

# 2.5D Finite Element Method for Electrical Impedance Tomography considering the Complete Electrode Model

Navid Bahrani<sup>1</sup>, Andy Adler<sup>2</sup>  
Systems and Computer Engineering  
Carleton University  
Ottawa, Canada

<sup>1</sup>nbahrani@sce.carleton.ca, <sup>2</sup>adler@sce.carleton.ca

**Abstract**— The 2.5D Finite Element Method is commonly used in geophysical applications where the goal is to solve a 3-dimensional problem by a set of 2-dimensional models. The key assumption in 2.5D Finite Element analysis is to approximate the medium to be translationally invariant along one of the coordinates. In this work, complimentary modules are developed to enhance the EIDORS [1] project by the 2.5D Finite Element technique based on the complete electrode model using the proper boundary condition. Furthermore, the efficiency of the method is discussed and the accuracy of the forward solution employing different number of 2-dimensional model is investigated.

**Keywords**- *Electrical Impedance Tomography, 2.5D Finite Element Analysis, Complete Electrode Model*

## I. INTRODUCTION

Electrical Impedance Tomography (EIT) is a soft-field tomographic technique in which the conductivity pattern, or generally the impedance pattern, of inside of a body is reconstructed by the means of measured boundary voltages produced from injected boundary currents. The body could be a human organ, human chest or a fluid pipe.

Compared to the CT or MRI scan, EIT offers poor imaging resolution. However, EIT is a radiation-free imaging modality and it is completely non-invasive as long as the drive current amplitude meets safety standards. Taking advantages of being a relatively low cost, fast imaging modality and portability besides its non-invasive feature make EIT a suitable technique for long term monitoring. EIT is also capable of providing impedance map of inside of human skull while ultrasonic techniques are unable to image it.

In EIT, a set of electrodes is installed around the boundary of a body, e.g. human chest. Due to better noise performance and stability issues, often, electrical current is injected and the electric potential is measured. Depending on the current stimulation pattern, current is injected from a pair of electrodes placed on the boundary of the body and difference voltage between all adjacent electrodes, also known as gradients, is measured. Then, the drive current is applied to the next pair of electrodes and again the difference voltage between all adjacent electrodes is measured. This procedure continues until all possible pairs of drive electrodes are covered. Two common stimulation patterns are ‘adjacent’ drive method, in which the

sink electrode is located adjacent to the source electrode and ‘opposite’ drive method, where the sink and source are located opposite to each other.

The EIT problem is a 3D problem due to the fact that the currents propagate in the 3D medium even if the electrodes are located in a plane. EIT contains a forward problem and an inverse problem. In the forward problem, the Laplace’s equation for a general 3D model, expressed in (1a), is solved inside the medium  $\Omega$  considering the boundary condition expressed in (1b).

$$\nabla_{3D} \cdot (\sigma(x, y, z) \nabla_{3D} \varphi(x, y, z)) = 0 \quad (1a)$$

$$\sigma(x, y, z) \frac{\partial}{\partial \hat{n}} \varphi(x, y, z) = J_s \quad (1b)$$

In equations (1),  $\nabla_{3D}$  is the gradient operator for the 3D coordinate system,  $\sigma$  is conductivity,  $\varphi$  is the potential,  $\hat{n}$  is the unit vector normal to the boundary and  $J_s$  is the boundary surface current density which is non-zero only under current driving electrodes and zero elsewhere.

Solving the 3D equation demands a 3D mesh containing a large number of nodes and a large number of elements which requires a large capacity of memory. In the forward EIT problem, a “system matrix” is constructed which is then inverted for solving the forward equation. The system matrix, or stiffness matrix, is a square matrix with dimensions equal to the number of nodes and expresses the relation between nodal voltages and currents. In the inverse EIT problem, a “sensitivity matrix” is constructed which is then inverted for solving the inverse equation. The sensitivity matrix expresses the relation between the boundary measurements and the conductivity of all elements having dimensions equal to number of measurements by number of elements.

Inversion and construction of these matrices require large capacities of memory and computation which compromise the portability of EIT; also, the run time required by the algorithm compromises the real time and fast imaging advantage of the EIT modality. In this article, the 2.5D finite element method is implemented for Electrical Impedance Tomography considering the Complete Electrode Model (CEM) [2] in order to solve a 3D problem by employing a 2D model and solving a few 2D-like equations. The 2.5D finite element method is briefly expressed in section II. The structure of the developed modules for 2.5D FEM is explained in section III. The

complete electrode modelling is explained in section IV. Section V contains discussions on solution accuracy, the sufficient number of equations needed to be solved given the accuracy, and time-memory performance.

