2.4.3. EIT Reconstruction Algorithm: Theory

For a two-dimensional system (i.e. with infinitesimal strip electrodes), the field equation and boundary condition for the voltage field $V^{(k)}(\mathbf{x})$ are as follows:

$$\nabla \cdot [\sigma \nabla V^{(k)}] = \delta(x)\delta(y) \text{ in the domain,} \tag{25}$$

$$\mathbf{n} \cdot \sigma \nabla V^{(k)} = \delta(s - s^{(k)}) \text{ on the boundary,} \tag{26}$$

where s is the arc length along the boundary, $s^{(k)}$ is the location of electrode k, δ is the Dirac delta function, and the reference point is placed at the origin without loss of generality. The delta-function terms in the field equation and the boundary condition represent current flow out of the domain at the reference point and into the domain from electrode k, respectively. For a three-dimensional system (i.e. with infinitesimal point electrodes), the single delta function in the boundary condition is replaced by the product of two delta functions in the coordinates needed to define the two-dimensional boundary surface, and the product of two delta functions in the field equation is replaced by a product of three delta functions in all three coordinates.

A Galerkin finite-element method (FEM) is used to discretize the voltage field:

$$V^{(k)}(\mathbf{x}) = \sum_{j=0}^{N_n} \phi_j(\mathbf{x}) V_j^{(k)}, \tag{27}$$

where $V_i^{(k)}$ is the voltage at node i and $\phi_i(\mathbf{x})$ is the (global) basis function for node i. The standard FEM procedure is applied to generate equations for the nodal values of voltage: multiply the above equation by $\phi_i(\mathbf{x})$, integrate over the entire domain, apply integration by parts, and insert the boundary condition to specify the flux on the boundary. As indicated above, one voltage must be specified to generate a unique solution. The voltage at the reference point, denoted node 0, is set equal to zero. The voltage equation results from this approach, where M_{ij} is the global stiffness matrix and δ_{ij} is the Kronecker delta:

$$\sum_{j=1}^{N_n} M_{ij} V_j^{(k)} = \delta_{ik}, \; M_{ij} = \int \sigma \nabla \phi_i \cdot \nabla \phi_j dx dy, \text{ for } i = 1, \ldots, N_n. \tag{28}$$

Note that the voltage equation is not applied at the reference node $i = 0$.

The conductivity field in the voltage field above is represented as a function of position:

$$\sigma = \sigma(\mathbf{x}; \{C_\alpha\}), \text{ with } \alpha = 1, \ldots, N_\sigma, \tag{29}$$

where the C_α are N_σ independent adjustable parameters. This representation can be fairly arbitrary so long as $0 < \sigma < \infty$. Although the conductivity functions subsequently used are global and typically smooth, these restrictions are not required in this development. The voltages $V_i^{(k)}$ depend on the conductivity parameters C_α in a complex way. This dependence can be simplified considerably if the voltages $V_i^{(k)}$ are linearized about the particular values $\{C_\alpha\}$. This necessitates computation of the Jacobians $\partial V_i^{(k)} / \partial C_\alpha$. The Jacobian equations are calculated directly from the voltage equation using the chain rule, where the $M_{\alpha ij}$ are the derivatives of the global stiffness matrix:

$$\sum_{j=1}^{N_n} M_{ij}\left(\frac{\partial V_j^{(k)}}{\partial C_\alpha}\right) = -\sum_{j=1}^{N_n} M_{\alpha ij} V_j^{(k)},\ M_{\alpha ij} = \int\left(\frac{\partial \sigma}{\partial C_\alpha}\right)\nabla\phi_i \cdot \nabla\phi_j dxdy,\ \text{for} \begin{cases} i = 1, \ldots, N_n \\ \alpha = 1, \ldots, N_\sigma \end{cases}. \tag{30}$$

Note that the global stiffness matrix appears on the left-hand sides of the voltage equation and all of the Jacobian equations (i.e. for all values of α). This affords considerable numerical economy since only one LU (lower triangular, upper triangular) decomposition of the stiffness matrix has to be performed to solve both the voltage equation and the Jacobian equation.

The reconstructed conductivity field is characterized by C_α values minimizing the rms difference between the computational and experimental EIT voltages excluding voltages from current-carrying electrodes. This necessitates adjusting the C_α to minimize the quantity

$$\Gamma = \frac{1}{2}\sum_{k=1}^{N_e}\sum_{(mn)} w_k^{(mn)}[V_k^{(mn)} + V_0^{(mn)} - v_k^{(mn)}]^2, \text{ with } V_k^{(mn)} = V_k^{(m)} - V_k^{(n)}, \tag{31}$$

where the $v_k^{(mn)}$ are the experimental EIT electrode voltages at electrode k for source electrode m and sink electrode n. The weights $w_k^{(mn)}$ are chosen to exclude current-carrying electrodes:

$$w_k^{(mn)} = \begin{cases} 0 \text{ if } k = m \text{ or } k = n \\ 1 \text{ otherwise} \end{cases}. \tag{32}$$

Note that arbitrary voltage offsets $V_0^{(mn)}$ have been included in this relation. It might be argued that there should be only N_e arbitrary voltage offsets $V_0^{(k)}$ appearing as the combination $[V_0^{(m)} - V_0^{(n)}]$ in the above equation rather than the $N_e(N_e-1)/2$ voltage offsets $V_0^{(mn)}$ that have been used. Under ideal circumstances, this would be correct. However, electronic and experimental complexities suggest that the larger number of voltage offsets is more appropriate.

