

$2\frac{1}{2}$ D Finite Element Method for Electrical Impedance Tomography Considering the Complete Electrode Model

by

Navid Bahrani

A Thesis submitted to the Faculty of Graduate and Postdoctoral Affairs
in partial fulfilment of the requirements for the degree of

Master of Applied Science

in

Electrical and Computer Engineering

Ottawa-Carleton Institute for Electrical and Computer Engineering (OCIECE)

Department of Systems and Computer Engineering

Carleton University

Ottawa, Ontario, Canada, K1S 5B6

January 2012

Copyright ©

2012 - Navid Bahrani

Abstract

Electrical Impedance Tomography (EIT) is a medical imaging technique in which diagnostic electrical current is driven into a body and surface electrical voltages are measured in order to reconstruct an image of the conductivity or permittivity of the body. EIT is categorized as a soft-field tomographic method because the current propagates in a 3D domain even if the electrodes are located in a plane. In addition, the current injection source has a 3D geometry in practice which implies 3D considerations. However, accurately solving the 3D problem employing Finite Element Method (FEM) needs a great number of elements leading to a large computational complexity.

The goal of this work is to develop and test a $2\frac{1}{2}$ D finite element method algorithm for the EIT problem which will reduce the memory and computation required. The goal of a $2\frac{1}{2}$ D FEM is to solve the 3D problem by solving a set of modified 2D problems instead. In this work, a $2\frac{1}{2}$ D finite element solver is implemented considering the complete electrode model (CEM). For this purpose, complimentary modules are developed to enhance the EIDORS¹ project by the $2\frac{1}{2}$ D FEM. In addition, the boundary current injection is calculated, the algorithm is validated and the time and memory performance is evaluated. Finally, the number of $2\frac{1}{2}$ D equations to be solved for a given accuracy is investigated and an explanation is also provided for improving the speed of the algorithm.

¹[www.eidors.org \(http://eidors3d.sourceforge.net\)](http://eidors3d.sourceforge.net)

Dedicated to my wife Niloufar
and my parents.

Acknowledgments

I would like to thank Prof. Andy Adler, my supervisor, for his valuable suggestions, advice and support during this research. I am also thankful to EIDORS² [1] [2] for providing me with a useful tool and software environment to implement the 2½D finite element model and algorithm.

I would like to thank Carleton University as well as the department of Systems and Computer Engineering for the scholarships and Prof. Andy Adler for the research grants awarded to me which was crucial to the successful completion of this project.

I would also like to express my greatest gratitude to Dr. Chung-Horng Lung, Dr. Yuu Ono and Dr. Adrian Chan for their kind support as well as Dr. Adrian Chan and Dr. Michel Labrosse for their valuable comments on this thesis.

Of course, I am grateful to my wife, Niloufar, and my parents for their patience, *love* and constant encouragements. Without them this work would never have come into existence.

²<http://eidors3d.sourceforge.net/docs.shtml>

Contents

Abstract	ii
Acknowledgments	iv
Table of Contents	v
List of Tables	ix
List of Figures	x
List of Acronyms	xiii
List of Symbols	xiv
1 Introduction	1
1.1 Background and Motivations	1
1.2 Objective	3
1.3 Contributions	3
1.4 Thesis Summary	4
2 Background	6
2.1 Tomography	6
2.1.1 Computed Tomography	7
2.2 Inverse Problems	7

2.3	What is EIT?	8
2.4	Why EIT is difficult?	9
2.5	Advantages/Disadvantages of EIT	10
2.6	Current Pattern Stimulation	12
2.6.1	Adjacent Current Pattern	12
2.6.2	Opposite Current Pattern	13
2.7	Forward Problem	14
2.7.1	Forward Equation	14
2.7.2	Finite Element Method	16
2.8	Inverse Problem	19
2.8.1	Jacobian or Sensitivity Matrix	21
2.8.2	Solving the Inverse Problem	22
2.8.3	Static EIT	24
2.8.4	Difference EIT vs. Static EIT	24
2.9	The 2.5D literature	25
2.9.1	What does 2.5D refer to?	25
2.9.2	Geophysics	27
2.9.3	Miscellaneous Non-tomographic Problems	29
2.9.4	Process Tomography	31
2.9.5	Biomedical Applications	33
2.9.6	Conclusion	35
3	$2\frac{1}{2}$D Finite Element Method	37
3.1	The $2\frac{1}{2}$ D Problem	38
3.2	$2\frac{1}{2}$ D Boundary Condition	41
3.3	Method of Weighted Residuals (MWR)	44
3.3.1	Left Side of the Governing Equation	46

3.3.2	Right Side of the Governing Equation	49
3.3.3	Complete Equation	50
3.3.4	Analytical Solution of the Integrals	52
3.4	Adding Complete Electrode Model	56
3.5	Inverse Problem	60
3.6	Example of a Simple 2D Mesh	62
3.6.1	Connectivity Matrix	62
3.6.2	Stiffness Matrix	64
3.6.3	CEM terms	68
3.7	A more complex 2D mesh	68
3.8	A 3D mesh having 2D boundary electrodes	73
3.9	A 3D mesh having 1D line electrodes	77
3.10	Chapter Summary	78
4	Validation	79
4.1	Finding a $2\frac{1}{2}$ D problem for comparison	80
4.1.1	Analyzing the set of $2\frac{1}{2}$ D equations	80
4.1.2	2D vs. 3D for $H=h=1$	81
4.1.3	Comparing Electrode Voltages	84
4.2	Variation of the 3D element height	86
4.2.1	Variations in the 3D solution	86
4.2.2	Variations in the 3D-2D difference	87
4.3	The first term of $2\frac{1}{2}$ D	88
4.3.1	Analysis of the terms of $2\frac{1}{2}$ D	89
4.3.2	Normalized 2D vs. 3D	90
4.4	$2\frac{1}{2}$ D vs. 3D and 2D	91
4.5	Sources of Errors	96

4.5.1	3D Interpolation Function	96
4.5.2	Injected Current Pattern	98
4.5.3	2D-based Complete Electrode Modelling	99
5	Implementation and Complementary Discussion	103
5.1	The Structure of Modules	103
5.2	Improving the Speed of the Algorithm	105
5.3	How Many Terms Are Needed?	109
5.4	What we gained	112
5.4.1	Memory Performance	112
5.4.2	Time Performance	114
6	Conclusions and Future Works	122
6.1	Summary and Conclusion	122
6.2	Future Works	124
6.2.1	Assuming Symmetric Conductivity	124
6.2.2	Windowing the Injected Current	127
6.2.3	2.5D image reconstruction employing $2\frac{1}{2}$ D finite element mod- elling	128
6.2.4	Error for the 'True' 3D Thorax	128
6.2.5	Generalization - Full Impedance	129
	Appendix A	130
	Appendix B	144
	References	147

List of Tables

5.1	Truncation point (n_{max}) for different Tank Height to radius ratio (H/r) and electrode height to Tank Height ratio (h/H) given 1% accuracy in measurements	112
5.2	Time performance for 3D	114
5.3	Time performance for $2\frac{1}{2}$ D	116
5.4	Number of Elements (K) and Nodes (N) in the 3D mesh	119
5.5	Number of Elements (K) and Nodes (N) in the $2\frac{1}{2}$ D mesh employing different number of xy-layers (circles)	119

List of Figures

3.1	A $2\frac{1}{2}$ D model - The green region is an electrode	38
3.2	A sample 2D mesh - Green regions are electrodes	63
3.3	A sample complex mesh - Green regions are electrodes	69
3.4	Global interpolation function for node 7 and its intersection with the electrode forming a triangle - Green regions are electrodes	70
3.5	Global interpolation function for (left) node 3 (right) node 2 - Green regions are electrodes	70
3.6	How integration is performed for an electrode end node and an elec- trode mid node	71
3.7	Use of colors for demonstrating 3D linear interpolation function, $\phi_A(x, y, z) = \frac{z}{z_1}$	74
3.8	Different kinds of nodes on the boundary of a 3D mesh. Green regions are electrodes, gray regions are subregion of electrode and selected nodes are labeled by alphabetical letter in blue	76
4.1	2D and 3D mesh	82
4.2	Max relative difference between the 2D solution and 3D solution on different layers for the 3D mesh in figure 4.1 with 41 layers, $H = h = 1$ - Complete Electrode Model. $\varepsilon(\%) = \max_{xy} V_{2D}^{CEM}(x, y) - V_{3D}^{CEM}(x, y, z) $	83

4.3	Max relative difference between the 2D and 3D on different layers for (left) solution (right) gradients, $H=h=1$ - Point Electrode Model vs. line electrode in 3D. $\varepsilon(\%) = \max_{xy} V_{2D}^{PEM}(x, y) - V_{3D}^{line}(x, y, z) $	84
4.4	Difference of Electrode voltage between 2D and 3D for all stimulation patterns, $H = h = 1$. $\varepsilon = E_{l_{2D}}^{CEM} - E_{l_{3D}}^{CEM} $	85
4.5	Variations of 3D solution by changing element height Δz compared to the last value plotted for different electrode height (h). $\varepsilon(\%) = \max V_{3D}^{CEM}(\Delta z) - V_{3D}^{CEM}(\Delta z = \frac{1}{160}) $	87
4.6	Maximum relative difference in measurements between 2D and different 3D solutions calculated for a same 3D mesh with different element height (Δz) for Electrode Width (W) = {0, 0.1}. $H = h = 1$, $\varepsilon(\%) = \Delta V_{2D} - \Delta V_{3D}(\Delta z) $	88
4.7	Max relative difference between normalized 2D and 3D solutions in different z-layer calculated for $H=h=\{2, 4, 6, 10\}$ - PEM. $\varepsilon(\%) = \max_{xy} V_{2D}^{PEM}(x, y) - V_{3D}^{line}(x, y, z) $	91
4.8	Comparing 3D, 2D, 2D/ H (first term of $2\frac{1}{2}$ D) and $2\frac{1}{2}$ D solutions for (up) electrode voltages and (down) measurements (adjacent difference electrode voltages), $H = 2$, $h = 0.4$, $n_{max} = 50$ and $\Delta z = \frac{2}{160}$ - PEM (line in 3D) ($W = 0$)	93
4.9	Comparing 3D and $2\frac{1}{2}$ D solutions for nodes at electrode positions at different z-layers $V_{El}(z)$ - $H = 2$, $h = 0.4$, $W = 0$, $\Delta z = \frac{2}{160}$ and $n_{max} = 50$	94
4.10	Difference between gradient at different z-layers for (left) 3D vs. $2\frac{1}{2}$ D (right) 3D vs. 2D/ H (down)3D or $2\frac{1}{2}$ D vs. 2D, $H = 2$, $h = 0.4$, $n_{max} = 50$ for $2\frac{1}{2}$ D and $\Delta z = \frac{2}{160}$. $\varepsilon(\%) = \Delta V_{El}^A(z) - \Delta V_{El}^B(z) $. . .	95

4.11	Comparing 3D, 2D, 2D/H (first term of 2½D) and 2½D CEM solutions for (up) electrode voltages and (down) measurements - CEM ($W = 0.1$, $H = 2$, $h = 0.4$, $n_{max} = 50$ and $\Delta z = \frac{2}{160}$)	97
4.12	Maximum relative difference between the 3D and 2½D measurements for different electrode height using different element height (Δz) or (number of z-layers) for $h = \{0.4, 1.0, 1.6\}$, $H=2$, $n_{max} = 50$	98
4.13	Distribution of current density (left) assumed for 2½D (right) real 3D mesh with complete electrode model - $\Delta z = \frac{2}{160}$ and electrode width $W = 0.1$	99
4.14	Comparing 3D and 2½D solutions for nodes at electrode positions at different z-layers electrode width (W) = 0.1, $H = 2$, $h = 0.4$, $\Delta z = \frac{2}{160}$ and $n_{max} = 50$	100
5.1	Block diagram of the 'fwd_solve' module	104
5.2	Block diagram of the 'calc_system_matrix' module - K is the total number of elements in the mesh	106
5.3	Block diagram of the 'fwd_solve' module explained in section 5.2.	108
5.4	Max relative error in measurements introduced if the summation is truncated at n_{max}	110
5.5	Max relative error in voltages introduced if the summation is truncated at n_{max}	111
5.6	Running time for 3D - N_circ is the number of circles (xy-layers)	118
5.7	Running time for 2½D - h is electrode height and $H=2$ is tank Height	120
5.8	3D/2½D Time Performance ratio - N_circ is the number of circles (xy-layers)	120
A.1	A sample triangular element	130

List of Acronyms

Acronyms	Definition
ARDS	Acute Respiratory Distress Syndrome
CEM	Complete Electrode Modelling
CT	Computed Tomography
EIDORS	Electrical Impedance Tomography and Diffuse Optical Tomography Reconstruction Software
EIT	Electrical Impedance Tomography
ERT	Electrical Resistance Tomography
FEM	Finite Element Method
KCL	Kirchhoff's Current Law
KVL	Kirchhoff's Voltage Law
MWR	Method of Weighted Residuals
MIT	Magnetic Induction Tomography
PDE	Partial Differential Equation
PEM	Point Electrode Modelling
SPECT	Single Photon Emission Tomography

List of Symbols

Symbols	Definition
h	electrode height
i	index of the node
j	normal (component) of boundary current density
k	index of the element
l	index of the electrode
n	harmonic number of spatial frequency
n_{max}	truncation point of the $2\frac{1}{2}$ D series
\hat{n}	unit vector normal to the boundary
r	tank radius
$\tilde{u}_n(\vec{x})$	n^{th} harmonic potential at \vec{x}
u_i^n	n^{th} harmonic potential of the i^{th} node
v	test function used in MWR
$v(\vec{x})$	test function used in MWR at \vec{x}
\vec{x}	position: (x, y) for 2D, (x, y, z) for 3D
\hat{x}	conductivity
z_l	l^{th} electrode contact impedance
z_m	one half of domain height
C	Connectivity Matrix
\vec{E}	Electric Field

E_k	k^{th} Element
E_l	l^{th} Electrode
H	domain Height
\mathbf{I}	Identity Matrix
I_l	l^{th} Electrode current
I_n	n^{th} harmonic current
\mathbf{J}	Jacobian (or Sensitivity) Matrix
\vec{J}	current density
J_s	surface current density
J_n	n^{th} harmonic of current density
J_l	line current density
K	number of elements
L	Number of Electrodes
M	number of 2D slices (z-layers) in a 3D mesh
N	number of nodes
\mathbf{R}	Regularization Matrix
R	Residual Matrix - $2\frac{1}{2}$ D extra terms
R_{ij}^k	ij^{th} element of the Residual Matrix of the k^{th} element - $2\frac{1}{2}$ D extra terms
R_E	a diagonal concatenation of local residual matrices without conductivity
S	System (Stiffness) Matrix - Generally for 2D or 3D
S_{ij}^k	ij^{th} element of the System (Stiffness) Matrix of the k^{th} element - Generally 2D
S_E	a diagonal concatenation of local system matrices without conductivity
S'	$2\frac{1}{2}$ D System (Stiffness) Matrix

S_{ij}^k	ij^{th} element of the $2\frac{1}{2}$ D System (Stiffness) Matrix of the k^{th} element
U_n	voltage of the n^{th} node
V	Voltage
V_l	l^{th} Electrode voltage
V_n	n^{th} harmonic voltage
W	electrode width
σ	conductivity
λ	regularization parameter
φ	voltage (electric potential)
$\phi_i(\vec{x})$	interpolation function for node i
ϕ_a	interpolation function for node a
Σ	Conductivity Matrix
Δz	element height
Δw	element width
∇_{2D}	two-dimensional gradient operator
∇_{3D}	three-dimensional gradient operator

Chapter 1

Introduction

1.1 Background and Motivations

Electrical Impedance Tomography (EIT) is a medical imaging technique in which electrical current is driven into a body and surface electrical voltages are measured in order to reconstruct an image of the conductivity or permittivity of the body. EIT is a suitable technique for long-term monitoring because it is relatively low cost, a fast imaging modality, portable and non-invasive. However, EIT is categorized as a soft-field tomographic method due to the fact that the current propagates in a 3D domain even if the electrodes are located in a plane. In addition, the current injection source has a 3D geometry in practice. Hence, the EIT problem is inherently a 3D problem.

For modelling the EIT problem, a Finite Element approach is often employed in which the medium is segmented into a finite number of elements. Accurately solving the 3D problem, however, requires a great number of elements (e.g., 1 million elements) which implies a large capacity of memory (on the order of 16-64 GBytes) and leads to a large computational complexity compared to a 2D FEM (e.g., 4,000 elements). In the forward EIT problem, a "system matrix" is constructed which is then inverted for solving the forward equation; and in the inverse EIT problem,

a "sensitivity matrix" is constructed which is again inverted for solving the inverse equation.

The $2\frac{1}{2}$ D finite element modelling provides an advantageous trade off for this problem which aims to solve the 3D equation by employing a 2D model and solving a small number (about 5 to 10 depending on the geometry) of 2D-like equations, by assuming translational invariance. There has been some work on $2\frac{1}{2}$ D finite element modelling of EIT with the same geometry and governing equation as in the biomedical applications [3], [4], and [5]; however, these works considered only the point electrode mode (electrodes are modelled by a single node). For biomedical applications, the complete electrode modelling is important due to the effect of skin-electrode contact impedance. The complete electrode modelling refers to the situation where each electrode includes more than one node in the FE mesh and the effect of contact impedance is considered in calculating the voltage of electrodes.

In addition, we could not find any open source code available for $2\frac{1}{2}$ D finite element modelling of the EIT to evaluate the previous works on this topic. This work aims to implement the $2\frac{1}{2}$ D finite element method for the Electrical Impedance Tomography considering also the Complete Electrode Model (CEM) in order to enhance the EIDORS¹ open source project with this capability. CEM refers to an electrode model in which more than one node is assigned for each electrode where the skin-electrode contact impedance is modelled (see [6] for more details). EIDORS is an open source Matlab-based software for Electrical Impedance Tomography and Diffuse Optical Tomography Reconstruction. The project aims to provide free software algorithms for forward and inverse modelling for EIT in medical and industrial settings, and to promote collaboration as well as resource sharing between groups working on this field [7].

¹www.eidors.org

1.2 Objective

The goal of this work is to develop a $2\frac{1}{2}$ D finite element algorithm, test and evaluate it, implement a $2\frac{1}{2}$ D solver considering the complete electrode model (CEM), and develop complementary modules to enhance the EIDORS project by the $2\frac{1}{2}$ D Finite Element technique. For the forward part of the EIT problem, the goal is to derive the $2\frac{1}{2}$ D system (stiffness) matrix from the set of partial differential equations and apply the Complete Electrode Model (CEM) to the $2\frac{1}{2}$ D system matrix. In addition, one of the objectives is to investigate how many 2D-like equations are sufficient to solve given a reasonable accuracy. For the inverse part of the EIT problem, the goal is to derive and implement the $2\frac{1}{2}$ D sensitivity matrix.

1.3 Contributions

In this work, a $2\frac{1}{2}$ D finite element method and algorithm has been applied to the EIT problem. The key contributions of this work are as follows:

1. The $2\frac{1}{2}$ D finite element method is formulated with a correct form of boundary current injection for EIT. We derived the boundary condition slightly different than what was reported in [8] and [4]; and tested that successfully.
2. The complete electrode model (CEM) is applied to the $2\frac{1}{2}$ D model in two stages, which is novel.
3. The developed forward solution was evaluated directly and the correct form of boundary current injection was tested and validated.
4. A matlab-based software was developed to implement the algorithm and enhance the EIDORS project with the $2\frac{1}{2}$ D FEM.

5. Some strategies are suggested to enhance the algorithm performance and reduce its computational complexity as well as processing time by avoiding unnecessary calculations.
6. An analysis was performed to quantify the improved performance between the $2\frac{1}{2}$ D and 3D.

1.4 Thesis Summary

In this work, the boundary condition for each partial differential equation (PDE) is calculated from the 3D boundary condition based on a uniform distribution of current under the electrodes using the cosine Fourier transform. In addition, the complete electrode model is applied to the $2\frac{1}{2}$ D model in two stages. In the first stage, the $2\frac{1}{2}$ D system matrix is extended by complementary CEM terms. In the second stage, based on the averaging weights obtained for CEM nodes, the implemented 2D-based CEM is modified in order to consider the 3D geometry by averaging the $2\frac{1}{2}$ D solutions calculated for different z-layers (z) under the electrodes. It is demonstrated that for calculating the $2\frac{1}{2}$ D solution at a different height only the inverse Fourier is required to be recalculated (It is not necessary to solve the forward problem for each layer). In addition, a complete description on construction of the system matrix with CEM is provided where it is investigated how, in CEM, the voltage of each electrode is calculated from the voltages of the nodes under that electrode.

In chapter 3, the $2\frac{1}{2}$ D system (stiffness) matrix is derived from the set of $2\frac{1}{2}$ D PDE using the method of weighted residuals and the Galerkin method inspired from the general derivation of system matrix for the 2D and 3D methods available in [9] and [10]. Here, for the purpose of demonstrations which will be used later in improving the speed of the algorithm, the $2\frac{1}{2}$ D system matrix is separated in two parts; one part is the same as a regular 2D system matrix and the other part is called 'residual

matrix' accounting for the extra terms of the $2\frac{1}{2}$ D method. It is shown that these two parts only depend on the problem geometry by factoring out the spatial frequency coefficient which eases the computation complexity. For the inverse part of the EIT problem, it is shown that the $2\frac{1}{2}$ D sensitivity matrix is calculated by adding the sensitivity matrix of all partial differential equations.

In chapter 4, the developed solution was systematically evaluated and the correct form of boundary current injection was tested and validated. It is shown that the $2\frac{1}{2}$ D solution works satisfactorily in modifying the 2D solution toward the 3D solution. Furthermore, the accuracy of the forward solver is evaluated directly and not based on the solution of the EIT inverse problem which was done in [4] for the point electrode model. Moreover, an analysis on the sources of error that prevent a perfect match between the $2\frac{1}{2}$ D and the 3D solution has been performed. In chapter 5, a study on the truncation point of the series is performed. It is shown that maximum 4 terms are sufficient for the $H = 2$ case (where H is the ratio of cylindrical height to radius) in calculation of measurements (gradients) considering 1% error for the remaining terms. The truncation point is greater when H increases compared to the radius and lower for smaller H . In addition, the result of the time performance analysis is provided which quantifies the improved performance of the algorithm. Finally, the structure of the EIDORS project was studied and the developed complementary modules were contributed to the EIDORS open source project.

Chapter 2

Background

This chapter includes a brief overview on the EIT problem and the "2.5D" literature. The first part of this chapter starts with the definition of tomography, computed tomography and EIT. Afterwards, some advantages and disadvantages of EIT are described. The following subjects are also briefly reviewed toward the end of the first part: current pattern stimulation, forward problem and forward equation, finite element method, inverse problem, Jacobian or Sensitivity matrix, solving the inverse problem, static EIT and difference EIT. The second part of the chapter, which starts from section 2.9, provides a review on the uses of the term "2.5D" in different literature where it is noted how this term is used for different purposes by authors.

2.1 Tomography

By definition, tomography is imaging by sections or sectioning, through the use of waves of energy [11]. Tomography is the reconstruction of an image from the scattering data associated with cross-sectional scans of an object. The word was derived from the Greek word 'tomos' meaning "a section", "a slice" or "a cutting". The device employed in tomography is called 'tomograph' and the image created is called 'tomogram'. Tomography is widely applied in radiology, archaeology, biology, geophysics,

oceanography, materials science, astrophysics and other sciences [12].

2.1.1 Computed Tomography

Computed tomography (CT) is a medical imaging method which uses tomography and creates images by computer processing [13]. In three-dimensional CT, digital geometry processing is employed to produce a three-dimensional image of the internal structures of a body from a large series of two-dimensional images taken around a single axis of rotation [14]. In two-dimensional CT, a similar process is used to generate a two-dimensional image of the internal structures of a body from a large series of one-dimensional images (projections) taken around a single axis of rotation. CT images are reconstructed employing one of a number of numerical techniques known as tomosynthesis. In a CT scan, an image of a target is reconstructed by emitting x-rays from a source and collecting corresponding measurements on the other side of the target [15]. Although the term CT might be known for X-ray CT scans, the term 'Computed Tomography' is also employed for other tomographic problems that include computation similar to the X-ray CT such as single positron emission computed tomography (SPECT) or some EIT algorithms.

2.2 Inverse Problems

Inverse problems techniques are used to solve problems in image processing, medical imaging, geophysics, remote sensing, nondestructive testing, ocean acoustic tomography, astronomy and many other fields in which it is needed to reconstruct a map of unknown parameters based on measurements. According to the Hadamard (inverse) definition of inverse problem, a problem is not an inverse problem if all of the following conditions are satisfied: 1) a solution exists 2) the solution is unique 3) the solution depends continuously on the data [9].

2.3 What is EIT?

There are various medical imaging techniques for investigating inside of a human body. Some techniques are free of risk while others might have the risk of radiation exposure. 'Electrical Impedance Tomography', abbreviated by EIT, is a relatively new medical imaging modality and risk-free technique which has been under development from the mid 1980's. The field was previously referred to as 'Current Computed Tomography' and is divided into 'Electrical Resistance Tomography (ERT)' and 'Electrical Capacitance Tomography' in industrial applications. Generally, Impedance Tomography Imaging is a technique for reconstructing 3D image of electrical conductivity by applying small alternating currents to a body. The field has several Industrial and Geophysics applications as well as some proposed application in Medical Imaging.

The goal of EIT techniques is to reconstruct image of the conductivity, or permittivity, of an internal structure, specific organ, or part of the body. For image reconstruction in EIT, an unknown impedance or conductivity distribution is recovered from electrical measurements captured by an array of electrodes installed on the medium boundary. In contrast to hard-field tomography in which a beam of energy passes through a medium in a direct path, EIT is considered as a soft-field tomography due to the fact that currents scatter in a 3D domain even if the electrodes are all located in a plane. Some authors [16], do not consider the EIT problem as a tomographic problem due to the current tendency to propagate in a 3D pattern. Hence, a good imaging technique should not be only achieved by slice by slice imaging of a cross-section like a 3D CT algorithm.

Due to recent advancements in computations, EIT is now being employed in biomedical applications and is used to monitor flow profiles in pipes as well as blood in vessels. These applications also include the measurement of fluid distribution in mixing vessels, and non-destructive testing such as crack detection. In an EIT system,

the measured electrical potentials, captured by the electrodes attached to the skin (medium surface) of the body, are analyzed usually using finite element modelling and mathematical techniques for inverse problems. Since this technology is under development, early adoptions of EIT were not used routinely; however, there are some proposed clinical application for EIT such as monitoring of lung function, detection of cancer in the skin and breast and location of epileptic foci [9].

Most EIT equipment employs alternating currents (AC); hence, different loads under analysis could have reactive components. The reactive component is often ignored for reducing the complexity of the equipment and the algorithms. However, the advantages of using alternative current are: reduction of electrode corrosion through electrolytic effects, possibility to extract the injected signal from the electrodes and filtering out other signals such as the cardiac cycle, and meeting the requirement of safety standards in medical applications [10].

2.4 Why EIT is difficult?

From a mathematical point of view, reconstructing image of conductivity from surface measurements of current or potential is a severely ill-conditioned, ill-posed and non-linear inverse problem. Hence, EIT is inherently a difficult problem because it is also a difficult problem to recover a signal from a noisy signal either as a forward problem or as an inverse problem. Another problem with EIT is the non-local property of the current which is still valid at the moderate frequencies used in EIT. According to this property, the current scatters while it is passing through the object. The problem of nonlinearity in EIT and the nonlinear inverse problem issues are much more challenging in absolute imaging or static imaging. Static imaging is explained in section 2.8.

According to Hadamard Criteria, a mathematical model of a physical problem is

well-posed if the three criteria mentioned below are held [9]:

1. Solution existence
2. Solution uniqueness
3. Solution stability and continuous dependency of the solution on the data

Since the problem of EIT is to reconstruct the map of conductivity from the boundary measurements, there are normally large more unknown parameters than the available measurement data to solve the problem. Therefore, either a solution does not exist (hence, the error must be minimized in some sense), or there is not a unique solution. In both cases the problem must be regularized in order to find a practical solution by regulating a regularization parameter well known as the 'hyper parameter'. This kind of problems is often referred to as an ill-posed problem. In addition, the regularization gets even more difficult when the solution is unstable which means that any small change in the measured data would result in large change in the solution. This kind of rank deficient inverse problem is more difficult since the singular values of the modelling matrix shows a soft continuity and therefore it is difficult to find a truncation point in singular value decomposition for the purpose of the regularization (condition number is high). In order to regularize this problem further assumptions and constraints must be applied. Such constraints can be the minimum length solution, the minimum error with respect to *a priori* solution or smoothness of the solution.

2.5 Advantages/Disadvantages of EIT

EIT is an inexpensive and relatively low cost imaging modality. In order to keep this as an advantage of an EIT system compared to the other imaging modalities for a specific application, it is assumed that the EIT algorithm can be performed

on a simple and portable computer. Therefore, the EIT interface device must also be portable. Another advantage of using EIT is that it is a safe imaging modality since it is radiation-free. However, the total amount of current injected to, or power dissipated in, the human body must be limited in an EIT system under the electrical safety standards. The amplitude of the current injected to the body should also be under a predetermined threshold in order to have measurements that are not sensed by the human body and the injected current does not interfere with biopotentials. This constraint does not allow to increase the signal to noise ratio by increasing the amplitude of the injected signal which makes the noise component to be dominant when the amount of the signal is near zero. Another main important property of the EIT systems is its imaging speed. In order to keep EIT a fast imaging modality it is necessary to use simple algorithm. Due to this fact most of the EIT image reconstruction methods have been based on 2D algorithms before. In recent years, great efforts have been made to design and implement less cumbersome 3D algorithms which can be applied in a practical real-time monitoring system. In [17], the drawbacks of some of the conventional imaging modalities for the problem of chest and ventilation monitoring are discussed.

On the other hand, the main disadvantage of EIT systems is its poor spatial resolution specially compared with X-ray or MRI images. The resolution of EIT is reported as 10% or at most 15% of the electrode dimensions [9]; although it offers a good time resolution (25 or 50 images per second [9] depending on the EIT system, or even in recent literature it has been reported as 1000 images per seconds [18]). Therefore, efforts for developing this technology are faced toward clinical applications especially long-time monitoring of the status of the patients, for example monitoring of the ventilation or perfusion in lungs. Another issue of the EIT is the presence of artifacts due to model errors in the models that have been used in EIT algorithms. There are various sources of artifacts such as electrode movement due to chest expansion while

breathing, susceptibility to noise and electrode errors, and large variability of images between subjects [19].

2.6 Current Pattern Stimulation

There are two general approaches for injecting the electrical current in EIT. The first approach is to simultaneously apply electrical current into all electrodes, and measure the voltages of the electrodes at the same time [20]. This method has the advantage of higher signal-to-noise ratio due to higher current densities inside the thorax; however, the disadvantage is the problem of skin-electrode contact impedance and dealing with active electrodes. Multiple drive systems are also more complex and expensive. The second approach is to inject electrical current at one pair of electrodes at each time, and measure the voltages at the remaining electrodes. This can be done using the different approaches described below:

2.6.1 Adjacent Current Pattern

In most of the common designs electrical current is injected to adjacent electrodes; yet there are many other possible designs. In the adjacent current pattern, the current is injected through two adjacent (or neighboring) electrodes and the voltages are measured from successive pairs of adjacent electrodes. Then the current is injected through the next pair of electrodes and the voltage measurements are repeated. The procedure continues until the current is injected from each possible pair of adjacent electrodes. There are $L \times (L - 3)$ voltage measurements available (L is the number of electrodes) at the end of the full cycle. Although only half of these measurements are independent, normally all of these measurements are fed into the reconstruction algorithm procedure. One advantage of this approach is that it is not needed to account for skin-electrode contact impedances since all voltage measurements occur

at near-zero current. In addition, the adjacent current pattern requires minimal hardware to implement. In this method, the current density decreases rapidly as we get far from the current drive electrodes; hence, it is highly sensitive to conductivity changes near the boundary and less sensitive to central contrasts. This method is also sensitive to perturbations in the boundary geometry of the object, in the positioning of the electrodes and is quite sensitive to measurement error and noise [21]. This approach has been used in most of the systems employed for lung function studies.

2.6.2 Opposite Current Pattern

In addition to the adjacent current pattern, there are other possibilities for current injection. One of the proposed alternatives is the opposite or polar drive pattern, which is commonly used in brain EIT [22], [23]. In this method, current is injected through electrodes that are diametrically opposite and difference voltages are measured on the remaining electrodes. There are various ways to collect the voltage measurements. The opposite current pattern will provide $L - 4$ voltage measurements per injection with an L electrode system. The next set of $L - 4$ voltage is measured by shifting the current injecting pair. The opposite strategy has the disadvantage of having fewer available measurements ($L \times (L - 4)$) than the adjacent strategy for the same number of electrodes which leads to having less known parameters. On the other hand, an advantage of the opposite current drive is that it is more sensitive to conductivity changes at the center compared to the adjacent strategy. This is due to the fact that the opposite method provides a better distribution of the sensitivity because the current travels with greater uniformity through the body. It is reported that the opposite current strategy optimizes the sensitivity for a contrast in the center of the body [24].

2.7 Forward Problem

In order to solve the inverse problem of EIT, a forward model is developed to calculate the voltages from the injected current for a known conductivity distribution. This could be used to predict the observed voltage given the estimated conductivity and compare the voltage with the measured data in an iterative approach for finding the unknown conductivity. In addition, the forward model provides the capability to calculate the interior electric fields to construct the Jacobian matrix using the adjacent field method. The Jacobian matrix is a matrix developed in the inverse modelling for EIT in order to solve the unknown conductivity given measured voltage and is necessary to solve the inverse problem expressed in more details in section 2.8.

2.7.1 Forward Equation

In this section, the physics of the forward problem are described. It is explained how the Laplace equation is derived from the Maxwell's laws. For low frequency EIT systems (typically 50 kHz), which are commonly used, the problem is approximated to be quasi-static; hence, the electrostatic forms of Maxwell's laws are normally employed in EIT. In this case the magnetic field is neglected; however, at high frequencies, magnetic effects cannot be ignored [25]. The Laplace equation is a general equation in electrostatic problems which is derived for the electric potential. It starts from the Kirchhoff's current law (KCL) written in the point form for current density vector \vec{J} as

$$\nabla \cdot \vec{J} = 0 \quad \text{in } \Omega \quad (2.1)$$

This equation expresses that in the absence of independent electric charges, the summation of outward and inward current at any point or any closed surface inside the domain Ω is zero. The point form of the Ohm's Law provides the relation between

the Electric field vector, \vec{E} , and the current density as:

$$\vec{J} = \sigma \vec{E} \quad (2.2)$$

where σ represents the conductivity. With quasi-static assumption in force, the Electric field would be curl-free; hence it is possible to write the electric field in the form of the gradient of an scalar potential as:

$$\vec{E} = -\nabla V \quad (2.3)$$

where V is the electric field potential. Finally, substituting the Ohm's law (equation 2.2) into the KCL (equation 2.1) and replacing the electric field with the gradient of the electric potential yields:

$$\nabla \cdot (\sigma \nabla V) = 0 \quad \text{in } \Omega \quad (2.4)$$

which is known as the Laplace equation. The boundary condition would depend on the type of the boundary method used for the EIT problem. In general, the current density at the boundary would have the following relation with the electric potential.

$$j = -\vec{J} \cdot \hat{n} = \sigma \nabla V \cdot \hat{n} = \sigma \frac{\partial V}{\partial \hat{n}} \quad \text{in } \partial\Omega \quad (2.5)$$

where $\partial\Omega$ represents the boundary of the domain. From the Gauss theorem, conservation of current applies in the absence of independent source of charge; hence, the boundary current density must satisfy the consistency condition: $\int_{\partial\Omega} j = 0$.

Normally in EIT, electrical energy is applied to the medium in the form of current injection on the boundary. The injected current density at the boundary produces

a distribution of voltage and a pattern of current flow inside the medium. Specification of boundary current density, j , results in the Neumann boundary conditions (Type II or derivative boundary condition). Specification of V at the boundary condition results in the Dirichlet boundary conditions (Type I or essential boundary conditions). Depending of the type of the electrode model employed, the boundary condition could be a mixture of the Dirichlet and Neumann boundary conditions (Type III or mixed) [10]. Regardless of the type of the boundary conditions, by solving the Laplace equation, the potential V is determined up to an additive constant; this is equivalent to choosing an earth (ground) point [9]. The Laplace equation forms the forward problem of the EIT which is solved for an arbitrary geometry with non-uniform conductivity by employing numerical techniques such as Finite Element Method.

2.7.2 Finite Element Method

In this section an introduction to finite element method is provided. The details of the finite element modelling is explained in the next chapter for the $2\frac{1}{2}$ D case. Finite Element Method (FEM) is a numerical technique for solving partial differential equations (PDE). In many situations, the geometry of the problem or even the type of the PDE does not permit an analytic solution in a closed form. FEM was initially developed for aircraft design [26]; yet, later it was extended for modelling of electromagnetic and electrostatic fields as well as many other engineering problems.

The FEM is the most common method currently used for the numerical solution of EIT problems thanks to its ability to model arbitrary geometries and various boundary conditions [27], [28]. The finite element method has also close relatives such as the finite difference method and finite volume method that use regular grids. Compared to the FEM, these two methods have the benefit that more efficient solvers can be employed; on the other hand, they suffer from having difficulty in modelling

curved boundaries or smooth interior structures accurately [9].

In FEM, the region in which the problem is solved is segmented into a finite number of 'Elements'. A collection of these elements is called a 'finite element mesh'. Each element could have different number of nodes. In 1D problems, elements are lines or curves. In 2D problems, elements could be triangles or L-shaped elements, and in 3D problems, elements are polyhedrons such as tetrahedral and hexahedral.

The partial differential equations need to be solved in the whole region; however, in a finite element model, the problem is reduced into a finite number of points called 'nodes' and the solution is interpolated (or approximated) over each element. Hence, selection of interpolation functions (also known as basis or shape functions) is also required.