## II. THE 2.5D FINITE ELEMENT METHOD

### A. The 2.5D Equation

The 2.5D problem requires the assumption of having translationally invariant conductivity along the  $z$ -axis in a 3D domain over  $-z_m < z < +z_m$ ; hence, the voltage distribution within the 3D domain would be symmetric with respect to the  $z = 0$  plane provided a symmetric boundary condition. By assuming translationally invariant conductivity and employing a 2-dimensional EIT model, the ‘‘off-slice currents’’ would not be considered [3]; hence some 3D considerations are required which results in the 2.5D method. Assuming symmetric voltage distribution, it is possible to calculate one dimensional cosine (Fourier) series over only the  $z$ -axis. The analysis and synthesis equations for this transform are:

$$V_n(x, y) = \int_{-z_m}^{+z_m} \varphi(x, y, z) \cos\left(\frac{n\pi}{z_m}z\right) dz \quad (2a)$$

$$\varphi(x, y, z) = \sum_{n=0}^{\infty} V_n(x, y) \cos\left(\frac{n\pi}{z_m}z\right) \quad (2b)$$

where  $\varphi(x, y, z)$  is the actual potential in the 3D domain  $\Omega$  which is transformed into an infinite number of 2D harmonic voltages,  $V_n(x, y)$ , each of which represents a spatial frequency,  $\frac{n\pi}{z_m}$ . Substituting the synthesis equation (2b) into the Laplace’s equation (1) and following up the derivations, demonstrated with details in [4], the following set of governing equations are obtained for the 2.5D method:

$$\nabla_{2D} \cdot (\sigma(x, y) \nabla_{2D} V_n(x, y)) - \sigma(x, y) \left(\frac{n\pi}{z_m}\right)^2 V_n(x, y) = 0 \quad (3)$$

$\forall n \in \{0, 1, 2, 3, \dots\}$

where  $\nabla_{2D}$  is the gradient operator for the 2D coordinate system. For the  $n = 0$  case, the set of equations (3) reduces to a classic 2D Laplace equation. However, in general, the boundary condition is not the same as the 2D case even for  $n = 0$ ; unless one assumes a unit height domain.

### B. The Boundary Condition

The boundary condition is an important issue in electromagnetic problems. Since the partial differential equation is same in almost all problems, the problem geometry and the boundary condition form the solution. Substituting synthesis equation (2b) into the boundary condition (1b) for the general 3D model would result in the 2.5D boundary condition expressed in (4), demonstrated in [2].

$$\sigma(x, y) \frac{\partial}{\partial \hat{n}} V_n(x, y) = J_n \quad (4)$$

where  $J_n$  are the coefficients of the cosine series expansion of the injected current density. Assuming a uniform distribution of current density under the electrodes of height  $h$ , the boundary condition is derived for each harmonic  $n$  as:

$$J_0 = \frac{h}{H} J_s = \frac{h}{2z_m} J_s, \quad (5a)$$

$$J_n = \frac{2}{n\pi} J_s \sin\left(\frac{n\pi h}{2z_m}\right), \quad n \in \{1, 2, 3, \dots\} \quad (5b)$$

where  $H$  is the height of the domain which is equal to  $2z_m$  and  $J_s$  is the surface current density. However, the unit of the current density when transforming from a 3D model into the 2D model should also be taken into account. In the 3D case  $J$  is a surface current density while in the 2D case  $J$  is a line/curve current density. The relation between the surface current density and the line/curve current density assuming uniform distribution of current is  $J_s = \frac{1}{h} J_l$ . Due to the fact that for solving a 3D equation in 2.5D modeling a series of 2-dimensional solvers, which use 2D models, are employed, the boundary condition of the 2.5D equations would be:

$$J_0 = \frac{1}{H} J_l = \frac{1}{2z_m} J_l, \quad (6a)$$

$$J_n = \frac{2}{n\pi h} J_l \sin\left(\frac{n\pi h}{2z_m}\right), \quad n \in \{1, 2, 3, \dots\}. \quad (6b)$$

where  $J_l$  is a line/curve current density. We used the boundary condition as expressed in equations (6) which is slightly different from the boundary conditions reported previously in [3] and [5].