Minimization of Γ is achieved by solving the following system for the C_α and the $V_0^{(mn)}$:

$$\frac{\partial \Gamma}{\partial C_\alpha} = 0, \text{ for } \alpha = 1, \ldots, N_\sigma; \frac{\partial \Gamma}{\partial V_0^{(mn)}} = 0, \text{ for all } (mn). \tag{33}$$

This minimization results in the following system of nonlinear equations for the C_α:

$$0 = \sum_{k=1}^{N_\sigma}\sum_{p=1}^{N_\sigma}\sum_{(mn)} w_k^{(mn)} w_p^{(mn)}[(V_k^{(mn)} - V_p^{(mn)}) - (v_k^{(mn)} - v_p^{(mn)})]\left(\frac{\partial V_k^{(mn)}}{\partial C_\alpha}\right), \tag{34}$$

where the $V_k^{(mn)}$ and the $V_p^{(mn)}$ are functions of the C_α.

Since these equations in this system are nonlinear, a Newton-Raphson (NR) technique is used to solve them iteratively. The voltages $V_k^{(mn)}$ and $V_p^{(mn)}$ (but not their derivatives with respect to the C_α) are linearized about a particular set of values for the C_α, where the departures c_α from these values are small. The departures c_α are found to satisfy the following system of equations:

$$\sum_{\beta=1}^{N_\sigma} A_{\alpha\beta} c_\beta = B_\alpha, \tag{35}$$

where the matrix $A_{\alpha\beta}$ and the vector B_α are given by the relations

$$A_{\alpha\beta} = \sum_{k=1}^{N_\sigma} \sum_{p=1(mn)}^{N_\sigma} \sum w_k^{(mn)} w_p^{(mn)} \left\{ \frac{\partial V_k^{(mn)}}{\partial C_\alpha} \right\} \left\{ \frac{\partial V_k^{(mn)}}{\partial C_\beta} - \frac{\partial V_p^{(mn)}}{\partial C_\beta} \right\}$$

$$= \frac{1}{2} \sum_{k=1}^{N_\sigma} \sum_{p=1(mn)}^{N_\sigma} \sum w_k^{(mn)} w_p^{(mn)} \left\{ \frac{\partial V_k^{(mn)}}{\partial C_\alpha} - \frac{\partial V_p^{(mn)}}{\partial C_\alpha} \right\} \left\{ \frac{\partial V_k^{(mn)}}{\partial C_\beta} - \frac{\partial V_p^{(mn)}}{\partial C_\beta} \right\}, \tag{36}$$

$$B_\alpha = \sum_{k=1}^{N_\sigma} \sum_{p=1(mn)}^{N_\sigma} \sum w_k^{(mn)} w_p^{(mn)} \left\{ \frac{\partial V_k^{(mn)}}{\partial C_\alpha} \right\} \{ [v_k^{(mn)} - v_p^{(mn)}] - [V_k^{(mn)} - V_p^{(mn)}] \}$$

$$= \frac{1}{2} \sum_{k=1}^{N_\sigma} \sum_{p=1(mn)}^{N_\sigma} \sum w_k^{(mn)} w_p^{(mn)} \left\{ \frac{\partial V_k^{(mn)}}{\partial C_\alpha} - \frac{\partial V_p^{(mn)}}{\partial C_\alpha} \right\} \{ [v_k^{(mn)} - v_p^{(mn)}] - [V_k^{(mn)} - V_p^{(mn)}] \}. \tag{37}$$

The c_α values obtained in this manner are used to update the conductivity parameters according to $C_\alpha \to C_\alpha + c_\alpha$. Since linearized voltages were used to determine these values, the resulting C_α values will not minimize Γ exactly. However, repetitions of this Newton-Raphson process ultimately converge to the C_α values that minimize Γ.

The resulting algorithm starts from an initial guess for the C_α and proceeds as follows:

1. Compute the M_{ij} and $M_{\alpha ij}$ matrices:

$$M_{ij} = \int \sigma \nabla \phi_i \cdot \nabla \phi_j dx dy, \; M_{\alpha ij} = \int \left(\frac{\partial \sigma}{\partial C_\alpha} \right) \nabla \phi_i \cdot \nabla \phi_j dx dy. \tag{38}$$

2. Perform the LU decomposition of M_{ij}.

3. Solve for the voltages $V_i^{(k)}$ and the Jacobians $(\partial V_i^{(k)} / \partial C_\alpha)$:

$$\sum_{j=1}^{N_n} M_{ij} V_j^{(k)} = \delta_{ik}, \; \sum_{j=1}^{N_n} M_{ij} \left(\frac{\partial V_j^{(k)}}{\partial C_\alpha} \right) = - \sum_{j=1}^{N_n} M_{\alpha ij} V_j^{(k)}. \tag{39}$$

4. Compute the matrix $A_{\alpha\beta}$ and the vector B_α:

$$A_{\alpha\beta} = \sum_{k=1}^{N_\sigma} \sum_{p=1(mn)}^{N_\sigma} \sum w_k^{(mn)} w_p^{(mn)} \left\{ \frac{\partial V_k^{(mn)}}{\partial C_\alpha} \right\} \left\{ \frac{\partial V_k^{(mn)}}{\partial C_\beta} - \frac{\partial V_p^{(mn)}}{\partial C_\beta} \right\},$$

$$B_\alpha = \sum_{k=1}^{N_\sigma} \sum_{p=1(mn)}^{N_\sigma} \sum w_k^{(mn)} w_p^{(mn)} \left\{ \frac{\partial V_k^{(mn)}}{\partial C_\alpha} \right\} \{ [v_k^{(mn)} - v_p^{(mn)}] - [V_k^{(mn)} - V_p^{(mn)}] \}. \tag{40}$$

