A Morphing Algorithm for Generating Near Optimal Grids: Applications in Computational Medicine

Steven G. Parker
Department of Computer Science, University of Utah
Salt Lake City, UT 84112, USA
sparker@cs.utah.edu

David M. Weinstein
Department of Computer Science, University of Utah
Salt Lake City, UT 84112, USA
dweinste@cs.utah.edu

Christopher R. Johnson
Department of Computer Science, University of Utah
Salt Lake City, UT 84112, USA
crj@cs.utah.edu

Abstract
We apply *morphing* to the problem of generating the initial mesh for finite element simulations. This algorithm reduces mesh adaptation time by integrating physical and geometric constraints to provide a near optimal initial mesh. We apply this method to large-scale bioelectric field problems involving the complex geometries of the human body.

Introduction

Over the past two decades, the techniques of computer modeling and simulation have become increasingly important to the fields of bioengineering and medicine. Although biological complexity outstrips the capabilities of even the largest computational systems, the computational methodology has taken hold in biology and medicine and has been used successfully to suggest physiologically and clinically important scenarios and results.

One class of important applications in computational medicine are volume conductor problems which arise in electrocardiography and electroencephalography. The solution to these problems have utility in defibrillation studies and in impedance imaging tomography, and they are important in the detection and location of arrhythmias and in the localization and analysis of spontaneous brain activity in epileptic patients[1]. In general, these methods are a form of electric and potential field imaging and can be used to estimate the electrical activity inside a bounded volume conductor, either from potential measurements on an outer surface or directly from interior bioelectric sources.

The bioelectric fields that arise in the human body are, in general, governed by Maxwell's equations. Because of the time scale of bioelectric signals within the volume conductors of the thorax and skull, charge is distributed throughout the volume virtually instantaneously such that we can invoke a quasi-static approximation. The bioelectric fields can thus be described by the Poisson equation for electrical conduction, if we know the current distribution within the volume, or by Laplace's equation, if we know the voltage distribution on a bounded surface. This yields the general formulation,

$$\nabla \cdot (\sigma \nabla \Phi) = -I_{SV} \tag{1}$$

where σ is the conductivity tensor, Φ is the potential, and I_{SV} the cardiac source-current density. Two primary problems can be formulated from Equation (1). The first is the direct problem in electrocardiography (ECG): given a subset of potentials on the surface of the heart, or a description of the primary current sources within the heart, calculate the electric and potential fields within the body and upon the surface of the torso. The second is the problem of cardiac defibrillation: given known currents or voltages which are applied from external sources (e.g. defibrillation electrodes), determine the distribution of applied current throughout the heart.

To solve these problems, we have constructed a geometric model of the human thorax [1, 2] from 116 MRI scans recorded in 5 mm increments. Images were digitized into a set of discrete contours (poly-lines) and after some smoothing, additional points were added between the contours and the images were tesselated into a discrete set of elements - triangles for two-dimensional models and tetrahedra for three-dimensional models.

A finite element (FE) analysis is then utilized to approximate the bioelectric fields throughout the discretized geometry according to (1). A problem which immediately arises in constructing such discrete models and the primary topic of this paper is, *how does one know, a priori, what is an appropriate level of mesh discretization which balances solution accuracy and computational efficiency?* While at this point, there does not exist an answer to this question, we have taken a step towards seeking a plausible (if not optimal) approximation.

Traditionally, in adaptive finite element methods, one would start with a discretization of the geometry which conforms to the topology of the solution domain. Then a finite element solution would be computed and an error analysis performed to find elements which need refinement. Additional elements would be included (or the order of the basis function increased) and this would continue in an iterative fashion until some *a priori* convergence criteria had been reached.

[1]In this paper, we will focus upon primarily applications in cardiology, but note that the methods we develop are directly applicable to electroencephalography and other problems in computational medicine.