### C. Finite Element Method

Finite Element method is the most popular numerical technique for solving differential equations which offers flexibility for complex geometric shapes and boundaries. One of the Finite Element Methods used to solve the Laplace equation is the Method of Weighted residuals (MWR), the most versatile approach in deriving element properties [6]. MWR is based on deriving a weak form of the governing equation and weighting the residuals in order to discretize the domain,  $\Omega$ , where the FEM is performed. The potential within the mesh,  $\tilde{u}_n(\vec{x})$ , is approximated by a summation of piecewise polynomial interpolation functions weighted by the potential at the nodes as:

$$\tilde{u}_n(\vec{x}) = \sum_{i=1}^N u_i^n \phi_i(\vec{x}). \quad (7)$$

where  $N$  is the number of nodes,  $u_i^n$  are the potential of the nodes in the FEM in which  $n$  denotes the harmonic number of the potential, and  $\phi_i$  are interpolation functions. The Galerkin method is the most common algorithm used in MWR in which the weighting functions, i.e. test functions, are selected the same as the interpolation functions. Employing the Galerkin method, the 2.5D ‘‘local systems matrix’’,  $S^{2.5D^k}$ , for each element,  $k$ , is computed by expression (8). This expression is derived for the 2.5D equation similar to the way the local system matrix is derived for 2D or 3D equation demonstrated with full details in [6] and [7] for the 2D or 3D case.

$$S^{2.5D^k}_{ij} = \int_{E_k} \nabla \phi_i \cdot \nabla \phi_j d\Omega + \left(\frac{n\pi}{z_m}\right)^2 \int_{E_k} \phi_i \phi_j d\Omega$$

$$= S_{ij}^k + \left(\frac{n\pi}{z_m}\right)^2 R_{ij}^k \quad (8)$$

In expression (8), matrix  $S^k$  is the classic integral term calculated for each element which also exists in the 2D model. The matrix  $R^k$  together with its coefficient is addressed in this work as the “residual” integral term. The residual terms modify the accuracy of the 2D model toward the 3D model. The 2.5D local system matrix  $S^{2.5D^k}$  is calculated for each element  $k$  and depends on the spatial frequency. The global systems matrix,  $S^{2.5D}(n)$  of the entire mesh for harmonic  $n$  is then constructed by combining all local stiffness matrices using the connectivity matrix. The global system matrix is used to solve the forward equation, expressed in (9), in order to calculate nodal harmonic voltages  $V_n$  from the driven current pattern  $I_n$ . The system matrix is also employed to construct the sensitivity matrix in the inverse solver.

$$S^{2.5D}(n)V_n = I_n \quad (9)$$

#### D. The Inverse Problem

The inverse problem is the main part of an EIT problem where the conductivity of each element is calculated from a set of boundary measurements. The inverse problem is divided into two categories: static EIT and difference EIT. In static EIT, the value of absolute conductivity is reconstructed using boundary gradient voltages; while in difference EIT, the conductivity difference is reconstructed from a reference conductivity. Applying linearization, the inverse EIT problem is modeled as:

$$\Delta v = \mathbf{J}\Delta\sigma \quad (10)$$

where  $\mathbf{J}$  is known as the “sensitivity matrix” or “Jacobian matrix”,  $\Delta v$  is the measurement voltages and  $\Delta\sigma$  is the difference in conductivity. The measurement voltages are the difference voltage of adjacent electrodes for all driving stimulation patterns.

In the 2.5D problem, for each single 2.5D equation, the relation between the boundary difference voltages,  $\Delta v_n$ , and the difference conductivity,  $\Delta\sigma$ , is written as

$$\Delta v_n = \mathbf{J}_n\Delta\sigma \quad (11)$$

where  $\mathbf{J}_n$  is the sensitivity matrix for harmonic  $n$ . According to the synthesis equation (2b), the summation of  $\Delta v_n$ 's, i.e., the boundary gradient voltages for each partial differential equation (PDE) in (3) forms the actual boundary voltage for the plane  $z=0$ , denoted by  $\Delta v$ . Hence, substituting the relation of each boundary voltage with the conductivity as expressed in (11) yields:

$$\Delta v = \sum_{n=0}^{\infty} \Delta v_n = \sum_{n=0}^{\infty} \mathbf{J}_n \Delta\sigma. \quad (12)$$

Factoring out the conductivity in equation (12) results in

$$\Delta v = \left(\sum_{n=0}^{\infty} \mathbf{J}_n\right) \Delta\sigma. \quad (13)$$

