SCI INSTITUTE
TECHNICAL REPORT

The Transfer Operator for Multigrid on Inverse Bioelectric Field Problems

Kris Zyp

UUSCI-2003-001

Scientific Computing and Imaging Institute
University of Utah
Salt Lake City, UT 84112 USA

May 2003

Abstract:

This thesis describes an accurate transfer operator for the implementation of multigrid on the inverse bioelectric field problem. The traditional computational methods that apply to bioelectric field problems often lead to inaccurate and inefficient solutions due to the ill-posed nature of the inverse problem.

Multigrid is a computational technique that researchers have effectively applied to a large number of computational simulations. It has an inherent regularization property and can achieve linear or near-linear convergence. When applied to the inverse bioelectric field problem, multigrid rapidly calculates internal bioelectric fields. The ill-posed nature of the inverse problem causes inaccuracies to infiltrate the multigrid coarse grid correction scheme, significantly challenging this application of multigrid. The complexity of accurately modeling the body also adds to this level of difficulty. However, with proper implementation, multigrid's fast convergence improves the efficiency of solving this problem.

The research in this thesis examines the accuracy of several methods for determining the transfer operator that drives the multigrid coarse grid correction. Also included is a proposal for a semi-coarsening strategy, called multistep, that facilitates accurate coarse grid correction, necessary for multigrid application to the inverse bioelectric problem. This thesis also contains tests of these methods on several datasets and a comparison and review of these test results, showing how multistep yields the greatest accuracy. The final section includes suggestions for an appropriate set of methods for future inverse bioelectric multigrid implementations.

THE UNIVERSITY OF UTAH

THE TRANSFER OPERATOR FOR MULTIGRID ON INVERSE BIOELECTRIC FIELD PROBLEMS

by

Kris Zyp

A thesis submitted to the faculty of
The University of Utah
in partial fulfillment of the requirements for the degree of

Master of Science

in

Computer Science

School of Computing

The University of Utah

May 2003

THE UNIVERSITY OF UTAH GRADUATE SCHOOL

SUPERVISORY COMMITTEE APPROVAL

of a thesis submitted by

Kris Zyp

This thesis has been read by each member of the following supervisory committee and by majority vote has been found to be satisfactory.

	Chair: Christopher R. Johnson
	Ross T. Whitaker
	Robert S. MacLeod

THE UNIVERSITY OF UTAH GRADUATE SCHOOL

FINAL READING APPROVAL

To the Graduate Council of the University of Utah:

I have read the thesis of Kris Zyp in its final form and have found that (1) its format, citations, and bibliographic style are consistent and acceptable; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the Supervisory Committee and is ready for submission to The Graduate School.

Date

Christopher R. Johnson
Chair, Supervisory Committee

Approved for the Major Department

Thomas C. Henderson
Chair/Dean

Approved for the Graduate Council

David S. Chapman
Dean of The Graduate School

ABSTRACT

This thesis describes an accurate transfer operator for the implementation of multigrid on the inverse bioelectric field problem. The traditional computational methods that apply to bioleletric field problems often lead to inaccurate and inefficient solutions due to the ill-posed nature of the inverse problem.

Multigrid is a computational technique that researchers have effectively applied to a large number of computational simulations. It has an inherent regularization property and can achieve linear or near-linear convergence. When applied to the inverse bioelectric field problem, multigrid rapidly calculates internal bioelectric fields. The ill-posed nature of the inverse problem causes inaccuracies to infiltrate the multigrid coarse grid correction scheme, significantly challenging this application of multigrid. The complexity of accurately modeling the body also adds to this level of difficulty. However, with proper implementation, multigrid's fast convergence improves the efficiency of solving this problem.

The research in this thesis examines the accuracy of several methods for determining the transfer operator that drives the multigrid coarse grid correction. Also included is a proposal for a semicoarsening strategy, called multistep, that facilitates accurate coarse grid correction, necessary for multigrid application to the inverse bioelectric problem. This thesis also contains tests of these methods on several datasets and a comparison and review of these test results, showing how multistep yields the greatest accuracy. The final section includes suggestions for an appropriate set of methods for future inverse bioelectric multigrid implementations.

CONTENTS

ABSTRACT . . . . . **iv**

LIST OF TABLES . . . . . **vii**

LIST OF FIGURES . . . . . **viii**

ACKNOWLEDGMENTS . . . . . **xi**

CHAPTERS

1. INTRODUCTION **1**

1.1 Motivation 1
1.2 Thesis Statement 2
1.3 Validation 2
1.4 Overview of the Thesis 2

2. BACKGROUND **4**

2.1 Bioelectric Fields 4
2.2 ECG/EEG Inverse Problems 4
2.2.1 Ill-Posedness 5
2.2.2 Models 6
2.2.3 Bioelectric Source Formulation 6
2.2.4 Model Construction 7
2.2.5 Mesh Generation 8
2.2.6 Regularization 11
2.2.7 Dipole Model 14
2.2.8 Current Challenges 15
2.3 Multigrid 15
2.3.1 Algebraic Multigrid 24
2.3.2 Other Inverse Multigrid Work 25

3. METHODS **27**

3.1 Applying Algebraic Multigrid to the Inverse ECG Problem 27
3.1.1 The Inverse Effect 30
3.2 Improved Coarsening 32
3.3 Multistep Approach 35

4. SIMULATIONS **43**
4.1 Test Procedure 47

5. RESULTS AND DISCUSSION **51**
5.1 Test Sets and Data 51
5.2 Discussion 61
5.3 Scalability 71
5.4 Future Work 75
5.5 Conclusion 77

REFERENCES **79**

LIST OF TABLES

5.1 Test set 1 - 541 Nodes 53

5.2 Test Set 1 - 541 Nodes - Test of error robustness with varying input noise addition. 54

5.3 Test set 2 - 7186 Nodes - Standard tests 54

5.4 Test Set 2 - 7186 Nodes - Auxillary tests using CGNE for the final solve. 55

5.5 Test set 3 forward problem - 7186 Nodes 57

5.6 Test set 4 - 300 Nodes 58

5.7 Test set 5 - 1195 Nodes 59

5.8 Test set 6 - 2692 Nodes 61

5.9 Test set 7 - 4789 Nodes 62

5.10 Test set 8 - 156,728 Nodes 63

LIST OF FIGURES

2.1 Diagram of the torso for source formulation. 8

2.2 Visualization of the interior mesh for the test set 2 problem (Utah torso) [74]. 9

2.3 The progression of full multigrid through the coarse grid levels. 18

2.4 Example of coarse grid selection. The large points are the coarse grid points and the smaller points are the fine grid points. 19

2.5 A sample problem at level 1 (coarsest level) uses a 3x3 grid and is solved directly. 19

2.6 A sample problem at level 2 uses a 5x5 grid. The graph shows values before smoothing. 20

2.7 A sample problem at level 2 uses a 5x5 grid. The graph after smoothing. 20

2.8 A sample problem at level 3 uses a 9x9 grid. The graph shows values before smoothing. 21

2.9 A sample problem at level 3 uses a 9x9 grid. The graph shows values after smoothing. 21

2.10 A sample problem at level 4 uses a 17x17 grid. The graph shows values before smoothing. 22

2.11 A sample problem at level 4 uses a 17x17 grid. The graph shows values after smoothing. 22

2.12 A sample problem at level 5 uses a 33x33 grid. The graph shows values after smoothing. 23

2.13 A sample problem at level 5 uses a 33x33 grid. The graph is the directly solved solution. 23

3.1 Mesh of the 541 node sample problem. 31

3.2 The result of a coarse grid solve on the forward test problem. The x-axis is position on the heart, y-axis is potential. 32

3.3 The result of a coarse grid solve on the inverse test problem. The x-axis is position on the heart, y-axis is potential. 33

3.4 Nodes are reduced over several steps with semicoarsening from 541 nodes to 399 nodes to 328 nodes to 283 nodes. This is the initial grid. 37

3.5 Semicoarsening reduction after one step. 37

3.6 Semicoarsening reduction after two steps. 38
3.7 Semicoarsening reduction after three steps. 38
3.8 Example 4 node "subcircuit" within a mesh. 39
3.9 Example 3 node "subcircuit" within a mesh. The inner node (node 0) has been removed, and equivalent resistance for the remaining resistors can be computed. 39
4.1 The number of nonzeros (y-axis) at the coarsest level versus weak connection tolerance level (x-axis) for test set 2. 45
4.2 The weak connection tolerance level's effect on performance. The setup time and the solve time (y-axis) plotted versus the weak connection tolerance level on the x-axis for test set 2. 46
4.3 The weak connection tolerance level's effect on accuracy. The error in the solution (y-axis) plotted versus the weak connection tolerance level on the x-axis for test set 2. 47
4.4 One of the random vectors used as the for epicardial potentials for test set 4. The remaining computed potentials are then based upon this test vector. The x-axis represents position on the heart, y-axis represents potentials. 49
5.1 Mesh used for test set 1. 52
5.2 Interior view of the model for test set 2. 56
5.3 Mesh constructed for test set 4. 58
5.4 Mesh constructed for test set 6 problem. 60
5.5 Performance scalability for multistep multigrid. Multigrid setup time (y-axis) in seconds is plotted versus the number of nodes on the x-axis for problems solved using the multistep algorithm on test sets 4-7. . . . 64
5.6 Test set 1 after one cycle (no smoothing). The colors represent potentials throughout the torso. 65
5.7 Test set 1 after one cycle (no smoothing). The colors represent the error in the computation (15% RMS error). 66
5.8 Test set 1 after one cycle and postsmoothing (with three smooths). The colors represent potentials throughout the torso. 66
5.9 Test set 1 after one cycle and postsmoothing (with three smooths). The colors represent the error in the computation (13% RMS error). . 67
5.10 Test set 1 after three cycles (three smooths per cycle). The colors represent potentials throughout the torso. 67
5.11 Test set 1 after three cycles (three smooths per cycle). The colors represent represent the error in the computation (5.6% RMS error). . . 68

5.12 Test set 1 correct solution. The colors represent potentials throughout the torso. 68

5.13 The tradeoff between accuracy and performance of the different algorithms. Setup time and solve time in seconds (y-axis) are plotted versus the percentage error in the solve on the x-axis for the different algorithms applied to test set 7. 70

5.14 Performance scalability for standard multigrid. Multigrid setup time in seconds (y-axis) is plotted versus the number of nodes (x-axis) for problems solved using the multigrid standard algorithm on test sets 4-7. 71

5.15 The solve time (y-axis) versus the number of nodes (x-axis) in the problem using the conjugate gradient algorithm for test sets 4-7. 72

5.16 The number of nodes and the number of nonzeros (y-axis) versus the depth (x-axis) during the coarsening steps for test set 8. 73

5.17 The effect of problem size on accuracy. The error in the solution (y-axis) is plotted versus the number of nodes (x-axis) for test sets 4-7 75

ACKNOWLEDGMENTS

I would like to thank my advisor, Chris Johnson, for all his support, encouragement, advice, and ideas for getting me through my master's degree. I certainly also appreciate the countless hours he spent revising my many drafts and enduring my constant nagging. His help has been invaluable to my success. I want to thank Dave Weinstein for his help and tutoring that introduced me to inverse problem solving. He also greatly assisted me with his ideas and by giving me problem sets to work with.

I want to thank Ross Whitaker and Rob McLeod for their advice and assistance on improving my thesis as well. I am very grateful for all of Blythe Nobleman's revisions and corrections on my paper. The SCI administrative staff, Heidi, Allyne, Raelynn, and the rest have always been wonderfully helpful, too.

I want to thank Marcus Mohr and Ulrich Rüde for their partnership in applying multigrid to the inverse bioelectric field problem. I also want to acknowledge Craig Douglas's bibliography [26].

I want to thank my parents and sister for all their love, support, and prayers as I have been working on my master's degree. Finally, I want to thank God for all He has done for me, and for giving me the ability and strength to make it through this great challenge.

CHAPTER 1

INTRODUCTION

1.1 Motivation

The inverse bioelectric field problem determines the electrical activity within the body based upon measurements on the surface and is an important problem for electrocardiography (ECG) and electroencephalography (EEG). Accurate EEG analysis could be used to locate current sources that trigger seizures in focal epilepsy patients. It could also reveal topographic asymmetries in the auditory P300 response as a result of left posterior superior temporal gyrus volume reduction in the brains of schizophrenics [49, 78, 93, 94, 114]. Such analysis could also guide functional magnetic resonance imaging (fMRI) acquisition for further investigations. Accurate solutions to the inverse ECG problem could assist in the determination of patterns of excitation [24], recovery of excitability, evaluation of myocardial ischemia [72, 73, 75], the localization of ventricular arrhythmias and the site of accessory pathways in Wolff-Parkinson-White (WPW) syndrome [36, 90, 106]. The technology to perform body surface potential mapping is reliable and is relatively inexpensive. Harnessing this technology to accurately determine the electrical activity within the body is a much more difficult problem. Computational methods for this problem yield slow convergence and are inaccurate.

The technique of multigrid is an iterative method for solving linear partial differential equations (PDE). Multigrid contributes to solving a number of physical simulation problems with improved computational efficiency by systematically removing the different frequency bands of noise. Applying multigrid to the inverse bioelectric field problem could benefit inverse problem solving by providing near linear time convergence and improvements to the solution accuracy. Linear time convergence occurs when the time required for convergence is proportional to n, the

size of problem, rather than n^2, which is generally necessary with other iterative methods.

1.2 Thesis Statement

By using more accurate transfer operators, we can use multigrid to efficiently and accurately solve the inverse bioelectric field problems. Typically, multigrid has been applied to well-posed problems and the transfer operators are sufficiently accurate for the interpolation and restriction steps. However, the inverse bioelectric field problem is ill-posed and the multigrid algorithm will not function correctly if applied in the standard manner. Until now, no one has developed a transfer operator to handle the difficulties of this problem. In this thesis, a more accurate transfer operator is introduced and applied in order to assist in the difficulty of applying multigrid to an ill-posed problem with a high level of sensitivity to errors.

1.3 Validation

This research will examine several different algorithms, including a new multi-step approach, and test them on multiple bioelectric problems to determine their proficiency and accuracy. The inverse problem presents unique difficulties with maintaining accuracy. In this thesis, we will choose from these algorithms by comparing their precision in each of the steps in the multigrid process of computation and evaluating how each algorithm handles the high error sensitivity, and affects the overall accuracy of the system. This comparison will give future researchers an analysis of which transfer operator implementation is necessary for successful multigrid implementation in the inverse bioelectric field problem.

1.4 Overview of the Thesis

This thesis begins with a brief review of the inverse problem in electrocardiography and outlines the significant techniques we have used to solve the problem. The next section provides an overview of the multigrid method, the concept behind it, and how it is implemented. After the multigrid principles are established, the thesis demonstrates how to apply multigrid to the inverse bioelectric problem and

the methods used to overcome the inherent challenges of this ill-posed problem. The methods section also describes a new multistep semicoarsening approach that increases the accuracy of the multigrid algorithm for ill-conditioned problems. The next section is an explanation of the test procedure and the research test results. Finally, the thesis ends with a summary of the results and their interpretations.

The test results demonstrate that multigrid has the capability to perform coarse grid approximations of the inverse problem with an appropriate transfer operator. Though different transfer operator algorithms prove better adapted to different situations, the linear convergence of multigrid shows great scalability and the multistep coarsening algorithm is the only transfer operator with sufficient accuracy to perform the solves. While the multigrid algorithm that I implemented did not have ability to reliably perform accurate large-scale full inverse solves due to ineffective relaxation methods, the data from the intermediate steps revealed conclusive information. The test results showed that a multistep coarsening approach is necessary to provide the coarse grid correction accuracy necessary to make multigrid efficiently solve the inverse bioelectric field problem.

CHAPTER 2

BACKGROUND

2.1 Bioelectric Fields

Within the body, the majority of bioelectric fields originate from two different sources: the heart and the brain. The ECG inverse problem and the EEG inverse problem of calculating the internal electrical potentials share many techniques, but this research will focus on the ECG problem.

The determination of internal electric potentials relies on computationally simulating these electric activities. Such computational simulations of bioelectric fields are based upon the biophysics of excitable membranes [92]. These membranes provide the elemental ionic current for all bioelectric phenomena. Electrical potentials arise within the conductive volume surrounding the excited membranes due to the integrated interactions of these currents. These potentials can be measured in variety of ways including cortical surface, heart surface, scalp, torso, and intravenous measurements. A high resolution technique for measuring potentials, *mapping,* uses a greater number of measurement sites (32-128 for the scalp or 32-200 for the torso) than typically used (3-16) for clinical applications. Mapping allows a more accurate analysis of the spatial distribution of electrical activity within the body[40, 81].

2.2 ECG/EEG Inverse Problems

The process of determining these bioelectric fields based upon surface measurements is an inverse problem. There has been a substantial amount of work on the use of computer simulation to solve this problem [1, 5, 6, 29, 31, 32, 42–44, 52, 53, 58, 61–63, 66, 75, 79, 86–88, 99, 100, 102, 106, 113, 119, 121, 124]. Applied to the brain, the goal of the inverse problem is to locate the point sources within the brain that deliver electric current by using EEG data. For the heart, the problem is similar,

trying to determine the tissue where electrical currents originate based on ECG data. A number of ways to determine these internal electrical sources are more invasive, but inverse analysis seeks to avoid these intrusions. These approaches are very important for verification of the noninvasive methods. Currently, the clinical use of ECG is based more on empirical pattern recognition than on biophysical modeling. The goal of inverse ECG problem solving is to improve the accuracy of biophysical modeling for clinical usage. Research has demonstrated that electrocardiography can help characterize and detect myocardial infarction [21, 22] and myocardial repolarization abnormalities [20].

Use of the inverse ECG problem has not yet achieved the adequate level of spatial resolution for widespread clinical use. Resolution in the range of 5–10 mm with less then 10% error is required to be useful for applications such as describing reentrant tachycardias, evaluating myocardial ischemia, or localizing ventricular arrythmias [72, 75]. Such spatial resolution is difficult to achieve because a unique solution for current sources, based upon surface measurements, does not exist. This has led to a number of different approaches for modeling the electrical sources [44].

The root reason for the lack of adequate spatial resolution is the dissipation of the electrical signal as it moves through the body, which leads to attenuation and spatial smoothing. Both of these must be reversed to recover the internal signals. This reversal process makes the solution very sensitive to noise, computational precision loss, and model inaccuracies.

2.2.1 Ill-Posedness

A well-posed problem, as defined by Hadamard [48], is one in which all the following criteria are met:

1. For each set of data, there exists a solution.

2. The solution is unique.

3. The solution depends continuously on the data.

As mentioned earlier, the inverse bioelectric field problem does not have a unique solution. In other words, different combinations of internal current sources can lead to the same observed external potentials. This means that the problem is ill-posed. Also, large changes in the internal sources can lead to only extremely small changes in the external potentials so the solution does not depend continuously on the data. This high sensitivity means the discrete from of the problem is ill-conditioned. Therefore, it is necessary to constrain the problem, and the solution must be limited to a subspace. We can limit the subspace in such a way that there is only one solution within that space. We can simplify the model we use to describe the current sources within the body to achieve this limited subspace [41]. Also, we must constrain the domain so that the solution will depend continuously on the data. That is, the solution must be regularized.

2.2.2 Models

One approach for the modeling of electrical sources is to localize the source dipoles. A simple analysis of the standard electrocardiogram is based on a simple current dipole vector model of the heart's electrical activity. Dipole modeling can be performed with this simple single dipole approach or using multiple dipoles. However, without constraints only a few dipoles can be reliably estimated. Alternatively, a more realistic, but computationally difficult approach involves modeling the heart as a surface of dipoles [23, 52, 53], where the boundary between excited and resting cells represents the surface.

This research will focus on another approach that attempts to create a map of the potentials on the epicardium based upon surface values [98, 124]. This approach does not attempt to determine the locality of the sources within the heart, but only determines the surface potentials of the heart.

2.2.3 Bioelectric Source Formulation

The bioelectric field problem must be mathematically formulated before it can be solved. To begin, we assume a quasistatic condition, that is, even though the potentials and currents are changing with time, we assume at every instance of

time that electrical activity is constant and we can apply direct current analysis. The electrical activity can be described mathematically with a generalized Poisson's equation to solve in terms of the primary current sources:

$$\nabla \cdot \sigma \nabla \Phi = -I_v, \tag{2.1}$$

using the boundary conditions:

$$\sigma \nabla \Phi \cdot \mathbf{n} = \mathbf{0} \text{ on } \mathbf{\Gamma_T}, \tag{2.2}$$

where Φ are the electrostatic potentials, I_v are the cardiac current sources, σ is the conductivity tensor, and Γ_T represents the surface of the torso. We assume current sources exist only on the surface of the heart so Poisson's equation can be reduced to Laplace's equation:

$$\nabla \cdot \sigma \nabla \Phi = 0, \tag{2.3}$$

with the Cauchy boundary conditions:

$$\Phi = \Phi_0 \text{ on } \Sigma \subset \Gamma_T \tag{2.4}$$

$$\sigma \nabla \Phi \cdot \mathbf{n} = \mathbf{0} \text{ on } \mathbf{\Gamma_T}. \tag{2.5}$$

This formulation represents the surface-to-surface inverse bioelectric field problem as represented in Figure 2.1.

While we now have a unique solution, the problem is still highly sensitive to initial conditions, and thus the problem is still ill-conditioned.

2.2.4 Model Construction

In order to solve for the epicardial potentials, we must construct a model by measuring and discretizing the anatomy. A noninvasive method of measurement for living beings is generally preferable. Most models are based on magnetic resonance imaging (MRI) or computed tomography (CT) images [112].

Next, the scanned images must be segmented. Unfortunately, fully automated segmentation does not always produce acceptable results, so the segmentation process often requires user guidance. Generally, users must choose control points

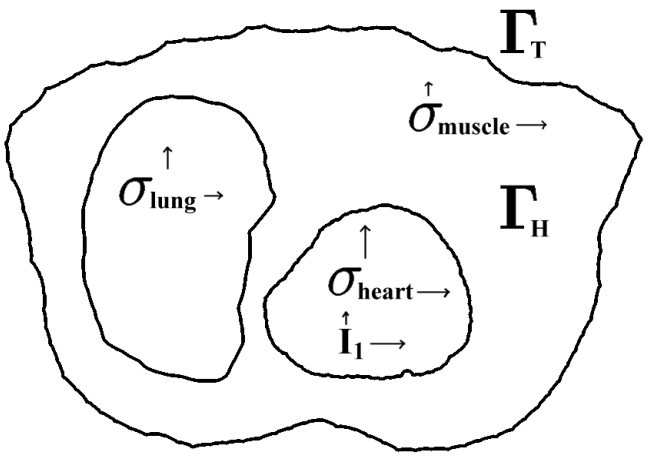

Figure 2.1. Diagram of the torso for source formulation.

outlining the individual objects of interest within the body. There are a number of techniques for attempting to automatically segment the images, and assist in manual segmentation [59]. These include edge detection [111], threshold [108], edge-based segmentation [71], and region growing [125].

From these segmentations, we must create surface boundaries. A triangulated surface is usually preferable for describing the irregular surfaces of the body. Although it is difficult to create surfaces from an irregular set of points, most scanning techniques yield points that are constrained to planes, and we can use a relatively simple lacing technique to create boundary surfaces [76].

2.2.5 Mesh Generation

Once a surface is created, we must generate a mesh for the volume. In order to solve for the epicardial potentials, the partial differential equations (PDE) must be approximated numerically on a grid. First, we must choose the discretization of the geometry from scanned images. We create a mesh of nodes that are connected to each other to form the connections and elements as displayed in Figure 2.2. Mesh generation is a large field in itself, but there are a few predominant strategies for the generation of nodes and elements. Structured partitioning is the simplest

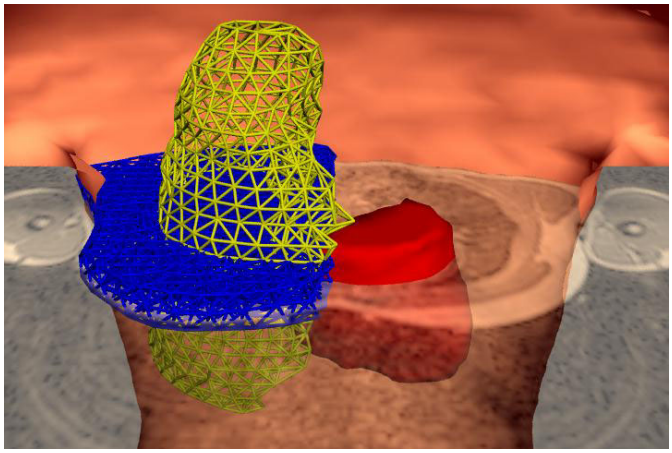

Figure 2.2. Visualization of the interior mesh for the test set 2 problem (Utah torso) [74].

method of generation [35, 70]. This strategy begins with a given boundary and recursively divides it. The divisions are then subdivided repeatedly in a divide and conquer manner until we achieve an appropriate level of divisions. Straightforward programming is the advantage of this technique.

