CHAPTER 2

RECONSTRUCTION OF ELECTRICAL IMPEDANCE TOMOGRAPHY IMAGES

In this chapter a detailed reconstruction technique adopted is presented. In addition, the theoretical formulations involved in the computation of finite element method (FEM) are also discussed.

2.1 INTRODUCTION TO RECONSTRUCTION

The procedure for computing an image from the signal is known as reconstruction. The reconstruction algorithms may be classified into numerous categories for EIT. They are difference imaging, static imaging, dynamic imaging and multiple frequencies imaging (Graham 2007).

2.1.1 Difference Imaging Method

This system is commonly used to generate the images based on the impedance changes for different time period. A first set of data $v_1$ is acquired at a time $t_1$ and a second set of data $v_2$ is acquired at a time $t_2$. Then changes in the conductivity from period $t_1$ to $t_2$ are calculated. Adler et al. (1996) used this method for imaging an impedance changes during respiration. The image stability is affected due to unknown contact impedance, incorrect electrode positions, poorly known boundary shape and non-linearity. This method is generally used to increase reconstructed image stability. The computation of conductivity change is done using a linear approximation operator. This linear
operator is established as a Jacobian or sensitivity matrix. This matrix is discussed in a following section.

2.1.2 Jacobian Matrix

The Jacobian matrix \( H \) is modelled as \( M \times E \) matrix, in which \( E \) is the elements and \( M \) is the boundary measurements. The Jacobian matrix is computed column by column in which the \( i^{th} \) column relating the conductivity changes of the \( i^{th} \) element on the signal \( z \). The signal \( z \) is

\[
\begin{align*}
z &= v_2 - v_1 \\
\end{align*}
\]

(2.1)

where \( v_1 \) is voltage measurements acquired at time \( t_1 \) and \( v_2 \) is the voltage measurement acquired at time \( t_2 \). The approximation of difference image is computed using following equation

\[
\hat{x} = Bz
\]

(2.2)

where \( \hat{x} \) is the conductivity changes between time \( t_1 \) and \( t_2 \) and \( B \) is regularized linearized reconstruction operator. For difference imaging, \( \hat{x} = \Delta\sigma = \sigma_2 - \sigma_1 \) is the change in a finite element conductivity distribution due to a change in signal \( z \) due to a time interval \( (t_1, t_2) \). \( \hat{x} \) is taken as the conductivity changes with respect to the unknown initial conductivity \( x = \Delta\sigma \), because \( \sigma_1 \) is unknown. The Jacobian for a linearized forward problem is designed as follows,

\[
\begin{align*}
z &= Hx + n \\
\end{align*}
\]

(2.3)

Where \( n \) is the measurement system noise. In Jacobian matrix, each element \( i, j \) is calculated as
and relates a small change in the $i^{th}$ difference measurement to a small change in the conductivity of $j^{th}$ element as (Adler 1996) mentioned. $H$ is a function of the FEM, the current injection pattern and the background conductivity. For every elements, the homogeneous background conductivity is used as $\sigma_0=1$. In case of forward model, to compute the jacobian matrix $H$, the difference signal $z$ is stated as

$$z = T[V(\sigma_2)] - T[V(\sigma_1)]$$

(2.5)

Where $T[]$ is an extraction operator which produces the measurements between electrodes from the nodal voltage matrix $V$. By assuming that the conductivity changes may occur only a small amount between the two different time and those value we can use $\sigma_1=\sigma$ and $\sigma_2=\sigma + \Delta \sigma$. By substituting this values in equation (2.5) which gives

$$z = T[V(\sigma)] - T[V(\sigma + \Delta \sigma)]$$

(2.6)

More manipulation gives

$$z = T \left[ - \frac{V(\sigma + \Delta \sigma) - V(\sigma)}{\Delta \sigma} \right]$$

(2.7)

In the limit as $\Delta \sigma \to 0$:

$$\lim_{\Delta \sigma \to 0} \frac{V(\sigma + \Delta \sigma) - V(\sigma)}{\Delta \sigma} = \frac{\partial V(\sigma)}{\partial \sigma}$$

(2.8)

By ignoring noise, this equation allow us to write the linearized form, as equation (2.3) as
\[ z = T \left[ -\frac{\partial V(\sigma)}{\partial \sigma} \right] \Delta \sigma \]  

(2.9)

Where the Jacobian is

\[ H = T \left[ -\frac{\partial V(\sigma)}{\partial \sigma} \right] \]  

(2.10)

And the terms of the Finite Element Model is

\[ \frac{\partial V(\sigma)}{\partial \sigma} = \frac{\partial}{\partial \sigma} (Y^{-1}(\sigma))l \]  

(2.11)

The chain rule offers the final form of the Jacobian in terms of the FEM which can be used as the basis for an algorithm to calculate the Jacobian matrix. Substituting the equation (2.11) into the equation (2.10), which gives

\[ H = T \left[ -\frac{\partial}{\partial \sigma} (Y^{-1}(\sigma))l \right] \]

and rearranging the above equation, we will get

\[ H = T \left[ Y^{-1}(\sigma) \frac{\partial}{\partial \sigma} Y(\sigma)Y^{-1}(\sigma)l \right] \]  

(2.12)

From equation (2.12), the derivative of the stiffness matrix must be computed. With linear basis functions, the derivative is a constant for matrix elements of the given element and zero elsewhere. Due to a small change in \( \sigma_i \), the another vector is representing the change in the M voltages.