The finite element method includes the following procedures:

1. Meshing: The region is segmented into non-overlapping subregions called 'Elements' connected via 'Nodes'. This part could be performed using one or a collection of different element types or meshing techniques.
2. Modelling: The partial differential equation is transformed into a Matrix Equation for computing the solution for nodes by substituting the interpolation function into the PDE and using finite element techniques. This part is the heart of finite element method where it might be required to derive a weak form of the PDE to be solved instead.
3. Solver: The Matrix Equation derived from the previous section is solved by inversion to calculate the solution at the nodes.
4. Solution: The solution for the whole mesh is approximated using the interpolation function and the desired parameters or variables, derived from the solution, are computed based on the solution in the entire domain. Additional computations might also be required for iterative algorithms, that employ adaptive

currents and adaptive mesh refinement [10].

There are three different Finite Element techniques typically used to model (or formulate) finite element problems [29]: (1) Direct approach which is the most intuitive way to understand the finite element method, (2) Variational approach which is based on the calculus of variation and used for problems which includes elements with non-constant conductivity or use higher order interpolation functions and for element shapes other than triangles and tetrahedrons (simplices), and (3) Method of Weighted Residuals (MWR) which is the most versatile approach in deriving element properties. This approach models the problem without using a variational statement and explained in chapter 3.

The modelling part of the finite element method addressed above could be described as the following sub-procedures:

- a. Choosing Interpolation Functions: The value of the function inside each element is approximated from the values at the nodes by the interpolation functions which can be any piecewise polynomial of fixed order. Most commonly, linear interpolation functions are employed due to the simplicity.
- b. Calculating the Local Stiffness Matrix: The local stiffness matrix is the matrix of element properties calculated for each element.
- c. Calculating the System Matrix: The global system (or stiffness) matrix is a combination of all local matrices into a single global matrix for the entire mesh considering the element connections (connectivity map).
- d. Imposing Boundary Conditions: Depending on the type of the boundary condition used, the system matrix is updated or extended appropriately. A Matrix system of equation is finally developed based on the known variable (here, current) for solving the unknowns (here, voltage).

As the order of the polynomial is increased or as the elements become more numerous (provided their interior angles remain bounded), the finite element method converges to the solution or at least the solution of the weak form of the PDE used to represent it [30]. Another method similar to the FEM is the boundary element method (BEM) in which only surfaces of regions are segmented. In BEM, an analytical expression for the Green function is employed within enclosed volumes that are assumed to be homogeneous. BEM is useful for EIT forward modelling provided the conductivity in regions with smooth boundaries (e.g. organs) is assumed to be piecewise constant. However, compared to the FEM, BEM results in a dense rather than a sparse linear system to be solved. Hence, computational advantage of BEM over FEM reduces when the number of regions in the model increases. On the other hand, the advantage of BEM is that it makes it possible to represent unbounded domains. Hence, a hybrid method could be computationally efficient for some applications of EIT where BEM is used to represent homogeneously assumed regions and FEM is used to represent inhomogeneous regions [31] [9].

2.8 Inverse Problem

In this section it is explained how the inverse problem is formulated to reconstruct the conductivity distribution. The inverse problem is also called 'reconstruction'. Reconstruction is the process of recovering an image from the EIT signal. According to [10], reconstruction algorithms are classified into several categories. Each of these categories is developed for imaging a different aspect of impedance:

1. Time difference imaging: This class of imaging algorithm is employed to recover the image of the impedance change over time.
2. Static (or absolute) imaging: This class of imaging systems is employed to

construct the image of the absolute impedance distribution.

3. Multiple frequency imaging: This class of imaging systems is employed to recover the image of impedance changes by frequency.
4. Dynamic imaging: This class of imaging systems is employed to recover fast conductivity changes. In dynamic EIT systems, it is assumed that the conductivity changes rapidly with respect to the acquisition interval between each measurement cycle; yet slow enough compared to the acquisition period of a frame of data [32].

In both static and difference image reconstructions, it is desired to estimate the conductivity (or conductivity change), \hat{x} , from the measured boundary voltages, z ; hence, the reconstruction problem could be modelled as:

$$\hat{x} = Bz \tag{2.6}$$

where the goal is to find matrix B. In difference imaging, a data set v_{t_1} is measured at a time t_1 labeled as 'reference' and another data set v_{t_2} is measured at a later time t_2 labeled as 'data'; hence, z is the difference between measured value at time t_1 and t_2 , i.e., $z = \Delta v = v_{t_2} - v_{t_1}$. The goal of the reconstruction is to calculate the change in conductivity from time t_1 to time t_2 ; hence, $\hat{x} = \Delta\sigma = \sigma_{t_2} - \sigma_{t_1}$. In difference imaging, σ_{t_1} is not known; hence, \hat{x} is interpreted as the change in conductivity with respect to the unknown initial conductivity. On the other hand, in static imaging, the goal is to recover the conductivity distribution assuming a pre-assumed initial conductivity by updating the conductivity distribution iteratively from the initial conductivity.

2.8.1 Jacobian or Sensitivity Matrix

In EIT problem, even large changes in the medium conductivity can result in small changes in measurements compared to the measurements of the background conductivity distribution. Therefore, the forward problem can be divided into a static and a dynamic component as equation 2.7, where the dynamic component is assumed to be a linear function of the conductivity change.

$$v_h + \Delta v = F(\sigma_h) + \mathbf{J}\Delta\sigma \quad (2.7)$$

In this equation, v_h is the voltage measurements from a homogeneous medium, $F(\sigma_h)$ is a function of the homogeneous conductivity distribution, $\Delta\sigma$ is the change in conductivity distribution, Δv is the change in measurements, and \mathbf{J} is the so called "Jacobian matrix" which describes this linear relation. This linear approximation is only valid for a limited range. Hence, the inverse problem is linearly modelled as (2.8) given a small change in conductivity $x = \Delta\sigma$. For large conductivity changes a non-linear model might be more accurate which leads into iterative approaches for solving the inverse problem.

$$z = \mathbf{J}x + n \quad (2.8)$$

In this equation, \mathbf{J} is the Jacobian matrix and n is the noise model that appears in the measurements which is considered an uncorrelated additive white Gaussian Noise(AWGN); x is the conductivity change ($\Delta\sigma$) and z is the difference measurement (Δv). Matrix \mathbf{J} is the matrix of partial derivatives of voltages with respect to conductivity parameters. In the medical and industrial EIT literature, \mathbf{J} is also called the 'Sensitivity matrix' and its rows are called sensitivity maps. The sensitivity

matrix could be derived from the system matrix by the following relation:

$$\mathbf{J} = T \left[-\frac{\partial}{\partial \sigma} S^{-1}(\sigma) I \right] = T \left[S^{-1}(\sigma) \frac{\partial}{\partial \sigma} S(\sigma) S^{-1}(\sigma) I \right] \quad (2.9)$$

where 'T' represents an operator for extracting measurements from the nodal voltages, S is the systems matrix developed in the forward model (see chapter 3), and I is the injected current. The term $S^{-1}(\sigma)I$ is the voltage of the nodes which is the solution of the forward problem. The details of derivation of the sensitivity matrix can be found in EIT texts such as [9] or [10]. In a model with K elements and b boundary measurements the Jacobian is a $K \times b$ matrix. The Jacobian matrix is calculated column by column since each column represents an element. Hence, the i^{th} column represents the effect of the conductivity change in the j^{th} element on the measurement voltage, v_i , measured between electrode pairs. Each element of \mathbf{J} relates the conductivity change in an element to a measured voltage at a background conductivity σ_0 as:

$$J_{ij} = \left. \frac{\partial v_i}{\partial \sigma_j} \right|_{\sigma_0} \quad (2.10)$$

where, here, v_i is the i^{th} measurement voltage. The sensitivity matrix, \mathbf{J} , is a function of the finite element model, the background conductivity and the current injection pattern. Normally, a the background conductivity is assumed to be homogenous for each of the elements; i.e. $\sigma_0 = 1$.

2.8.2 Solving the Inverse Problem

The sensitivity matrix, \mathbf{J} , is not a square matrix. Hence, unlike the forward problem, solving the linearized inverse problem in equation (2.8) is not straight forward. One approach is the least squares solution in order to minimize $\|\mathbf{J}\mathbf{x} - \mathbf{z}\|$. Using this

approach, the solution would be calculated by:

$$\hat{x} = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T z \quad (2.11)$$

However, here, the problem in equation (2.8) is ill-posed since normally there are more unknown conductivities than the known measurements; therefore, matrix $\mathbf{J}^T \mathbf{J}$ is rank deficient and thus cannot be inverted. In ill-conditioned problems, a small perturbation in the input data results into a large change in the output; hence, even if the solution exists and is unique, it can be completely corrupted by a small error in the data or by noise. Employing regularization methods, the conductivity, \hat{x} could be reconstructed by the generalized regularization formulation:

$$\hat{x} = (\mathbf{J}^T \mathbf{J} + \lambda^2 \mathbf{R}^T \mathbf{R})^{-1} \mathbf{J}^T z = Bz \quad (2.12)$$

where λ is called a regularization parameter or 'hyper-parameter' and matrix \mathbf{R} is called the regularization matrix. In the simplest form, the regularization matrix is the identity matrix \mathbf{I} , i.e. $\mathbf{R} = \mathbf{I}$, which reduces the regularization problem to the Tikhonov regularization. The identity matrix and the matrices corresponding to the first and second difference operators are the most common regularization matrices employed in EIT [33]. The regularization aims to solve the following minimization problem instead of the least square minimization:

$$\hat{x} = \arg \min_x \{ \|\mathbf{J}x - z\|^2 + \lambda^2 \|\mathbf{R}x\|^2 \} \quad (2.13)$$

where $\lambda^2 \|\mathbf{R}x\|^2$ expresses some prior information about the conductivity and reduces the ill-conditioning of the problem by adding more constraints. From another point of view, for the Tikhonov Regularization case, λ is filtering the singular values of $\mathbf{J}^T \mathbf{J}$ such that the effect of the singular values smaller than λ in dominating the solution

is bounded. It acts like a smooth truncation of small singular values. More details about regularization methods, generalized singular value decomposition and inverse problem technique are available in [34], [35] and [10].

2.8.3 Static EIT

For static conductivity construction, a set of measurement voltages is simulated by employing the forward model given an initial estimation of the conductivity σ_0 . The simulated set of measurement is used as a reference and the actual measurement is considered as data. Then, having difference measurement, the difference conductivity is calculated the same way as it was mentioned for the difference imaging. The resultant conductivity change, $\Delta\sigma$ is used to update the initial estimate in an iterative approach, i.e., $\sigma_k = \sigma_{k-1} + \Delta\sigma$.

In each iteration, the forward equation is solved given the updated conductivity in order to obtain a [simulated] reference set of measurement. Then, the sensitivity matrix is updated for the new conductivity estimate, i.e., $J_{ij} = \frac{\partial v_i}{\partial \sigma_j} |_{\sigma_k}$ and the inverse equation is solved by the updated sensitivity matrix and the new difference measurement between the actual measurements and the updated simulated measurements. This procedure continues until an error criterion is satisfied such as the maximum number of iterations is reached or the absolute difference between the measurements and the simulated measurement gets smaller than a pre-determined error bound parameter.

2.8.4 Difference EIT vs. Static EIT

In static reconstruction, an image of the distribution of the conductivity in a medium is produced based on a single set of data as it was defined before. In contrast, in difference image reconstructions, an image of the change in conductivity distribution

between the acquisitions times of two data sets is produced.

The advantage of using the difference EIT technique is that it reduces the sensitivity to electrode position, electrode movements and electrode error; also it leads to a linear inverse problem which makes this method suitable for real time imaging. It is often known that difference imaging is able to improve stability of the recovered image when the contact impedance is unknown, the boundary geometry is poorly known, the problem is nonlinear, and the 2D approximations is used for 3D electrical fields [36] [16]. Further explanation are provided in [36] in which it is shown that as long as errors in electrode placements remain relatively constant during the measurements, difference imaging is relatively insensitive to error of electrode placements [17].

2.9 The 2.5D literature

In this section, some applications of the 2.5D methods are reviewed. The section starts by definitions of "2.5D" stating that different authors refer to the 2.5D term for different purposes. Then applications of the 2.5D modelling are classified into geophysics, electromagnetic, process tomography and biomedical applications. Some applications in which the 2.5D modelling is used are: imaging of a medical process in reactor-stirred vessel, under semi-batch operation exhibiting imperfect fluid mixing, earth resistivity arrays, imaging of particle shear migration, and ventilation. The section ends by a conclusion on the previous works in 2.5D methods.

2.9.1 What does 2.5D refer to?

The term 2.5D (which is denoted by $2\frac{1}{2}$ D in the following chapters in order to exclude from other definitions) is used for different purposes in different fields of science. In our study, the "2 $\frac{1}{2}$ D finite element method" refers to a method in which the 3D Laplace equation is solved using a set of modified 2D equations employing 2D finite

element mesh under the $2\frac{1}{2}$ D assumptions. The $2\frac{1}{2}$ D (or 2.5D) assumptions means having translationally invariant environment along one axis. Other authors, e.g. [37], might only refer to the "2.5D modelling" as a model in which a stretched form of a 2D model having translation invariance along the translation axis is assumed for 3D modelling. In this case, the 3D mesh is referred as a "2.5D model" due to the fact that it does not have all aspects of a real 3D model such as deformation of the model along the height or variation of conductivity along with the height. Moreover, in some works such as [38], the "2.5D imaging" (or "2.5D reconstruction") refers to a method in which the conductivity is reconstructed at different slices of a 3D model using a 2D image reconstruction algorithm and interpolated under some 2.5D assumptions. This approach is also called "pseudo-3D" method and most often, this specific method of reconstruction is referred to "2.5D". In addition, in [39] the terms "2.5D" and "2.75D" refer to meshing techniques for generating finite element mesh of 3D objects in specific perspectives. In the EIDORS tutorial [40], the " $2\frac{1}{2}$ D" refers to a method where the z-direction is part of the forward model, but not the inverse model. The tutorial describes how this could be performed as an application of coarse/fine mapping in which the a fine (high density) forward model is used with a coarse (low density) inverse model. In other words, here, the " $2\frac{1}{2}$ D" refers to an EIT method which uses 3D solver in forward solver and 2D model in the inverse solver.

In all of the works mentioned above and also other similar works to them, the term 2.5D is used for a different concept than what it is used in our study. In contrast, the 'true', 'real', 'full', or 'complete' 3D term is used in order to emphasis that the method is not a 2.5D algorithm. In summary, in this work, the " $2\frac{1}{2}$ D finite element method" is referred as a solver, called " $2\frac{1}{2}$ D solver", for solving the 3D Laplace equation using a series of modified 2D equations, called " $2\frac{1}{2}$ D equations", for a 2.5D model (3D model assuming translation invariance). The $2\frac{1}{2}$ D finite element method can be applied to any problem where the medium in which the problem is being solved is by its nature

a 2.5D model or can be approximated to a 2.5D model which is a 3D model having transactionally invariant property along one axis.

2.9.2 Geophysics

According to [41], for the 2.5D electromagnetic problem, the first published theoretical finite element derivation was reported in Coggon [42]. The DC resistivity and induced polarization problems have been studied in Coggon [42], Snyder [43], and Fox *et al* [44]. Everett [45] and Moghaddam *et al* [46] studied the 2.5D time domain problem. The former presented a solution for a transient seafloor exploration system and the later provided a solution for ground-penetrating radar.

In [47], the 2.5D resistivity tomographic imaging is employed for modelling of embankment dams to evaluate geometry influence and the effect of material properties. It is noted that for the purpose of monitoring development of internal erosion and "anomalous seepage" in earth embankment dams, "repeated resistivity measurement" is a potentially powerful approach. The study was done to improve long term monitoring methods and data interpretation and in order to provide better understanding in interpreting existing data. The properties of two rock-fill embankment dams in the north of Sweden was used to model various occurrences typical of embankment structure. The study evaluates the influence from 3D effects created by specific dam geometry and effects of water level fluctuations in the reservoir. For this purpose a special 2.5D software was implemented to model apparent resistivity for geometries and material distributions for embankment dams. Here, the 2.5D refers to the $2\frac{1}{2}$ D finite element method.

Zhou [48] has checked the accuracy of 2.5D modelling by comparing it with known analytic solutions. It was demonstrated that the modelling accuracy mainly depends on the element size, electrode spacings (which produce different ranges of the wavenumber), and the wavenumber sampling for accurate inverse-Fourier transform.

In [49], a comparative study was performed to examine the effect of surface topographic variations on several DC earth resistivity arrays. For the purpose of that study, a modified $2\frac{1}{2}$ D finite element method (FEM) modelling scheme was employed in order to evaluate the performance of different survey arrays such as dipole-dipole, Wenner, pole-pole, pole-dipole and Square. It is stated that "local" four-electrode arrays are less sensitive to terrain topography variations with respect to arrays employing remote probes for resistivity profiling applications. In that work, the 2.5D finite element method is modified because it is stated that in modelling arrays parallel to the strike dimension the cosine term disappears. It is stated that by simple integration schemes it is possible to obtain acceptable accuracy; however, the cosine term and the logarithmic discontinuity at $k = 0$ "renders" the integration much more complicated for modelling arrays vertical to the strike direction (e.g. square array).

The modified scheme was proposed to find a solution for the secondary potential by assuming that the total potential is produced from the superposition of a homogeneous background (primary field) and the modelled structure (secondary field). Adding the resultant secondary potential to the primary potential calculated at a small distance from the source makes it possible to avoid the singularity introduced by the source. Finally, a combination of logarithmic and exponential interpolation as proposed in [50] was used for the integration. Although the 2.5D finite element method is similar to the approach used in this study, the boundary geometry and condition is different in the EIT application as a medical imaging modality.

In [51], a 2.5D finite element scheme is employed for the forward modelling in field surveys and the potential variations parallel to the strike direction was calculated by modifying an existing approach. Here, the square array configuration was studied in which three different resistances can be measured directly; like any four-electrode measuring configuration. Employing the square array has the advantage that the resistance values can be employed to gain a measure of the 'apparent anisotropy'

which is called azimuthal inhomogeneity ratio (AIR). The AIR is widely employed to obtain information about the directional variation of the subsurface resistivity.

In [52], the effect of a multi-coil electromagnetic tool offset on the measured signals with respect to the borehole axis is discussed. In this study the EIT method was not used; however, the method employs the finite element method. In the numerical experiment, the 2D and 2.5D modelling methods were used for comparison in which the tests demonstrated a good agreement of the results.

In [41], electromagnetic (EM) induction of a finite electric dipole source over a 2-D earth is modelled using the 2.5D finite element modelling since the excitation source had a 3D structure. Although the concept of the 2.5D FEM is same, the formulation of the problem differs by comparing to the electro-static problem of EIT. In that work, for each set of discrete spatial wavenumbers (harmonic), a finite-element method is employed to solve a solution for the secondary electromagnetic fields. The paper considers some simple seafloor examples in conclusion. In addition, for the finite element mesh, exponential elements are used in order to model the fields at the far-field efficiently. The boundary geometry is theoretically open in the far field which makes a problem that need to be tackled too.

Complex Electric Impedance Tomography in Geoelectrics (CEITiG) is a MATLAB based software for geological applications that has also implemented the 2.5D finite element method. One of the advantages of this software is that it provides graphical user interface GUI features. It is a part of a research project to increase the resolution power and acceleration of tomographic reconstruction methods in the geoelectric.

2.9.3 Miscellaneous Non-tomographic Problems

In [53], a 2.5D finite element modelling difference method is developed for EM modelling in stratified anisotropic media for marine controlled-source electromagnetic (CSEM) applications. In order to model the seabed logging problem, finite element

model is used in axis-aligned anisotropic media for a 2.5D problem. Here, a 2.5D problem is referred to a problem in which the energy is radiated from a 3D source in the presence of 2D structures. In implementing the method, the delta sources were employed to solve the set of governing partial differential equation. It is stated that the formulation of the 2.5D finite-difference method was first proposed by Stoyer and Greenfield [54] for a two-dimensional earth to an oscillating magnetic dipole source.

In [55] and [56], an integral equation framework is presented for 2.5D frequency domain EM modelling in conductive media. Here, 2.5D is referred to a medium which is assumed to be invariant in one direction. Moreover, a media having transverse isotropic properties in the vertical direction (TIV) is also considered; this makes possible to have different horizontal and vertical conductivities. The method was applied to marine Controlled Source ElectroMagnetic (mCSEM) surveys. It is stated that a hydrocarbon reservoir will act as a waveguide for the low-frequency electromagnetic signal because the reservoir typically have a low electric conductivity with respect to its surroundings in the subsurface.

It has been demonstrated that the 2.5D framework has the capability of simulating mCSEM experiments with respect to the model size and conductivity ranges by only solving between 20 to 30 2D problems for having reasonable results. It is stated that for that problem, this proves the computational efficiency of the 2.5D method with respect to solving the full 3D problem.

In [57], 2.5D cavity balancing is proposed which employs 2.5D idea for mesh generation. The process of altering the flow front inside a cavity by the means of thickness and design changes in order to obtain the desired fill pattern is known as cavity balancing. Since most or (all) of the practical injections parts are not 2D parts, a 2.5D cavity balancing optimization routine was developed for a 2.5D finite elements domain. Plastic injection-molded parts are not flat typically; however, since they are mostly thin, it is possible to approximate them as 2.5D. Therefore, the physical

domain is approximated by faceted objects for the 2.5D finite element simulation which simplifies the 2.5D problem into a series of 2D problems.

2.9.4 Process Tomography

In [58], 2.5D linear back projection (LBP) is considered in comparison with the 2D and 3D LBP. In that work, the 2D and 2.5D LBP data were acquired from the ITS P2000 standard software, ERTWin [59]. The ITS P2000 has been chosen for the experiments done for process tomography [60] which is claimed to be the best performing EIT instrument, available, for experiments requiring high temporal resolution and is capable of successfully monitoring homogeneity [61]. LBP is the method normally used in (X-ray) CT scans where the beams travel in a direct path inside the medium. In LBP, the sensitivity matrix and the forward model is defined differently than the Finite Element approach. LBP is not a popular method for EIT since normally EIT is considered as a soft tomographic problem where the current is scattered in the medium.

In [60], the 3D EIT is employed to monitor reactive systems under realistic process conditions. The purpose of the study was to examine the semi-batch precipitation of barium sulphate from barium chloride and sodium sulphate in a "200l, semi-tech scale, stirred tank reactor". For this purpose, some experiments were carried out under a range of agitation rates with a surface feed of barium chloride employing both Rushton and pitched blade turbines. EIT's ability to visualize the plume structure and evolution was investigated during the semi-batch reaction in 3D and linked to the mixing curve. The study was performed for a real 3D case and was compared to the pseudo-3D reconstruction implemented earlier.

In [62], the $2\frac{1}{2}$ D finite element method is applied to investigate the net migration of particles in a suspension undergoing pressure-driven flow through a tube at low

Reynolds number. EIT is used to the flowing suspension by two flush-mounted electrodes on the pipe wall. In that work, an image of the suspension particle volume fraction was reconstructed using the relationship between the local conductivity and local suspension concentration. The result of particle distribution measured by the EIT approach was claimed to be in very good agreement with magnetic resonance imaging. The work employs the $2\frac{1}{2}$ D finite element method where it is claimed that ten terms of the series are sufficient to determine the solution for an infinitely long cylinder when the periodicity of the solution, $2z_m$, is set equal to eight times the diameter of the pipe.

The pseudo three-dimensional EIT (which is often referred to 2.5D EIT) has been often utilized for applications of EIT in fluid mixing in a stirred vessel. In pseudo three-dimensional EIT, two-dimensional slices are interpolated into a three-dimensional reconstruction. Some articles found in the literature are [63], [64], and [65].

In [37], a real 3D image reconstruction algorithm is implemented for spatial imaging where sensors provide 3D measurements. The results were compared to the 2D and 2.5D method. In that work, the 2.5D tomography is referred to a method which uses a few independent 2D images to interpolate them into a 3D image; therefore, 3D visualization can be achieved by interpolating series of 2D slices from different planes into a 3D image as in many medical applications. The paper concentrates on the capacitance imaging (ECT). The 3D sensor configuration is referred to a sensor configuration where electrodes are placed in multi-layers (slices) on the boundary and not only on a single layer. It is concluded that employing present computing power does not allow us to run on-line 3D imaging for monitoring applications due to the significant calculation complexity of the 3D reconstruction algorithms; hence, 3D imaging could be mostly applied to off-line applications.

2.9.5 Biomedical Applications

The work in [3] is one of the earliest work where a finite element algorithm is developed for EIT assuming translationally uniform conductivity distributions for finite cylinders with general cross-sectional boundaries. The 3D forward problem was solved by a series of 2D problems by employing a finite element mesh considering only the point electrode model (PEM). For image reconstruction purpose, the ratios of the gradients computed for both 2D and 3D homogeneous objects are calculated given a cross-sectional boundary; next, the real measurements from the 3D object are multiplied by these ratios and the conductivity image is reconstructed by the 2D iterative equipotential lines method. It is argued the method provide an "equivalent" translationally uniform object which produces the same gradients as the real measurements. It is reported that the maximum relative difference of the solutions from with the cylindrical harmonics and FEM is less than 2%.

In [5], a $2\frac{1}{2}$ D finite element forward simulation is employed for EIT because of its much shorter calculation times. It is stated that resistivities are constant in cranio-caudal direction; hence, the $2\frac{1}{2}$ D approach was employed for the 2D model parametrization of a 3D measurement. In this study, the point electrode model was employed where the electrodes were located in one plane of the body. The modelling technique described in that work was originally proposed by Dey and Morrison [66] for geophysical application and followed recently in Rucker *et al* [67]. For testing purpose, a complete 3D forward simulation with bounded lungs and heart was used. It was stated that there was no significant improvement in the results of 3D for functional data sets.

In [68], a measurement system and image reconstruction was presented for magnetic induction tomography employing 2.5D finite element modelling. Magnetic induction tomography (MIT) is an imaging modality similar to EIT which is a non-invasive yet non-contact imaging technique used for biomedical applications. In MIT, instead of current injection electrodes, current-carrying coils are applied to induce eddy currents in the object and the induced voltages are measured with other coils. That work employs the finite element method using edge elements for solving the equation and a generalized Tikhonov regularization approach for image reconstruction. The work considers and tests both static and difference imaging. In both cases, the 2.5D reconstruction method is used where the Jacobian is calculated assuming that the conductivity distribution is translationally invariant in the direction of the z-axis.

For the MIT, it has been reported that the "2.5D reconstruction" approach works well in a low conducting test case with real measurements. However, some artifacts in difference imaging and fairly poor resolution in static imaging can be observed. Two full 3D cases were compared to the 2.5D image by extended simulation from the same author in [55] and [56]; the first 3D case uses one ring of 16 coils and the other uses an increased number of measured data points obtained from 3 ring of 16 coils in different vertical positions. The author recommends that the 3D image reconstruction should be employed especially for static imaging, due to the fact that the 2.5D assumption is rarely valid in real imaging situations. Some reasons were given for artifacts in static imaging in MIT. The major reason is the use of a 2.5D image reconstruction method for the simulated 3D case. Another reason for artifacts was truncation of the static imaging iterative approach in the second step (one step after homogeneous distribution). Another source of error comes from the fact that for the proposed method piecewise constant conductivity was modelled in the forward problems while piecewise linear presentation was modelled for the inverse problem. In addition, it

is stated that when comparing the simulation results to the real measurement cases, it could be seen that the real measurements provide better results; this happens due to the fact that the resulted negative values were truncated to zero after the reconstruction.

The work presented in [37] aims to study different effects of different thorax models with either conductivity or complex conductivity on the reconstructed images and ventilation indices. For this purpose, true 3D thorax models with conductivity distribution or complex conductivity distribution under different ARDS conditions are developed and the solutions are compared to the "2.5D thorax" model. Here, the 2.5D is referred to the mesh style used for modelling the thorax in which the 3D model is developed through stretching from the 2D slices; which results in a simple case a circular, elliptical, or arbitrary-base cylinder. ARDS refers to acute respiratory distress syndrome which is a life-threatening state of the lung specified by lung collapse and water in the lung which makes inhomogeneous variations in the conductivity distribution inside of the thorax.

2.9.6 Conclusion

It was shown that the "2.5D" term is used for different purposes by different authors. In our study, the " $2\frac{1}{2}$ D" refers to the $2\frac{1}{2}$ D finite element modelling where a 3D Laplace equation is solved by a set of 2D-like equations under a $2\frac{1}{2}$ D assumption. Those works that are similar to our work could be classified into geophysical, process tomography and biomedical application. It was shown that only a few authors have used $2\frac{1}{2}$ D finite element modelling for EIT. In some geophysical applications together with EM problems, a non tomographic approach is employed where the type of energy is electromagnetic waves. In this kind of problems, a 3D set of electromagnetic equations (derived from the Maxwell's equations) is solved in which both electric field and magnetic field are involved having curl equations. In other geophysical applications where

EIT is used, the type of boundary condition and boundary geometry as well as the geometry of the problem is different than the EIT problem in biomedical applications.

The use of $2\frac{1}{2}$ D finite element modelling for the EIT problem is quite similar for process tomography and the biomedical applications in which an electro-static equation is solved having divergence rather than curl; the situation is similar for magnetic induction tomography MIT. It was found that, there are more works performed in process tomography using $2\frac{1}{2}$ D finite element modelling comparing to the biomedical applications. The use of $2\frac{1}{2}$ D finite element modelling for the biomedical applications is still under investigations since the $2\frac{1}{2}$ D approach might not be sufficient in complex areas where the true 3D nature of the experiment is significant. However, in real time application, the real 3D algorithm is not computationally efficient; thus still 2D methods are employed. All of the works currently found in EIT either for process tomography or in biomedical field continue to employ the point electrode model which is simply modelling a single node for each electrode. The complete electrode model seems to be necessary especially in biomedical applications where skin-electrode contact impedance is not negligible.

Chapter 3

$2\frac{1}{2}$ D Finite Element Method

In this chapter, the $2\frac{1}{2}$ D Finite Element method is formulated. The chapter starts with a definition of the $2\frac{1}{2}$ D problem as well as the $2\frac{1}{2}$ D assumptions. In the first section, a set of partial differential equations is derived from the 3D Laplace equation under $2\frac{1}{2}$ D assumptions followed by the correct boundary condition in the second section. The Method of Weighted Residuals (MWR) is explained in section 3.3 where in a step by step description, it is shown how the finite element method models each $2\frac{1}{2}$ D PDE into a matrix equation.

Next, using linear interpolation functions, the integral terms resulting from the previous section are solved analytically. The closed form of the analytical solution of the integrals are provided in section 3.3.4 and complete details for deriving the closed forms are explained in appendix A. In section 3.4, the complete electrode model is applied to the right side of the governing equation for the $2\frac{1}{2}$ D case; and in section 3.5 the modification of the inverse problem is explained.

The second part of this chapter (which starts from 3.6) contains the construction process of the global system matrix in a step by step description through some examples. In section 3.6 for a simple 2D mesh, the process of calculating the connectivity matrix and local stiffness matrix are explained followed by the construction of the entire system matrix. In addition, the values of the system matrix are provided for the

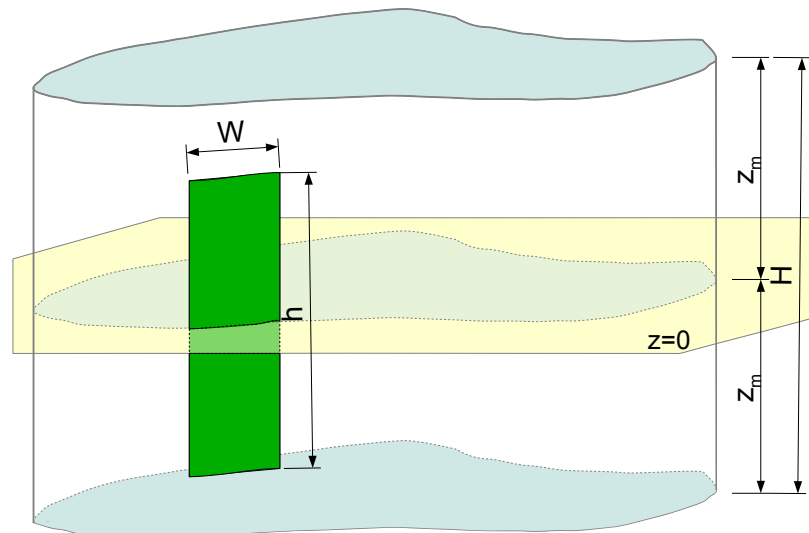


Figure 3.1: A $2\frac{1}{2}$ D model - The green region is an electrode

sample mesh followed by a description on what each value means. The last subsection provides the full system matrix with the additional terms coming from complete electrode modelling. Section 3.7 provides an explanation of how the CEM adds extra terms in a more complex 2D mesh. Matrix A_W describes the weights of the voltage of the nodes under the electrode derived by a KVL between the 'electrode node' and the 'nodes under the electrode' (or 'electrode nodes'). In the next sections, it is explained how the weights of the voltage of nodes under each electrode are calculated for a 3D mesh having 2D boundary electrodes. Section 3.9 describes the same calculation for a 3D mesh having 1D boundary electrodes.

3.1 The $2\frac{1}{2}$ D Problem

The $2\frac{1}{2}$ D problem starts by assuming having translationally invariant conductivity along the z -axis over the domain $-z_m < z < +z_m$ shown in figure 3.1. The total height of the domain, $2z_m$, is also denoted as H . Taking the assumption of translational invariance for the conductivity into effect, the voltage distribution is symmetric within

the 3D domain with respect to the $z = 0$ plane provided that the boundary condition is also symmetric. This voltage symmetry is formulated in the following equation:

$$\varphi(x, y, z) = \varphi(x, y, -z) \quad (3.1)$$

where φ is the actual electric potential in the 3D domain. Assuming symmetric voltage distribution makes it possible to take a one dimensional cosine (Fourier) series expansion over only the z -axis. Hence, the 3D voltage for each point is transformed into an infinite number of 2D voltages denoted by $V_n(x, y)$ each of which represents a spatial frequency $\frac{n\pi}{z_m}$ with the harmonic number n . Harmonic voltages, $V_n(x, y)$, which are not real potentials, are calculated by the following analysis equation:

$$V_n(x, y) = \int_{-z_m}^{+z_m} \varphi(x, y, z) \cos\left(\frac{n\pi}{z_m} z\right) dz \quad (3.2a)$$

The original 3D voltages are then calculated back using the inverse cosine series transform which is also known as the synthesis equation formulated as follows:

$$\varphi(x, y, z) = \sum_{n=0}^{\infty} V_n(x, y) \cos\left(\frac{n\pi}{z_m} z\right) \quad (3.2b)$$

Substituting equation (3.2b) into the Laplace equation for 3D domain expressed in equation (3.3) finally results in the $2\frac{1}{2}$ D governing equation expressed in equation (3.5).

$$\nabla_{3D} \cdot [\sigma(x, y, z) \nabla_{3D} \varphi(x, y, z)] = 0 \quad (3.3)$$

The details of this derivation are as follows:

$$\nabla_{3D} \cdot \left(\sigma(x, y) \nabla_{3D} \left[\sum_{n=0}^{\infty} V_n(x, y) \cos\left(\frac{n\pi}{z_m} z\right) \right] \right) = 0$$

The operator ∇_{3D} could be written as a combination of a ∇_{2D} operator for x, y and

a $\frac{\partial}{\partial z}$ operator for z coordinate as

$$\nabla_{3D} \triangleq \begin{bmatrix} \nabla_{2D} \\ \frac{\partial}{\partial z} \end{bmatrix}$$

Applying the derivatives separately makes it possible to continue the derivations more conveniently. First, the inside gradient operator is applied which results in the following expression

$$\nabla_{3D} \cdot \left(\sigma(x, y) \sum_{n=0}^{\infty} \begin{bmatrix} \nabla_{2D} V_n(x, y) \cos\left(\frac{n\pi}{z_m} z\right) \\ -\frac{n\pi}{z_m} V_n(x, y) \sin\left(\frac{n\pi}{z_m} z\right) \end{bmatrix} \right) = 0$$

Next, by applying the divergence which is the outer gradient operator, the resultant expression would be

$$\sum_{n=0}^{\infty} \left[\nabla_{2D} \cdot \left(\sigma(x, y) \nabla_{2D} V_n(x, y) \cos\left(\frac{n\pi}{z_m} z\right) \right) - \sigma(x, y) \left(\frac{n\pi}{z_m}\right)^2 V_n(x, y) \cos\left(\frac{n\pi}{z_m} z\right) \right] = 0$$

Finally, by factoring out the common cosine coefficient, the Laplace equation (3.3) transforms into equation (3.4) assuming translationally invariant conductivity along z .

$$\sum_{n=0}^{\infty} \left(\nabla_{2D} \cdot (\sigma(x, y) \nabla_{2D} V_n(x, y)) - \sigma(x, y) \left(\frac{n\pi}{z_m}\right)^2 V_n(x, y) \right) \cos\left(\frac{n\pi}{z_m} z\right) = 0 \quad (3.4)$$

Equation (3.4) is valid for each spatial frequency n . Due to the orthogonality of the cosine terms, each coefficient must be zero independently; hence, each coefficient forms a governing equation corresponding to a spatial frequency expressed in equation (3.5).

$$\nabla_{2D} \cdot [\sigma(x, y) \nabla_{2D} V_n(x, y)] - \sigma(x, y) \left(\frac{n\pi}{z_m}\right)^2 V_n(x, y) = 0 \quad (3.5)$$

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

For the $n = 0$ case, the set of equations (3.5) reduces to a regular 2D 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.