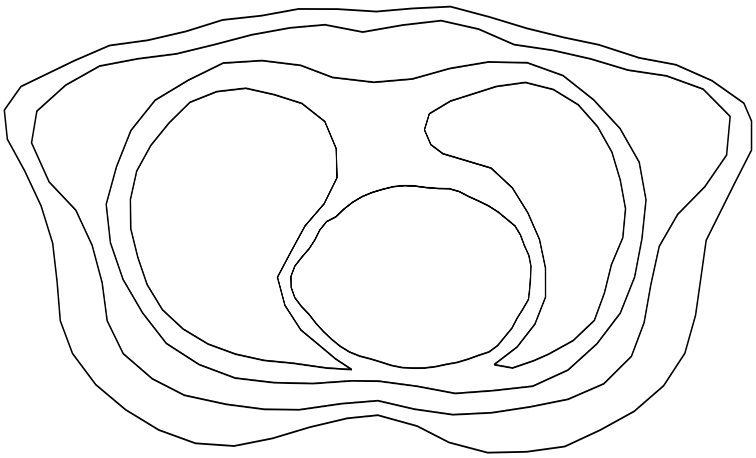

Figure 1: Surface boundaries from a single slice of MRI data

Mesh refinement would be unnecessary if one could somehow guess the final mesh from the start. In the face of complex geometries and inhomogeneities, this seems to be an impossible task. *However, if we could get close in the initial stage, fewer refinement steps would be required to reach the final stage.* For our grid to be nearly optimal, it must accurately reflect the geometry and the physics of our system. Effectively, it must be composed of small patches in the areas of high gradient and maintain the integrity of all boundaries.

To generate a grid with these properties, we "morph" or interpolate between the shapes of internal source boundaries and external insulating boundaries. Morphing provides the geometric characteristics of the mesh, and the *rate* of the morph controls the density of the mesh. The intermediate interpolated shapes are resampled in space and "time" to provide the actual mesh points. For models with simple topologies, this method produces results similar to mapping methods for grid generation[3]. However, as we will demonstrate, the morphing method is also capable of handling domains with complex topologies.

Methods

As an initial testbed, we have constructed an algorithm to generate meshes for a two-dimensional slice of our thorax data. A single trans-thoracic slice is taken from the MRI data, and a mesh is constructed using the boundaries obtained from segmenting this slice. The tissue boundaries (body, fat, muscle, lungs and heart) obtained from one slice of a male patient is shown in Figure 1. In two dimensions, the boundaries of the voltage sources, current sources, internal geometries, and the exterior of the model are closed planar curves.

A curve in two-dimensional space may be represented in several different ways. These representations can be divided into three classes: parametric, explicit, and implicit[4]. For this application, we will exploit the properties of the implicit representation in order to perform shape interpolation.

Let f_1 and f_2 be the implicit representations of two curves, c_1 and c_2. A blend function between c_1 and c_2 can be created by linearly interpolating between them. If we use t (for time) as our interpolant, then we can define this resulting blend function f_m as:

$$f_m(x, y, t) = (1 - t)f_1(x, y) + tf_2(x, y)$$

At time $t = 0$, $f_m(x, y, 0) = f_1(x, y)$, and at time $t = 1$, $f_m(x, y, 1) = f_2(x, y)$. For $0 < t < 1$, there is a smooth transition between f_1 and f_2.

Beyond simple geometric shapes, the implicit representation for a curve is usually not obtainable. This is particularly true for curves which were generated from real world data. There are at least two methods of obtaining an implicit representation for a curve from data points.

1. Function fitting: Find A_j, X_j and Y_j, such that:

$$f(x, y) \approx \sum_j \frac{A_j}{\sqrt{(x - X_j)^2 + (y - Y_j)^2}}$$

   for the known data points.

2. Signed distance function. Let $f(x, y)$ be < 0 if (x, y) is inside the polygon formed by the data points, and $f(x, y) > 0$ if (x, y) is outside that polygon. $f(x, y) = 0$ if (x, y) is on the polygon. $|f(x, y)| =$ the smallest distance to any point on the polygon. Thus $f(x, y)$ is a "signed distance" from the polygon edges. This method could also be applied to higher order curves such as splines.

We chose the 2nd option because of the simplicity of implementation, and it's faithful reproduction of the original curve.

Different Distributions

The primary goal is to create an initial mesh which reflects areas of high gradients in the domain, while at the same time conforming to the topology of the solution domain. The boundary conditions and sources give rise to these gradients. For the human thorax model, the heart is considered to be a voltage source, and the body exterior an insulating boundary, while any additional defibrillator electrodes are modeled as current sources. We make the following assumptions about the different boundary conditions.

1. **Dirichlet**: Produce gradients which have a $\frac{1}{r}$ falloff, meaning that the size of the elements should be directly proportional to r. This would correspond to a voltage source in our problem domain, such as the endogenous fields of the heart.

2. **Neumann**: Specified by $\nabla\Phi \cdot \mathbf{n} = f$. For our purposes, the Neumann boundary conditions are used in two different ways.