\[ \frac{\partial}{\partial \sigma} (Y^{-1}(\sigma))l \in E \]  

(2.13)
The result is prearranged in columns of length M with each column is representing \( \frac{\partial}{\partial \sigma_i} \left( Y^{-1}(\sigma) \right) \). Thus the Jacobian matrix H is M x E. The standard method (Vauhkonen et al. 2001; Vauhkonen 1998) has been used for calculating the Jacobian Matrix. But the concept of Measurement fields (Breckon 1990) is a more efficient method to calculate the Jacobian. The measurement fields are defined as the fields that would have been developed if the currents were injected from the measurement electrodes. If we denote \( \nabla \phi \) as the gradient of the current fields obtained by normal forward solution and \( \nabla \psi \) as the gradient of the measurement fields then the \( i^{th} \) column of the Jacobian corresponding to the \( i^{th} \) element is given by the dot product of the two fields integrated over each mesh elements as:

\[
\frac{\partial U^k}{\partial \sigma_i} = -\int_i \nabla \phi \cdot \nabla \psi
\]  

(2.14)

The Naïve approach is used to solving the linearized problem, from equation (2.3), for x is to find the least squares solution. This is to use Moore-Penrose generalized inverse

\[
\hat{x} = (H^T H)^{-1} H^T z
\]  

(2.15)

which is the least square solution in that

\[
\hat{x} = \arg \min_x \| Hx - z \|
\]  

(2.16)

Unfortunately, the equation cannot be solved in this way because \( H^T H \) is rank deficient and ill-conditioned. In ill-conditioned problem, a small change of the input data can leads to a large change in the output.
2.1.2.1 **Ill Posed Problems**

In 1902 Jacques Hadamard classified all mathematical problems as either well-posed or ill-posed based on the three conditions which are existence, uniqueness and stability. For a well posed problem, a solution must exist and be unique, and small changes in the data must not be a large change in the solution. For an ill-posed problem, at least one of the three conditions may not be satisfied. In terms of EIT, lack of stability is the most tedious problem that the inverse solution will be dominated by noise unless additional conditions are imposed. Ill-posed problem can only exist in the continuous domain. But under the assumption of infinite precision arithmetic, the discrete problem is never ill-posed. However with finite precision arithmetic, the discrete of an ill-posed problem can leads to a numerical problem that is ill-conditioned. The ill-conditioning of problem is defined by condition number of its matrix. To define the condition number requires an understanding of the Singular Value Decomposition (SVD) of a matrix.

2.1.2.2 **Singular Value Decomposition (SVD)**

The singular value decomposition is a method of factoring a matrix that does not require the matrix to be either symmetric or have full rank. Detailed theory and examples can be found in (Hansen 1998; Hansen 1994; Vogel 2002). Let $A \in \mathbb{R}^{m \times n}$ be a rectangular matrix with $m \geq n$. Then the SVD of $A$ is a decomposition of the form

$$ A = U \Sigma V^T = \sum_{i=1}^{n} u_i \sigma_i v_i^T \quad (2.17) $$

Where $U = (u_1, ..., u_n)$ and $V = (v_1, ..., v_n)$ are matrices with orthonormal columns, $U^T U = V^T V = I_n$ and where $\Sigma = \text{diag}(\sigma_1, ..., \sigma_n)$ has non-negative
diagonal elements, which by convention, are arranged in non-increasing order such that

$$\sigma_1 \geq \cdots \geq \sigma_n \geq 0$$

The numbers $\sigma_i$ are the singular values of $A$ while $u$ and $v$ are the left and right singular vectors of $A$ respectively. The condition number of $A$ is equal to the ratio $\sigma_1 / \sigma_n$. A problem with a low condition number is said to be well-conditioned and a problem with a high condition number is said to be ill-conditioned. In terms of EIT reconstruction, the noise in the data will be amplified by the inversion of the small singular values. So the solution to equation (2.18) will be dominated by the noise in the signal. To overcome the ill-conditioning of $H^T H$ requires the use of regularization techniques. These techniques generally involve either truncating or filtering the singular values corresponding to the singular vectors with high frequency components.

### 2.1.2.3 Regularization

A regularization method is defined as an inversion method depending on a single real parameter $\lambda \geq 0$, which yields a family of approximate solutions (Karl 2000). There are different types of discrete regularization techniques are used such as truncated singular value decomposition, maximum entropy, and a number of generalized least squares methods including Twomey and Tikhonov regularization methods (Cheney 1995). All these methods are try to reduce the effect of solving ill-conditioned system by restoring continuity of the solution on the data (Cheney 1995).

