ACM Symposium on Solid Modeling and Applications (2004)
G. Elber, N. Patrikalakis, P. Brunet (Editors)

Medial Axis Based Solid Representation

Amir Shaham
Tel-Aviv University

Ariel Shamir
The Interdisciplinary Center

Daniel Cohen-Or
Tel-Aviv University

Abstract

The medial axis (MA) of an object and medial axis transform (MAT) have many applications in solid modeling, computer graphics and other areas. Exact computation of MA is complex and various medial axis approximation algorithms were studied. One of the most successful is based on the computation of the Voronoi diagram of a set of sample points on the boundary of the object. Based on this method we present a novel representation of solids, which we call a pair-mesh. *The pair-mesh is a deformable manifold surface triangulation where each node deforms between a pair of vertices one on the MA approximation and one on the boundary. Consequently, it provides a continuous map between the inner Voronoi based structure and the boundary of the shape, encompassing the topological structures of them both. This representation can also be seen as a partitioning of the volume between the two, where each element in the partition is either a tetrahedron or a pyramid and includes vertices from both the MA approximation and the reconstructed boundary.*

Keywords: Medial-axis, Voronoi-diagram, Delaunay Triangulation, Solid-representation

1. Introduction

The medial axis (MA) of an object O in $\mathbb{R}^3$ is defined as the locus of points, which lie at the centers of all closed balls that are maximal in O (*medial balls*), together with the limit points of this locus. The properties of the medial axis (MA) and its close relation to the object make it a useful tool for many applications in solid modeling, computer graphics, motion planning and more [HK00, TH03, BTSG00, DWZ03, YBS03]. The medial axis transform (MAT) is defined by the MA itself and a function $R : \mathrm{MA} \rightarrow \mathbb{R}$ that assigns to each MA point the radius value of its sphere. It was shown that the object O can be extracted from $MAT(O)$ as the union of all the medial balls. Thus the boundary of O can be calculated as the boundary of the union of all medial balls (inverse MAT). Furthermore, an object O and it's MA have the same topology structure (the same homotopy type) and can continuously deform from one to the other (see e.g. [SPFE96, CCM97, Ede01]). Thus, the MAT can be used as a compact and effective representation for objects in $\mathbb{R}^3$.

The main motivation of this work is the definition of a solid representation with similar properties as the MAT. In practice however, both the computation of the medial axis and the MAT are extremely difficult and the results can become complicated even in simple cases. Numerous methods for approximating the MA have been proposed. Recently, several works were presented based on the computation of the Voronoi diagram of a set of sample points residing on the object's boundary (either by surface sampling or from range scans) [ACK01, DZ02]. The output of such approximation methods is a polygonal representation approximating the medial axis. It has been shown that the resulting approximation is guaranteed to be close to the true medial axis with some tolerance, and converges to it when the boundary sampling rate approaches infinity. Still, the resulting polygonal shape is unstable with respect to the model boundary and may include many excessive parts and spikes. This shape is non-manifold, extremely complex, and difficult to work with.

In [ACK01] the MA is approximated as part of the dual shape of the power diagram of a set of approximated medial balls. A different approach, used also in this work, is to approximate the MA directly as a subset of the Voronoi diagram applying some filtering criteria to remove excessive parts [DZ02]. Nevertheless, during the filtering process holes may occur in the medial axis, as well as the separation of the medial axis to non-connected parts. Furthermore, at the end of the process there are sites on the boundary that do not have corresponding elements on the MA anymore and the link between the MA and the boundary is lost. In [HBK02] this link is preserved simply by copying the original boundary mesh connectivity to the poles on the Voronoi based MA approximation. However the MA is represented by a two-side surface that may cut itself and the link between opposite sides of the boundary is lost.

In this paper we define a novel solid object representation, which we call the *pair-mesh*. The pair-mesh definition is based on the Voronoi diagram of sampled boundary points. This representation preserves the link between the boundary and the inner Voronoi structure, representing the skeleton of the shape. The end result of this process is a simple and effective representation of the solid, which is composed of both a medial axis approximation and the approximated boundary of the shape. Furthermore, the pair-mesh is in fact a parametric solid representation realizing the continuous map between the MA and the boundary.

The rest of the paper is organized as follows. We begin by reviewing related work. Next, in Section 3 we describe the pair-

mesh data-structure and its construction from the Voronoi diagram of the boundary sample points. Section 4 defines the filtering of the Voronoi creating a MA approximation and the method to reconnect it to the shape boundary to create a pair-mesh. We conclude in Section 5 and discuss future applications.