   (a) $f = 0$: Used to specify an insulating boundary. In this case, the boundary has no effect on the final mesh other than shape. The body exterior is an insulating boundary.

   (b) $f = g(x, y)$: Used to specify a current source. In this case, the gradients exhibit a $\frac{1}{r^2}$ falloff from the boundary, which would make the size of the elements proportional to r^2. Defibrillator electrodes are modeled as current sources.

Thus, the proper distribution for a voltage source (element size proportional to r), would require equally spaced contours, and the proper distribution for a current source (element size proportional to r^2), would require contours spaced proportionally to r. The proper distributions are given by the following interpolations:

Voltage source:

$$tD_{boundary}(x, y) + (1 - t)D_{vsource}(x, y) = 0$$

Current source:

$$t^2 D_{boundary}(x, y) + (1 - t^2)D_{isource}(x, y) = 0$$

where $D_{boundary}$, $D_{vsource}$ and $D_{isource}$ are the signed distance functions for the respective boundaries.

Consider two circles of radius 1 and radius 2, represented by the implicit equation $\sqrt{x^2 + y^2} - R$:

- For a voltage source on the inner circle:

$$\begin{aligned} f_m(x, y, t) &= t(\sqrt{x^2 + y^2} - 2) + (1 - t)(\sqrt{x^2 + y^2} - 1) \\ &= \sqrt{x^2 + y^2} - (1 + t) \end{aligned}$$

  producing circles of radius $1 + t$.

- For a current source on the inner circle:

$$\begin{aligned} f_m(x, y, t) &= t^2 \left(\sqrt{x^2 + y^2} - 2\right) + \left(1 - t^2\right) \left(\sqrt{x^2 + y^2} - 1\right) \\ &= \sqrt{x^2 + y^2} - \left(1 + t^2\right) \end{aligned}$$

  producing circles of radius $(1 + t^2)$.

For problems with non-circular boundaries, these interpolations are used to give the approximate distributions. If the sources are on the outer circle, different distributions must be formulated.

For a single problem which contains multiple sources of one type, they are combined into a single "metasource" by using the minimum value obtained from each of the implicit equations.

If a single domain contains both voltage sources (Dirichlet boundaries), and current sources (Neumann boundaries), then the resulting curves must be combined. Since the interpolation is done differently for the two cases, they cannot be combined into a single "metasource". Therefore, we combined the two contours at each contour level. This is done to the implicit representations with the following function:

$$f(x,y) = |f_v(x,y) f_c(x,y)| \ \mathbf{Or}\ (f_v(x,y), f_c(x,y))$$

where the **Or** function is defined as:

$$\mathbf{Or}\ (a,b) = \begin{cases} -1 & \text{if } a < 0 \text{ or } b < 0 \\ 1 & \text{if } a >= 0 \text{ or } b >= 0 \end{cases}$$

This produces the union of the two curves, with the overlapping region removed.

Building the Mesh

The first step in building the mesh is to locate the isocontours. For each t value, the linear combination of our sources and boundary produces an implicit function for which we wish to trace the zero-set. Our trace algorithm finds a point on this set for each source, and traces along it until the contour is complete. The contour is then resampled to contain a fixed number of points. We repeat this process for each isocontour, and finally utilize a Delauney triangulation method to mesh our points into an optimal configuration.

Results

A mesh generated using the morphing algorithm is shown in Figure 2. Figure 3 shows the same space meshed using evenly spaced points, and Figure 4 shows this space meshed using random points. All three mesh discretizes the region between the epicardium and the torso boundary (skin) for a single slice of MRI data. The gray lines show the mesh discretization obtained by each of the methods, and the dark lines show iso-voltage contours generated from the endogenous fields of the heart. The iso-voltage contours on each of the meshes were obtained by solving Laplace's equation for electrical conduction using experimentally obtained voltages as the Dirichlet boundary conditions. Initial simulations show that the mesh generated using the morphing algorithm is effective in reducing the number of iterations required to fully adapt the mesh. Observe that the element sizes in Figure 2 reflect the $\frac{1}{r}$ falloff of the fields induced by the heart, yielding a more accurate solution to the voltage field. In addition, the final number of nodes in our fully adapted mesh is significantly lower than the number of nodes produced from algorithms with standard initial meshes. Complete descriptions of these results will be included in the final paper.

Another set of meshes is shown for a region with two current sources (as would be the case for modeling defibrillation electrodes) in Figures 5, 6 and 7. These figures show the algorithm's performance with multiple sources. It is worth noting that the element sizes in Figure 5 reflect the $\frac{1}{r^2}$ falloff of the fields induced by current sources.