The Tikhonov regularization is commonly used method in EIT. Prior information is incorporated into the solution as an additional term in the least squares minimization. That is
Here $\mathbf{R}$ is a regularization matrix which is commonly diagonal or banded diagonal and $\lambda^2\|\mathbf{Rx}\|^2$ is some prior information about the conductivity. This is a quadratic minimization which is certain to have a unique solution for $\lambda > 0$. In EIT, identity matrix and the matrices corresponding to first and second difference operators (Vauhkonen et al. 1998) are commonly used. When these matrices are used, the resultant implied prior assumptions are $x$ is small, slowly changing or smooth. Classic Tikhonov regularization denotes to the $\mathbf{R}=\mathbf{I}$, but Tikhonov is mostly used to any solution of equation (2.18) irrespective of the choice of $\mathbf{R}$. The solution to equation (2.18) is calculated from the regularized inverse

$$
\mathbf{x} = (\mathbf{H}^T \mathbf{H} + \lambda^2 \mathbf{R}^T \mathbf{R})^{-1} \mathbf{H}^T \mathbf{z} = \mathbf{Bz}
$$

Here $\mathbf{B}$ is the regularized linearized reconstruction operator and this equation is a linear equation which must be solved iteratively for the non-linear solution (Heath 2002). The regularized inverse has two important properties:

1. For large enough $\lambda$ the regularized solution, $\mathbf{x}(\lambda)$ is stable in the face of perturbations or noise in the data and,

2. As $\lambda$ goes to zero, the un-regularized generalized solution is recovered: $\mathbf{x}(\lambda) \rightarrow \mathbf{x}$ as $\lambda \rightarrow 0$.

The parameter $\lambda$ is called the regularized parameter or hyper parameter which controls the trade-off between solution stability and nearness of the regularized solution (Adler et al. 1996), $\mathbf{x}(\lambda)$, to the un-regularized solution $\mathbf{x}$. This can be assumed as the approximation error in the absence of measurement noise and discretization noise due to finite precision arithmetic. Equation (2.19) is the most general form of reconstruction model in EIT and most of the algorithms are constructed from this model.
2.1.3 Static Imaging

In EIT, various groups have been proposed the static image reconstruction. The basic technique is to use a modified Gauss-Newton algorithm with Tikhonov regularization. In this technique, an error function is defined as

\[
f(\sigma) = \frac{1}{2} (\|T[V(\sigma)] - \nu_{\text{measured}}\|^2 + \lambda^2 \|\sigma - \sigma_0\|^2) \quad (2.20)
\]

where \(\nu_{\text{measured}}\) is a vector of voltage measurements from the physical medium, \(\sigma_0\) is the initial estimate of the background conductivity, and \(V(\sigma)\) is the forward operator which simulates the voltage measurements from a medium with conductivity distribution \(\sigma\). The desired reconstructed conductivity distribution is the vector \(\sigma\) that minimizes \(f\). The non-linear solution of equation (2.20) is solved iteratively using a linearized step at each iteration. In this process, a set of currents is injected into the medium and the resulting voltage recorded. This data is compared to voltages, \(\nu_{\text{simulated}} = T[V(\sigma)]\), generated by simulating the same process (current injection and voltage measurement) on a Finite Element Model (FEM) of the medium. Initially, the medium is assumed to be homogeneous (Adler 1995). If the simulated data approximates the measured data by some measure then the conductivity of the model is assumed to approximate the conductivity of the medium and the problem is solved. If the simulated data does not approximate the medium then iteration is executed. The steps of the iterative Gauss-Newton method for static reconstruction are as follows:

1. Obtain an initial approximation for the conductivity distribution. The initial conductivity (\(\sigma_0\)) distribution of the model is \textit{apriori} assumption about the conductivity distribution of the medium. But, it is often a rough estimate
of the equivalent homogenous conductivity of the medium based on the data.

2. To obtain the simulated measurements ($\mathbf{v}_{simulated}$), solve the forward problem.

3. Compute the conductivity change,

$$\Delta \sigma = (H^T H + \lambda^2 R^T R)^{-1} H^T z$$  \hspace{1cm} (2.21)

Where $z = \mathbf{v}_{measured} - \mathbf{v}_{simulated}$ \hspace{1cm} (2.22)

4. Update the absolute conductivity,

$$\sigma_{k+1} = \sigma_k + \Delta \sigma$$  \hspace{1cm} (2.23)

where $\sigma_0$ is an initial conductivity and also vector of length E. The Jacobian is used for static imaging in which each element is $H_{ij} = \frac{\partial z_i}{\partial \sigma_j} |_{\sigma_k}$ and relates a small change in the $i^{th}$ error measurement ($z_i$) to a small change in the conductivity of $j^{th}$ element. H is a function of current injection pattern. Thus calculation of the Jacobian is identical for both difference and static imaging but with static imaging the signal is the error signal (2.22) and the conductivity change is used in the iterative buildup of the absolute conductivity via equation (2.23).

5. Update the admittance matrix with the current estimation of the conductivity, $Y (\sigma_{k+1})$.

6. Testing the stopping condition. The stopping condition is a single iteration or after some fixed number of iterations or after the difference between the two sets of measurements drops below some threshold, i.e. $\varepsilon \leq \| \mathbf{v}_{measured} - \mathbf{v}_{simulated} \|$. If the current solution
satisfies the stopping condition then exit, otherwise continue to step 7.

7. Update the Jacobian based on the current estimate of the conductivity.

8. Go to step 2. Where \( \mathbf{v}_{\text{simulated}} \) calculated at step 2 is a function of the iteration number, \( k \).

Equation (2.21) is similar to the difference image equation (2.18) with \( \hat{x} = \Delta \sigma \) and \( z \) is defined as the difference between the measured voltages and the set of simulated voltages. Frequently the regularization matrices are also the same as those used in difference imaging.

