The BioPSE Inverse EEG Modeling Pipeline

David Weinstein
Petr Krysl
Christopher Johnson

Scientific Computing and Imaging Institute
Department of Computer Science
University of Utah
Salt Lake City, UT 84112
dmw@cs.utah.edu

Abstract

The goal of the inverse problem in electroencephalography is to determine electrical activity within the cranial volume based on potential measurements taken from the scalp. In this paper we discuss the modeling components of our software system for solving the inverse EEG problem. Our implementation is constructed within the BioPSE problem solving environment, a flexible framework for implementing large-scale scientific and engineering applications. We leverage the power of this underlying software architecture in the design and implementation of our inverse EEG pipeline.

Introduction

The inverse EEG problem can be described as the mathematical mapping of EEG scalp recordings back onto the cortical surface or within the cortex to approximate fundamental current sources. This inverse problem lies at the foundation of surgical planning and prognosis for neurological conditions ranging from epilepsy to schizophrenia [2] and to brain tumors. The goal of cortical mapping is to integrate patient anatomical information and measured voltage potential recordings from the surface of the patient's scalp in order to non-invasively determine the electrical activity on and within the patient's cortical surface [6].

There has been much research into computationally modeling the electrical activity of the brain, but only a few successful systems have been implemented. These systems range in geometric model complexity from grossly simplified, spherical representations to patient-specific finite element models. In this paper we detail the modeling tools we have developed in SCIRun/BioPSE[1] for constructing patient-specific finite element models.

Background

Modeling the large-scale electrical activity of the human brain is an active and still growing area of research. Also called source imaging or source localization, these techniques represent an attempt to understand the underlying neural generators of the spatiotemporal patterns observed in the scalp recorded EEG. Researchers are beginning to incorporate structural information from MRI and CT, physiological information from fMRI, PET, and SPECT, comparative neurological information from animal and human intracranial recordings, and theoretical and observed spatiotemporal brain dynamics.

The first significant commercial source localization software package for source imaging was BESA, introduced by Scherg [15] for the MS-DOS environment. It provided spherical head models and the specification of several types of dipole source configurations. Subsequently, a group of people from the European MEG community, sponsored by Phillips, came out with a UNIX package called CURRY. With CURRY one could begin to utilize realistic models of the head and brain and constrain the sources to lie on the cortical surface as defined from an MRI. Recently, Gevins and associates have begun marketing a new dense array EEG system, ManScan. Although it includes features for conventional dipole localization, the focus of the package is the inward continuation method shown in Gevins' publications [3, 5], producing the "cortical" or dural imaging output.

While the above products have proven adequate for approximate inverse solutions, none of them offers patient anatomy-specific solutions (*i.e.,* a solution based upon a patient's true anatomy). In the next section we present our implementation of the inverse EEG modeling pipeline that can incorporate patient specific solutions.

Methods

A schematic overview of our modeling pipeline is shown in Figure 1. A segmented MR volume provides the anatomical data required for accurate conductivity and boundary condition information in our model. Functional

[1] SCIRun is pronounced "ski-run" and derives its name from the Scientific Computing and Imaging (SCI) research group which is pronounced "ski" as in "ski Utah." BioPSE, Biomedical Problem Solving Environment, is a public domain version of SCIRun (specifically tailored for bioelectric field problems) that is being developed through the SCI Institute's NIH NCRR for Bioelectric Field Modeling, Simulation and Visualization.

data (the known EEG potentials at the scalp boundary) are read in from a raw file and stored with the digitized locations of the electrodes. This data forms the basis for a finite element inverse problem, whereby either the electric sources within the brain that induced the recorded EEG potentials, or the corresponding potentials on the cortical surface, can be computationally recovered.

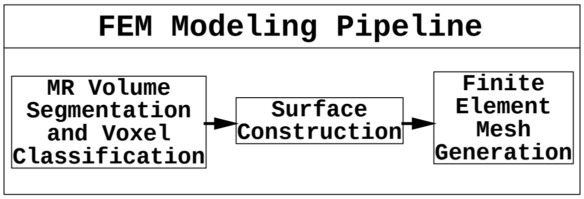
FEM Modeling Pipeline
MR Volume Segmentation and Voxel Classification
Surface Construction
Finite Element Mesh Generation

Figure 1: A schematic representation of the EEG finite element modeling pipeline.