Future Work

Ultimately, we will expand these tools to work for three-dimensional spaces. Most of the extensions are straightforward — isocontours become isosurfaces, and the same linear morphing still applies, but the surfaces are more difficult to parameterize. We would also like to use coherence obtained from the point generation algorithm in conjunction with an incremental Delauney Triangulation algorithm to optimize the triangulation process.

Acknowledgments

The authors would like to thank Dr. John Schmidt for his valuable comments and suggestions. This research was supported in part by awards from the Whitaker Foundation and the National Institutes of Health.

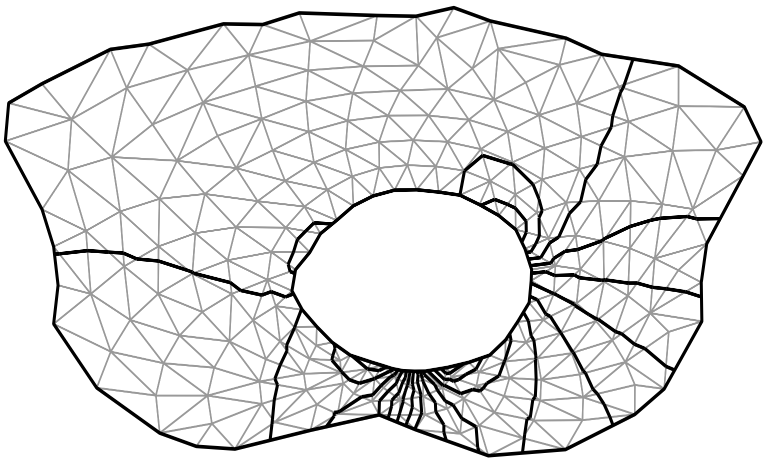

Figure 2: Heart and Torso meshed using Morphing algorithm (237 points)

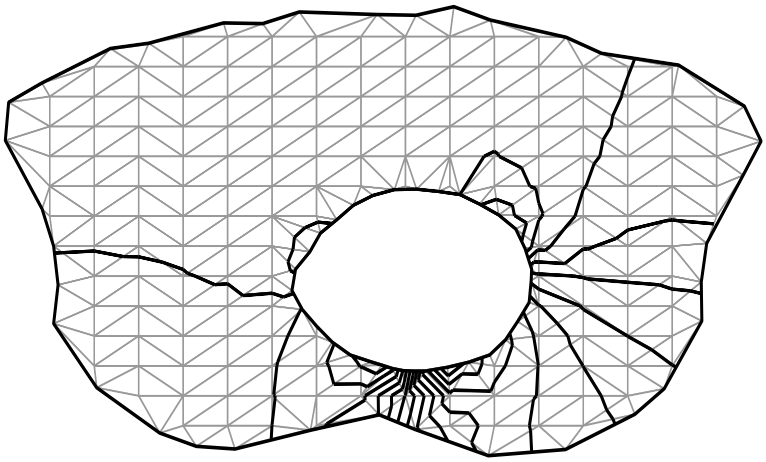

Figure 3: Heart and Torso meshed using regularly sampled points (237 points)

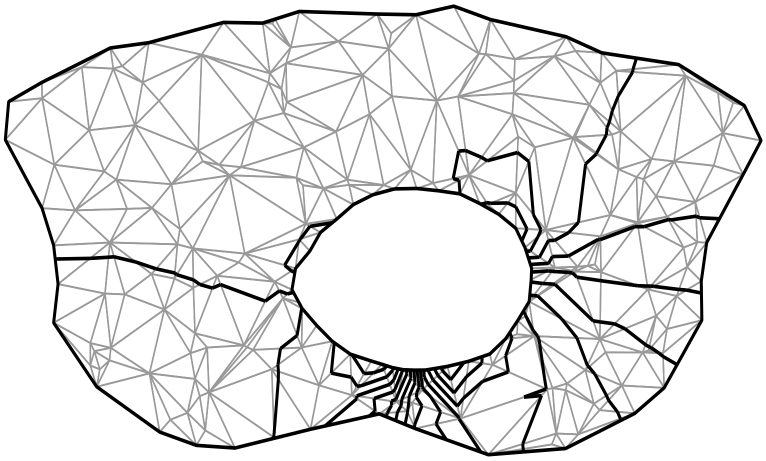

Figure 4: Heart and Torso meshed using randomly placed points (237 points)