5. Solve for the c_α:

$$\sum_{\beta=1}^{N_\sigma} A_{\alpha\beta} c_\beta = B_\alpha. \tag{41}$$

6. Update $C_\alpha \to C_\alpha + c_\alpha$ and test for convergence.

One special choice of the functional dependence of σ on the C_α is of particular importance because it reduces by one the number of Jacobian matrices that must be computed:

$$\sigma = \frac{\sigma_1(\mathbf{x};\{C_\alpha\})}{C_1}, \text{ with } \alpha = 2, \ldots, N_\sigma, \tag{42}$$

where $0 < C_1 < \infty$. From the voltage equation, it is clear that all voltages are proportional to C_1:

$$V_k^{(mn)} = C_1 V_{k,1}^{(mn)}, \tag{43}$$

where the voltages $V_{k,1}^{(mn)}$ are computed for $C_1 = 1$ and are functions of the C_α for $\alpha = 2, \ldots, N_\sigma$. Minimization results in the following system of (slightly less) nonlinear equations for the C_α:

$$0 = \sum_{k=1}^{N_\sigma} \sum_{p=1(mn)}^{N_\sigma} \sum w_k^{(mn)} w_p^{(mn)} [C_1 (V_{k,1}^{(mn)} - V_{p,1}^{(mn)}) - (v_k^{(mn)} - v_p^{(mn)})] \left\{ \begin{array}{ll} V_{k,1}^{(mn)} & \text{for } \alpha = 1 \\ \dfrac{\partial V_{k,1}^{(mn)}}{\partial C_\alpha} & \text{for } \alpha = 2, \ldots, N_\sigma \end{array} \right\}, \tag{44}$$

Linearization of the voltages about particular values of the C_α for $\alpha = 2, \ldots, N_\sigma$ yields the following linear system for the small departures c_α:

$$\sum_{\beta=1}^{N_\sigma} \tilde{A}_{\alpha\beta} \left\{ \begin{array}{ll} C_1 & \text{for } \beta = 1 \\ C_1 c_\beta & \text{for } \beta = 2, \ldots, N_\sigma \end{array} \right\} = \tilde{B}_\alpha, \tag{45}$$

where the matrix $\tilde{A}_{\alpha\beta}$ and the vector $\tilde{B}_\alpha$ are given by the relations

$$\tilde{A}_{\alpha\beta} = \sum_{k=1}^{N_\sigma} \sum_{p=1(mn)}^{N_\sigma} \sum w_k^{(mn)} w_p^{(mn)} \left\{ \begin{array}{ll} V_{k,1}^{(mn)} & \text{for } \alpha = 1 \\ \dfrac{\partial V_{k,1}^{(mn)}}{\partial C_\alpha} & \text{for } \alpha = 2, \ldots, N_\sigma \end{array} \right\} \left\{ \begin{array}{ll} V_{k,1}^{(mn)} - V_{p,1}^{(mn)} & \text{for } \beta = 1 \\ \dfrac{\partial V_{k,1}^{(mn)}}{\partial C_\beta} - \dfrac{\partial V_{p,1}^{(mn)}}{\partial C_\beta} & \text{for } \beta = 2, \ldots, N_\sigma \end{array} \right\}, \tag{46}$$

$$\tilde{B}_\alpha = \sum_{k=1}^{N_\sigma} \sum_{p=1(mn)}^{N_\sigma} \sum w_k^{(mn)} w_p^{(mn)} \left\{ \begin{array}{ll} V_{k,1}^{(mn)} & \text{for } \alpha = 1 \\ \dfrac{\partial V_{k,1}^{(mn)}}{\partial C_\alpha} & \text{for } \alpha = 2, \ldots, N_\sigma \end{array} \right\} \{[v_k^{(mn)} - v_p^{(mn)}]\}. \tag{47}$$

When this particular approach is used, the quantities C_1 and $C_1 c_\alpha$ for $\alpha = 2, \ldots, N_\sigma$ are the unknowns when solving the linear system. From these quantities, it is straightforward to determine the c_α for $\alpha = 2, \ldots, N_\sigma$. The c_α values obtained in this manner are used to update the conductivity parameters according to $C_\alpha \rightarrow C_\alpha + c_\alpha$ for $\alpha = 2, \ldots, N_\sigma$. Of course, the updated value of C_1 is determined directly. As previously indicated, since linearized voltages were used to determine these values, the resulting C_α values will not minimize Γ exactly. However, repetitions of this Newton-Raphson process ultimately converge to the C_α values that minimize Γ.