3.2 $2\frac{1}{2}$ D 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 shape and the boundary condition form the solution. Substituting synthesis equation (3.2b) into the boundary condition for the general 3D equation expressed in (3.6) results in the $2\frac{1}{2}$ D boundary condition expressed in (3.8).

$$\sigma(x, y) \frac{\partial \varphi(x, y, z)}{\partial \hat{n}} = J_s(x, y, z) \quad (3.6)$$

In equation 3.6, J_s is the surface current density on the boundary which is symmetric with respect to $z = 0$ and \hat{n} is the unit vector normal to the boundary of the medium Ω . The details of the derivation of $2\frac{1}{2}$ D boundary condition are as follows:

$$\sigma(x, y) \frac{\partial}{\partial \hat{n}} \sum_{n=0}^{\infty} V_n(x, y) \cos\left(\frac{n\pi}{z_m} z\right) = J_s(x, y, z)$$

where \hat{n} is the unit vector normal to the boundary of the medium Ω while n is the harmonic number of special frequency. As mentioned previously, the boundary condition must also be symmetric with respect to the $z = 0$ plane. Therefore, it is possible to also expand the current density, J_s , in cosine series. Changing the order of the summation and derivative together with expansion of current density results in

$$\sum_{n=0}^{\infty} \sigma(x, y) \frac{\partial}{\partial \hat{n}} V_n(x, y) \cos\left(\frac{n\pi}{z_m} z\right) = \sum_{n=0}^{\infty} J_n(x, y) \cos\left(\frac{n\pi}{z_m} z\right) \quad (3.7)$$

where J_n is the harmonic coefficient of current density corresponding to the spatial frequency $\frac{n\pi}{z_m}$. Again, due to the orthogonality of the cosine terms, the corresponding coefficients of the cosine terms from both sides of the equation must be independently equal. Hence, we have the boundary condition expressed in (3.8) for each 2½D governing equation in (3.5).

$$\sigma(x, y) \frac{\partial V_n(x, y)}{\partial \hat{n}} = J_n(x, y) \quad (3.8)$$

Assuming a uniform distribution of current density under the electrodes of height h , the boundary condition values for each harmonic n is calculated as

$$J_0 = \frac{h}{H} J_s = \frac{h}{2z_m} J_s$$

$$J_n = \frac{2}{n\pi} J_s \sin\left(\frac{n\pi h}{2z_m}\right)$$

However, the nature of the current density when it is transformed from a 3D situation into the 2D situation should also be considered. In the 3D case, the current vector is a surface current density, J_s , satisfying equation (3.10) while in the 2D case the current vector is a line/curve current density, J_l , satisfying equation (3.11)

$$I_{3D} = \iint J_s \cdot \hat{n} dS \quad (3.10)$$

$$I_{2D} = \int J_l \cdot \hat{n} d\Gamma \quad (3.11)$$

Consequently, in order to have equal current for the same problem in both 3D and 2D models, the left sides of both equations (3.10) and (3.11) should be equal. Hence, assuming a uniform distribution of current along z-axis, the relation between

the current density in 3D and 2D is be derived as follows:

$$J_s = \frac{1}{h} J_l \quad (3.12)$$

Equation (3.12) notifies that when the same amount of current, e.g. assume $I = 1$, is uniformly distributed once into a surface and another time into only one edge of that surface, the line/curve density would be greater than the surface density by a factor equal to the length of the other edge of that surface. Finally, since a 2-dimensional forward solver is used with modified stiffness matrices, for solving a 3D equation by a series of 2D equations (solvers), the boundary condition should be divided by h as expressed in (3.13).

$$J_0 = \frac{1}{H} J_l = \frac{1}{2z_m} J_l \quad (3.13a)$$

$$J_n = \frac{2}{n\pi h} J_l \sin\left(\frac{n\pi h}{2z_m}\right) \quad (3.13b)$$

In summary, for calculating the boundary condition, the following matters are considered: (1) the coefficient of the DC term ($n = 0$) is one half of the resulted coefficient from 'sinc' function (Fourier transform of a rectangular pulse) and (2) since the current for the 2D and 3D solver is same, the current density distribution for the 3D mesh is $\frac{1}{h}$ along the z coordinate, where h is the electrode height.

Finally, the 2½D modelling is solving a series of 2D partial differential equations; hence, the core equations follow a 2D attribute. Since J_l is intrinsically implemented in the 2D solver when considering the complete electrode model (see section 3.4), we only need to set current vector, I , (or the matrix consisted of current vectors for all stimulation patterns) as

$$I_0 = \frac{1}{H} = \frac{1}{2z_m} I \quad (3.14a)$$

$$I_n = \frac{2}{n\pi h} \sin\left(\frac{n\pi h}{2z_m}\right) I \quad (3.14b)$$

3.3 Method of Weighted Residuals (MWR)

The Method of Weighted Residuals, MWR, is one of the Finite Element Methods used to solve the Laplace equation and it is the most versatile approach in deriving element properties. The MWR algorithm begins by discretizing the domain Ω as a combination of finite number of elements E_k , which are also known as simplices. A simplex could be a triangle in 2D or a tetrahedron in 3D finite element model. A collection of finite elements is called a finite element mesh. In a finite element mesh there are K elements, i.e. simplices, totally having N vertices, i.e. nodes. For any position \vec{x} , the harmonic potential within the mesh, $\tilde{u}_n(\vec{x})$, is approximated by a summation of piecewise polynomial interpolation function weighted by the potential at the nodes expressed in (3.15).

$$\tilde{u}_n(\vec{x}) = \sum_{i=1}^N u_i^n \phi_i(\vec{x}) \quad (3.15)$$

where N is the number of nodes in the FEM, u_i^n are the harmonic potential of the nodes in the FEM in which n denotes the harmonic number of the potential, and ϕ_i are interpolation functions, also known as 'shape functions' or 'basis functions'. For the case of linear interpolation function, i.e. first order interpolation function, ϕ_i are expressed in (3.16) for a 2D FEM.

$$\phi_1 = \frac{1}{2A} \{(x_2 y_3 - x_3 y_2) + (y_2 - y_3)x + (x_3 - x_2)y\} \quad (3.16a)$$

$$\phi_2 = \frac{1}{2A} \{(x_3 y_1 - x_1 y_3) + (y_3 - y_1)x + (x_1 - x_3)y\} \quad (3.16b)$$

$$\phi_3 = \frac{1}{2A} \{(x_1 y_2 - x_2 y_1) + (y_1 - y_2)x + (x_2 - x_1)y\} \quad (3.16c)$$

where, (x_i, y_i) are the positions of three nodes and A is the area of the 2D finite element. Equation (3.15) provides only a finite approximation of the potential. Hence,

the Laplace equation would not be zero in a general case due to the fact that employing the summation of the weighted basis functions produces errors or 'residuals'. In addition, the basis functions ϕ_i are not differentiable at the edges and nodes; hence it is not possible to satisfy the Laplace equation directly. Therefore, in the method of weighted residuals, MWR, a weak form of the Laplace governing equation is derived by multiplying the governing equation by some arbitrary test function v and integrating over the domain Ω as expressed in (3.17).

$$\int_{\Omega} v[\nabla \cdot \sigma \nabla \tilde{u}_n - \sigma \left(\frac{n\pi}{z_m}\right)^2 \tilde{u}_n] d\Omega = 0 \quad (3.17)$$

In equation (3.17), the test function v weights the residual in order to make them zero in some average sense. Equation (3.17) should be satisfied for all test functions in a certain class. By employing the vector identity for divergence of a scalar multiplied by a vector expressed in (3.18) with notations adapted to our problem, equation (3.5) is re-formulated into equation (3.19).

$$\nabla \cdot (v\sigma \nabla(\tilde{u})) = \sigma \nabla \tilde{u} \cdot \nabla v + v \nabla \cdot (\sigma \tilde{u}) \quad (3.18)$$

$$\int_{\Omega} \nabla \cdot (v\sigma \nabla \tilde{u}_n) d\Omega = \int_{\Omega} \sigma \nabla \tilde{u}_n \cdot \nabla v d\Omega + \int_{\Omega} v\sigma \left(\frac{n\pi}{z_m}\right)^2 \tilde{u}_n d\Omega \quad (3.19)$$

Next, the Gauss-Divergence theorem is employed in order to introduce the boundary condition. The Gauss-Divergence theorem adapted to our notation is expressed in (3.21) considering the vector relation expressed in (3.20).

$$\nabla \tilde{u}_n \cdot \hat{n} = \frac{\partial \tilde{u}_n}{\partial \hat{n}} \quad (3.20)$$

$$\int_{\Omega} \nabla \cdot (v\sigma \nabla \tilde{u}_n) d\Omega = \int_{\partial\Omega} v\sigma \frac{\partial \tilde{u}_n}{\partial \hat{n}} d\Gamma \quad (3.21)$$

where $\partial\Omega$ is the boundary of the medium Ω and \hat{n} is the unit vector normal to the

boundary $\partial\Omega$. Substituting the right side of equation (3.19) with its equivalent in Gauss' theorem gives the new formulation of the problem which is expressed in (3.22).

$$\int_{\Omega} \sigma \nabla \tilde{u}_n \cdot \nabla v \, d\Omega + \int_{\Omega} v \sigma \left(\frac{n\pi}{z_m}\right)^2 \tilde{u}_n \, d\Omega = \int_{\partial\Omega} v \sigma \frac{\partial \tilde{u}_n}{\partial \hat{n}} \, d\Gamma \quad (3.22)$$

From this point the left side or the right side is referred to the corresponding side in equation (3.22). The right side of equation (3.22) is a boundary integral; hence, the boundary condition determines its value. For the simplest case, this integral is only performed underneath the current drive electrodes. The right side of the equation as well as the boundary condition will be discussed later.

3.3.1 Left Side of the Governing Equation

The integral in the left hand side of equation (3.22) is performed for the whole medium. At this point, the conductivity is discretized. There could be many methods to approximate conductivity over each element. However, the simplest method is to assume that each finite element has a constant conductivity. This approach is known as piecewise-constant (PWC) where the conductivity is formulated as:

$$\sigma_{PWC} = \sum_{k=1}^K \sigma_k \delta_k^{FE} \quad (3.23)$$

where σ_k is the conductivity of the k^{th} element and δ_k^{FE} is one under the k^{th} element and zero elsewhere. Equation (3.23) provides an piecewise-constant approximation of the conductivity σ and therefore has the advantage of separating and factoring the conductivity of each finite element out of the integral over the finite element mesh. Higher order approximations might result in better accuracy at the cost of complexity and nonlinearity. Considering PWC conductivities, for each simplex E_k , the left side

of equation (3.22) is written as

$$\int_{E_k} \sigma_k \nabla \tilde{u}_n \cdot \nabla v \, d\Omega + \int_{E_k} v \sigma_k \left(\frac{n\pi}{z_m} \right)^2 \tilde{u}_n \, d\Omega \quad (3.24)$$

Different approaches in MWR differ by the selection of the test function v . Some of these approaches are: Collocation, Least Squares, and Galerkin [10]. In the following, the Galerkin method, which is the most common method employed in MWR, is briefly described:

In the Galerkin method, the test function v is taken from a same class of functions used for approximating the potentials $\tilde{u}_n(\vec{x})$ expressed in (3.15). Hence, the same interpolation functions, $\phi_i(\vec{x})$, are used for the test function v in order to weight the residuals produced in the weak form of the governing equation. The test function employed in the Galerkin method is expressed in equation (3.25).

$$v(\vec{x}) = \sum_{i=1}^N w_i \phi_i(\vec{x}) \quad (3.25)$$

where w_i are the coefficients which weight the interpolation functions $\phi_i(\vec{x})$ in order to make the residuals zero. The weak form of the governing equation must vanish for all w_i ; therefore, the problem reduces to finding the main coefficients \tilde{u}_i which represent the potential at the nodes. By substituting the definitions of $v(\vec{x})$ and $\tilde{u}_n(\vec{x})$ into the integrals expressed in (3.24) the discretization proceeds as

$$\int_{E_k} \sigma_k \nabla \left(\sum_{i=1}^3 u_i^n \phi_i \right) \cdot \nabla \left(\sum_{j=1}^3 w_j \phi_j \right) \, d\Omega + \int_{E_k} \sigma_k \left(\frac{n\pi}{z_m} \right)^2 \left(\sum_{j=1}^3 w_j \phi_j \right) \left(\sum_{i=1}^3 u_i^n \phi_i \right) \, d\Omega$$

Performing the multiplication of two summation and taking all coefficients and

summations outside of the integral yields:

$$\begin{aligned} & \sigma_k \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j \int_{E_k} \nabla \phi_i \cdot \nabla \phi_j d\Omega + \sigma_k \left(\frac{n\pi}{z_m}\right)^2 \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j \int_{E_k} \phi_j \phi_i d\Omega \\ &= \sigma_k \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j \left(\int_{E_k} \nabla \phi_i \cdot \nabla \phi_j d\Omega + \left(\frac{n\pi}{z_m}\right)^2 \int_{E_k} \phi_j \phi_i d\Omega \right) \end{aligned} \quad (3.26)$$

In equation (3.26), σ_k is also factored out since it is constant inside each element as mentioned previously. Here, the core integrals are separated and marked with a new notation to simplify the writing; which are defined in (3.28a) and (3.28b).

$$\sigma_k \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j (S_{ij}^k + \left(\frac{n\pi}{a}\right)^2 R_{ij}^k) \quad (3.27)$$

$$S_{ij}^k = \int_{E_k} \nabla \phi_i \cdot \nabla \phi_j d\Omega \quad (3.28a)$$

$$R_{ij}^k = \int_{E_k} \phi_i \phi_j d\Omega \quad (3.28b)$$

where S_{ij}^k is the integral term which also exists in the 2D governing Laplace equation. S_{ij}^k is easy to calculate since the gradients are constant for a linear, i.e. first order, interpolating function ϕ_i . In contrast, R_{ij}^k is the additional integral term added to the problem and represents the non-zero spatial frequencies in the $2\frac{1}{2}$ D governing equation. From another point of view, it is possible to combine both integrals and build a modified 'local stiffness matrix', which is denoted as S'_{ij}^k and expressed in (3.29).

$$S'_{ij}^k = S_{ij}^k + \left(\frac{n\pi}{a}\right)^2 R_{ij}^k = \int_{E_k} \nabla \phi_i \cdot \nabla \phi_j + \left(\frac{n\pi}{a}\right)^2 \phi_i \phi_j d\Omega \quad (3.29)$$

The local stiffness matrix, also known as element stiffness matrix, is the stiffness matrix constructed for each element. The local stiffness matrix is a 3×3 matrix for

2D elements and a 4×4 matrix for 3D elements. Finally, the left side of the governing equation for each element expressed in (3.24) yields equation (3.30) or in the extended form equation (3.26).

$$\sigma_k \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j S'_{ij}{}^k \quad (3.30)$$

3.3.2 Right Side of the Governing Equation

The right hand side of the equation is again written in equation (3.31), where $\sigma \nabla \tilde{u}_n \cdot \hat{n}$ is the normal component of the current density which is zero on the boundary $\partial\Omega$, except underneath the electrodes defined by Γ . Γ is the union of the electrodes, E_l , expressed in equation (3.32) in which L is the total number of electrodes employed in the system. Note that the notation E_k was previously used for the finite 'element' index, k , in the mesh and should not be mistaken with E_l used as the electrode index, l , here. Since $\sigma \frac{\tilde{u}_n}{\partial \hat{n}}$ is equal to the current density normal to the boundary, i.e. $J \cdot \hat{n}$ and the integral is zero except underneath the electrodes, the integral is performed only over Γ .

$$\int_{\partial\Omega} v \sigma \frac{\partial \tilde{u}_n}{\partial \hat{n}} d\Gamma = \int_{\partial\Omega} v \sigma \nabla \tilde{u}_n \cdot \hat{n} d\Gamma = \int_{\Gamma} v \sigma \nabla \tilde{u}_n \cdot \hat{n} d\Gamma \quad (3.31)$$

$$\Gamma = \bigcup_l^L E_l, \quad \Gamma' = \partial\Omega \quad (3.32)$$

By substituting the definitions of $v(x)$ and $\tilde{u}_n(\hat{x})$ into the right-hand side of equation (3.22), the discretization of the boundary proceeds as

$$\int_{\Gamma} v \sigma \frac{\partial \tilde{u}_n}{\partial \hat{n}} d\Gamma = \int_{\Gamma} v \sigma \nabla \tilde{u}_n \cdot \hat{n} d\Gamma = \int_{\Gamma} \sigma \left(\sum_{j=1}^3 w_j \phi_j \right) \nabla \left(\sum_{i=1}^3 u_i^n \phi_i \right) \quad (3.33)$$

It is worth remarking that the right side of equation (3.22) represents the boundary condition; hence it may change for different boundary condition types. Considering

piece-wise constant conductivity for each element k , the right-hand side of equation (3.22) is written as

$$\sigma_k \sum_{j=1}^3 w_j \sum_{i=1}^3 u_i^n \int_{\partial\Omega} \phi_j \nabla \phi_i \cdot \hat{n} d\Gamma = \sigma_k \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j \int_{\partial\Omega} \phi_j \nabla \phi_i \cdot \hat{n} d\Gamma \quad (3.34)$$

For simplicity in future reference, as we defined S'_{ij}^k before, here the integral term is defined as Q_{ij}^k which is expressed in (3.35).

$$Q_{ij}^k = \int_{\partial\Omega} \phi_j \nabla \phi_i \cdot \hat{n} d\Gamma \quad (3.35)$$

Hence, the right side of equation (3.22) would be written as (3.36).

$$\sigma_k \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j Q_{ij}^k d\Gamma \quad (3.36)$$

3.3.3 Complete Equation

Both sides of equation (3.22), which is itself derived from the weak form of the governing equation, are now developed for each element. At this point the whole equality of equation (3.22) is expressed by equation (3.37) in an expanded form and by (3.38) in a closed form as

$$\begin{aligned} & \sigma_k \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j \left(\int_{E_k} \nabla \phi_i \cdot \nabla \phi_j d\Omega + \left(\frac{n\pi}{z_m}\right)^2 \int_{E_k} \phi_j \phi_j d\Omega \right) \\ & = \sigma_k \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j \int_{\partial\Omega} \phi_j \nabla \phi_i \cdot \hat{n} d\Gamma \end{aligned} \quad (3.37)$$

$$\sigma_k \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j S'_{ij}^k = \sigma_k \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j Q_{ij}^k \quad (3.38)$$

The equation must be valid for all test functions in a certain class (MWR, e.g in

Galerkin method v is the same as the interpolation function), therefore w_j can be removed from both sides of equation (3.37) and results in the following equation for each w_j :

$$\sigma_k \sum_{i=1}^3 u_i^n S'_{ij}{}^k = \sigma_k \sum_{i=1}^3 u_i^n Q_{ij}^k \quad (3.39)$$

In summary, for the whole mesh, the left side of equation (3.22) would be written as follows:

$$\begin{aligned} & \int_{\Omega} \sigma \nabla \tilde{u}_n \cdot \nabla v \, d\Omega + \int_{\Omega} v \sigma \left(\frac{n\pi}{z_m} \right)^2 \tilde{u}_n \, d\Omega = \\ & \sum_{k=1}^K \left(\int_{E_k} \sigma_k \nabla \tilde{u}_n \cdot \nabla v \, d\Omega + \int_{E_k} v \sigma_k \left(\frac{n\pi}{z_m} \right)^2 \tilde{u}_n \, d\Omega \right) = \sum_{k=1}^K \sigma_k \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j S'_{ij}{}^k \end{aligned} \quad (3.40)$$

From equation (3.39), the global system matrix, $S'(n)$, of the entire mesh for harmonic n is constructed by combining the local stiffness matrices using the connectivity matrix. The global system matrix is then used to solve the forward equation expressed in (3.41) for calculating the nodal voltages U_n from the forcing vector of driven current pattern I_n or J_n depending on type of the employed boundary method. The system matrix is also employed to construct the sensitivity matrix in the inverse solver.

$$S'(n)U_n = I_n \quad (3.41)$$

One approach to compute the system matrix for the entire mesh is to calculate the local stiffness matrix for each element and construct a connectivity matrix for the entire mesh based on the connectivity map of the elements. When the local system (stiffness) matrices are all calculated then the system matrix for the entire mesh is computed by the following equation:

$$S = C^T S_E \Sigma C \quad (3.42)$$

where $S_E\Sigma$ is a diagonal concatenation of local system matrices in which the conductivity of each element is also included, C is the connectivity map matrix and 'T' represents the transpose operation. Connectivity map is the relation between the node index inside each element and the general node index for the entire mesh. Construction of the connectivity map will be illustrated later in this chapter by an example.

The right side of equation, i.e. the forcing vector, could be written in different ways depending on how the electrode model imposes the boundary condition. It is also possible to write the equation as the following [4]

$$\int_{\partial\Omega} v\sigma \frac{\partial u_n}{\partial \hat{n}} d\Gamma = \int_{\partial\Omega} \sum_{i=1}^3 w_i \phi_i J_n d\Gamma = \sum_{i=1}^3 w_i \int_{\partial\Omega} \phi_i J_n \cdot \hat{n} d\Gamma \quad (3.43)$$

As mentioned previously, both side of equation should be equal for all test functions in a certain class or in order words for all w_i ; hence, w_i is removed from the equation which yields:

$$\int_{\partial\Omega} \phi_i J_n \cdot \hat{n} d\Gamma \quad (3.44)$$

3.3.4 Analytical Solution of the Integrals

The linear interpolation functions ϕ_i are written in the set of equations expressed in (3.16). For simplicity in calculation, ϕ_i and ϕ_j are written as

$$\phi_i = \frac{1}{2S} \{A_1x + B_1y + C_1\} \quad (3.45a)$$

$$\phi_j = \frac{1}{2S} \{A_2x + B_2y + C_2\} \quad (3.45b)$$

In these equations, the variables are x , y and the constants are A , B , C , S . The

integrals defined in equation (3.28) are then calculated as the following:

$$\begin{aligned}
 S_{ij}^k &= \int_{E_k} \nabla \phi_i \cdot \nabla \phi_j d\Omega = \frac{1}{(2S)^2} \int_{E_k} \begin{bmatrix} A_1 \\ B_1 \end{bmatrix} \cdot \begin{bmatrix} A_2 \\ B_2 \end{bmatrix} d\Omega = \frac{A_1 A_2 + B_1 B_2}{(2S)^2} \int_{E_k} d\Omega \\
 &= \frac{A_1 A_2 + B_1 B_2}{(2S)^2} S
 \end{aligned} \tag{3.46}$$

Calculation of S_{ij}^k is simple since the gradient of the linear interpolation function is constant. Hence, the integral is reduced to an integral over the surface which results in the area of the surface S . The integral R_{ij}^k is, however, more difficult since it results in the integrals of second degree polynomials over an arbitrary triangular surface.

$$\begin{aligned}
 R_{ij}^k &= \int_{E_k} \phi_i \phi_j d\Omega \\
 &= \frac{1}{(2S)^2} \int_{E_k} \{A_1 A_2 x^2 + (A_1 B_2 + A_2 B_1)xy + (A_1 C_2 + A_2 C_1)x \\
 &\quad + (B_1 C_2 + B_2 C_1)y + B_1 B_2 y^2 d\Omega\}
 \end{aligned} \tag{3.47}$$

The integral R_{ij}^k is divided into simpler integrals over each triangular element and each integral is calculated separately.

$$\begin{aligned}
 R_{ij}^k &= \frac{1}{(2S)^2} \{A_1 A_2 \int_{E_k} x^2 d\Omega + (A_1 C_2 + A_2 C_1) \int_{E_k} x d\Omega + (A_1 B_2 + A_2 B_1) \int_{E_k} xy d\Omega \\
 &\quad + (B_1 C_2 + B_2 C_1) \int_{E_k} y d\Omega + B_1 B_2 \int_{E_k} y^2 d\Omega\}
 \end{aligned} \tag{3.48}$$

Derivation of each integral solution over an arbitrary triangular element is expressed with complete details in Appendix A. Assuming that $P_1(x_1, y_1)$, $P_2(x_2, y_2)$ and $P_3(x_3, y_3)$ are the vertices of a triangular simplex where $x_1 \leq x_2 \leq x_3$, each integral is calculated as follows:

First, for simplicity in demonstration, four coefficients are defined, denoted by T , as

$$T_1 = \left| \frac{(y_3 - y_1)(x_2 - x_1)}{(x_3 - x_1)} - (y_2 - y_1) \right| \quad (3.49a)$$

$$T_2 = \left| \frac{(y_3 - y_1)(x_3 - x_2)}{(x_3 - x_1)} - (y_3 - y_2) \right| \quad (3.49b)$$

$$T_3 = \frac{(y_3 - y_1)(x_2 - x_1)}{(x_3 - x_1)} + (y_2 - y_1) \quad (3.49c)$$

$$T_4 = \frac{(y_3 - y_1)(x_3 - x_2)}{(x_3 - x_1)} + (y_3 - y_2) \quad (3.49d)$$

Next, the area of the triangular surface is calculated as

$$S = \int_{E_k} d\Omega = \frac{1}{2}T_1(x_2 - x_1) + \frac{1}{2}T_2(x_3 - x_2) \quad (3.50)$$

Also, integral of x over the triangular region is calculated by

$$\int_{E_k} x d\Omega = T_1 \left[\frac{1}{3}(x_2 - x_1)^2 + \frac{1}{2}(x_2 - x_1)x_1 \right] + T_2 \left[-\frac{1}{3}(x_3 - x_2)^2 + \frac{1}{2}(x_3 - x_2)x_3 \right] \quad (3.51)$$

Next, integral of x^2 over the triangular region is calculated by

$$\begin{aligned} \int_{E_k} x^2 d\Omega = & T_1 \left[\frac{1}{4}(x_2 - x_1)^3 + \frac{2}{3}(x_2 - x_1)^2 x_1 + \frac{1}{2}(x_2 - x_1)x_1^2 \right] \\ & + T_2 \left[\frac{1}{4}(x_3 - x_2)^3 - \frac{2}{3}(x_3 - x_2)^2 x_3 + \frac{1}{2}(x_3 - x_2)x_3^2 \right] \end{aligned} \quad (3.52)$$

Also, integral of y over the triangular region is calculated by

$$\int_{E_k} y d\Omega = T_1 \left[\frac{1}{2}y_1(x_2 - x_1) + \frac{1}{6}T_3(x_2 - x_1) \right] + T_2 \left[\frac{1}{2}y_3(x_3 - x_2) - \frac{1}{6}T_4(x_3 - x_2) \right] \quad (3.53)$$

Equation (3.53) could be further simplified to

$$\int_{E_k} y d\Omega = T_1\left[\left(\frac{1}{2}y_1 + \frac{1}{6}T_3\right)(x_2 - x_1)\right] + T_2\left[\left(\frac{1}{2}y_3 - \frac{1}{6}T_4\right)(x_3 - x_2)\right] \quad (3.54)$$

Next, integral of xy over the triangular region is calculated by

$$\begin{aligned} \int_{E_k} xy d\Omega = & T_1\left(y_1\left[\frac{1}{3}(x_2 - x_1)^2 + \frac{1}{2}(x_2 - x_1)x_1\right] + \frac{1}{2}T_3\left[\frac{1}{4}(x_2 - x_1)^2 + \frac{1}{3}(x_2 - x_1)x_1\right]\right) \\ & + T_2\left(y_3\left[-\frac{1}{3}(x_3 - x_2)^2 + \frac{1}{2}(x_3 - x_2)x_3\right] + \frac{1}{2}T_4\left[\frac{1}{4}(x_3 - x_2)^2 - \frac{1}{3}(x_3 - x_2)x_3\right]\right) \end{aligned} \quad (3.55)$$

Equation (3.55) could also be further simplified to

$$\begin{aligned} \int_{E_k} xy d\Omega = & T_1\left[\left(\frac{1}{3}y_1 + \frac{1}{8}T_3\right)(x_2 - x_1)^2 + \left(\frac{1}{2}y_1 + \frac{1}{6}T_3\right)(x_2 - x_1)x_1\right] \\ & + T_2\left[\left(-\frac{1}{3}y_3 + \frac{1}{8}T_4\right)(x_3 - x_2)^2 + \left(\frac{1}{2}y_3 - \frac{1}{6}T_4\right)(x_3 - x_2)x_3\right] \end{aligned} \quad (3.56)$$

Integral of y^2 over each triangular simplex is difficult to calculate analytically if the vertices are sorted with respect to x coordinate. However, if we reorder the vertices such that the new set of vertices $P'_1(x'_1, y'_1), P'_2(x'_2, y'_2)$, and $P'_3(x'_3, y'_3)$ is sorted with respect to y , the last integral would be calculated in the same way as the integral of x^2 was calculated previously. Hence, for the general case of $y_1 \leq y_2 \leq y_3$, while vertices are sorted with respect to the y coordinate, the integral of y^2 over the triangular region is calculated by

$$\begin{aligned} \int_{E_k} y^2 d\Omega = & \left| \frac{(x'_3 - x'_1)(y'_2 - y'_1)}{(y'_3 - y'_1)} - (x'_2 - x'_1) \right| \cdot \left[\frac{1}{4}(y'_2 - y'_1)^3 + \frac{2}{3}y'_1(y'_2 - y'_1)^2 + \frac{1}{2}(y'_2 - y'_1)y'^2_1 \right] \\ & + \left| \frac{(x'_3 - x'_1)(y'_3 - y'_2)}{(y'_3 - y'_1)} - (x'_3 - x'_2) \right| \cdot \left[\frac{1}{4}(y'_3 - y'_2)^3 - \frac{2}{3}y'_3(y'_3 - y'_2)^2 + \frac{1}{2}(y'_3 - y'_2)y'^2_3 \right] \end{aligned} \quad (3.57)$$

3.4 Adding Complete Electrode Model

The complete electrode model (CEM) refers to a kind of boundary condition which is neither Dirichlet nor Neumann. CEM was introduced and verified for the first time by Cheng *et al* [6]. In traditional boundary models, each electrode was modelled by a single node also being referred as point electrode model (PEM). In contrast, in CEM, each electrode is modelled by more than one node considering electrode-skin contact impedance z_l . In CEM, the boundary condition is derived by a KVL between the 'electrode voltage' and the 'voltage of the nodes under the electrode'. The electrode voltage is the voltage of a virtual node added to the system matrix (see section 3.7 for more details). CEM only changes the right-hand side of the equation (also called the current side or the forcing vector); therefore, development of the left hand side of the weak form of the governing equation for $2\frac{1}{2}$ D is the same as a 2D or 3D complete electrode model.

The right hand side of equation is again written here from equation (3.31), where $\sigma \nabla \tilde{u}_n \cdot \hat{n}$ is the normal component of the current density which is zero on the boundary $\partial\Omega$; except underneath the electrodes defined by Γ . Γ is the union of the electrodes (E_l) expressed in equation (3.32) in which L is the total number of electrodes employed in the system. It is worth remarking that the notation E_k was previously used for the Finite 'Element', k , in the mesh and should not be mistaken with E_l used as the electrode, l , here.

$$\int_{\partial\Omega} v\sigma \frac{\partial \tilde{u}_n}{\partial \hat{n}} d\Gamma = \int_{\partial\Omega} v\sigma \nabla \tilde{u}_n \cdot \hat{n} d\Gamma = \int_{\Gamma} v\sigma \nabla \tilde{u}_n \cdot \hat{n} d\Gamma$$

$$\Gamma = \bigcup_l^L E_l, \quad \Gamma' = \partial\Omega$$

The expression written in equation (3.58) for the boundary condition is commonly used for the complete electrode model, CEM. Rearranging this boundary condition

yields equation (3.59).

$$\tilde{u}_n + z_l \sigma \frac{\partial \tilde{u}_n}{\partial \hat{n}} = V_l \quad (3.58)$$

$$\sigma \frac{\partial \tilde{u}_n}{\partial \hat{n}} = \sigma \nabla \tilde{u}_n \cdot \hat{n} = \frac{1}{z_l} (V_l - \tilde{u}_n) \quad (3.59)$$

where V_l are the voltages on the electrodes ('electrode voltage') and z_l represents the contact impedance. Substituting equation (3.59) into the right hand side of the weak form of the governing equation reformulated in equation (3.31) results in the following expression

$$\int_{\Gamma} v \sigma \nabla \tilde{u}_n \cdot \hat{n} d\Gamma = \sum_{l=1}^L \int_{E_l(\Gamma)} v \frac{1}{z_l} (V_l - \tilde{u}_n) d\Omega \quad (3.60)$$

Substituting the resultant expression (3.60) into the whole weak form of governing equation expressed in (3.22) leads to the following equation

$$\int_{\Omega} \sigma \nabla \tilde{u}_n \cdot \nabla v d\Omega + \int_{\Omega} v \sigma \left(\frac{n\pi}{z_m} \right)^2 \tilde{u}_n d\Omega = \sum_{l=1}^L \int_{E_l(\Gamma)} v \frac{1}{z_l} (V_l - \tilde{u}_n) d\Omega \quad (3.61)$$

Considering the definitions of $\tilde{u}_n(\vec{x})$ expressed in (3.15) and $v(\vec{x})$ expressed in (3.25) for discretizing the right side of equation (3.61) results in the following expression

$$\sum_{l=1}^L V_l \int_{E_l(\Gamma)} \frac{1}{z_l} \left(\sum_{i=1}^3 w_i \phi_i \right) d\Gamma - \sum_{l=1}^L \int_{E_l(\Gamma)} \frac{1}{z_l} \left(\sum_{i=1}^3 w_i \phi_i \right) \left(\sum_{i=1}^3 u_i^n \phi_i \right) d\Omega \quad (3.62)$$

Rearranging the multiplication of the summations yields

$$\sum_{l=1}^L V_l \int_{E_l(\Gamma)} \frac{1}{z_l} \left(\sum_{i=1}^3 w_i \phi_i \right) d\Gamma - \sum_{l=1}^L \int_{E_l(\Gamma)} \frac{1}{z_l} \left(\sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j \phi_i \phi_j \right) d\Omega \quad (3.63)$$

Factoring the coefficients as well as the summation out of the integral and substituting the expression in (3.61) which is derived for the left hand side of the weak

form of the governing equation yields

$$\begin{aligned}
& \sum_{k=1}^K \sigma_k \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j \left(\int_{E_k(\Omega)} \nabla \phi_i \cdot \nabla \phi_j d\Omega + \left(\frac{n\pi}{z_m} \right)^2 \int_{E_k(\Omega)} \phi_i \phi_j d\Omega \right) \\
&= \sum_{l=1}^L V_l \sum_{j=1}^3 w_j \int_{E_l(\Gamma)} \frac{1}{z_l} \phi_j d\Gamma - \sum_{l=1}^L \sum_{i=1}^3 u_i^n \sum_{j=1}^3 w_j \int_{E_l(\Gamma)} \frac{1}{z_l} \phi_i \phi_j d\Gamma
\end{aligned} \tag{3.64}$$

In this equation, K is the total number of elements, L is the total number of electrodes, $E_k(\Omega)$ represents the k^{th} element and $E_l(\Gamma)$ represents the l^{th} electrode. Equation (3.64) is valid for all test functions v in a certain class, in this case the same class as the potentials, u ; hence, for each w_j it can be written as

$$\begin{aligned}
& \sum_{k=1}^K \sigma_k \sum_{i=1}^3 u_i^n \left(\int_{E_k(\Omega)} \nabla \phi_i \cdot \nabla \phi_j d\Omega + \left(\frac{n\pi}{z_m} \right)^2 \int_{E_k(\Omega)} \phi_i \phi_j d\Omega \right) \\
&= \sum_{l=1}^L V_l \int_{E_l(\Gamma)} \frac{1}{z_l} \phi_j d\Gamma - \sum_{l=1}^L \sum_{i=1}^3 u_i^n \int_{E_l(\Gamma)} \frac{1}{z_l} \phi_i \phi_j d\Gamma
\end{aligned} \tag{3.65}$$

Equation (3.65) alone is not enough to form a set of equations for solving the unknown voltages since there is no relation between the injected current and the unknown parameters. The relation between the current and unknown variables is written as follows

$$I_l = \int_{E_l(\Gamma)} \sigma \frac{\partial \tilde{u}_n}{\partial \hat{n}} d\Gamma = \int_{E_l(\Gamma)} \sigma \nabla \tilde{u}_n \cdot \hat{n} d\Gamma \tag{3.66}$$

where I_l is the current of the l^{th} electrode. Using the relation derived in equation (3.59) followed by substituting the definition of \tilde{u}_n yields

$$I_l = \int_{E_l(\Gamma)} \frac{1}{z_l} (V_l - \tilde{u}_n) d\Gamma = \int_{E_l(\Gamma)} \frac{1}{z_l} V_l d\Gamma - \sum_{i=1}^3 u_i^n \int_{E_l(\Gamma)} \frac{1}{z_l} \phi_i d\Gamma \tag{3.67}$$

The contact impedance z_l is assumed to be constant on the electrode E_l . Therefore, it can be taken out of the integral; hence, equation (3.21) leads to the following equation:

$$I_l = \frac{1}{z_l} A_{E_l} V_l - \frac{1}{z_l} \sum_{i=1}^3 u_i^n \int_{E_l(\Gamma)} \phi_i d\Gamma \quad (3.68)$$

where A_{E_l} is the area under the l^{th} electrode which is the length of the l^{th} electrode in the 2D case. Finally, equation (3.65) for all w_j together with equation (3.68) for all electrodes l form a system of equations which could be written in a matrix format as the following

$$\begin{bmatrix} A_M + A_Z & A_W \\ A_W^T & A_D \end{bmatrix} \begin{bmatrix} U \\ V \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ I \end{bmatrix} \quad (3.69)$$

In equation (3.69), U is the vector of unknown potentials on the nodes u_i^n , V is the vector of unknown voltages on the electrodes considering the electrode contact impedance, I is the vector of the injected current through the electrodes, $\mathbf{0}$ is a zero vector in which all elements are zero with the suitable dimension, and A_M is the regular stiffness matrix, also known as systems matrix, which is defined by equation (3.70) for the 2 $\frac{1}{2}$ D case.

$$[A_M]_{ij} = \sigma S'_{ij} = \sigma (S_{ij} + (\frac{n\pi}{a})^2 R_{ij}) = \sigma \int_{\Omega} \nabla \phi_i \cdot \nabla \phi_j + (\frac{n\pi}{a})^2 \phi_i \phi_j d\Omega \quad (3.70)$$

Matrix A_M does not contain any information about the boundary condition; instead, A_Z represents the effect of the contact impedance on the system matrix which is reflected into all mesh vertices underneath the electrodes. Matrix A_Z is defined by the following equation

$$[A_Z]_{ij} = \sum_{l=1}^L \int_{E_l(\Gamma)} \frac{1}{z_l} \phi_i \phi_j d\Gamma \quad (3.71)$$

The remaining part in equation (3.65) forms the matrix A_W which represents the weights of the voltage on the electrodes in that equation. The matrix A_W is defined

in equation (3.72).

$$[A_W]_{ij} = - \int_{E_i(\Gamma)} \frac{1}{z_l} \phi_i d\Gamma \quad (3.72)$$

The close form of equation (3.65) adapted to the new notation used here is expressed in (3.73).

$$(A_M + A_Z)U + A_WV = \mathbf{0} \quad (3.73)$$

Finally, in equation (3.69), the matrix A_D is a diagonal matrix in which the values in the main diagonal are the ratio $\frac{1}{z_l}A_{E_l}$ for all of the electrodes and defined as

$$[A_D]_{LL} = \text{diag}\left(\frac{1}{z_l}A_{E_l}\right) \quad (3.74)$$

Matrix A_D together with the transpose of matrix A_W form the closed form of equation (3.68) which is expressed in equation (3.75). This equation represents the relation between the injected current through each electrodes and those nodes in the mesh that lie underneath the electrodes as well as the additional virtual nodes (considered in the mesh) which represent the real electrodes. The matrix A_D , itself, represents the coefficients of the unknown voltages on the electrode.

$$A_W^T U + A_D V = I \quad (3.75)$$

3.5 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. From chapter 2, the difference EIT problem is modelled as:

$$\Delta v = \mathbf{J}\Delta\sigma \quad (3.76)$$

where \mathbf{J} is known as the "sensitivity matrix", Δv 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.

