Conformal multi-material mesh generation from labelled medical volumes

Christian Kehl,Daniel F. Malan and Elmar Eisemann

Delft University of Technology
Computer Graphics and Visualization,
Mekelweg 4, NL-2628 CD Delft
eMail: c.kehl@tudelft.nl – fmalan@medvis.org
URL: http://graphics.tudelft.nl/MedVis/MMMeshing

Abstract: Generating accurate volume meshes from segmented image volumes is important in certain medical applications. One application is finite element analysis (FEA) for orthopaedic pre-operative planning. The resulting FEA models can be used in patient-specific, bio-mechanical simulations and implants positioning.

Currently available software packages for volume meshing of segmented data can generate accurate, usable meshes. However many of them cannot create feature-adaptive volume meshes of multiple intersecting structures while maintaining high precision. An approach that respects both these requirements is based on a dynamic particle system. In this paper, we propose a system that specifically includes the ability to handle multiple labels and their interfaces.

Dynamic particle systems are computationally expensive and may be slow. As an improvement to our reference method, we implement a fast Integer Medial Axis algorithm. Furthermore, we propose an improved local triangulation scheme that may enhance the ability to model sharp features with a given particle count. Strict theoretical sampling restrictions limit the flexibility of Delaunay-based meshing methods. We suggest that using a local reconstruction scheme is a good approach to decouple the system from strict sampling limitations.

We evaluate our system by comparing it to a reference particle-based meshing implementation. Results show that the choice of the Medial Axis Transform (MAT) methods greatly influences quality and speed of feature-adaptive volume meshing.

1 Introduction

The generation of volumetric meshes from medical scans is an often-required step in patient-oriented orthopaedic planning. The generated volume meshes can be used in bio-mechanical simulations to optimize positioning and orientation of orthopaedic implants as well as to adapt the implant design to the needs of the patient. Such a workflow example is shown in Fig. 1. Bio-mechanical simulations demand topologically correct meshes to successfully compute stress and strain in

implants and bone. The precision of the simulation is directly dependent on the precision of the mesh, which is why very precise meshes are needed for such simulations. Also the runtime of bio-mechanical simulations is a relevant issue that has a practical impact on a system's usability. Accurate models with a large number of elements are often required to represent an anatomical region, but may result in a prohibitive computational load. Therefore, required medical volume meshes should try to intelligently minimize the number of mesh elements to reduce the computation time.

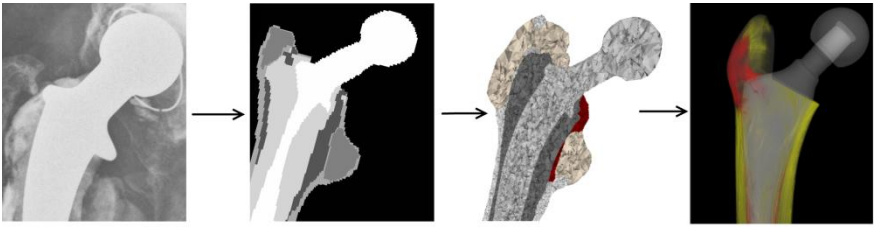


Fig. 1 Orthopaedic workflow for hip prosthesis replacement: A 3D CT-scan (a) is segmented into labels (b). This discrete volume image is converted to a volume mesh (c). Based on medical simulations (d) **[1]**, optimal implant design and positioning can be determined.

One common way to generate volume meshes out of segmented medical scans is by generating of conformal surface meshes for each segmented sub-structure. These surface meshes are then filled by volume meshing algorithms. An example of this approach is the Delaunay Surface Triangulation of each labelled medical structure and its volumetric refinement via 3D Delaunay Tetrahedralization [2] [3].

2 Related Work

The base challenge of our research is the creation of volume meshes. Here, we focus on Delaunay Tetrahedralization concepts. Two major approaches for feature-adaptive Delaunay Tetrahedralization are meshing by refinement, and meshing of iteratively optimized surface-point vertices. Volume meshing by refinement commonly starts by building an initial volume mesh [4]. Then, material interfaces are determined. Afterwards, a weighted Delaunay Tetrahedralization iteratively refines the initial mesh by vertex insertion while protecting material interfaces [5].

In meshing via iteratively optimized surface-point vertices, we build up an optimal surface mesh that is successively filled by volume meshing methods (i.e. Delaunay Tetrahedralization, Advancing Fronts). The major assumption is that a conformal, precise surface mesh is a sufficient description to subsequently generate a conformal volume mesh. The initial surface mesh vertices are determined by a particle system optimization step [6]. This particle distribution needs to adhere to 3D Delaunay mesh topology guarantees and limitations for generic point clouds [7], which results in the ε-sample requirement, first stated by Amenta et al. [8] [9] [10].