2.4.4. EIT Reconstruction Algorithm: FEMEIT Implementation

A computer code called FEMEIT has been written to implement the approach discussed in the previous section. Although intended for two-dimensional problems, FEMEIT was written so that development of a three-dimensional version would be straightforward. Although to date it has been applied only to circular domains, FEMEIT can handle general two-dimensional domains, including multiply connected domains and domains with internal electrodes, so long as electrode widths are small compared to inter-electrode distances. Since electrodes are represented as mathematical points, a node must be placed at each electrode. Currently FEMEIT uses 3-node linear triangle elements to assemble the M_{ij} and $M_{\alpha ij}$ matrices (a three-dimensional extension would probably use 4-node tetrahedrons). These elements, although of low order, were chosen for several reasons. The quantities $\nabla\phi_i \cdot \nabla\phi_j$ are constant within an element, which is both straightforward to implement and allows this quantity to be brought out from inside the integrals. Thus, the integration can focus on integrating the conductivity function over the elements, independent of the values of $\nabla\phi_i \cdot \nabla\phi_j$. Also, circular domains are easily filled using triangles. The conductivity function is chosen to be a global function and is currently selected from a library of choices in a subroutine. At present, Cartesian polynomials, radial polynomials, products of radial polynomials and sines and cosines, and a 5-parameter quadrant interpolation function are available. Incorporation of additional functional forms into this subroutine is straightforward. The integration of the conductivity function is performed using a 4-point quadrature over each triangular element. FEMEIT uses SLATEC subroutines for the solution of the linear systems and has been compiled on both Sun and IBM machines. A 15-parameter reconstruction of 16-electrode data using a mesh with 441 nodes requires somewhat under 2 minutes on the IBM.

FEMEIT requires four input data files. Examples of each are shown in Appendix A.

1. `nodelm.dat` This file contains the mesh information.
   (a) The first line has the number of nodes.
   (b) Subsequent lines have node number, node x, node y (1 node per line).
   Important: the last node must be an internal non-electrode node and will have zero voltage.
   (c) The first line after the nodes has the number of elements.
   (d) Subsequent lines have element number and node numbers (1 element per line).
2. `exinfo.dat` This file contains the electrode nodal locations.
   (a) The first line has the number of electrodes N_e.
   (b) Subsequent lines have the electrode number and the node number (1 electrode per line).
3. `exdata.dat` This file contains the experimental data. If there are N_e electrodes, then results for $N_e(N_e-1)/2$ experiments are expected.
   (a) The first line for each experiment has the current-injecting electrode number, the current-withdrawing electrode number, and the current per unit length (in A/m).
   (b) Subsequent lines for each experiment have the electrode voltages (in V) given in order from electrode 1 to electrode N_e.
4. `conpar.dat` This file contains conductivity function and parameter information and convergence requirements.
   (a) The first line has the following six variables:

The maximum relative change in nodal conductivity per iteration.

Damping factor lower and upper bounds between 0 and 1. The relative change in the conductivity parameters is bounded by these parameters.

A tolerance on the relative change in the conductivity parameters. Iterations continue until the relative change falls below this value.

A tolerance on the relative degree to which the equations are satisfied. Iterations continue until the equations are satisfied to better than this value.

A geometric length scale used to normalize all coordinates, including those used within the selected model for σ.

(b) The second line has the following variables:

The maximum number of iterations.

An integer denoting the model for spatial variation of σ.

The number of conductivity (fitting) parameters used with this model.

The number of internal (nonvarying) parameters used with this model.

(c) The following lines contain:

Starting guesses for all the conductivity parameters.

Values for all the internal parameters.

FEMEIT performs some error checking to make sure that the correct number of parameters have been entered for the model selected. Five models have been implemented:

1. Sums of polynomials in x and y: $\sigma = \Sigma_\beta x^{m_\beta} y^{n_\beta} C_\beta$ with conductivity parameters $\{C_\beta\}$ and internal parameters $\{m_\beta\}$ and $\{n_\beta\}$ (integers). There are exactly twice as many internal parameters as conductivity parameters.

2. Sums of products of powers of r with functions of θ: $\sigma = \Sigma_\beta r^{m_\beta} \mathrm{trig}(|n_\beta|\theta) C_\beta$ with conductivity parameters $\{C_\beta\}$ and internal parameters $\{m_\beta\}$ and $\{n_\beta\}$ (integers), where trig = cos if $n_\beta < 0$, trig = sin if $n_\beta > 0$, and trig = 1 if $n_\beta = 0$. There are exactly twice as many internal as conductivity parameters.

3. An axisymmetric piecewise linear interpolation in radius with $\sigma = C_\beta$ at $r = R_\beta$. There are equal numbers of conductivity and internal parameters. The radii R_β are supplied in ascending order. The last one must be the radius of the outer circle. The first one cannot be 0. The conductivity is constant at C_1 within the first radius.

4. A "bubble" centered at the origin with conductivity parameters C_1, C_2 and internal parameters P_1, P_2 such that $C_1 > 0$, $C_2 > 0$, $0 < P_1 < 1$, and $P_2 > 0$:

$$\sigma = C_1 \left\{ 1 + \frac{P_1}{2} \left[\tanh\left(\frac{r - C_2}{P_2} \right) - \tanh\left(\frac{r + C_2}{P_2} \right) \right] \right\}. \tag{48}$$

5. A "bubble" centered at (C_3, C_4): as above but having $r = \sqrt{(x-C_3)^2+(y-C_4)^2}$ with conductivity parameters C_1, C_2, C_3, C_4 and internal parameters P_1, P_2 such that $C_1 > 0$, $C_2 > 0$, $0 < P_1 < 1$, and $P_2 > 0$:

FEMEIT produces three output data files. Examples of each are shown in Appendix A.

1. `parcon.dat` This file contains the values of the conductivity parameters at the end of the last iteration. Its format is essentially identical to `conpar.dat`, and it can be copied directly to `conpar.dat` if a restart is desired. It also shows the change in each parameter during the last iteration.

2. `nodcon.dat` This file contains the conductivity values at each node. Each line contains the node number, the conductivity at that node and the change in the conductivity at that node during the last iteration.

3. `iter.log` This file contains the iteration history (also written to the screen).

Several other smaller codes have been written to assist in validating and using FEMEIT.