For implementing the functional units of this finite element modeling pipeline and connecting them together, we have chosen to use the SCIRun/BioPSE software system [9, 10, 11]. SCIRun is a problem solving environment that uses a visual dataflow and computational steering framework. SCIRun provides an underlying architecture and development environment for applications. A system that provides both a visual dataflow interface and computational steering capabilities is highly advantageous, as it allows the user great freedom to interactively explore a problem and/or solution [1]. In a visual dataflow environment such as SCIRun, the user interactively selects building block components and attaches the inputs and outputs of the components to accomplish complex tasks. Taken as a whole, a collection of these building blocks, called *modules*, and input/output connections, called *data pipes*, forms a complete program or *dataflow network*. An example dataflow network is shown in Figure 2.

A publicly available version of SCIRun, termed BioPSE, is presently under development and will be released in the Spring of 2001. BioPSE is being developed as part of the NIH National Center Research Resource for Bioelectric Field Modeling, Simulation, and Visualization. Below, we introduce the BioPSE environment and describe how the Modeling subsystem of the inverse EEG pipeline has been constructed on top of this infrastructure.

BioPSE

In BioPSE, the user can interactively construct, edit, and save dataflow networks. When such a network is run, data progresses from upstream to downstream modules. Execution of the various modules is governed by a scheduler. When new data is passed to modules downstream, the scheduling algorithm forces the downstream modules to execute.

Computational Steering

Even during the execution of a module, the user has the ability to interactively control or *steer* the computation. For example, when coregistering two surfaces, the user can nudge one of the surfaces out of a local minimum without having to stop or restart the algorithm. Similar steering capabilities are available in the finite element matrix solver, where the user can change iterative solution methods before the system converges (perhaps from a steepest-descent to a conjugate-gradient method) or relax the convergence criterion, once again without having to restart the module. The ability to steer a particular computation offers an additional level of power and interactivity to the user.

Algorithm Development

The modularity of the visual dataflow interface is echoed by the plug-in style of adding new modules. To integrate a new module into BioPSE, a user simply implements the base functionality of the module, adds the appropriate input/output hooks, and the module can be compiled into the system. Furthermore, in implementing the base algorithm, the user can leverage BioPSE's built-in support for resource management (file I/O, thread management, memory management), libraries (common data structures, such as arrays, hash tables and queues; numerical methods such as matrix operations; and geometric structures, such as points, vectors and planes), and three-dimensional rendering capabilities.

As a plug-in system, BioPSE allows rapid prototyping and analysis of new implementations. The user can evaluate different methods by implementing them in different modules and "hot-swapping" them within a common dataflow network to compare results.

Modeling

The modeling components of the inverse EEG pipeline collectively enable the user to construct a full finite element mesh with appropriate boundary conditions and conductivity tensors from segmented MRI images, raw EEG potentials and digitized positional information. This modeling pipeline, shown in Figure 2, is detailed below.

Import Data

The user can format (or import) BioPSE objects from raw output data by executing a data conversion utility from a UNIX shell command line. The output of such a conversion program is an object stored in a file which can then be read into a BioPSE network and passed between modules. In this case, we import a segmented MR volume, generated by the Brigham and Women Hospital's E-M segmenter [18] (with each voxel classified as air, skin, skull, cerebro-spinal fluid, grey matter, or white matter).