Meyer et al. use this sample requirement to dynamically distribute point-particles on 3D surface structures for subsequent meshing. This particle system approach also lends itself to multi-material medical volume meshes [2]. The advantages of this meshing approach are:

- application to multi-material medical structures
- possibility of feature-adaptive, non-uniform mesh output
- theoretical guarantees

This approach also has some drawbacks, which we want to address in our research. These drawbacks include:

- long computation time
- oversampling of edges and corners (see Fig. 2)
- lack of sharp-feature recreation due to ε-sample requirement

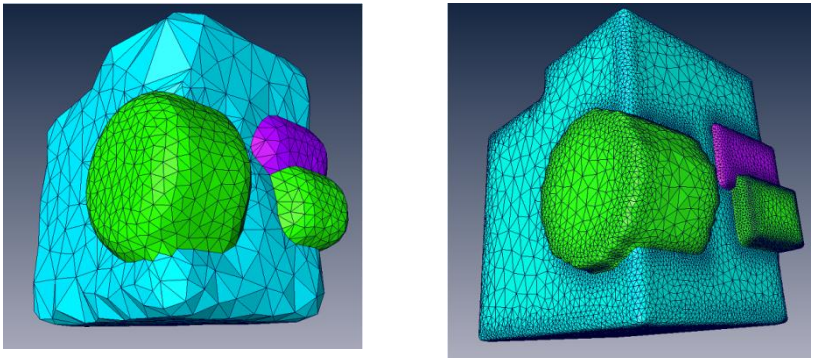


Fig. 2 Too few samples lead to wrong topology or excessive smoothing (left), too many samples lead to an excessive number of mesh elements (right).

3 Contribution

Our main contribution is the application of a fast, discrete Medial-Axis scheme to speed up the volume mesh generation process from image volumes. Our side contribution is the proposal of a local surface triangulation scheme for image volumes to overcome the ε-sampling requirement during the volume meshing process.

4 Concepts

We have two major objectives in our work. First, we want to speed up the meshing system. Second, the 3D reconstruction of segmented structures should happen with a minimal amount of sample-point vertices while maintaining topological correctness of the resulting model. We handle both challenges separately.

4.1 *System Speed-Up by fast, discrete Medial Axis Transform*

Our initial analysis of BioMesh3D [2], our reference system, revealed that its MAT implementation is the computationally most expensive sub-process (see Fig. 3). BioMesh3D calculates the MAT with centres of maximal spheres [11]. This algorithm is not optimal for the task because, although generating a highly precise MAT, the Centre-of-Maximal-Spheres algorithm is not designed for discrete objects such as digital volume images. A suitable MAT algorithm requires:

- high precision on sub-voxel level
- fast, efficient computation
- a design for discrete space

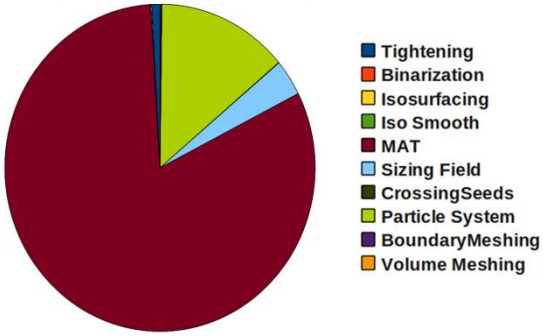


Fig. 3 BioMesh3D runtime assessment: the chart shows the relative runtime of each meshing sub-process with respect to the total runtime. It reveals the major computational expense of the Medial Axis-computation.

We reviewed existing MAT algorithms on the basis of these requirements. Neither the common thinning algorithm [12] [13] nor the highly-precise reduced discrete Medial Axis (RDMA) [14] [15] were suitable candidates. The Integer Medial Axis (IMA) algorithm [16] met all listed requirements, which is why we decided to apply it in our case. Notable properties of the IMA are:

- linear complexity with respect to the number of input voxels
- flexibility, due to pruning parameters of axis branches
- acceptable accuracy
- discrete data-centred design

4.2 *Minimal Sample for topologically accurate 3D Reconstruction*

BioMesh3D controls the model's smoothness via a process called Tightening [17] [6]. The Tightening-filter limits the maximum radius of curvature. The relation between radius of curvature and the number of sample points is as follows:

ε-sampling requirement:

$$\forall x \in S, E \cap B\big(x, \varepsilon \cdot \lambda(x)\big) \neq 0 \quad (1)$$

$$\lambda(p_1) \sim \text{ radius of curvature} \quad (2)$$

with λ being the local feature size (LSF) [8] [10]. As a consequence, areas with a large radius of curvature need fewer samples to be reconstructed. Areas with a small radius of curvature need more samples to be reconstructed.