1. `postfd` This code takes the FEMEIT output files and creates a FIDAP neutral file `postfd.fdneut`, which can be processed with `ficonv` and `fipost`.

2. `msheit` This code generates triangular element meshes for circles. It creates files `nemesh.dat` and `prinfo.dat`, to be copied to `nodelm.dat` and `exinfo.dat`. The origin is always the last node.

3. `ansoln` This code uses the analytical solution for the voltage field observed with a constant conductivity in a circular domain with a source at (x_0, y_0) and a sink at $(x_0, -y_0)$ to generate "numerical experimental data" in the file `andata.dat`, to be copied to `exdata.dat`.

$$V(x, y) = \left(\frac{I}{2\pi\sigma_0 R}\right)\ln\left[\frac{(y_0+y)^2+(x_0-x)^2}{(y_0-y)^2+(x_0-x)^2}\right]. \tag{49}$$

4. `fuldat` This code takes adjacent-electrode results from FIDAP simulations and uses linearity to create a full data set for all pairwise electrode combinations, which is contained in `fuldat.dat`, to be copied to `exdata.dat`.

FEMEIT is capable of reconstructing a conductivity field accurately so long as two things are true. First, the conductivity field must be capable of being represented by the conductivity function used by FEMEIT to describe it. Second, the FEM mesh used by FEMEIT must be adequate to represent both the solution and the conductivity function. Not only must there be sufficient nodes and elements, but also the elements in the vicinity of electrodes must have aspect ratios close to unity. The latter requirement arises from the singular behavior of the voltage near mathematical point electrodes, as discussed earlier. Failing this, excessive resolution may be needed near electrodes. Also, FEMEIT will not be able to return a conductivity field if it is allowed to use more parameters to represent the conductivity field than it has pieces of information (not data).

2.4.5. EIT Reconstruction Algorithm: FEMEIT Validation

Three approaches have been used to validate FEMEIT. The first involved comparison with the analytical solution for the voltage fields produced by infinitesimal strip electrodes in a circular domain when the conductivity is constant. The second involved a systematic mesh-refinement study of the reconstruction of a nontrivial conductivity field from voltages calculated by FIDAP using this conductivity field. The third involved comparison with a BEM code called 2DynaEIT developed by Dynaflow, Inc. (Duraiswami et al., 1995; O'Hern et al., 1995a) to determine the size and location of an insulating cylindrical inclusion (ICI) placed within a circular domain.

The first type of validation exercise involved using the analytical result for the voltage distribution with a constant conductivity σ_0 in a circular domain in which a current I/R per unit length is injected at the point (x_0, y_0) and withdrawn at the point $(x_0, -y_0)$:

$$V(x, y) = \left(\frac{I}{2\pi\sigma_0 R}\right)\ln\left[\frac{(y_0+y)^2+(x_0-x)^2}{(y_0-y)^2+(x_0-x)^2}\right]. \tag{50}$$

As in O'Hern et al. (1995), a 16-electrode EIT apparatus was considered. The codes `ansoln` and `fuldat` were used to generate the boundary voltages for all possible electrode pairs from the analytical solution. These boundary voltages, along with mesh information and specification of the conductivity function type, were the inputs to the FEMEIT code. For this study, the conductivity is taken to be an unknown constant (i.e. there was exactly one unknown parameter in the minimization of the rms voltage difference). In all cases examined, the conductivity value was correctly determined by FEMEIT when a moderate nodal density was employed (a few hundred nodes and elements).

The second type of validation exercise involved using the finite-element code FIDAP (Fluid Dynamics International, 1995) to compute the boundary voltages for a 16-electrode apparatus in the presence of a complicated electrical conductivity spatial distribution. The prescribed conductivity field is specified to be a linear combination of $\{1, x, y, x^2, xy, y^2, x^3, x^2y, xy^2, y^3, x^4, x^3y, x^2y^2, xy^3, y^4\}$ that goes to unity on the perimeter of the circular domain and dips to slightly below 0.4 in the upper right quadrant, as shown in the bottom right of Figure 24. This pattern was chosen to represent what might be observed if an excess of gas volume fraction were present in the upper right quadrant. FIDAP was used to solve the 16 adjacent-electrode conduction problems on the highly refined mesh of 9-node isoparametric quadrilaterals, shown in the bottom left of Figure 24, and `fuldat` was used to create a full EIT data set containing the results of all 120 distinct pairwise experiments. FEMEIT was used to analyze this numerically created data set. The conductivity field was represented by the function

$$\begin{aligned}\sigma = {} & C_1 + C_2x + C_3y + C_4x^2 + C_5xy + C_6y^2 \\ & + C_7x^3 + C_8x^2y + C_9xy^2 + C_{10}y^3 + C_{11}x^4 + C_{12}x^3y + C_{13}x^2y^2 + C_{14}xy^3 + C_{15}y^4,\end{aligned} \tag{51}$$

and 5 different meshes were examined, all of which are shown in Figure 24. Although mesh A, the most coarse mesh, did little more than indicate the rough amount of spatial variation in the conductivity field, mesh B, the next most coarse mesh, yielded a reasonable result. The most highly refined mesh employed here, mesh E, did an excellent job in reproducing the conductivity field. If the conductivity function $\sigma = C_1$ were used instead of the above conductivity function, FEMEIT with the most refined mesh yields a (uniform) conductivity of 0.603, which is a reasonable estimate of the average conductivity of the complicated conductivity field, $2/3$. Note that for this example, FIDAP has been used to generate a data set and to postprocess FEMEIT results but not to solve the FEM problems during the EIT reconstruction.

