

Multimaterial Meshing of MRI Head Data for Bioelectric Field Simulations

Ross Whitaker, Robert M. Kirby, Jeroen Sinstra, Miriah Meyer, Martin Cole
Scientific Computing and Imaging Institute, University of Utah
Salt Lake City, UT 84112

1 Introduction

The problem of body fitting meshes that are both adaptive and geometrically accurate is important in a variety of biomedical applications in a multitude of clinical settings, including electrocardiology, neurology, and orthopedics. *Adaptivity* is necessary because of the combination of large-scale and small-scale structures (*e.g.* relatively small blood vessels spanning a human head). *Geometric accuracy* is important for several reasons. In some cases, such as computational fluid dynamics, the fine-scale structure of the fluid domain is important for qualitative and quantitative accuracy of the solutions. More generally, finite element approximations of elliptic problems with rough coefficients require increased spatial resolution normal to material boundaries [3]. The problem of constructing meshes from biomedical images is particularly difficult because of the complexity and irregularity of the structures, and thus tuning or correcting meshes by hand is quite difficult and time consuming. Many researchers and, indeed, commercial products simply subdivide the underlying hexahedral image grid and assign material properties to tetrahedra based on standard decomposition of each hexahedron into tetrahedra.

This paper presents a small case study of the results of a recently developed method for multimaterial, tetrahedral meshing of biomedical volumes [6]. The method uses an iterative relaxation of surface point point positions that are constrained to subsets of the volume that correspond to boundaries between different materials. In this paper we briefly review the method and present results on a set of MRI head images for use in bioelectric field simulation and source localization.

2 Mesh Generation Methodology

The goal of this paper is to examine the problem of forming anatomy-conforming tetrahedral meshes from segmented, three-dimensional MRI images of the human head for the purposes of finite element simulations of the electric fields. The pipeline for generating tetrahedral meshes from images consists of three parts: volume preprocessing, surface meshing, and volume meshing. This meshing method is presented in [6], but we briefly review it here for completeness.

We represent interfaces in a multimaterial dataset using a model that describes each material with a smooth, volumetric *indicator function*, f_i [4]. A

set of N indicator functions $F = \{f_i | f_i : V \mapsto \mathbb{R}\}$ represents n materials. In practice these functions are formed by smooth interpolations of the output of segmented volumes or label maps, such as shown in Figure 1. For this work, we process each label map using a level-set implementation of the grayscale morphology algorithm proposed by Williams and Rossignac [10] called *tightening*, which limits the radius of curvature of the resulting boundary.

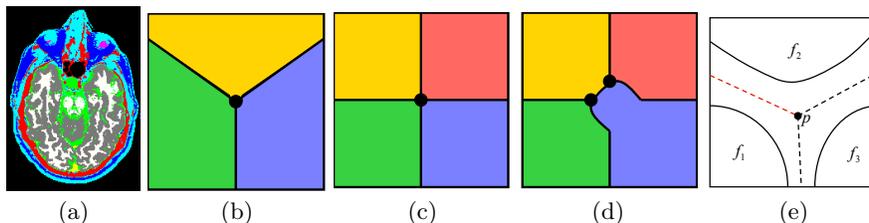


Fig. 1: (a) Multimaterial volumes, such as this segmentation of a head MRI are used to generate 3D meshes. (b) Generically, in 2D, two or three materials can meet (up to four in 3D). (c)–(d) Four material junctions are not generic, and result in generic conditions with small perturbations. (e) The generic case can be represented through smooth indicator functions.

A material label i is assigned to a point $x \in V$ if (and only if) $f_i(x) > f_j(x) \forall j \neq i$. In the case where $N > 2$, the boundaries that separate materials are not necessarily manifold, and can form sharp corners and edges. We characterize each material interface in terms of the number of material indicator functions that are maximal (and equal) at that junction. After this preprocessing, we can define *inside/outside* (IO) functions for each material so that the boundaries between different materials, represented as zero sets of the IO functions, coincide. We define these functions:

$$\tilde{f}_i = f_i - \max_{j=1, j \neq i}^n f_j. \quad (1)$$

Material junctions occur at points where two or more of the indicator functions (f_i) have equal, maximal value. We characterize the order of the junction by the number of coincident materials. For $V \subset \mathbb{R}^2$, 2-junctions and 3-junctions occur generically, as shown in Figure 1 (b–c), while a 4-junction is a nongeneric case, as shown in Figure 1 (d–e). For $V \subset \mathbb{R}^d$ each K -junction forms a subset of V that is topologically equivalent (homeomorphic) to a P -disk, where $P = d - K + 1$. Thus each type of material junction can be considered a P -cell, as described in the literature on discrete topology [1]. Generically, for $d = 3$ we have 4-junctions (0-cells), points; 3-junctions (1-cells), and 2-junctions (2-cells) or surfaces. For the case of the head data in this paper, we use six materials: grey matter, white matter, CSF, skin, bone, and air. Thus we have the possibility of 15 different types of 2-junctions (2-cells), 20 types of 3-junctions (1-cells), and 15 types of 4-junctions (0-cells).

The meshing method uses a set of points, which we call *particles* whose positions are updated along the gradient of a objective function that prefers regular configurations. In previous work, we have shown that these objective functions can be constructed to adaptive, high-quality sampling of implicit surfaces [5]. In this work each particle is constrained to a particular material junction. This formulation for each type of cell includes a set of projection operators that force particle motions to remain in the tangent space of the associated material boundary. A hierarchy of particle systems then samples each type of generically occurring material junction such that each junction is represented in the final mesh. That is, in 3D the 0-cells (points) are sampled first, followed by the 1-cells (curves), and concluding with the 2-cells (surfaces). Finally, a simple labeling algorithm extracts the multimaterial surface meshes as a subset of a Delaunay tetrahedralization of the samples, as described below.

To represent the interface between sets of materials, we define a *cell indicator function* that identifies points in V where a set of IO functions evaluate to zero, such as the material interface between materials 1 and 2 shown as the red dashed line in Figure 1. For a set of materials M (where $2 \leq |M| \leq 4$ generically in 3D), we have

$$J_M = \sum_{i \in M} \tilde{f}_i^2, \quad (2)$$

and the zero set of this function is the cell defined by the interface of the materials M .

We use the particle system framework of Meyer et. al [5, 6] for placing points along each material junction. This is an iterative algorithm that relaxes the particle positions, prior to a triangulation or meshing, in order to minimize an energy function that favors equal distributions of points or *particles*. This framework uses a *sizing field* that informs particles of how far they should be from their neighbors. We set this sizing field to be proportional to the *local feature size* (LFS) [2], which is the distance to the nearest point on the medial axis. In this way, the sizes of triangles on the surface meshes will be proportional to the sizes of the tetrahedra that will be needed to fill the adjacent volumes. We generate a sizing field volume, at the same resolution as the input data, for a multimaterial dataset, by first computing an approximate medial axis of each IO function. We then store at each grid point in the sizing field volume the minimum LFS for the set evaluated at the grid point location. We smooth this field using a gradient-limiting PDE proposed by Persson [7].

For the tetrahedral meshing we use the open-source software package *Tetgen*¹. We run Tetgen in a constrained-Delaunay mode so that it adds points to the volume interior but does not add points to the surface. This Delaunay tetrahedralization (DT) uses the constrained DT method proposed in [9]. The Tetgen software includes a Delaunay refinement approach [8] for ensuring mesh quality, as measured by *edge-radius ratios*, which does not, unfortu-

¹<http://tetgen.berlios.de>

nately, penalize slivers. Tetgen does include a sliver removal step, consisting of edge-flips and peeling away of slivers near the boundary, which is only moderately effective.

3 Case Studies

In this section we present a small set of case studies that demonstrates the use of this meshing strategy for indicator functions derived from smoothed label maps of segmented, 3D, MRI images. The three data sets are from two pediatric patients suffering from neurological symptoms (*e.g.* seizures), ages 12 and 15, and a single normal volunteer, age 28, which we call cases 1, 2, and 3, respectively. The segmentations come from a semiautomated tissue classification algorithm followed by a manual inspection and hand editing of mislabeled pixels.