At this point, we want to highlight that the ε-sampling requirement only applies to full 3D Delaunay Triangulations without prior surface knowledge. It is possible to generate a conformal tetrahedral volume mesh via 3D Delaunay Tetrahedralization from any conformal triangular surface mesh without the insertion or displacement of surface vertices (e.g. Steiner Points) [18]. We therefore avoid the strict ε-sampling limitation by replacing the 3D Delaunay Triangulation with 2D Triangulation in local tangent space, for which the sampling requirement doesn't apply. We extract the necessary tangent space parameters (i.e. vertex normals) via point sample projection on a isosurface that is generated directly from the input volume image (see Fig. 4). For reasons of simplification, we approximate the isosurface via a Marching Cubes-surface of the original volume image structures.

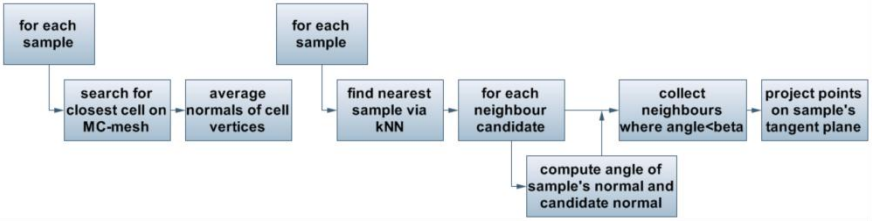
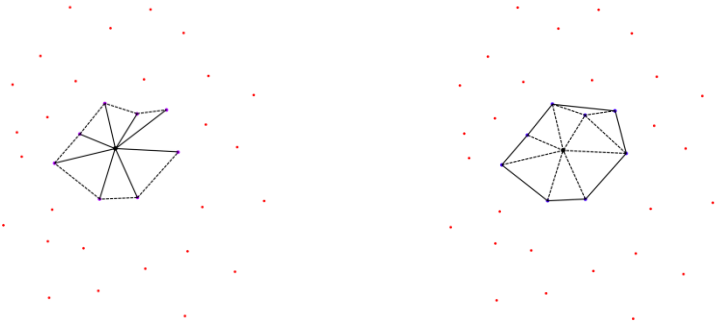


Fig. 4 Re-projection process: A first step is the determination of point sample normals via projection on the isosurface (left chart). Subsequently, we determine the neighbourhood of each sample via k-nearest neighbour extraction and the application of an inter-normal angle criterion.

The extracted neighbourhood is then meshed via 2D Local Delaunay Triangulation (LDT). The extracted mesh connectivity is then applied in 3D space. We designed three triangulation algorithms: an unconstrained LDT, a star-constrained LDT (fig. 5a) and a Convex Hull-constrained LDT (fig. 5b). For more information on the implementation with CGAL, we refer the reader to the online material [19].



(a) (b)

Fig. 5 constrained Local Delaunay Triangulation variants: star (5a) / convex hull (5b)

5 Results

We integrated the volume meshing concept described in Sec. 4 in the open source "Delft Visualisation and Image processing Development Environment" (DeVIDE) [20]. DeVIDE's modular development environment is well-suited to our modular modifications to the existing methodology of Biomesh3D. In the remainder of this section, we separately describe the results and improvements we obtained with IMA and LDT.

5.1 *Integer Medial Axis*

We used the IMA to speed up the meshing process while maintaining the precision of the reference system, BioMesh3D. We compared our results with BioMesh3D, using the common "tooth" volume image sample dataset. The runtime comparison is presented in Table 1 and Fig. 6. One can see that the IMA effectively reduced the mesh computation time.