Modification of the inverse problem for 2½D method demonstrates that the sensitivity matrix of the whole problem is the summation of sensitivity matrix for each 2½D equation in (3.5). The procedure for proving this claim is as follows:

For each single 2½D equation, the relation between the boundary difference voltages, Δv_n , and the difference conductivity, σ , is written as

$$\Delta v_n = \mathbf{J}_n \Delta\sigma \quad (3.77)$$

where \mathbf{J}_n is the sensitivity matrix for harmonic n . According to the synthesis equation (3.2b), the summation of Δv_n 's, i.e., the boundary difference voltage for each partial differential equation (PDE) in (3.5) corresponding to a spatial frequency, forms the actual boundary voltage for the plane $z = 0$, denoted by Δv . Hence, substituting the relation of each boundary voltage with the conductivity as expressed in (3.77) yields:

$$\Delta v = \sum_{n=0}^{\infty} \Delta v_n = \sum_{n=0}^{\infty} \mathbf{J}_n \Delta\sigma \quad (3.78)$$

Factoring out the conductivity in equation (3.78) results in

$$\Delta v = \left(\sum_{n=0}^{\infty} \mathbf{J}_n \right) \Delta\sigma \quad (3.79)$$

Equation (3.79) is actually the definition of the entire inverse problem expressed in (3.76). 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 which

is defined in (3.80).

$$\mathbf{J} = \sum_{n=0}^{\infty} \mathbf{J}_n \quad (3.80)$$

3.6 Example of a Simple 2D Mesh

In this section, the construction process of the stiffness (system) matrix for a simple sample mesh is described. Suppose that a 2D problem with Complete Electrode Model has the 2D mesh depicted in figure 3.2. In this figure (and the following figures), the green lines represent the electrodes, the 'boxed' numbers are the global index of the nodes for the entire mesh, the 'circled' numbers represent the local index of the nodes inside of each element and the numbers in 'diamond' represent the index of the elements. From section 3.3.3, we know that the system matrix for the entire mesh is constructed based on the connectivity matrix C and the local system matrices S_E . The connectivity matrix C is a matrix (or simply a table) which represents the connectivity or relation between the local node index in each single element and the global node index in the entire mesh. This matrix would have as many columns as the number of nodes in the element and as many rows as the total number of nodes in the mesh.

3.6.1 Connectivity Matrix

The connectivity matrix C can be expressed as a concatenation of local connectivity matrices, C_{E_i} , placed by the order of elements' index as expressed in (3.81). Constructed for each single element, the local connectivity matrix C_{E_i} expresses the connectivity map or relation between the local index number assigned to the nodes of a single element within that element and the global index number assigned to the

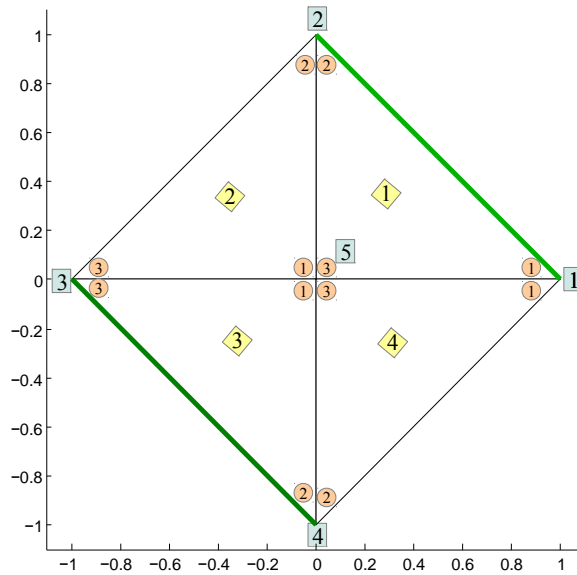


Figure 3.2: A sample 2D mesh - Green regions are electrodes

same nodes within the entire mesh.

$$C^T = \left[C_{E_1}^T \mid C_{E_2}^T \mid C_{E_3}^T \mid C_{E_4}^T \right]^T \quad (3.81)$$

A single node is normally involved in multiple elements. This node would have only one unique index number with respect to the entire mesh, while the node can have multiple index numbers each of which represents a local node index within different elements. The connectivity matrix depends on the way the nodes are indexed both for the entire mesh and inside of each element. For the sample mesh depicted in figure 3.2, the connectivity matrix C , associated with the numbering order depicted

in the figure, is calculated as:

$$C = \left[\begin{array}{ccc|ccc|ccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{array} \right]^T \quad (3.82)$$

3.6.2 Stiffness Matrix

Local Stiffness Matrix

Calculated for each single element, the local stiffness matrix is the solution of the integral (3.28a) for a 2D or 3D problem; or the integral (3.29) for a $2\frac{1}{2}$ D problem. For the purpose of matrix multiplication, the local stiffness matrices are diagonally placed in a matrix; which contains all local stiffness matrices placed by the order of element index and zero elsewhere as:

$$S_E = \left[\begin{array}{cccc} S_{E_1} & 0 & 0 & 0 \\ 0 & S_{E_2} & 0 & 0 \\ 0 & 0 & S_{E_3} & 0 \\ 0 & 0 & 0 & S_{E_4} \end{array} \right] \quad (3.83)$$

Matrix S_E is not the stiffness matrix of the entire mesh since it is only a matrix of local stiffness matrices. The stiffness matrix of the entire mesh is a square matrix having as many rows and columns as the number of nodes in the entire mesh; while S_E is a sparse matrix with dimensions: number of nodes in each element times number of elements. Obviously, S_E contains redundancy due to the fact that there are repeated rows for each nodes. In addition, until this stage, the conductivity value of each element is not considered in the calculation of the local stiffness matrices. The stiffness

(or system) matrix S for the whole mesh is then calculated based on the following expression:

$$\begin{aligned}
 S &= C^T S_E \Sigma C \\
 &= \left[C_{E_1}^T \mid C_{E_2}^T \mid C_{E_3}^T \mid C_{E_4}^T \right] \cdot \begin{bmatrix} \sigma_1 S_{E_1} & & & \\ & \sigma_2 S_{E_2} & & \\ & & \sigma_3 S_{E_3} & \\ & & & \sigma_4 S_{E_4} \end{bmatrix} \cdot \begin{bmatrix} C_{E_1} \\ C_{E_2} \\ C_{E_3} \\ C_{E_4} \end{bmatrix} \quad (3.84)
 \end{aligned}$$

where Σ is a diagonal matrix containing the element conductivities in its diagonal in the form of tensors. Pursuing the matrix multiplications for isotropic conductivities would result in the following expression for the stiffness matrix of the entire mesh:

$$\begin{aligned}
 S &= C^T S_E \Sigma C = \left[C_{E_1}^T \sigma_1 S_{E_1} \mid C_{E_2}^T \sigma_2 S_{E_2} \mid C_{E_3}^T \sigma_3 S_{E_3} \mid C_{E_4}^T \sigma_4 S_{E_4} \right] \cdot \begin{bmatrix} C_{E_1} \\ C_{E_2} \\ C_{E_3} \\ C_{E_4} \end{bmatrix} \quad (3.85) \\
 &= \sum_{i=1}^4 C_{E_i}^T \sigma_i S_{E_i} C_{E_i}
 \end{aligned}$$

Equation (3.85) demonstrates that the stiffness matrix for the whole mesh can be calculated without constructing large sparse matrices. Here, the summation has 4 terms because the mesh in figure 3.2 contains four elements. For a general problem, the stiffness matrix of the entire mesh is the summation of the stiffness matrices calculated for each element as expressed in (3.86). The stiffness matrix calculated for each element is itself computed by: $C_{E_i}^T \sigma_i S_{E_i} C_{E_i}$; where C_{E_i} is the local connectivity matrix, S_{E_i} is the local stiffness matrix which is the solution of the integral express

in (3.29) and σ_i is the conductivity associated with element i .

$$S = \sum_{i=1}^K C_{E_i}^T \sigma_i S_{E_i} C_{E_i} \quad (3.86)$$

In this equation K is the total number of elements in the mesh.

Stiffness (System) Matrix

Finally, the system (stiffness) matrix of the mesh depicted in figure 3.2 is written as:

$$S = C^T S_E \Sigma C = \sum_{i=1}^K C_{E_i}^T S_{E_i} \sigma_i C_{E_i} = \begin{bmatrix} \sigma_1 S_{11}^1 + \sigma_4 S_{11}^4 & \sigma_1 S_{12}^1 & 0 & \sigma_4 S_{12}^4 & \sigma_1 S_{13}^1 + \sigma_4 S_{13}^4 \\ \sigma_1 S_{22}^1 + \sigma_2 S_{22}^2 & \sigma_2 S_{23}^2 & 0 & 0 & \sigma_1 S_{23}^1 + \sigma_2 S_{23}^2 \\ \sigma_2 S_{33}^2 + \sigma_3 S_{33}^3 & \sigma_3 S_{32}^3 & \sigma_3 S_{31}^3 & \sigma_3 S_{32}^3 + \sigma_4 S_{32}^4 & \sigma_2 S_{31}^2 + \sigma_3 S_{31}^3 \\ \sigma_3 S_{22}^3 + \sigma_4 S_{22}^4 & \sigma_4 S_{23}^4 & \sigma_4 S_{21}^4 & \sigma_4 S_{22}^4 & \sigma_3 S_{21}^3 + \sigma_4 S_{23}^4 \\ \sigma_1 S_{33}^1 + \sigma_2 S_{11}^2 + \sigma_3 S_{11}^3 + \sigma_4 S_{33}^4 & \sigma_2 S_{12}^2 & \sigma_3 S_{12}^3 & \sigma_4 S_{12}^4 & \sigma_1 S_{13}^1 + \sigma_2 S_{13}^2 + \sigma_3 S_{13}^3 + \sigma_4 S_{13}^4 \end{bmatrix} \quad (3.87)$$

where, S_{ij}^e represents the ij th element of the local stiffness matrix of the e th element or in other words: $[S_{E_i}]_{ij}$. The system matrix S is a symmetric matrix for isotropic conductivities; hence, for simplicity, here, only the upper triangle terms are expressed. As an example, the values of the column number 5 which corresponds to node number five is described:

In order to avoid conflict, the global nodal index numbers for the entire mesh are specified in alphabetic format while the local nodal index numbers for each elements are specified by digits. So, for example, node #five refers to the fifth ('boxed') node of the mesh while the same node is indexed as #1 ('circled') in the second and third element and indexed as #3 ('circled') in the first and fourth elements.

- S_{15} represents the geometrical relation between node #one and node #five. These two nodes are connected to each other by two different elements. Hence the value is the summation of the cross stiffness between these two nodes for

each of elements #one and #four. For element #one, node #one is indexed as #1 and node #five is indexed as #3. Hence, the corresponding value is element (1,3) from the local stiffness matrix of element #one, S_{13}^1 . Similarly, for element #four, due this specific numbering chosen, node #one is indexed as #1 and node #five is indexed as #3. Hence, the corresponding value is element (1,3) from the local stiffness matrix of element #4, S_{13}^4 .

- S_{25} represents the geometrical relation between node #two and node #five. These two nodes are connected to each other by elements #one and #two. For element #one, node #two is indexed as #2 and node #five is indexed as #3. Hence, the corresponding stiffness value is element (2,3) from the local stiffness matrix of element #one, S_{23}^1 . Also, for element #four, node #two is indexed as #2 and node #five is indexed as #1. Hence, the corresponding value is element (2,1) from the local stiffness matrix of element #two, S_{21}^2 .
- S_{55} represents the self-connection of node #five. This value includes all self-stiffness values corresponding to this node from all elements that contains node #five. For element #one, this node is indexed as #3; hence, S_{33}^1 . For element #two, this node is indexed as #1; hence S_{11}^2 . For element #three, this node is indexed as #1; hence, S_{11}^3 . Finally, For element #four, this node is indexed as #3; hence, S_{33}^4 .
- The value of the stiffness for those nodes that are not connected to each other via any elements is clearly zero. e.g., S_{13} and S_{24} .

3.6.3 CEM terms

Adding the Complete Electrode Model extra nodes to the system matrix results in:

$$S = \begin{bmatrix} C^T S_E \Sigma C + A' & A_W \\ A_W^T & A_D \end{bmatrix} = \begin{bmatrix} \sigma_1 S_{11}^1 + \sigma_4 S_{11}^4 & \sigma_1 S_{12}^1 & 0 & \sigma_4 S_{12}^4 & \sigma_1 S_{13}^1 + \sigma_4 S_{13}^4 & -\frac{\Delta}{2z_c} & 0 \\ & \sigma_1 S_{22}^1 + \sigma_2 S_{22}^2 & \sigma_2 S_{23}^2 & 0 & \sigma_1 S_{23}^1 + \sigma_2 S_{23}^2 & -\frac{\Delta}{2z_c} & 0 \\ & & \sigma_2 S_{33}^2 + \sigma_3 S_{33}^3 & \sigma_3 S_{32}^3 & \sigma_2 S_{31}^2 + \sigma_3 S_{31}^3 & 0 & -\frac{\Delta}{2z_c} \\ & & & \sigma_3 S_{22}^3 + \sigma_4 S_{22}^4 & \sigma_3 S_{21}^3 + \sigma_4 S_{23}^4 & 0 & -\frac{\Delta}{2z_c} \\ & & & & \sigma_1 S_{33}^1 + \sigma_2 S_{11}^2 + \sigma_3 S_{11}^3 + \sigma_4 S_{33}^4 & 0 & 0 \\ -\frac{\Delta}{2z_c} & -\frac{\Delta}{2z_c} & 0 & 0 & 0 & \frac{\Delta}{z_c} & 0 \\ 0 & 0 & -\frac{\Delta}{2z_c} & -\frac{\Delta}{2z_c} & 0 & 0 & \frac{\Delta}{z_c} \end{bmatrix} \quad (3.88)$$

where, Δ is the distance between node 1 and 2, or 3 and 4, which is $\sqrt{2}$ in this example. The matrix A_W , which is added to the original stiffness matrix, is calculated based on the following integral:

$$[A_W]_{ij} = -\frac{1}{z_j} \int_{El_j} \phi_i dS \quad (3.89)$$

where j represents the electrode index and i represents the node index. The integral is operated under the j th electrode where ϕ_i is the union of all interpolation functions associated to the i th node from every element that contains the i th node (see figure 3.4). Obviously, this integral is zero for those nodes that are not under the j th electrode. For the 2D mesh depicted in figure 3.2, this integral is $-\frac{\Delta}{2z_j}$ for nodes under the j electrode; where Δ is the distance between two adjacent nodes under the electrode, i.e., here, the electrode width; which is $\sqrt{2}$. More details about this calculation are provided in the next example.

3.7 A more complex 2D mesh

The previous example was a sample of a simple 2D mesh where there were only two nodes under each electrodes; hence, both nodes were located at the electrode ends.

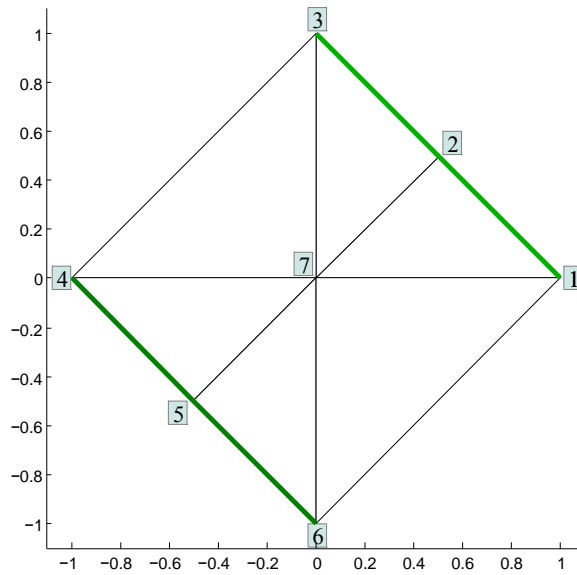


Figure 3.3: A sample complex mesh - Green regions are electrodes

Figure 3.3, shows another mesh in which there are three nodes under each electrode. The calculation of the integral expressed in (3.89) for this case can be extended (generalized) to a 2D mesh where there are at least three nodes under an electrode.

The integral is calculated under each electrode; hence, the region subject to the integral is the intersection of the j th electrode and the interpolation function ϕ_i (for the i th node). In a 2D mesh, the interpolation function for the i th node forms a pyramid in $(x, y, \phi_i(x, y))$ coordinate system where all nodes connected to the j th node form the base of the pyramid and the j th node is the vertex of the pyramid. Since the value of the interpolation function for each node at the position of that node is one, the height of the pyramid is one. Figure 3.4 shows the global interpolation function for node 7 in figure 3.3 which is the union of all interpolation functions for node 7 from all elements that contains node 7.

The intersection of the 2D (plane) mesh with a boundary electrode is a line; hence, the cross sectional cut (or intersection) of the pyramidal interpolation function under the line or curve of the boundary electrode is a triangle with a height equal to one. The desired integral is the area of this triangle overlapping with the electrode. For

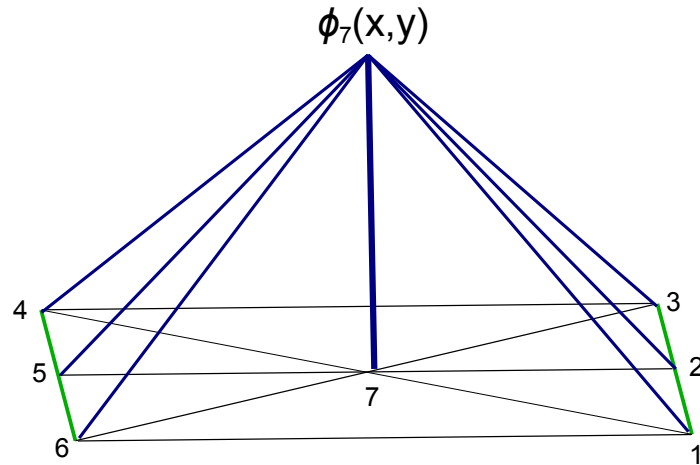


Figure 3.4: Global interpolation function for node 7 and its intersection with the electrode forming a triangle - Green regions are electrodes

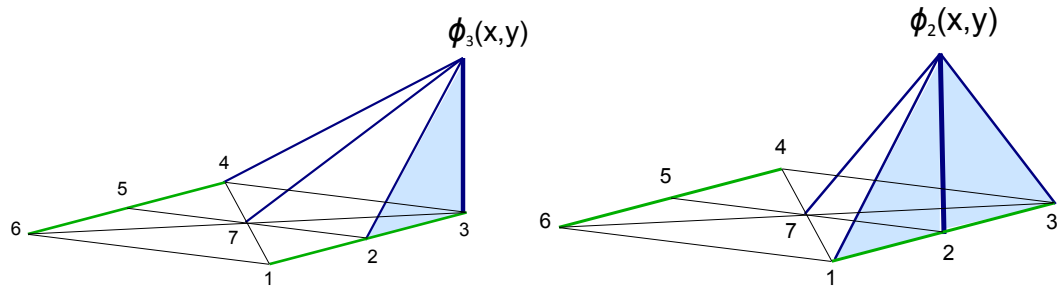


Figure 3.5: Global interpolation function for (left) node 3 (right) node 2 - Green regions are electrodes

the end nodes in electrodes, e.g., 1,3,4, and 6, the overlapping part is only a part of this triangle having width equal to the distance between the end node and the nearest node under the electrode (see figure 3.5-left). However, for middle nodes, e.g., 2 and 5, the neighboring nodes under the electrode lie in both right and left side (see figure 3.5-right); hence, the base of the triangle is equal to the sum of all distances between the node and its two neighboring nodes. In other words the intersection is two triangles having unit height and each of them have a base equal to the distance between the node and a neighboring node.

Figure 3.6 shows the intersection of the interpolation functions with an electrode

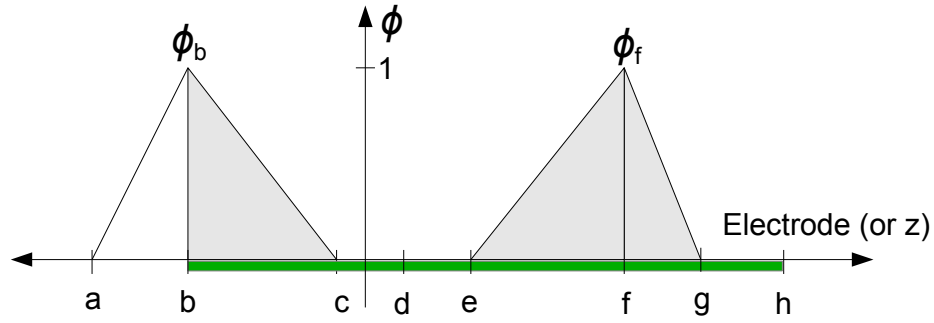


Figure 3.6: How integration is performed for an electrode end node and an electrode mid node

(the green region) for a general case of non-evenly spaced nodes. $a, b, c, d, e, f, g,$ and h are the node labels. As discussed for the previous figure (figure 3.5), this intersection is a triangular region which is the region over which the integral in (3.89) is performed. Since the height of the triangle (value of the interpolation function) is one, the result of the integrals (the area of the triangles) is proportional to the base of the triangles (distance between the electrode nodes). The triangle on the left side of figure 3.6 describes the integration for a typical node located at the electrode end labeled as ϕ_b as while the right side triangle in the same figure (labeled as ϕ_f) describes the integration for any node under the electrode except the two end nodes. For an electrode end-node (like node b in the figure), the integral result is only a part of the triangle (the shaded region) which includes only one node distance; while for any electrode mid-node (like node f), the integral result is the whole area of the triangle and includes two node distances.

For the sample mesh depicted in figure 3.3, the core system matrix is calculated similarly by following the instructions provided for the previous example. However, the matrix A_W for this mesh would be:

$$A_W = \begin{bmatrix} -\frac{\Delta}{2z_c} & -\frac{\Delta}{z_c} & -\frac{\Delta}{2z_c} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{\Delta}{2z_c} & -\frac{\Delta}{z_c} & -\frac{\Delta}{2z_c} & 0 \end{bmatrix}^T \quad (3.90)$$

For a 2D mesh with Complete Electrode Model where the nodes are evenly spaced under the electrodes, the value of A_W associated to an end node under the electrode is one half of that value for a middle point under that electrode. In other words, each middle point has two times more effect on the electrode voltage than each end node. The relation between the voltage of the nodes under the electrode and the electrode voltage, i.e., the voltage of the virtual node added for each electrode, is calculated back by equation (3.75) which is expressed again here:

$$A_W^T U + A_D V = I \quad (3.91)$$

where U is the voltage of the real nodes in the mesh and V is the electrode voltages which is the voltage of the virtual nodes added to the forward model representing each electrode voltage. Since the integral in (3.89) is taken under each electrode, the matrix A_W has only non-zero values for electrode nodes. Considering one row of the matrix equation (3.91) and employing the definition of matrix A_D in (3.74), the relation between the electrode voltage and the voltage of the nodes under the electrode is:

$$W_j^T U_j + l_{E_j} V_j = Z c_j I$$

$$\sum_{i=\text{jth electrode nodes}} w_i \cdot U_i = Z c_j I_j - l_{E_j} V_j$$

where l_{E_j} is the length (or area in a 2D electrode) of the j^{th} electrode, u_i is the voltage of the i^{th} node under the j^{th} electrode and w_i is the weight of u_i which is calculated from the mesh geometry (see section 3.8 for a 2D electrode). For a passive electrode the injected current is zero, i.e., $I_j = 0$. Therefore the electrode voltage is:

$$V_j = \frac{-\sum_{i=\text{jth electrode nodes}} w_i \cdot u_i}{l_{E_j}} = - \sum_{i=\text{jth electrode nodes}} \bar{w}_i \cdot u_i \quad (3.92)$$

where, \bar{w}_i is the normalized w_i with respect to the electrode length (or area). Hence, for a passive electrode the electrode voltage is a weighted average of the voltage of each node under the electrode. In general, the effect of the voltage of each node under the electrode on the electrode voltage is proportional to the total distance between that node and all of its neighboring nodes connected to the electrode ('electrode nodes'). The weights are normalized with respect to the electrode length.

For an active electrode, however, the relation between the electrode voltage and the node voltages would be:

$$V_j = \frac{Zc_j I_j - \sum_{i=\text{jth electrode nodes}} w_i \cdot u_i}{l_{E_j}} = \frac{Zc_j}{l_{E_j}} I_j - \sum_{i=\text{jth electrode nodes}} \bar{w}_i \cdot u_i \quad (3.93)$$

3.8 A 3D mesh having 2D boundary electrodes

The system matrix of a 3D mesh is calculated similar to the way the system matrix is calculated for a 2D mesh; because, all of the equations are same. For a 3D mesh, the matrix A_W is computed using the same integral in (3.89). However, here, the interpolation functions approximate the voltages of 3D elements having 3D positions for nodes. Hence, for the purpose of plotting, we require a 4-dimensional sketch. A 3D interpolation function could be imagined with the help of colors. For example, a 3D interpolation function for tetrahedral element can be imagined as a tetrahedron filled with colorful water. Using linear interpolation functions, the interpolation function for each node is one on the node and zero for its front base/face. The function values which are, here, imagined as colors would uniformly decrease from one to zero toward the front base through the height (see figure 3.7).

For the 3D case of equation (3.89), we are integrating a 3D interpolation function over each electrode which is a subset of the boundary surface. In fact, this electrode surface contains the whole side (face) of one or more boundary elements which are in

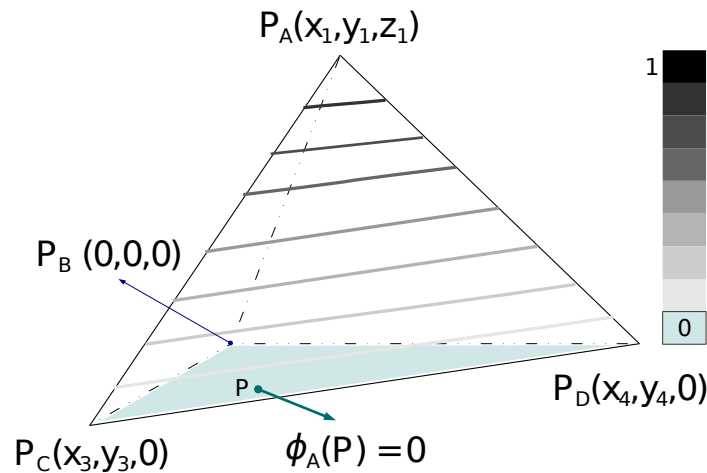


Figure 3.7: Use of colors for demonstrating 3D linear interpolation function, $\phi_A(x, y, z) = \frac{z}{z_1}$

contact to the electrode. From section 3.3, we know that the value of the interpolation function for each node is one at that node and zero in other nodes in the front base/face; this statement is expressed in equation (3.94)

$$\begin{cases} \phi_A(P_A) = 1 \\ \phi_A(P) = 0, \quad P \in \Delta(P_B, P_C, P_D) \end{cases} \quad (3.94)$$

where P_A , P_B , P_C and P_D , are the four nodes of a tetrahedral element, and Δ represents a triangle formed by the three (boundary) nodes P_B , P_C and P_D . Employing linear interpolation functions, the value of the interpolation function for each face of a 3D tetrahedral element only depends on the three nodes forming that face and does not depend on the corresponding node/vertex in front of that face (see figure 3.7). Considering this fact, the value of the interpolation function for any face of a tetrahedral element (such as the boundary face) can be written as 2D interpolation functions; hence, the value could be demonstrated as pyramids (see section 3.7). In a 3D boundary element, suppose that P_A is the node that is not in the boundary and P_B , P_C and P_D forms a 2D boundary element (triangle). Here, the intersection of the 3D interpolation function and the boundary surface is the pyramid shape function of

the 2D boundary element. Therefore, the interpolation function for the 2D boundary nodes are similar to figure 3.4.

The integral in equation (3.89) is taken over the electrode area which contains the boundary face of 3D boundary elements. The total interpolation function for each node would be the union of all interpolation functions associated to that node for all elements containing that node. Hence, the integral is actually for a pyramidal integrand having unit height in each electrode node and performed over a surface 2D region (union of boundary elements). In other words, the integral solution would be the volume of a unit height pyramid (similar to figure 3.4; however, for 3D, the integrals are taken over the element surface not over the edges)

From 3D geometry and Cavalieri's principle we know that the volume of a pyramid is equal to one third of its height multiplied by the area of its base, i.e. $V = \frac{1}{3}h \times A_{base}$. Thus, the weight of each node under the electrode in calculating the electrode voltage, i.e. voltage of the virtual node added to the mesh nodes for each electrode, is proportional to the total area of the region formed by this node together with those neighboring nodes under the electrode that are directly connected to this node.

Figure 3.8 shows a sample 3D mesh with tetrahedral elements having equal-size side faces under the boundary electrode. Supposed that the 3D mesh depicted in figure 3.8 is employed in the forward modelling for an EIT problem. The weight of each node under the electrode in the averaging process for computing the electrode voltage would then be proportional to the number of neighboring nodes connected directly to that node. However, as can be observed from the figure, the number of connected neighboring nodes is different for each node even if the node is inside the electrode boundary area. This would also affect the number of neighboring nodes under the electrode connected to a corner node or an edge node in the electrode boundary area. Hence, the electrode voltage, which is the weighted average of the voltages of the nodes under the electrode, depends on the connection structure used

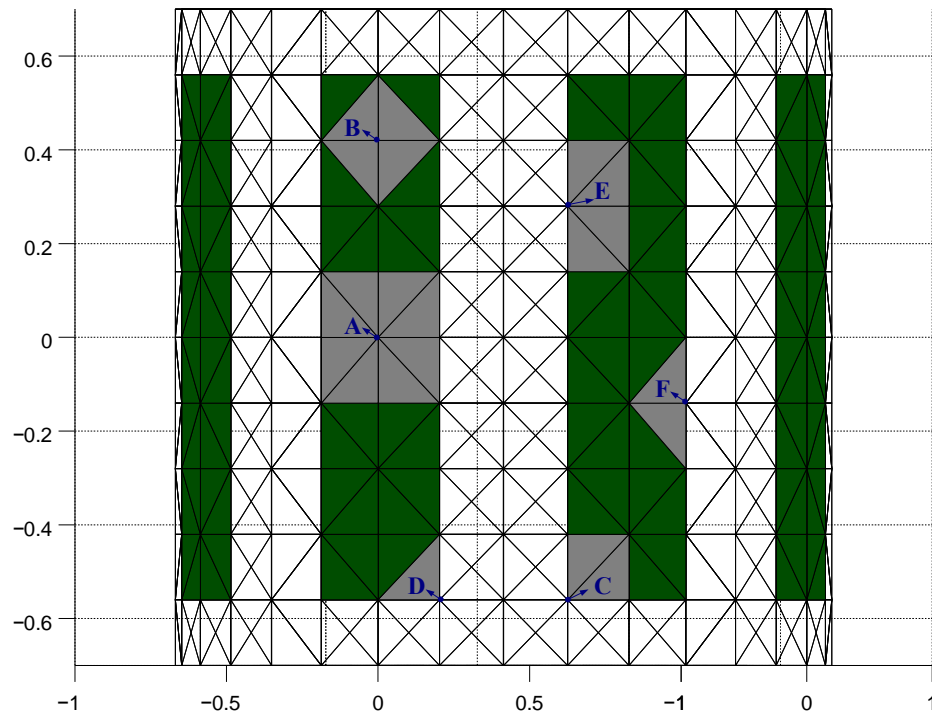


Figure 3.8: Different kinds of nodes on the boundary of a 3D mesh. Green regions are electrodes, gray regions are subregion of electrode and selected nodes are labeled by alphabetical letter in blue

for connecting nodes in a 3D mesh. In the following, some examples for different kinds of node underlying the electrode in figure (3.8) are provided:

- Node A lies inside the electrode region. This node is connected to 8 neighboring nodes which form 8 triangular areas.
- Node B lies inside the electrode region. This node is connected to 4 neighboring nodes which form 4 triangular areas.
- Node C lies in the corner of the electrode region. This node is connected to 3 neighboring nodes underlying the electrode which form 2 triangular areas under the electrode.
- Node D lies in the corner of the electrode region. This node is connected to 2 neighboring nodes underlying the electrode which form 1 triangular areas

under the electrode.

- Node E lies in the edge of the electrode region. This node is connected to 5 neighboring nodes underlying the electrode which form 4 triangular areas under the electrode.
- Node F lies in the edge of the electrode region. This node is connected to 3 neighboring nodes underlying the electrode which form 2 triangular areas under the electrode.

3.9 A 3D mesh having 1D line electrodes

In a simplified Complete Electrode Model of a 3D mesh, electrodes can be modelled as lines having zero width. In this case, the area of each electrode in the boundary surface is zero. Hence, the method mentioned in the previous section does not work. As stated previously, the interpolation function for each node on the boundary surface only depends on its neighboring node on the boundary and does not depend on the nodes inside the mesh (outside the boundary).

Hence, the interpolation function for each node on the boundary surface is a pyramid which is the union of all tetrahedral interpolation functions of this node for all elements containing this node. The integral in equation (3.89) is taken over each electrode, which is a line here, for the interpolation function of each node under the electrode. The intersection of the pyramid shape interpolation function for each node with the electrode line is a triangle having unit height and a base equal to that part of electrode line which has intersection with the base of the pyramid. In fact, the solution of the integral in (3.89) is the area under a triangle function that connects the node to its neighboring node toward the electrode path with value one at the node and zero for the neighboring nodes. The integral in (3.89) is, in this case, calculated

using the same way as the situation of a 2D mesh with complete electrode model which was stated in section 3.7 and demonstrated in figure 3.6.

3.10 Chapter Summary

In this chapter, first the $2\frac{1}{2}$ D problem was formulated and the boundary condition was derived correctly. Then, the complete electrode model added to the derived system matrix and the modification of the inverse problem was discussed. In the second part of the chapter, it was shown how the system matrix is constructed for two sample meshes. It was described how the electrode voltage is calculated as a weighted average of voltages of all nodes under the electrode. These weights are calculated based on the length or area of the intersection of the electrode and the interpolation function of each node. For 2D electrodes, the weight for each node is proportional to the area of the region which is formed by that node together with all of its neighboring nodes under the electrode that are directly connected to that node; the weights are normalized by the total area. For 1D electrodes, regardless of the type of the actual mesh (3D or 2D), the weight for each node is proportional to the summation of the distances between that node and its neighboring nodes under the electrode; the weights are normalized by the total length/area of the electrode.

Chapter 4

Validation

In this chapter, the $2\frac{1}{2}$ D algorithm and code are validated. It is shown how the $2\frac{1}{2}$ D solver modifies the 2D solution toward the 3D solution. First, it is shown that the 2D and 3D solvers work differently since the 2D solver employs 2D interpolation functions and the 3D solver employs 3D interpolation functions; hence, the solutions are not completely matched for a $2\frac{1}{2}$ D mesh with a 'full $2\frac{1}{2}$ D boundary'. It is also discussed that 3D solution is not completely reliable in comparisons because it varies by changing the element height in the 3D mesh while the 2D and $2\frac{1}{2}$ D solutions are not dependent of the element height.

In consistent with the notation in figure 3.1, here, we denote h as the height of electrodes, H as the height of the domain where translational invariancy is assumed (the tank height) and W as electrode width. In this chapter, the term 'gradients at a specific z layer' refers to the difference between the node voltage under the adjacent passive electrodes (excluding the current drive electrodes) at that specific z position while the 'measurements' refer to the difference between the voltage on adjacent passive electrodes.

First, assuming $H = h$, it is demonstrated that the set of $2\frac{1}{2}$ D equations reduces to a simple 2D equation. Next, assuming $H = h = 1$, results of the 2D and 3D solver are compared at different layer in z -coordinate as well as at the electrode node. In

section 4.2, it is shown that by reducing the element height, the 3D solution changes. For a sample case of $H = h = 1$, it is shown that the difference between the 3D and 2D solution is reduced while the 2D solution is not a dependent of the 3D element height.

Next, in section 4.3, the terms of $2\frac{1}{2}$ D solution are analyzed. It is demonstrated that the first term of the $2\frac{1}{2}$ D is a normalized 2D solution with respect to the domain height, H . Assuming $H = h$, the normalized 2D solution is then compared to the 3D solution for different H . In section 4.3.2, the $2\frac{1}{2}$ D solution is compared to the 3D solution and the 2D solution for a sample mesh ($H = 2$ and $h = 0.4$). Lastly, the sources of errors which prevent a perfect match are discussed.

4.1 Finding a $2\frac{1}{2}$ D problem for comparison

4.1.1 Analyzing the set of $2\frac{1}{2}$ D equations

Theoretically, solving the set of $2\frac{1}{2}$ D equations (3.5) and taking the inverse Fourier of the result is equal to the 3D solution. Assuming $h = H$, which means full height electrodes, the Fourier series of the current density, which is the right hand side of equation (3.22), reduces to a single non-zero coefficient at $n = 0$; hence, from the set of $2\frac{1}{2}$ D equations expressed in (3.5), only the first equation remains which is expressed in (4.1) since the solution of higher order equations would be zero.

$$\nabla_{2D} \cdot \sigma \nabla_{2D} V = \begin{cases} \frac{J_0}{H} & \text{under boundary} \\ 0 & \text{elsewhere} \end{cases} \quad (4.1)$$

In equation (4.1), V is the voltage in the mesh which is the solution to be calculated, σ is the conductivity, J_0 is the current density of the injected current under the electrode and ∇_{2D} is the 2D gradient operator. Employing a unit height tank,

i.e. $H = 1$, equation (4.1) would be the Laplace equation for a 2D domain. Hence the $2\frac{1}{2}$ D solution which is itself theoretically equal to the 3D solution would be equal to the 2D solution for this case.