Delaunay triangulation is a more finely controlled method for the bioelectric field problem [35, 116]. This method begins with a given set of points that define the nodes. The nodes are then connected to form an optimal mesh of tetrahedra. This method allows the mesh to be generated to conform to the given regions and boundaries with the model and provide improved spacing. The disadvantage of this method is that it is more difficult to program. Other mesh generation techniques include mapping methods, octree methods [109, 110], and paving (advancing front methods) [35, 37].

Element spacing is another important decision in mesh generation. Although the simplest approach is to use an approximately constant distance between nodes in the mesh generation, the mesh can later be improved by adaptively refining the mesh based on a solution estimate [62, 103]. An algorithm called *h-adaptation* further refines the mesh, that is, creates more nodes in areas of greater potential

gradients. Alternatively, *p-adaptation* substitutes higher order elements in higher gradient areas. *Hp-adaptation* combines these techniques for improved adaptation [25, 85, 95].

Next, we must describe the physics of the problem in a numerical discretization. Methods for this discretization include the finite difference method (FDM), finite element method (FEM), or boundary element method (BEM). The finite difference method is based upon approximating derivatives on a uniformly spaced grid. The meshes generated for the bioelectric field problem can be unstructured (and they are in our case) and such meshes can not easily be used with the finite difference method. The boundary element method relies on mathematically computing the relationship between the nodes on the exterior of a volume. This method can provide a significant improvement in calculation time, as we do not have to compute interior nodes. Unfortunately, the mathematics to determine the boundary element equations are usually limited to isotropic volumes. The torso and brain have many anisotropic features. If we wish to accurately handle these features we must look to another method.

In comparison, the finite element method uses interpolation functions substituted into an equivalent integral equation for the problem [57]. This allows for the flexibility to formulate the equations from unstructured grids and handling of anisotropies. Therefore, we choose this method to discretize the continuous bioelectric field problem. We start with the equations from above, and use arbitrary test function $\overline{\Phi}$, to define boundary conditions:

$$\int_{\Omega} \sigma \nabla \Phi \cdot \nabla \overline{\Phi} \cdot d\Omega = - \int I_v \cdot \overline{\Phi} \cdot d\Omega, \tag{2.6}$$

where Ω represents the solution domain that must be discretized:

$$\Omega = \bigcup_{x=1}^{N} \Omega_x. \tag{2.7}$$

A discrete subspace must also be defined:

$$V_h \subset V = \overline{\Phi}. \tag{2.8}$$

The parameters of the function $\overline{\Phi} \in V_h$ are defined at node points, $\alpha_i \Psi(x_I)$, $i = 1, 2, ..., N$ and the basis function is defined as a linear piecewise continuous

function $\Psi_i \in V_h$ with a value of 1 at nodes and 0 elsewhere. This leads to the formulation for $\overline{\Phi} \in V_h$:

$$\overline{\Phi} = \Sigma_{i=1}^{N} \alpha_i \Psi(x_i), \tag{2.9}$$

where a linear combination of $\Psi_i \in V_h$ can be found to describe $\overline{\Phi} \in V_h$. We want to find $\Psi_h \in V_h$ to satisfy equation 2.9 and since Ψ_h can be a linear combination, we write:

$$\Psi_h = \Sigma_{i=1}^{N} \phi_i \Psi_i(x) \;\; \phi_i = \Phi_h(x_i). \tag{2.10}$$

Equation 2.6 can be rewritten:

$$\Sigma_{i=1}^{N} \phi_i \int_\Omega \sigma_{ij} \nabla \Psi_i \cdot \nabla \Psi_j \cdot d\Omega = - \int_\Omega I_v \cdot \Psi_j \cdot d\Omega \;\; j = 1, ..., N. \tag{2.11}$$

We now have the equation in the form necessary for linear system solve. That is, the finite element approximation has a set of N equations with N unknowns corresponding to the ϕ vector. The equation can be put in matrix form $A\phi = b$ where A is the stiffness matrix with elements $\alpha_{ij} = \int_\Omega \sigma_{ij} \nabla \Psi_i \cdot \nabla \Psi_j d\Omega$ and b is the source contributions with elements $b_i = - \int_\Omega I_v \cdot \Psi_i \cdot d\Omega$. Now the forward problem can be expressed in the form $A \cdot \phi = b$. The inverse bioelectric field problem requires further modification for multigrid application, to be discussed later.

2.2.6 Regularization

Next, the equations must be constrained with regularization. With discrete ill-posed problems, small changes in the input can cause unbounded changes in the output so the solutions often lack stability. Errors in measurements and calculations can lead to large errors in the solution. With regularization, we seek to make the solution continuously dependent on the input. It is therefore necessary to apply constraints in order to achieve a realistic solution. This can be as simple as minimizing the two-norm solution during the solve, but various other constraints can be applied as well, such as giving a priori estimate for the data, and using more sophisticated minimizations [39, 77, 117, 118].

The Tikhonov method is one of the most frequently used methods of regularization [117]. With this method, one attempts to minimize the residual $Ax - b$ and

a measurable constraint at the same time. An example of a constraint would be the root mean square of x. This leads to the following formulation:

$$(K^T \cdot K + \alpha \cdot I) \cdot x = K^T \cdot y. \tag{2.12}$$

We could also use a higher order constraint term that uses the gradient or the laplacian of x. Also multiple constraints can be combined into the constraint term, allowing greater flexibility in how we constrain the problem [15, 16]. Either way, we are no longer minimizing only a residual, but also the residual and the constraint term.

Another method for regularization is singular value decomposition [39]. The A matrix can be broken up:

$$A = U \cdot S \cdot V^T, \tag{2.13}$$

where U and V are orthogonal matrices and S is a diagonal matrix with the singular values as the diagonal entries. If the smaller entries in S are eliminated then we can create an A matrix that is better conditioned. Once factorization is complete, the system can then be solved:

$$A \cdot x = b \tag{2.14}$$

$$U \cdot S \cdot V^T \cdot x = b \tag{2.15}$$

$$x = (U \cdot S \cdot V^T \cdot)^{-1} \cdot b \tag{2.16}$$

$$x = (V^T)^{-1} \cdot S^{-1} \cdot U^{-1} \cdot b \tag{2.17}$$

$$x = V \cdot S^{-1} \cdot U^T \cdot b. \tag{2.18}$$

Limited iteration conjugate gradient is a similar method that eliminates smaller eigenvalues and their corresponding eigenvectors from the solution. When applying a conjugate gradient solver, we can stop before performing the full n iterations in order to keep only the major eigenvectors and obtain a smoother result [39, 118].

These methods require at least one parameter α to choose how much regularization to apply (for example, the magnitude of the measured constraint with Tikhonov or the minimum allowed singular value with SVD). This choice is very

important and can be a difficult decision to make. The L-curve method is one method to determine a reasonable *a priori* value of α [51]. An L-curve graphs the constraint term versus the residual (the residual and the constraint change with the regularization parameter) with a log-log scale. The corner of the L-curve, that is, where the curve begins to rise sharply, is used to choose the parameter, by choosing the point of maximum curvature.

The discrepancy principle is another method for choosing α. With this method, we choose a reasonable level for $|A \cdot x - b|_2$ and call this δ. We then minimize x while maintaining $|A \cdot x - b|_2 \leq \delta$. A possible choice δ is the median of the χ^2 distribution.

We can also use generalized cross validation to determine α [38, 77]. For a given α, we first apply Tikhonov regularization to $A' \cdot x = b'$ which is $A \cdot x = b$ without the ith equation. The resulting approximation is denoted $x_{\alpha,i}$. Then, we use $x_{\alpha,i}$ to estimate $b_i = (A \cdot x_{\alpha,i})_i$. We seek to find an α that minimizes the error of the above estimation.

Another, more recently developed method selects the corner based on a minimum distance function, that is, finding the point that is closest to the origin [7]. Huiskamp wrote a paper on choosing more physiologically based constraints rather then simple magnitude RMS or gradient based measurements [54].

Application of differing parameterized regularization to each submatrices is a recent approach for regularization [63, 64]. The submatrices that are used are discussed in the methods section of this paper. Each of these submatrices has widely different condition numbers. Therefore, it is logical to apply a different amount of regularization to each.

Rather than using the Tikhonov equation to minimize a cost function constraining solution, the admissible solution approach uses an optimization algorithm to find an solution that satisfies a set of user defined criteria. This approach allows greater flexibility, as multiple constraints can be employed and these constraints do not have to follow the Tikhonov formulation. Constraints can be defined on more physiologically meaningful measures [2].

Another method, borrowed from geophysics, for regularizing the solution is focusing inversion [84]. This method uses more advanced constraints that encourage localization of the sources.

Further, when multiple measurements are made at different times, we can apply a time dependent method. A time dependent method utilizes the data from different time steps to do more accurate analysis. Multiple signal classification (MUSIC) has borrowed from signal processing techniques to distinguish between multiple sources [104, 105]. An advanced form of MUSIC, called recursively applied and projected (RAP) MUSIC [82], iteratively improves on the source classification.

2.2.7 Dipole Model

When using a dipole model to estimate the sources, the typical approach is to guess the dipoles' locations and then perform a forward computation of the torso surface potentials for the given locations [34]. The error in the solution can be calculated and the dipoles can then be moved around to optimize the solution.

Creating a lead field can speed the process of calculating the surface potentials from the dipole location. A lead field is a transfer matrix that maps sources to their resultant surface potentials [17–19]. This lead field can be created by doing a forward solve of a dipole from each to be included in the lead field matrix. This can be limited to the epicardium nodes which is useful for epicardium potential to torso surface potential mapping, or can include surface potentials for every node within the heart. The latter is a desirable lead matrix for guessing at dipole locations. However, performing forward solves from every node within the heart is excessively time consuming. Fortunately, using reciprocity theory [27, 91] it is possible to only perform forward solves from each of the surface points of measurement and record the results for each of interior heart nodes to create the lead matrix [122]. Since the forward solve is being compared only to actual points of measurements, there are relatively few solves necessary.

2.2.8 Current Challenges

Regardless of which of these methods one chooses, there remains the challenge of solving a large-scale linear problem that involves an ill-conditioned matrix. The main difficulties with solving such problems are maintaining accuracy and solving quickly. Typically, one uses Krylov subspace methods, such as conjugate gradient, to solve these systems because of their rapid convergence and stability. However, the conjugate gradient method still requires $O(n^2)$ time for convergence (with preconditioning this can be lowered, although ill-conditioned problems can still force $O(n^2)$ time). As we pursue greater accuracy, it is necessary to work with larger numbers of nodes. These larger-scale problems become prohibitively slow with such an algorithm. As the need for spatial accuracy necessitates more nodes, the need for lower order solving that can still handle highly ill-conditioned systems becomes greater.

2.3 Multigrid

Multigrid is a method for solving linear systems. Multigrid creates lower resolution grids and estimates solutions on these grids as an approximation to higher resolution grids. This approach allows for a systematic removal of different frequencies of errors. For elliptic problems, multigrid can find a solution in linear time, and is easily parallelized [9, 14, 123].

In 1961, Fedorenko first introduced the concept of multigrid [28]. It was not until 1973, that Brandt realized its efficiency [9]. In 1976, Hackbusch independently developed the multigrid method as well [47]. Since then, multigrid has been applied to a vast array of applications, with just a few recent papers cited here, for example: Navier-Stokes [55], radiation-transport, diffusion [4], wave propagation, and image segmentation [107]. Unfortunately, multigrid can be a difficult to implement and, for certain problems, requires customization. Because of the ill-posed nature of the inverse bioelectric field problem, this problem requires customization of the algorithm.

The basic methodology of multigrid is as follows:

1. Relax $A_h u_h = f_h$ on the fine grid to obtain u_h
2. Compute residual $r_h = A_h u_h - f$
3. Transfer (restrict) residual to coarse grid $r_h \rightarrow r_H$
4. Relax on residual equation $A_H e_H = r_H$on coarse grid
5. Transfer (interpolate) approximation of error on coarse grid $e_H \rightarrow e_h$ to fine grid and correct solution: $u_h = u_h + e_h$

where h denotes the fine grid and H denotes the coarse grid. Relaxation, restriction, and interpolation are the important procedures involved in the multigrid process.

The relaxation (also known as smoothing) step reduces the high frequency component of the error. This is normally performed with a few iterations of an iterative solver. Jacobi and Gauss-Seidel are examples of solvers that work by first reducing the high frequency error and therefore they make effective smoothers. The most frequently used smoothers for multigrid are variations on the Gauss-Seidel method, such as successive over-relaxation, and different coloring schemes. An approximate inverse smoother has also shown promising results [120]. However, Krylov subspace methods like conjugate gradient are not typically useful as smoothers as they do not focus on the high frequency component. Note, this is different from regularization, in that regularization usually reduces the high frequency component of the computed solution while relaxation reduces the high frequency component of the error (the difference between the computed and true solution).

After relaxation, the restriction step attempts to make an accurate approximation of the fine grid residual on the coarse grid. The residual from the fine grid must be solved on the coarse grid in order to make a coarse grid approximation. There a residual vector corresponding to the coarse grid points must be generated from the fine grid residual. This is performed by taking a weighting average of the values of the fine grid points that are connected to a given coarse grid point.

Next, a solve or approximation is performed on the coarse grid level to create the coarse grid approximation for correction. After this, the coarse grid approximation

is interpolated (or prolonged) onto the fine grid to provide the error correction. The interpolation step reverses the restriction step to transform the coarse grid error correction onto the fine grid. The interpolation algorithm also utilizes a weighted average. The algorithm determines each value in the error correction vector at the fine grid by taking a weighted average of the coarse grid points that are connected to the given fine grid point. Generally, the restriction and prolongation operators are transposes of each other. The restriction R and interpolation I operators work as follows:

$$b_H = R \cdot b_h \tag{2.19}$$

$$b_h = I \cdot b_H. \tag{2.20}$$

$$\tag{2.21}$$

The restriction operator must be applied twice for the coarsening of the matrix:

$$A_H = R \cdot A_h \cdot R^T. \tag{2.22}$$

We want our weighting to be consistent with interpolation and restriction so:

$$R = I^T. \tag{2.23}$$

In addition, this is the restriction operator of the residual:

$$r_H = R \cdot r_h. \tag{2.24}$$

And this is the interpolation operator of the error correction:

$$e_h = I \cdot e_H. \tag{2.25}$$

Multigrid can then further be used in a nested approach, where the multigrid process does not only create one coarse grid, but creates successively coarser grids recursively to form a V-cycle. At the coarsest level, a direct solve is usually performed on the system. This cycle is repeated to iterate towards the solution. Multigrid cycles can also take the form of W-cycle, or with full multigrid, the V-cycle starts at the coarsest grid and grows in size until reaches the finest grid as

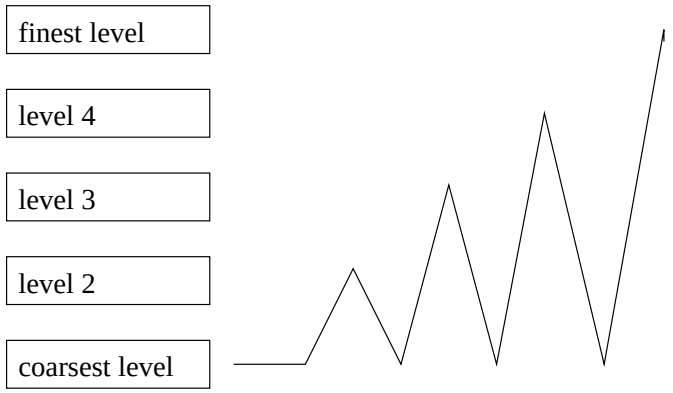

Figure 2.3. The progression of full multigrid through the coarse grid levels.

shown in Figure 2.3. Theoretically, full multigrid is necessary to achieve true linear convergence, otherwise convergence time is $O(log(log(n)) \cdot n)$ because of the growth of the depth of the cycles.

Before the multigrid cycle can begin, the coarse grid points must be chosen. Typically, the nodes chosen for the coarse grids are based upon the underlying geometry. Coarse grid points are preferably evenly spaced and chosen so as not to remove any fine grid points that are not connected to at least two coarse grid points. The goal is that the coarse grid should accurately reflect the original geometry. Figure 2.4 demonstrates a uniform two-dimensional grid with a nine-point stencil.

To demonstrate the multigrid process, we look at a simple elliptic problem with a point source and Newmann boundary condition. We have Laplace's equation:

$$\nabla^2 \Phi = 0, \tag{2.26}$$

and boundary conditions:

$$\nabla \Phi(x, y) \cdot \mathbf{n} = \delta(x, y) - 0.25. \tag{2.27}$$

This problem uses a nine-point stencil with bilinear interpolation. For simplicity, this problem starts at the coarsest grid and ascends through the levels as shown in Figures 2.5 - 2.13. At each level, we apply two iterations of Jacobi refinement.

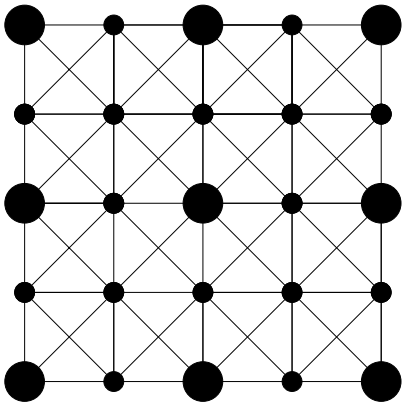

Figure 2.4. Example of coarse grid selection. The large points are the coarse grid points and the smaller points are the fine grid points.

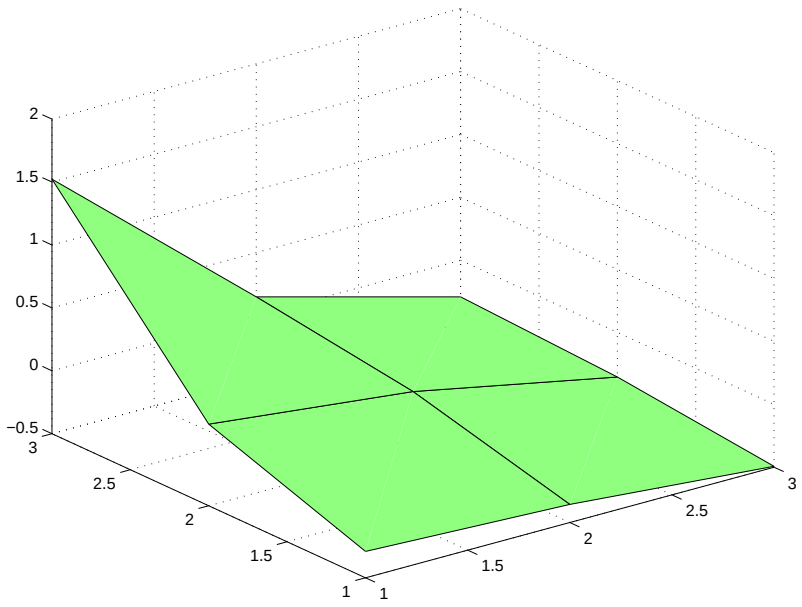

Figure 2.5. A sample problem at level 1 (coarsest level) uses a 3x3 grid and is solved directly.

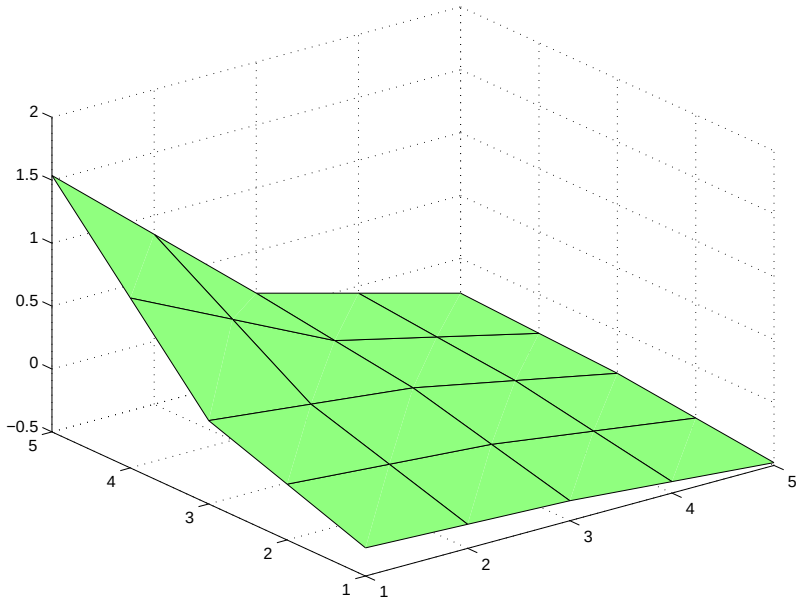

Figure 2.6. A sample problem at level 2 uses a 5x5 grid. The graph shows values before smoothing.

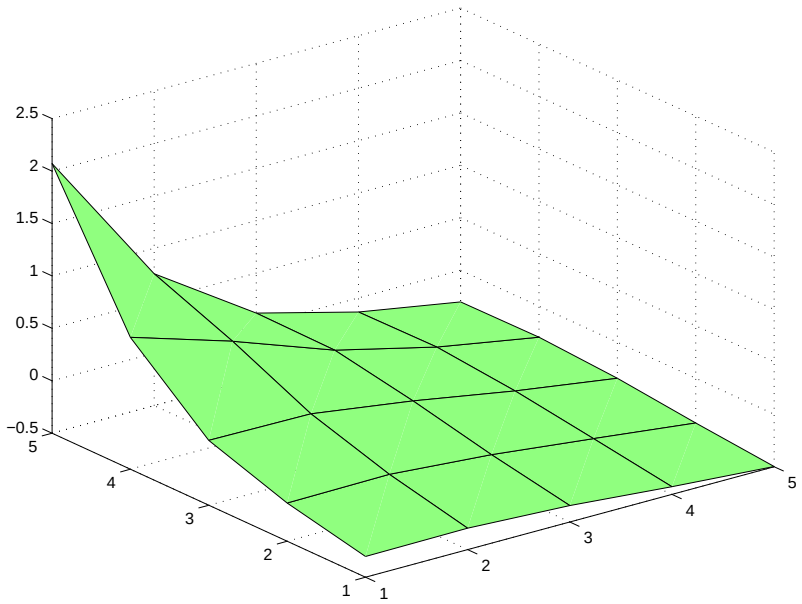

Figure 2.7. A sample problem at level 2 uses a 5x5 grid. The graph after smoothing.

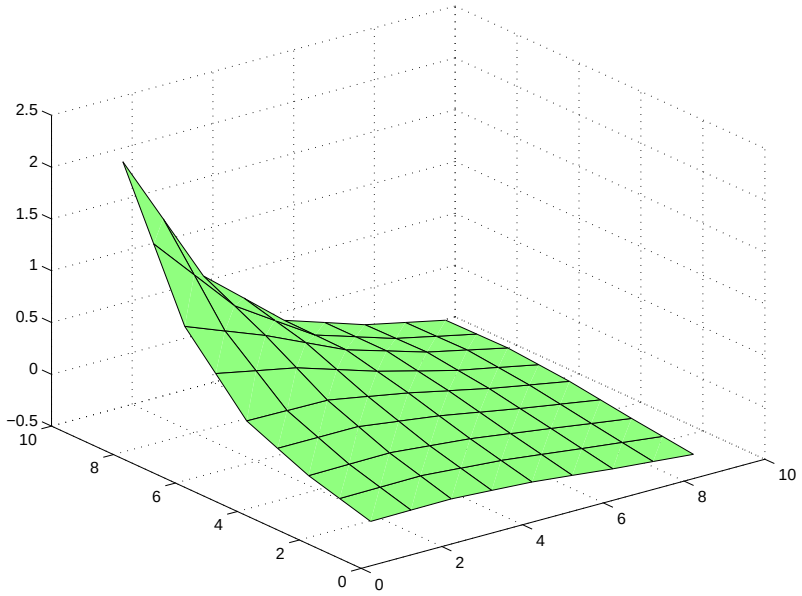

Figure 2.8. A sample problem at level 3 uses a 9x9 grid. The graph shows values before smoothing.