### 2.1.3.1 MAP Regularized Inverse

The commonly used reconstruction model for 2D difference imaging of Maximum a Posteriori (MAP) algorithm was introduced by Adler & Guardo (1996). This algorithm gives the solution as the most likely estimate of \( \hat{x} \) from given the measured signal \( z \) and certain statistical information about the medium. In order to simplify the reconstruction algorithm the image statistical properties are modeled by a Gaussian distribution of mean \( x_{\infty} \) and covariance \( R_x \)

\[
x_{\infty} = E(x) \\
R_x = E(x - x_{\infty}) = E(x^T x) - x_{\infty}^T x_{\infty}
\]

with these parameters, the distribution function of the image \( f(x) \) is modeled as
The a posteriori distribution function of \( z \) given a conductivity distribution \( x \) is derived from the definition of the inverse problem, equation (2.3): \( f(x) = \frac{1}{2\pi^{N/2} \sqrt{|R_x|}} e^{-(1/2)(x-x_{\infty})^T R_x^{-1} (x-x_{\infty})} \) (2.25)

The difference \( (z-Hx) \) is completely due to the noise \( n \) which is assumed to be Gaussian, white, zero mean with covariance \( R_n \). Thus

\[
R_n = E[n^T n] = \begin{bmatrix}
\sigma_n^2 & 0 & \ldots & 0 \\
0 & \sigma_n^2 & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & \sigma_n^2
\end{bmatrix}
\] (2.27)

where \( \sigma \) is represents the square root of the variance. The MAP estimate \( (\hat{x}) \) maximizes the a posteriori probability distribution \( f(x|z) \). This can be find the image, \( x \),

\[
f(x|z) = \frac{f(z|x) f(x)}{f(z)}
\] (2.28)

\[
f(x|z) = \frac{e^{-\frac{1}{2}[(z-Hx)^T R_n^{-1} (z-Hx) + (x-x_{\infty})^T R_x^{-1} (x-x_{\infty})]}}{2\pi^{(M+N)/2} \sqrt{|R_n||R_x|} f(z)}
\] (2.29)

Here \( f(x|z) \) is maximized when the exponent is minimized

\[
\hat{x} = \arg \min_x [(z-Hx)^T R_n^{-1} (z-Hx) + (x-x_{\infty})^T R_x^{-1} (x-x_{\infty})]
\] (2.30)

which yields the estimate
\[ \hat{x} = (H^T R_n^{-1} H + R_x^{-1})^{-1} (H^T R_n^{-1} z + R_x^{-1} x_\infty) \quad (2.31) \]

The noise covariance \( R_n \) measures the noise power in each component of the signal. Measurement noise on each channel of the scanner can be determined from the hardware. Adler (1996) take the case where each channel has equal noise variance \( \sigma_n \). Using the signal, \( z \), \( R_n \) is calculated as \( [R_n]_{ii} = (\sigma_n / \psi_i^h)^2 \) where \( \psi_i^h \) are the measurements from the medium on which the noise measurements were taken.

The properties of the image \( R_n \) and \( x_\infty \) are less concrete than the noise properties and can be estimated from a knowledge of the experimental configuration. The expected change in the image \( E[x] \) is represented by \( x_\infty \). The diagonal elements of \( [R_x]_{ii} \) represent the variance of amplitude of each image element, whereas off diagonal elements are a function of the correlation coefficient \( r \) between a pixel in element \( I \) and a pixel in element \( j \) as follows,

\[ [R_x]_{ij} = \sqrt{[R_x]_{ii}[R_x]_{jj}} \quad (2.32) \]

Meanwhile EIT has low spatial resolution due to the small number of measurements, indicating that the spatial frequency of the reconstructed distribution of conductivity change has little high frequency content. Therefore elements close to each other will have correlated reconstruction values. Adler (1996) set diagonal elements of \( R_x \) to \( \sigma_x^2 \) and they assume that the pixels closer than this distance are highly correlated and pixels further away from each other are not correlated with a gradual diminishing between the two extremes. Therefore \( R_x \) is interpreted as a low pass filter. Since the formulation \( R_x \) is numerically unstable a regularization matrix \( Q \), was constructed directly using a high pass filter to represent \( R_x^{-1} \).
Adler (1996) define the regularization hyper parameter in terms of the statistical standard deviations as \( \lambda = \frac{\sigma_\text{noise}^2}{\sigma_n^2} \) from which they arrive at the MAP regularized inverse

\[
\hat{x} = (H^T WH + \lambda Q)^{-1} H^T W z = B(\lambda) z
\]  
(2.33)

Where \( B(\lambda) \) is the image reconstruction matrix. Then noise is uncorrelated in the system. \( W \) is a diagonal matrix with \( W_{ii} = 1/\sigma_i^2 \) where \( \sigma_i^2 \) is the noise variance for measurement \( i \).

One more important consideration is to ensure the proper electrode measurement. Electrodes can be detached, the contact impedance can change because of sweat or peripheral oedema, and changes in subject posture can move electrodes and corrupt readings (Meeson 1996; Adler 1996). Electrode errors indicate high noise, which is reflected in the measurement noise variance values contained in the diagonal elements of \( R_n \) of equation (2.31). Thus, if measurement \( i \) is subject to increased noise (by a factor \( \sigma_n^2 \)), \( W_{ii} \) is reduced by the same factor. If an electrode becomes completely disconnected, or noise levels are very high, the noise variance, \( \sigma_n \), can be assumed to be infinite, and the corresponding elements of \( W \) set to \( 1/\infty = 0 \). Electrode errors will reduce the number of available measurements.