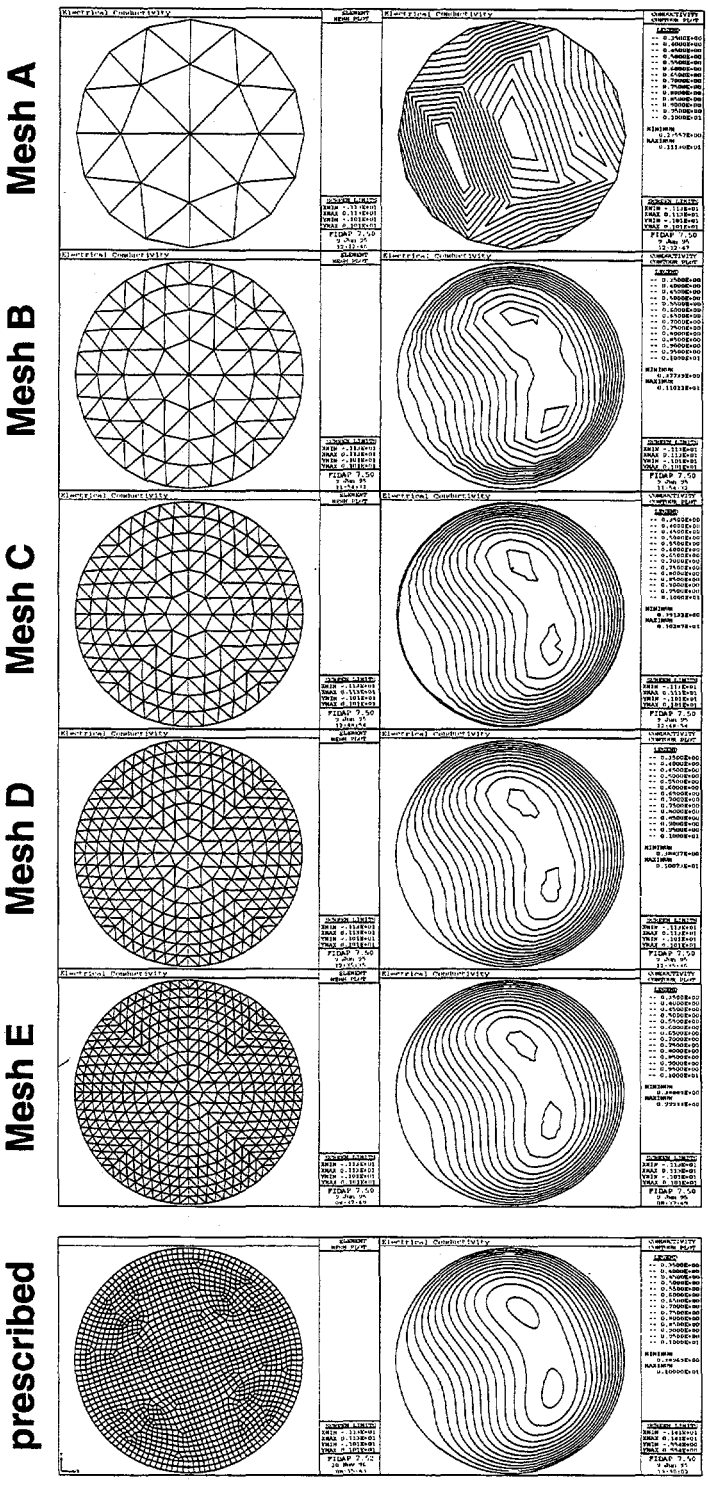


Figure 24. EIT mesh refinement studies. Good convergence is observed.

The third type of validation exercise involved using the codes FEMEIT and 2DynaEIT (Duraiswami et al., 1995; O'Hern et al., 1995a) to reconstruct the size and position of an ICI placed within a circular domain with constant conductivity. In this particular example, the domain was taken to have a radius of $R = 1$, within which the conductivity was prescribed to have the following spatial variation:

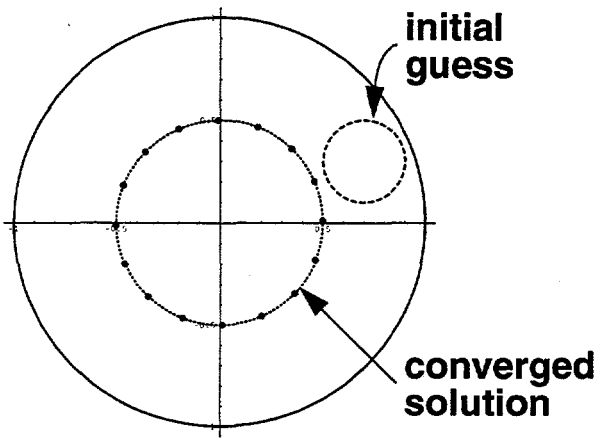
$$\sigma = \begin{cases} 0 \, , \, 0 \le r < 0.5 \\ 1 \, , \, 0.5 \le r \le 1 \end{cases} . \tag{52}$$

For both codes, numerical data sets were generated to provide the required input. FIDAP (Fluid Dynamics International, 1995) was used to generate the required voltage data for FEMEIT as previously discussed, and the validation code 2DynaEIT_fwd was used to generate the voltage data for 2DynaEIT. Unfortunately, the FIDAP data set could not be used by 2DynaEIT, which requires smooth boundary data at present, rather than voltages only at electrodes, as would be provided by an experiment. It is anticipated that this limitation will be removed in the future from 2DynaEIT.