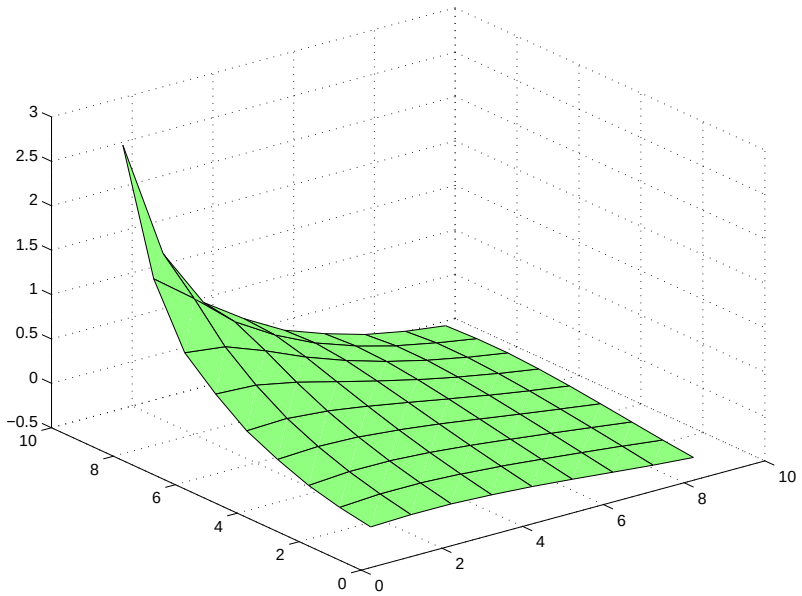

Figure 2.9. A sample problem at level 3 uses a 9x9 grid. The graph shows values after smoothing.

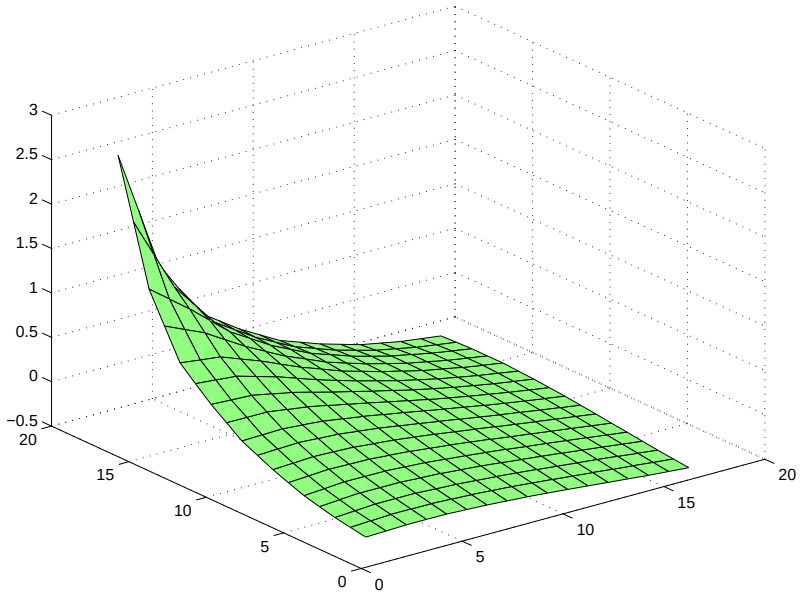

Figure 2.10. A sample problem at level 4 uses a 17x17 grid. The graph shows values before smoothing.

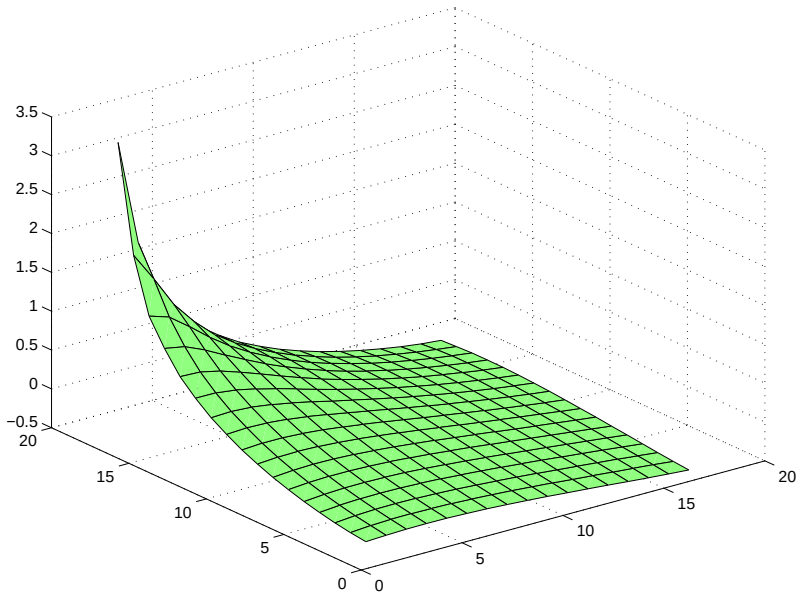

Figure 2.11. A sample problem at level 4 uses a 17x17 grid. The graph shows values after smoothing.

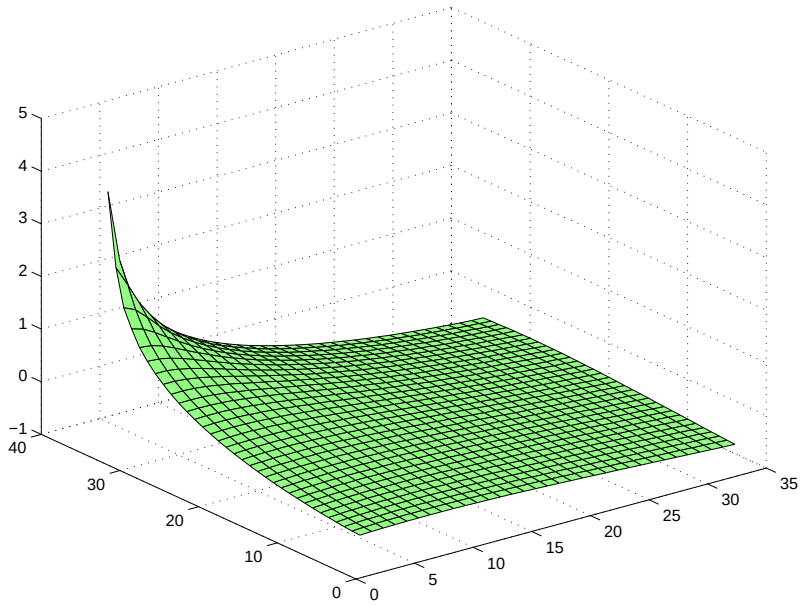

Figure 2.12. A sample problem at level 5 uses a 33x33 grid. The graph shows values after smoothing.

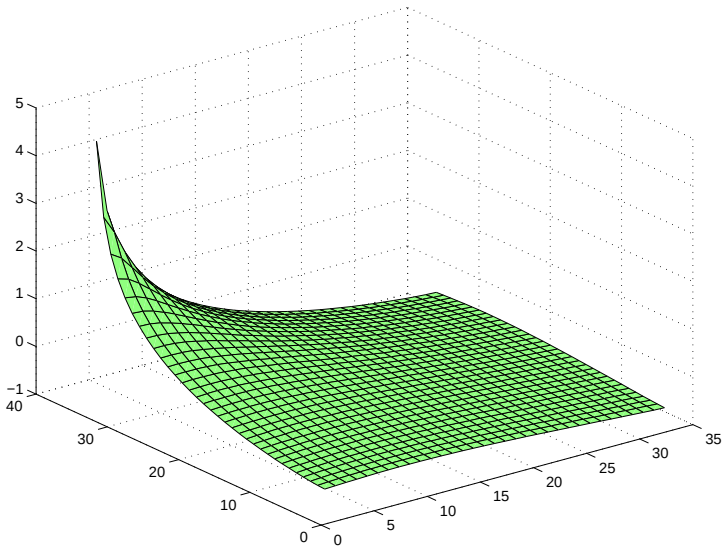

Figure 2.13. A sample problem at level 5 uses a 33x33 grid. The graph is the directly solved solution.

Rather than being used alone, multigrid is also often used as a preconditioner. This can take advantage of Krylov subspace convergence methods like preconditioned conjugate gradient, along with the power of multigrid in the form of preconditioner. With preconditioning, an approximate matrix inversion is applied to the residual at each iteration to improve the conditioning of the system. We can use multigrid to generate a coarse grid correction from the residual as a form of preconditioning to improve the conditioning of the problem, as well. Eigenvalue analysis shows multigrid preconditioning yields eigenvalues mostly clustered around one, and consequently, lowers the condition number. Although multigrid also tends to produce a few eigenvalues near zero, the conjugate gradient eliminates these quickly [115]. Thus, by improving the condition number of the system, multigrid can accelerate the conjugate gradient process. Also, since iterative methods like conjugate gradient automatically scale each vector correction, this allows for automatic scaling of the coarse grid correction, which can sometimes be too large or too small.

2.3.1 Algebraic Multigrid

Algebraic Multigrid (AMG) refers to using multigrid on matrices where the underlying geometry is not given [10, 12, 13]. The coarse nodes are then chosen purely on the given discretization matrix. This method can actually be much simpler in the case of unstructured meshes. Unstructured meshes can be difficult to work with when trying to choose a set of coarse grids based upon geometry, but the algorithms for AMG are often more robust in handling the irregularities of such meshes. It is difficult to perform mesh generation on complex geometries at different resolutions and ensure that each mesh is a properly registered subset of the finer meshes. The coarse grid creation is handled by AMG without the difficulty of reducing unstructured grids. One disadvantage of AMG is that it often requires more setup time when the geometry information could otherwise lead to a quick determination of appropriate coarse nodes. The choice of coarse grid points is still very important with AMG, especially in more difficult problems such as

unstructured meshes and anisotropic problems [8]. Coarse grid selection can be based purely on the graph of the matrix. However, higher quality coarse grids can be generated by using the connection strengths as well as the graph of the matrix, although incorporating connection strengths can be slower. Generally, it is advantageous to choose coarse grid points that will coarsen across strong connections, avoiding coarsening across interfaces [11, 68].

AMG also uses the matrix to produce the transfer operators. The transfer operators are the restriction and interpolation (also referred to as the prolongation) operators that are used to restrict and interpolate the matrix as well as the vectors. Using the matrix to create these operators can be simpler and more accurate than attempting to create them from geometries with complex unstructured problems.

2.3.2 Other Inverse Multigrid Work

There has been work on applying multigrid to other difficult and ill-posed problems. One example is the application of multigrid to highly anisotropic problems [56, 69, 101]. This is not an ill-posed problem, but anisotropies create problems for multigrid, a geometrically based algorithm, and anisotropies distort the effective geometry. The anisotropies tend to create ill-conditioned systems. These problems are tackled using line relaxation (using differing degrees of smoothing in each direction) and an approach that involves coarsening in the same direction as anisotropies. This coarsening strategy referred to as *semicoarsening* is similar to the approach that I will describe later, though it does segment the steps up in the same manner that I do. A paper by Akcelik and Ghattas discusses applying multigrid to an inverse problem [3]. However, their work is with inverse wave propagation, and this is a much different problem than the problem we are solving. There is also research at Weizmann Institute of Science in Israel by Achi Brandt on applying multigrid to solving the inverse problem of electrical impedance tomography (EIT), although, once again, the EIT problem is quite different from the inverse bioelectric field problem because the EIT problem solves for the conductivities instead of solving for the potentials.

Ongoing research projects that apply multigrid to the inverse bioelectric field

problem use dipole localization. One approach uses lower resolution grids to estimate source locality limits [33]. These limits are then used as constraints on source localization at a higher resolution to locate the dipoles quicker and more accurately. Also, at the University of Erlangen, Mohr and Rüde are implementing multigrid to perform the forward solves for lead field construction [65, 80]. An advantage of using multigrid for this application is that while the forward problem must be solved many times, the multigrid setup needs only be performed once to handle all the solves.

CHAPTER 3

METHODS

3.1 Applying Algebraic Multigrid to the Inverse ECG Problem

To formulate the inverse ECG surface-to-surface problem for algebraic multigrid (AMG) we need the problem in the form of $\overline{A}x = b$ ($\overline{A}$ is defined in Equation 3.6 and is distinguished from the stiffness matrix A) in order to carry out the multigrid procedures [12]. There are different ways to arrange the problem into a linear matrix solve. With AMG, it is important that this equation meet certain criteria. First, AMG is reliant upon iterative smoothers in order to function properly. This smoother can be an iterative method like Jacobi, Gauss-Seidel (in our case), or one of the many other methods, but the smoother must exhibit some convergence on $\overline{A}x = b$. Specifically, the iterative method must be able to diminish the high-frequency error in a given guess. It is not always necessary for the iterative method to reduce low-frequency error since the coarse level part of AMG excels at this. Another important requirement of the $\overline{A}x = b$ setup, is that $\overline{A}$ must have a geometric interpretation so that the algorithm can accurately perform coarsening. This issue requires an alteration in the implementation that will be discussed later.

As mentioned earlier, the physics of the bioelectric field problem are described with a discretization that relates the current sources to the potentials throughout the body. The equation that describes this relation can be written in the form:

$$A\phi = b, \tag{3.1}$$

where ϕ is the potentials and b is a vector of the current sources. We can then break the matrix and vectors up into the following regions: torso (t) nodes, volume (v) nodes, and epicardial (e) nodes. This leads to:

$$\phi = \begin{bmatrix} \phi_t \\ \phi_v \\ \phi_e \end{bmatrix} \tag{3.2}$$

$$b = \begin{bmatrix} b_t \\ b_v \\ b_e \end{bmatrix} \tag{3.3}$$

$$A = \begin{bmatrix} A_{tt} & A_{tv} & A_{te} \\ A_{vt} & A_{vv} & A_{ve} \\ A_{et} & A_{ev} & A_{ee} \end{bmatrix}. \tag{3.4}$$

Traditionally, we compute the lead field matrix that directly relates the epicardial potentials to the surface potentials while ignoring the rest of the potentials (nodes between the heart and surface) [17–19]. However, by discarding the values we lose the geometric descriptiveness of the full solution. Retaining the estimates of these potentials from coarse grid corrections is critical to performing the geometrically complete corrections. Multigrid's spectral mechanism relies on the local relations of a geometrically based system. Therefore, my approach leads to the formulation below. In addition, this method inherently solves for all the potentials between the torso and epicardium. These additional potentials give us insight into the bigger picture of the interior electric activity.

From this logic, we create a formulation. Electrocardiograms provide measured torso surface potentials ϕ_m (the measured nodes can be a subset of the ϕ_t nodes, but the entire surface can be estimated through interpolation, or the non-measured torso nodes can be treated as volume nodes) and we seek to determine epicardial potentials ϕ_e. In this research, the entire surface is treated as known torso nodes. We assume no current sources outside of the heart, so the boundary conditions tell us that b_t and b_v, which represent torso and volume current sources, should be equal to zero. Also, ϕ_t should equal the measured surface potentials ϕ_m. b_e represents the current influx across the epicardium, and since this is unknown and unnecessary we can discard this part of the problem, leaving zero for the right hand side. Since we are solving for ϕ, but ϕ_t is already known, we can subtract ϕ_t from the problem and eliminate these nodes from the problem as well. We now have:

$$\overline{\phi} = \begin{bmatrix} \phi_v \\ \phi_e \end{bmatrix} \tag{3.5}$$

$$\overline{A} = \begin{bmatrix} A_{tv} & A_{te} \\ A_{vv} & A_{ve} \end{bmatrix} \tag{3.6}$$

$$\overline{A}\phi = 0. \tag{3.7}$$

And we subtract $\overline{A} \cdot \phi_t$:

$$\overline{A} \cdot \phi - \overline{A} \cdot \phi_t = \overline{A} \cdot (\phi - \phi_t), \tag{3.8}$$

which we write as:

$$\overline{A}\overline{\phi} = \overline{b}, \tag{3.9}$$

where

$$\overline{b} = \begin{bmatrix} -A_{tt} \cdot \phi_t \\ -A_{vt} \cdot \phi_t \end{bmatrix}. \tag{3.10}$$

Regularization will also be necessary. We use the Tikhonov equation:

$$(K^T \cdot K + \alpha \cdot I) \cdot x = K^T \cdot y, \tag{3.11}$$

and we substitute the matrix and vector above:

$$(\overline{A}^T \cdot \overline{A} + \alpha \cdot I) \cdot \overline{\phi} = \overline{A}^T \cdot \overline{b}, \tag{3.12}$$

and solve:

$$\overline{\phi} = (\overline{A^T} \cdot \overline{A} + \alpha \cdot I)^{-1} \cdot \overline{A}^T \cdot \overline{b}. \tag{3.13}$$

Here shown with expanded matrix and vectors:

$$\begin{bmatrix} \phi_v \\ \phi_e \end{bmatrix} = (\begin{bmatrix} A_{tv} & A_{te} \\ A_{vv} & A_{ve} \end{bmatrix}^T \cdot \begin{bmatrix} A_{tv} & A_{te} \\ A_{vv} & A_{ve} \end{bmatrix} + \alpha \cdot \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix})^{-1} \cdot \begin{bmatrix} A_{tv} & A_{te} \\ A_{vv} & A_{ve} \end{bmatrix}^T \cdot \begin{bmatrix} -A_{tt} \cdot \phi_t \\ -A_{vt} \cdot \phi_t \end{bmatrix}. \tag{3.14}$$

This equation is now in the preferred $\overline{A}x = b$ form, and we can apply an iterative solver. Also, as mentioned before, $\overline{A}$ must exhibit a geometric interpretation. This is necessary for the coarsening algorithm of AMG. $\overline{A}$ has an increased number of entries due to the above equations and no longer has a direct geometric interpretation to the nodes and their connections. Therefore, with this approach, the original A is used to create the coarsening graph. This is a deviation from normal AMG since we are solving with a different matrix ($\overline{A}$) than the matrix (A) used for coarsening construction.

3.1.1 The Inverse Effect

To illustrate the challenge of applying multigrid to the inverse problem, we will examine a 2D problem with 541 nodes (shown in Figure 3.1). Of these, there are 60 epicardial nodes, 376 volume nodes, and 105 torso nodes. The potentials on the heart are a subset of real measured values taken from a three-dimensional model [74]. The torso values are calculated from the forward problem and have noise added to them in order to simulate torso measured values. These simulated torso measured values ϕ_m are the input and the algorithm attempts to determine heart potentials ϕ_e. We can compare the calculated heart potentials with the real measured values. During this test, the coarsening is one level deep to simply illustrate the effect of an ill-conditioned matrix. The interpolation and restriction steps use simple bilinear approximation. Both the forward problem and the inverse problem are solved on this test.

From this test, we can see that accuracy is difficult to maintain with the inverse problem. The main reason for this inaccuracy is because the inverse problem is quite ill-conditioned and does not continuously depend on the data. Small changes in the torso surface potential measurements and also small changes in the conductivity model cause large errors in the calculated epicardial potentials. Unfortunately, the coarsening of matrix A does indeed cause small distortions in the conductivity model. This means that the solution to the coarsened level equations is not a precise match to the solution on the fine level. An accurate coarse matrix should exactly compute a restricted solution vector from the restricted initial conditions vector. The problem of coarse matrix inaccuracy does not manifest itself in more well-conditioned problems [23, 97].

Consider these graphs with the true solution and the coarse grid computed solution. Figure 3.2 shows the well-posed forward ECG test problem, whereas Figure 3.3 shows the ill-posed inverse ECG test problem.

Quantitatively speaking, the inverse problem can be expressed as more difficult because the condition number for the matrix to solve is much higher than for the forward problem. For instance, with Test Set 2 and 7 (Section 5.1) the forward

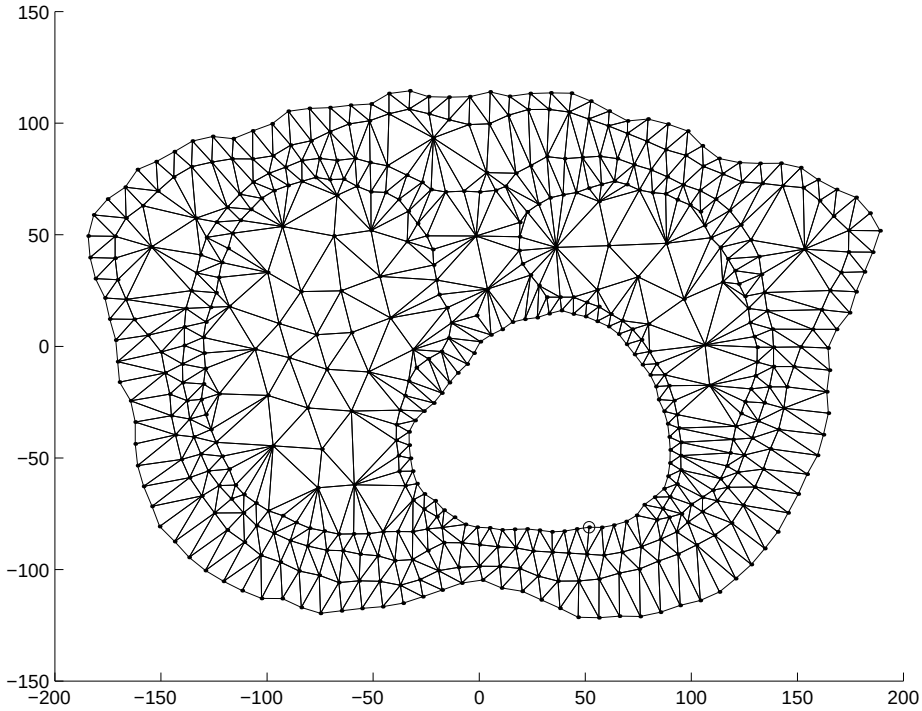

Figure 3.1. Mesh of the 541 node sample problem.

problem has a condition number of approximately $1x10^3$ and the inverse problem has a condition number of approximately $1x10^{15}$. The graphs above were taken from Test Set 1. For this dataset, the forward problem has a condition number of approximately $1x10^3$ and the inverse problem has a condition number of approximately $1x10^{17}$.

The ill-posedness of the inverse problem also makes the smoothing process of multigrid much less effective. Iterative solvers generally converge much slower with ill-posed problems, and the desired convergence in the high frequency range suffers as well. There are a number of smoothers that improve upon the basic Jacobi or Gauss-Seidel algorithm [45, 46, 67]. A more recent approach, which uses an approximate inverse smoother, has performed very well on ill-conditioned problems [120]. However, I will not be doing a comparison of the iterative solvers effectiveness with different smoothing processes.

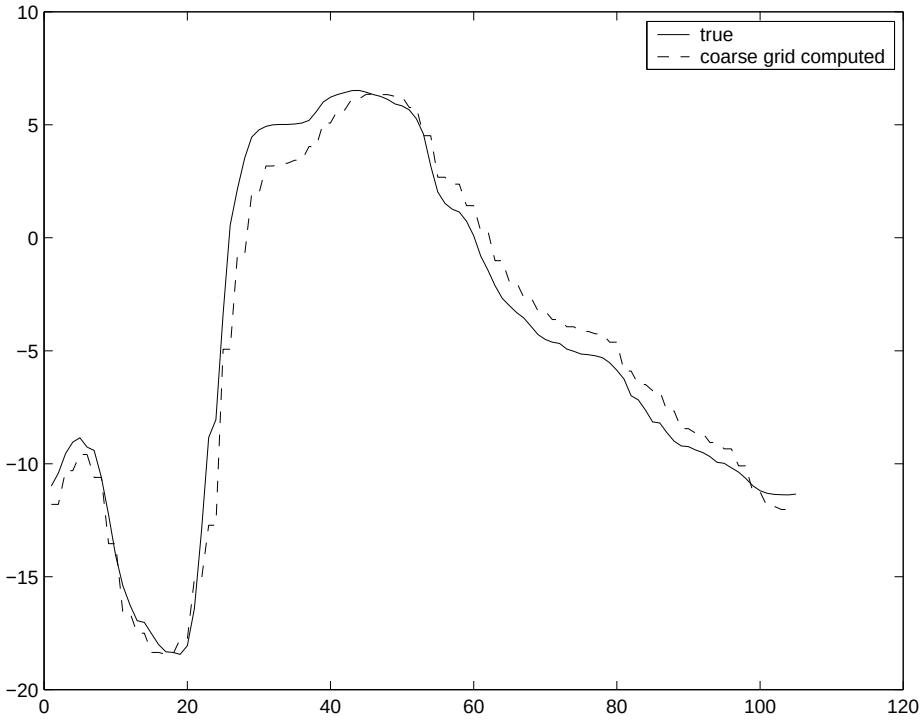

Figure 3.2. The result of a coarse grid solve on the forward test problem. The x-axis is position on the heart, y-axis is potential.