### 2.2 FINITE ELEMENT METHOD

The Finite Element Method (FEM) is a numerical analysis technique for obtaining approximate solutions to a wide variety of engineering problems. It was originally developed as a tool for aircraft design but has since been extended too many other fields including the modeling of electromagnetic and electrostatic fields (Silvester 1996). Due to its ability to model arbitrary geometries and various boundary conditions the Finite
Element Method is the most common method currently used for the numerical solution of EIT problems because of its ability to model arbitrary geometries and various boundary conditions (Murai 1985; Pilkington 1985). In a continuum problem of any dimension, the field variable, such as the electric potential in EIT, is defined over an infinite number of values since it is a function of the infinite number of points in the body. The finite element method first discretizes the medium under analysis into a finite number of elements (finite element mesh). Within each element, the field variable is approximated by simple functions that are defined only within the individual element. The approximating functions (sometimes called interpolation or shape functions) are defined in terms of the values of the field variables at specified points on the element called nodes. Most EIT work uses linear shape functions in which all nodes lie on the element boundaries where adjacent elements are connected. Higher order shape functions will have interior nodes (Huebner 2001; Graham 2007).

There are three different methods normally used to formulate Finite Element problems:

1. Direct approach. This approach is created from the direct stiffness method of structural analysis. This method has limitation, but it is the most intuitive way to understand the finite element method.

2. Variational approach. Element properties can be determined by the Variational approach which relies on the calculus of variations and involves extremizing a functional such as the potential energy. The variational approach is necessary to extend the finite element method to a class of problems that cannot be handled by direct methods. For example problems involving elements with non-constant conductivity, for
problems using higher order interpolation functions and for
element shapes other than triangles and tetrahedrons.

3. Method of Weighted Residuals (MWR). The most adaptable
approach to deriving element properties is the Method of
Weighted Residuals. The weighted residuals approach
begins with the governing equations of the problem and
proceeds without relying on a variational statement. This
approach can be used to extend the finite element method to
problems where no functional is available.

The finite element method solves the continuum problems with the following
sequence of operations (Singiresu 2004):

1. Discretize the continuum: The first step in the FE method is
to divide the solution region into the subdivisions or
elements. The FE method consists of discretizing the spatial
domain (Ω) into a number of non-uniform, non-overlapping,
elements connected via nodes. The number, size, type, and
arrangement of the elements are to be decided. Triangles and
rectangles are used in 2D problems while tetrahedral and
hexahedral elements are used for 3D. Additionally meshes
using a mixture of different types of elements are possible.

2. Selection of a proper interpolation: Assume some
appropriate solution within an element to approximate the
unknown solution. The assumed solution must be simple
from a computational point of view, but it should satisfy
certain convergence requirements. Interpolation functions
can be any piecewise polynomial function defined at a
number of nodes.
3. Find the element properties: From the assumed model, the local matrix for each element is to be calculated. The symbol Y with appropriate subscripts is used to denote the components of local matrices throughout this thesis.

4. Assemble the element properties to obtain a system equation. The structure composed of several elements; the individual element local matrices are to be assembled into a single master or global matrix. The symbol Y or A without subscripts are used throughout to denote the global matrix.

5. Impose the Boundary Conditions (BC). The overall system equations to be modified to description for the boundary conditions of the problems. Boundary conditions can be fixed (also known as Type I, Dirichlet, or essential boundary conditions), derivative (also known as Type II, Neumann, or natural boundary conditions), or a combination of both (mixed, also known as Type III boundary conditions). With Dirichlet boundary conditions the value of the field variable is prescribed for selected boundary nodes. For Neumann conditions the derivative of the field variable is prescribed at selected boundary nodes.

6. Solve the system of equations. The algebraic system is of the form \( YV = I \). The system is solved by Linear Algebra software such as Matlab.

7. Make additional computations if desired. This would be required for iterative algorithms, algorithms that use adaptive currents and adaptive mesh refinement.
2.2.1 Direct Approach

The Direct Approach can be used to simple types of elements and simple problems (Silvester & Ferrari 1996; Hua et al. 1993; Holder 2005; Murai & Kagawa 1995; Holder 1992). In this method, direct physical reasoning is used to establish the element properties. In EIT, the direct method can be used for finite elements of constant conductivity with linear shape functions. The finite element model for EIT is equivalent to a linear electrical network (lumped resistor network) that connects the nodes (Murai & Kagawa 1985). In the 2D case, a triangle such as Figure 2.1(a) is equivalent to the electrical network of Figure 2.1(b).

![Figure 2.1](image)