To sum up, the 2D model is an equivalent model of a 3D model when $H = h = 1$. This means that, the 2D model is equivalent to a 3D tank having translational invariant conductivity with a unit height ($H = 1$), electrodes with full height ($H = h$); i.e., unit height electrode ($h = 1$), and a same base as the 2D model.

4.1.2 2D vs. 3D for H=h=1

CEM

Figure 4.1 shows the 2D and 3D model which are employed for solving the EIT problem. Based on the requirements for the 3D model discussed earlier, the 3D model is composed of multiple layers of the 2D model aligned in z-coordinate having unit height and unit height electrodes. In the 2D and 3D mesh, the element width (Δw) is equal to 0.05, which is equivalent to 32 layers in xy plane in a circular model with unit radius; and in the 3D mesh, element height (Δz) is equal to 0.025, which is equivalent to 41 layers in z direction in a tank with unit height (The 3D mesh is composed of 41 layers of identical 2D mesh).

For the model depicted in figure 4.1, a 16-electrode system is employed. The electrode width for this model is about 0.1 (0.0982); hence, the complete electrode model is employed.

Using opposite current pattern stimulation, the solution of an empty 2D model and 3D model (tank) is calculated. Figure 4.2 shows the maximum relative difference between the 2D solution and the solution of different layers of the 3D model ($\varepsilon(\%) = \max_{xy} |V_{2D}^{CEM}(x, y) - V_{3D}^{CEM}(x, y, z)|$). From this figure, it is observed that the result of the 3D forward solver for a 3D tank even with the full height electrodes is not

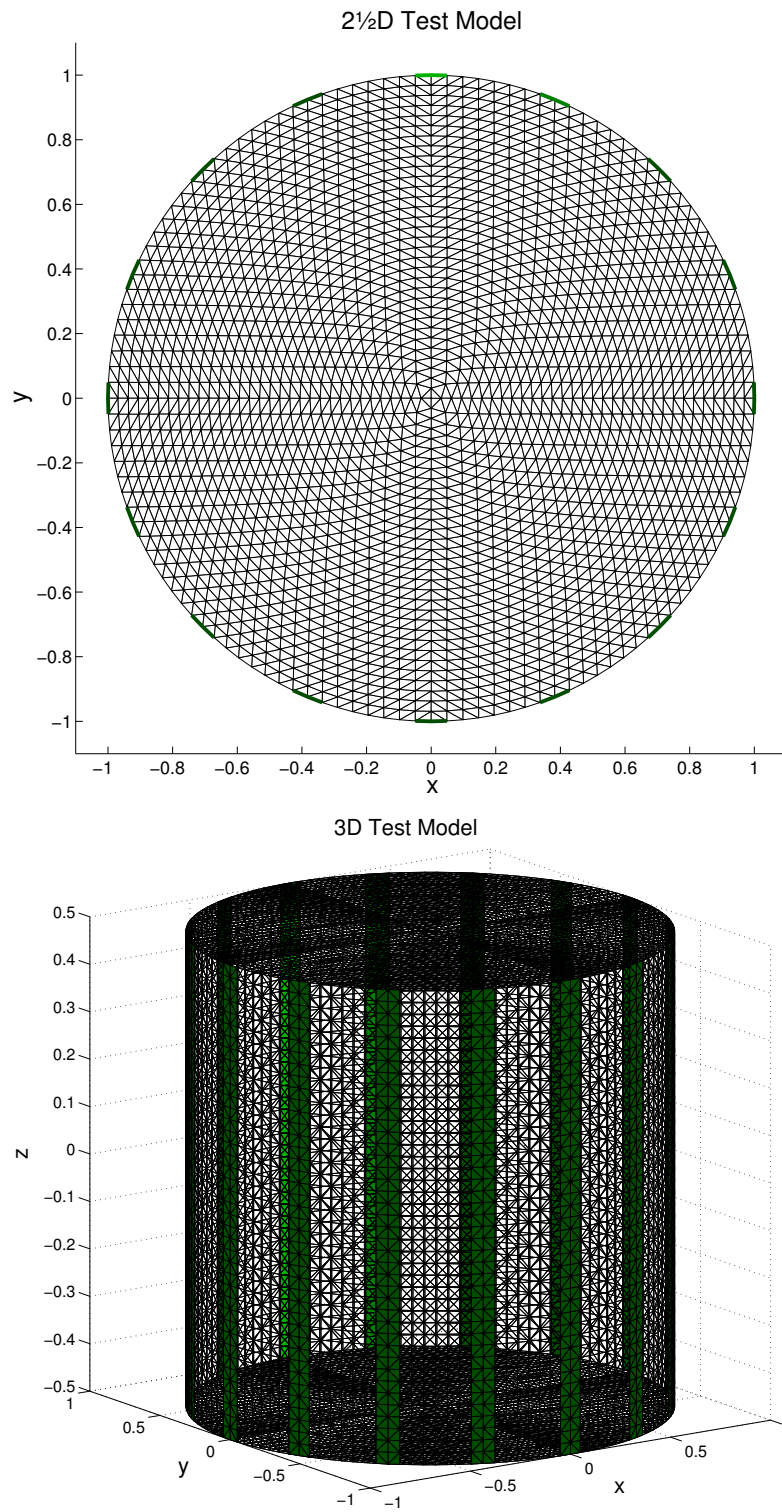


Figure 4.1: 2D and 3D mesh

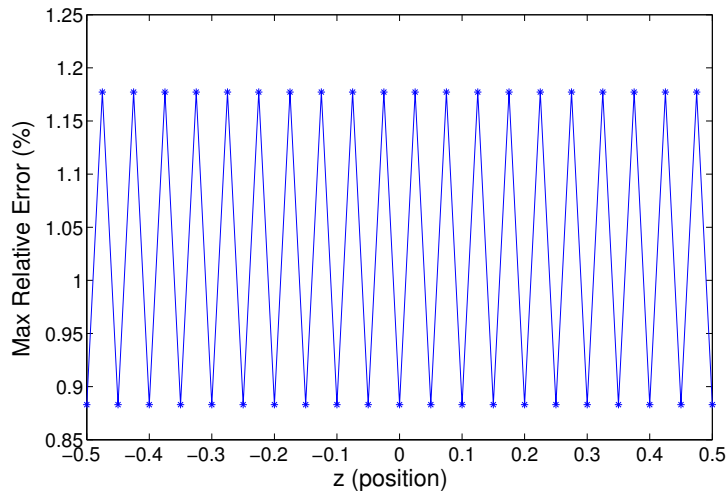


Figure 4.2: Max relative difference between the 2D solution and 3D solution on different layers for the 3D mesh in figure 4.1 with 41 layers, $H = h = 1$ - Complete Electrode Model. $\varepsilon(\%) = \max_{xy} |V_{2D}^{CEM}(x, y) - V_{3D}^{CEM}(x, y, z)|$

completely constant since it bounces between two different values. Although the amount of difference is not large it could be significant. The maximum absolute error between the 2D and 3D ($\varepsilon = |V_{2D} - V_{3D}|$) is on the order of 0.012 while the mean of absolute error is on the order of 7×10^{-5} . If the difference voltages at electrode positions excluding the current drive electrode for different z-layers in 3D (gradients at different z-layers) are compared with the 2D gradient for all stimulation patterns, the plot would be a same plot as figure 4.2 with maximum of 0.273 % and minimum of 0.243 %.

PEM

If in the model depicted in figure 4.1, the point electrode model is employed (electrode width is zero; hence, for 3D mesh each electrode is a line with unit height), then the maximum relative difference between the 2D solution and the solution of different layers of the 3D model would be figure 4.3(left) in which the value of the relative difference at both ends is 25.7 %.

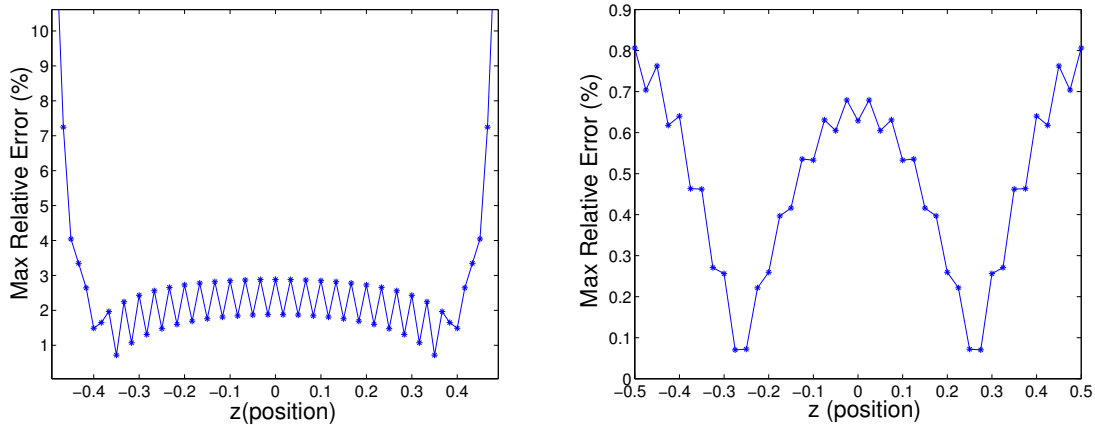


Figure 4.3: Max relative difference between the 2D and 3D on different layers for (left) solution (right) gradients, $H=h=1$ - Point Electrode Model vs. line electrode in 3D. $\varepsilon(\%) = \max_{xy} |V_{2D}^{PEM}(x, y) - V_{3D}^{line}(x, y, z)|$

Figure 4.3(right) shows maximum relative difference that occurs between the 2D gradients and 3D gradients at different layers. Gradients are important because an average of them forms the voltage measurements. Figure 4.3 shows that employing point electrode model leads to inaccurate solutions for nodes under the electrode especially at the vertical electrode ends or tank edges. This could be due to effects in the implementation of the line electrode model. However, this phenomena does not occur in the complete electrode model.

4.1.3 Comparing Electrode Voltages

As discussed previously, in a 3D mesh with the complete electrode model, the electrode voltage (the voltage of the virtual node assigned for each electrode) is a weighted average of the voltage of all nodes under the electrode. Figure 4.4 shows the difference between electrode voltages (E_i) of all stimulation pattern for the 2D and 3D model depicted in figure 4.1. Here, it is observed that the difference between the electrode voltages for 2D and 3D model is much smaller than the differences between the 2D solution and 3D solution in various layers in z-coordinate depicted in figure 4.2.

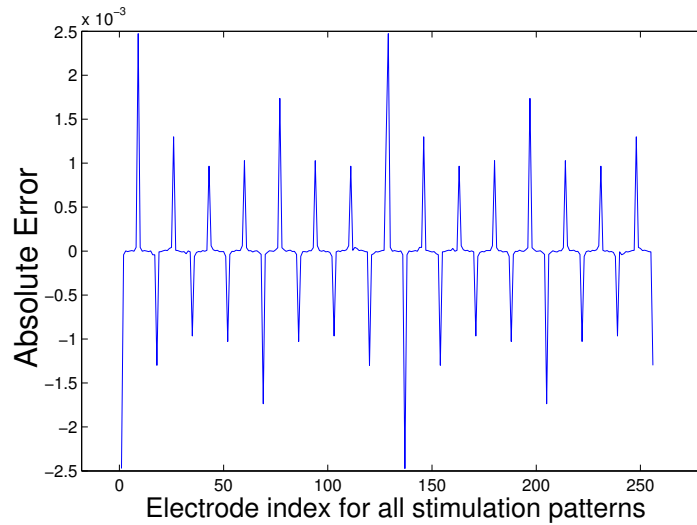


Figure 4.4: Difference of Electrode voltage between 2D and 3D for all stimulation patterns, $H = h = 1$. $\varepsilon = |E_{l_{2D}}^{CEM} - E_{l_{3D}}^{CEM}|$

For example, the maximum relative difference reported in figure 4.2 is 1.18 % and the minimum value for the same mesh is 0.88 %; while the maximum relative difference of the electrode voltages between 3D and 2D is 0.19% for a mesh employing complete electrode model having electrode width $W = 0.1$. The electrode voltage in the 2D model is the voltage of only one z-layer under the electrode. However, for the 3D mesh, the voltage on each electrode is a weighted average of the voltage of all electrode nodes in different z-layers. The reason why the difference between the 2D solution and the average of voltages in different layer is lower than the difference between the 2D solution and each individual layers depicted in figure 4.2 is that for some layers the value of the difference is negative while for other layers this value is positive; hence, they cancel each other out. In addition, the error in the voltage of electrode nodes is less than the maximum error of the solution. Employing line electrodes in 3D and point electrode in 2D (electrode width is 0), the maximum relative difference of the electrode voltage is 1.2% (0.02). One possible explanation of this could be as a result of the end effect in implementation of line electrode in the

3D solver.

Comparing Measurements

Here, the measurements are defined as the actual measured (gradient) voltages; which are the difference voltages between adjacent passive electrodes. In other words, measurements are the difference voltages measured between those electrodes which are not driving current. For both the CEM and line electrode model, the maximum absolute error in the measurements between the 3D and 2D is on the order of 5×10^{-5} (0.023 %). The difference between the measurement is much less than the electrode voltage because the maximum errors belong to the active electrodes and the voltage of active electrodes for each current pattern is not involved in the measurements. In addition, reducing the element size would result in lower value for this difference.

4.2 Variation of the 3D element height

Normally, it is expected that the 3D solution is more accurate than the 2D solution; hence, the 3D solution is considered as the reference solution. However, the 3D solution would also change with variation of the element height in the 3D mesh. Here, the variation of the 3D solution is investigated where a unit height tank is employed as the 3D model using different height for elements. For the 3D model, the element size and also the coordinates of the elements in the xy plane are fixed and equal to a 2D model. Using different number of layers for the z-coordinate in a unit height tank, the only parameter that changes is the element height.

4.2.1 Variations in the 3D solution

Figure 4.5 shows the relative difference between the 3D solutions calculated using different 3D mesh having different element height for three electrode heights ($h =$

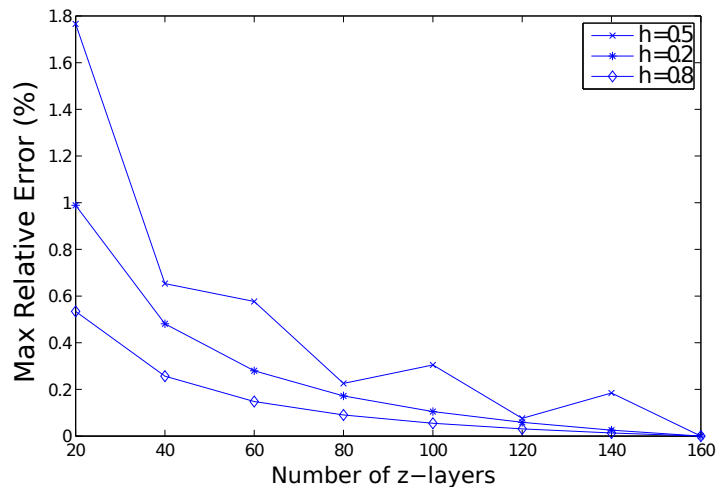


Figure 4.5: Variations of 3D solution by changing element height Δz compared to the last value plotted for different electrode height (h). $\varepsilon(\%) = \max|V_{3D}^{CEM}(\Delta z) - V_{3D}^{CEM}(\Delta z = \frac{1}{160})|$

$\{0.2, 0.5, 0.8\}$). For the purpose of comparison, the 3D solutions are compared to its last value which has the smallest element height (Δz). In this investigation, 32 layers used in xy plane in a circular model with unit radius which corresponds to $\Delta w = 0.05$. This figure demonstrates that the 3D forward solution changes when the element height is reduced. The number of layers in z-coordinate for the 3D mesh forms the x-axis of the plot which is equal to $\frac{H}{\Delta z}$.

4.2.2 Variations in the 3D-2D difference

In the case of unit height electrode ($h = 1, H = 1$), it is possible to compare the 3D solution directly with the 2D solution. In this case, the variation in measurements for a 3D mesh with $h = 1$ employing 40 z-layers with respect to the same 3D mesh employing 160 z-layers ($|\Delta V_{3D}(\Delta z = \frac{1}{40}) - \Delta V_{3D}(\Delta z = \frac{1}{160})|$) is 0.03% using line electrodes and 0.06% using a CEM with $W = 0.1$; here, the error in measurement is itself lower than the previous situation calculated for the whole 3D solution.

Figure 4.6 plots the difference between the 2D solution and the 3D solutions

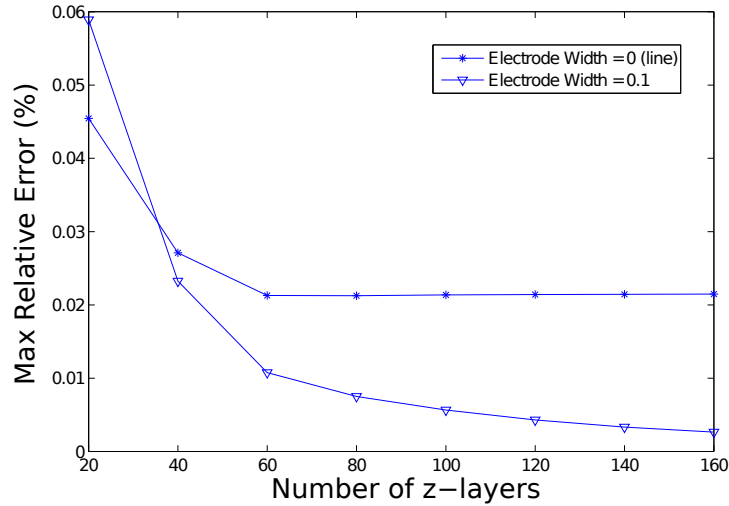


Figure 4.6: Maximum relative difference in measurements between 2D and different 3D solutions calculated for a same 3D mesh with different element height (Δz) for Electrode Width (W) = $\{0, 0.1\}$. $H = h = 1$, $\varepsilon(\%) = |\Delta V_{2D} - \Delta V_{3D}(\Delta z)|$

calculated for a same 3D mesh with different element height (Δz). Obviously, here, the 2D solution is constant since the 2D model is not dependent on Δz . Hence, in this case, it can be concluded that the 3D solution converges toward the 2D solution. In the case of line electrode modelling for the 3D mesh, it is observed that there is not much improvement by employing more z-layers than 60 z-layers while there still exists a small amount of error. However, for the 3D model with $W = 0.1$ the 3D solution keeps converging to the 2D solution. This again raises the issue that something might be wrong in the implementation of the line electrode in a 3D mesh.

4.3 The first term of $2\frac{1}{2}$ D

This section aims to relieve the assumption of $H = 1$. By analyzing the $2\frac{1}{2}$ D solution, a normalized 2D solution (normalized by the domain height H) is introduced; this makes it possible to consider more general cases where $H \neq 1$.

4.3.1 Analysis of the terms of $2\frac{1}{2}D$

In 2D models, the conductivity has a unit of Ω^{-1} while in 3D models the conductivity has a unit of $\frac{\Omega^{-1}}{m}$. Therefore, the 2D conductivity in a surface is equal to the conductivity of a 3D volume having unit height ($H=1$) and the same surface as base. In other word, in transforming any 3D conductivity into 2D conductivity for a 2D model, the equivalent 2D conductivity would be the 3D conductivity multiplied by the 3D domain height. Therefore, if same conductivity values are used for the 2D and 3D model, the 2D solutions should be divided by the 3D domain height.

Also, looking at equation (4.1) again which is written for a 3D model using full height electrodes $h = H$, we would understand that it is, in fact, the solution of a 2D model normalized by the domain height. Hence, the first term of the $2\frac{1}{2}D$ is a normalized 2D solution regardless of the electrode height.

The 2D forward solution is the result of a matrix inversion for calculating node voltages from the injected current which is expressed in (4.2).

$$V_{2D} = S^{-1}I \quad (4.2)$$

where S is the system matrix of the mesh, I is the injected currents and V_{2D} is the voltage of the nodes. The solution (voltage) of the forward problem using $2\frac{1}{2}D$ modelling is calculated by substituting the solution of equation (3.41) for each harmonic n into the synthesis equation (3.2b). Doing this yields:

$$V_{2\frac{1}{2}D} = S'_{0^{-1}}I_0 + \sum_{n=1}^{\infty} S'_{n^{-1}}I_n \cos\left(\frac{n\pi}{z_m}z\right) \quad (4.3)$$

where S'_0 is equal to the 2D system matrix S because $S'_0 = S + \frac{0\pi}{z_m}R$ according to (3.29); and I_n together with I_0 are defined in (3.14). Substituting the current sequence results in:

$$V_{2\frac{1}{2}D} = \frac{1}{H}S^{-1}I + \sum_{n=1}^{\infty} \frac{2}{n\pi h} \sin\left(\frac{n\pi h}{2z_m}\right) \cos\left(\frac{n\pi}{z_m}z\right)S_n'^{-1}I \quad (4.4)$$

where S'_n is the $2\frac{1}{2}D$ system matrix for the n^{th} harmonic defined by (3.29) or more precisely later in (5.5), I is the injected current, $V_{2\frac{1}{2}D}$ is the voltage of the nodes for the $2\frac{1}{2}D$ model, z is the position in z coordinate where the solution is to be calculated and z_m is one half of the domain height, i.e. $\frac{H}{2}$. Comparing equation (4.4) with (4.2) also shows that the first term of the $2\frac{1}{2}D$ solution is the 2D solution normalized by the domain height as

$$V_{2\frac{1}{2}D} = \frac{1}{H}V_{2D} + \sum_{n=1}^{\infty} \frac{2}{n\pi h} \sin\left(\frac{n\pi h}{2z_m}\right) \cos\left(\frac{n\pi}{z_m}z\right)S_n'^{-1}I \quad (4.5)$$

4.3.2 Normalized 2D vs. 3D

The difference between the 2D and 3D model was illustrated in figure 4.2 and 4.3 which express that the voltage solution inside a 3D mesh is not constant along with z direction. Since the 2D solution is constant, it is clear that the 3D solution for a tank with full height electrodes varies at different z -layers. The variant difference could be as the result of closed end-surfaces in the unit height 3D tank specially in the case of point (line) electrode model; or in CEM, it could be due to the fact that for this specific meshing technique the nodes are symmetric decussately in z -layers. For example, for the model depicted in figure 4.1, an element of injected current density at $z = 0.45$ faces with a closed path at $z = +0.5$ while an element of current density faces equal paths at $z = 0$.

Figure 4.7 shows the maximum difference between the normalized 2D solution and the 3D solution for various tank height all having same element height for a 3D model with line electrodes. It demonstrates that the difference between the normalized 2D solution and the 3D solution reduces as the height of the tank increases. However,

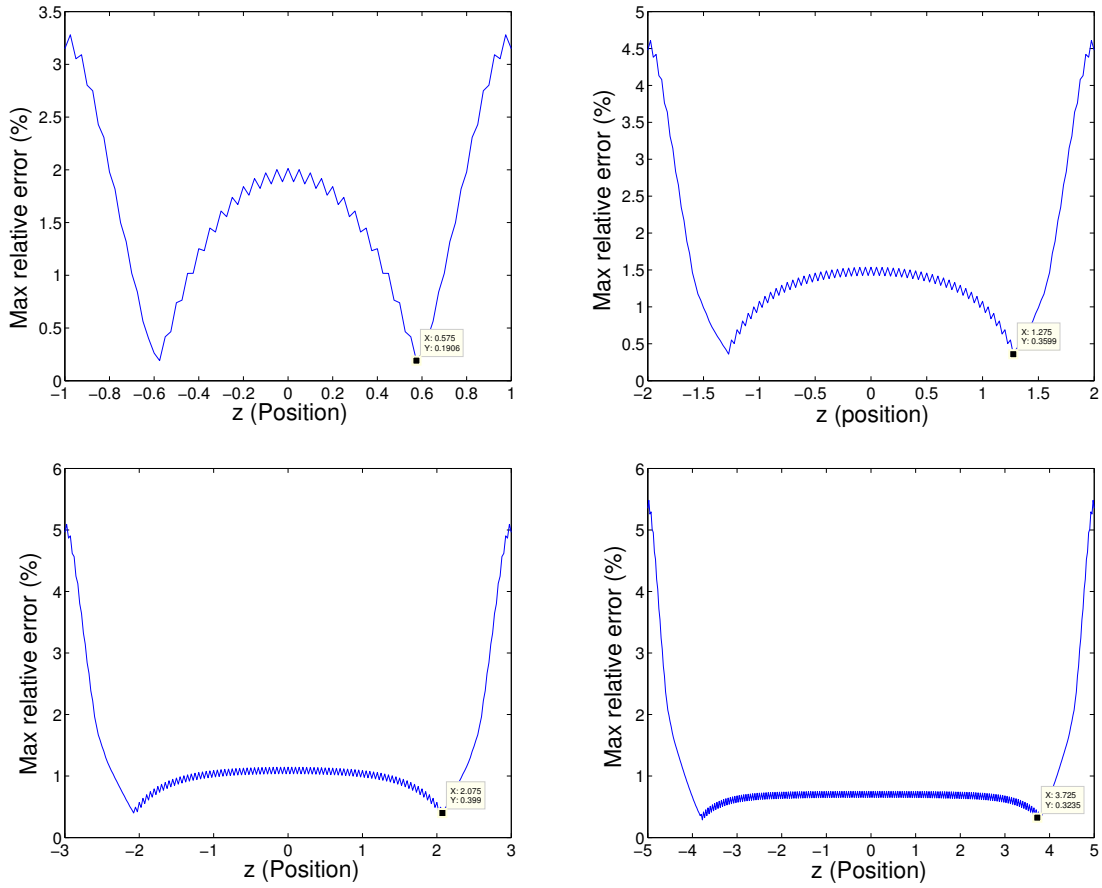


Figure 4.7: Max relative difference between normalized 2D and 3D solutions in different z-layer calculated for $H=h=\{2, 4, 6, 10\}$ - PEM. $\varepsilon(\%) = \max_{xy} |V_{2D}^{PEM}(x, y) - V_{3D}^{line}(x, y, z)|$

when non-zero width electrodes are used in the model, the maximum difference between the normalized 2D solution (therefore, $2\frac{1}{2}$ D solution) and the 3D solution would not change by increasing the tank height.

4.4 $2\frac{1}{2}$ D vs. 3D and 2D

The $2\frac{1}{2}$ D solution is calculated based on a 2D model for the purpose of reaching the 3D solution. Comparing the $2\frac{1}{2}$ D solution with the 3D solution and the 2D solution shows that $2\frac{1}{2}$ D method is successful in modifying the 2D solution toward the 3D

solution.

Figure 4.8(up) illustrates the voltages of the electrodes calculated by 3D, 2D, $2D/H$ (first term of $2\frac{1}{2}D$) and $2\frac{1}{2}D$ modelling for the fourth stimulation pattern at different z layers for a 3D mesh containing 160 z -layers with height $H = 2$ and electrode height $h = 0.4$ employing line electrodes. This mesh is a sample mesh used for comparison without loss of generality (for H) which is used in the next investigations from this point. Figure 4.8(down) illustrates the measurement (gradients) voltage for the same mesh. In both figures, the $2\frac{1}{2}D$ and the 3D solutions lie on top of each other (with a possible mismatch) while the 2D and the normalized 2D (first term of the $2\frac{1}{2}D$) solution are far from the 3D solution.

Figure 4.9 illustrates the voltages of the nodes at the electrode positions calculated by 3D and $2\frac{1}{2}D$ modelling for the fourth stimulation pattern. These plots are not the voltage of (on) the electrodes; the plots are the voltages of the nodes under the electrodes at different z layers ($V_{El}(z)$) for the same mesh. Figure 4.8(up) can be interpreted as a vertical cut (cross-section) of figure 4.9 at $z = 0$. Both figures 4.8 and 4.9 confirm that the $2\frac{1}{2}D$ solver is working well in modifying the 2D solution toward the 3D solution.

Figure 4.10 illustrates the difference between the 'gradient voltages at different z -layers' ($\Delta V_{El}(z)$) calculated using 3D, 2D and $2\frac{1}{2}D$ modelling. Here, the gradient voltages calculated for different z -layers (z position) refer to the difference voltages between adjacent nodes under the passive electrode at different z -layers (z position). The top-left figure shows the difference between the 3D and $2\frac{1}{2}D$ at different z -layers. The figure at down shows the difference between the 3D and 2D; and the figure at top-right illustrates the difference between 3D and first term of $2\frac{1}{2}D$ ($2D/H$). It can be observed from these figures that the error between the 3D and 2D solutions is much larger than the error between the 3D and $2\frac{1}{2}D$ solutions. The large difference between the 3D and 2D solutions is due to the fact that the 2D solver employs a

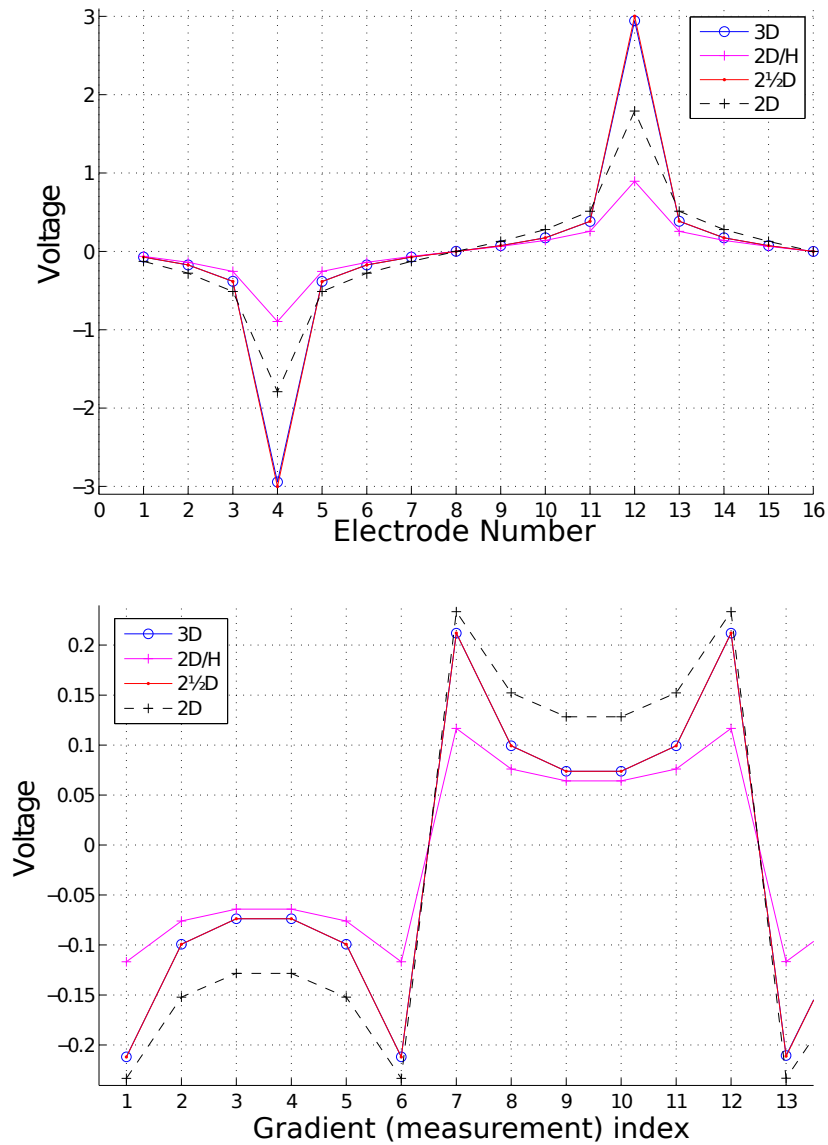


Figure 4.8: Comparing 3D, 2D, 2D/ H (first term of $2\frac{1}{2}D$) and $2\frac{1}{2}D$ solutions for (up) electrode voltages and (down) measurements (adjacent difference electrode voltages), $H = 2$, $h = 0.4$, $n_{max} = 50$ and $\Delta z = \frac{2}{160}$ - PEM (line in 3D) ($W = 0$)

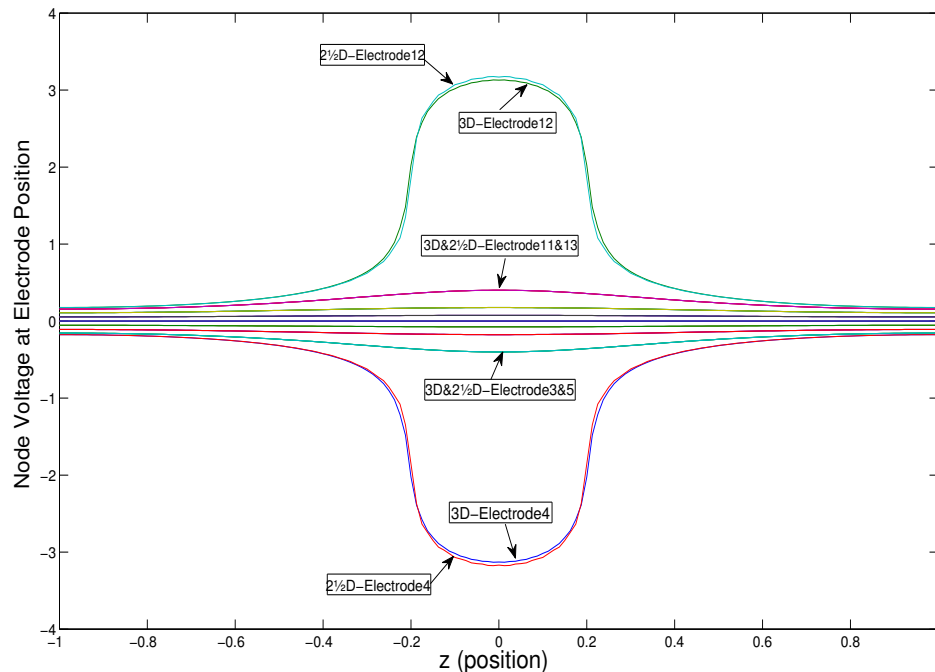


Figure 4.9: Comparing 3D and $2\frac{1}{2}$ D solutions for nodes at electrode positions at different z -layers $V_{El}(z)$ - $H = 2$, $h = 0.4$, $W = 0$, $\Delta z = \frac{2}{160}$ and $n_{max} = 50$.

2D mesh and does not have any information about the electrode height (h) and the tank Height (H). Improving this large difference is the goal of the $2\frac{1}{2}$ D method given the electrode height (h) and tank Height (H) while still employing a 2D mesh. The small difference between the 3D and $2\frac{1}{2}$ D in the top-right figure compared with the other two figures demonstrates how the $2\frac{1}{2}$ D improves the 2D solution toward the 3D solution. Moreover, the small error in the top-left figure compared to the large error in the top-right figure, demonstrates how higher order $2\frac{1}{2}$ D terms modify the solution of the first $2\frac{1}{2}$ D term ($2D/H$) and improves the accuracy toward the 3D solution. The maximum relative error in the measurement voltages between the 3D and $2\frac{1}{2}$ D is 0.42% (0.0009) for this case.

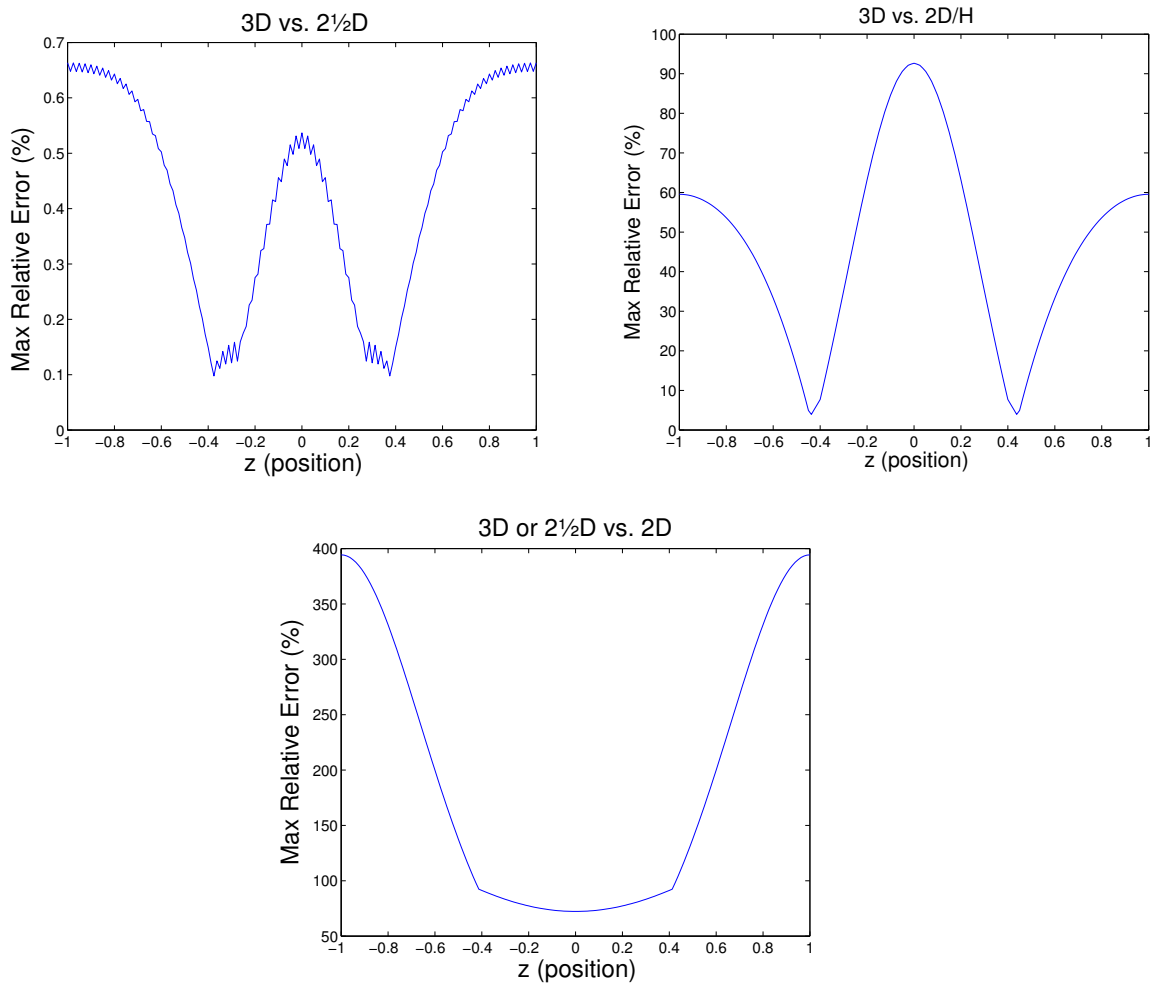


Figure 4.10: Difference between gradient at different z -layers for (left) 3D vs. $2\frac{1}{2}$ D (right) 3D vs. 2D/H (down) 3D or $2\frac{1}{2}$ D vs. 2D, $H = 2$, $h = 0.4$, $n_{max} = 50$ for $2\frac{1}{2}$ D and $\Delta z = \frac{2}{160}$. $\varepsilon(\%) = |\Delta V_{El}^A(z) - \Delta V_{El}^B(z)|$

In CEM, the measurements are calculated as the difference between adjacent passive electrode voltage which is the voltage of the virtual node assigned for each electrode. The $2\frac{1}{2}$ D solution is computed for a given z -layer (z position); hence, in $2\frac{1}{2}$ D solution, the voltages at electrode position could be calculated (therefore, known) for each z , yet the 'electrode voltage' (voltage of the virtual node considered for each electrode) needs to be calculated the same way as it is calculated in the 3D model. Assigning a virtual node for each electrode and implementing the way the voltages are averaged under each electrode in complete electrode model express in section 3.9, the 'electrode voltage' for the $2\frac{1}{2}$ D solution is calculated. Figure 4.11 shows the measurement voltages calculated by 3D, 2D and $2\frac{1}{2}$ D modelling for a mesh with height $H = 2$, electrode height $h = 0.4$ and electrode width $W = 0.1$. Maximum relative error in the gradients is 3% (0.006) for this case.