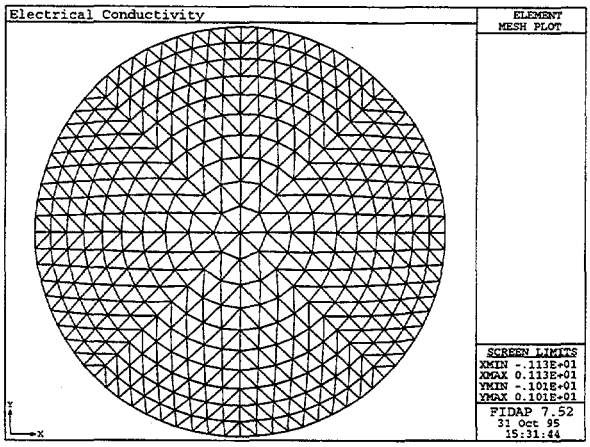
Both codes require the type of conductivity variation to be specified. FEMEIT was instructed to use model 5, a "bubble" of arbitrary radius and position with arbitrary conductivity outside the bubble. The width of the bubble boundary was set to $2P_2 = 0.1$, and the ratio of the conductivities inside and outside the bubble was set to $1 - P_1 = 0.02$. 2DynaEIT was instructed to look for a circular "bubble" of arbitrary radius and position, where the bubble here is a zero-thickness boundary separating a region of zero conductivity from a region of unity conductivity. In both cases, the bubble was specified to have an initial radius of 0.2 and an initial position of $(0.7, 0.3)$, and the conductivity was taken to be zero inside the bubble and unity outside the bubble.

Figure 25 shows the results from the FEMEIT and 2DynaEIT reconstructions. The mesh used by FEMEIT is shown and consists of 441 nodes and 800 linear triangle elements. Since 2DynaEIT is a BEM code, only the boundary was meshed. In this exercise, both the domain boundary and the bubble boundary had 16 nodes equidistantly placed. The final positions of the nodes on the bubble are shown in Figure 25. Although the initial guess was quite far from the actual conductivity field, both codes accurately reconstructed the actual field. Interestingly, the times required by the codes to converge were almost the same (a few minutes on a workstation).

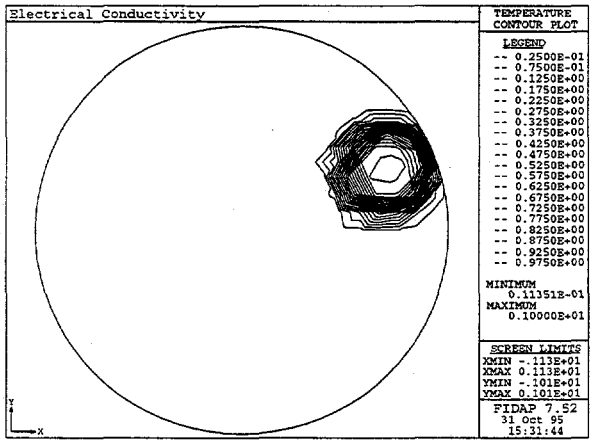
insulating cylindrical inclusion (ICI) reconstructions



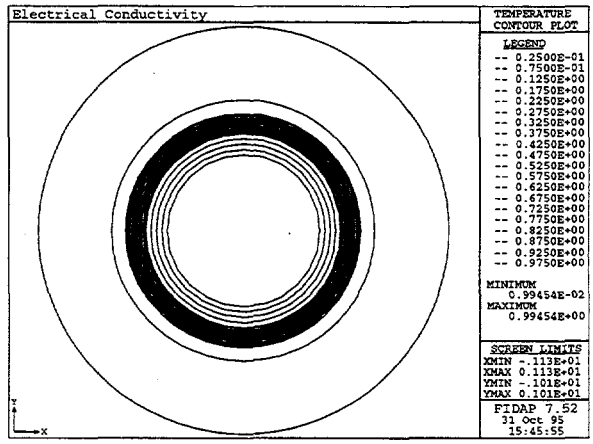
2DynaEIT



FEMEIT mesh



FEMEIT initial guess



FEMEIT converged solution

Figure 25. 2DynaEIT (Duraiswami et al., 1995) and FEMEIT ICI reconstructions.

2.4.6. EIT Reconstruction Algorithm: Extensions

As presently configured, FEMEIT is capable of utilizing two-dimensional experimentally generated data: electrodes must have small widths and long lengths (infinitesimal strip electrodes) compared to their separations. As previously discussed, the assumption of two-dimensionality is difficult to satisfy in practice. As a first step toward a three-dimensional code with capabilities similar to FEMEIT, another code called EITA3D has been written for infinitesimal point electrodes around an infinite cylindrical domain within which the conductivity field is parametrized according to $\sigma = \sigma_1(r;C_2)/C_1$. With this choice, no additional Jacobian information is required for the parameter C_1, but a Jacobian must still be computed for the parameter C_2. Since the voltages $V_{k,l}^{(m)}$ depend only on the parameter C_2 and the conductivity spatial variation is purely radial, it is practical to compute the voltages once for all (e.g. with FIDAP) for particular choices of the function σ_1 at many values of the parameter C_2. Two types of conductivity fields have been examined thus far:

$$\sigma_1 = \begin{cases} 0\ , 0 \le r/R < C_2 \\ 1\ , C_2 \le r/R \le 1 \end{cases} \text{ with } 0 \le C_2 < 1 \text{ (an ICI or "hole")}, \tag{53}$$

$$\sigma_1 = 1 + C_2[2(r/R)^2 - 1] \text{ with } -1 < C_2 \le 1 \text{ (a parabolic variation)}. \tag{54}$$

For each type of conductivity field, FIDAP was used to compute the $V_{k,l}^{(m)}$ at roughly 20 C_2 values spanning the allowed range. A variant of a standard cubic-spline algorithm (Press et al., 1986) is used to compute both the $V_{k,l}^{(m)}$ and the $\partial V_{k,l}^{(m)}/\partial C_2$ for arbitrary C_2 values within the allowed range. Thus, EITA3D performs no FEM calculations directly but rather uses the voltage values previously computed by FIDAP to determine the updates.