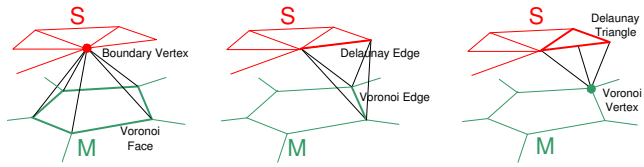
2. Related work

The complexity of the medial axis and its computation time prevent methods for the exact calculation of the medial axis such as for polyhedra in[CKM99]. Furthermore, an input 3D object O may be available only as a set of sample points instead of explicitly defined. Some works studied approximations of the MA using discrete space representations. In [ER99] the medial axis is calculated for polyhedron input by separating the calculation of geometric and symbolic parts. The symbolic representation is calculated using space subdivision. In [VO95] a medial axis for general objects in 3D space is calculated by hierarchical subdivision of the space. [FLM03] also uses spatial subdivision of the solid object to approximate the MA. The MA simplification (the θ SMA) is parameterized by a separation angle between the vectors connecting a MA point to the closest points on the boundary.

A subset of the Voronoi diagram of a sample point set in O was proved to approximate the MA with guaranteed accuracy. In [ACK01] the medial axis approximation is derived from the power diagram of a subset of Voronoi vertices. First the algorithm finds the inner and outer poles of each site (the Voronoi vertices farthest from the site). A power diagram is calculated using the poles with a weight that relates to their Voronoi balls (polar balls), i.e. the maximal ball of each pole that doesn't include other samples. The connectivity of the inner pole power diagram cells is used as an approximation of the MA. The Local Feature Size function, $LFS(x)$ at a point x on the model boundary W, is the minimum Euclidean distance from point x to any point of the medial axis [AB98]. A Set $S \subset W$ is an *r-sample* of W if no point p on W is farther than $r \cdot LFS(p)$ from a point of S. These definitions can guide us to find the required local sampling rate in the model boundary in order to have a good Voronoi based approximation.

In [DZ02] the medial axis is approximated as a subset of the original Voronoi diagram faces. The algorithm first finds for each site an estimated normal according to its poles. Then it finds the set of Voronoi edges that cut the orthogonal plane through the site, to get an umbrella (fan) of Delaunay triangles that are dual to these Voronoi edges. The Delaunay edges in the diagram are now checked against two tests. The angle test uses the angle between the Delaunay edge to any of its sites vertex's umbrella triangles normal-vectors. The ratio test uses the ratio between the Delaunay edge length and the largest circum-radius of the umbrella triangles (the distance to the farthest Voronoi edge). The Voronoi faces that are dual to the Delaunay edges that pass the tests were proved to approximate a subset of the MA. Nevertheless, after filtering, only the remaining faces are linked to their dual vertices on the boundary and all other boundary parts remain with no correspondence to the inner structure.

Volumetric partitioning and meshing are important tools in finite-element simulations. There are numerous works, which provide algorithms for creating a quality FEM mesh for an object's boundary, and some also rely on skeleton computations [TKGC03, RBO02, BWS03]. Our work does not target such quality volume meshing of elements and we concentrate more on the mapping structure between a quality MA approximation and the shape's boundary. Nevertheless, such mapping could in the future be the basis on which quality meshes can be produced.



(a) (b) (c)

Figure 1: Different elements types of the solid mesh: (a) Face-Vertex pyramid (b) Edge-Edge tetrahedron (c) Vertex-Face tetrahedron.

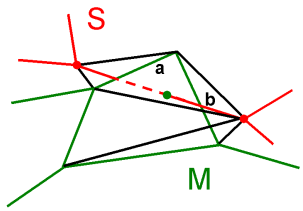


Figure 2: An example of a situation where an edge-edge tetrahedra (a-b) folds on a face-vertex tetrahedra.

3. Voronoi based pair mesh

In this section we will introduce our data structure, the *pair-mesh*, using a subset of the Voronoi diagram of the surface point set. We start with a watertight surface S reconstructed from the model samples using the tcocone algorithm from [DG03]. This surface, representing the boundary of the object, is a subset of faces from the 3D Delaunay triangulation of the point set. As the reconstructed surface is a restricted Delaunay triangulation of the sample points, it has a direct connection to its dual graph, the Voronoi diagram of the sample points. Hence we combine S with a subset of the Voronoi elements V of the same point-set to create a solid representation of the object. This solid representation is composed of a set of pyramids and tetrahedra elements. Each element has some vertices on the boundary surface S and some on the Voronoi subset V.