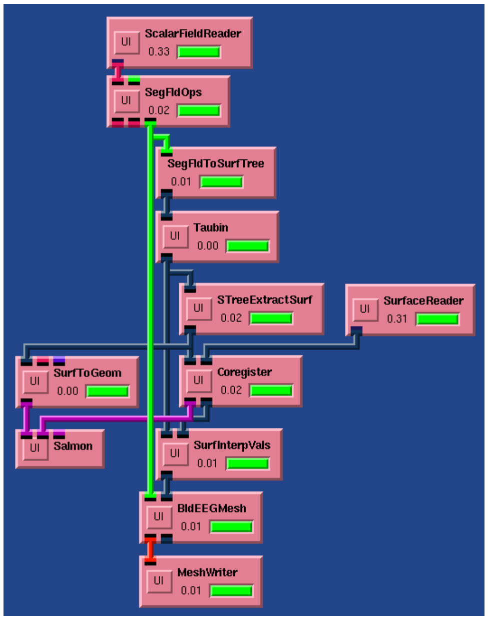

Figure 2: The BioPSE inverse EEG modeling pipeline.

Surface Construction

After the data has been imported into BioPSE, we extract the *separating surfaces* [7], that is the boundaries that contain uniform material regions of the volume. The uniform material regions correspond to anatomical structures, such as the skull, grey matter and white matter; the separating surfaces denote the interfaces (boundaries) between these structures, such as the cortical surfaces, the grey- / white-matter interface and the outside of the scalp.

The algorithm we have developed for generating an efficient separating surface representation from the segmented volume has five phases: component generation, component absorption, interface tracking, and surface fairing.

Component Generation: For the first phase, we process the segmented volume and generate a connected component list, where a single connected component consists of all the voxels of the same material that are face-connected (an individual voxel has six face-connected neighbors).

Component Absorption: Next, the connected components list is culled to remove components with very small volumes—these components generally correspond to noise in the original MR data or misclassifications from the segmenter. While they have little impact on the global conductivity of the system, they induce unnecessarily refined regions of the mesh if they are not removed. These small components are culled by "absorption," meaning they are added to the largest neighboring component. Before absorption, we can have up to 10,000 components in a 256-cubed segmented cranial volume. Following the absorption of all components containing less that 20 voxels (less than one cubic millimeter), we can reduce this count to approximately 30 components.

Interface Tracking: The third step in surface extraction is to track the interfaces between components and use them to generate a surface hierarchy of oriented surfaces. This data structure, called a *SurfaceTree*, enables us to rapidly classify elements during mesh generation, and is a compact representation for the important features of the volume.

Surface Fairing: Since the original volume consisted of small cuboid voxels, the initial SurfaceTree consists of many small axis-aligned rectangles. This blocky structure does not accurately represent the fine-grained structure of the real data (*i.e.,* the patient's head), and induces singularities in the finite element method. We smooth out the surfaces by exchanging the rectangles for triangles and applying a variation of Taubin's surface fairing algorithm [16]. This algorithm, extended to handle the non-manifold topology of the SurfaceTree, iteratively applies local-neighborhood attraction and repulsion operations to smooth the surface without significantly reducing the bounded volume.

Mesh Generation

The separating surfaces are then passed into our variational Delaunay tetrahedrization module [4]. The volume mesh generator accepts as input the faceting of the external and interface constraining surfaces, providing for each facet the regions bounded by its two sides. Additional points may be generated inside the solid regions by octree-driven insertion.

The variational Delaunay approach is based on the premise that the triangulation of a set of points in a three-dimensional space by an *unconstrained* Delaunay algorithm is an efficient and robust procedure, provided the points are in general positions. Furthermore, it can be done efficiently both on serial computers and in parallel. Therefore, we formulate our algorithm in such a way as to modify the constraining surfaces so as to appear as a collection of tetrahedral faces *a priori*, i.e. before the volume mesh generation is initiated. This is at variance with the established approaches based on *a posteriori* editing of the volume mesh. To assess the presence of a constraining facet in the volume mesh as a tetrahedron face, we use, among other checks, the result of Rajan [14], which formulates the construction of a Delaunay tetrahedron as a linear programming problem.

The *a priori* modification of the constraining boundary facets is done in a topology-preserving way by applying two operators: edge flip, and edge split. This constitutes a heuristic technique in that unless some restrictions are placed on the input surface triangulations, a proof of termination is not available. However, in the case of our cranial MR data sets, the generator performs reliably and efficiently.