4.5 Sources of Errors

As it is observed from figure 4.9, there exists a small difference between the 3D solution and $2\frac{1}{2}$ D solution. This amount of difference might be from various sources which are discussed here.

4.5.1 3D Interpolation Function

From the result of the analysis illustrated in figure 4.5, it has been observed that the 3D solution itself changes by using different heights for 3D elements. This shows that the 3D solution improves as smaller element height is used while the $2\frac{1}{2}$ D solution is not dependent on the 3D element height since the element height is not a parameter in the $2\frac{1}{2}$ D method where a 2D model is employed. Figure 4.12 shows the difference between the measurement voltages calculated by 3D and $2\frac{1}{2}$ D modelling using different element height for a unit radius mesh with 32 layers in the xy plane. It is observed

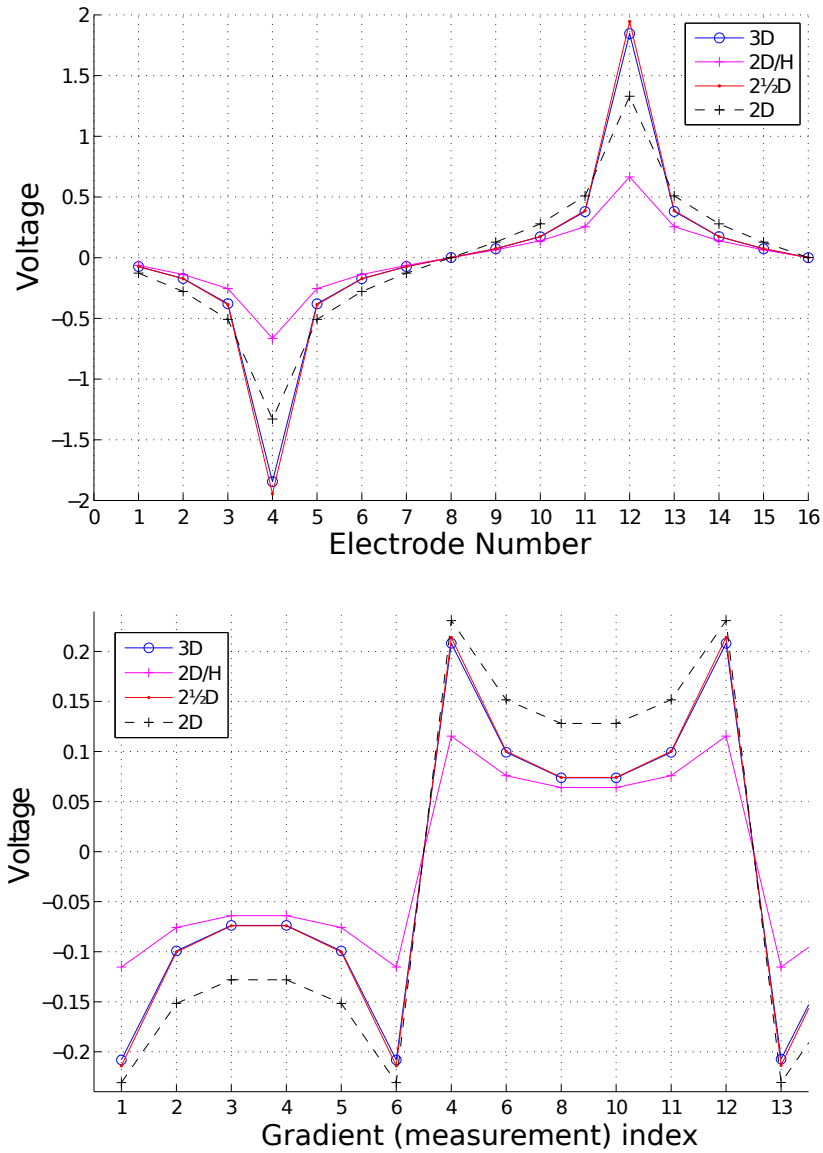


Figure 4.11: Comparing 3D, 2D, 2D/H (first term of $2\frac{1}{2}$ D) and $2\frac{1}{2}$ D CEM solutions for (up) electrode voltages and (down) measurements - CEM ($W = 0.1$, $H = 2$, $h = 0.4$, $n_{max} = 50$ and $\Delta z = \frac{2}{160}$)

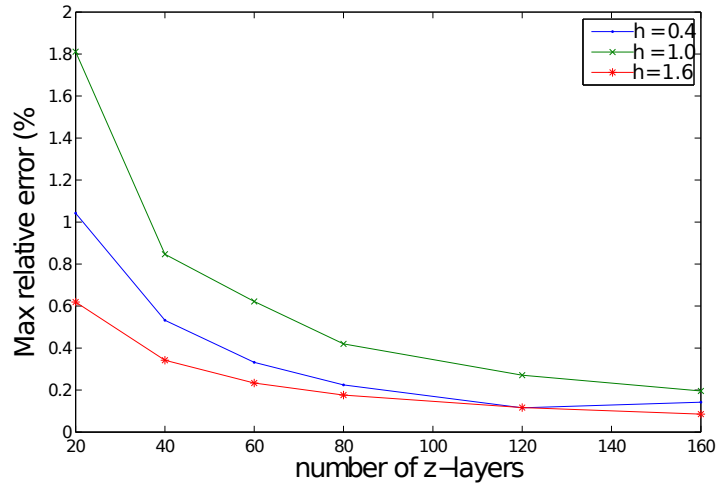


Figure 4.12: Maximum relative difference between the 3D and $2\frac{1}{2}$ D measurements for different electrode height using different element height (Δz) or (number of z-layers) for $h = \{0.4, 1.0, 1.6\}$, $H=2$, $n_{max} = 50$

that the difference between the 3D and $2\frac{1}{2}$ D solution is reduced (an small increment observed for $h = 0.2$ at $\Delta z = \frac{2}{160}$ could be due to instability). While the $2\frac{1}{2}$ D solution is not dependent on the 3D element height it can be concluded that the 3D solution converges to the $2\frac{1}{2}$ D solution.

The above mentioned error in the 3D solution can be due to truncation error in 3D interpolation functions since the 3D domain is segmented into a finite number of elements and nodes where inside each element the potential is approximated by linear shape functions.

4.5.2 Injected Current Pattern

For calculating the $2\frac{1}{2}$ D boundary condition in equation (3.13), it is assumed that the injected current is uniformly distributed under active electrode; therefore, the injected current pattern (or current density) in z direction is a rectangular pulse which is constant under the electrode region and zero outside of the electrode region having an area equal to the current amplitude depicted in figure 4.13(left).

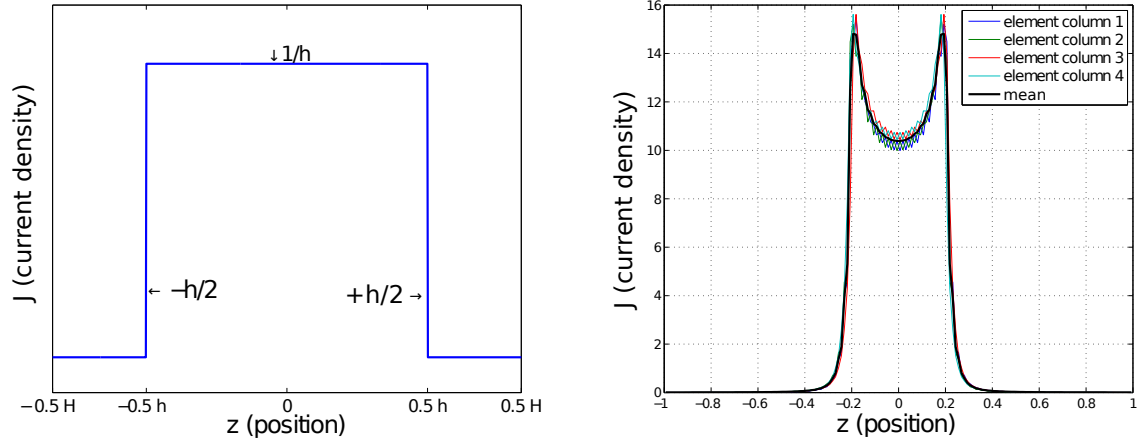


Figure 4.13: Distribution of current density (left) assumed for $2\frac{1}{2}$ D (right) real 3D mesh with complete electrode model - $\Delta z = \frac{2}{160}$ and electrode width $W = 0.1$

However, an analysis for calculating the current density under the active electrode demonstrates that in the 3D mesh employing complete electrode model, the voltage is not constant under the electrode region. The procedure of calculating the current distribution in the boundary is expressed in Appendix B. Figure 4.13(right) shows the distribution pattern of the injected current density under the electrode for a 3D mesh using complete electrode model. As can be observed, the current distribution under the electrode is not constant for CEM.

4.5.3 2D-based Complete Electrode Modelling

Generally, the $2\frac{1}{2}$ D solution is a 2D-based solution. This means that the system matrix of the $2\frac{1}{2}$ D method is derived from the 2D system matrix and therefore it uses the 2D complete electrode complementary elements (A_Z , A_W and A_D ; referring to chapter 3.4) in its system matrix keeping the same size as 2D; hence, only the relation in xy plane between the nodes under each electrode is modelled in constructing the system matrix. However, in a 3D mesh employing complete electrode model, the relationship of all of the nodes in different z-layers is modelled in the system matrix

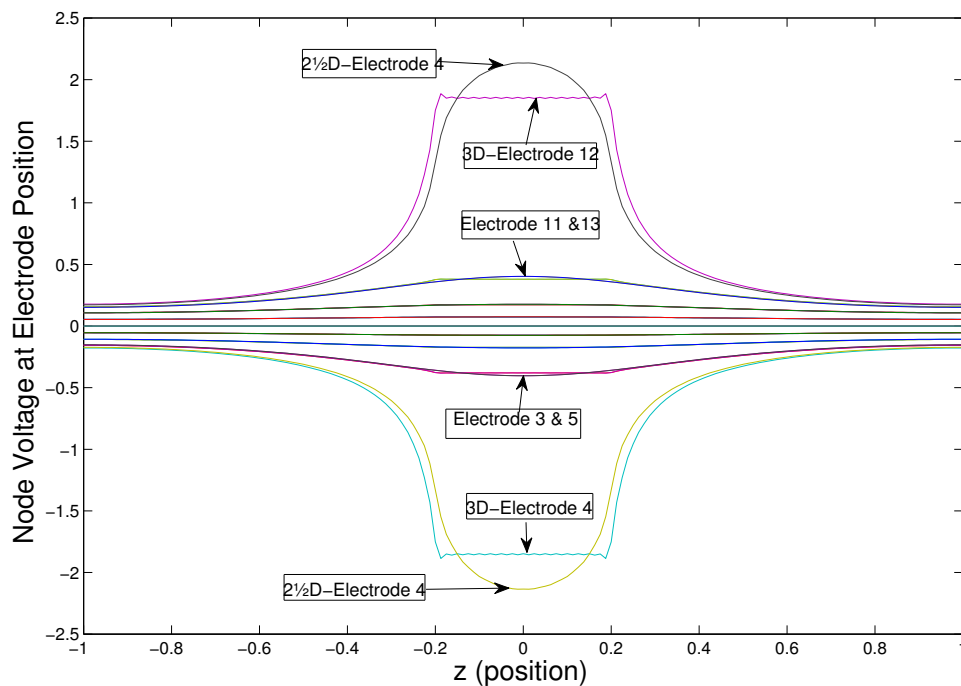


Figure 4.14: Comparing 3D and $2\frac{1}{2}$ D solutions for nodes at electrode positions at different z -layers electrode width (W) = 0.1, H = 2, h = 0.4, Δz = $\frac{2}{160}$ and n_{max} = 50.

which is used to solve node voltages from injected current.

Figure 4.14 illustrates the voltages of the nodes at the electrode positions calculated by 3D and $2\frac{1}{2}$ D modelling for the fourth stimulation pattern at different z layers for the same mesh as the experiment resulted in figure 4.9 (height H = 2 and electrode height h = 0.4) this time with W = 0.1. This figure is plotted for the center node under each electrode for both 3D and $2\frac{1}{2}$ D. The difference between the 3D and $2\frac{1}{2}$ D is less for the side nodes. Unlike the line electrode case, here, it is observed that the $2\frac{1}{2}$ D is not following the 3D solution attributes for each layer in z coordinate. However, taking the average of nodes under each electrode at different z -layer would result in much smaller difference in the electrode voltage and measurements.

In order to simulate the 3D complete electrode model for $2\frac{1}{2}$ D completely, we

suggest that the average of the nodes under the electrode at different z-layer is calculated the same way as it is computed in 3D. However, this approach would not be a complete solution due to the following reasons:

1. The complete electrode modelling is not done completely inside of the system matrix. This means that due to employing 2D-based models only the relation between the nodes under the electrodes in xy plane is considered inside the system matrix while the relation between the nodes under the electrodes in different z-layers is modelled after the $2\frac{1}{2}$ D solution of all layers are calculated which would be a post processing approach.

2. The coefficients or the weights by which the voltages of the nodes under each electrode are averaged are unknown from the $2\frac{1}{2}$ D solver point of view unless one constructs a 3D system matrix and a 3D mesh in order to find the weights. The situation gets worse when adding that these weights depend on the 3D meshing method, how each node is connected to its neighboring nodes, and on which type of the nodes the electrode ends lie in the boundary which seems to be random. Hence, a rule of thumb would be calculating the electrode voltage (voltage of the virtual node assigned for each electrode) from the $2\frac{1}{2}$ D solution on each z-layer that electrodes lie, in which only the relation in xy plane is considered; then treating these electrode voltages as a 3D mesh with line electrode since there is only one voltage for each z-layer; and average them using the weights for line electrode model which are: $[1, 2, \dots, 2, 1]/2(t - 1)$ where t is the number of assumed z-layers which lie under the electrode. Hence, another problem would be the number of z-layers in the 3D mesh which is another unknown from the the $2\frac{1}{2}$ D solver points of view.

In addition, in imposing the complete electrode model for $2\frac{1}{2}$ D equation in the same way done for the 2D mesh (see equation (3.58)), there was a assumption that the Kirchhoff's voltage law (KVL) is valid. Although this KVL is valid for the 2D and 3D situation, the validity should be further investigated in the $2\frac{1}{2}$ D case since the

harmonic voltages are not real potentials. However, it seems that KVL and KCL are valid for harmonic voltages and currents.

Chapter 5

Implementation and Complementary Discussion

In this chapter the structure of the main modules which are developed to expand $2\frac{1}{2}$ D capability to the EIDORS project are described. First, the 'fwd_solve' and 'calc_system_matrix' modules are explained. Next, it is shown how the speed of the algorithm can be improved by keeping the constant parts out of the 'for' loop. In section 5.3, it is discussed that how many terms are sufficient to be solved in order to satisfy a reasonable accuracy. The last section provides a comparison between the 3D solver and $2\frac{1}{2}$ D solver memory and computation performance.

5.1 The Structure of Modules

The following modules were developed in order to append $2\frac{1}{2}$ D FEM ability to the EIDORS project:

The 'fwd_solve' module is a module where the forward equation is solved. In this module, first, for each $2\frac{1}{2}$ D partial differential equation, a system matrix is reconstructed. The module 'calc_system_matrix' calculates the system matrix for the

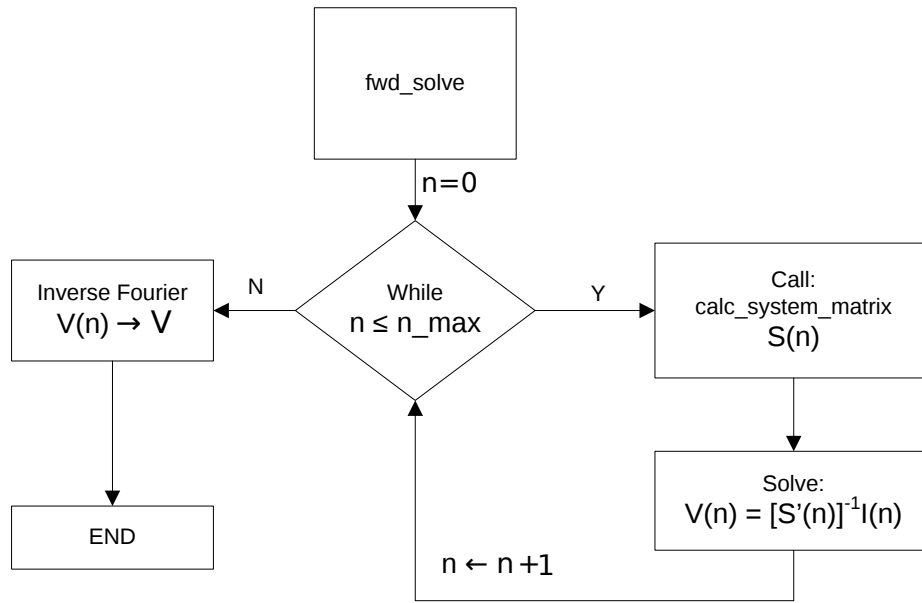


Figure 5.1: Block diagram of the 'fwd_solve' module

entire mesh for each harmonic n ; which is explained later. Having the system matrix of each PDE's expressed in (3.5), a forward equation is solved for each spatial frequency in order to compute the nodal voltages from the driving currents. This procedure continues until the truncation point, n_{max} , is reached or a sufficient number of equations are solved based on an error criterion; then, based on the synthesis equation (3.2b), the inverse Fourier transform of the nodal voltages is computed which forms the $2\frac{1}{2}$ D forward solution. The block diagram of different stage of this module has been plotted in figure 5.1.

In order to calculate the voltages (or gradients) at any height, the solution of all harmonics is required. For computing the $2\frac{1}{2}$ D solution at a different height, only the Fourier summation is required to be performed again for the new z . Therefore, the output of the module could be extended to calculate the average of the voltage of all nodes under each electrode without solving extra equations.

The 'calc_system_matrix' module, calculates the $2\frac{1}{2}$ D 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. In this work, the system matrix for the entire mesh is computed by calculating the local system matrix for each element and construct a connectivity matrix for the entire mesh based on the connectivity map of the elements. When the local system matrices are all calculated, then the system matrix for the entire mesh is computed by equation (3.42) which is written again here:

$$S'(n) = C^T S'_E(n) \Sigma C \quad (5.1)$$

where $S'_E \Sigma$ is a diagonal concatenation of local system matrices, S'^k , in which the conductivity of each element is also included (see in chapter 3), C is connectivity map matrix and 'T' represents the transpose operation. The $2\frac{1}{2}$ D system matrix S' is a function of the spatial frequency $(\frac{n\pi}{z_m})$. The local sensitivity matrix for each element is constructed by the 'calc_local_stiffness_matrix' and the connectivity map is the relation between the node index inside each element and the general node index for the entire mesh. The block diagram of different stage of this module is plotted in figure (5.2).

In the 'calc_local_stiffness_matrix' module, the matrix S'^k is calculated for each element based on the solution of the integrals expressed in (3.29) given the following parameters: 'n', 'z_m', and the nodal position of that element. The details of computing the integral is expressed in chapter 3 and appendix A.

$$S'^k_{ij} = S^k_{ij} + \left(\frac{n\pi}{z_m}\right)^2 R^k_{ij} = \int_{E_k} \nabla \phi_i \cdot \nabla \phi_j + \left(\frac{n\pi}{z_m}\right)^2 \phi_i \phi_j d\Omega \quad (5.2)$$

5.2 Improving the Speed of the Algorithm

The $2\frac{1}{2}$ D local system matrix, S'^k , consists of two parts which is expressed again in (5.2). The first part is the matrix S^k which is easy to be calculated since for linear

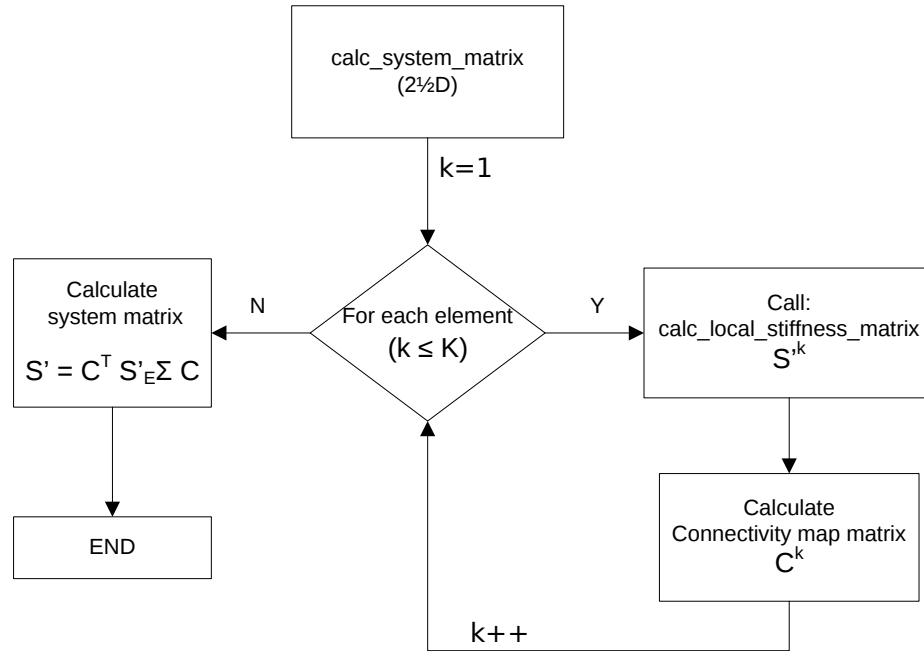


Figure 5.2: Block diagram of the 'calc_system_matrix' module - K is the total number of elements in the mesh

interpolating functions, ϕ_i , the gradients are constant over the integral. However, calculating R^k , the second part, is more difficult and requires computing the integrals of the following integrands for all elements of the whole mesh which are taken over each element separately: x , y , x^2 , xy , y^2 . Yet, both S^k and R^k are not dependent of the spatial frequency $n \frac{\pi}{z_m}$ and are constant for a certain element geometry. In other words, for each harmonic n , only the coefficient of the local residual matrix R^k is a dependent of n and z_m ; therefore, a residual matrix R for the entire mesh could also be constructed from all local residual matrices, R^k , and the connectivity matrix the same way mentioned previously in (3.42) for the S' (or S) as:

$$R = C^T R_E \Sigma C \quad (5.3)$$

where $R_E \Sigma$ is a diagonal concatenation of local Residual matrices, R^k including the conductivity of each element. The proof of the previous argument is based on substituting equation (5.2) into equation (5.1) and taking the summation outside of the multiplication. By doing this, the following expression is derived for the system matrix of a $2\frac{1}{2}$ D problem for each harmonic n :

$$S'(n) = C^T S_E \Sigma C + \left(\frac{n\pi}{z_m}\right)^2 (C^T R_E \Sigma C) \quad (5.4)$$

The first matrix term in equation (5.4) is a normal 2D systems matrix, S , and the second matrix term excluding the scalar coefficient was defined as R in (5.3) called 'Residual' matrix. Due to the fact that the residual matrix itself is independent of the harmonic n , it could be calculated only once for the whole $2\frac{1}{2}$ D set of equations and added to the 2D system matrix each time for a harmonic number n .

In summary, the $2\frac{1}{2}$ D system matrix of the entire mesh for each equation (or harmonic) would be calculated as:

$$S'(n) = S + \left(\frac{n\pi}{z_m}\right)^2 R \quad (5.5)$$

where S is the 2D system matrix and R represents the additional term in the $2\frac{1}{2}$ D model; both of which are constant and only depend on the position of the nodes in the mesh geometry. Hence, the running time of the $2\frac{1}{2}$ D method after algorithm rearrangement would be equal to the running time of one 2D module plus the running time for solving extra forward equations, expressed in (3.41), having the same dimension as the 2D problem for the non-zero harmonic terms, plus the time for calculating the residual matrix (t_R) as well as the inverse Fourier transform (t_{IFT}).

$$t_{2\frac{1}{2}D} = t_{2D} + n_{max} \times t_{2D-size \text{ Forward_Solve}} + t_R + t_{IFT} \quad (5.6)$$

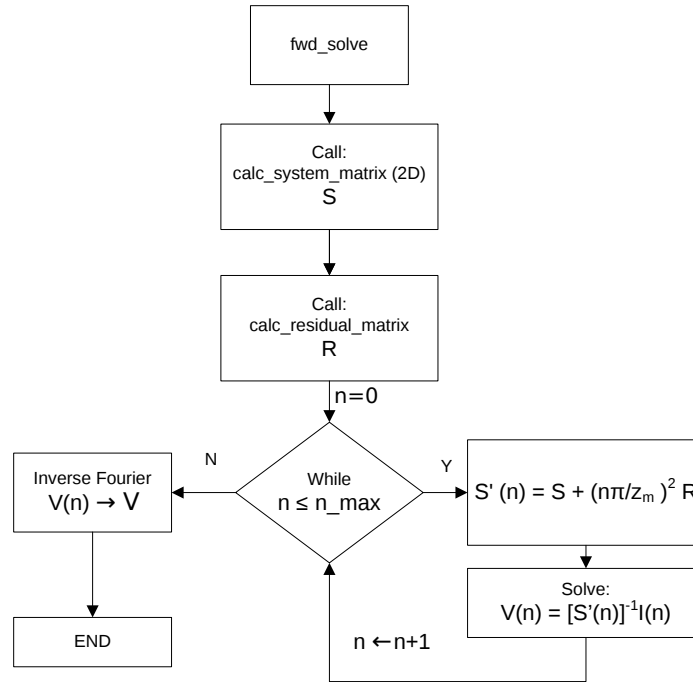


Figure 5.3: Block diagram of the 'fwd_solve' module explained in section 5.2.

The running time of the $2\frac{1}{2}D$ method without the algorithm rearrangement would be:

$$t_{2\frac{1}{2}D} = (n_max + 1) \times (t_{2D\text{-size Forward_Solve}} + t_R) + t_{IFT} \quad (5.7)$$

Finally, the new block diagram of the 'fwd_solve' would be as figure 5.3. The 'calc_residual_matrix' module would be the same as the 'calc_system_matrix' in $2\frac{1}{2}D$; yet, the only difference is the change in variable label from S' to R which is clearly not a change in the algorithm. In this case, the 'calc_local_stiffness_matrix' module would only calculate the residual part of the local system matrix excluding the coefficient which is:

$$R_{ij}^k = \int_{E_k} \phi_i \phi_j d\Omega \quad (5.8)$$

5.3 How Many Terms Are Needed?

The number of equations sufficient to be solved for the $2\frac{1}{2}$ D solution with an acceptable accuracy differs depending on the problem parameters such as the electrode height (h) and the domain height (H). Substituting equation (5.5) into equation (4.4) results in the more complete equation written in (5.9). Here, as n increases, the driven currents (I) in the nominator decrease by a 'sinc' function pattern while the system matrix increases in the denominator (inverse) by the n^2 coefficient.

$$V_{2\frac{1}{2}D} = \frac{1}{H}S^{-1}I + \sum_{n=1}^{\infty} \frac{2}{n\pi h} \sin\left(\frac{n\pi h}{2z_m}\right) \cos\left(\frac{n\pi}{z_m}z\right) \left(S + \left(\frac{n\pi}{z_m}\right)^2 R\right)^{-1}I \quad (5.9)$$

Simulation results show that even 4 terms are sufficient for gradients considering 1% accuracy for a mesh with $H = 2$ (Height is 2 times the radius). Figure 5.4, illustrates the maximum relative error in measurements introduced if the summation of harmonics is truncated at n_{max} . It can be observed from this plot that for greater $\frac{h}{H}$ values, the truncation point is smaller and 3 terms are sufficient. In this analysis, the error (ε) is calculated by equation (5.10) where MAX should tend to positive infinity ($+\infty$); however, for the purpose of stimulation $MAX = 100$ is used.

$$\begin{aligned} \varepsilon(\%) &= \max\left(\frac{\sum_{n=0}^{MAX} V_n \cos\left(\frac{n\pi}{z_m}z\right) - \sum_{n=0}^{n_{max}} V_n \cos\left(\frac{n\pi}{z_m}z\right)}{\sum_{n=0}^{MAX} V_n \cos\left(\frac{n\pi}{z_m}z\right)}\right) \\ &= \max\left(\frac{\sum_{n=N_{max}+1}^{MAX} V_n \cos\left(\frac{n\pi}{z_m}z\right)}{\sum_{n=0}^{MAX} V_n \cos\left(\frac{n\pi}{z_m}z\right)}\right) \end{aligned} \quad (5.10)$$

After the 4th term, mostly the harmonic voltage under the active electrodes is considerable and other harmonic voltages are approximately zero. As it was discussed previously, the voltage on the active electrodes is not counted in measurements. Figure 5.5, illustrates the maximum relative error in electrode voltage introduced if the

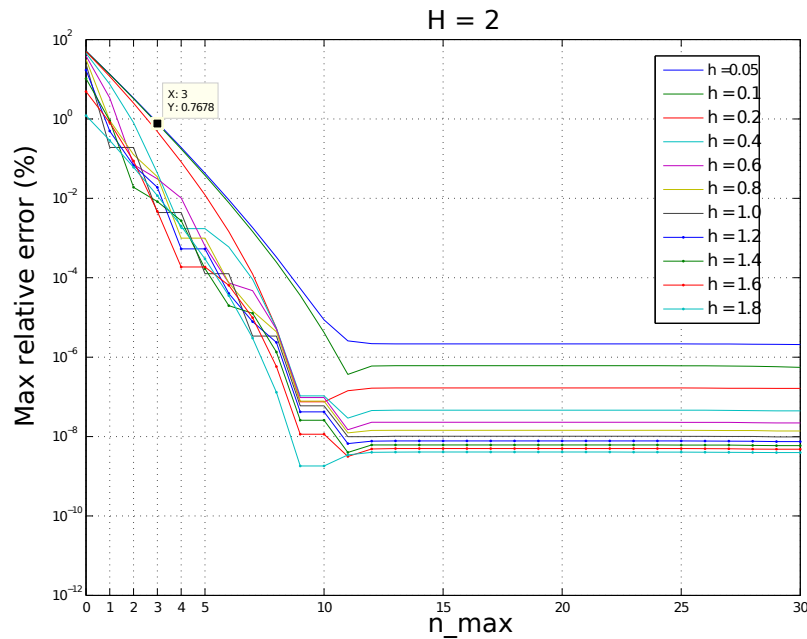


Figure 5.4: Max relative error in measurements introduced if the summation is truncated at n_{max}

summation of harmonics is truncated at n_{max} . Unlike the plot for the measurements, here, the plots which are for the electrode voltage decay slowly as n_{max} increases. Therefore, for those applications that consider the active electrode voltage in the measurements, or require its value, the truncation point of the series is higher. The fluctuations on the plot are due to the fact that the value of the series at $n_{max} = 100$ is used as a reference for calculation of the error while the value at ∞ should be used for this purpose and also it can be due to the rippling nature of the 'sinc' function. The 'sinc' ripples are more obvious in the un-averaged node voltages and become wider (as seen in the figure) when averaged over the electrode length for CEM implementation.

When the height of the electrodes, h , is small compared to the height of the domain height 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

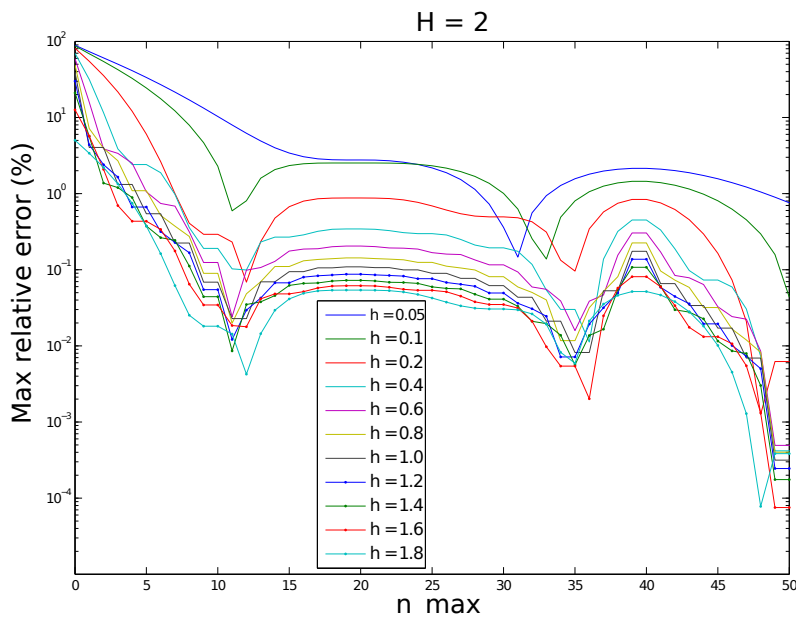


Figure 5.5: Max relative error in voltages introduced if the summation is truncated at n_{max}

the h/H ratio reaches 1, harmonic voltages vanish faster by increment of n ; hence the truncation point of the series, n_{max} , does not increase. However, in this case, the voltage under the electrodes for the 3D model at different heights is not constant; hence, it is required to completely calculate the $2\frac{1}{2}$ D solution for all positions under the electrode and average the voltages of different z -layers which was mentioned earlier. 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. Table 5.1 lists the truncation point for different H/r given 1% accuracy; where r is the radius of the cylindrical tank. The truncation point is greater for larger H/r and lower for smaller H/r .

$\frac{H}{r}$	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)	5	(≥ 0.1)	7
			(≥ 0.2)	3		

Table 5.1: Truncation point (n_{max}) for different Tank Height to radius ratio (H/r) and electrode height to Tank Height ratio (h/H) given 1% accuracy in measurements

5.4 What we gained

5.4.1 Memory Performance

Compared to the 2D modelling, the $2\frac{1}{2}$ D modelling offers better accuracy while it requires solving more equations or matrix inversions. On the other hand, the computational complexity of the $2\frac{1}{2}$ D, is extraordinarily less than the 3D modelling while the accuracy is almost same.

For a same model, the $2\frac{1}{2}$ D solver requires as many elements and nodes as the 2D mesh; while the 3D model requires many more elements and memory. Assuming employing a 3D mesh consisting 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 elements as the 2D system matrix. Also, the 3D sensitivity matrix contains $3(M - 1)$ times as many elements as the 2D sensitivity matrix. However, the matrix to be inverted in the inverse problem for the 3D model contains $9(M - 1)^2$ times element as the 2D case. This comparison can be shown as the following where $[Matrix]_{a \times b}$ represents a matrix with a rows and b columns, K is the number of elements, N is the number of nodes in the mesh, and M is the number of z-layers (slices in z).

Mesh Structure:

$$[2\frac{1}{2}D : Mesh.Nodes]_{N \times 2} \quad \text{vs.} \quad [3D : Mesh.Nodes]_{MN \times 3}$$

$$[2\frac{1}{2}D : Mesh.Elements]_{K \times 3} \quad \text{vs.} \quad [3D : Mesh.Nodes]_{3(M-1)K \times 4}$$

System Matrix:

$$[S_{2\frac{1}{2}D}]_{N \times N} \quad \text{vs.} \quad [S_{3D}]_{MN \times MN}$$

Sensitivity Matrix:

$$[H_{2\frac{1}{2}D}]_{K \times (\text{number of measurements})} \quad \text{vs.} \quad [H_{3D}]_{3(M-1)K \times (\text{number of measurements})}$$

Matrix Inversion at the Inverse Problem:

$$[(H^TWH)_{2\frac{1}{2}D}^{-1}]_{K \times K} \quad \text{vs.} \quad [(H^TWH)_{3D}^{-1}]_{3(M-1)K \times 3(M-1)K}$$

Matrix Inversion at the Forward Problem:

$$[S_{2\frac{1}{2}D}^{-1}]_{N \times N} \quad \text{vs.} \quad [S_{3D}^{-1}]_{MN \times MN}$$

For example, a typical 2D mesh contains 2,113 nodes and 4,094 elements. The 3D mesh which employs 61 slices of the 2D mesh as its base would then contain 61 times nodes, i.e., 128,893, and 180 times as many elements, i.e., 736,920. Then, the system matrix of the 3D mesh is $61 \times 61 = 3,681$ times larger than the 2D system matrix; hence, the 3D forward problem involves inversion of a 3,681 times larger matrix. For this specific example, the number of non-zero elements in the $2\frac{1}{2}$ D system matrix is: 14,529 (0.32%) and in 3D system matrix is: 1,884,789 (0.011%); therefore, the effective ratio of the element density between the sparse system matrices is: 130 for this specific case. The 3D sensitivity matrix would be 180 times larger than the 2D sensitivity matrix and the 3D problem involves inversion of a 32,400 times larger matrix than the 2D.