**Figure 2.1 Derivation from resistor network, a) Triangular element, b) Equivalent circuit**

In this conversion each edge of the triangle is replaced by a resistor whose conductance is \( \sigma \cot \theta_j \), where resistor \( j \) is the resistor opposite the \( j^{th} \) angle (David 2005). The 3D case is similar with \( \theta_j \) being the angle between the two faces meeting at the \( j^{th} \) edge. For nodal coordinates, the conductance \( Y_{ij} \) between node \( i \) and node \( j \) is determined by the triangle-to-network conversion as

\[
Y_{ij} = \frac{\sigma_e}{2A_e} (b_i b_j + c_i c_j), \quad (i \neq j)
\]  
(2.34)
with \( b_1 = y_2 - y_3, b_2 = y_3 - y_1, b_3 = y_1 - y_2 \) and \( c_1 = x_3 - x_2, c_2 = x_1 - x_3, c_3 = x_2 - x_1 \) where \((x_i, y_i) (i = 1, 2, 3)\) denotes a coordinate of each node, \(A_e\) is the area of an element and \(\sigma_e\) is the element conductivity which is assumed to be constant. Kirchhoff’s current law for the circuit is written as

\[
\begin{bmatrix}
Y_{11} & Y_{12} & Y_{13} \\
Y_{21} & Y_{22} & Y_{23} \\
Y_{31} & Y_{32} & Y_{33}
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2 \\
u_3
\end{bmatrix}
= \begin{bmatrix}c_1 \\
c_2 \\
c_3\end{bmatrix}
\text{ or } Y^e U^e = I^e
\tag{2.35}
\]

With \( Y_{11} = -Y_{12} - Y_{13}, Y_{22} = -Y_{21} - Y_{23}, Y_{33} = -Y_{31} - Y_{32}, Y_{ij} = Y_{ji} \) for \( i = 1, 2, 3 \). Where \( x_i (i = 1, 2, 3) \) are nodal potentials and \( c_i (i = 1, 2, 3) \) is the prescribed current which flows in the \( i^{th} \) node. The two element mesh is used to assembling of the global admittance matrix.

**Figure 2.2  Connection of two elements** (Murai & Kagawa 1985)

The main matrix \( Y \) is assembled with the conductances between adjacent elements adding in parallel as shown in Figure 2.2. These two elements share nodes 2 and 4. But \( Y_{24} \) will be different for each triangle since conductivity and the geometry is different for each element. The local matrices are:
\[ Y^{(1)}_{ij} = \begin{bmatrix} Y_{11} & Y_{12} & Y_{14} \\ Y_{21} & Y_{22} & Y_{24} \\ Y_{31} & Y_{32} & Y_{34} \end{bmatrix}, \quad i,j \in [1,2,4] \] are the global node indices for element 1

\[ Y^{(2)}_{ij} = \begin{bmatrix} Y_{22} & Y_{23} & Y_{24} \\ Y_{32} & Y_{33} & Y_{34} \\ Y_{42} & Y_{43} & Y_{44} \end{bmatrix}, \quad i,j \in [2,3,4] \] are the global node indices for element 2

These two are combined as

\[ Y = \begin{bmatrix} Y_{11} & Y_{12} & Y_{14}Y_{14} \\ Y_{21} & Y_{22} & Y_{24}Y_{24} \\ Y_{31} & Y_{32} & Y_{33}Y_{34} \\ Y_{41} & Y_{42} & Y_{43}Y_{44} \end{bmatrix}, \quad i,j \in [1:4] \]

The four common boundary conditions in EIT are Continuum, Gap, Shunt and Complete Electrode Models (Vauhkonen 1997). The Gap electrode model is simplest model to implement numerically. In this model, electrodes are connected directly to selected nodes on the boundary of the FEM. With the adjacent current pattern, current is applied at the two boundary nodes that represent a pair of electrodes while the currents at the remaining nodes are set to 0 in accordance with Kirchhoff’s current law. The resulting potential field is calculated by solving the following equations

\[ YV = I \quad (2.36) \]

where \( Y \) is the global admittance matrix which is a function of the FEM and the conductivity, \( V \) is a vector of nodal voltages

\[ V = [u_1, u_2, \ldots, u_N]^T \quad (2.37) \]

and \( I \) is the current vector which for the adjacent drive is a permutation of
The non-zeros represent the current driven between a pair of electrodes whereas the zeros represent the current at each node which is zero by Kirchoff’s current law. The two non-zero elements of equation 2.38 are not necessarily adjacent elements of the vector $I$. By convention $\pm 1$ are used as the current values. Equation (2.36) is solved for the nodal voltages as $V = Y^{-1}I$. Therefore the Direct approach along with the Gap model, the forward problem can be viewed as using Kirchoff’s current law to solve a large network assembled as a set of simultaneous linear equations. With the adjacent current pattern equation (2.36) is solved once for each pair of electrodes being driven thus $I$ can be assembled as a matrix where each column is a permutation of the vector equation (2.38) and $V$ then becomes a matrix in which each column contains the nodal voltage for a single current injection. The entire algebraic system has the form:

$$I = [0, 0, ..., -1, 1, ..., 0]^T$$  \hspace{1cm} (2.38)

where $u_{ij}$ is the voltage at the $i^{th}$ node due to the $j^{th}$ current injection pattern whereas $c_{ij}$ is the current at the $i^{th}$ node during the $j^{th}$ current injection pattern.