Results

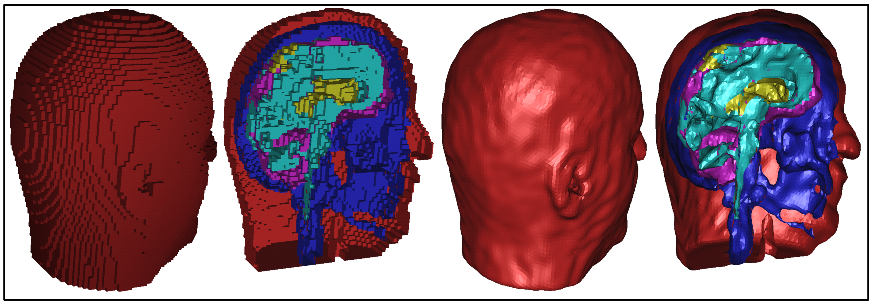

Figure 3: From left to right, we show the outside surface of the head and a cut through of the internal surfaces, first shown before smoothing, and then after smoothing.

The steps of our algorithm are illustrated in Figures 3 and 4. An MRI volume of a patient's head was segmented using the Brigham and Women Hospital's E-M segmenter [18]. We resampled the segmented volume to a resolution of 64 × 64 × 64 voxels. The resulting volume contained 99 connected components; following our absorption algorithm, this number was reduced to 8. Extracting the separating surfaces of those 8 components resulted in the jaggy surfaces shown in the left side of Figure 3. Applying our smoothing algorithm, we removed the terracing artifacts, producing the smooth surfaces shown on the right side of Figure 3. These smoothed separating surfaces contained a total of 92032 triangles. Passing the smoothed surfaces into our mesh generator, we produced a tetrahedral mesh containing 72745 nodes and 406493 elements.

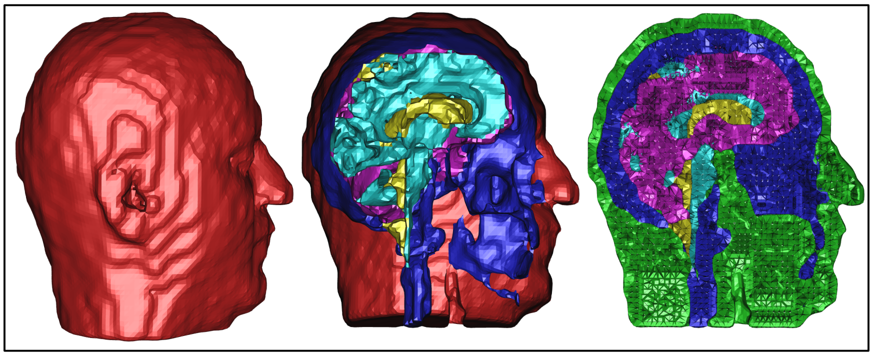

Figure 4: From left to right, we show the outside surface of the head, a cut through of the internal surfaces, and a cut through of the finite element mesh.

The above modeling pipeline is currently being used to construct patient-specific computational models for cortical mapping and source localization studies. Results produced using this system include new algorithms for source localization [17, 19], as well as cognitive neuroscience studies using this system [8, 12, 13]. It is our hope that such a modeling pipeline will ultimately be a central component in neuroscience tools for analysis, diagnosis, and surgical planning.

Acknowledgments

This work was supported in part by awards from the Department of Energy, the National Institutes of Health NCRR Program and the National Science Foundation.

References

[1] M. Burnett, R. Hossli, T. Pulliam, B. VanVoorst, and X. Yang. Toward visual programming languages for steering scientific computations. *IEEE Computational Science and Engineering*, 1(4):44–62, 1994.

[2] S.F. Faux, R.W. McCarley, P.G. Nestor, M.E. Shenton, S.D. Pollak, V. Penhume, E. Mondrow, B. Marcy, A. Peterson, T. Horvath, and K.L. Davis. P300 topographic assymetries are present in unmedicated schizophrenics. *Electroencephalography and clinical Neurophysiology*, 88:32–41, 1993.