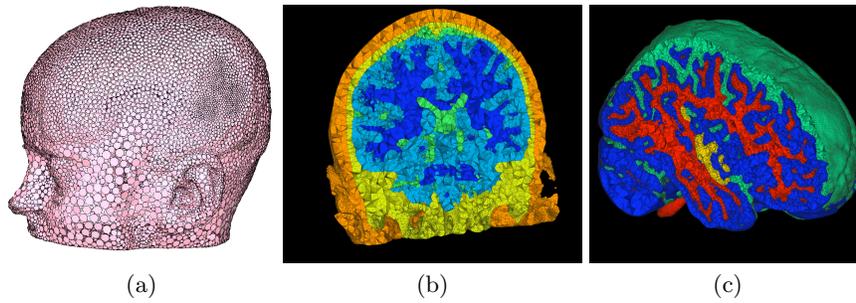


Fig. 2: (a) A particle distribution for Case 2 shows the adaptivity of the particles for thin regions of the skin/skull. (b) The tetrahedral mesh for Case 2. (c) The tetrahedral mesh for cortex and CSF of Case 2b, showing higher resolution.

Cases	1	2	2b (high res)	3
Num tris	265,989	450,945	3,510,322	321,532
Num tets	1,067,541	1,765,216	15,561,759	1,190,125
Tris RR avg(min)	0.89(0.031)	0.89(1.8×10^{-4})	0.93(2.6×10^{-4})	0.89(7.6×10^{-5})
Tets RR avg(min)	0.61(0.013)	0.61(1.0×10^{-5})	0.64(2.0×10^{-5})	0.62(5.1×10^{-5})

Each material is extracted into a binary volume, and a distance transform is constructed to the interfaces for that material, and the surface is smoothed with a *tightening* of radius 0.5, and approximate medial axes of each material are constructed by detecting coincidences of foot points on a grid with the same resolution as the original image. The feature size volume resulting from the distance to this medial axis is gradient limited (smoothed to enforce maximum on gradient values [7]) to a value of 0.5 (unitless), and then multiplied

by a value ϵ , which controls the resolution of the final mesh, is 2.5, except Case 2b, which is the high-resolution version of Case 2, with $\epsilon = 1.0$. Figure 2 shows surface and tetrahedral meshes that result from the proposed, particle-based multimaterial meshing method. Table 3 shows quantitative analyses of the triangles and tetrahedra for the three cases.

The results of this method are quite promising. The triangle mesh quality is excellent, especially given the adaptivity and high geometric accuracy exhibited by the method. The tetrahedral quality, as measured by radius ratio (which is indicative of the conditioning of our stiffness matrix in the resulting linear system), is not as good, but acceptable for simulations. The worst tetrahedra are slivers, which are an expected outcome of the use of Tetgen for the final volumetric step. Particularly important is that the system used one set of parameters for all three cases and required no manual intervention after the initial tuning of parameters. Future work will focus on the computation time, which is 8-12 hours for each of these datasets, and is mostly spent of preprocessing and distributing particles, and the tetrahedralization, which would benefit from one of the variety of method that explicitly reduces slivers.

References

1. A. Adamson and M. Alexa. Point-sampled cell complexes. *ACM Trans. Graph.*, 25(3):671–680, 2006.
2. N. Amenta, M. Bern, and D. Eppstein. The crust and the beta-skeleton: Combinatorial curve reconstruction. *Graphic Models and Image Processing*, 60(2):125–135, Mar. 1998.
3. I. Babuska, B. Andersson, B. Q. Guo, and H. Oh. Finite element method for solving problems with singular solutions. *J Computational and Applied Math*, 74:51–70, 1996.
4. B. Merriman, J. K. Bence, and S. J. Osher. Motion of multiple junctions: A level set approach. *J. Comput. Phys.*, 112(2):334–363, 1994.
5. M. Meyer, R. M. Kirby, and R. Whitaker. Topology, accuracy, and quality of isosurface meshes using dynamic particles. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1704–1711, September/October 2007.
6. M. Meyer, C. Ledergerber, H.-P. Pfister, and R. Whitaker. Particle-based sampling and meshing of multimaterial volumes. In *Proc IEEE Visualization*, page To appear, 2008.
7. P.-O. Persson. Mesh size functions for implicit geometries and pde-based gradient limiting. *Eng. with Comput.*, 22(2):95–109, 2006.
8. J. Shewchuk. Tetrahedral mesh generation by delaunay refinement. In *Proc Fourteenth Annual Sym on Computational Geometry*, pages 86–95, 1998.
9. H. Si and K. Gartner. Meshing piecewise linear complexes by constrained delaunay tetrahedralizations. In *Proc 14th Int Meshing Roundtable*, 2005.
10. J. Williams and J. Rossignac. Tightening: Curvature-limiting morphological simplification. In *Proceedings of the Symposium on Solid and Physical Modeling*, pages 107–112, 2005.