Therefore, the trade-off for the $2\frac{1}{2}$ D forward problem is that: onetime inversion of a M^2 times larger matrix vs. multiple inversion of a much smaller matrix while

Process description	Time	
	161 layers	61 layers
System Matrix Construction:	110 sec	30 sec
Forward Matrix Inversion:	135 sec	42 sec
Total Forward Solver:	267 sec	76 sec

Table 5.2: Time performance for 3D

maintaining a reasonable accuracy. As discussed in the previous section, the number of inversions could be as few as 5; or at most 10. The trade off for the $2\frac{1}{2}$ D inverse problem is: onetime calculation and inversion of a $9(M - 1)^2$ times larger matrix vs. multiple calculation of a much smaller matrix yet one time inversion of that.

5.4.2 Time Performance

As an example, for a typical 3D mesh with 161 z-layers having around 2 million elements (1,966,080) compared with a 3D mesh with about 0.8 million elements (737,280) depicted in figure 4.1 (see chapter 4), a typical calculation time of system matrix construction and matrix inversion in the forward solver is presented in table 5.2. This running time is calculated using 'tic' and 'toc' command of Matlab in a computer with the following specifications: Intel Xeon(R) CPU X5550 @ 2.67 GHz and 64 GBytes of RAM.

The running time for "System Matrix Construction" was measured as the running time required for the line # 24 of the 'aa_fwd_solve.m' module in EIDORS which executes the 'aa_calc_system_mat.m' module. This line is written with the tic toc command as:

```

t_handle = tic;
24  s_mat= calc_system_mat( fwd_model, img );
    toc(t_handle)

```

Furthermore, the running time for "Forward Matrix Inversion" was measured as the running time of the line # 32 of the the 'aa_fwd_solve.m' module which operates the 'forward_solver' function for computing the matrix inversion. This line is written with the tic toc command as:

```

t_handle = tic;
32 v(idx,:)= forward_solver( s_mat.E(idx,idx), pp.QQ(idx,:), tol);
toc(t_handle)

```

Finally, the running time for "Total Forward Solver" was measured as the running time of the the 'fwd_solve.m' module which operates all required calculations for solving the forward problem on the mesh including system matrix construction and inversion but not limited to them. The following line together with the tic toc command measures this time:

```

t_handle = tic;
inh_data_3d = fwd_solve( sim_img_3d );
toc(t_handle)

```

where the 'sim_img_3d' contains the 3D inverse model structure which includes the 3D 'fwd_model', 3D mesh node positions, elements definitions (node connections), element conductivities, electrode model information, the stimulated current pattern, etc according the EIDORS 'fwd_models'¹ or 'mk_image' module.

On the other hand, for the $2\frac{1}{2}$ D method with a 2D mesh with 4,096 elements depicted in figure 4.1, the calculation time of matrix reconstruction and matrix inversion is presented in table 5.3. The running time varies in a narrow range depending on the number of zero elements in the system or residual matrix. The performance ratio could be approximated as 15 ~ 50 (15 when using 61 z-layers and 50 when using 161 z-layers).

¹http://eidors3d.sourceforge.net/tutorial/EIDORS_basics/tutorial010.shtml

Process description	Time
2D System Matrix Construction (S):	0.16 sec
Residual Matrix Construction (R):	5.0 sec
Inversion time for each harmonic:	0.025 sec
Total (average for 5 terms):	5.3 sec

Table 5.3: Time performance for $2\frac{1}{2}$ D

Normally, it is expected that the result of the 3D algorithm using 161 layers along the z-axis is more accurate than the result of the 3D algorithm for the same mesh employing 61 z-layers. In chapter 4, it was shown that the results of the 3D algorithm employing 161 z-layer is closer to the $2\frac{1}{2}$ D solution which does not depend on the number of z-layers. The $2\frac{1}{2}$ D solution requires a specific amount of time regardless of the number of z-layers. On the other hand, the 3D solution is improved while its running time increases by employing more z-layers. In summary, the more layers used for the 3D solver results in a better accuracy between the 3D and $2\frac{1}{2}$ D while taking more time and leading to a higher running time performance ratio between the $2\frac{1}{2}$ D and 3D. Here, the implemented algorithm for calculating the residual matrix is not computationally efficient which makes the calculation time of the residual matrix to be dominant. A more efficient residual calculation algorithm could potentially increase the $2\frac{1}{2}$ D time performance significantly.

For the $2\frac{1}{2}$ D, the running time of the "2D System Matrix Construction (S)" was measured the same way as the "System Matrix Construction" for the 3D case; this time is the calculation time required for operating the line # 24 of the 'aa_fwd_solve.m' module in EIDORS using a 2D mesh which executes the 'aa_calc_system_mat.m' module. However, in our modified modules 'aa_fwd_solve_modif.m', this line is written with the tic toc command as:

```

t_handle = tic;
26 s_mat_old = calc_system_mat( fwd_model, img );
toc(t_handle)

```

Similarly, the running time for "Residual Matrix Construction (R)" was measured as the time required for calculation of the line # 19 in our modified 'aa_fwd_solve_modif.m' module which executes the 'nb_calc_system_mat.m' module to calculate the higher order $2\frac{1}{2}D$ terms. This line is written with the tic toc command as:

```

18 t_handle = tic;
19 s_mat_new = calc_system_mat( fwd_model, img );
20 toc(t_handle)

```

where, unlike the previous cases, the 'fwd_model.system_mat' parameter is set to 'nb_calc_system_mat'.

Furthermore the running time for "Inversion time for each harmonic" was measured as the time required for calculating the following line of our modified 'aa_fwd_solve_modif.m' module which operates the 'forward_solver' function for computing the matrix inversion. This line is written with the tic toc command as:

```

76 t_handle = tic;
77 v(idx,:)= forward_solver( s_mat.E(idx,idx), coef * pp.QQ(idx,:), tol); // NBA
78 toc(t_handle)

```

Finally, the "Total (average for 5 terms)" running time was measured as the time required for calculating the following line of our developed module 'nb_fwd_solve_new' which is written based on the algorithm explained in section 5.2. The time is an average of the measured running time for different value of h/H . This running time can also be calculated using the formula provided in equation (5.6).

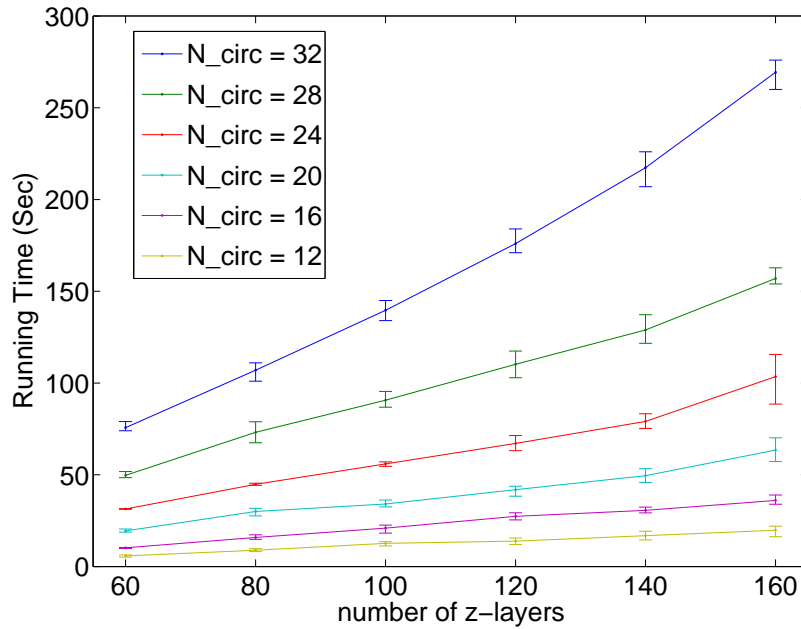


Figure 5.6: Running time for 3D - N_circ is the number of circles (xy-layers)

```

sim_img_25d.fwd_model.system_mat = 'nb_calc_system_mat' ;
sim_img_25d.fwd_model.solve = 'nb_fwd_solve_new' ;

t_handle = tic;

inh_data_25d = fwd_solve( sim_img_25d );

toc(t_handle)

```

where the 'sim_img_25d' contains the 2D forward model structure which includes the 2D 'fwd_model', 2D mesh node positions, elements definitions (node connections), element conductivities, electrode model information, the stimulated current pattern, etc according the EIDORS 'fwd_models'² or 'mk_image' module.

Figure 5.6 shows the running time measured for the 3D forward solver employing different 3D mesh with different number of layers in z and xy (circles). The values are an average for three different electrode height over tank height ratio (h/H). The

²http://eidors3d.sourceforge.net/tutorial/EIDORS_basics/tutorial010.shtml

circles		z-layers					
		61	81	101	121	141	161
12	$N :$	19,093	25,353	31,613	37,873	44,133	50,393
	$K :$	103,680	138,240	172,800	207,360	241,920	276,480
16	$N :$	33,245	44,145	55,045	66,640	76,845	87,745
	$K :$	184,320	245,760	307,200	65,945	430,080	491,520
20	$N :$	51,301	68,121	84,941	101,761	118,581	135,401
	$K :$	288,000	384,000	480,000	576,000	672,000	768,000
24	$N :$	73,261	97,281	121,301	145,321	169,341	193,361
	$K :$	414,720	552,960	691,200	829,440	967,680	1,105,920
28	$N :$	99,125	131,625	164,125	196,625	229,125	261,625
	$K :$	564,480	752,640	940,800	1,128,960	1,317,120	1,505,280
32	$N :$	128,893	171,153	213,413	255,673	297,933	340,193
	$K :$	737,280	983,040	1,228,800	1,474,560	1,720,320	1,966,080

Table 5.4: Number of Elements (K) and Nodes (N) in the 3D mesh

	xy-layers (circles)					
	12	16	20	24	28	32
$N :$	313	545	841	1,201	1,625	2,113
$K :$	576	1,024	1,600	2,304	3,136	4,096

Table 5.5: Number of Elements (K) and Nodes (N) in the $2\frac{1}{2}$ D mesh employing different number of xy-layers (circles)

figure shows how employing more layers significantly increases the 3D running time.

The number of nodes and elements required for each 3D mesh is listed in table 5.4.

Figure 5.7 shows the running time measured for the $2\frac{1}{2}$ D forward solver employing different 2D mesh with different number of layers in xy (circles). The figure shows how employing more xy-layers increases the $2\frac{1}{2}$ D running time. The number of nodes and elements required for each 2D mesh is listed in table 5.5.

Figure 5.8 illustrates the $3D/2\frac{1}{2}D$ time performance ratio calculated for different number of layers in z and xy (circles) from the measured time shown in the previous two figures. The values are an average for three different electrode height over tank height ratio (h/H). The figure demonstrates how employing more layers significantly increases the $3D/2\frac{1}{2}D$ time performance ratio while the accuracy of the 3D solution

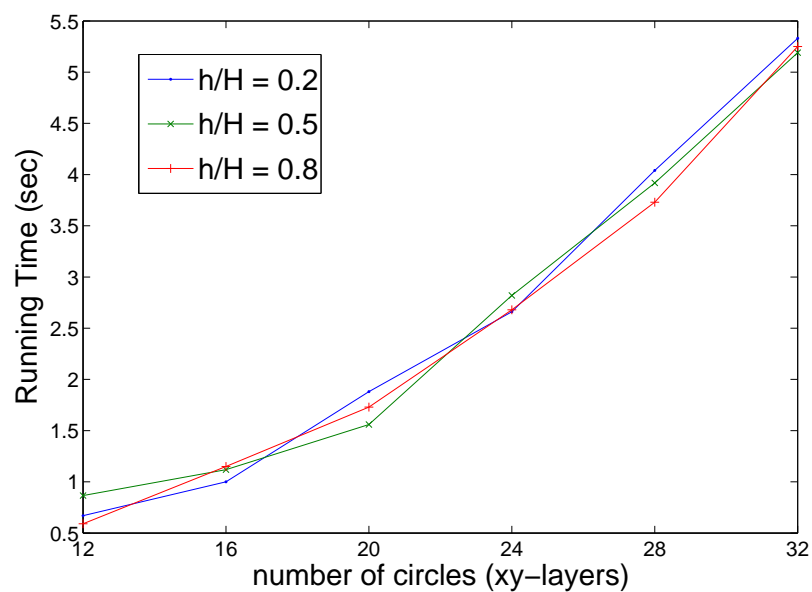


Figure 5.7: Running time for $2\frac{1}{2}$ D - h is electrode height and $H=2$ is tank Height

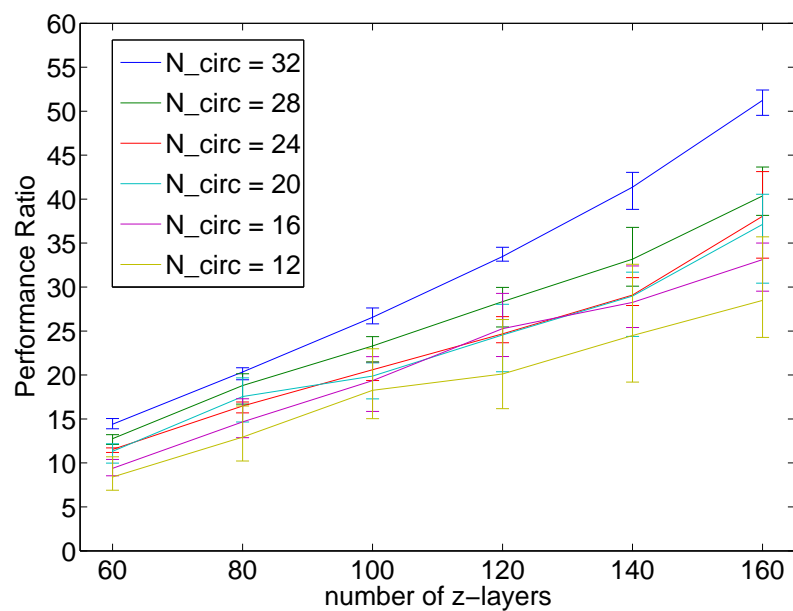


Figure 5.8: $3\text{D}/2\frac{1}{2}\text{D}$ Time Performance ratio - N_{circ} is the number of circles (xy-layers)

improves.

Chapter 6

Conclusions and Future Works

6.1 Summary and Conclusion

In this work, a $2\frac{1}{2}$ D finite element method was applied to the EIT problem considering complete electrode modelling. The correct boundary condition was derived from the original 3D boundary equation and tested. The complete electrode model assigns more than one node for each electrode and introduces a parameter for electrode-skin contact impedance to model the relation between the electrode node and nodes under the electrode.

A key conclusion of this work is: compared to the 3D modelling, the $2\frac{1}{2}$ D offers better memory performance (200-400 times less elements) and much less calculation complexity as well as the calculation time (about 15-50 times less while the algorithm for residual matrix calculation is not efficient, see figure 5.8). It was also concluded that using more layers in 3D such as $\Delta z = \frac{2}{160}$ provides better accuracy for the 3D solution. On the other hand, compared to the 2D modelling, the $2\frac{1}{2}$ D provides more accurate results due to the fact that it accounts for the 3D current source geometry and off-slice currents.

In addition, in section 2.9 a brief literature review on the 2.5D methods was provided where it was described how the term "2.5D" is used differently by different

authors. It was concluded that the $2\frac{1}{2}$ D finite element for EIT problem differs by the problem shape as well as the boundary conditions and modelling. For applications of EIT in medical imaging, it can be concluded that the skin-electrode contact impedance is more important than for applications in process tomography. In chapter 3, a complete process of $2\frac{1}{2}$ D finite element modelling was explained. In the second part of chapter 3, it was demonstrated how the system matrix is constructed in complete electrode modelling. It was concluded that the electrode voltage is a weighted average of voltages of all nodes under the electrode. These weights are calculated based on the length or area of the intersection of the electrode and the interpolation function of each node. For 2D electrodes, the weight for each node is proportional to the area of the region which is formed by that node together with all of its neighboring nodes under the electrode that are directly connected to that node; the weights are normalized by the total area. For 1D electrodes, regardless of the type of the actual mesh (whether 3D or 2D), the weight for each node is proportional to the summation of the distances between that node and its neighboring nodes under the electrode; the weights are normalized by the total length.

In chapter 4, we demonstrated that the $2\frac{1}{2}$ D solver successfully modifies the 2D solution toward the 3D solution. In addition, it was concluded that the 2D and 3D solvers work differently since the 2D solver employs 2D interpolation functions and the 3D solver employs 3D interpolation functions; hence, the solution are not completely matched for a 2D mesh and a 3D mesh with a full $2\frac{1}{2}$ D boundary. The chapter demonstrates that the developed modules work well in modifying the 2D solution toward the 3D solution. Since the $2\frac{1}{2}$ D employs a 2D FE mesh and 2D interpolation functions, a small amount of difference is introduced which decreases with element density.

In chapter 5, the structure of the developed modules was explained. It was concluded that it is possible to improve the speed of the algorithm by keeping the calculations that are independent of n and z_m out of the loop. In other words, it is not necessary to calculate the system matrix and the residual matrix for all PDEs since these matrices only depend on the geometry of the mesh. An analysis on how many terms are required for calculating the $2\frac{1}{2}$ D shows that for a mesh with height equal to 2 ($H = 2$ times radius), 4 terms are sufficient for achieving 1% accuracy in measurements. The truncation point is greater for longer meshes and lower for shorter meshes. Finally, a description of the memory and computation performance of the $2\frac{1}{2}$ D and 3D solver was provided.

6.2 Future Works

6.2.1 Assuming Symmetric Conductivity

In this section we start an investigations on the assumption of the symmetric conductivity. In general, for having a symmetric voltage around the $z = 0$ plane, it is only required to have symmetric conductivities as well as the boundary condition. From this point, we assume that we have 3D conductivity yet symmetric around $z = 0$ plane. Putting this conductivity into the Laplace equation (3.3), would result into the following equation instead of equation (3.4):

$$\sum_{n=0}^{\infty} \left\{ \left[\nabla_{2D} \cdot [\sigma(x, y, z) \nabla_{2D} V_n(x, y)] - \sigma(x, y, z) \left(\frac{n\pi}{z_m} \right)^2 V_n(x, y) \right] \cos\left(\frac{n\pi}{z_m} z \right) - \left[\frac{n\pi}{z_m} \frac{\partial \sigma(x, y, z)}{\partial z} V_n(x, y) \right] \sin\left(\frac{n\pi}{z_m} z \right) \right\} = 0 \quad (6.1)$$

where the variables are the same as chapter 3. This equation is valid for all values of n . Considering the orthogonality of cosines and sinusoids, both sides of equation must

be zero; hence, the coefficient of 'cos' and 'sin' terms must be independently zero. However, before forcing the 'sin' terms to be zero it is required to apply the symmetric property of the conductivity. Assuming symmetric conductivity it is possible to write the 3D conductivity in the form of Fourier cosine terms as

$$\sigma(x, y, z) = \sum_{m=0}^{\infty} \sigma_m(x, y) \cos\left(\frac{m\pi}{z_m} z\right) \quad (6.2)$$

where $\sigma_m(x, y)$ is the m^{th} Fourier coefficient of the conductivity. Substituting equation (6.2) into equation (6.1) would result in the following expression:

$$\begin{aligned} \sum_{n=0}^{\infty} \left\{ \left[\nabla_{2D} \cdot [\sigma(x, y, z) \nabla_{2D} V_n(x, y)] - \sigma(x, y, z) \left(\frac{n\pi}{z_m}\right)^2 V_n(x, y) \right] \cos\left(\frac{n\pi}{z_m} z\right) \right. \\ \left. - \left[\frac{n\pi}{z_m} \left(\sum_{m=0}^{\infty} -\frac{m\pi}{z_m} \sigma_m(x, y) \sin\left(\frac{m\pi}{z_m} z\right) \right) V_n(x, y) \right] \sin\left(\frac{n\pi}{z_m} z\right) \right\} = 0 \end{aligned} \quad (6.3)$$

The outer coefficients of the inner 'sin' term contain another sinusoid term. The multiplication of these two sinusoid results into two cosine terms by the following identity in (6.4a):

$$\sin\left(\frac{m\pi}{z_m} z\right) \sin\left(\frac{n\pi}{z_m} z\right) = -\frac{1}{2} \left[\cos\left(\frac{(n+m)\pi}{z_m} z\right) - \cos\left(\frac{(n-m)\pi}{z_m} z\right) \right] \quad (6.4a)$$

$$\cos\left(\frac{m\pi}{z_m} z\right) \cos\left(\frac{n\pi}{z_m} z\right) = \frac{1}{2} \left[\cos\left(\frac{(n+m)\pi}{z_m} z\right) + \cos\left(\frac{(n-m)\pi}{z_m} z\right) \right] \quad (6.4b)$$

The resulted cosine terms for the multiplication of two 'sin' terms are then absorbed into the 'cos' part of equation (6.3). This could be also written as a sort of convolution between the voltage and conductivity. Furthermore, it is possible to substitute the conductivity in the 'cos' part of equation (6.3) by it's cosine Fourier transform in

equation (6.2). Doing this for the cosine part yields:

$$\begin{aligned} \sum_{n=0}^{\infty} \{ [\nabla_{2D} \cdot [\left(\sum_{m=0}^{\infty} \sigma_m(x, y) \cos\left(\frac{m\pi}{z_m} z\right) \right) \nabla_{2D} V_n(x, y)] - \\ \left(\sum_{m=0}^{\infty} \sigma_m(x, y) \cos\left(\frac{m\pi}{z_m} z\right) \right) \left(\frac{n\pi}{z_m}\right)^2 V_n(x, y)] \cos\left(\frac{n\pi}{z_m} z\right) \end{aligned} \quad (6.5)$$

This expression which has been written for the cosine part of equation (6.1) could be simplified to

$$\begin{aligned} \sum_{n=0}^{\infty} \sum_{m=0}^{\infty} \cos\left(\frac{m\pi}{z_m} z\right) \{ \nabla_{2D} \cdot [\sigma_m(x, y) \nabla_{2D} V_n(x, y)] - \\ \sigma_m(x, y) \left(\frac{n\pi}{z_m}\right)^2 V_n(x, y) \} \cos\left(\frac{n\pi}{z_m} z\right) \end{aligned} \quad (6.6)$$

Here, again the multiplication of the two cosine terms using the identity expressed in (6.4b) results in two separate cosine terms which could be written in the form of convolution. Putting the cosine part and the new form of the sinusoid part (which is written in the form of subtraction of two cosine terms) together results in:

$$\begin{aligned} \sum_{n=0}^{\infty} \sum_{m=0}^{\infty} \{ \nabla_{2D} \cdot [\sigma_m(x, y) \nabla_{2D} V_n(x, y)] - \sigma_m(x, y) n(n+m) \left(\frac{\pi}{z_m}\right)^2 V_n(x, y) \} \cos\left(\frac{(n+m)\pi}{z_m} z\right) \\ + \{ \nabla_{2D} \cdot [\sigma_m(x, y) \nabla_{2D} V_n(x, y)] - \sigma_m(x, y) n(n-m) \left(\frac{\pi}{z_m}\right)^2 V_n(x, y) \} \cos\left(\frac{(n-m)\pi}{z_m} z\right) = 0 \end{aligned} \quad (6.7)$$

Finally, this equation must be valid for all values of n . After rewriting equation (6.7) in the form of a single summation with a single summation index, both sides of the equation must be zero due to the orthogonality of cosines and sinusoids; hence, the coefficient of 'cos' and 'sin' terms must be independently zero. Forcing the cosine terms to zero results in a set of 2½D equation for the case of symmetric conductivity. Forcing any possible remaining sinusoid term to zero yields some possible

additional conditions to the $2\frac{1}{2}$ D set of equations.

6.2.2 Windowing the Injected Current

In chapter 4 section 4.5.2, it was discussed that the current under the electrode in a 3D model with CEM is not constant along with the z direction (see figure 4.13). However, for modelling the $2\frac{1}{2}$ D boundary current, it was assumed that the current is uniformly distributed along the z coordinate. Figure 4.14 shows that voltage under the electrode nodes in a 3D model is approximately uniform while the current density is not uniform and demonstrates a concave pattern; In contrast, the injected current for the $2\frac{1}{2}$ D is modelled uniformly while the results show a convex pattern in voltage. This difference in the modelling of the boundary condition makes a possible source of error between the solutions.

For the purpose of modifying the $2\frac{1}{2}$ D model boundary condition, we suggest to employ a windowing strategy in order to shape the boundary current as what it is actually in a 3D model with CEM. The window should aim to model the concave attribute of the current density for different electrode heights, h . Assuming $w_h(z)$ is a window in spatial domain which aims to compensate the current injection difference, we would have the following expression for the current density in the spatial domain:

$$J(x, y, z) = \frac{1}{h} \text{rect}\left(\frac{z}{h}\right) w_h(z) \quad (6.8)$$

where J is the current density and $\text{rect}\left(\frac{z}{h}\right)$ is a rectangular pulse between $-\frac{h}{2}$ and $+\frac{h}{2}$. Then, the updated current terms for the $2\frac{1}{2}$ D solver in the Fourier domain would be the convolutions of the 'sinc' terms defined previously in (3.14) and the cosine Fourier transform of the window in the spatial frequency domain as:

$$J_n(x, y) = \frac{1}{hz_m} \text{sinc}\left(\frac{nh}{2z_m}\right) * W_h\left(\frac{n\pi}{z_m}\right) = \frac{2}{n\pi h} \sin\left(\frac{n\pi h}{2z_m}\right) * W_h\left(\frac{n\pi}{z_m}\right) \quad (6.9)$$

where '*' represents the convolution operator and $W_h(\frac{n\pi}{z_m})$ is defined by:

$$W_h(\frac{n\pi}{z_m}) = \int_{-z_m}^{+z_m} w_h(z) \cos(\frac{n\pi}{z_m} z) dz \quad (6.10)$$

6.2.3 2.5D image reconstruction employing 2½D finite element modelling

The 3D models often employ multiple rings of electrode around the body (or medium). However, either a full 3D method or a 2.5D algorithm might be used for image reconstruction. The 2.5D image reconstruction refers to a method which uses a 2D image reconstruction technique for each ring of electrodes and then interpolates the results from all of the rings of electrodes to reconstruct a 3D image of the conductivity similar to the 2.5D CT. It is suggested that a 'full' 2½D image reconstruction is developed which employs the 2½D finite element modelling for image reconstruction of each ring of electrode. In this method, the results of the 2½D method employing 2½D finite element model (instead of the 2D image reconstruction) is used in interpolation to reconstruct a 3D image of conductivity. The advantage of this system is that it accounts for the off-slice currents while it is computationally efficient compared to the full 3D image reconstruction.

6.2.4 Error for the 'True' 3D Thorax

The 2½D FE method could be employed in any medium where the 2½D assumptions are valid. In a human body this could be hands or legs, or lower portions of stomach. For the upper part of a human chest, the 2½D assumption (translationally invariant environment) could hardly be satisfied. For this case recently 'true' 3D thorax models have been developed. Although the 2½D method is not applicable to the 'true' 3D thorax models, it is suggested to examine the amount of error introduced by using

$2\frac{1}{2}$ D method on true 3D thorax. The solution of this experiment should be compared to the following situation:

- the solution of a 2D solver for a 3D thorax
- the solution of a 3D solver for a 3D thorax
- the solution of a 3D solver for a 2.5 D thorax

It is also beneficial to evaluate the error performance of the $2\frac{1}{2}$ D method for non-invariant (variant) conductivity cases along the z coordinate.

6.2.5 Generalization - Full Impedance

It is suggested to update the $2\frac{1}{2}$ D derivations and modify the developed method in order to consider the susceptance component of the complex conductivity (or reactive component of the impedance). In addition to this extension, it is suggested that the method accounts for anisotropic conductivity cases for image reconstruction.

Appendix A

Analytical Solution of the Integrals

Assuming that $P_1(x_1, y_1)$, $P_2(x_2, y_2)$ and $P_3(x_3, y_3)$ are the vertices of the triangular simplex where $x_1 < x_2 < x_3$ (see figure A.1), each integral is calculated as follows:

The easiest integral to be calculated is the integral over each element which results in the surface area of the triangular simplex in the 2D case. Generally, the 2-dimensional integral is divided into a summation of two integrals in which one integral is over the x from x_1 to x_2 while y is varying between two lines connecting P_1 to P_2 and P_3 and the other integral is over x from x_2 to x_3 while y is varying between two lines connecting P_3 to P_2 and P_1 . This idea is expressed in equation (A.1)

$$S = \int_{E_k} d\Omega = \int_{x_1}^{x_2} \int_{\min(A,B)}^{\max(A,B)} dydx + \int_{x_2}^{x_3} \int_{\min(C,D)}^{\max(C,D)} dydx \quad (\text{A.1})$$

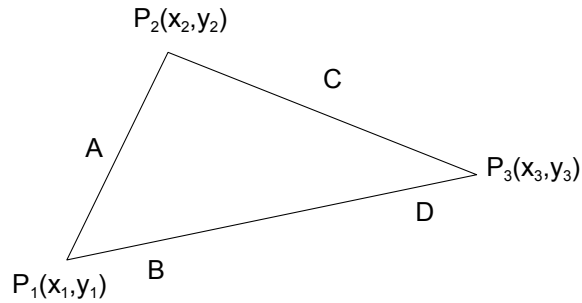


Figure A.1: A sample triangular element

where A is the equation of the line connecting P_1 to P_2 , B is the equation of the line connecting P_1 to P_3 , C is the equation of the line connecting P_3 to P_2 , and D is the equation of the line connecting P_3 to P_1 with respect to variable x which are defined in set of equations (A.2).

$$A = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1) + y_1 \quad (\text{A.2a})$$

$$B = \frac{y_3 - y_1}{x_3 - x_1}(x - x_1) + y_1 \quad (\text{A.2b})$$

$$C = \frac{y_3 - y_2}{x_3 - x_2}(x - x_3) + y_3 \quad (\text{A.2c})$$

$$D = \frac{y_3 - y_1}{x_3 - x_1}(x - x_3) + y_3 \quad (\text{A.2d})$$

The inner integral in the first term of equation (A.1) is calculated as:

$$\begin{aligned} \int_{\min(A,B)}^{\max(A,B)} dy &= [y]_{\min(A,B)}^{\max(A,B)} = [\max(A, B) - \min(A, B)] = |A - B| \\ &= \left| \frac{y_3 - y_1}{x_3 - x_1} - \frac{y_2 - y_1}{x_2 - x_1} \right| (x - x_1) = T_1(x - x_1) \end{aligned} \quad (\text{A.3a})$$

Similarly, the inner integral in the second term of equation (A.1) is calculated as:

$$\begin{aligned} \int_{\min(C,D)}^{\max(C,D)} dy &= [y]_{\min(C,D)}^{\max(C,D)} = [\max(C, D) - \min(C, D)] = |C - D| \\ &= \left| \frac{y_3 - y_1}{x_3 - x_1} - \frac{y_3 - y_2}{x_3 - x_2} \right| [-(x - x_3)] = -T_2(x - x_3) \end{aligned} \quad (\text{A.3b})$$

where, for simplicity in future expressions, T_1 and T_2 are defined as:

$$T_1 = \left| \frac{(y_3 - y_1)}{(x_3 - x_1)} - \frac{(y_2 - y_1)}{(x_2 - x_1)} \right| \quad (\text{A.4a})$$

$$T_2 = \left| \frac{(y_3 - y_1)}{(x_3 - x_1)} - \frac{(y_3 - y_2)}{(x_3 - x_2)} \right| \quad (\text{A.4b})$$

where $|\cdot|$ is the absolute value operator. Finally the surface area, S , is derived as

$$S = \int_{E_k} d\Omega = T_1 \int_{x_1}^{x_2} (x - x_1) dx + T_2 \int_{x_2}^{x_3} (x_3 - x) dx \quad (\text{A.5})$$

The resulted one-dimensional integrals are easy to be solved over x . However, in order to have the full procedure and footsteps for more complex integrals later, the integrals solutions are expressed here:

$$\int_{x_1}^{x_2} (x - x_1) dx = \int_0^{x_2-x_1} u du = \frac{1}{2}(x_2 - x_1)^2 \quad (\text{A.6a})$$

$$\int_{x_2}^{x_3} (x_3 - x) dx = \int_0^{x_3-x_2} u du = \frac{1}{2}(x_3 - x_2)^2 \quad (\text{A.6b})$$

By substituting the integral solutions in (A.6) into equation (A.5), the surface area of each element in a 2-dimensional mesh is calculated as

$$S = T_1 \left[\frac{1}{2}(x_2 - x_1)^2 \right] + T_2 \left[\frac{1}{2}(x_3 - x_2)^2 \right] \quad (\text{A.7})$$

The value for S obtained by equation (A.7) is exactly equal to the value obtained by the formula of a triangle surface, in general, regardless of the ordering of the vertices given by equation (A.8) which validates our procedure.

$$S = \frac{1}{2}(x_2y_3 - x_3y_2 - x_1y_3 + x_3y_1 + x_1y_2 - x_2y_1) \quad (\text{A.8})$$

Integral of x

Following the footsteps of the previous derivation, the integral of x over each simplex could be calculated as

$$\int_{E_k} x d\Omega = \int_{x_1}^{x_2} x \int_{\min(A,B)}^{\max(A,B)} dy dx + \int_{x_2}^{x_3} x \int_{\min(C,D)}^{\max(C,D)} dy dx \quad (\text{A.9})$$

where the inner integrals are previously calculated in (A.3) followed by the definitions of T_1 and T_2 in (A.4). Substituting the results from the set of equations (A.3) into equation (A.9), the integrals of equation (A.9) would then be simplified to

$$\int_{E_k} x d\Omega = T_1 \int_{x_1}^{x_2} x(x - x_1) dx + T_2 \int_{x_2}^{x_3} x(x_3 - x) dx \quad (\text{A.10})$$

Similar to the approach employed previously, the one-dimensional integrals are calculated as:

$$\int_{x_1}^{x_2} x(x - x_1) dx = \int_0^{x_2 - x_1} (u + x_1)u du = \frac{1}{3}(x_2 - x_1)^3 + \frac{1}{2}(x_2 - x_1)^2 x_1 \quad (\text{A.11a})$$

$$\int_{x_2}^{x_3} x(x_3 - x) dx = \int_0^{x_3 - x_2} (x_3 - u)u du = -\frac{1}{3}(x_3 - x_2)^3 + \frac{1}{2}(x_3 - x_2)^2 x_3 \quad (\text{A.11b})$$

Finally, by substituting the results of integrals in (A.11), the integral of x over each simplex is calculated by

$$\int_{E_k} x d\Omega = T_1 \left[\frac{1}{3}(x_2 - x_1)^3 + \frac{1}{2}(x_2 - x_1)^2 x_1 \right] + T_2 \left[-\frac{1}{3}(x_3 - x_2)^3 + \frac{1}{2}(x_3 - x_2)^2 x_3 \right] \quad (\text{A.12})$$

Integral of x^2

The same way the integral of x is calculated, we can calculate the integral of x^2 over each simplex. The procedure is almost same to the previous one; however, in order to have full details the derivation procedure is expressed here.