3.2 Improved Coarsening

The problem of inaccurate coarse grid correction introduced by ill-conditioned problems must be addressed by use of more accurate coarsening schemes than those traditionally used for well-conditioned multigrid problems. Another problem with deeper nesting of coarse grids is that, with a small dataset, the torso nodes are only three to six layers of nodes away from the epicardial nodes, in places. Attempting a second level of coarsening obviously creates some abnormal geometries with the torso nodes either touching, or only having only one layer of nodes, between epicardial and torso nodes. This, coupled with the coarsening effect of an ill-conditioned matrix, creates problems with maintaining an accurate geometry. One approach to alleviate this problem is to coarsen only along layers (reduce the nodes in each layer, but not to reduce the number of layers) in the certain levels

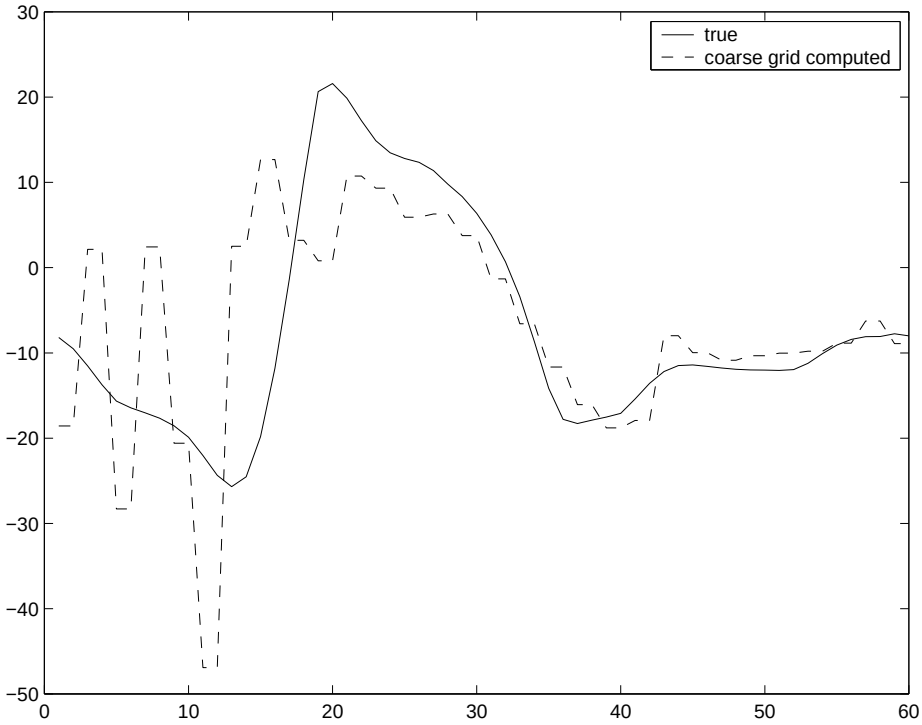

Figure 3.3. The result of a coarse grid solve on the inverse test problem. The x-axis is position on the heart, y-axis is potential.

of coarsening. This problem should not be manifested in early stages of nesting in larger datasets. In datasets where there are 20 or more layers between torso and epicardial nodes, the nested coarsening should progress much longer without problems.

This research has mostly focused on creating more accurate schemes for the interpolation and restriction of coarsened grids for use in solving inverse bioelectric field problems. There has been a large focus on the choice of the coarse grid points in multigrid applications, but even after the choice has been made there is still the question of exactly how to incorporate the information about the fine grid points. We can discard the values for the fine grid points and keep only the coarse grid information. Alternatively, an arithmetic approach can be used to combine the coarse and fine grid point values. We must then decide how to weight the coarse

and fine points. The first simple weighting approach to calculating a new coarse grid point is bilinear approximation. With this method, we give the old coarse grid point a unit weight and each of the connected fine grid points an equal weight that sums to the unit weight.

The second weighting approach, connection based weighting (CBW), makes each of the weights proportional to the value of the connection [14]. This should improve the ability to maintain the operation of the A matrix. However, the A matrix is not the only set of information that must be accurately estimated in the coarse grid. The b vector (the residual in coarser grids) must also be accurately coarsened. In addition, it is useful to have some type of mechanism that will allow weak connections to be dropped, otherwise the coarsening produces overly dense matrices.

The third approach, which is the standard method used with AMG, uses a tolerance level for the connections. If the strength of the connection is below a tolerance level, the connection is not fully computed in the weighting calculations. A lower tolerance level means greater accuracy, but denser matrices. The equation for determining weights begins with the smooth-error approximate relation:

$$a_{ii} \cdot e_i \approx -\Sigma_{j \in C_i} a_{ij} \cdot e_j - \Sigma_{j \in D_i^s} a_{ij} \cdot e_j - \Sigma_{j \in D_i^w} a_{ij} \cdot e_j, \tag{3.15}$$

where C are the coarse grid points, D^s are the strongly connected fine grid points, and D^w are the weakly connected find grid points. For D^w we approximate $e_j \approx e_i$. For D^s we e_j by a weighted average of e_k in the coarse interpolatory set $C_j \bigcap C_i$:

$$e_j \approx \frac{\Sigma_{k \in C_i} a_{jk} \cdot e_k}{\Sigma_{k \in C_i} a_{jk}}. \tag{3.16}$$

Finally, we approximate and substitute to get the standard equation used in AMG for determining weighting:

$$w_{ij} = -\frac{a_{ij} + \sum_{k \in D_i^s} \frac{a_{ik} a_{kj}}{\sum_{m \in C_i} a_{km}}}{a_{ii} + \sum_{n \in D_i^w} a_{in}}, \tag{3.17}$$

where w are the weights. This will be referred to as the standard method. This equation will breakdown to the simple weighted connection method when the

tolerance level is set at zero and every connection is considered a strong connection [14].

The fourth approach is multistep, which I have developed to specifically deal with the inaccuracy of coarse grid correction with the inverse bioelectric field problem. This method establishes several intermediate semicoarsened grids between the fine grid and coarse grid. The interpolation operator is then built by successively moving between these intermediate grids. The next section (Multistep Approach, 3.3) will discuss this in detail.

The fifth approach is to use an approximate inverse. This can refer to any method that uses any approximation of the A matrix inverse to interpolate the vectors. The particular approach I have implemented, solves for the block of nodes surrounding each coarse grid node. The values from this block inversion yield values or potentials to be expected with a unit point potential at the coarse grid. These simulated fine grid potentials make a logical weight for the interpolation of the fine grids. The cumulative block inversions represent the approximation of the inverted A matrix that is used as the transfer operator [83, 96].

In addition, I tested two hybrid approaches. These approaches included using the CBW and bilinear approach for each of the levels of the multistep approach (the standard approach is the default).

All of the operators are normalized in the sense that each of the weightings that contribute to the interpolation of a fine grid point must sum to one. If this rule does not hold then a grid of uniform nonzero values that could be interpolated to a fine grid of nonuniform values.

3.3 Multistep Approach

The multistep approach is based on the idea that by removing fewer nodes at once we can generate a more accurate transfer operator. With the removal of fewer nodes also comes the need to combine several coarsening steps into one, in order to maintain speed while moving through the coarse grids of multigrid cycle.

We first create a goal coarse grid by using a standard coarse grid selection. We then create intermediate coarse grids which are supersets of the goal coarse grid.

That is, the fine grid points from the goal coarse grid selection will always be fine grid points with all of the intermediate coarse grids. In order to limit node removal at any intermediate step, we create a rule that for the creation of the intermediate grids, each time a coarser grid is generated, adjacent (connected) nodes must not be eliminated. In other words, the grids must not have any noncoarse grid points that are connected (points that are eliminated in the next coarse grid selection). Figures 3.4 - 3.7 demonstrate the process on the model in Figure 3.1.

Next, we create a interpolation operator for each of these progressively coarser intermediate grids using ones of the methods mentioned above (CBW was used in the demonstration, but the tests use the standard AMG weighting scheme to prevent excessively dense matrices from forming). The interpolation operator for the actual multigrid step, I, is the product of the interpolation operator for each of the intermediate semicoarsening steps. The restriction operator is still the transpose of the interpolation operator.

$$I = I_1 \cdot I_2 \cdot I_3 \cdot ... \tag{3.18}$$

When we remove nodes with the multistep approach we are performing a transform basically analogous to solving for equivalent resistance in circuit analysis. When we look at the bioelectric field problem as solving for a large circuit, each node is connected to the surrounded nodes with resistors. Therefore, we can imagine removing nodes from the "circuit" and then altering the resistors in the surrounding nodes to create an "equivalent circuit." Consider the four nodes in Figure 3.8 as a subcircuit within a grid of electrically connected nodes that represents the mesh of the bioelectric field problem from the finite element discretization. With the multistep approach, the inner node can be removed with assurance that none of the surrounding nodes will be removed to permit a transform similar to equivalent circuit analysis this yields the circuit shown in Figure 3.9

If we remove the middle node, we will have the subcircuit in Figure 3.9 where R_{12}, R_{23}, and R_{31} must be computed to form a circuit equivalent to the four node circuit.

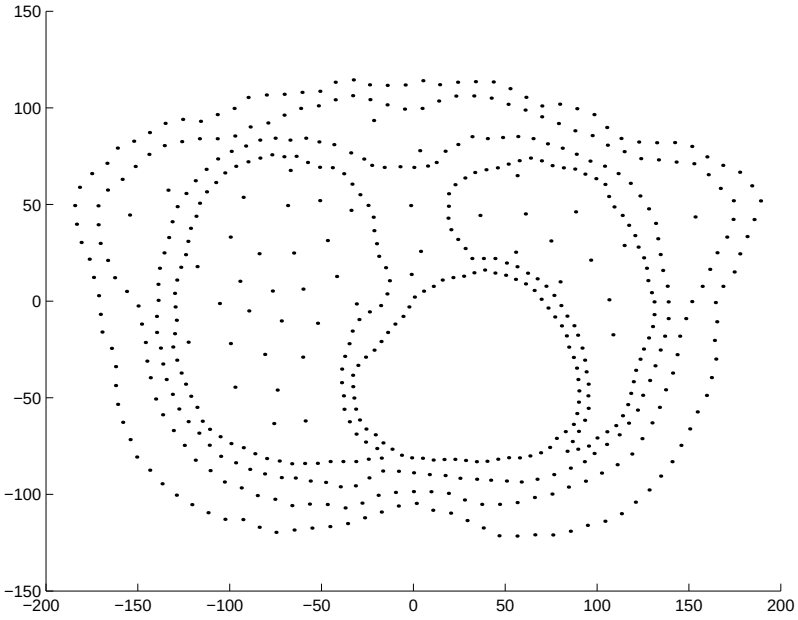

Figure 3.4. Nodes are reduced over several steps with semicoarsening from 541 nodes to 399 nodes to 328 nodes to 283 nodes. This is the initial grid.

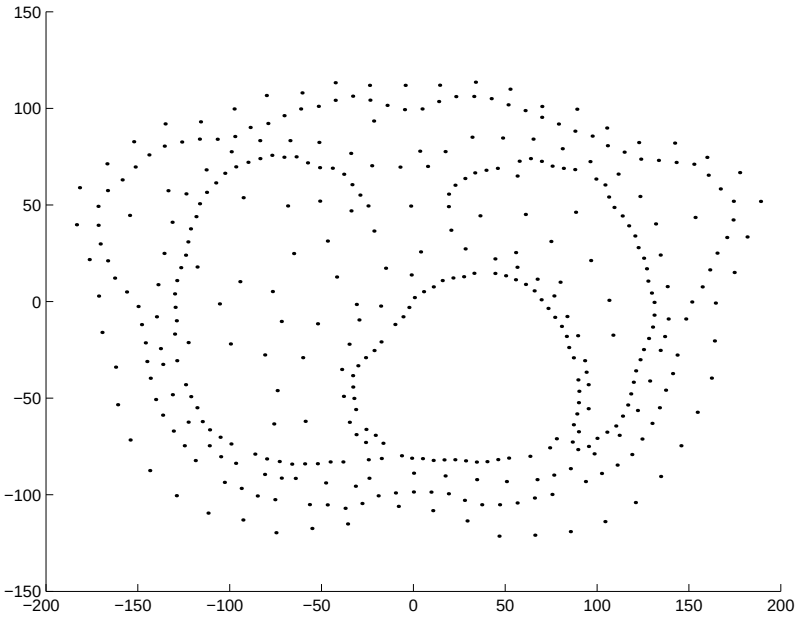

Figure 3.5. Semicoarsening reduction after one step.

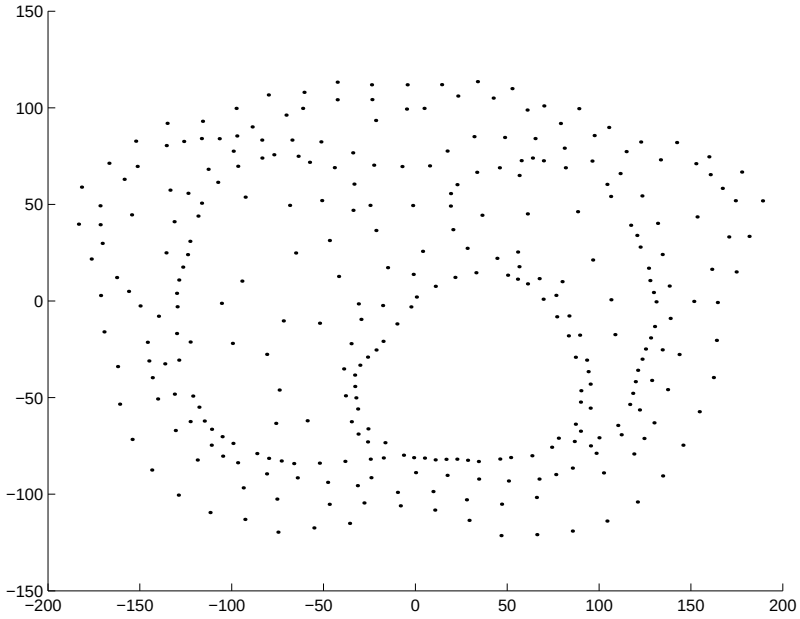

Figure 3.6. Semicoarsening reduction after two steps.

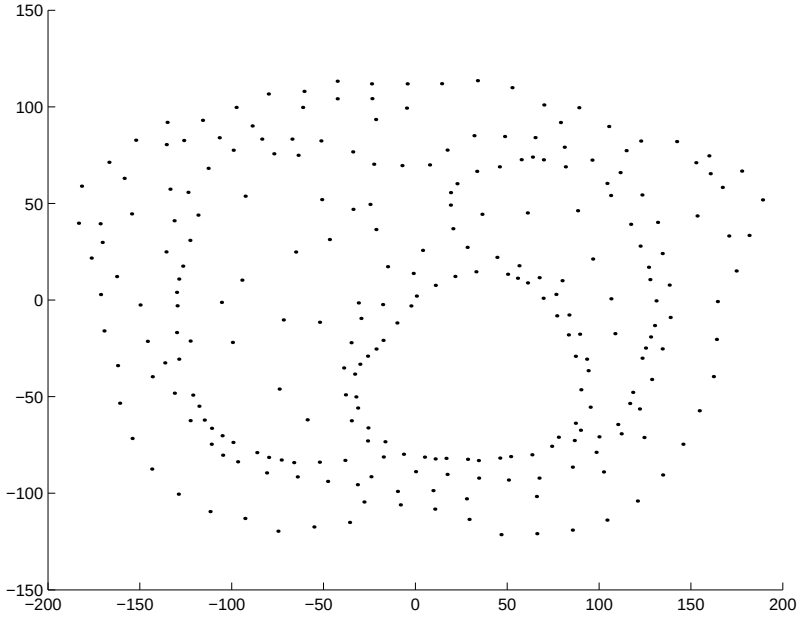

Figure 3.7. Semicoarsening reduction after three steps.

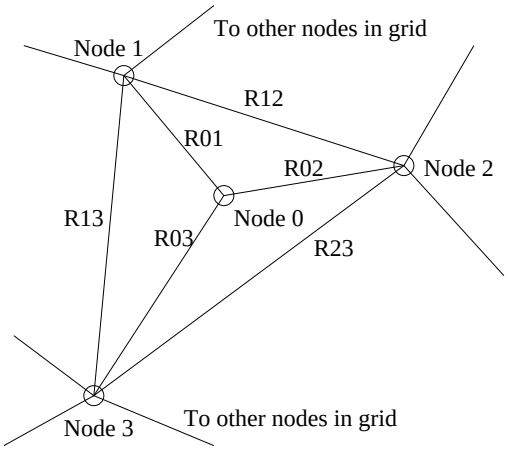

Figure 3.8. Example 4 node "subcircuit" within a mesh.

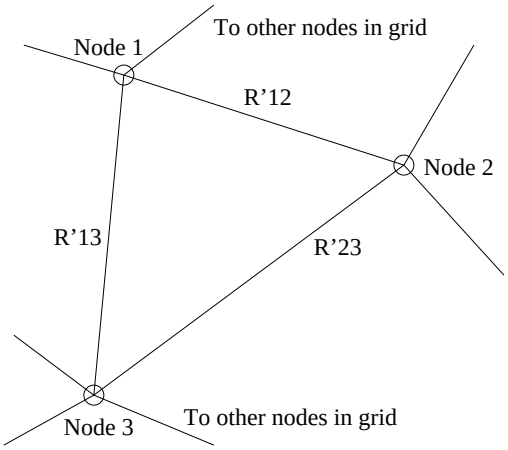

Figure 3.9. Example 3 node "subcircuit" within a mesh. The inner node (node 0) has been removed, and equivalent resistance for the remaining resistors can be computed.

The equations for the currents in the original circuit:

$$0 = \frac{V_0 - V_3}{R_{03}} + \frac{V_0 - V_2}{R_{02}} + \frac{V_0 - V_1}{R_{01}} \tag{3.19}$$

$$I_1 = \frac{V_1 - V_3}{R_{13}} + \frac{V_1 - V_2}{R_{12}} + \frac{V_1 - V_0}{R_{01}} \tag{3.20}$$

$$I_2 = \frac{V_2 - V_1}{R_{12}} + \frac{V_2 - V_3}{R_{23}} + \frac{V_2 - V_0}{R_{02}} \tag{3.21}$$

$$I_3 = \frac{V_3 - V_2}{R_{23}} + \frac{V_3 - V_1}{R_{13}} + \frac{V_3 - V_0}{R_{03}}. \tag{3.22}$$

And for the currents in the equivalent circuit:

$$I_1 = \frac{V_1 - V_3}{R'_{13}} + \frac{V_1 - V_2}{R'_{12}} \tag{3.23}$$

$$I_2 = \frac{V_2 - V_1}{R'_{12}} + \frac{V_2 - V_3}{R'_{23}} \tag{3.24}$$

$$I_3 = \frac{V_3 - V_2}{R'_{23}} + \frac{V_3 - V_1}{R'_{13}}. \tag{3.25}$$

Solving for the resistance in the equivalent circuit would be:

$$\frac{1}{R'_{12}} = \frac{1}{R_{12}} + \frac{1}{R_{01} + R_{02} + \frac{R_{01} \cdot R_{02}}{R_{03}}} \tag{3.26}$$

$$\frac{1}{R'_{23}} = \frac{1}{R_{23}} + \frac{1}{R_{02} + R_{03} + \frac{R_{02} \cdot R_{03}}{R_{01}}} \tag{3.27}$$

$$\frac{1}{R'_{31}} = \frac{1}{R_{31}} + \frac{1}{R_{03} + R_{01} + \frac{R_{03} \cdot R_{01}}{R_{02}}}. \tag{3.28}$$

If we apply the restriction operator constructed with the multistep algorithm to the same problem, we would construct an A matrix using a system of equations (note we use conductivity values instead of resistances to create the stiffness matrix):

$$0 = V_0 \cdot (C_{01} + C_{02} + C_{03}) - V_1 \cdot C_{01} - V_2 \cdot C_{02} - V_3 \cdot C_{03} \tag{3.29}$$

$$I_1 = -V_0 \cdot C_{01} + V_1 \cdot (C_{01} + C_{12} + C_{13}) - V_2 \cdot C_{12} - V_3 \cdot C_{13} \tag{3.30}$$

$$I_2 = -V_0 \cdot C_{02} - V_1 \cdot C_{12} + V_2 \cdot (C_{02} + C_{12} + C_{23}) - V_3 \cdot C_{23} \tag{3.31}$$

$$I_3 = -V_0 \cdot C_{03} - V_1 \cdot C_{13} - V_2 \cdot C_{23} + V_3 \cdot (C_{03} + C_{13} + C_{23}), \tag{3.32}$$

and convert into a matrix:

$$A = \begin{bmatrix} S_0 & -C_{01} & -C_{02} & -C_{03} \\ -C_{01} & S_1 & -C_{12} & -C_{13} \\ -C_{02} & -C_{12} & S_2 & -C_{23} \\ -C_{03} & -C_{13} & -C_{23} & S_3 \end{bmatrix} \tag{3.33}$$

$$b = \begin{bmatrix} 0 \\ i_1 \\ i_2 \\ i_3 \end{bmatrix}, \tag{3.34}$$

where C represents the conductivity, and S is the sum of conductivities for each node. The multistep algorithm stipulates that we can remove nodes so long as the removed node is not connected to another node to be concurrently removed. For this demonstration, we will just remove the middle node (node 0). If we use the CBW method to produce the restriction operator, we get the following operator:

$$I = \begin{bmatrix} \frac{C_{01}}{S_0} & \frac{C_{02}}{S_0} & \frac{C_{03}}{S_0} \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3.35}$$

To restrict A we apply the operator:

$$A' = I^T \cdot A \cdot I \tag{3.36}$$

$$b' = I^T \cdot b \tag{3.37}$$

$$A' = \begin{bmatrix} S_1' & -C_{12} - \frac{C_{01} \cdot C_{02}}{C_{01}+C_{02}+C_{03}} & -C_{13} - \frac{C_{01} \cdot C_{03}}{C_{01}+C_{02}+C_{03}} \\ -C_{12} - \frac{C_{01} \cdot C_{02}}{C_{01}+C_{02}+C_{03}} & S_2' & -C_{23} - \frac{C_{02} \cdot C_{03}}{C_{01}+C_{02}+C_{03}} \\ -C_{13} - \frac{C_{01} \cdot C_{03}}{C_{01}+C_{02}+C_{03}} & -C_{23} - \frac{C_{02} \cdot C_{03}}{C_{01}+C_{02}+C_{03}} & -S_3' \end{bmatrix} \tag{3.38}$$

$$b' = b = \begin{bmatrix} i_1 \\ i_2 \\ i_3 \end{bmatrix}. \tag{3.39}$$

So, our conductivity values for our new matrix are:

$$C_{12}' = C_{12} + \frac{C_{01} \cdot C_{02}}{C_{01} + C_{02} + C_{03}} \tag{3.40}$$

$$C_{13}' = C_{13} + \frac{C_{01} \cdot C_{03}}{C_{01} + C_{02} + C_{03}} \tag{3.41}$$

$$C_{23}' = C_{23} + \frac{C_{02} \cdot C_{03}}{C_{01} + C_{02} + C_{03}}, \tag{3.42}$$

and this turns out to be mathematically equivalent to the equivalent circuit resistances we calculated above.

$$C_{12} + \frac{C_{01} \cdot C_{02}}{C_{01} + C_{02} + C_{03}} = \frac{1}{R_{12}} + \frac{1}{R_{01} + R_{02} + \frac{R_{01} \cdot R_{02}}{R_{03}}} \tag{3.43}$$

$$C'_{12} = \frac{1}{R'_{12}} \tag{3.44}$$

$$C_{23} + \frac{C_{02} \cdot C_{03}}{C_{02} + C_{03} + C_{01}} = \frac{1}{R_{23}} + \frac{1}{R_{02} + R_{03} + \frac{R_{02} \cdot R_{03}}{R_{01}}} \tag{3.45}$$

$$C'_{23} = \frac{1}{R'_{23}} \tag{3.46}$$

$$C_{13} + \frac{C_{03} \cdot C_{01}}{C_{03} + C_{01} + C_{02}} = \frac{1}{R_{13}} + \frac{1}{R_{03} + R_{01} + \frac{R_{03} \cdot R_{01}}{R_{02}}} \tag{3.47}$$

$$C'_{13} = \frac{1}{R'_{13}}. \tag{3.48}$$

This equivalency works by mathematically grafting two connections of a removed node into one connection. We eliminate the connections from the removed node and reconnect the surrounding nodes to each other. We create new connections that are equivalent to the connections that were eliminated. However, if we were to remove two nodes that were connected to each other at the same time, this equivalency will no longer hold. Indeed, the standard coarsening procedure removes multiple nodes that are connected to each other and therefore the CBW algorithm does not usually produce coarsened matrices that accurately reflect the original matrix.