In the following description we assume that the sample point-set are in general position, thus each Voronoi vertex is closest to 4 samples and each edge to 3. We will use the following notations and definitions. A Delaunay element on S will be called a *boundary* vertex, edge or face. As we start from a surface that is made from Delaunay elements, the restricted Voronoi elements are easily found as the dual to the surface elements. We call the elements dual to the boundary faces and edges, *cut-edges* and *cut-faces* respectively. The water tight boundary surface imposes a partition of the Delaunay tetrahedra graph to inner and outer parts (this is directly enforced by the tcocone algorithm). We thus define every Voronoi vertex that is dual to an inner Delaunay tetrahedron as an *inner* Voronoi vertex. A Voronoi edge or face in which all vertices are inner-vertices is defined as an *inner* element. Duality implies that if the sub-graph of inner tetrahedra is face connected, the same is true for the edge connected graph of the dual Voronoi vertices.

(a) (b) (c) (d) (e) (f)

Figure 3: The pair-mesh of a cube and a head figure: (a,d) Only the face-vertex tetrahedra (b,e) Adding the edge-edge tetrahedra (c,f) Adding the vertex-face tetrahedra. Note that the surface mesh has been altered slightly as a result of local correction operators that prevent flipped tetrahedra.

The inner edges and vertices that are a part of a cut face are called *inner-cut edges* and *inner-cut vertices* respectively.

With these definitions we can proceed to the construction of the solid representation. Each Voronoi inner-face in V is connected to the Voronoi site of the face to create *face-vertex* pyramids. Every pair of inner-cut edge and the boundary edge that is dual to the Voronoi cut-face that includes it create an *edge-edge* tetrahedron. Every Delaunay triangle on the boundary S and the inner-cut vertex at the end of its dual Voronoi edge create a *vertex-face* tetrahedron (see Figure 1 and Figure 3). Note that all the face-vertex pyramids can be tetrahedralized by triangulating the Voronoi face, and connecting the triangles to the boundary site. For simplicity, we will sometime refer to all these volume elements as tetrahedra although in terms of implementation there is no reason to actually subdivide the face-vertex pyramid into tetrahedra. The sub-elements of the tetrahedra will be classified as *Voronoi* for Voronoi elements, *Boundary* for boundary elements, and *interim* otherwise.

Given that the inner Voronoi subset V is connected and the boundary is water tight it is easy to verify that all the inner volume is covered by the set of tetrahedra and pyramids, and every tetrahedron or pyramid element is connected either to other elements or to a boundary face in all four faces. However, there might be cases in which this construction scheme leads to overlapping elements as some tetrahedra may fold over others (Figure 2). Hence, there may be faces that have two tetrahedra on the same side geometrically. The algorithm finds folded tetrahedra using volume calculation as described below.

Every surface vertex is a Voronoi site and thus resides inside its Voronoi cell, leading to correct orientation of the face-vertex pyramids. However, other types of tetrahedra might be flipped. Such cases may occur especially when the inner part of the Voronoi diagram is close to, or crossing, the reconstructed model surface. Local manipulation of the structure can solve this problem. For example by moving Voronoi vertices which are outside toward the inside of the model without breaking the tetrahedra orientation constraints. Another case is when a Delaunay edge on the surface doesn't cross it's dual Voronoi cut-face. In such case, we find the closest point to the Delaunay-edge on the dual cut-face edge. Then, we choose a point slightly inside the face to ensure a positive volume of the resulting tetrahedra and replace the edge with two edges joined at this point. In this way the edge-edge tetrahedron is replaced by two tetrahedra that are properly connected to the adjacent face-vertex tetrahedra of the edge vertices. Consequently, a Delaunay triangle

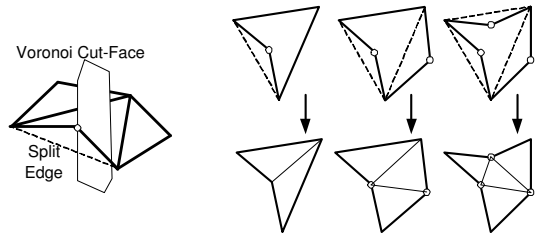
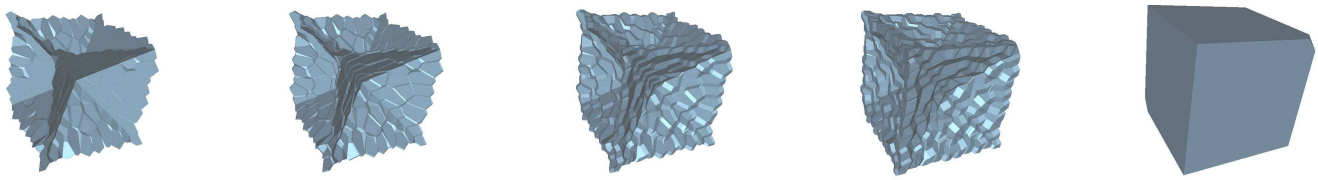


