Anisotropy in Diffusion and Electrical Conductivity Distributions of TX-151 Phantoms

by

Neeta Ashok Kumar

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved November 2015 by the
Graduate Supervisory Committee:

Rosalind Sadleir, Chair
Vikram Kodibagkar
Jitendran Muthuswamy

ARIZONA STATE UNIVERSITY

December 2015

ABSTRACT

Among electrical properties of living tissues, the differentiation of tissues or organs provided by electrical conductivity is superior. The pathological condition of living tissues is inferred from the spatial distribution of conductivity. Magnetic Resonance Electrical Impedance Tomography (MREIT) is a relatively new non-invasive conductivity imaging technique. The majority of conductivity reconstruction algorithms are suitable for isotropic conductivity distributions. However, tissues such as cardiac muscle and white matter in the brain are highly anisotropic. Until recently, the conductivity distributions of anisotropic samples were solved using isotropic conductivity reconstruction algorithms. First and second spatial derivatives of conductivity ($\nabla\sigma$ and $\nabla^2\sigma$) are integrated to obtain the conductivity distribution. Existing algorithms estimate a scalar conductivity instead of a tensor in anisotropic samples.

Accurate determination of the spatial distribution of a conductivity tensor in an anisotropic sample necessitates the development of anisotropic conductivity tensor image reconstruction techniques. Therefore, experimental studies investigating the effect of $\nabla^2\sigma$ on degree of anisotropy is necessary. The purpose of the thesis is to compare the influence of $\nabla^2\sigma$ on the degree of anisotropy under two different orthogonal current injection pairs.

The anisotropic property of tissues such as white matter is investigated by constructing stable TX-151 gel layer phantoms with varying degrees of anisotropy. MREIT and Diffusion Magnetic Resonance Imaging (DWI) experiments were conducted to probe the conductivity and diffusion properties of phantoms. MREIT involved current injection synchronized to a spin-echo pulse sequence. Similarities and differences in the

divergence of the vector field of $\nabla\sigma$ ($\nabla^2\sigma$) among anisotropic samples subjected to two different current injection pairs were studied. DWI of anisotropic phantoms involved the application of diffusion-weighted magnetic field gradients with a spin-echo pulse sequence. Eigenvalues and eigenvectors of diffusion tensors were compared to characterize diffusion properties of anisotropic phantoms.

The orientation of current injection electrode pair and degree of anisotropy influence the spatial distribution of $\nabla^2\sigma$. Anisotropy in conductivity is preserved in $\nabla^2\sigma$ subjected to non-symmetric electric fields. Non-symmetry in electric field is observed in current injections parallel and perpendicular to the orientation of gel layers. The principal eigenvalue and eigenvector in the phantom with maximum anisotropy display diffusion anisotropy.

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The interaction of an electromagnetic field with an object depends on the shape and dielectric properties of the material composing the object. In particular, the complex relative permittivity influences the relative amounts of electromagnetic radiation reflected, absorbed or transmitted from the object. Dielectric properties of a medium such as relative permittivity and conductivity are obtained from the complex relative permittivity as:

Complex relative permittivity, $\hat{\varepsilon} = \varepsilon' - j\varepsilon''$ $\qquad(1)$

where ε' is the relative permittivity

ε'' is the out-of-phase loss factor ($\varepsilon'' = \dfrac{\sigma}{\varepsilon_0 \omega}$)

σ is the total conductivity

ε_0 is the permittivity of free space

ω is the angular frequency of the electromagnetic field

As biological molecules are polar, the complex relative permittivity is dependent on the frequency of applied alternating electromagnetic field. It follows that relative permittivity decreases and conductivity increases with increasing frequency. This behavior in biological tissues is shown in Figure 1. Some tissues such as muscle and white matter exhibit anisotropic conductivity at low frequency. However, a majority of techniques assume isotropic or equivalent isotropic conductivity distribution [1].
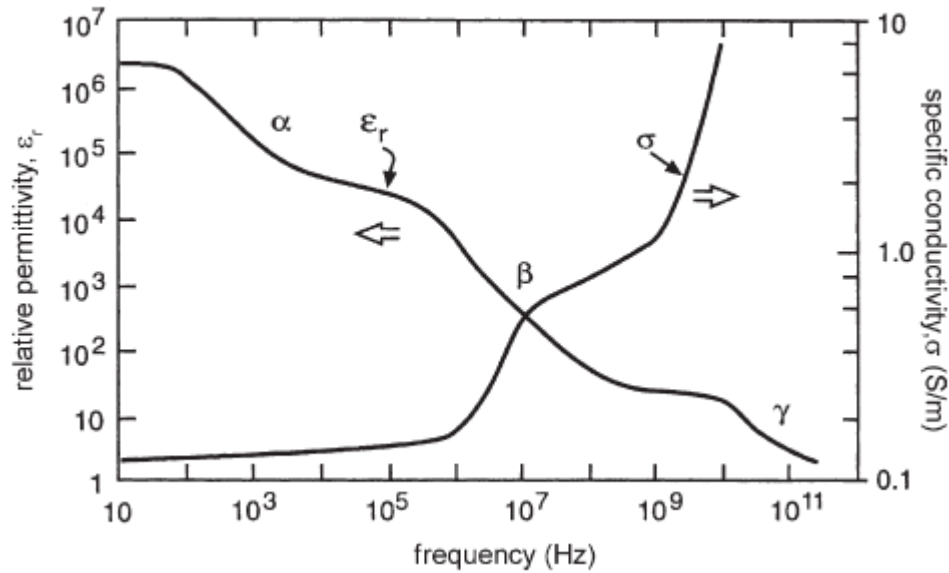
Figure 1: Frequency dependence of dielectric parameters (relative permittivity and conductivity) in biological tissues [2].