When we do enforce the rule of not removing connected nodes in this semicoarsening approach prescribed by the multistep method, we see a great improvement in accuracy. Since this does not remove very many nodes at once, in order to create coarsened grids that are adequately spaced, we must combine several of these semicoarsening operators into one operator as described previously.

CHAPTER 4

SIMULATIONS

In the investigation, I evaluated several different weighting schemes for interpolation and restriction of coarsened grids. The transfer operator weighting schemes I tested included bilinear, connection based weighting (CBW), connection based weighting with tolerance level (standard), multistep, approximate inverse, and a hybrid.

This investigation will not evaluate other aspects of multigrid, such as coarse grid selection, relaxation methods, and stopping criteria. This research is focused on which transfer operators are most efficient in this particular application multigrid. This is not an attempt to fully implement the multigrid algorithm for real world large scale problem sets in a clinically usable application. Although several of the data sets are from real patients, the importance of this research is the evaluation of different algorithms for determining the transfer operator for AMG coarsening. This will potentially allow future researchers to implement the multigrid on the inverse bioelectric field (and other) problem and easily select a proper coarsening algorithm for their application.

The two main metrics that we consider are accuracy and speed. Accuracy is measured as the root mean square (RMS) error between the multigrid solution and true solution. I also measured the speed of the process, reported by the length of time each step takes.

There are also different aspects of the multigrid process that can be individually measured. To test the integrity of the discretization matrix at a coarsened level, we coarsen to fewer nodes (using one of the given methods weighting schemes), and then solve on the coarse grid. The solution matrix is also restricted using the same weighting scheme. Then, we compare the coarsened solution and the

solution from the coarse problem for accuracy. This test eliminates inaccuracies due to interpolation and the accuracy of the restriction operator is measured. This is a useful metric for comparing the accuracy of a given coarsening scheme on the matrix versus the vector. This test is referred to as the *matrix test.*

To test the accuracy of a coarsening scheme on the vector alone, we restrict the solution vector and then interpolate back to its original size. Then, the interpolated version is compared to the original for accuracy. This metric reveals restriction and interpolation accuracy on the vector. This test is referred to as the *vector test.*

However, correctly performing multigrid involves more than just solving a coarsened version of the original problem, but rather solving for the residual from the fine grid ($Ax - b$). When the residual is coarsened, the accuracy of coarsening both the vector and matrix contribute to the overall accuracy. I measured both the accuracy and the speed of these methods on one multigrid cycle. For true multigrid evaluation, this test was performed on several cycles, as well.

I also measured multistep multigrid against the solution from an exact inverse solve using conjugate gradient (with $3n$ iterations, where n is the number of nodes). This test demonstrates the rate at which the multigrid algorithm converges to the most precise solution available from inverse solving. This test was performed only on test set 1.

This investigation also compares the effect of varying levels of input noise added to the surface measurement input conditions. I measured the solution error for conjugate gradient and multistep multigrid algorithms with input noise ranging from 0% to 20% RMS of surface levels. Multigrid performed five cycles with one level of coarsening for this test. This was only performed on test set 2.

The choice of the weak connection tolerance (WCT) level, (the relative connection strength threshold for strong versus weak connections in the standard AMG algorithm) also affects the system performance. Selecting the tolerance level is a compromise between accuracy and the space/time constraints. A lower tolerance level produces a greater number of nonzero entries (see Figure 4.1). This, in turn, causes slower performance (see Figure 4.2). However, accuracy degrades with a

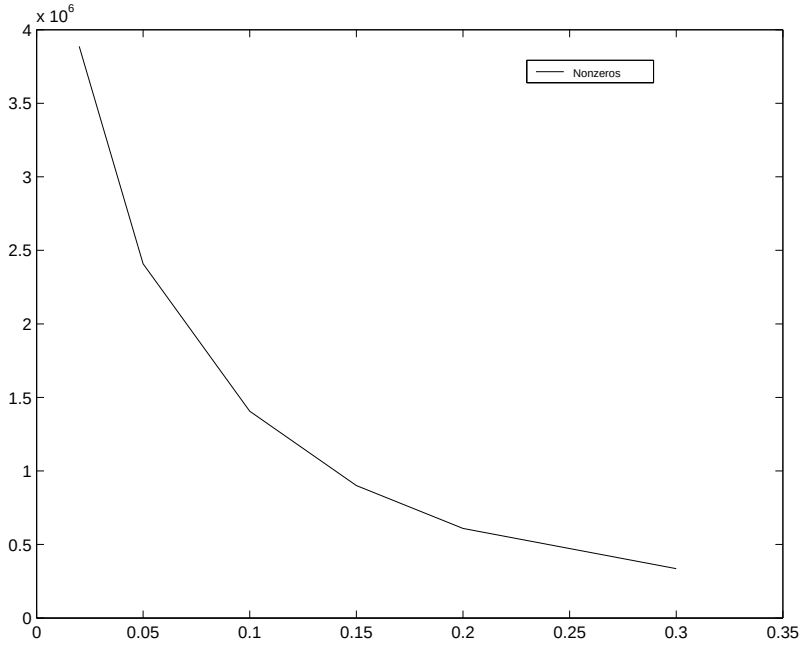

Figure 4.1. The number of nonzeros (y-axis) at the coarsest level versus weak connection tolerance level (x-axis) for test set 2.

higher tolerance level as seen in Figure 4.3. For the majority of the problems the tolerance level was set to 0.02, but as mentioned above, a greater tolerance level was used for the largest data set.

Another method supplements the standard method of limiting weak connections to reduce unnecessary matrix entries. After each restriction, we drop any connection in the matrix that does not meet a global tolerance level. This is different than the standard WCT tolerance, as the standard method only drops entries from the transfer operator matrix, whereas this operation is applied directly to the stiffness matrix and does not affect the vectors. This method was used for the tests on test set 8 and the test on test set 2 referred to as the stiffness matrix entry tolerance (SMET) test. For test set 8, SMET of $3x10^{-7}$ was used as the matrix entries are much smaller. Proportionally, this was approximately the same as the 0.01 SMET value used on test set 2 (about 1/5000 of the average matrix entry). These tests also

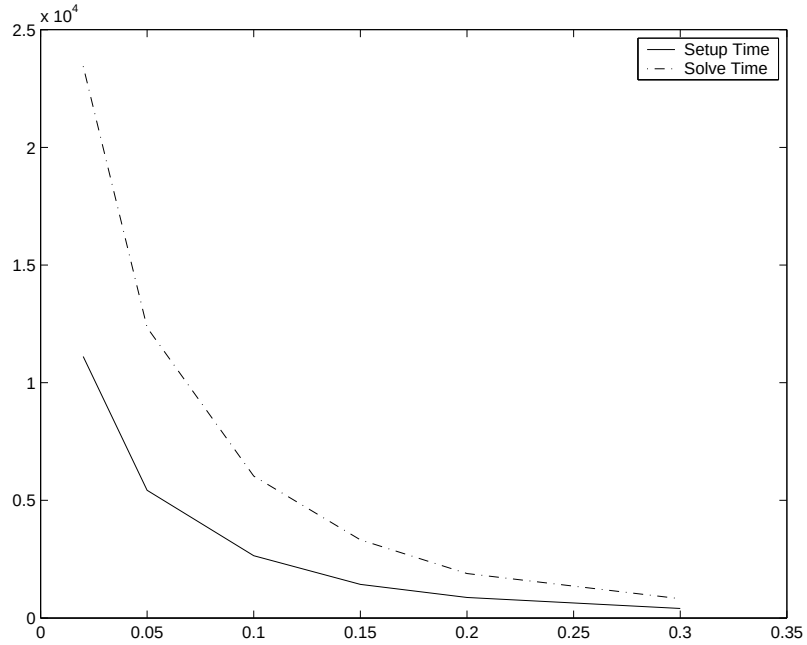

Figure 4.2. The weak connection tolerance level's effect on performance. The setup time and the solve time (y-axis) plotted versus the weak connection tolerance level on the x-axis for test set 2.

compared traditional solvers. I ran the tests solving with conjugate gradient using a Tikhonov regularization (with an alpha of 0.02). I also solved using conjugate gradient on the normal equations with limited iterations for regularization (two fifths of the eigenvectors were eliminated in lieu of using Tikhonov regularization). It should be noted that the methods were carried out on the system described in the methods section, allowing a consistent comparison with the multigrid algorithms. To view an analysis of how common solvers work on traditional formulations, please refer to papers, [6, 30, 60, 106].

I also measured the scalability of each algorithm. The accuracy and speed was measured on a range of problems from 300 nodes to 156,728 nodes. These results are crucial in determining which coarsening schemes are most appropriate, and whether a mix of schemes is superior for the inverse bioelectric field problem.

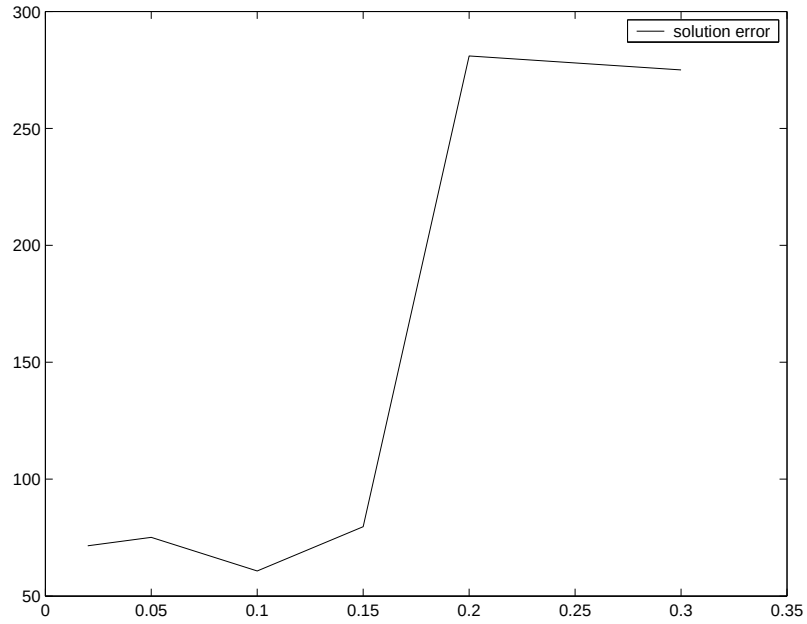

Figure 4.3. The weak connection tolerance level's effect on accuracy. The error in the solution (y-axis) plotted versus the weak connection tolerance level on the x-axis for test set 2.

4.1 Test Procedure

For the test procedure, the algorithms were written in C++ and ran on a dual processor (although, I did not use parallelization) 250Mhz UltraSparc with 768MB of memory. Some of the tests were performed on other machines, but the results were scaled to the performance of the UltraSparc. The matrices were stored in a sparse matrix format. An array corresponding to rows in the matrix held the matrix data. Each element contained a list of values and column position for each of the nonzeros within that row. Most of the algorithms were first written and tested in Matlab before being implemented in C++. Matlab was used extensively throughout the tests for vector manipulation and visualization.

I created three test vectors for each test. Test vector creation begins by generating pseudo heart current sources. For the two-dimensional problems, a vector of random values was generated (uniform density random variable). Next, the vector

was smoothed, by averaging each epicardial node with its neighbors, using this mean as the new value. This cycle was executed twice (an example vector is shown in Figure 4.4).

For the three-dimensional problems, the first test vector was a linear gradient in the x direction, the second vector was a linear gradient in the y direction, and the third vector was a parabolic gradient in x direction, combined with a linear gradient in the z direction. This is easier to solve than the randomized vectors described for the two-dimensional problems because of fewer peaks and areas of high gradient, although the three-dimensional problem is inherently much more difficult to solve. Next, each of the vectors are normalized to have a sum of zero (subtracting the mean from each of the values). From the current source values, the voltages are solved for the entire problem using the conjugate gradient method. This can be computed without too much difficulty since this is a well-posed forward problem. The resulting torso values are used as the measurement values for the multigrid problem and the epicardial voltages are used as the correct solution with which to measure the error values.

Each test program begins by either reading in the appropriate test matrix from a file or generating the matrix. Then it reads in the appropriate test vector. In the next step, the program generates the transfer operators using the selected coarsening algorithm (bilinear, weighted, etc.). Finally, the chosen multigrid (matrix, vector, one cycle, or multicycle) steps are applied to attempt to solve the inverse problem. At the coarsest level the linear system is always solved with the conjugate gradient method. However, with the largest three-dimensional problem the Tikhonov setup is not used and instead the conjugate gradient for normal equations with limited iterations for regularization [50, 89] is used instead, in order to run the tests within a reasonable amount of time. This method was also used on the smaller three-dimensional problem (test set 2) for the sake of comparison between the large and small three-dimensional problem to test scalability. This is not intended to be a comparison of regularization methods.

The final solution is then measured against the epicardial heart voltage values

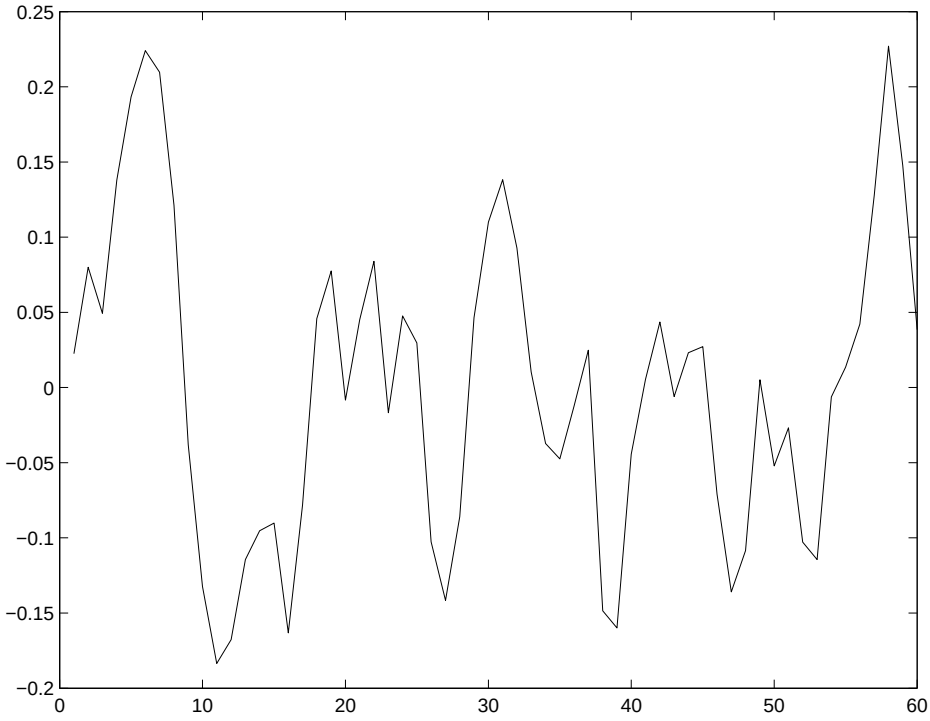

Figure 4.4. One of the random vectors used as the for epicardial potentials for test set 4. The remaining computed potentials are then based upon this test vector. The x-axis represents position on the heart, y-axis represents potentials.

from the initial solve to determine the accuracy (measured with RMS error). The last two steps, the multigrid setup and the multigrid solve, are also each timed. The restriction and interpolation time is also measured for some problems and is part of the solve time and is calculated by subtracting the coarse grid solve time.

I also recorded how many levels of coarsening were used in the cycle. The deeper the cycle went, the more difficult to maintain accuracy, as the coarser grids will introduce more inaccuracies than the fine grids.

I carried out all the tests three times and all the results (errors and times) are averaged. Not every possible combination of tests, test sets, and measurements are performed for the sake of brevity. The larger problem test sets have a more abbreviated set of experiments, due to their much greater run time. Many of

the tests do not have the multigrid time measurement because it was insignificant for a problem of that size. Also, the vector and matrix tests do not have time measurements because these are just subsets of the cycle which is already broken down into different time measurements. Likewise, tests that include smoothing operation do not differ in their setup process, so measurements are not retimed.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 Test Sets and Data

This section includes the data results from all the simulations. There were eight test sets. Each of these test sets represents a different model of the body, varying in geometry and size. Each test set result includes a description of the test set model and a table of the results from the simulations. Speed is measured in seconds and error is percent RMS from the correct value.

The following tables show the test results. The first column in each of the tables tells whether a vector test, matrix test, or cycle test was executed in the procedure. If the procedure was a multicycle test, this column tells how many cycles were performed to achieve the given results. This column also identifies the tests performed on Krylov subspace iterative solvers. The second column describes which transfer operator the test utilized for coarsening. Further right, the error column lists error values from the solution estimate of each test. The fourth column gives the elapsed time for the setup of the transfer operators and coarsened matrices and vectors. The fifth column shows the solve time for conjugate gradient on the coarsest level, and the sixth column gives the processing time for applying the transfer operator to the vectors. Each section includes a header briefly describing the preceding tests, and tells how many levels of coarsening the procedure applied to the problem set.

Test sets for these simulations include:

`Test set 1` A two-dimensional torso model with 541 nodes. Of these, there were 60 epicardial nodes, 376 volume nodes, and 105 torso nodes as shown in Figure 5.1. The potentials on the heart were real measured values. The torso

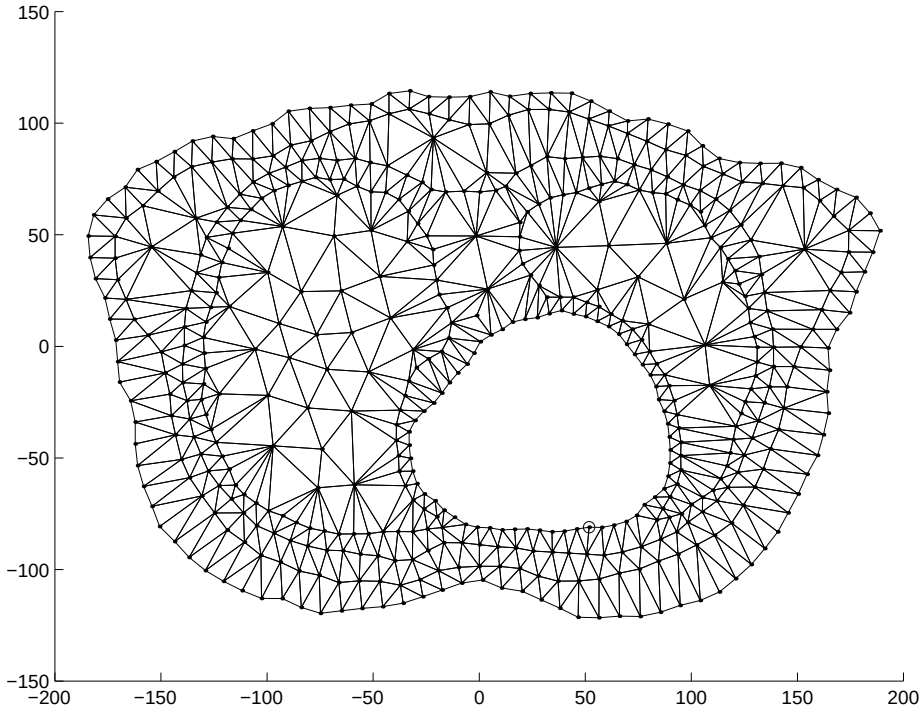

Figure 5.1. Mesh used for test set 1.

values were calculated from the forward problem. The inverse problem had an approximate condition number of $9\text{x}10^{16}$. Tables 5.1 - 5.2 show the simulation results.

`Test set 2` A three-dimensional torso model that was discretized from magnetic resonance data [74]. It had 7186 nodes and of these, there were 296 epicardial nodes, 6188 volume nodes, and 702 torso nodes. The stiffness matrix had 95,189 nonzeros. The condition number for the inverse problem was incalculable for Matlab. Tables 5.3 - 5.4 show the simulation results. Figures 2.2 and 5.2 show the mesh.

Table 5.1. Test set 1 - 541 Nodes

Test	Coarsening method	Error	Setup time	CG time	MG time
1 Level deep tests					
1 Cycle	Bilinear	65%	0.27s	0.55s	Insig-
	Standard AMG	74%	0.34s	0.62s	nificant
	Weighted	63%	0.26s	0.56s	
	S. AMG w/3 smooths	52%	0.26s	0.56s	
	Multistep	16%	0.79s	2.18s	
Vector	Bilinear	18%			
	CBW	37%			
	Standard AMG	33%			
	Multistep	24%			
Matrix	Bilinear	67%			
	CBW	95%			
	Standard AMG	98%			
	Multistep	25%			
2 Level deep tests					
1 Cycle	Standard AMG	81%	0.49s	0.62s	Insig-
1 Cycle	Multistep	19%	1.73s	1.59s	nificant
2 Cycles	Multistep	11%		3.23s	
3 Cycles	Multistep	15%		4.96s	
4 Cycles	Multistep	17%		6.60s	
5 Cycles	Multistep	17%		8.22	
Conjugate gradient with Tikhonov regularization		5.9%		7.87s	
Conjugate gradient with limited iterations		39%		2.19s	
Test of correct convergence: 1 level deep multigrid measured against correct inverse solve ($3n$ iterations of conjugate gradient)					
1 Cycle	Multistep	10.9%			
2 Cycles	Multistep	2.37%			
3 Cycles	Multistep	1.97%			
4 Cycles	Multistep	1.92%			
5 Cycles	Multistep	1.91%			

Table 5.2. Test Set 1 - 541 Nodes - Test of error robustness with varying input noise addition.

Input noise addition	Coarsening method	Error
1 Level Deep - 5 Cycle Tests		
0%	Multistep	4.8%
5%	Multistep	7.9%
10%	Multistep	14%
15%	Multistep	20%
20%	Multistep	26%
Conjugate gradient with Tikhonov regularization		
0%	Conjugate gradient	4.3%
5%	Conjugate gradient	9.7%
10%	Conjugate gradient	17%
15%	Conjugate gradient	25%
20%	Conjugate gradient	33%

Table 5.3. Test set 2 - 7186 Nodes - Standard tests

Test	Coarsening method	Error	Setup time	CG time	MG time
1 Level deep tests					
1 Cycle	Standard	99.9%	33.4s	195.1s	0.31s
	Bilinear	1100%	29.95s	198.8s	0.31s
	Approximate inverse	661%	38.39s	196.9s	0.32s
	CBW	109%	29.37s	195.9s	0.29s
	Multistep	54%	1518s	11632s	0.58s
2 Levels Deep Tests					
1 Cycle	Multistep	56%	19653s	24120s	4.82s
Vector	Standard	37%			
	Multistep	23%			
Matrix	Standard	37%			
	Multistep	56%			
Conjugate gradient with Tikhonov regularization		53%		6071s	
Conjugate gradient with limited iterations		80%		2740s	

Table 5.4. Test Set 2 - 7186 Nodes - Auxillary tests using CGNE for the final solve.