Figure 26 shows the results of applying EITA3D to a data set generated by FIDAP using a conductivity field having the "hole" type of variation with $C_1 = 1$ and $C_2 = 0.1$. Despite being provided with the rather poor initial guesses of $C_1 = 0.5$ and $C_2 = 0.8$, EITA3D is observed to converge smoothly to the correct values after only 9 iterations. Note, however, that this cannot be called a true validation exercise since FIDAP was used to create both the input data set and the FEM data used by EITA3D to determine C_1 and C_2.

As presently configured, 2DynaEIT is not yet capable of utilizing experimentally generated data: current injection is taken to be a smooth function of arc length around the outer boundary rather than at discrete electrodes as in the experiments, exactly one boundary node per electrode is strictly enforced, and the conductivity spatial variation consists of regions of zero conductivity embedded in a constant-conductivity background. All of these issues appear to be resolvable if given sufficient resources.

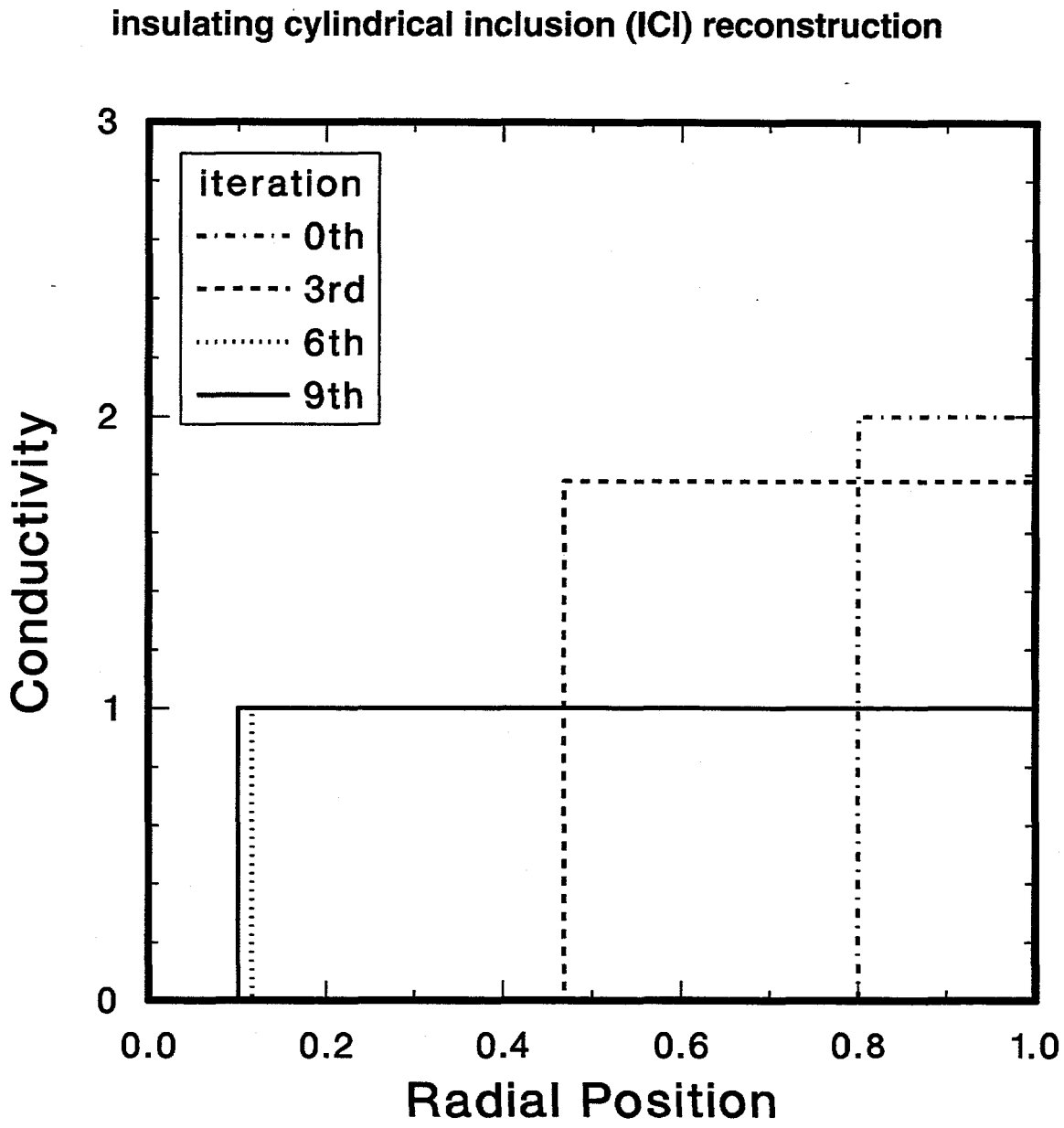


Figure 26. EIT determination of radius of an ICI using point electrodes and EITA3D.