Figure 4: Left: splitting an edge of a boundary triangle. Right: some examples of Delaunay triangles with one, two and three edges split, and their polygon triangulations.

including such edges can be deformed into a polygon in which all of its vertices are either Voronoi sites (original Delaunay vertices) or lie inside one of the three faces near its dual Voronoi edge. This polygon can be triangulated as shown in Figure 4, and up to four vertex-face tetrahedra are created by this triangulation instead of one. Obviously these manipulations may lead to slight perturbations on the reconstructed surface.

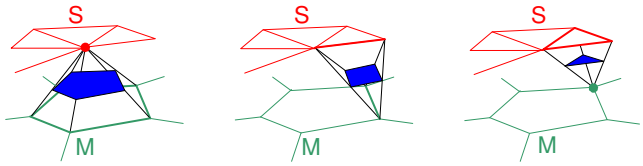
As mentioned before, all the defined tetrahedra have vertices both on S and on V and thus form a scaffolding envelope around V as the subset of the Voronoi diagram and form a partition of the solid volume that is enclosed within S.

Finally we define what we call the *pair-mesh*. Each node in the pair-mesh is an interim edge, which in fact connects a pair of vertices one from the boundary S and one from V. Each edge in the pair-mesh is defined by an interim face and each face in the pair-mesh is a pyramid or tetrahedra. In situations where the boundary is made of several manifold components, the same topological structure will emerge in the pair-mesh. This is still true even if the Voronoi subset V is only edge connected. We can refer to each pair-mesh node as a parametric line segment with a parameter $t \in [0,1]$, when $t = 0$ the node resides on the inner vertex of V and when $t = 1$ the node resides on the boundary vertex. The pair-mesh thus defines a parametric mesh deforming from an envelope around the Voronoi subset V, to the input boundary surface S (see Figure 5). Furthermore, for every $t \in [0,1]$, the mesh is composed of faces that are a cut in the related pyramid or tetrahedra. These faces are planar convex polygons. In the case of a face to vertex pyramid the related cut-face is parallel to the convex base of the pyramid and thus is a convex planar polygon similar to the base Voronoi face (Figure 6(a)). A similar argument holds for the vertex to boundary face



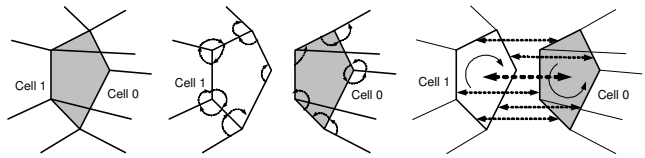
(a) (b) (c) (d) (e)

Figure 5: Deforming from the inner subset of the Voronoi V to the boundary surface S using a parameter from $t = 0$ to 1 on the pair-mesh.



(a) (b) (c)

Figure 6: Three Different convex iso-faces for the three element types of the solid mesh.



(a) (b) (c)

Figure 7: Illustration of the non-manifold DCEL data structure elements: (a) Two adjacent cells (b) Each cell is a standard manifold DCEL with some of the usual links shown (c) The new half-face and half-edge new links between cells.

tetrahedra: the cut-faces are triangles similar to the boundary Delaunay triangles (Figure 6(c)). For an edge to edge tetrahedron the cut face is a quadrilateral. However, each two opposite segments in the face are parallel to the Voronoi edge or to the boundary Delaunay edge. Thus, the shape is a parallelogram and therefore planar and convex (Figure 6(b)).

3.1. Data Structure

We use a single data structure to represent the tetrahedra, their connectivity, the related model boundary S and the Voronoi diagram subset V. We define an extension of the DCEL data structure [dBvKOS97] for the non-manifold case of this composite structure. Each face in the structure is represented as two half-faces with a dual pointer from one to the other. Each half-face polygon is represented as a doubly connected edge list similar to the original DCEL structure. However, each half edge includes an additional pointer to its opposite half edge in the dual half-face. Each half-face edge list is right handed oriented with respect to the normal of its side. This way we can iterate on edges around a half-face polygon and also iterate on faces around a non manifold edge. Moreover, in the case of Voronoi diagram cells or in our pyramid structure the entire data set is represented as a set of manifold meshes (one for each cell) that are connected within them using the ordinary DCEL pointers, and are connected between them with the new half-face and half-edges pointers (see Figure 7).