Equation (13) is actually the definition of the entire inverse problem expressed in (10). This fact becomes more obvious when we interpret the summation of the sensitivity matrix of

each PDE's as the sensitivity matrix of the entire problem defined in (14).

$$\mathbf{J} = \sum_{n=0}^{\infty} \mathbf{J}_n \quad (14)$$

### III. THE STRUCTURE OF MODULES

EIDORS [9] is a modular-based open source software for Electrical Impedance Tomography and Diffuse Optical Tomography Reconstruction. The following modules are developed in order to append 2.5D FEM ability to the EIDORS project:

The forward solver module is a module where the forward equation is solved. In this module, ‘fwd\_solve’, a system matrix is reconstructed for each 2.5D partial differential equation. The module ‘calc\_system\_matrix’ calculates the system matrix of the entire mesh for each harmonic,  $n$ , which is explained later. In order to compute nodal voltages from the injected currents, a forward equation is solved for each spatial frequency based on the obtained system matrix of each PDE expressed in (3). This procedure continues until the truncation point,  $n\_max$ , is reached or a sufficient number of equations are solved based on an error criterion. Afterwards, based on the synthesis equation (2b), the Inverse Fourier transform of nodal voltages is computed to form the 2.5D forward solution. The flow diagram of different stages of ‘fwd\_solve’ module is displayed in Fig. 1.

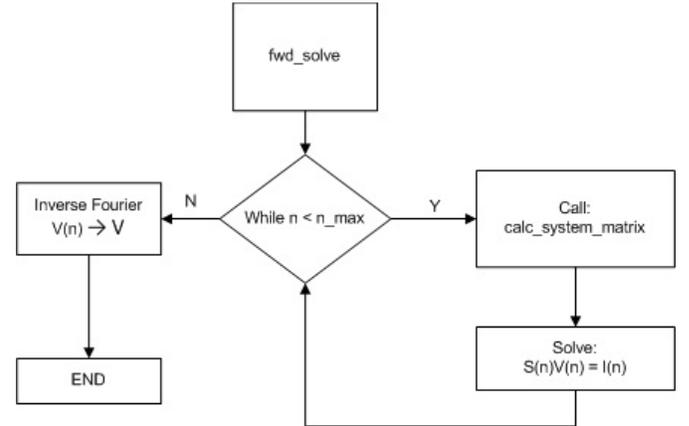


Figure 1. Flow diagram of the ‘fwd\_solve’ module

The ‘calc\_system\_matrix’ module, calculates the 2.5D system matrix of the entire mesh based on the harmonic number  $n$ , the parameter ‘ $z_m$ ’, i.e. one half of the domain height, and the position of the nodes. For this purpose, we calculated the local system matrix of each element and constructed a connectivity matrix for the entire mesh according to the elements’ connectivity map. After calculating all local system matrices, the entire mesh system matrix is computed by the following equation:

$$S = C^T S_E \Sigma C \quad (15)$$

In equation (15),  $S_E$  is a diagonal concatenation of local system matrices which were developed by the ‘calc\_local\_stiffness\_matrix’ in the previous stage, matrix  $\Sigma$  is the conductivity matrix,  $C$  is the “connectivity map” matrix and ‘ $T$ ’ represents

the transpose operation. Connectivity map is the relation between the local node index inside each element and the general node index for the entire mesh. The flow diagram of different stage of this module is plotted in Fig. 2.

In the ‘calc\_local\_stiffness\_matrix’ module, the matrix  $S^{2.5Dk}$  is calculated for each element based on integral solutions expressed in (8) given the following parameters: ‘ $n$ ’, ‘ $z_m$ ’, and the nodal position of that element.