Test	Coarsening method	Error	Setup time	CG time	MG time
Multigrid 1 Cycle 1 Level deep	Multistep	72%	195.2s	5561s	11.85s
2 Levels deep	Multistep	65%	2734s	7295s	18.72s
3 Levels deep	Multistep	80%	11835s	5114s	54.94s
Multigrid cycling test (2 levels deep)					
2 Cycles	Multistep	59%	2734s	13079s	
3 Cycles	Multistep	66%		19682s	
4 Cycles	Multistep	925%		26253s	
Smoothing tests (2 Deep)					
1 Level deep	Multistep	75%	281.3s	7101s	
Postsmoothing		70%			
2 Levels deep	Multistep	72%	11112s	23449s	
Postsmoothing		70%			
2 Levels deep	Multistep				
Tests measuring varying tolerance levels all performed on 1 cycle with 1 level deep with multistep					
Weak connection tolerance (WCT) test					
WCT = 0.02		72%	11112s	23449s	
WCT = 0.05		75%	5427s	12342s	
WCT = 0.1		61%	2648s	6032s	
WCT = 0.15		80%	1431s	3323s	
WCT = 0.2		281%	872.5s	1888s	
WCT = 0.3		275%	404s	819.2s	
Varying stiffness matrix entry tolerance (SMET) test with (WCT = 0.1))					
SMET = 0.001		68%	281.6s	743.5s	
SMET = 0.01		68%	136.5s	450.3s	
SMET = 0.05		69%	70.13s	281.0s	
SMET = 0.1		100.2%	50.53s	219.1s	

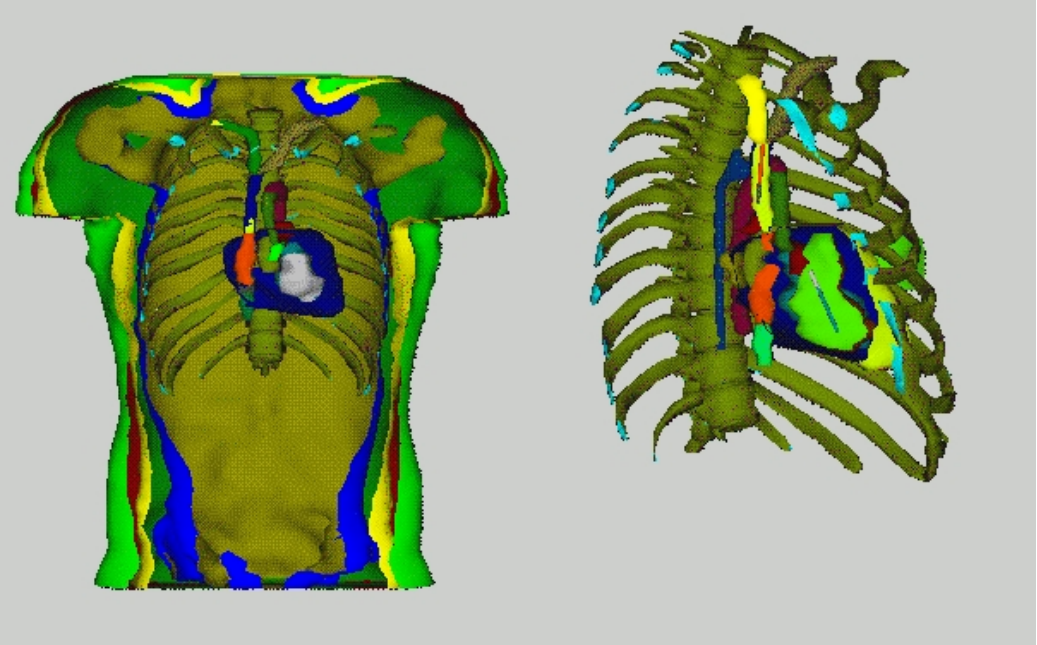

Figure 5.2. Interior view of the model for test set 2.

`Test set 3` The forward bioelectric field problem computed on the same model used in test set 2. These tests were based on a given vector of epicardial potentials and measured the iterative solvers estimate of the solution on the torso against a direct forward solve. Table 5.5 shows the simulation results.

`Test set 4` A two-dimensional model with 300 nodes. Of these, there were 40 epicardial nodes, 200 volume nodes, and 60 torso nodes as shown in Figure 5.3. This problem was created by generating nodes in concentric circles. Each node had a random value included in its cartesian coordinates and its position determined connected nodes. Nodes that were within a defined distance, and which would not cause connection overlap were connected, maintaining a proper triangular discretization. The stiffness matrix had 1400 nonzero entries and the inverse solve had an approximate condition number of $1x10^{6}$. Table 5.6 shows the simulation results.

`Test set 5` The same model as test set 4, except it was composed of 1195 nodes, of which there were 120 epicardial nodes, 995 volume nodes, and 80 torso nodes. The stiffness matrix had 6966 nonzero entries and the inverse solve had an approximate condition number of $3x10^{13}$. Table 5.7 shows the simulation results.

Table 5.5. Test set 3 forward problem - 7186 Nodes

Test	Coarsening method	Error	Setup time	CG time	MG time
1 Level deep tests					
1 Cycle	Bilinear	11%	3.51s	26.62s	0.29s
	Standard	21%	7.99s	25.58s	0.27s
	Approximate inverse	10%	13.05s	26.72s	0.27s
	CBW	31%	13.05s	26.72s	0.27s
	Multistep	3.1%	27.85s	960.1s	0.34s

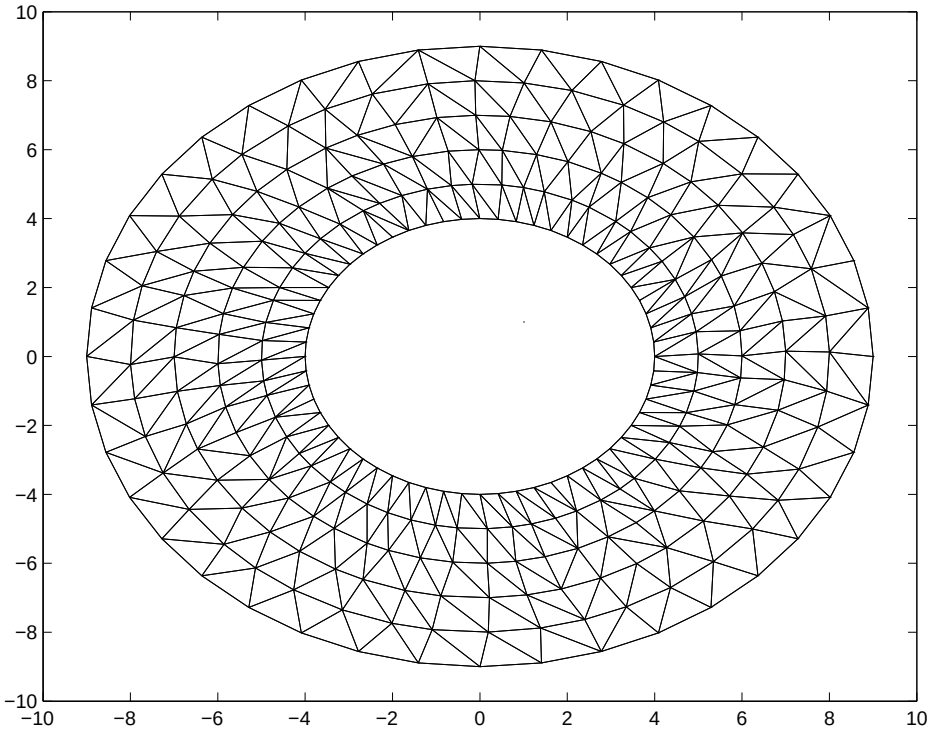

Figure 5.3. Mesh constructed for test set 4.

Table 5.6. Test set 4 - 300 Nodes

Test	Coarsening method	Error	Setup time	CG time	MG time
1 Level deep tests					
1 Cycle	Multistep	9.2%	0.35s	0.56s	Insig-
	Multistep w/ 3 smooths	9.3%		0.58s	nificant
	CBW	29%	0.10s	0.10s	
	Bilinear	21%	0.10s	0.9s	
	Standard	24%	0.15s	0.11s	
	Standard w/ 3 smooths	22%		0.11s	
	Approximation	20%	0.20s	0.11s	
Conjugate gradient with Tikhonov regularization		16%		2.35s	
Conjugate gradient with limited iterations		34%		0.73s	

Table 5.7. Test set 5 - 1195 Nodes

Test	Coarsening method	Error	Setup time	CG time	MG time
1 Level deep tests					
1 Cycle	Multistep	10%	3.19s	19.73s	
	Multistep w/ 3 smooths	10%		20.18	
	CBW	23%	0.43s	2.03s	
	Bilinear	20%	0.45s	2.06s	
	Standard	18%	0.68s	2.08s	
	Standard w/ 3 smooths	17%		2.11s	
	Approximation	19%	0.85s	2.32s	0.04s
Vector	Multistep	17%			
	CBW	17%			
	Bilinear	16%			
	Standard	19%			
	Approximation	17%			
Matrix	Multistep	19%			
	CBW	28%			
	Bilinear	20%			
	Standard	23%			
	Approximation	29%			
Conjugate gradient with Tikhonov regularization		12%		49.72s	
Conjugate gradient with limited iterations		32%		14.51s	

`Test set 6` The same model as test set 4, except it was composed of 2692 nodes, of which there were 180 epicardial nodes, 2392 volume nodes, and 120 torso nodes as shown in Figure 5.4. The stiffness matrix had 16,745 nonzero entries and the inverse solve had an approximate condition number of $5x10^{15}$. Table 5.8 shows the simulation results.

`Test set 7` The same model as test set 4, except it was composed of 4789 nodes, of which there were 240 epicardial nodes, 4389 volume nodes, and 160 torso nodes. The stiffness matrix had 30,724 nonzero entries. The inverse problem had an approximate condition number of $5x10^{15}$ and the forward problem had an approximate condition number of $1x10^{3}$. Table 5.9 shows the simulation results.

`Test set 8` The same torso model as test set 2, except it was a higher resolu-

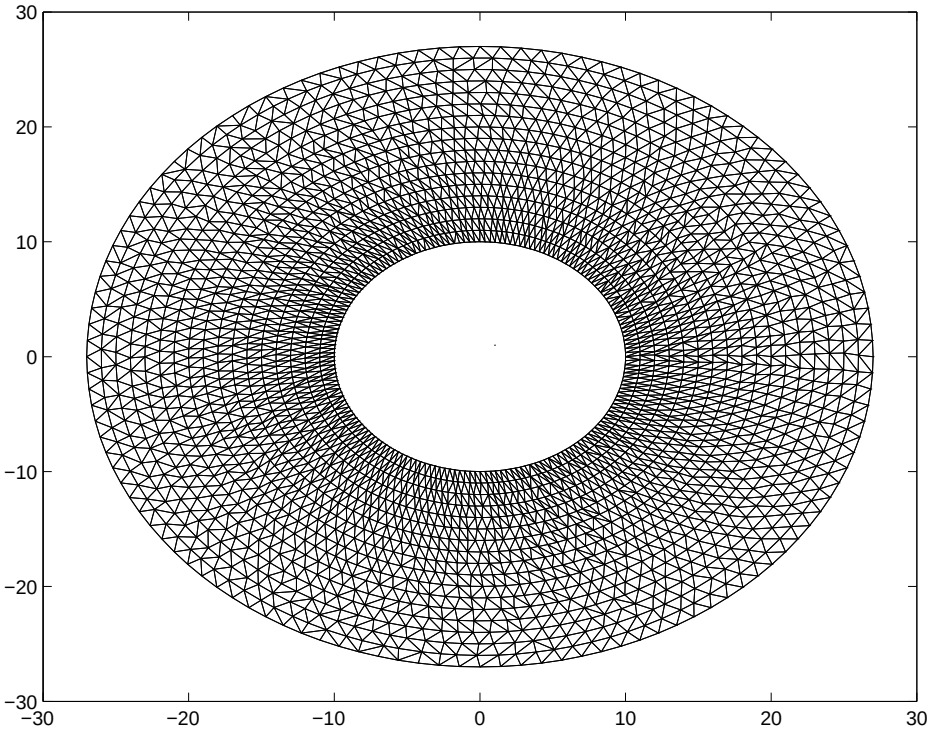

Figure 5.4. Mesh constructed for test set 6 problem.

Table 5.8. Test set 6 - 2692 Nodes

Test	Coarsening method	Error	Setup time	CG time	MG time
1 Level deep tests					
1 Cycle	Multistep	18%	8.70s	113.20s	
	Multistep w/ 3 smooths	15%		129.95s	
	CBW	30%	1.12s	11.84s	
	Bilinear	25%	1.09s	11.91s	
	Standard	25%	1.45s	11.45s	
	Approximate inverse	22%	2.03s	13.00s	0.10s
	Standard w/ 3 smooths	25%		11.59s	
Conjugate gradient with Tikhonov regularization		15%		275.85s	
Conjugate gradient with limited iterations		30%		79.74s	

tion model and composed of 156,728 nodes, 4735 of which were epicardial nodes, 146,937 volumes nodes, and 3056 torso nodes. The stiffness matrix had 2,463,486 nonzero entries. The inverse problem's condition number was incalculable by Matlab. The full set of tests were not run on this test set. To test scalability, I ran only a multistep and bilinear test. In addition, the test was broken into each level to see the speed and accuracy at every depth. The main test was a seven level deep test, with each of the deeper levels tested for one depth of coarse grid correction. The lower depths could not be tested as they would take too long. One test was performed with a complete one multigrid cycle with three levels of coarsening. Table 5.10 shows the simulation results.

5.2 Discussion

The data from this research show that different algorithms perform better in different situations. No one method proves itself to be better than the others all the time. The multigrid algorithms with a single v-cycle performed very competitively with the conjugate gradient methods in accuracy, and show potential for good scalability as seen in Figure 5.5.

Table 5.9. Test set 7 - 4789 Nodes

Test	Coarsening method	Error	Setup time	CG time	MG time
1 Level deep tests					
1 Cycle	Multistep	15%	18.29s	471.42s	
	Multistep w/ 3 smooths	13%		451.04s	
	Multistep bilinear	34%	24.45s	539s	0.23s
	Multistep CBW	12%	24.03s	543s	0.31s
	CBW	24%	2.06s	41.79s	
	Bilinear	23%	2.00s	41.52s	
	Standard	22%	2.91s	40.79s	
	Approximate inverse	22%	3.84s	44.23s	0.14s
	Standard w/ 3 smooths	22%		41.14s	
Vector	Multistep	16%			
	Multistep bilinear	12%			
	CBW	17%			
	Bilinear	16%			
	Standard	19%			
	Approximate inverse	15%			
Matrix	Multistep	21%			
	CBW	29%			
	Bilinear	31%			
	Standard	31%			
	Approximate inverse	35%			
2 Levels deep tests					
1 Cycle	Multistep	14%	24.03s	556s	0.31s
2 Cycles	Multistep	10%		1090s	
3 Cycles	Multistep	9.1%		1642s	
4 Cycles	Multistep	8.4%		2197s	
5 Cycles	Multistep	8.0%		2750s	
Conjugate gradient with Tikhonov regularization		13%		906.32s	
Conjugate gradient with limited iterations		29%		262.69s	

Table 5.10. Test set 8 - 156,728 Nodes

Test	Coarsening method	Error	Setup time	CG time	MG time
1 Cycle 7 Deep	Multistep		19,252s		
The test is broken down into each step					
Depth 1			4750s		
Depth 2			11,908s		
Depth 3			23,412s		
One deep coarse grid correction with given coarse grid solve depth					
Depth 4		106%	14,337s	144,671s	27.68
Depth 5		87%	24,586s	74,869s	19.14s
Depth 6		85%	8458s	41,340s	14.08s
This a full depth test					
3 Deep - 1 Cycle		88%		327,161s	153.7s
1 Cycle 1 Deep	Bilinear	908%	139.1s	35,573s	7.90s
2 Deep		166%	426.9s	1276s	22s
3 Deep		97%	672.8s	12.13s	28.6s

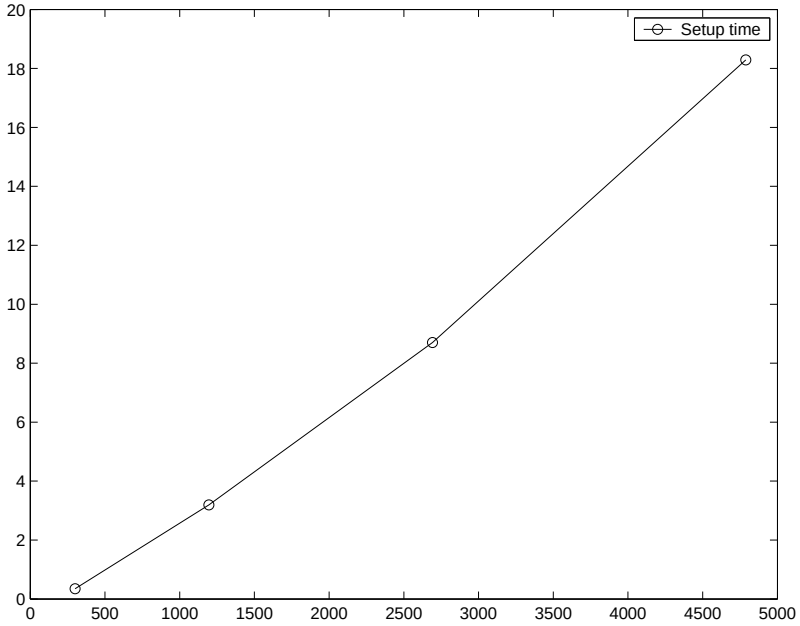

Figure 5.5. Performance scalability for multistep multigrid. Multigrid setup time (y-axis) in seconds is plotted versus the number of nodes on the x-axis for problems solved using the multistep algorithm on test sets 4-7.

Figures 5.6 - 5.12 depict the potentials through the torso of the two-dimensional slice in test set 1 as multigrid executed each cycle and converged closer to the correct solution. This example used a random vector for epicardial potentials and a forward solution determined the simulated measured torso potentials input. This test utilized the multistep transfer operator. The error decreased significantly with each cycle in these figures, and a careful inspection also reveals smoother estimates and errors after each smoothing step. This example also illustrates the spatial smoothing and attenuation of potentials that occurs within the body.

Among the multigrid coarsening methods, the multistep method was most accurate, while the bilinear method was fastest. For example, on test set 2, multistep multigrid was the only multigrid algorithm sufficiently accurate to achieve convergence. However, for the same test set, bilinear finished approximately 28 times faster. Generally, the bilinear method performed well, especially on the vector

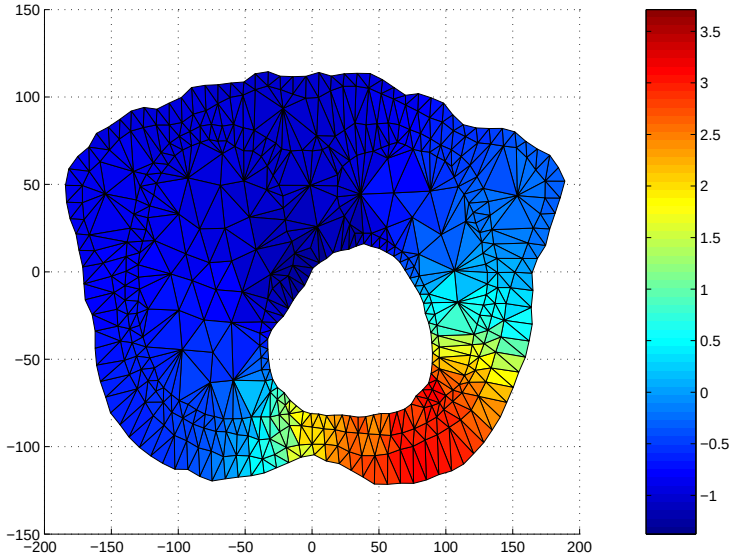

Figure 5.6. Test set 1 after one cycle (no smoothing). The colors represent potentials throughout the torso.

manipulation. The hybrid methods of combining different weighting schemes with the multistep method did not perform as well as the default choice of standard AMG weighted multistep. The conjugate gradient method was robust and predictable, albeit with a performance penalty when used with larger problems.

Multigrid methods show positive results, especially with single cycle tests, but unfortunately, the multigrid algorithms did not always continue to converge quickly and for the larger, more difficult problems, multigrid algorithms began to diverge after a few v-cycles, while the conjugate gradient continued to converge and remain stable. Although, with the large three-dimensional problems, all the algorithms performed poorly, most likely due in part to extreme ill-conditioning of the matrix without finely tuned regularization.

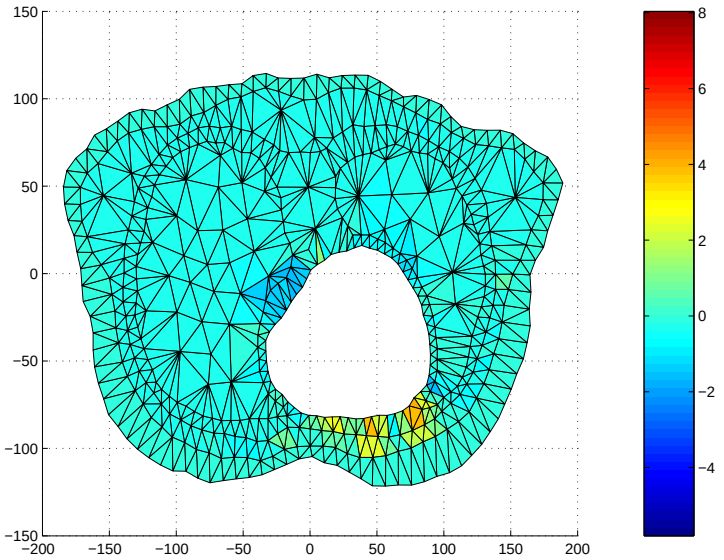

Figure 5.7. Test set 1 after one cycle (no smoothing). The colors represent the error in the computation (15% RMS error).

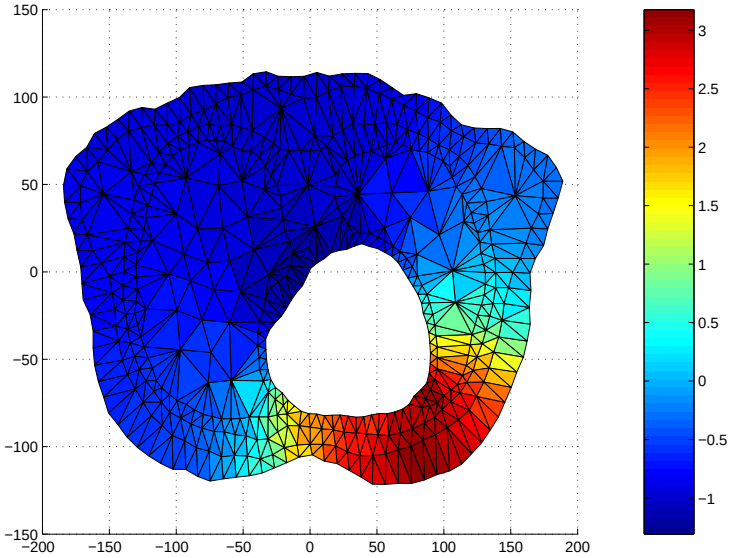

Figure 5.8. Test set 1 after one cycle and postsmoothing (with three smooths). The colors represent potentials throughout the torso.

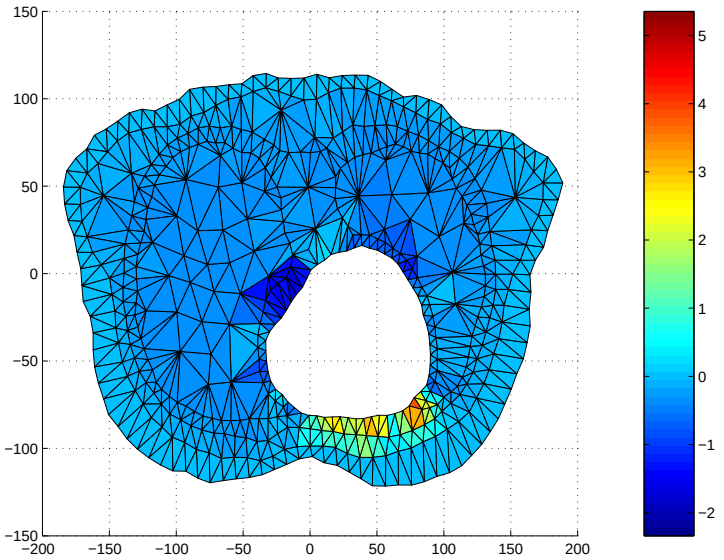

Figure 5.9. Test set 1 after one cycle and postsmoothing (with three smooths). The colors represent the error in the computation (13% RMS error).

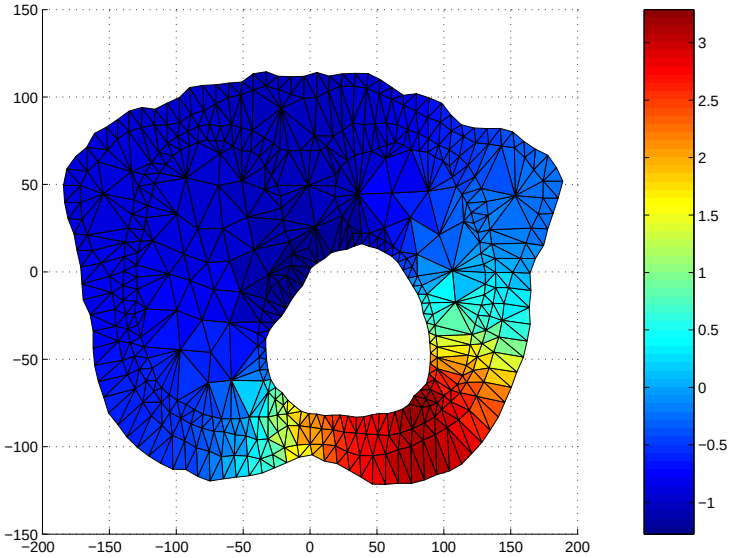

Figure 5.10. Test set 1 after three cycles (three smooths per cycle). The colors represent potentials throughout the torso.