Using this structure every face-connected part of the inner-subset of the Voronoi diagram V is represented with a manifold mesh that wraps it. However, the dual pointers provide the important link between the two sides of V for the Voronoi faces, edges and vertices. Another advantage of this structure is the simple method to check for folded pyramids. Since the non-manifold DCEL structure imposes consistent half-edge orientation, we can use the edge order as a reference. Using one of the base faces and its opposite vertex, a folded pyramid would have a negative volume. The enhanced DCEL data structure also restricts the pair-mesh validity as a partition of the volume. This is because all the elements (pyramids or tetrahedra) must have neighbor elements or neighboring boundary face and all of them are oriented properly.

4. Medial Axis based pair mesh

In general, the inner Voronoi diagram subset V of the sample points is not an approximation of the solid object medial axis. In order to create a medial axis based representation we have to build it on top of a medial axis approximation. Furthermore, using all the inner subset of the Voronoi diagram leads to a structure with space complexity of the order of the 3D Voronoi diagram. For complex solids, this may lead to impractical time complexity for operations on this structure. Using only the subset of the Voronoi diagram that approximate the MA reduces the structure size substantially. In order to approximate the medial axis as a Voronoi subset we follow a similar approach to the filtering scheme of [DZ02]. When a surface mesh is given the vertices normals are estimated using discrete curvature normal [MDSB02] otherwise we use the original scheme of the pole direction. After the filtering process we remain with a subset of the Voronoi which approximates the MA, which we will denote as M (See Figure 8).

In contrast to the inner Voronoi subset V and the tcocone boundary surface reconstruction of the previous section, using the filtered MA approximation we no longer have a direct mapping between the boundary elements of S and the inner Voronoi elements of M. The smaller subset of the Voronoi elements in M are mapped only to the sites of which part of their Voronoi cell is included in M. Therefore, we must extend the scheme used for the Voronoi based structure to construct the pair-mesh based on the MA approximation M.

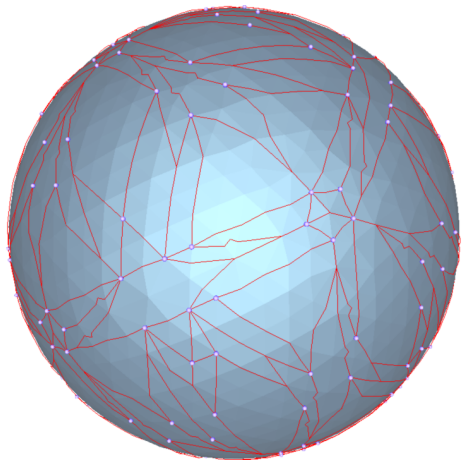
Each Voronoi face in M is connected, as before, to its site on the boundary with a face to vertex convex pyramid. Each Voronoi edge in the M is closest, for general position input, to 3 sites. The Delaunay edges between pairs of such sites are dual to Voronoi faces, which were removed in the filtering process. However, in this case

Figure 8: The inner Voronoi diagram subset V of the head figure and the set of faces that are removed (yellow) in order to arrive at an approximated medial axis M (blue).

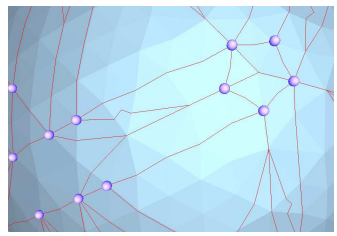
these Delaunay edges are no longer edges of the boundary surface. As an extension to the former structure, we find a path of edges on the surface between such pairs of sites. We take the shortest path between the pair of sites on the boundary as a reasonable map to the Voronoi edges. For each of the edges in the path we create an edge to edge tetrahedra, and thus create a chain of edge to edge tetrahedra between the face to vertex pyramids of the sites. This phase of the algorithm is not guaranteed to succeed as will be described below. The edge paths must not cross each other, although they may share boundary edges. An additional restriction is that all the tetrahedra created must have a positive volume with respect to the orientation of their half edges (no flipped tetrahedra).

After finding all pairs of sites that should be linked by an edge path there are two cases. The first case is a manifold edge that only one of its Voronoi faces was removed. Such edge will have one pair of sites to be linked. The second case is a skeletal edge that has two adjacent faces that were removed and thus two site pairs. As we must use only boundary edges that form a positive volume (valid) tetrahedron with the Voronoi edge, we search for a possible path in a small sub-mesh neighborhood around the segment connecting the two sites. Edges that create valid tetrahedron are gathered to a sub graph. We now perform a Dijkstra single source shortest path algorithm on the sub graph with the edges length as weights. If the sub graph is connected such path will be found. In order to increase the possibility for path existence we apply a subdivision on the triangles connecting the centroid to the triangle vertexes and edge centers. After all the paths are successfully found, we check that there are no two paths that cross each other (see Figure 9(b)).