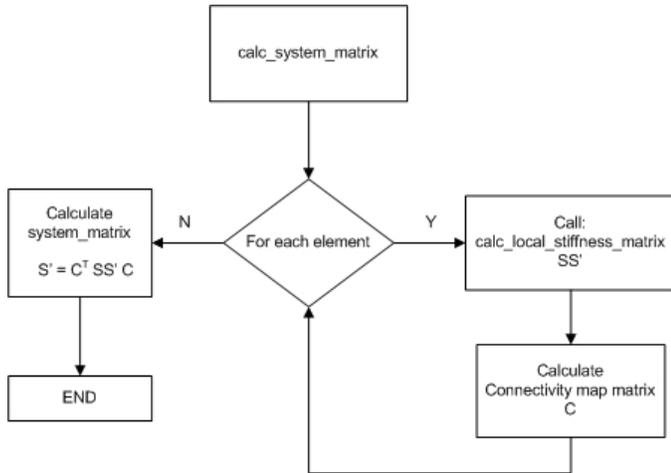


Figure 2. Flow diagram of the ‘calc\_system\_matrix’ module

#### IV. COMPLETE ELECTRODE MODEL

The Complete Electrode Model (CEM) refers to an electrode modelling technique where each electrode is modelled by more than one single node in the finite element mesh. In addition, skin-electrode contact impedance,  $z_c$ , is modelled in system matrix calculations. The contact impedance is the impedance assumed under the electrode area due to imperfect conductivity in electrode-skin contact region. In [4] and [8], only the Point Electrode Model is considered where each electrode is modelled as one single node and CEM is not modelled.

The 2.5D method employs a 2D mesh for its model. The right hand side, or the current side, of the weak form of the governing equation resulted from the PDEs in (3) remains unchanged within the procedure of deriving expression (8) for the 2.5D problem. Therefore, the additional matrix terms for CEM are the same as the 2D CEM until this stage.

In CEM a virtual node is assumed for each electrode. The voltage of each electrode in a mesh using CEM is the voltage of this virtual node. This voltage is a weighted average of the voltages of all nodes under that electrode. In a 3D mesh, the voltage of the electrode nodes, located in different slices ( $z$ -layers) on the boundary, are averaged for computing the electrode voltage. Hence, in order to completely model CEM for computing the 2.5D electrode voltage, it is required to calculate the inverse Fourier in (2b) at all  $z$ -coordinates where an electrode node exists in the 3D mesh; and then compute the average voltage of nodes under the electrode by the same weights used for the 3D mesh. These weights are calculated based on the area of the region that each node builds with its

directly connected neighboring nodes under the electrode. In case the structure of the 3D mesh is not known from the 2.5D point of view, the weights could be approximated as the weights of a line electrode in which the weights of end nodes are half of the weights of the midpoints and all midpoints have equal weights.

#### V. RESULTS & DISCUSSION

Theoretically, the 2D model is an equivalent model of a 3D tank having translational invariant conductivity along  $z$  with unit height, electrodes with full heights, i.e. unit height, and a same cross section (base) as the 2D model. This statement is based on the fact that by assuming full height electrodes ( $h = 2z_m$ ), the Fourier series of the current density which forms the boundary condition in (6) is reduced to a single coefficient at  $n=0$ ; hence, the set of equations (3) is reduced to a single regular 2D equation. However, only in the case of a unit height domain the boundary condition is equal to a regular 2D model.

In addition, in a 2D model the conductivity has a unit of  $\Omega^{-1}$  while in 3D the conductivity has the unit of  $\Omega^{-1}/m$ . Hence, the solution of a 3D mesh with translation invariant conductivity along  $z$  with full height electrodes is equal to the solution of its base 2D mesh normalized by the 3D mesh height. In general, the 2.5D solution,  $V_{2.5D}$ , is written as:

$$V_{2.5D} = \frac{1}{H} V_{2D} + \sum_{n=1}^{\infty} \frac{2}{n\pi h} \sin\left(\frac{n\pi h}{2z_m}\right) \cos\left(\frac{n\pi}{z_m} z\right) S'(n)^{-1} I \quad (16)$$

where  $V_{2D}$  is the 2D solution,  $I$  is the current vector or the matrix of all current patterns and  $H$  is the domain height, i.e.  $2z_m$ . Equation (16), demonstrates how, in the 2.5D FEM, higher order terms modify the normalized 2D solution toward the 3D solution.

##### A. Accuracy

Evaluation of forward problem could be a measure to validate the performance of the system matrix. Figure 3a shows a 2D finite element mesh with 4096 elements and 2113 nodes employed for modelling the 3D problem. In fig. 3b, a 3D finite element mesh consisted of 61 layers of the 2D mesh is illustrated having 737,280 elements and 128,893 nodes. Figure 4 illustrates the measurement (gradient) voltages resulted from 2D, 2.5D, and 3D solvers. In this evaluation the opposite current pattern is employed in 16-electrode system. Comparing the results of the 2.5D solution with the solution of the 2D model, 3D model and the first term of the 2.5D model ( $2D/H$ ) in fig. 4 demonstrates that the 2.5D algorithm works well in modifying the solution of the 2D model toward the solution of the 3D model. For the 2.5D solution, the electrode voltage and the measurement electrodes lie on top of the 3D case with a precision improved by decrement in the element size.