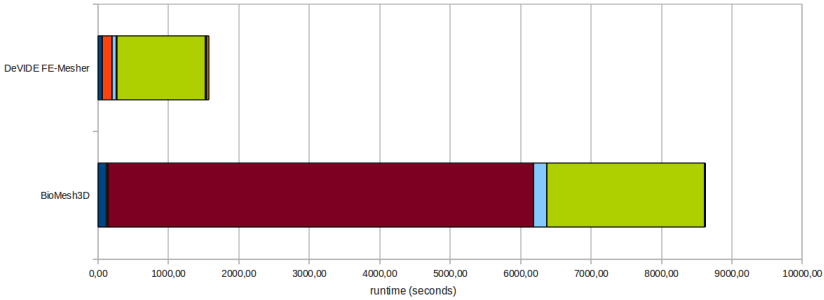


Fig. 6 "Tooth" runtime measurement comparison (b), showing our solution (top bar) and BioMesh3D (bottom bar)

	BioMesh3D	DeVIDE FE-Mesher
Tightening	2 min 1 sec	1 min 2 sec
Binarization		2 min 18 sec
Isosurfacing	14 sec	0.6 sec
Isosurface Smoothing		1.4 sec
Medial Axis Transform	1 h 40 min 52 sec	1 sec
Sizing Field	3 min 1 sec	52 sec
Crossing Seeds	8 sec	14 sec
Particle System	37 min 15 sec	20 min 5 sec
Boundary Meshing	8 sec	18 sec
Volume Meshing	7 sec	35 sec
TOTAL	2 h 23 min 5 sec	26 min 17 sec

Table 1 comparison of runtime for Tooth dataset

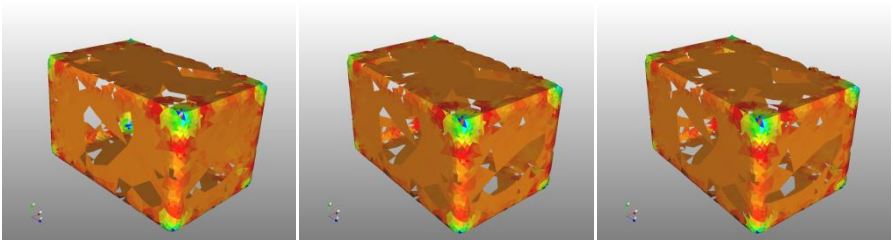
We assess the precision of the mesh by assessing the volume-mesh quality with respect to aspect ratio, radius ratio and tetrahedral volume. The aspect and radius ratio describe the mesh quality, since large values indicate the presence of degenerated triangles. The range of volumes is only a hint regarding the mesh grading. Tetrahedra that deviate a lot from the mesh average are considered tetrahedra of bad quality. The results of our meshing process (Table 2) show that we still need to address the high amount of bad tetrahedra compared to other comparable software packages [21] in the future. On the other hand, one has to note that results of BioMesh3D and our meshing system are similar with respect to quality range, average and variance.

Tooth	# Tetra	142795				
	Max.	Min.	Avg.	Variance	# bad Tetra	% bad
Aspect Ratio	119.69	1.01	1.94	1.00	9282	6.50
Radius Ratio	105.43	1.00	1.69	0.83	6736	4.72
Volume	272.34	0.0	2.96	21.74	0	0.0

Table 2 DeVIDE FE-Mesher Quality: Tooth dataset

5.2 *Local Delaunay Triangulation*

The LDT of non-uniformly sampled data is currently not successful. In this first conceptual implementation, we used the kNN-Algorithm to determine the local neighbourhood of each sampled particle. Our results show that kNN algorithms are not suited to sparsely-sampled, fast-grading point clouds. A consequence is the formation of holes in the mesh, making it useless for volume meshing.



(a) (b) (c)

Fig. 7 meshing results with non-constrained LDT via VTK (a), non-constrained LDT via CGAL (b) and convex hull-constrained LDT via CGAL (c) on an artificial example dataset. The colours indicate the curvature of the base MarchingCubes mesh, with orange representing no curvature, red representing concave curvatures and blue representing convex curvatures.

6 Conclusion and Future Work

Our objective was to improve the computational performance and to reduce the number of tetrahedral elements in dynamic particle-based multi-material tetrahedral meshing of medical image volumes. Our design consists of the IMA calculation to speed up the application and the use of a local triangulation algorithm. The change from the “centre-of-maximal-spheres” calculation to the IMA resulted in an improved runtime while maintaining the original precision. Despite the problems we encountered in practice, we still think that local triangulation algorithms could be a promising direction for non-uniform meshing because they are not bound to the strict ε-sampling criterion. A working local triangulation algorithm would need a precise 1-ring neighbourhood description because simple nearest-neighbour determination seems unsuitable for the task.

In the future, we plan to focus on a correct 1-ring neighbourhood localisation to generate closed, watertight surface meshes using a variant of LDT. One approach in this domain is the use of Natural Neighbours. Additionally, we are also investigating different ways of conformal volume meshing by reusing the initially-computed Marching Cubes surfaces.