At the end of this phase, the boundary is segmented into a set of closed polygons. Each polygon includes key vertices, which are one of the Voronoi sites corresponding to a Voronoi face in M, and creating the face vertex pyramids. The paths of edges between these key vertices belong to edge-edge tetrahedra that sit on top of edges in M (Figure 9). Hence, for each set of boundary faces enclosed by one of these polygons, the visible part of M from this set is a single Voronoi vertex on M. This vertex is now connected to the boundary faces using a vertex-face tetrahedra to complete the pair-mesh solid representation.



(a)



(b)

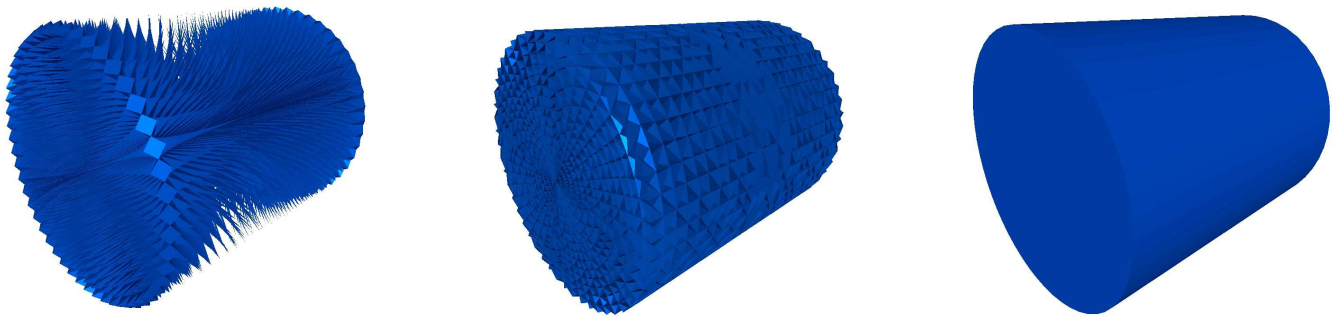
Figure 9: (a) Example of the key vertices forming the pyramids with the medial axis faces (purple balls), and the paths of edges found between them (red). (b) Non-crossing paths found using Dijkstra algorithm with some additional triangles subdivision.

The most difficult of the above construction phases is finding the edge pyramids. The first problem that may arise follows from the fact that M may not have the same topology structure as the solid. For example, some threshold values in the filtering stage may lead to the creation of holes in M. Such holes may lead to the construction of an edge path from one side of the model to the other, crossing other edge paths. M may even have a different number of components from the solid object. Another problem may occur when we have complex 'bays' in M. Skeletal edges in these bays may only be connected to the boundary with flipped tetrahedra. All these may prevent the creation of a valid pair-mesh structure. We currently demonstrate our methods for input sets where such cases do not occur. Nevertheless, these cases can in fact be seen as restrictions on the subset of Voronoi elements that approximate the medial axis. As the valid structure presents a continuous map between the MA approximation and the boundary, finding this structure enforces the MA to have similar topology structure as the input object, and thus improves approximation.

Figures 10, 11 and 12 illustrate the different elements which create the solid pair-mesh representation. Figures 13, 14 and 15 demonstrate the iso-value manifold meshes between a wrapped mesh around the medial axis approximation until the object's boundary mesh.

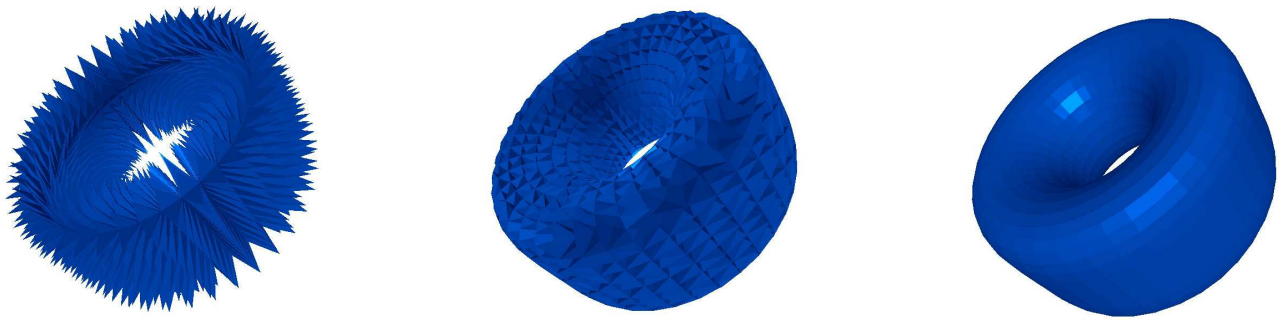
(a) (b) (c) (d)

Figure 10: (a) The approximated medial axis of a ball enlarged by a factor of around 100 (b) The face-vertex pyramids of the ball: note there are much fewer such pyramids than in the case of the non-filtered Voronoi. (c) The paths of edge-edge tetrahedra added between the key vertices of the boundary (d) The solid ball with all tetrahedra.