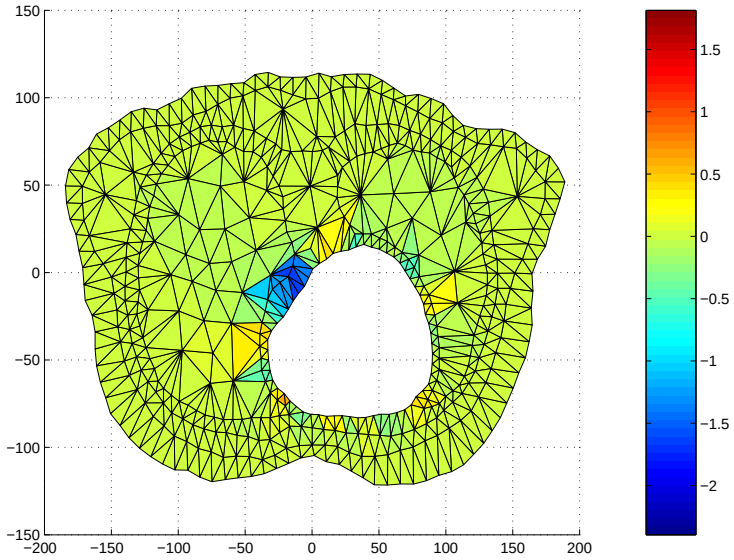

Figure 5.11. Test set 1 after three cycles (three smooths per cycle). The colors represent represent the error in the computation (5.6% RMS error).

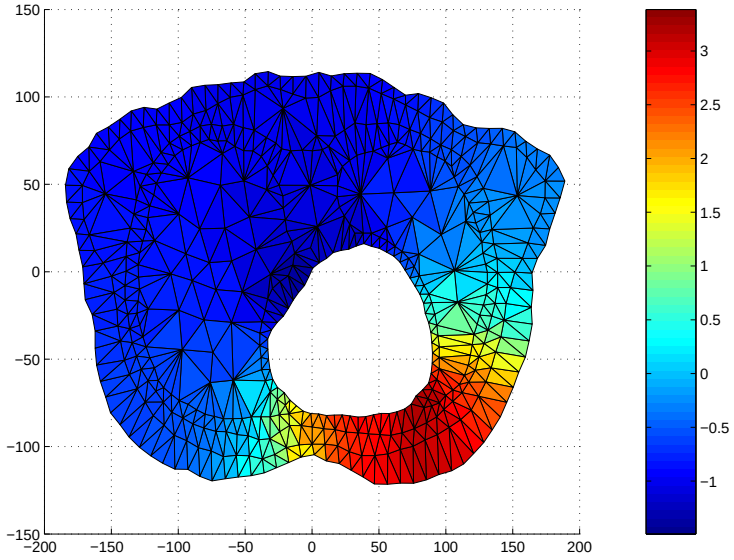

Figure 5.12. Test set 1 correct solution. The colors represent potentials throughout the torso.

With the smaller test set 1, after three cycles we had a close approximation to the solution, but with the larger test set 2, after three cycles, significant divergence occurred. This is because of the inability of the smoother to assist in convergence. The smoother must be able to reduce high-frequency error, but the Gauss-Seidel algorithm that I used was ineffective on the very ill-conditioned problems. The results show that the error measurements for postsmoothing do not exhibit substantially less error than the presmoothing error. The importance of an effective smoother becomes greater with deeper coarsening. Solution approximation relies mostly on the coarse grid solve with solves that do not use many coarse grid layers, but as the coarse grids go deeper the solution becomes more dependent on the smoothers ability to converge towards a solution. This is one of test set 8's major hindrances toward achieving high accuracy results. Deep coarse grids are a necessity with such large test sets and therefore it is unavoidable to incur an accuracy penalty without effective smoothing. An effective smoother must be able to quickly iterate towards an accurate estimate of the relative values of neighboring nodes. That is, the smoother must rapidly eliminate high frequency error in the estimate. Gauss-Seidel did not do this adequately with the inverse bioelectric field problem.

We can see a clear tradeoff between accuracy and computational efficiency with the multistep algorithm compared with the other weighted algorithms (Figure 5.13). The multistep is definitely more accurate. In general, the multistep algorithm roughly halved the error in the coarse grid correction. However, this advantage came with substantial slowing. The setup process involves a greater number of steps to perform the interpolation operator which caused this step to take approximately eight times as long to complete. The multistep approach also causes a drastic increase in the number of nonzeros in the coarsened matrices. The increased number of nonzeros raised the solve time by a factor of approximately 10.

When performing calculations on the simpler, well-conditioned two-dimensional problems (and forward problems as they are much more well-conditioned) the superior speed of the single step approaches are certainly more desirable. The approximate inverse and bilinear approach were the most accurate methods on

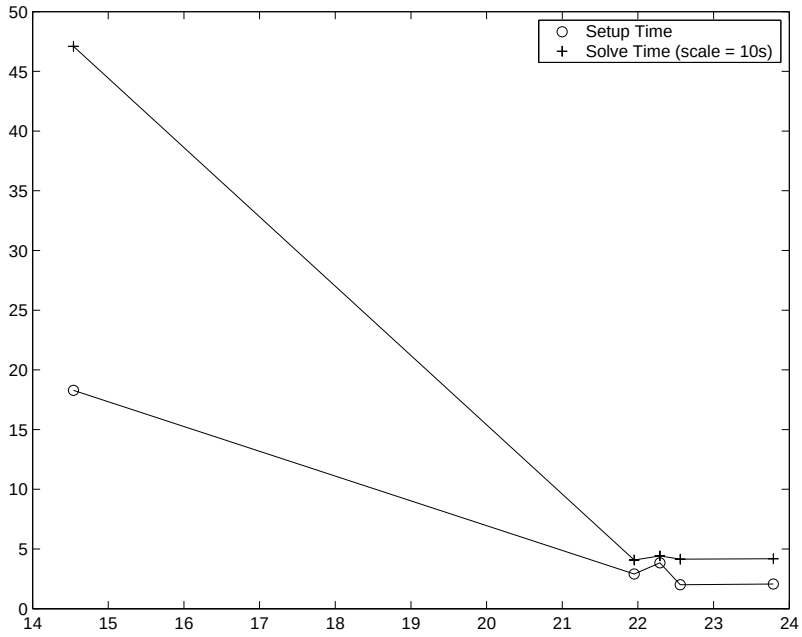

Figure 5.13. The tradeoff between accuracy and performance of the different algorithms. Setup time and solve time in seconds (y-axis) are plotted versus the percentage error in the solve on the x-axis for the different algorithms applied to test set 7.

the well-conditioned problems, although the bilinear method would probably be preferable because of its speed. Part of the bilinear effectiveness was due to its accuracy with the vector interpolation. As the problems grew in difficulty the standard approach become very competitive, but bilinear still proved to be a winner for the easier problems. However, with the more difficult ill-conditioned problems, the improved accuracy of multistep approach is the only viable method for multigrid to converge. The slower performance of multistep must be tolerated. While slower performance is one obstacle we are trying to overcome with this algorithm, once again, note that if linear $O(n)$ convergence can be maintained, then even this decreased performance will be faster than other non-multilevel algorithms on large datasets.

5.3 Scalability

The scalability of a near $O(n)$ algorithm is a significant reason for using multigrid. So the question arises, does multigrid continue to scale linearly with the inverse problem? If we look at the generated problems we get a sampling of sizes over about a factor of 23. Figures 5.14 and 5.5 show the setup time as the number of nodes increase. Figure 5.14 is with the standard AMG algorithm and Figure 5.5 is with the multistep algorithm. Both of the plots only slightly deviated from a straight line through the origin.

From these plots, we can see that the setup time and the multigrid processing time (apart from the conjugate gradient final solve) were proportional to the number of nodes. For example, with simple bilinear weighting the setup processed approximately 250 nodes per second and the multigrid steps went through about 30,000 nodes per second per cycle. On the other hand, the conjugate gradient

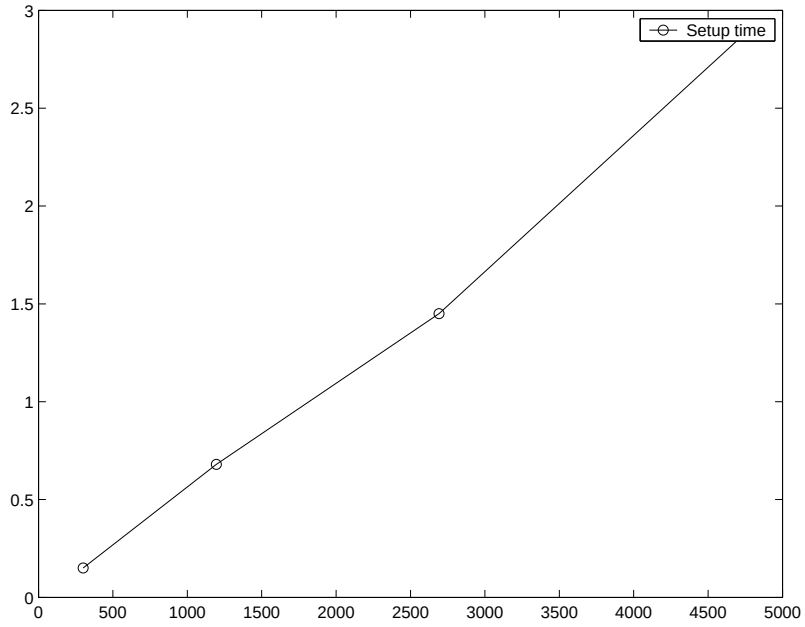

Figure 5.14. Performance scalability for standard multigrid. Multigrid setup time in seconds (y-axis) is plotted versus the number of nodes (x-axis) for problems solved using the multigrid standard algorithm on test sets 4-7.

method followed the expected $O(n^2)$ time for convergence (see Figure 5.15). For this reason, I did not run conjugate gradient on the largest test set, as it would take approximately 3.4 million seconds, or 40 days to finish with the computer I was using.

One drawback to the multistep algorithm is that a large number of nonzeros are retained or accumulated in the coarsened matrix as the coarsening deepens. This means the final solve and smoothing steps are slower. However, each step in an iterative solver is proportional to the nonzeros in the A matrix. The number of nonzeros is linearly proportional to the size of the A matrix as we scale up, so the relaxation steps still retain $O(n)$ speed. Generally, the final solve is performed at the coarsest level and the multigrid coarsens until a certain size limit is achieved. Therefore, the final solve is usually solvable in constant time. If an iterative solver is used for the final solve, with most of the nonzeros being retained as the algorithm coarsens, the final solve becomes slower as the problem scales larger. For conjugate

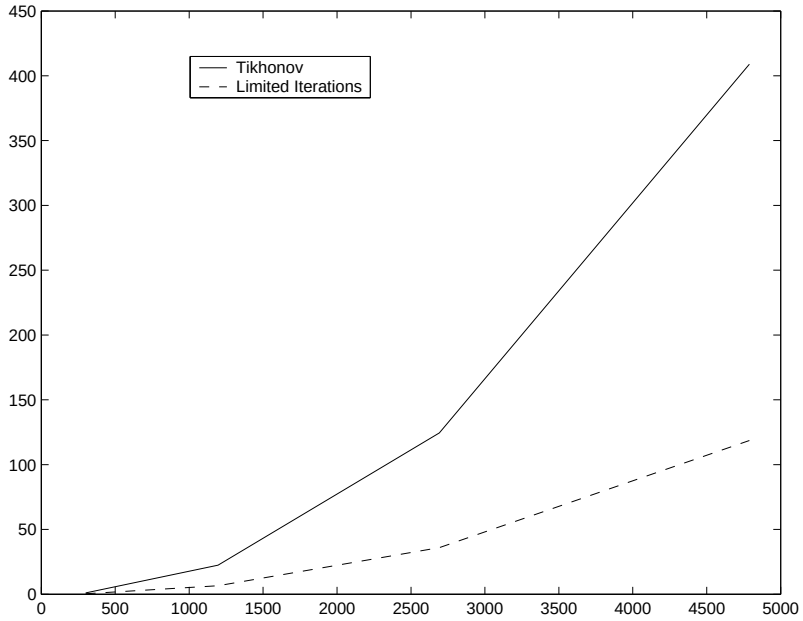

Figure 5.15. The solve time (y-axis) versus the number of nodes (x-axis) in the problem using the conjugate gradient algorithm for test sets 4-7.

gradient the time is proportional to the number of nonzeros (plus n, but the contributions of nonzeros form the dominant term) multiplied by n (the size of the matrix). But, we see that with n assumed constant (n being the size of the final coarsened matrix, is approximately constant as long as the final solve coarsens to within a given maximum size), and the nonzeros will be proportional to the initial problem size, because the number of connections for each node does not change with the problem size. Therefore, the solve time will remain linear, and we continue to avoid any step that would prevent $O(n)$ convergence.

With the large scale dataset (test set 8), we begin to see that the time is not the only cost with the nonzero accumulation, but that memory space becomes an issue. In Figure 5.16, we can see growth of the nonzeros even though the nodes reduce. The graph is based on a tolerance level of 0.3 for the standard AMG algorithm used with the multistep approach. This tolerance level was necessary to stay within the

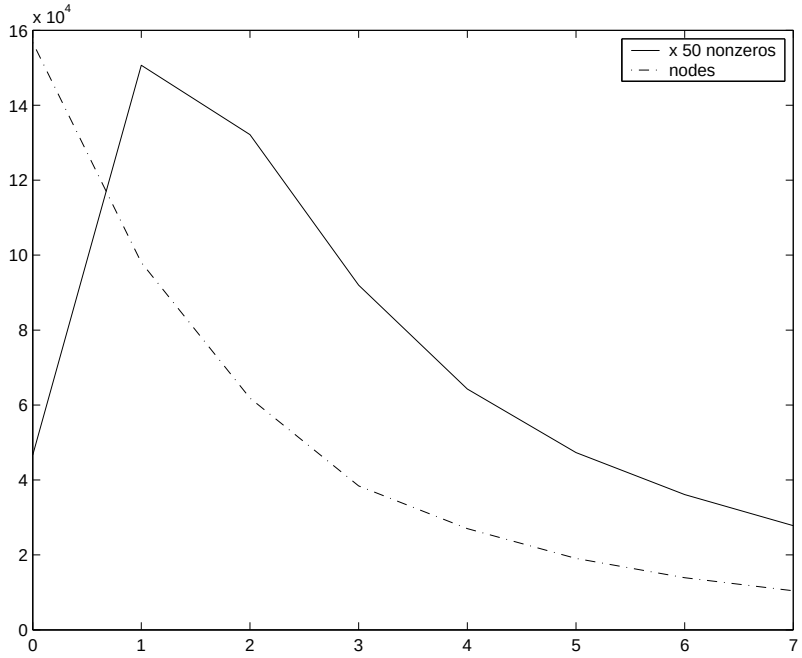

Figure 5.16. The number of nodes and the number of nonzeros (y-axis) versus the depth (x-axis) during the coarsening steps for test set 8.

memory limits (2 Gigabytes).

The multiplication of A^T with A before adding the Tikhonov regularization (from equation 3.12) is another time-consuming step with my implementation that must be performed at the coarsest level before the conjugate gradient solve. This step involves a matrix-matrix multiplication. The time required for such a multiplication is proportional to the number of nonzeros squared divided by n (the size of the matrix). With a sparse diagonal only matrix this would breakdown to $O(n)$, and with a fully dense matrix this becomes $O(n^3)$ because the nonzeros equals n^2. Since the number of nonzeros cannot be greater than the square of the size matrix (nz is the number of nonzeros and n is the size of the matrix):

$$nz^2 \leq n \tag{5.1}$$

So we can write:

$$\frac{nz^2}{n} \leq nz \cdot n \tag{5.2}$$

Therefore, in theory, the matrix-matrix multiply will always be faster than the conjugate gradient process, so the $O(n)$ term remains dominant. However, due to inefficiencies in my implementation, I did not perform this step with larger matrices, and instead used the faster and more efficiently implemented conjugate gradient normal equation solving algorithm.

The comparison test between multigrid and conjugate gradient solutions on test set 1 depicted how closely multigrid converged to the exact conjugate gradient Tikhonov solution. With one deep coarsening, multigrid converged to solution 2% different than the conjugate gradient solution. However, using multigrid with two deep coarsening showed convergence to a solution approximately 15% different from the exact Tikhonov solution.

The error in the problems rose slightly with the increase in number of nodes (see Figure 5.17). However, this small aberration is most likely related to the greater ill-conditioning associated with the larger problems and does not appear to indicate a convergence rate that is node size dependent, or at least not to the degree to diverge significantly from $O(n)$ convergence.

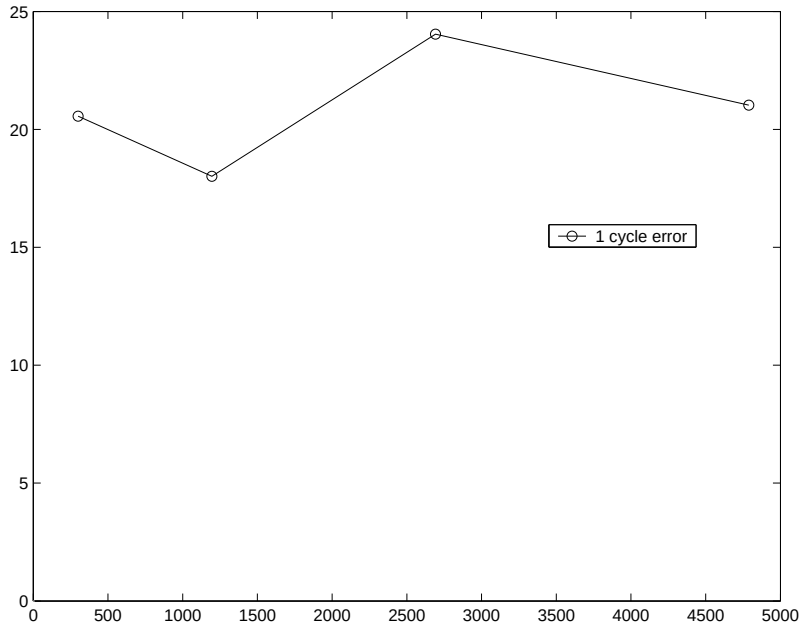

Figure 5.17. The effect of problem size on accuracy. The error in the solution (y-axis) is plotted versus the number of nodes (x-axis) for test sets 4-7

Also, the test of varied measurement noise showed the solution error of multigrid increased linearly in relation to the addition of noise to the surface initial conditions. In addition, multigrid demonstrated a greater accuracy than conjugate gradient when the additive noise levels were increased. With 20% RMS error added to the torso initial conditions, multigrid converged to 26% error after five cycles and conjugate gradient converged to 33% error.

5.4 Future Work

Fully implementing multigrid to effectively solve the inverse bioelectric field problem is a large and involved project that will most likely require the most advanced multigrid techniques, parallelized implementation, and advanced hardware. This research is the first step to achieving this goal.

For convergence, it is necessary to implement a robust, accurate smoother that can reduce high-frequency error on an ill-conditioned problem. The lack of

such a smoother in this study, resulted in the multigrid diverging after several iterations for the larger test sets (2 and 8). While the multigrid successfully reduced low-frequency error in the beginning cycles, with very little high-frequency error reduction, stability was difficult to sustain. Smaller test sets had reasonable success with the Gauss-Seidel smoothing (see Figures 5.6 - 5.12 for a full sequence of the potential values in test set 1 progressing through a multigrid cycle), but for larger, more ill-conditioned problems, improved smoothers are necessary.

In order to achieve true scalability, an improved method of dropping connections during grid coarsening must be applied. With the matrix density becoming an important issue with larger datasets, it would be beneficial to be able to intelligently eliminate smaller, fewer important entries and adjust the remaining entries to maintain accuracy. As mentioned earlier, with the standard algorithm, the higher tolerance levels for categorizing connections into weak connections causes the accuracy of transfer operator to suffer greatly. To apply this approach with large datasets, with reasonable memory constraints, further research is needed for a method of deciding which connections will have the least effect on the system when dropped, and how adjacent connections can be altered to compensate for the dropped connections.

Multigrid's main advantage is its scalability and, therefore, is most needed in problems with a large number of nodes. In order for multigrid to truly be a high performance algorithm, it must be programmed to run on multiple processors. Fortunately, all aspects of multigrid work in a local manner and therefore it inherently works efficiently in a parallel structure. The grid can be broken up into blocks for every step of multigrid, including coarsening, interpolation, restriction, and smoothing. The one exception is the coarsest level solve. This depends on what method one uses to perform this solve. If you use a serial solver, with a parallelized implementation, using many levels of coarsening to achieve a very coarse deep level would lead to a quicker coarse level solve. Of course, programming a complex algorithm in parallel is more difficult and dealing with boundaries between blocks adds to the programming difficulty.

There are a number of optimizations that can be utilized to enhance performance. Research should be performed on optimizing the parameters for multigrid. There are different tolerance levels (strong versus weak connections, and stiffness matrix entry) that can be adjusted; the regularization parameters can also be adjusted. Also, different numbers of semicoarsened sublevels for each multistep level could be tested. In addition, there are different coarse grid selections schemes that could be researched (for the creation of the coarse grid goal for the multistep approach). Another important contribution would be to determine an appropriate stopping criteria for the multigrid cycles.

5.5 Conclusion

My research introduces an approach to implementing algebraic multigrid for the inverse bioelectric field problem. This involves a specific formulation of the surface to surface equation with regularization and a corresponding algebraic multigrid implementation designed to work with this problem, using separate stiffness and geometry matrices. This work shows how this approach can be applied to smaller problems and achieve promising results. My initial work on this implementation has been published in a joint work with Chris Johnson, Marcus Mohr, Ulrich Rüde, and Alexey Samsonov for the *Yosemite Workshop on Multilevel Methods* in 2001[65].

In addition to detailing an algebraic multigrid implementation, a multistep approach has also been described to handle the coarse grid correction inaccuracies introduction with the ill-conditioned matrices. While the choice of a transfer operator is a tradeoff between accuracy and computational efficiency, the multistep approach I have described shows a significant improvement in accuracy, but at a large performance cost. With sensitive ill-conditioned problems like the inverse bioelectric field problem, this tradeoff needs to stress accuracy more than in typical applications of multigrid. Most applications of multigrid do not require this level of accuracy and the multistep algorithm will only slow the process down, but this algorithm effectively deals with the sensitivity of the inverse bioelectric problem and makes it possible to solve using multigrid.

Multigrid proves useful in a large number of scientific computing arenas such thermodynamics, fluid dynamics, and radiation simulation. In the bioengineering arena, the treatment of a number of major medical issues would benefit greatly from informative diagnosis assistance from accurate internal data through non-invasive electrocardiography and electroencephalography tests. With multigrid successfully applied in this field, the increase in speed and accuracy could be a significant contribution to the treatment of a number of medical problems. Hopefully, these findings will allow researchers to wisely choose transfer operators so that multigrid may be used to efficiently solve the inverse bioelectric field problem.

REFERENCES

[1] Ahmad, G., Brooks, D., Jacobson, C., and MacLeod, R. A feasability study of inverse electrocardiography by convex optimization. In *Proceedings of the 21st N.E. Bioengineering Conference* (Bar Harbour, ME, 1995), pp. 245–246.

[2] Ahmad, G., Brooks, D. H., and MacLeod, R. An admissible solution approach to inverse electrocardiography. *Annal. Biomed. Eng. 26* (1998), 278–292.

[3] Akcelik, V., and Ghattas, O. Multiscale Newton-Krylov methods for inverse wave propagation. In *Copper Mountain Conference on Iterative Methods* (2002).

[4] Bader, M. A robust and parallel multigrid method for convection diffusion equations. In *Copper Mountain Conference on Multigrid Methods* (2001).

[5] Barr, R. *Constrained Inverse Electrocardiography.* PhD thesis, Duke University, Durham, NC, 1968.

