and Applications, volume 12 no. 5, pp. 69-77.

Montanari, U. (1969), "Continuous skeletons from digitized images", Journal of the Association of Computing Machinery, vol. 16, number 4, pp 543-549.

Nackmann, L. R., Pizer, S. M. (1985), "Three-dimensional shape description using the symmetric axis transform I: Theory", IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI volume 7 no. 2, pp 187-202.

Preparata, F. P. (1977), "The medial axis of a simple polygon", in Lecture Notes in Computer Science: Mathematical Foundations of Computer Science, eds. Goos and Hartmanis, pp. 443-450, Springer-Verlag.

Reddy, J., Turkiyyah, G. (1995), "Computation of 3d skeletons by a generalised Delaunay triangulation technique", Computer-Aided Design, volume 27 no. 9, pp. 677-694.

Scott, G. L., Turner, S. C., Zisserman, A. (1989), "Using a wave/ diffusion process to elicit the symmetry set", Image and Vision Computing, volume 7, pp. 63-70.

Sheehy, D. J., Armstrong, C. G., Robinson, D. J. (1995), "Computing the Medial Surface of a Solid from a Domain Delaunay Triangulation", in Proceedings of the Third Symposium on Solid Modeling and Applications, ed. C.M.Hoffmann and J.R.Rossignac, pub. ACM, pp 201-212.

Sherbrooke, E. C., Patrikalakis, N. M., Brisson, E. (1993), "Computation of the Medial Axis Transform of 3-D Polyhedra", in Proceedings of the Third Symposium on Solid Modeling and Applications, ed. C.M.Hoffmann and J.R.Rossignac, pub. ACM, pp. 187-199.

Srinivasan, V., Nackman, L. R. (1987), "Voronoi diagram for multiply connected polygonal domains", I: Algorithm, IBM Journal of Research and Development, volume 31, number 3, pp. 361-372.

Sudhalkar, A., Gursoz, L., Prinz, F. (1993), "Continuous skeletons of discrete objects", in Proceedings of the Second Symposium on Solid Modeling and Applications, ed. J.R.Rossignac, J.Turner and G.Allen, May 19-21.

Wolter, F.-E. (1992), "Cut locus and Medial axis in global interrogation and representation", Computer Aided Geometric Design.

Yu, X., Goldak, J. A., Dong, L., (1991) "Constructing 3D discrete medial axis", in Proceedings of the First Symposium on Solid Modeling Foundations and CAD/CAM Applications, ed. J.R.Rossignac and J.Turner, June 5-7.