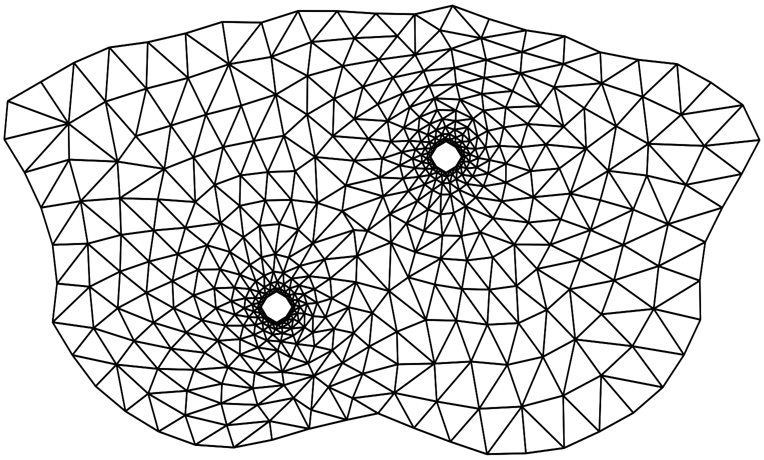

Figure 5: Heart and Torso meshed using Morphing algorithm (508 points)

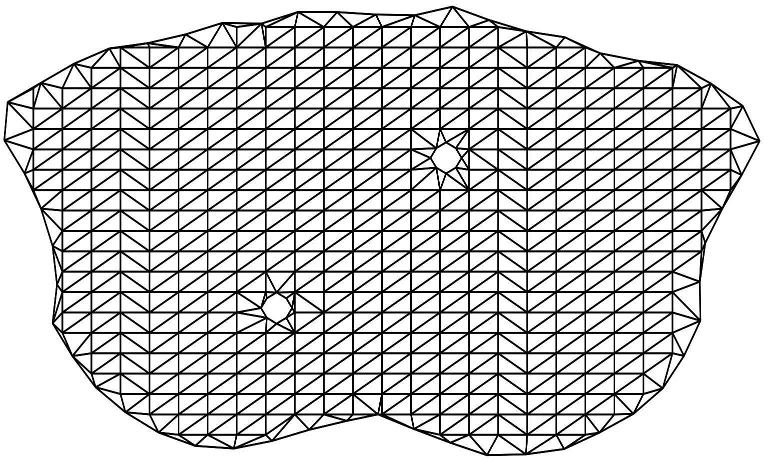

Figure 6: Heart and Torso meshed using regularly sampled points (506 points)

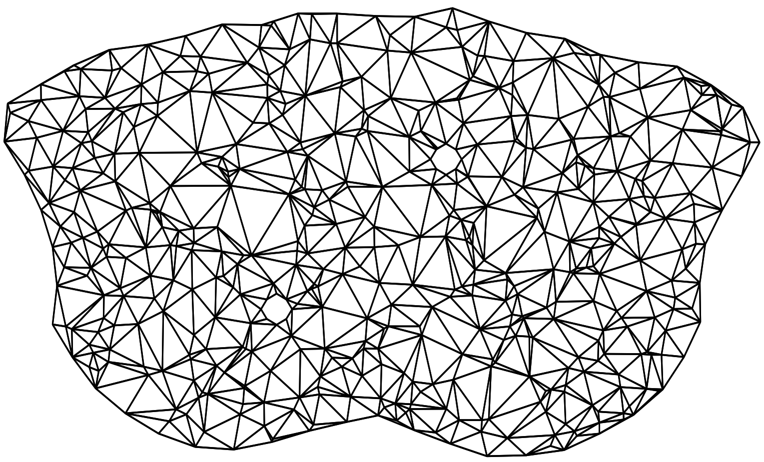

Figure 7: Heart and Torso meshed using randomly placed points (508 points)

References

[1] C.R. Johnson, R.S. MacLeod, and P.R. Ershler. A computer model for the study of electrical current flow in the human thorax. *Computers in Biology and Medicine*, 22(3):305–323, 1992.

[2] C.R. Johnson, R.S. MacLeod, and M.A. Matheson. Computer simultions reveal complexity of electrical activity in the human thorax. *Comp. in Physics*, 6(3):230–237, May/June 1992.

[3] P.L. George. *Automatic Mesh Generation.* Wiley, New York, 1991.

[4] Brian Wyvill. Building models with implicit surfaces. *IEEE Potentials*, 1992.