From equation (2.38), each column of $I$ has only two non-zero entries and is a permutation of $I = [0, 0, ..., -1, 1, ..., 0]^T$. In the Gap model map to a single node, each the voltages measured between a pair of electrodes is determined by the difference between two nodal values where the specific nodes are those corresponding to the electrodes. The voltages measured between adjacent electrodes are collected into a column vector by using an extraction operator, $T \left[ \right]$. 

$$\begin{bmatrix} u_{11} & \cdots & \cdots & u_{1M} \\ \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ u_{1M} & \cdots & \cdots & u_{NM} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & Y_{22} & \cdots & Y_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & Y_{N2} & \cdots & Y_{NN} \end{bmatrix}^{-1} \begin{bmatrix} c_{11} & \cdots & \cdots & c_{1M} \\ \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ c_{1M} & \cdots & \cdots & c_{NM} \end{bmatrix}$$  \hspace{1cm} (2.39)
By solving equation (2.36) for V requires the inversion of Y. Even though Y is square and sparse it is also singular. To make the system non-singular, a reference node is selected. For suitability node 1 is arbitrarily selected. To implement this in the linear equation, all entries in row 1 and column 1 of the admittance matrix are set to 0 and the diagonal elements are set to 1. To ensure that the potential will remain zero at that node during each injection pattern the corresponding element of each current vector in I is set to zero. The result of equation (2.39) delivers the potential values at the nodes. Interpolation functions are used to calculate the potential within each element.

2.2.2 Method of Weighted Residuals (MWR)

There are several methods are used in weighted residuals. They are least squares, Collocation, subdomain Collocation and Galerkin methods. The development of the MWR starts with a discretization in which the field variable is represented as a linear combination of piecewise polynomial interpolation functions of limited support:

$$\hat{u}(\vec{x}) = \sum_{i=1}^{N} u_i \phi_i(\vec{x}) \text{ where } \phi_i = \begin{cases} 1 & \text{on vertex } i \\ 0 & \text{otherwise} \end{cases} \quad (2.40)$$

Where $\phi_i$ are the interpolation functions and N is the number of nodes in the FEM. Since $\hat{u}$ represents only a finite approximation of the potential, the Laplacian is not equal to zero due to the error introduced by using the approximating functions. The weighted residuals method proceeds by deriving the weak form of the governing equation through the multiplication of Laplace’s equation by some arbitrary test function $v$ and integration over the domain, $\Omega$: 
\[
\int_\Omega v[\nabla \cdot (\sigma \nabla \bar{u})] d\Omega = 0 \tag{2.41}
\]

Here \(v\) is an arbitrary test function that weighs the residual such that it is zero in some weighted or average sense. The Galerkin method differs from other MWR methods in the choice of weighing function. With the Galerkin method the test function \(v\) has the same form as the trial function \(\bar{u}\) in that it uses the same interpolation functions \(\phi_i\). In other words

\[
v(\bar{x}) = \sum_{i=1}^{N} w_i \phi_i(\bar{x}) \tag{2.42}
\]

Where \(w_i\) are the coefficients that weigh the interpolation functions \(\phi_i\). The product rule for vectors is:

\[
\nabla \cdot (f A) = f (\nabla \cdot A) + \nabla f \cdot A \tag{2.43}
\]

In terms of the variables in equation (2.41), the product rule for vectors is

\[
[\nabla \cdot (v \sigma \nabla \bar{u})] = \sigma \nabla \bar{u} \cdot \nabla v + v \nabla \cdot (\sigma \nabla \bar{u}) \tag{2.44}
\]

Substituting equation (2.44) into equation (2.41) which yields

\[
\int_\Omega [\nabla \cdot (v \sigma \nabla \bar{u}) - \sigma \nabla \bar{u} \cdot \nabla v] d\Omega = 0 \tag{2.45}
\]

Rearranging the above equation gives

\[
\int_\Omega \nabla \cdot (v \sigma \nabla \bar{u}) d\Omega = \int_\Omega \sigma \nabla \bar{u} \cdot \nabla v d\Omega \tag{2.46}
\]
Applying Gauss’ Theorem on equation (2.46) allows the introduction of boundary conditions

\[
\int_{\Omega} \nu \sigma \nabla \vec{u} \cdot d\Gamma = \int_{\Omega} \sigma \nabla \vec{u} \cdot \nabla v \, d\Omega \quad (2.47)
\]

Note that \( \nabla \phi \cdot \vec{n} = \frac{\partial \phi}{\partial \vec{n}} \) so equation (2.44) could be written as

\[
\int_{\Omega} \sigma \nabla \vec{u} \cdot \nabla v \, d\Omega = \int_{\Omega} \nu \sigma \frac{\partial \phi}{\partial \vec{n}} \, d\Gamma \quad (2.48)
\]

The left side of the equation (2.48) is for the entire mesh. When examined for a single triangular 2D element, \( k \), the left side is

\[
\int_{E_k} \sigma_k \nabla \vec{u} \cdot \nabla v \, d\Omega \quad (2.49)
\]

Substituting the definitions of the interpolating versions of \( \nu \) and \( u \) in equation (2.49) which yields

\[
\int_{E_k} \sigma_k \nabla \sum_{i=1}^{3} u_i \phi_i \cdot \nabla \sum_{j=1}^{3} w_j \vec{u}_j \, d\Omega \quad (2.50)
\]

If the conductivity \( (\sigma_k) \) is constant for a single element then the nodal voltage \( (u_i) \), the coefficients for the weighing functions, and the summations can be moved outside of the integral:

\[
\sigma_k \sum_{i=1}^{3} u_i \sum_{j=1}^{3} w_j \int_{E_k} \nabla \phi_i \cdot \nabla \vec{u}_j \, d\Omega \quad (2.51)
\]
And the equation (2.48) is rewrite as

$$\sigma_k \sum_{i=1}^{3} u_i \sum_{j=1}^{3} w_j S^k_{ik}$$

Where $S^k_{ij} = \int_{E_k} \nabla \phi_i \cdot \nabla \tilde{u}_j \, d\Omega$. So each element of the mesh produces a 3 by 3 matrix. The right side of the equation (2.48) is represents the boundary conditions. As per the interpolation functions, the boundary conditions are:

$$\int_{\Omega} v \sigma \frac{\partial \phi}{\partial n} \, d\Gamma = \sigma_k \sum_{i=1}^{3} w_i \sum_{j=1}^{3} u_i \int_{E_k} \phi_i \nabla \phi_j \cdot \tilde{u}_j \, d\Gamma$$