Figure 5 illustrates maximum relative error that occurs between the solution of 3D gradients and 2.5D gradients for three different electrode heights ( $h$ ). The element height in the 3D mesh decreases as the number of  $z$ -layers increases. This figure shows that the 3D solution itself improves since by decrement of the element height, the 3D solution changes while the 2.5D solution does not depend on the element height. Therefore, reducing only the 3D finite element height results in

a lower error rate; this shows that the 3D solution approaches

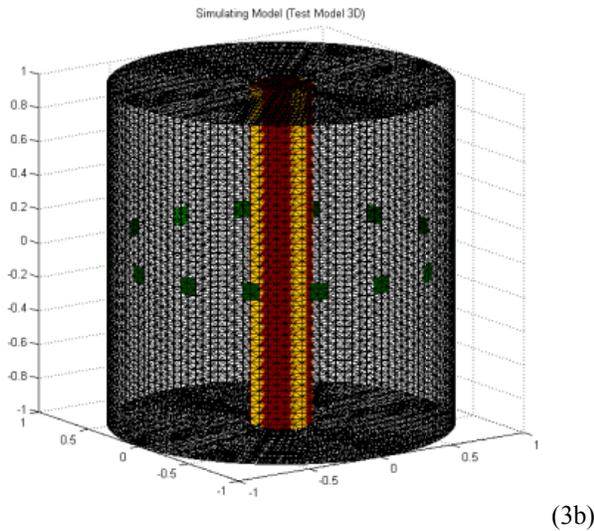
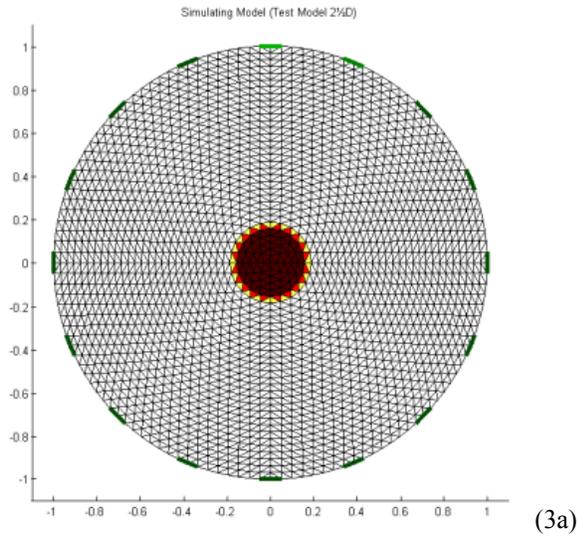


Figure 3. (3a) 2D mesh with 4096 elements used for the 2.5D method (3b) 3D mesh with 737,280 elements

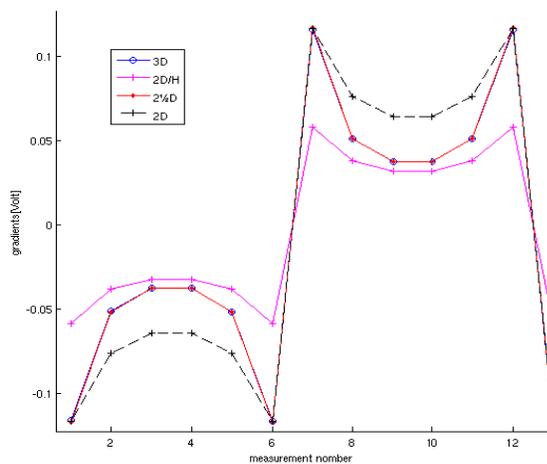


Figure 4. Gradients for opposite current pattern for only first 5 terms of the summation.  $H=2$ ;  $h=0.1$ ,  $w \approx 0.1$ ,  $N_{Max} = 4$ ; Maximum error: 0.82% (0.001)

the 2.5D solution.