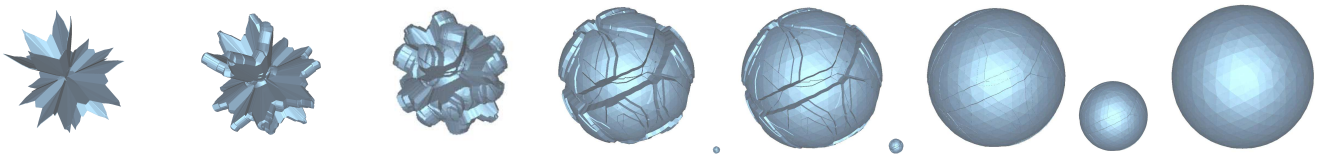
(a) (b) (c)

Figure 11: (a) The face-vertex pyramids of a cylinder (b) The paths of edge-edge tetrahedra (c) The solid cylinder with all tetrahedra.



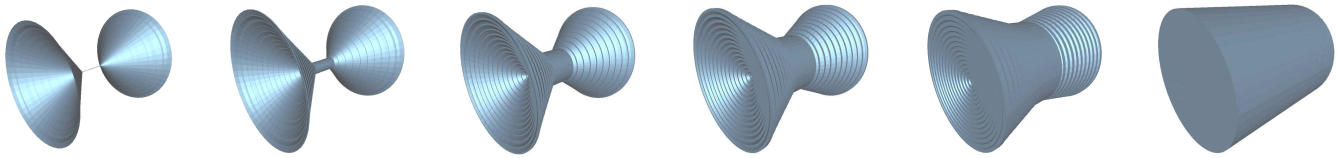
(a) (b) (c)

Figure 12: (a) The face-vertex pyramids of a wheel model (b) Including the edge-edge tetrahedra (c) The solid wheel with all tetrahedra.



(a) (b) (c) (d) (e) (f) (g)

Figure 13: The iso mesh interpolating between the medial axis and the boundary of the ball at various t values: (a)0.001, (b)0.003, (c)0.01, (d)0.05, (e)0.1, (f)0.5, (g)1.0. Note that the small value balls (a)-(f) were scaled. The non scaled sizes can be seen to the right of (d)-(f) balls, all others are too small to be seen in scale.



(a) (b) (c) (d) (e) (f)

Figure 14: The iso mesh interpolating between the medial axis and the boundary of a Cylinder at the following t values: (a)0.001, (b)0.1, (c)0.3, (d)0.5, (e)0.75, (f)1.0.

(a) (b) (c) (d) (e) (f)

Figure 15: The iso mesh interpolating between the medial axis and the boundary of a Wheel model at the following t values: (a)0.001, (b)0.1, (c)0.3, (d)0.5, (e)0.75, (f)1.0.

5. Conclusions

In this paper, we have developed a new approach to represent the volume of an input solid by an approximated skeleton based 3D mesh. This representation uses recent results on mesh reconstruction as well as MA approximation algorithms based on the Voronoi diagram of sample points. The result volume mesh is a parametric map between the model surface and its skeleton, and its vertex pair-mesh connectivity can continuously deform between the two, realizing a parameterizations of the solid volume.

It is interesting to distinguish between the real MAT and our parametric map. The MAT maps the MA to the boundary using a scalar function, and the object can be retrieved using the union of the balls centered on the MA. In our pair-mesh the mapping is done using *vectors* (i.e. size and direction) which associate points on the MA to points on the boundary (and vice verse). This map can be many to one in any direction, but not many to many.

In ongoing work we explore the creation of the pair mesh structure using more robust approaches. We are also looking at several applications for using the volume partition structure. For example, for skeleton-based meshing, for skeleton-based animation or free-form deformation. The volume cells around the rigid skeleton can restrict the deformation leading to a well behaved deformation of the volume.

References

[AB98] AMENTA N., BERN M. W.: Surface reconstruction by voronoi filtering. In *Symposium on Computational Geometry* (1998), pp. 39–48.