(2.52)

For a single element of the FEM equation (2.44) becomes

$$\sigma_k \sum_{i=1}^{3} u_i \sum_{j=1}^{3} w_j S^k_{ik} = \sigma_k \sum_{i=1}^{3} w_i \sum_{j=1}^{3} u_i \int_{E_k} \phi_i \nabla \phi_j \cdot \tilde{u}_j \, d\Gamma$$

(2.53)

Both sides can be divided by the summation of the weighting function coefficients to yield:

$$\sigma_k \sum_{i=1}^{3} u_i \sum_{j=1}^{3} S^k_{ik} = \sigma_k \sum_{i=1}^{3} \sum_{j=1}^{3} u_i \int_{E_k} \phi_i \nabla \phi_j \cdot \tilde{u}_j \, d\Gamma$$

(2.54)

In terms of the entire domain, the left hand side of the equation (2.47) will be

$$\int_{\Omega} \sigma \nabla \tilde{u} \cdot \nabla \phi \, dV = \sum_{E=1}^{K} \sigma_k \sum_{i=1}^{3} \sum_{j=1}^{3} u_i \int_{E_k} S^k_{ij} \, i, j = 1, ..., N$$

(2.55)

The potential within the element can be found from the linear interpolation equations as
\[ \nabla U = \sum_{i=1}^{3} U_i \nabla \phi_i \]  \hspace{1cm} (2.56)

Taking the divergence of equation (2.56) yields

\[ \nabla \cdot \nabla U = \sum_{i=1}^{3} \sum_{j=1}^{3} U_i \int \nabla \phi_i \cdot \nabla \phi_j \ d\Gamma U_j \]

\[ \nabla \cdot \nabla U = S^{(e)} U \]

Where \( S_{ij}^{(e)} = \int \nabla \phi_i \cdot \nabla \phi_j \ d\Gamma \) is the local stiffness matrix.

The gradient of the linear interpolation functions becomes the vector

\[ \nabla \phi_i = \frac{1}{2A} (y_2 - y_3, x_3 - x_2) \]  \hspace{1cm} (2.57)

which, the local matrix for an element, is the same as the matrices derived from the direct method of equation (2.34).

2.2.3 Experimental Results

The Electrical Impedance Tomography and Diffuse Optical Tomography Reconstruction Software are free software algorithms for forward and inverse modelling for Electrical Impedance Tomography in medical applications.

The forward problem calculates the voltages on the electrodes by using the injected current and assumed resistivity distribution. Finite Element Method is used to solve the forward problem which creates a 2D model using the given injected current values.
The inverse problem reconstructs the resistivity distribution by using the voltage measurements on the electrodes. Reconstruction algorithms are used to solve the inverse problem. Gauss-Newton reconstruction algorithm is implemented to solve the inverse problem.

Normal and Abnormal EIT lung images are reconstructed using the GN algorithm which is shown in fig.

Figure 2.3 Reconstructed EIT Lung Image 1 using GN Algorithm
Figure 2.4 Reconstructed EIT Lung Image 2 using GN Algorithm

Figure 2.5 Reconstructed EIT Lung Image 3 using GN Algorithm
Figure 2.6 Reconstructed EIT Lung Image 4 using GN Algorithm
2.3 BACK PROJECTION METHOD

The principle is similar to the back-projection reconstruction used in CT. For EIT, each measured voltage is assumed to be proportional to the impedance between the driven and the measuring electrode pairs. Equipotential back projection method (Barber & Brown 1986) is back-projects the impedance change recorded between two electrodes onto the reconstruction model (normally a 2D circular image), along the region defined by the equipotential lines ending on those electrodes shows in Figure 2.3. The back-projection method intrinsically introduces blur effect so that normally a filtering process is applied to compensate high frequency information afterward.

![Figure 2.7 EIT image reconstruction principle by back-projection based equipotential region](image)

**Figure 2.7** EIT image reconstruction principle by back-projection based equipotential region (Holder 2005) a) Forward projection of EIT. b) Back projection of EIT

2.3.1 Experimental Results for Back Projection

The normal and abnormal lung images are reconstructed using Back projection method. The reconstructed images are shown in Figure 2.8 and 2.9.
Figure 2.8 Reconstructed Lung Image 1 using Back Projection Method

Figure 2.9 Reconstructed Lung Image 2 using Back Projection Method
2.4 SUMMARY

A detailed reconstruction techniques implemented in this work is explained in this chapter. The different reconstruction algorithms like difference imaging and static imaging are explained. In this work, the difference imaging reconstruction algorithm along with Tikhonov regularization is used for EIT image reconstruction. This is done by using EIDORS software version 3.7.1. The back-projection reconstruction algorithm is also used to reconstruct the EIT images. The back-projection reconstruction algorithm is only employed to support the comparative analysis. The reconstructed EIT images are used as input for the further processing which is explained in the next chapter.