The integral of x^2 over each simplex is calculated by equation (A.13).

$$\int_{E_k} x^2 d\Omega = \int_{x_1}^{x_2} x^2 \int_{\min(A,B)}^{\max(A,B)} dy dx + \int_{x_2}^{x_3} x^2 \int_{\min(C,D)}^{\max(C,D)} dy dx \quad (\text{A.13})$$

Next, by substituting the solutions of the inner integrals derived in (A.3) equation

(A.13) is simplified into the following equation:

$$\int_{E_k} x^2 d\Omega = T_1 \int_{x_1}^{x_2} x^2(x-x_1) dx + T_2 \int_{x_2}^{x_3} x^2(x_3-x) dx \quad (\text{A.14})$$

Employing the same technique for solving one-dimensional integrals over x , these integrals are calculated as

$$\int_{x_1}^{x_2} x^2(x-x_1) dx = \int_0^{x_2-x_1} (u+x_1)^2 u du = \frac{1}{4}(x_2-x_1)^4 + \frac{2}{3}(x_2-x_1)^3 x_1 + \frac{1}{2}(x_2-x_1)^2 x_1^2 \quad (\text{A.15a})$$

$$\int_{x_2}^{x_3} x^2(x_3-x) dx = \int_0^{x_3-x_2} (x_3-u)^2 u du = \frac{1}{4}(x_3-x_2)^4 - \frac{2}{3}(x_3-x_2)^3 x_3 + \frac{1}{2}(x_3-x_2)^2 x_3^2 \quad (\text{A.15b})$$

Finally, by substituting the solutions of the integrals derived by equation (A.15b) into equation (A.14), the integral of x^2 over each simplex is calculated by the following equation

$$\begin{aligned} \int_{E_k} x^2 d\Omega = & T_1 \left[\frac{1}{4}(x_2-x_1)^4 + \frac{2}{3}(x_2-x_1)^3 x_1 + \frac{1}{2}(x_2-x_1)^2 x_1^2 \right] \\ & + T_2 \left[\frac{1}{4}(x_3-x_2)^4 - \frac{2}{3}(x_3-x_2)^3 x_3 + \frac{1}{2}(x_3-x_2)^2 x_3^2 \right] \end{aligned} \quad (\text{A.16})$$

Integral of y

Unlike the calculation of the previous three integrals, the integral of y over each simplex is more difficult when the vertices are sorted with respect to x . The integral of y is calculated as

$$\int_{E_k} y d\Omega = \int_{x_1}^{x_2} \int_{\min(A,B)}^{\max(A,B)} y dy dx + \int_{x_2}^{x_3} \int_{\min(C,D)}^{\max(C,D)} y dy dx \quad (\text{A.17})$$

Here, the inner integrals are not just a simple integral over y ; hence it is not similar to the previous ones. In the first term, the inner integral is the integral of y between the line connecting P_1 and P_2 defined by A in equation (A.2a) and the line

connecting P_1 and P_3 defined by B in equation (A.2b). The first term of equation (A.17) is calculated as

$$\begin{aligned}
\int_{\min(A,B)}^{\max(A,B)} y dy &= \left[\frac{1}{2} y^2 \right]_{\min(A,B)}^{\max(A,B)} \\
&= \frac{1}{2} [\min(A, B) + \max(A, B)] \cdot [\max(A, B) - \min(A, B)] \\
&= \frac{1}{2} (A + B) \cdot |A - B| = \frac{1}{2} \left[\left(\frac{y_2 - y_1}{x_2 - x_1} + \frac{y_3 - y_1}{x_3 - x_1} \right) (x - x_1) + 2y_1 \right] \\
&\quad \cdot \left| \frac{y_3 - y_1}{x_3 - x_1} - \frac{y_2 - y_1}{x_2 - x_1} \right| (x - x_1) \\
&= T_1 \left[\frac{1}{2} T_3 (x - x_1)^2 + (x - x_1) y_1 \right]
\end{aligned} \tag{A.18a}$$

Also, in the second term of equation (A.17), the inner integral is the integral of y between the line connecting P_3 and P_2 defined by C in equation (A.2c) and the line connecting P_3 and P_1 defined by D in equation (A.2d). The second term of equation (A.17) is calculated as

$$\begin{aligned}
\int_{\min(C,D)}^{\max(C,D)} y dy &= \left[\frac{1}{2} y^2 \right]_{\min(C,D)}^{\max(C,D)} \\
&= \frac{1}{2} [\min(C, D) + \max(C, D)] \cdot [\max(C, D) - \min(C, D)] \\
&= \frac{1}{2} (C + D) \cdot |C - D| = \frac{1}{2} \left[\left(\frac{y_3 - y_2}{x_3 - x_2} + \frac{y_3 - y_2}{x_3 - x_2} \right) (x - x_3) + 2y_3 \right] \\
&\quad \cdot \left| \frac{y_3 - y_1}{x_3 - x_1} - \frac{y_3 - y_2}{x_3 - x_2} \right| [-(x - x_3)] \\
&= -T_2 \left[\frac{1}{2} T_4 (x - x_3)^2 + (x - x_3) y_3 \right]
\end{aligned} \tag{A.18b}$$

where T_1 and T_2 are the coefficients defined previously in (A.4). The new coefficients

T_3 and T_4 are also defined here in equations (A.19) for simplicity in demonstration.

$$T_3 = \left(\frac{y_3 - y_1}{x_3 - x_1} + \frac{y_2 - y_1}{x_2 - x_1} \right) \quad (\text{A.19a})$$

$$T_4 = \left(\frac{y_3 - y_1}{x_3 - x_1} + \frac{y_3 - y_2}{x_3 - x_2} \right) \quad (\text{A.19b})$$

Next, by substituting the solution derived in (A.18a) into the first term of equation (A.17), the integral in the first term of expression (A.17) is solved by the following procedure:

$$\begin{aligned} \int_{x_1}^{x_2} \int_{\min(A,B)}^{\max(A,B)} y \, dy \, dx &= \int_{x_1}^{x_2} T_1 \left[\frac{1}{2} T_3 (x - x_1)^2 + (x - x_1) y_1 \right] dx \\ &= T_1 \left[\frac{1}{2} T_3 \int_{x_1}^{x_2} (x - x_1)^2 dx + y_1 \int_{x_1}^{x_2} (x - x_1) dx \right] \quad (\text{A.20}) \\ &= T_1 \left[\frac{1}{2} T_3 \frac{1}{3} (x_2 - x_1)^3 + \frac{1}{2} (x_2 - x_1)^2 y_1 \right] \end{aligned}$$

For deriving the end results in equation (A.20), the first integral over x is calculated by equation (A.21a) while the second integral over x was solved previously in equation (A.6a).

$$\int_{x_1}^{x_2} (x - x_1)^2 dx = \int_0^{x_2 - x_1} u^2 du = \frac{1}{3} (x_2 - x_1)^3 \quad (\text{A.21a})$$

$$\int_{x_2}^{x_3} (x - x_3)^2 dx = \int_{x_2 - x_3}^0 u^2 du = \frac{1}{3} (x_3 - x_2)^3 \quad (\text{A.21b})$$

Similarly, by substituting the solution derived in (A.18b) into the second term of equation (A.17), the integral in the second term of the solution (A.17) is calculated

by the following procedure

$$\begin{aligned}
\int_{x_2}^{x_3} \int_{\min(C,D)}^{\max(C,D)} y \, dy \, dx &= \int_{x_2}^{x_3} -T_2 \left[\frac{1}{2} T_4 (x - x_3)^2 + (x - x_3) y_3 \right] dx \\
&= -T_2 \left[\frac{1}{2} T_4 \int_{x_2}^{x_3} (x - x_3)^2 dx + y_3 \int_{x_2}^{x_3} (x - x_3) dx \right] \quad (\text{A.22}) \\
&= T_2 \left[-\frac{1}{2} T_4 \frac{1}{3} (x_3 - x_2)^3 + \frac{1}{2} (x_3 - x_2)^2 y_3 \right]
\end{aligned}$$

For deriving the end results in equation (A.22), the first integral over x is calculated by equation (A.21b) while the second integral over x was solved previously in equation (A.6b). Finally, the entire integral of equation (A.17) which is the integral of y over each simplex is calculated by the following expression:

$$\begin{aligned}
\int_{E_k} y \, d\Omega &= T_1 \left[\frac{1}{2} T_3 \frac{1}{3} (x_2 - x_1)^3 + \frac{1}{2} (x_2 - x_1)^2 y_1 \right] \\
&\quad + T_2 \left[\frac{1}{2} T_4 \left(-\frac{1}{3}\right) (x_3 - x_2)^3 + \frac{1}{2} (x_3 - x_2)^2 y_3 \right] \quad (\text{A.23})
\end{aligned}$$

Integral of xy

The previous approach is a clue for the next integral which is the integral of xy over each simplex. Here, the inner integrals have a same form as we had in equation (A.17).

$$\int_{E_k} xy \, d\Omega = \int_{x_1}^{x_2} x \int_{\min(A,B)}^{\max(A,B)} y \, dy \, dx + \int_{x_2}^{x_3} x \int_{\min(C,D)}^{\max(C,D)} y \, dy \, dx \quad (\text{A.24})$$

Using the results previously derived for the inner integrals expressed in equations (A.18), the first integral term in equation (A.24) is calculated as

$$\begin{aligned}
\int_{x_1}^{x_2} x \int_{\min(A,B)}^{\max(A,B)} y \, dy \, dx &= \int_{x_1}^{x_2} T_1 \left[\frac{1}{2} T_3 x (x - x_1)^2 + x (x - x_1) y_1 \right] dx \\
&= T_1 \left[\frac{1}{2} T_3 \int_{x_1}^{x_2} x (x - x_1)^2 dx + y_1 \int_{x_1}^{x_2} x (x - x_1) dx \right] \quad (\text{A.25a})
\end{aligned}$$

where T_1 and T_3 are defined in (A.4a) and (A.19a) respectively. In a similar way, using the results previously derived in equation (A.18b), the second integral term in equation (A.24) is calculated as

$$\begin{aligned} \int_{x_2}^{x_3} x \int_{\min(C,D)}^{\max(C,D)} y dy dx &= \int_{x_2}^{x_3} -T_2 \left[\frac{1}{2} T_4 x (x - x_3)^2 + x (x - x_3) y_3 \right] dx \\ &= T_2 \left[\frac{1}{2} T_4 \int_{x_2}^{x_3} -x (x - x_3)^2 dx + y_3 \int_{x_2}^{x_3} x (x_3 - x) dx \right] \end{aligned} \quad (\text{A.25b})$$

where, T_2 and T_4 are defined in (A.4b) and (A.19b) respectively. Following the same technique employed for solving the integrals, the first integral term over x in equation (A.25a) is calculated by equation (A.26a) while the second term was solved previously in equation (A.11a). Also, the first integral term over x in equation (A.25b) is calculated by equation (A.26b) while the second term was solved previously in equation (A.11a).

$$\int_{x_1}^{x_2} x (x - x_1)^2 dx = \int_0^{x_2 - x_1} (u + x_1) u^2 du = \frac{1}{4} (x_2 - x_1)^4 + \frac{1}{3} (x_2 - x_1)^3 x_1 \quad (\text{A.26a})$$

$$\int_{x_2}^{x_3} -x (x - x_3)^2 dx = \int_0^{x_3 - x_2} (u - x_3) u^2 du = \frac{1}{4} (x_3 - x_2)^4 - \frac{1}{3} (x_3 - x_2)^3 x_3 \quad (\text{A.26b})$$

By substituting the results of the integral from equations (A.26) into equations (A.25), it is now possible to solve the entire integral of equation (A.24). First, the solution of equations (A.25a) and (A.25b) is expressed separately in (A.27a) and (A.27b) respectively:

$$\begin{aligned} \int_{x_1}^{x_2} x \int_{\min(A,B)}^{\max(A,B)} y dy dx &= T_1 \left\{ \frac{1}{2} T_3 \left[\frac{1}{4} (x_2 - x_1)^4 + \frac{1}{3} (x_2 - x_1)^3 x_1 \right] \right. \\ &\quad \left. + \left[\frac{1}{3} (x_2 - x_1)^3 + \frac{1}{2} (x_2 - x_1)^2 x_1 \right] y_1 \right\} \end{aligned} \quad (\text{A.27a})$$

$$\int_{x_2}^{x_3} x \int_{\min(C,D)}^{\max(C,D)} y dy dx = T_2 \left\{ \frac{1}{2} T_4 \left[\frac{1}{4} (x_3 - x_2)^4 - \frac{1}{3} (x_3 - x_2)^3 x_1 \right] \right. \\ \left. + \left[-\frac{1}{3} (x_3 - x_2)^3 + \frac{1}{2} (x_3 - x_2)^2 x_3 \right] y_3 \right\} \quad (\text{A.27b})$$

Finally combining the two equations in (A.27) lead into the solution of the entire integrals in equation (A.24) which is the integral of xy over each simplex and expressed as

$$\int_{E_k} xy d\Omega = T_1 \left\{ \frac{1}{2} T_3 \left[\frac{1}{4} (x_2 - x_1)^4 + \frac{1}{3} (x_2 - x_1)^3 x_1 \right] + \left[\frac{1}{3} (x_2 - x_1)^3 + \frac{1}{2} (x_2 - x_1)^2 x_1 \right] y_1 \right\} \\ + T_2 \left\{ \frac{1}{2} T_4 \left[\frac{1}{4} (x_3 - x_2)^4 - \frac{1}{3} (x_3 - x_2)^3 x_3 \right] + \left[-\frac{1}{3} (x_3 - x_2)^3 + \frac{1}{2} (x_3 - x_2)^2 x_3 \right] y_3 \right\} \quad (\text{A.28})$$

Integral of y^2

The last integral remained to be solved is the integral of y^2 over each triangular simplex; which is hard to be analytically calculated if the vertices are sorted with respect to the x coordinate. However, if we reorder the vertices such that the new set of vertices $P'_1(x'_1, y'_1)$, $P'_2(x'_2, y'_2)$, and $P'_3(x'_3, y'_3)$ is ordered with respect to y , i.e. $y'_1 < y'_2 < y'_3$, the last integral would be calculated in the same way as the integral of x^2 was calculated previously. The procedure starts by

$$\int_{E_k} y^2 d\Omega = \int_{y'_1}^{y'_2} y^2 \int_{\min(A',B')}^{\max(A',B')} dx dy + \int_{y'_2}^{y'_3} y^2 \int_{\min(C',D')}^{\max(C',D')} dx dy \quad (\text{A.29})$$

where A' is the equation of the line connecting P'_1 to P'_2 , B' is the equation of the line connecting P'_1 to P'_3 , C' is the equation of the line connecting P'_3 to P'_2 , and D' is the equation of the line connecting P'_3 to P'_1 this time with respect to variable y , which are defined as:

$$A' = \frac{x'_2 - x'_1}{y'_2 - y'_1} (y - y'_1) + x'_1 \quad (\text{A.30a})$$

$$B' = \frac{x'_3 - x'_1}{y'_3 - y'_1}(y - y'_1) + x'_1 \quad (\text{A.30b})$$

$$C' = \frac{x'_3 - x'_2}{y'_3 - y'_2}(y - y'_3) + y'_3 \quad (\text{A.30c})$$

$$D' = \frac{x'_3 - x'_1}{y'_3 - y'_1}(y - y'_3) + y'_3 \quad (\text{A.30d})$$

The inner integral in the first term of equation (A.29) is calculated as:

$$\begin{aligned} \int_{\min(A',B')}^{\max(A',B')} dx &= [x]_{\min(A',B')}^{\max(A',B')} = [\max(A', B') - \min(A', B')] = |A' - B'| \\ &= \left| \frac{x'_3 - x'_1}{y'_3 - y'_1} - \frac{x'_2 - x'_1}{y'_2 - y'_1} \right| (y - y'_1) = T'_1(y - y'_1) \end{aligned} \quad (\text{A.31a})$$

The inner integral in the second term of equation (A.29) is calculated as:

$$\begin{aligned} \int_{\min(C',D')}^{\max(C',D')} dx &= [x]_{\min(C',D')}^{\max(C',D')} = [\max(C', D') - \min(C', D')] = |C' - D'| \\ &= \left| \frac{x'_3 - x'_1}{y'_3 - y'_1} - \frac{x'_3 - x'_2}{y'_3 - y'_2} \right| [-(y - y'_3)] = -T'_2(y - y'_3) \end{aligned} \quad (\text{A.31b})$$

Next, by substituting the solutions of the inner integrals derived in (A.31), equation (A.29) is simplified into the following expression:

$$\begin{aligned} \int_{E_k} y^2 d\Omega &= \left| \frac{x'_3 - x'_1}{y'_3 - y'_1} - \frac{x'_2 - x'_1}{y'_2 - y'_1} \right| \int_{y'_1}^{y'_2} y^2 (y - y'_1) dy \\ &\quad + \left| \frac{x'_3 - x'_1}{y'_3 - y'_1} - \frac{x'_3 - x'_2}{y'_3 - y'_2} \right| \int_{y'_2}^{y'_3} y^2 (y'_3 - y) dx \end{aligned} \quad (\text{A.32})$$

By employing the same technique for solving the one-dimensional integrals over x , it is possible to calculate these integrals. However, by a quick view in equations (A.15) it is realizable that the integrals were solved previously and we only need to substitute the variables in equations (A.15). Next, substituting the results of integral into equation (A.32) lead into the solution of integral of y^2 over each triangular

simplex expressed by

$$\begin{aligned} \int_{E_k} y^2 d\Omega = & \left| \frac{x'_3 - x'_1}{y'_3 - y'_1} - \frac{x'_2 - x'_1}{y'_2 - y'_1} \right| \left[\frac{1}{4}(y'_2 - y'_1)^4 + \frac{2}{3}(y'_2 - y'_1)^3 y'_1 + \frac{1}{2}(y'_2 - y'_1)^2 y'^2_1 \right] \\ & + \left| \frac{x'_3 - x'_1}{y'_3 - y'_1} - \frac{x'_3 - x'_2}{y'_3 - y'_2} \right| \left[\frac{1}{4}(y'_3 - y'_2)^4 - \frac{2}{3}(y'_3 - y'_2)^3 y'_3 + \frac{1}{2}(y'_3 - y'_2)^2 y'^2_3 \right] \end{aligned} \quad (\text{A.33})$$

Handling Division by Zero

In the all of the above equations there was an assumption embedded. The denominators of T_1 , T_2 , T_3 and T_4 must not be zero. The same assumption must be held for equation (A.33). In other words, the triangular simplex must not have any edge in parallel with neither the x axis nor the y axis which may occur frequently. This may lead to an infinite slope for the line connecting the nodes. When vertices are sorted with respect to x , only one of these two situations might happen: 1) $x_1 = x_2$ or 2) $x_2 = x_3$. In this case, the first vertex x_1 and the last vertex x_3 could never be equal. The same argument is valid for y . When the vertices are sorted with respect to y , the first vertex y_1 and the last vertex y_3 could never be equal and only one of these two situations might happen: 1) $y_1 = y_2$ or 2) $y_2 = y_3$.

By taking a short look at the definitions of T's and the solutions in equations (A.7), (A.12), (A.16), (A.23), (A.28) and (A.33), it is realizable that the zero terms, i.e. $x_2 - x_1$ and $x_3 - x_2$ can be canceled out by the same terms in the coefficients of T's. Therefore, for the general case of $x_1 \leq x_2 \leq x_3$, while vertices are sorted with respect to the x coordinate, all solutions are re-formulated as the following:

First, the T's are re-defined as

$$T_1 = \left| \frac{(y_3 - y_1)(x_2 - x_1)}{(x_3 - x_1)} - (y_2 - y_1) \right| \quad (\text{A.34a})$$

$$T_2 = \left| \frac{(y_3 - y_1)(x_3 - x_2)}{(x_3 - x_1)} - (y_3 - y_2) \right| \quad (\text{A.34b})$$

$$T_3 = \frac{(y_3 - y_1)(x_2 - x_1)}{(x_3 - x_1)} + (y_2 - y_1) \quad (\text{A.34c})$$

$$T_4 = \frac{(y_3 - y_1)(x_3 - x_2)}{(x_3 - x_1)} + (y_3 - y_2) \quad (\text{A.34d})$$

Next, equation (A.7) is re-formulated as

$$S = \int_{E_k} d\Omega = \frac{1}{2}T_1(x_2 - x_1) + \frac{1}{2}T_2(x_3 - x_2) \quad (\text{A.35})$$

Also, equation (A.12) is re-formulated as

$$\int_{E_k} x d\Omega = T_1 \left[\frac{1}{3}(x_2 - x_1)^2 + \frac{1}{2}(x_2 - x_1)x_1 \right] + T_2 \left[-\frac{1}{3}(x_3 - x_2)^2 + \frac{1}{2}(x_3 - x_2)x_3 \right] \quad (\text{A.36})$$

Next, equation (A.16) is re-formulated as

$$\begin{aligned} \int_{E_k} x^2 d\Omega &= T_1 \left[\frac{1}{4}(x_2 - x_1)^3 + \frac{2}{3}(x_2 - x_1)^2 x_1 + \frac{1}{2}(x_2 - x_1)x_1^2 \right] \\ &+ T_2 \left[\frac{1}{4}(x_3 - x_2)^3 - \frac{2}{3}(x_3 - x_2)^2 x_3 + \frac{1}{2}(x_3 - x_2)x_3^2 \right] \end{aligned} \quad (\text{A.37})$$

Also, equation (A.23) is re-formulated as

$$\int_{E_k} y d\Omega = T_1 \left[\frac{1}{2}y_1(x_2 - x_1) + \frac{1}{6}T_3(x_2 - x_1) \right] + T_2 \left[\frac{1}{2}y_3(x_3 - x_2) - \frac{1}{6}T_4(x_3 - x_2) \right] \quad (\text{A.38})$$

Equation (A.38) could be further simplified to

$$\int_{E_k} y d\Omega = T_1 \left[\left(\frac{1}{2}y_1 + \frac{1}{6}T_3 \right) (x_2 - x_1) \right] + T_2 \left[\left(\frac{1}{2}y_3 - \frac{1}{6}T_4 \right) (x_3 - x_2) \right] \quad (\text{A.39})$$

Next, equation (A.28) is re-formulated as

$$\begin{aligned} \int_{E_k} xy \, d\Omega &= T_1(y_1[\frac{1}{3}(x_2 - x_1)^2 + \frac{1}{2}(x_2 - x_1)x_1] + \frac{1}{2}T_3[\frac{1}{4}(x_2 - x_1)^2 + \frac{1}{3}(x_2 - x_1)x_1]) \\ &\quad + T_2(y_3[-\frac{1}{3}(x_3 - x_2)^2 + \frac{1}{2}(x_3 - x_2)x_3] + \frac{1}{2}T_4[\frac{1}{4}(x_3 - x_2)^2 - \frac{1}{3}(x_3 - x_2)x_3]) \end{aligned} \quad (\text{A.40})$$

Equation (A.40) could also be further simplified to

$$\begin{aligned} \int_{E_k} xy \, d\Omega &= T_1[(\frac{1}{3}y_1 + \frac{1}{8}T_3)(x_2 - x_1)^2 + (\frac{1}{2}y_1 + \frac{1}{6}T_3)(x_2 - x_1)x_1] \\ &\quad + T_2[(-\frac{1}{3}y_3 + \frac{1}{8}T_4)(x_3 - x_2)^2 + (\frac{1}{2}y_3 - \frac{1}{6}T_4)(x_3 - x_2)x_3] \end{aligned} \quad (\text{A.41})$$

Finally, for the general case of $y_1 \leq y_2 \leq y_3$, while vertices are sorted with respect to the y coordinate, the solution for y^2 in equation (A.33) is re-formulated as

$$\begin{aligned} \int_{E_k} y^2 \, d\Omega &= \left| \frac{(x'_3 - x'_1)(y'_2 - y'_1)}{(y'_3 - y'_1)} - (x'_2 - x'_1) \right| \cdot \left[\frac{1}{4}(y'_2 - y'_1)^3 + \frac{2}{3}y'_1(y'_2 - y'_1)^2 + \frac{1}{2}(y'_2 - y'_1)y'^2_1 \right] \\ &\quad + \left| \frac{(x'_3 - x'_1)(y'_3 - y'_2)}{(y'_3 - y'_1)} - (x'_3 - x'_2) \right| \cdot \left[\frac{1}{4}(y'_3 - y'_2)^3 - \frac{2}{3}y'_3(y'_3 - y'_2)^2 + \frac{1}{2}(y'_3 - y'_2)y'^2_3 \right] \end{aligned} \quad (\text{A.42})$$

Appendix B

Current Density under CEM

This appendix briefly explain how current distribution is on the boundary is calculated.

Solving the forward equation for a mesh, the voltage of all nodes is available. The current density at the boundary could be calculated back from the voltage at the boundary surface. From the electromagnetic theory we have:

$$\vec{J} = \sigma \vec{E} = -\sigma \nabla V \quad (\text{B.1})$$

where J is the current density, \vec{E} is the electric field, V is the voltage and σ is the conductivity. By solving the forward equation we would only have voltage at each node. The relation between the voltage of each node and the voltage at the boundary surface is derived from the definition of interpolation function which is:

$$V(x, y, z) = \sum_{i=1}^4 u_i \phi_i(x, y, z) \quad (\text{B.2})$$

where V is the voltage, u_i is the voltage at the i^{th} node and ϕ_i is the corresponding interpolation function for the i^{th} node. The equation of interpolation functions are derived by solving the coefficients a, b, c, d in (B.3) using the voltages at each node.

Employing the matrix format of the equations makes calculation of the gradients much easier. Equation (B.3) expresses the matrix definition of equation (B.2) expressed in equation (2.17) reference [10].

$$V(x, y, z) = \begin{bmatrix} 1 & x & y & z \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} 1 & x & y & z \end{bmatrix} \begin{bmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{bmatrix}^{-1} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \quad (\text{B.3})$$

where $P_i(x_i, y_i, z_i)$ are the coordinates of the nodes for an element. Substituting relation of voltage inside an element with node voltages expressed in equation (B.3) into current density relation expressed in equation (B.1) would result in the relation between the current density and the voltage of the node as:

$$\vec{J} = -\sigma \nabla V = \left(\sigma \begin{bmatrix} b \\ c \\ d \end{bmatrix} \right) \quad (\text{B.4})$$

Finally, we need the value of current density normal to the boundary under the electrode area. The injected current pattern would be calculated by $\vec{J} \cdot \hat{n}$ where \hat{n} is the unit vector normal to the boundary surface for each element which is calculated by:

$$\hat{n} = \frac{(\vec{P}_1 - \vec{P}_2) \times (\vec{P}_1 - \vec{P}_3)}{|(\vec{P}_1 - \vec{P}_2) \times (\vec{P}_1 - \vec{P}_3)|} \quad (\text{B.5})$$

where \vec{P}_1 , \vec{P}_2 and \vec{P}_3 are those three nodes of each boundary element located at the boundary which form a triangular 2D boundary element. Since the voltage in the boundary region is the parameter of interest, it is possible to write equations (B.3) and (B.4) in the 2D format due to the fact that the effect of the off-boundary plane

point is zero in the boundary.

References

- [1] N. Polydorides and W. R. B. Lionheart, “A matlab toolkit for three-dimensional electrical impedance tomography: a contribution to the electrical impedance and diffuse optical reconstruction software project,” *Measurement Science and Technology*, vol. 13, no. 12, p. 1871, 2002.
- [2] A. Adler and W. R. B. Lionheart, “Uses and abuses of eiders: an extensible software base for eit,” *Physiological Measurement*, vol. 27, no. 5, p. S25, 2006.
- [3] Y. Ider, N. Gencer, E. Atalar, and H. Tosun, “Electrical impedance tomography of translationally uniform cylindrical objects with general cross-sectional boundaries,” *Medical Imaging, IEEE Transactions on*, vol. 9, no. 1, pp. 49–59, mar 1990.
- [4] K. Jerbi, W. Lionheart, P. Vauhkonen, and M. Vauhkonen, “Sensitivity matrix and reconstruction algorithm for eit assuming axial uniformity,” *Physiol. Meas.*, vol. 21, no. 1, p. 6166, 2000.
- [5] G. Hahn, J. Dittmar, A. Just, M. Quintel, and G. Hellige, “Different approaches to quantifying ventilation distribution and lung properties by functional eit,” *Physiol. Meas.*, vol. 31, no. 8, pp. S73–S84, 2010.
- [6] K.-S. Cheng, D. Isaacson, J. Newell, and D. Gisser, “Electrode models for electric current computed tomography,” *Biomedical Engineering, IEEE Transactions on*, vol. 36, no. 9, pp. 918–924, sept. 1989.
- [7] [Http://eiders3d.sourceforge.net](http://eiders3d.sourceforge.net).
- [8] W. Lionheart, “Uniqueness, shape, and dimension in eit,” *Annals of the New York Academy of Sciences*, vol. 873, no. 1, pp. 466–471, 1999.
- [9] D. Holder, *Electrical Impedance Tomography: Methods, History and Applications*. Institute of Physics, 2004.

- [10] B. Graham, "Enhancements in electrical impedance tomography (eit) image reconstruction for 3d lung imaging," Ph.D. dissertation, University of Ottawa, Canada, April 2007.
- [11] "Tomography" - Definition from the MerriamWebster Online Dictionary. Retrieved 2009-08-12.
- [12] [Http://en.wikipedia.org/wiki/Tomography](http://en.wikipedia.org/wiki/Tomography). Retrieved 2010-04-12.
- [13] "Computed Tomography" Definition from the Merriam-Webster Online Dictionary. Retrieved 2009-08-18.
- [14] [Http://en.wikipedia.org/wiki/Computed tomography](http://en.wikipedia.org/wiki/Computed_tomography). Retrieved 2010-04-12.
- [15] A. Kak and M. Slaney, "Principles of computerized tomographic imaging," *Society for Industrial and Applied Math*, 2001.
- [16] W. R. B. Lionheart, "Eit reconstruction algorithms: pitfalls, challenges and recent developments," *Physiological Measurement*, vol. 25, no. 1, p. 125, 2004.
- [17] A. Adler, "Measurement of pulmonary function with electrical impedance tomography," Ph.D. dissertation, Institut de Genie Biomedical Universite de Montreal, 1995.
- [18] T. Dai, "Image reconstruction in eit using advanced regularization frameworks," Ph.D. dissertation, Carleton University, Canada, 2008.
- [19] A. Adler and W. R. B. Lionheart, "Minimizing eit image artefacts from mesh variability in finite element models," *Physiological Measurement*, vol. 32, no. 7, p. 823, 2011.
- [20] B. Brown, "Electrical impedance tomography (eit): a review," *Journal of Medical Engineering & Technology*, vol. 27, no. 3, pp. 97–108, 2003.
- [21] F. Dickin and M. Wang, "Electrical resistance tomography for process applications," *Meas. Sci. Technol.*, vol. 7, no. 3, pp. 247–260, 1996.
- [22] R. Bayford, A. Gibson, A. Tizzard, T. Tidswell, and D. Holder, "Solving the forward problem in electrical impedance tomography for the human head using ideas (integrated design engineering analysis software), a finite element modelling tool," *Physiol. Meas.*, vol. 22, no. 1, pp. 55–64, 2001.

- [23] N. Avis and D. Barber, "Image reconstruction using non-adjacent drive configurations (electric impedance tomography)," *Physiol. Meas.*, vol. 15, no. 2A, pp. A153–A160, 1994.
- [24] A. Seagar and R. Bates, "Full-wave computed tomography. part 4: Low-frequency electric current ct," *Physical Science, Measurement and Instrumentation, Management and Education - Reviews, IEE Proceedings A*, vol. 132, no. 7, pp. 455–466, november 1985.
- [25] N. Soni, K. Paulsen, H. Dehghani, and A. Hartov, "Image reconstruction in electrical impedance tomography using the full set of maxwell equations," in *Scientific Abstracts from the 6th Conference on Biomedical Applications of Electrical Impedance Tomography*. University College London, June 2005.
- [26] P. Silvester and R. Ferrari, *Finite Elements for Electrical Engineers*, 3rd ed. Cambridge University Press, 1996.
- [27] T. Murai and Y. Kagawa, "Electrical impedance computed tomography based on a finite element model," *Biomedical Engineering, IEEE Transactions on*, vol. BME-32, no. 3, pp. 177–184, march 1985.
- [28] T. Pilkington, M. Morrow, and P. Stanley, "A comparison of finite element and integral equation formulations for the calculation of electrocardiographic potentials," *Biomedical Engineering, IEEE Transactions on*, vol. BME-32, no. 2, pp. 166–173, feb. 1985.
- [29] K. Huebner, D. Dewhirst, D. Smith, and T. Byrom, *The Finite Element Method for Engineers*, 4th ed. Wiley-Interscience, Aug 2001.
- [30] G. Strang and G. J. Fix, *An Analysis of the Finite Element Method*, 1st ed. New York: Prentice-Hall, 1973.
- [31] J. Sikora, S. Arridge, R. Bayford, and H. L., "The application of hybrid bem/femmethods to solve electrical impedance tomography forward problemfor the human head." Proc X ICEBI and V EIT, Gdansk, 2024 June 2004.
- [32] M. Vauhkonen, "Electrical impedance tomography and prior information," Ph.D. dissertation, Department of Applied Physics, Kuopio University, 1997.
- [33] M. Vauhkonen, D. Vadasz, P. Karjalainen, E. Somersalo, and J. Kaipio, "Tikhonov regularization and prior information in electrical impedance tomography," *Medical Imaging, IEEE Transactions on*, vol. 17, no. 2, pp. 285–293, april 1998.

- [34] R. Aster, B. Borchers, and C. Thurber, *Parameter Estimation and Inverse Problems*, I. Geophysics, Ed. Academic Press, June 2004.
- [35] P. C. Hansen, *Rank Deficient and Discrete Ill-Posed Problems: numerical aspects of linear inversion*. SIAM, Philadelphia, 1998.
- [36] D. Barber and B. Brown, “Errors in reconstruction of resistivity images using a linear reconstruction technique,” *Clin. Phys. Physiol. Meas.*, vol. 9, no. 4A, pp. 101–104, 1988.
- [37] W. Fan and H. Wang, “3d modelling of the human thorax for ventilation distribution measured through electrical impedance tomography,” *Measurement Science and Technology*, vol. 21, no. 11, p. 115801, 2010.
- [38] R. Wajman, R. Banasiak, L. Mazurkiewicz, T. Dyakowski, and S. D., “Spatial imaging with 3d capacitance measurements,” *Measurement Science and Technology*, vol. 17, no. 8, p. 21132118, 2006.
- [39] S. Moenickes, T. Taniguchi, R. Kaiser, and W. Zielke, “A 2.75 d finite element model of 3d fracture network systems,” in *Proceedings, 11th International Meshing Roundtable, IMR’02*, September 2002, pp. 161–168.
- [40] [Http://eidors3d.sourceforge.net/tutorial/dual_model/two_and_half_d.shtml](http://eidors3d.sourceforge.net/tutorial/dual_model/two_and_half_d.shtml), Retrieved on Dec 1, 2011.
- [41] M. Unsworth, B. Travis, and A. Chave, “Electromagnetic induction by a finite electric dipole source over a 2-d earth,” *Geophysics*, vol. 58, no. 2, pp. 198–214, 1993.
- [42] J. H. Coggon, “Electromagnetic and electrical modeling by the finite-element method,” *Geophysics*, vol. 36, no. 1, pp. 132–155, 1971.
- [43] D. Snyder, “A method for modeling the resistivity and induced polarization response of two-dimensional bodies,” *Geophysics*, vol. 41, pp. 997–1015, 1976.
- [44] R. C. Fox, G. Hohmann, J. Killpack, and R. L., “Topographic effects in resistivity and induced polarization surveys,” *Geophysics*, vol. 45, pp. 75–93, 1980.
- [45] M. Everett, “Mid-ocean ridge electromagnetics,” Ph.D. dissertation, University of Toronto, 1990.
- [46] W. Chew, M. Moghaddam, and E. Yannakakis, “Modeling of the subsurface interface radar,” in *Geoscience and Remote Sensing Symposium, 1990. IGARSS*

- '90. *'Remote Sensing Science for the Nineties'*, 10th Annual International, may 1990, p. 31.
- [47] P. Sjödal, T. Dahlin, and B. Zhou, "2.5d resistivity modeling of embankment dams to assess influence from geometry and material properties," *Geophysics*, vol. 71, no. 3, pp. G107–G114, May-June 2006.
- [48] B. Zhou, "Crosshole resistivity and acoustic velocity imaging, in 2.5-d helmholtz equation modeling and inversion," Ph.D. dissertation, The University of Adelaide, 1998.
- [49] P. Tsourlos, J. Szymanski, and J. Dittmer, "The topographic effect in earth resistivity arrays: a comparative study," in *Geoscience and Remote Sensing Symposium, 1995. IGARSS '95. 'Quantitative Remote Sensing for Science and Applications', International*, vol. 1, 10-14 1995, pp. 30–32 vol.1.
- [50] P. Queralt, J. Pous, and A. Marcuello, "2-d resistivity modeling; an approach to arrays parallel to the strike direction," *Geophysics*, vol. 56, no. 7, pp. 941–950, 1991.
- [51] G. N. Tsokas, P. I. Tsourlos, and J. E. Szymanski, "Square array resistivity anomalies and inhomogeneity ratio calculated by the finite-element method," *Geophysics*, vol. 62, no. 2, pp. 426–435, 1997.
- [52] E. Onegova, "Effect of multicoil electromagnetic tool eccentricity on measured signals," *Russian Geology and Geophysics*, vol. 51, no. 4, pp. 423 – 427, 2010.
- [53] F. N. Kong, S. E. Johnstad, T. R. sten, and H. Westerdahl, "A 2.5d finite-element-modeling difference method for marine csem modeling in stratified anisotropic media," *Geophysics*, vol. 73, no. 1, pp. F9–F19, 2008.
- [54] C. H. Stoyer and R. J. Greenfield, "Numerical solutions of the response of a two-dimensional earth to an oscillating magnetic dipole source," *Geophysics*, vol. 41, no. 3, pp. 519–530, 1976.
- [55] T. Wiik, B. Ursin, and H. K., "2.5d em modeling in tiv conductive media," October 2010, submitted to *Geophysics* 2009.
- [56] T. Wiik, B. Ursin, and K. Hokstad, "2.5d frequency domain em modeling in conductive tiv media," in *71st EAGE Conference, Amsterdam, Extended abstracts, P076*. SPE, EAGE, June 2009.

- [57] S. Jin and Y. Lam, “2.5d cavity balancing,” *Journal of Injection Molding Technology*, vol. 6, no. 4, pp. 284–296, Jan 2002.
- [58] D. R. Stephenson, R. Mann, and T. A. York, “The sensitivity of reconstructed images and process engineering metrics to key choices in practical electrical impedance tomography,” *Measurement Science and Technology*, vol. 19, no. 9, p. 094013, 2008.
- [59] [Http://www.itoms.com/](http://www.itoms.com/).
- [60] T. Rodgers, D. Stephenson, M. Cooke, T. York, and R. Mann, “Tomographic imaging during semi-batch reactive precipitation of barium sulphate in a stirred vessel,” *Chemical Engineering Research and Design*, vol. 87, no. 4, pp. 615 – 626, april 2009, 13th European Conference on Mixing: New developments towards more efficient and sustainable operations.
- [61] D. Stephenson, T. York, and R. Mann, “Performance and requirements of process ert instruments,” in *5th World Congress on Industrial Process Tomography*. WCIPT5 Bergen, Norway, Sep. 2007, pp. 895–901.
- [62] J. E. Butler and R. T. Bonnecaze, “Imaging of particle shear migration with electrical impedance tomography,” *Physics of Fluids*, vol. 11, no. 8, pp. 1982–1994, 1999.
- [63] P. J. Holden, M. Wang, R. Mann, F. J. Dickin, and R. B. Edwards, “Imaging stirred-vessel macromixing using electrical resistance tomography,” *AIChE Journal*, vol. 44, no. 4, pp. 780–790, 1998.
- [64] S. Stanley, “Tomographic imaging during reactive precipitation in a stirred vessel: Mixing with chemical reaction,” *Chemical Engineering Science*, vol. 61, no. 24, pp. 7850 – 7863, 2006.
- [65] M. Kagoshima and R. Mann, “Interactions of precipitation and fluid mixing with model validation by electrical tomography,” *Chemical Engineering Research and Design*, vol. 83, no. 7, pp. 806 – 810, 2005, 7th World Congress of Chemical Engineering.
- [66] A. Dey and H. F. Morrison, “Resistivity modelling for arbitrarily shaped two-dimensional structures,” *Geophysical Prospecting*, vol. 27, no. 1, pp. 106–136, 1979.

- [67] T. Gnther, C. Rcker, and K. Spitzer, “Three-dimensional modelling and inversion of dc resistivity data incorporating topography ii. inversion,” *Geophysical Journal International*, vol. 166, no. 2, pp. 506–517, 2006.
- [68] M. Vauhkonen, M. Hamsch, and C. H. Igney, “A measurement system and image reconstruction in magnetic induction tomography,” *Physiological Measurement*, vol. 29, no. 6, p. S445S454, 2008.