[3] A. Gevins, J. Le, P. Brickett, B. Reutter, and J. Desmond. Seeing through the skull: Advanced EEGs use MRIs to accurately measure cortical activity from the scalp. *Brain Topography*, 4(2):125–131, 1991.

[4] P. Krysl and M. Oritz. Variational delaunay approach to the generation of tetrahedral finite element meshes. *International Journal for Numerical Methods in Engineering*, (to appear), 1999.

[5] J. Le and A. Gevins. Method to reduce blur distortion from EEG's using a realistic head model. *IEEE Trans. Biomed. Eng.*, 40(6):517–528, 1993.

[6] B. Lutkenhoner, E. Menninghaus, O. Steinstrater, C. Wienbruch, M. Gissler, and T. Elbert. Neuromagnetic source analysis using magnetic resonance images for the construction of source and volume conductor model. *Brain Topography*, 7(4):291–299, 1995.

[7] G. M. Nielson and R. Franke. Computing the separating surface of segmented data. In *Visualization '97*, pages 229–233. IEEE Press, 1997.

[8] B.F. O'Donnell, R.W. McCarley, G.F. Potts, D.F. Salisbury, P.G. Nestor, Y.O. Hirayasu, M.A. Niznikiewicz, J. Barnard, Z.J. Shen, D.M. Weinstein, F. Bookstein, and M.E. Shenton. Identification of electrical sources underlying p300 abnormalities in schizophrenia. In *Psychophysiology*, page (in press), 2000.

[9] S.G. Parker and C.R. Johnson. SCIRun: A scientific programming environment for computational steering. In *Supercomputing '95*. IEEE Press, 1995.

[10] S.G. Parker, M. Miller, C.D. Hansen, and C.R. Johnson. An integrated problem solving environment: the SCIRun computational steering system. In Hesham El-Rewini, editor, *Proceedings of the 31st Hawaii International Conference on System Sciences (HICSS-31)*, volume VII, pages 147–156. IEEE Computer Society, January 1998.

[11] S.G. Parker, D.M. Weinstein, and C.R. Johnson. The SCIRun computational steering software system. In E. Arge, A.M. Bruaset, and H.P. Langtangen, editors, *Modern Software Tools in Scientific Computing*, pages 1–40. Birkhauser Press, 1997.

[12] G.F. Potts, D.M. Weinstein, B.F. O'Donnell, M.E. Shenton, C.R. Johnson, and R.W. McCarley. Bioelectric modeling of the p300 in schizophrenia. In *Biological Psychiatry (suppl.)*, page 396, 1998.

[13] G.F. Potts, C.G. Wible, M.E. Shenton, D.M. Weinstein, I.A. Fisher, M.E. Levention, L.D. Gugino, and R.W. McCarley. Localization of visual cortex with coregistered functional magnetic resonance imaging, bioelectrically modeled cortical visual evoked potentials, and transcranial magnetic stimulation induced visual suppression. In *Journal of Cognitive Neuroscience (suppl.)*, page 42, 1998.

[14] V. T. Rajan. Optimality of the Delaunay triangulation in R^d. *Discrete Comput. Geom.*, 12(202):189, 1994.

[15] M. Scherg. *Auditory Evoked Magnetic Fields and Potentials. Advances in Audiology*, chapter Fundamentals of dipole source analysis, pages 1–30. S. Karger AG, 1990.

[16] G. Taubin. A signal processing approach to fair surface design. In *ACM SIGGRAPH Computer Graphics*, pages 351–358. IEEE Press, 1995.

[17] D.M. Weinstein, L. Zhukov, and C.R. Johnson. Lead field basis for fem source localization. *University of Utah Technical Report*, UUCS-99-014, 1999.

[18] W.M. Wells, W.E.L. Grimson, R. Kikinis, and F.A. Jolesz. Statistical intensity correction and segmentation of MRI data. In *Visualization in Biomedical Computing*, pages 13–24, 1994.

[19] L. Zhukov, D.M. Weinstein, and C.R. Johnson. Reciprocity basis for eeg source imaging. In *NeuroImage*, page S598, 2000.