[ACK01] AMENTA N., CHOI S., KOLLURI R. K.: The power crust. In *Proceedings of the sixth ACM symposium on Solid modeling and applications* (2001), ACM Press, pp. 249–266.

[BTSG00] BLANDING R. L., TURKIYYAH G. M., STORTI D. W., GANTER M. A.: Skeleton-based three-dimensional geometric morphing. *Computational Geometry: Theory and Applications 15*, 3 (February 2000), 129 – 148.

[BWS03] BEALL M., WALSH J., SHEPHARD M.: Accessing cad geometry for mesh generation. In *Proceedings of the 12th International Meshing Roundtable* (September 2003), pp. 33–42.

[CCM97] CHOI H., CHOI S., MOON H.: Mathematical theory of medial axis transform. *Pacific Journal of Mathematics 181*, 1 (1997), 57–88.

[CKM99] CULVER T., KEYSER J., MANOCHA D.: Accurate computation of the medial axis of a polyhedron. In *Proceedings of the fifth ACM symposium on Solid modeling and applications* (1999), ACM Press, pp. 179–190.

[dBvKOS97] DE BERG M., VAN KREFELD M., OVERMARS M., SCHWARZKOPF O.: *Computational Geometry: Algorithms and Applications*. Springer Verlag, 1997.

[DG03] DEY T. K., GOSWAMI S.: Tight cocone: a watertight surface reconstructor. In *Proceedings of the eighth ACM symposium on Solid modeling and applications* (2003), ACM Press, pp. 127–134.

[DWZ03] DEY T. K., WOO H., ZHAO W.: Approximate medial axis for cad models. In *Proceedings of the eighth ACM symposium on Solid modeling and applications* (2003), pp. 280 – 285.

[DZ02] DEY T. K., ZHAO W.: Approximate medial axis as a voronoi subcomplex. In *Proceedings of the seventh ACM symposium on Solid modeling and applications* (2002), ACM Press, pp. 356–366.

[Ede01] EDELSBRUNNER H.: *Geometry and Topology for Mesh Generation*. Cambridge University Press, May 2001.

[ER99] ETZION M., RAPPOPORT A.: Computing the voronoi diagram of a 3-d polyhedron by separate

computation of its symbolic and geometric parts. In *Proceedings of the fifth ACM symposium on Solid modeling and applications* (1999), ACM Press, pp. 167–178.

[FLM03] FOSKEY M., LIN M. C., MANOCHA D.: Efficient computation of a simplified medial axis. In *Proceedings of the eighth ACM symposium on Solid modeling and applications* (2003), ACM Press, pp. 96–107.

[HBK02] HISADA M., BELYAEV A. G., KUNII T. L.: A skeleton-based approach for detection of perceptually salient features on polygonal surfaces. *Computer Graphics Forum 21*, 4 (2002), 689–700.

[HK00] HOLLEMAN C., KAVRAKI L. E.: A framework for using the workspace medial axis in prm planners. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation* (2000), pp. 1408–1413.

[MDSB02] MEYER M., DESBRUN M., SCHRODER P., BARR A. H.: Discrete differential-geometry operators for triangulated 2-manifolds. In *Proc. International Workshop on Visualization and Mathematics* (Berlin-Dahlem, 2002).

[RBO02] RIBO R., BUGEDA G., ONATE E.: Some algorithms to correct a geometry in order to create a finite element mesh. *Computers and Structures 80* (2002), 1399–1408.

[SPFE96] SHERBROOKE E. C., PATRIKALAKIS N. M., , F.-E.WOLTER: Differential and topological properties of medial axis transform. *Graphical Models and Image Processing 58*, 6 (1996), 574–592.

[TH03] TAM R., HEIDRICH W.: Shape simplification based on the medial axis transform. In *Proceedings of the 2003 IEEE Visualization Conference* (2003).

[TKGC03] TCHON K.-F., KHACHAN M., GUIBAULT F., CAMARERO R.: Constructing anisotropic geometric metrics using octrees and skeletons. In *Proceedings of the 12th International Meshing Roundtable* (September 2003), pp. 293–304.

[VO95] VLEUGELS J., OVERMARS M.: *Approximating generalized Voronoi diagrams in any dimension*. Tech. Rep. UU-CS-95-14, Dept. Comput. Sci., Utrecht Univ., Utrecht, Netherlands, 1995.

[YBS03] YOSHIZAWA S., BELYAEV A. G., SEIDEL H.-P.: Free-form skeleton-driven mesh deformations. In *Proceedings of the eighth ACM symposium on Solid modeling and applications* (2003), ACM Press, pp. 247–253.