Acknowledgement

The authors would like to thank dr. ir. Charl P. Botha (Computer Graphics and Visualization / TU Delft) and Prof. Dr. rer. nat. Herbert Litschke (Multimedia Engineering / HS Wismar) for their excellent collaboration and supervision during the Master Thesis [21] of Christian Kehl, on which this paper is based. We also thank dr. ir. Edward R. Valstar for his guidance and advice during our research.

References

[1] C. Dick, J. Georgii and R. Westermann, “A real-time multigrid finite hexahedra method for elasticity simulation using CUDA,” *Simulation Modelling Practice and Theory*, vol. 19, pp. 801-816, 2011.

[2] M. Meyer, R. Whitaker, R. M. Kirby, C. Ledergerber and H. Pfister, “Particle-based Sampling and Meshing of Surfaces in Multimaterial Volumes,” *#IEEE_J_VCG#*, vol. 14, pp. 1539-1546, 2008.

[3] S. Oudot, L. Rineau and M. Yvinec, “Meshing volumes bounded by smooth surfaces,” 2005.

[4] J. P. Pons, E. Ségonne, J. D. Boissonnat, L. Rineau, M. Yvinec and R. Keriven, “High-quality consistent meshing of multi-label datasets.,” *Inf Process Med Imaging*, vol. 20, pp. 198-210, 2007.

[5] D. Boltcheva, M. Yvinec and J.-D. Boissonnat, “Mesh Generation from 3D Multi-material Images,” *Medical Image Computing and Computer-Assisted Intervention*, vol. 5762, pp. 283-290, 2009.

[6] M. Meyer, R. Kirby and R. Whitaker, “Topology, Accuracy, and Quality of Isosurface Meshes Using Dynamic Particles,” *Visualization and Computer Graphics, IEEE Transactions on*, vol. 13, pp. 1704-1711, 2007.

[7] N. Amenta, M. Bern and M. Kamvysselis, “A new Voronoi-based surface reconstruction algorithm,” 1998.

[8] N. Amenta, M. Bern and D. Eppstein, “The crust and the beta-Skeleton: combinatorial curve reconstruction,” *Graph. Models Image Process.*, vol. 60, no. 2, pp. 125-135, 1998.

[9] N. Amenta and M. Bern, “Surface reconstruction by Voronoi filtering,” 1998.

[10] J. Boissonnat and S. Oudot, “Provably good sampling and meshing of surfaces,” *Graphical Models*, vol. 67, pp. 405-451, 2005.

[11] D. Sheehy, C. Armstrong and D. Robinson, “Shape description by medial surface construction,” *Visualization and Computer Graphics, IEEE Transactions on*, vol. 2, pp. 62-72, 1996.

[12] T. Lee, R. Kashyap and C. Chu, “Building Skeleton Models via 3-D Medial Surface Axis Thinning Algorithms,” *CVGIP: Graphical Models and Image Processing*, vol. 56, pp. 462-478, 1994.

[13] G. Bertrand, “A parallel thinning algorithm for medial surfaces,” *Pattern Recognition Letters*, vol. 16, pp. 979-986, 1995.

[14] D. Coeurjolly and A. Montanvert, “Optimal Separable Algorithms to Compute the Reverse Euclidean Distance Transformation and Discrete Medial Axis in Arbitrary Dimension,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, pp. 437-448, 2007.

[15] D. Coeurjolly, J. Hulin and I. Sivignon, “Finding a minimum medial axis of a discrete shape is NP-hard,” *Theoretical Computer Science,* vol. 406, pp. 72-79, 2008.

[16] W. Hesselink and J. Roerdink, “Euclidean Skeletons of Digital Image and Volume Data in Linear Time by the Integer Medial Axis Transform,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 30, pp. 2204-2217, 2008.

[17] J. Williams and J. Rossignac, “Tightening: Curvature-Limiting Morphological Simplification”, GVU,” 2004.

[18] P. George, F. Hecht and E. Saltel, “Automatic mesh generator with specified boundary,” *Computer Methods in Applied Mechanics and Engineering,* vol. 92, pp. 269-288, 1991.

[19] J.-D. Boissonnat, H. Brönnimann, O. Devillers, A. Fabri, F. Fichel, J. Flötotto, M. Teillaud and M. Yvinec, *2D Triangulations,* 2012.

[20] C. P. Botha and F. H. Post, “Hybrid Scheduling in the DeVIDE Dataflow Visualisation Environment,” 2008.

[21] C. Kehl, “Conformal multi-material mesh generation from labelled medical volumes,” 2011.