[6] Barr, R., and Spach, M. Inverse solutions directly in terms of potentials. In *The Theoretical Basis of Electrocardiography*, C. Nelson and D. Geselowitz, Eds. Clarendon Press, Oxford, 1976, pp. 294–304.

[7] Belge, M., Kilmer, M. E., and Miller, E. L. Efficient determination of multiple regularization parameters in a generalized l-curve framework.

[8] Bordner, J. An adaptive multilevel method for anisotropic elliptic problems, 1999. University of Illinois at Urbana-Champaign.

[9] Brandt, A. Multi–level adaptive technique (MLAT) for fast numerical solution to boundary value problems. In *Proceedings of the Third International Conference on Numerical Methods in Fluid Mechanics* (Berlin, 1973), H. Cabannes and R. Teman, Eds., vol. 18 of *Lecture Notes in Physics*, Springer–Verlag, pp. 82–89.

[10] Brandt, A. Algebraic multigrid theory: The symmetric case. *Appl. Math. Comput. 19* (1986), 23–56.

[11] Brandt, A. General highly accurate algebraic coarsening. *Elect. Trans. Numer. Anal. 10* (2000), 1–20.

[12] Brandt, A., McCormick, S., and Ruge, J. Algebraic multigrid (amg) for automatic multigrid solutions with applications to geodetic computations, 1982. Institute for Computational Studies, POB 1852, Fort Collins, Colorado.

[13] Brandt, A., McCormick, S. F., and Ruge, J. W. Algebraic multigrid (AMG) for sparse matrix equations. In *Sparsity and Its Applications*, D. J. Evans, Ed. Cambridge University Press, Cambridge, 1984.

[14] Briggs, W., Hensen, V., and McCormick, S. *A Multigrid Tutorial*, 2nd ed. SIAM, Philadelphia, 2000.

[15] Brooks, D., Ahmad, G., and MacLeod, R. Multiply constrained inverse electrocardiology: Combining temporal, multiple spatial, and iterative regularization. In *Proceedings of the IEEE Engineering in Medicine and Biology Society 16th Annual International Conference* (1994), IEEE Computer Society, pp. 137–138.

[16] Brooks, D., Ahmad, G., MacLeod, R., and Maratos, G. Inverse electrocardiography by simultaneous imposition of multiple constraints. *IEEE Trans Biomed. Eng. 46*, 1 (1999), 3–18.

[17] Burger, H., and van Milaan, J. Heart-vector and leads. Part I. *Br. Heart J. 8* (1946), 157–61.

[18] Burger, H., and van Milaan, J. Heart-vector and leads. Part II. *Br. Heart J. 9* (1947), 154–60.

[19] Burger, H., and van Milaan, J. Heart-vector and leads. Part III: Geometrical representation. *Br. Heart J. 10* (1948), 229–33.

[20] Burnes, J., Ghanem, R., Waldo, A., and Rudy, Y. Imaging dispersion of myocardial repolarization, i: comparison of. *Circ. 104*, 11 (Sep 2001), 1299–1505.

[21] Burnes, J., Taccardi, B., and Rudy, Y. A noninvasive imaging modality for cardiac arrhythmias. *Circ. 102* (2000), 2152–2158.

[22] Burns, J., Taccardi, B., MacLeod, R., and Rudy, Y. Noninvasive electrocardiographic imaging of electrophysiologically abnormal substrates in infarcted hearts: A model study. *Circ. 101* (2000), 533–540.

[23] Cuppen, J. Calculating the isochrones of ventricular depolarization. *SIAM J. Sci. Statist. Comp. 5* (1984), 105–120.

[24] Cuppen, J., and van Oosterom, A. Model studies with the inversely calculated isochrones of ventricular depolarization. *IEEE Trans Biomed. Eng. 31* (1984), 652–659.

[25] Demkowicz, L., Oden, J., Rachowicz, W., and Hardy, O. Toward a universal h-p adaptive finite element strategy, Part 1. Constrained approximation and data structure. *Computational Methods in Applied Mechanical Engineering 77* (1989), 79.

[26] Douglas, C. C., and Douglas, M. B. Mgnet bibliography. Tech. rep., Yale University, Department of Computer Science, New Haven, CT (USA), 1991-1999.

[27] Driscoll, D., and Rush, S. EEG electrode sensitivity - an application of reciprocity. *IEEE Trans. Biomed. Eng. 16* (1969), 15–22.

[28] Fedorenko, R. P. A relaxation method for solving elliptic difference equations. *Z. Vycisl. Mat. i. Mat. Fiz. 1* (1961), 922–927. Also in U.S.S.R. Comput. Math. and Math. Phys., 1 (1962), pp. 1092–1096.

[29] Franzone, P. C., Gassaniga, G., Guerri, L., Taccardi, B., and Viganotti, C. Accuracy evaluation in direct and inverse electrocardiology. In *Progress in Electrocardiography* (1979), P. Macfarlane, Ed., Pitman Medical, pp. 83–87.

[30] Franzone, P. C., Guerri, L., Taccardi, B., and Viganotti, C. The direct and inverse problems in electrocardiology. Numerical aspects of some regularization methods and applications to data collected in isolated dog heart experiments. *Lab. Anal. Numerica C.N.R. Pub. N* (1979), 222.

[31] Franzone, P. C., Guerri, L., Taccardi, B., and Viganotti, C. Finite element approximation of regularized solution of the inverse potential problem of electrocardiography and application to experimental data. *Calcolo 22* (1985), 91.

[32] Franzone, P. C., Taccardi, B., and Viganotti, C. An approach to inverse calculation of epicardial potentials from body surface maps. *Adv. Cardiol. 21* (1978), 50–54.

[33] Gavit, L., Baillet, S., Mangin, J., Pescatore, J., and Garnero, L. A multiresolution framework to meg/eeg source imaging. *IEEE Transactions on Biomedical Engineering 48*, 10 (2001).

[34] Gelernter, H., and Swihart, J. A mathematical-physical model of the genesis of the electrocardiogram. *Biophys. J. 4* (1964), 285–301.

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

[36] Giorgi, C. ad Nadeau, R., Primeau, R., Campa, M., Cardinals, R., Shenasa, M., and Pagé, P. Comparative accuracy of the vectorcardiogram and electrocardiogram in the localization of the accessory pathway in patient with Wolff-Parkinson-White syndrome: Validation of a new vectorcardiographic algorithm by intrapoerative epicardial mapping and electrophysiological studies. *Am. Heart J. 119* (1990), 592–598.

[37] Golgolab, A. Malilleur tridimensionnel automatique pour des geometries complexes. Tech. rep., Rapport de Recherche no. 1004, INRIA, 1989.

[38] Golub, G., Heath, M., and Wahba, G. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics 21* (1979), 215–223.

[39] Golub, G., and Loan, C. V. *Matrix Computations.* Johns Hopkins, Baltimore, 1989.

[40] Green, L., Lux, R., Haws, C., Williams, R., Hunt, S., and Burgess, M. Effects of age, sex, and body habitus on QRS and ST-T potential maps of 1100 normal subjects. *Circ. 71* (1985), 244–253.

[41] Greensite, F., Huiskamp, G., and van Oosterom, A. New quantitative and qualitative approaches to the inverse problem of electrocardiology: their theoretical relationship and experimental consistency. *Medical Physics 17*, 3 (1990), 369–379.

[42] Gulrajani, R., Roberge, F., and Mailloux, G. The forward problem of electrocardiography. In *Comprehensive Electrocardiology*, P. Macfarlane and T. V. Lawrie, Eds. Pergamon Press, Oxford, England, 1989, pp. 197–236.

[43] Gulrajani, R., Roberge, F., and Savard, P. The inverse problem of electrocardiography. In *Comprehensive Electrocardiology*, P. Macfarlane and T. V. Lawrie, Eds. Pergamon Press, Oxford, England, 1989, pp. 237–288.

[44] Gulrajani, R., Savard, P., and Roberge, F. The inverse problem in electrocardiography: Solutions in terms of equivalent sources. *Crit. Rev. Biomed. Eng. 16* (1988), 171–214.

[45] Hackbusch, W. Multi-grid methods and applications. Tech. rep., Springer, Berlin, 1985.

[46] Hackbusch, W. Iterative solution of large sparse systems. Tech. rep., Springer, New York, 1994.

[47] Hackbush, W. Ein iteratives verfahren zur schnellen auflosung ellipticasher randwertprobleme. Tech. rep., Universitat Koln, 1976.

[48] Hadamard, J. Sur les problemes aux derivees parielies et leur signification physique. *Bull. Univ. of Princeton* (1902), 49–52.

[49] Halgren, E., Baudena, P., Clarke, J., Hite, G., Liegeois, C., Shauvel, P., and Musolino, A. Intracerebral potentials to rare target and distractor auditory and visual stimuli: I. Superior temporal lobe. *Electroencephalography and clinical Neurophysiology 94* (1995), 191–220.

[50] Hanke, M. Conjugate gradient type methods for ill posed problems. In *Pitman Reserch Notes in Mathematics Series.* Longman, 1995.

[51] HANSEN, P. Analysis of discrete ill-posed problems by means of the L-curve. *SIAM Review 34*, 4 (1992), 561–580.

[52] HUISKAMP, G., AND VAN OOSTEROM, A. The depolarization sequence of the human heart surface computed from measured body surface potentials. *IEEE Trans Biomed. Eng. 35* (1989), 1047–1059.

[53] HUISKAMP, G., AND VAN OOSTEROM, A. Tailored versus standard geometry in the inverse problem of electrocardiography. *IEEE Trans Biomed. Eng. 36* (1989), 827–835.

[54] HUISKAMP, G., VAN OOSTEROM, A., AND GREENSITE, F. Physiologically based constraints in the inverse problem of electrocardiography. In *Proceedings of the IEEE Engineering in Medicine and Biology Society 17th Annual International Conference* (1995), IEEE Press, pp. 207–208.

[55] J. THOMAS, B. DISKIN, A. B. Textbook multigrid efficiency for the incompressible navier-stokes equations: High reynolds number wakes and boundary layers. In *Copper Mountain Conference on Multigrid Methods* (2001).

[56] J. VAN LENT, S. V. Multigrid waveform relaxation for anisotropic partial differential equations. In *Copper Mountain Conference on Multigrid Methods* (2001).

[57] JIN, J. *The Finite Element Method in Electromagnetics.* John Wiley and Sons, 1993.

[58] JOHNSON, C. *The Generalized Inverse Problem in Electrocardiography: Theoretical, Computational and Experimental Results.* PhD thesis, University of Utah, Salt Lake City, Utah, 1989.

[59] JOHNSON, C. Computational and numerical methods for bioelectric field problems. *Critical Reviews in BioMedical Engineering 25*, 1 (1997), 1–81.

[60] JOHNSON, C. Adaptive finite element and local regularization methods for the inverse ECG problem. In *Inverse Problems in Electrocardiology*, P. Johnston, Ed., Advances in Computational Biomedicine. WIT Press, 2000.

[61] JOHNSON, C., AND MACLEOD, R. Inverse solutions for electric and potential field imaging. In *Physiological Imaging, Spectroscopy, and Early Detection Diagnostic Methods*, R. Barbour and M. Carvlin, Eds., vol. 1887. SPIE, 1993, pp. 130–139.

[62] JOHNSON, C., AND MACLEOD, R. Nonuniform spatial mesh adaptation using a posteriori error estimates: applications to forward and inverse problems. *Applied Numerical Mathematics 14* (1994), 311–326.

[63] JOHNSON, C., AND MACLEOD, R. Local regularization and adaptive methods for the inverse Laplace problem. In *Biomedical and Life Physics*, D. Ghista, Ed. Vieweg-Verlag, Braunschweig, 1996, pp. 224–234.

[64] JOHNSON, C., AND MACLEOD, R. Adaptive local regularization methods for the inverse ECG problem. *Progress in Biophysics and Biochemistry 69* (1998), 405–423.

[65] JOHNSON, C., MOHR, M., RUEDE, U., SAMSONOV, A., AND ZYP, K. Multilevel methods for inverse bioelelectric field problems. In *Yosemite Workshop on Multilevel Methods*, Lecture Series in Computer Science. Springer-Verlag, 2001, p. (in press).

[66] JOHNSON, C., AND POLLARD, A. Electrical activation of the heart: Computational studies of the forward and inverse problems in electrocardiography. In *Computer Assisted Modeling on the IBM 3090*, K. Billingsley, H. B. III, and E. Durohanes, Eds. Baldwin Press, University of Georgia, Athens, Georgia, 1992, pp. 583–628.

[67] KETTLER, R. Analysis and comparison of relaxation schemes in robust multigrid and preconditioned conjugate gradient methods. In *Multigrid Methods* (Berlin, 1982), W. Hackbusch and U. Trottenberg, Eds., vol. 960 of *Lecture Notes in Mathematics*, Springer–Verlag, pp. 502–534.

[68] KICKINGER, F. Algebraic multigrid for discrete elliptic second order problems. Tech. rep., MGNet, November 1997.

[69] KIM, C. A two-level preconditioner for an anisotropic mixed finite element problem. In *Copper Mountain Conference on Multigrid Methods* (2001).

[70] KNUPP, P., AND STEINBERG, S. *Fundamentals of Grid Generation.* CRC Press, Boca Raton, 1993.

[71] KUNDU, A., AND MITRA, S. K. A new algorithm for image edge extraction using a statistical classifier approach. *IEEE Trans. on Pattern Analysis and Machine Intelligence 9*, 4 (1987), 569–577.

[72] MACLEOD, R. *Percutaneous Transluminal Coronary Angioplasty as a Model of Cardiac Ischemia: Clinical and Modelling Studies.* PhD thesis, Dalhousie University, Halifax, N.S., Canada, 1990.

[73] MACLEOD, R., BROOKS, D., ON, H., KRIM, H., LUX, R., AND KORNREICH, F. Analysis of PTCA-induced ischemia using both an electrocardiographic inverse solution and the wavelet transform. *J. Electrocardiol. 27*, Suppl (1994), 90–96.

[74] MACLEOD, R., JOHNSON, C., AND ERSHLER, P. Construction of a human torso model from magnetic resonance images for problems in computational electrocardiography. Tech. Rep. UUCS-94-017, University of Utah, Department of Computer Science, 1994.

[75] MACLEOD, R., JOHNSON, C., GARDNER, M., AND HORÁČEK, B. Localization of ischemia during coronary angioplasty using body surface potential

mapping and an electrocardiographic inverse solution. In *IEEE Computers in Cardiology* (1992), IEEE Press, pp. 251–254.

[76] MacLeod, R., Johnson, C., and Matheson, M. Visualization tools for computational electrocardiography. In *Visualization in Biomedical Computing* (Bellingham, Wash., 1992), Proceedings of the SPIE #1808, pp. 433–444.

[77] Martin Hanke, K., and Per Christian Hansen, L. Regularization methods for large-scale problems. *Survey on Mathematics in Industry* (1993).

[78] McCarley, R., Shenton, M., O'Donnell, B., Faux, S., Kikinis, R., Nestor, P., and Jolesz, F. Auditory P300 abnormalities and left posterior superior temporal gyrus volume reduction in schizophrenia. *Arch. Gen. Psychiatry 87* (1993), 175–184.

[79] Messinger-Rapport, B., and Rudy, Y. Regularization of the inverse problem in electrocardiography: A model study. *Math. Biosci. 89* (1988), 79–118.

[80] Mohr, M., and Rüde, U. Multilevel Techniques for the Solution of the Inverse Problem of Electrocardiography.

[81] Montague, T., Smith, E., Cameron, D., Rautaharju, P., Klassen, G., Flemington, C., and Horáček, B. Isointegral analysis of body surface maps: Surface distribution and temporal variability in normal subjects. *Circ. 63* (1981), 1167–1172.

[82] Mosher, J., and Leahy, R. Source localization using recursively applied and projected (rap) music. *IEEE Transactions on Signal Processing 47*, 2 (1999).

[83] Notay, Y. Using approximate inverses in algebraic multilevel methods. *Numerische Mathematik 80*, 3 (1998), 397–417.

[84] O. Portniaguine, D. W., and Johnson, C. Focusing inversion of electroencephalography and magnetoencephalography data. In *3rd International Symposium On Noninvasive Functional Source Imaging* (2001).

[85] Oden, J., Demkowicz, L., Rachowicz, W., and Hardy, O. Toward a universal h-p adaptive finite element strategy, Part 2. A posteriori error estimation. *Computational Methods in Applied Mechanical Engineering 77* (1989), 113.

[86] Oostendorp, T., and van Oosterom, A. Source parameter estimation in inhomogeneous volume conductors of arbitrary shape. *IEEE Trans Biomed. Eng. 36* (1989), 382–391.

[87] Oostendorp, T., and van Oosterom, A. The potential distribution generated by surface electrodes in inhomogeneous volume conductors of arbitrary shape. *IEEE Trans Biomed. Eng. 38* (1991), 409–417.

[88] Oster, H., and Rudy, Y. The use of temporal information in the regularization of the inverse problem of electrocardiography. *IEEE Trans Biomed. Eng. 39*, 1 (1992), 65–75.

[89] P.C. Hansen, M. H. Regularization methods for large-scale problems. *Surv. Math. Ind.*, 3 (1993), 253–315.

[90] Penney, C., Clements, J., Gardner, M., Sterns, L., and Horáček, B. The inverse problem of electrocardiography: Application to localization of Wolff-Parkinson-White pre-excitation sites. In *Proceedings of the IEEE Engineering in Medicine and Biology Society 17th Annual International Conference* (1995), IEEE Press, pp. 215–216.

[91] Plonsey, R. Reciprocity applied to volume conductors and the ecg. *IEEE Trans. Biomed Electron 10*, 1 (1963), 9–12.

[92] Plonsey, R., and Fleming, D. *Bioelectric Phenomena.* McGraw-Hill Book Company, New York, 1969.

[93] Potts, G., Weinstein, D., O'Donnell, B., Shenton, M., Johnson, C., and McCarley, R. Bioelectric modeling of the P300 in schizophrenia. *Biological Psychiatry (supl.)* (1998), 396.

[94] Potts, G., Wible, C., Shenton, M., Weinstein, D., Fisher, I., Levention, M., Gugino, L., and McCarley, R. Localization of visual cortex with coregistered functional magnetic rersonance imaging, bioelectrically modeled cortical visual evoked potential, and transcranial magnetic stimulation induced visual suppression. *Cognitive Neuroscience (supl.)* (1998), 42.

[95] Rachowicz, W., Oden, J., and Demkowicz, L. Toward a universal h-p adaptive finite element strategy, Part 3. Design of h-p meshes. *Computaitonal Methods in Applied Mechanical Engineering 77* (1989), 181.

[96] Robert Bridson, W.-P. T. Multiresolution approximate inverse preconditioner. *SIAM Journal on Scientific Computing 23*, 2 (2001), 463–479.

[97] Roozen, H., and van Oosterom, A. Computing the activation sequence at the ventricular heart surface from body potentials. *Med. Biol. Eng. Comput. 25* (1987), 250–260.

[98] Rudy, Y., and Messinger-Rapport, B. The inverse solution in electrocardiography: Solutions in terms of epicardial potentials. *Crit. Rev. Biomed. Eng. 16* (1988), 215–268.

[99] Rudy, Y., and Oster, H. The electrocardiographic inverse problem. *Crit. Rev. Biomed. Eng. 20* (1992), 22–45.

[100] Rudy, Y., and Oster, H. The electrocardiographic inverse solution. In *High-Performance Computing in Biomedical Research*, T. Pilkington, L. B., J. Thompson, S. Woo, T. C. Palmer, and T. Budinger, Eds. CRC Press, 1993, ch. 6, pp. 135–155.

[101] Schaffer, S. A semicoarsening multigrid method for elliptic partial differential equations with highly discontinuous and anisotropic coefficients. *Siam Journal on Scientific Computing 20*, 1 (1999), 228–242.

[102] Scher, A., and Corbin, L. Present state of the electrocardiographic forward problem. *Adv. Cardiol. 21* (1978), 2–7.

[103] Schmidt, J., Johnson, C., Eason, J., and MacLeod, R. Applications of automatic mesh generation and adaptive methods in computational medicine. In *Modeling, Mesh Generation, and Adaptive Methods for Partial Differential Equations*, I. Babuska, J. Flaherty, W. Henshaw, J. Hopcroft, J. Oliger, and T. Tezduyar, Eds. Springer-Verlag, 1995, pp. 367–390.

[104] Schmidt, R. *A signal subspace approach to multiple emitter location and spectral estimation.* PhD thesis, Stanford University, 1981.

[105] Schmidt, R. Multiple emitter location and signal parameter estimation. *IEEE Trans. Antennas Propagat. AP-34* (1986), 276–280.

[106] Shahidi, A., Savard, P., and Nadeau, R. Forward and inverse problems of electrocardiography: Modeling and recovery of epicardial potentials in humans. *IEEE Trans Biomed. Eng. 41*, 3 (1994), 249–256.

[107] Sharon, E., Brandt, A., and Basri, R. Amg-based fast multiscale image segmentation. In *Copper Mountain Conference on Multigrid Methods* (1999).

[108] Shen, H., and Johnson, C. Semi-automatic image segmentation: A bimodel thresholding approach. Tech. rep., UUCS-94-019, Dept. of Computer Science, Univ. of Utah, 1994.

[109] Shephard, M., Guerinoni, F., Flaherty, J., Ludwig, R., and Baehmann, P. Adaptive solutions of the Euler equations using finite quadtree and octree grids. *Computers and Structures 30* (1988), 327–336.

[110] Shephard, M., and Yerri, M. An approach to automatic finite element mesh generation. *Comp. in Eng. 3* (1982), 21–28.

[111] Sonka, M., Hlavac, V., and Boyle, R. *Image Processing, Analysis and Machine Vision.* Chapman and Hall, 1993.

[112] Stytz, M., Frieder, G., and Frieder, O. Three-dimensional medical imaging: Algorithms and computer systems. *ACM Trans. Comp. Surveys. 23*, 4 (1991), 421–499.

[113] SWIHART, J. Numerical methods for solving the forward problem in electrocardiology. In *The Theoretical Basis of Electrocardiology*, C. Nelson and D. Geselowitz, Eds. Claredon Press, Oxford, 1976, pp. 257–293.

[114] TARKKA, I., STOKIC, D., BASILE, L., AND PAPANICOULAOU, A. Electric source localization of the auditory P300 agrees with magnetic source localization. *Electroencephalography and clinical Neurophysiology 96* (1995), 538–545.

[115] TATEBE, O. The multigrid preconditioned conjugate gradient method. In *Sixth Copper Mountain Conference on Multigrid Methods* (Hampton, VA, 1993), N. D. Melson, T. A. Manteuffel, and S. F. McCormick, Eds., vol. CP 3224, NASA, pp. 621–634.

[116] THOMPSON, J., AND WEATHERILL, N. Structed and unstructed grid generation. In *High-Performance Computing in Biomedical Research*, T. Pilkington, B. Loftis, J. Thompson, S.-Y. Woo, T. Palmer, and T. Budinger, Eds. CRC Press, Boca Raton, 1993, pp. 63–112.

[117] TIKHONOV, A., AND ARSENIN, V. *Solution of Ill-posed Problems*. Winston, Washington, DC, 1977.

[118] VAN DER SLUIS, A., AND VAN DER VORST, H. The rate of convergence of conjugate gradients. *Numer. Math.* (1986).

[119] VAN OOSTEROM, A., AND HUISKAMP, G. The effect of torso inhomogeneities on body surface potentials quantified using "tailored" geometry. *J. Electrocardiol. 22* (1989), 53–72.

[120] W. TANG, W. W. Sparse approximate inverse smoother for multigrid. *Siam Journal on Scientific Computing 21*, 4 (2000), 1236–1252.

[121] WALKER, S., AND KILPATRICK, D. Forward and inverse electrocardiographic calculations using resistor network models of the human torso. *Circ. Res. 61* (1987), 504–513.

[122] WEINSTEIN, D., ZHUKOV, L., AND JOHNSON, C. Lead-field bases for EEG source imaging. *Annals of Biomedical Engineering 28* (2000), 1–7.

[123] WESSELING, P. *An Introduction to Multigrid Methods*. John Wiley & Sons, Chichester, 1992. Reprinted by www.MGNet.org.

[124] YAMASHITA, Y. Theoretical studies on the inverse problem in electrocardiography and the uniqueness of the solution. *IEEE Trans Biomed. Eng. 29* (1982), 719–725.

[125] ZAMPERONI, P. Analysis of some region growing operators for image segmentation. In *Advances in Image processing and Pattern Recognition*, V. Cappelini and R. Marconi, Eds. North Holland, Amsterdam, 1986, pp. 204–208.