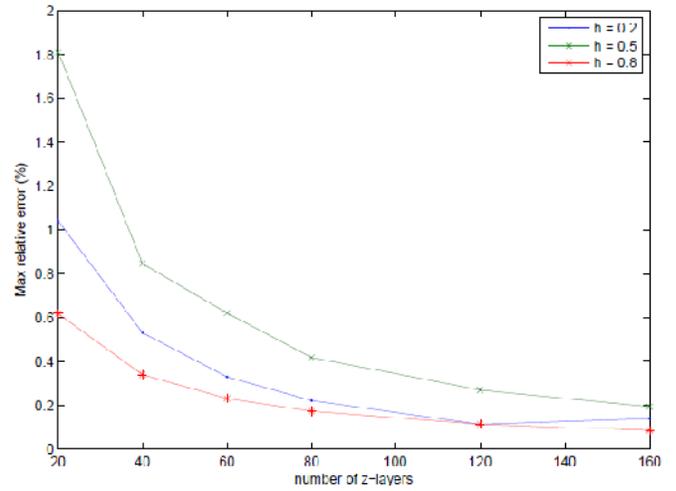


Figure 5. Max relative difference between the 3D and 2.5D gradient solutions for 3 different electrode heights versus the element height (z-layers)

### B. How many terms are needed?

The number of equations sufficient to be solved for the 2.5D solution with an acceptable accuracy differs by the problem parameters such as the height of the electrodes and the height of the domain. As  $n$  increases in equation (16), the driving currents in the nominator decrease by a 'sinc' function pattern while the system matrix increases in the denominator by the  $n^2$  coefficient. Table 1 lists the truncation points for gradients given 1% error caused by the truncation which is the relative error introduced if the summation of harmonics is truncated at  $n_{max}$ . The table contains three different tables for three tank Height/radius situations. It can be observed from this table that for greater  $h/H$  values (electrode height over tank Height), the truncation point is even smaller. For example when the tank height/radius ( $H/r$ ) is 2 which might represent a typical body phantom, even 3 extra 2.5D terms are sufficient.

Table 1: Truncation point ( $n_{max}$ ) for gradient solution for different tank Height ( $H$ ) over tank radius ( $r$ ) and electrode height ( $h$ ) given 1% accuracy in truncation.

$H/r$	2		4		10	
	$\frac{h}{H}$	$n_{max}$	$\frac{h}{H}$	$n_{max}$	$\frac{h}{H}$	$n_{max}$
(0.025, 0.05, 0.1)		3	(0.025)	7	(0.025)	13
(0.1)		2	(0.05)	6	(0.05)	11
(> 0.1)			(0.1)	5	(> 0.1)	7
			(> 0.2)	3		

After  $(n_{max})^{\text{th}}$  extra 2.5D terms, mostly the harmonic voltage under the active electrodes remains non-zero and other harmonic voltages are approximately zero. However, the harmonic voltages under the active electrode cancel out the effect of each other over  $n$ . In addition, the voltage on the active electrodes is not counted in the gradient measurements. When the height of the electrodes,  $h$ , is small compared to the height of the domain,  $H$ , the voltages under the electrode is approximately constant; hence the electrode voltage could be approximated as the voltage of the node at  $z=0$ . As the  $h/H$  ratio reaches 1, harmonic voltages converge faster to zero when

$n$  increases; hence, the truncation point of the series,  $n_{max}$ , is smaller. However, in this case, the voltage under the electrode for the 3D model in different heights is not constant. In this case, it is required to completely calculate the 2.5D CEM solution for all positions under the electrode and average the voltages of different  $z$ -layers mentioned earlier in section IV. This is due to the fact that the current is not distributed homogeneously over the surface of the electrode in a 3D CEM which is more realizable when the height of the electrode is comparable to the height of the domain.

### C. Time/Memory Performance

The matrix  $S^k$  is easy to be calculated since for linear interpolation functions,  $\phi_i$ , the gradients are constant over the integral. However, calculating the residual matrix is more difficult and requires computing the following integrands for all elements of the whole mesh which are taken over each element separately:  $x, y, x^2, xy, y^2$ . The global residual matrix  $R$  for the entire mesh could also be constructed by all local residual matrices,  $R^k$ , and the connectivity matrix the same way mentioned previously in (15) as:

$$R = C^T R^k \Sigma C \quad (17)$$

In addition, since, the residual matrix itself is independent of the spatial frequency  $\frac{n\pi}{z_m}$ , it could be calculated only once for the whole 2.5D set of equations. Therefore, the 2.5D global system matrix,  $S^{2.5D}(n)$ , for each equation in (3) would be calculated as:

$$S^{2.5D}(n) = S + \left(\frac{n\pi}{z_m}\right)^2 R \quad (18)$$

where  $R$  and  $S$  are constant matrices and only depend on the position of the nodes, i.e. mesh geometry. The run time of the 2.5D method would be equal to the running time of one 2D module plus the running time for solving extra forward equations, expressed in (9), having the same dimension as the 2D problem for the non-zero harmonic terms,  $t_{2D-Fwd}$ , and the time for calculating the residual matrix,  $t_R$ , as well as the inverse Fourier transform  $t_{IFT}$ .

$$t_{2.5D} = t_{2D} + n_{max} \times t_{2D-Fwd} + t_R + t_{IFT} \quad (19)$$

On the other hand, for a same model, the 2.5D solver requires as many elements and nodes as the 2D mesh requires; while the 3D model requires much more elements and memory. Assuming employing a 3D mesh consisted of  $M$  slices ( $z$ -layer) of a typical 2D mesh, the whole 3D mesh would then contain  $M$  times as many nodes as the 2D mesh nodes and  $3(M-1)$  times as many elements as the 2D mesh elements. Then, the system matrix of the 3D mesh contains  $M^2$  times as many elements as the 2D system matrix. Also, the 3D sensitivity matrix contains  $3(M-1) \times (\text{number of measurements})$  times as many elements as the 2D sensitivity matrix. The 3D inverse problem involves inversion of a  $9(M-1)^2$  larger matrix than the 2D inverse problem. For example, a typical 2D mesh contains

2,013 nodes and 4,094 elements. The 3D mesh employing 61 frames of the 2D mesh as its base would then contain 61 times as many nodes, i.e., 86,633, and 180 times as many elements, i.e., 491,520 than the 2D mesh. Then, the system matrix of the 3D mesh is  $41 \times 41 = 1,681$  times larger than the 2D system matrix. Assuming a 16-electrode system and employing adjacent current pattern, the number of measured voltages would be  $16 \times 13 = 208$ ; then, in this case, the 3D sensitivity matrix would be  $120 \times 208 = 24,960$  times larger than the 2D sensitivity matrix.

Therefore, the trade of for the forward problem is this: inversion of a  $M^2$  times larger matrix vs. multiple inversion of a much smaller matrix. As discussed in the previous section, the number of inversions could be as few as 4 depending on the model specification; or at most 10 reported in [8].

## VI. CONCLUSION

We have modified the EIDORS project by developing new modules -and extending the existing ones- in order to improve this project with the 2.5D Finite Element Method. Also, the proper boundary condition was validated considering the 3D forward solution as the reference. The implementation of complete electrode model has also been considered in the algorithm. In addition, we showed that only 4 to 8 equations are sufficient to be solved given 1% accuracy caused by the truncation for a problem similar to human body scales; while it is not necessary to calculate the system matrix of these equations separately which would also lead to save more time. The accuracy issue and the time/memory performance were also discussed.

## REFERENCES

- [1] A Adler and W.R B Lionheart, "Uses and abuses of EIDORS: An extensible software base for EIT", *Physiol. Meas.* 27:S25-S42, 2006.
- [2] KS Cheng, D Isaacson, JC Newell, DG Gisser, "Electrode models for electric current computed tomography", *IEEE Trans. Biomed. Eng* 1989; 36:918-24.
- [3] K. Jerbi, W.R.B. Lionheart, P.J. Vauhkonen, M. Vauhkonen, "Sensitivity Matrix and Reconstruction Algorithm for EIT Assuming Axial Uniformity" *J. Physiological Measurement*, Volume 21, Issue 1, pp. 61-66, 2000.
- [4] Y.Z. Ider, N.G. Gencer, E. Atalar and H. Tosun, "Electrical impedance tomography of translationally uniform cylindrical objects with general cross-sectional boundaries" *IEEE Trans. Med. Imaging* 9 49-59, 1990.
- [5] W.R.B. Lionheart, "Uniqueness, shape and dimension in EIT" *Ann. NY Acad. Sci.* 873 466-71, 1999.
- [6] B. Graham, "Enhancements in EIT Image Reconstruction for 3D Lung Imaging", Ph.D. dissertation, Dept. Sys. & Comp. Eng., Carleton University, Ottawa, Canada, 2007.
- [7] D. Holder, "The reconstruction problem," in *Electrical Impedance Tomography: Methods, History, and Applications*, Phys. Med. Biol. 50 (2005) 2427-2428.
- [8] J.E. Butler and R.T. Bonnecaze, "Imaging of particle shear migration with electrical impedance tomography" *Am. Inst. Chem. Eng. Annu. Meeting* (1997).
- [9] EIDORS: <http://eidors3d.sourceforge.net/>