In biological tissues, electrical conductivity is highly dependent on the molecular composition, structure, concentration and mobility of ions, temperature, extra- and intra-cellular fluids and other factors. Conductivity is representative of the physiological and pathological state of a tissue and hence, provides useful diagnostic information [1]

1.1   PURPOSE

The purpose of the thesis is to identify incongruities in reconstructions of cross-sectional conductivity distributions of electrically anisotropic phantoms. Stable and reproducible (accurate) gel phantoms with varying degrees of anisotropy were designed for use as samples for imaging by Magnetic Resonance Electrical Impedance Tomography and Diffusion Tensor Magnetic Resonance Imaging (DT-MRI).   The presence of anisotropy in phantoms is demonstrated by Diffusion Tensor imaging and the

effect of the measurement scale on DTI is demonstrated by changing the resolution. The conductivity distributions of anisotropic phantoms were reconstructed using the Harmonic B_z algorithm, which assumes an isotropic conductivity distribution. Finite-element models of the phantoms were solved numerically to calculate synthetic B_z distributions. Conductivity distributions reconstructed using the Harmonic B_z algorithm from experimental and synthetic B_z were compared at different resolutions. Conductivity contrast reconstruction resulting from the isotropic assumption were compared in terms of the laplacian of conductivity distributions.

CHAPTER 2

BACKGROUND

2.1   Previous Work

2.1.1   Impedance imaging

The objective of Impedance Imaging is to map cross-sectional conductivity distributions inside an electrically conducting subject. The subject is electrically interrogated by injecting current through a pair of surface electrodes and recording resultant boundary voltages [3]. Internal current flow pathways establish internal current density, internal magnetic flux density and voltage distributions. Internal current flow depends on electrode configuration, conductivity distribution (σ) and geometry of the subject. Under the assumption of fixed boundary geometry and electrode configuration, the internal current density is dictated by the conductivity distribution to be imaged [1]. A local change in the conductivity alters the internal current pathway, which is manifested as a change in boundary voltage and internal magnetic flux density [4].

2.1.2   Electrical Impedance Tomography

Electrical Impedance Tomography (EIT) reconstructs conductivity images from measured boundary current-voltage data. However, spatial resolution and accuracy of the reconstructed conductivity distribution in EIT is poor due to the following reasons:

1. The relationship between internal conductivity distribution and boundary current-voltage data is highly non-linear. Additionally, boundary voltages are insensitive to local

4

changes in conductivity. Owing to this non-linearity and sensitivity, the reconstruction of conductivity images, based on boundary current-voltage measurement pairs, is complicated. This is formally described as, "The inverse problem of reconstructing the conductivity distribution is ill-posed in EIT".

2. The inverse problem is sensitive to the boundary geometry and electrode positions. This information is inaccurately modeled thereby affecting the reconstruction by EIT.

3. Current-voltage data is limited by a finite number of electrodes (usually 8 to 32) and the data is contaminated by measurement artifacts and noise.

Nevertheless, EIT is desirable in clinical applications for high temporal resolution and portability. As of today, EIT is useful to track changes in conductivity over time or frequency [1]. A number of different approaches were suggested to transform the inverse problem in EIT into a well-posed one. One such proposal suggested integrating the resultant magnetic and electric fields induced in an electrically conducting subject following current injection through surface electrodes. This idea sparked interest in the science community which was followed by extensive research on methods to measure the internal magnetic field and utilize this newfound information in conductivity image reconstruction [4].

2.1.3 Magnetic Resonance Current Density Imaging

An internal magnetic flux density $\mathbf{B}=(B_x, B_y, B_z)$, current density $\mathbf{J}=(J_x, J_y, J_z)$ and voltage distribution is developed when a current I is injected into an electrically conducting subject. A magnetic resonance imaging (MRI) scanner can measure the component of $\mathbf{B}$ parallel to the main magnetic field B_0. Assuming B_0 is in the z-direction, the scanner can measure B_z. The other two components of $\mathbf{B}$ are measured similarly following two object rotations. The internal current density $\mathbf{J}$ is calculated using Ampere's law. This technique, Magnetic Resonance Current Density Imaging (MRCDI), aims at non-invasively imaging and reconstruction of internal current density $\mathbf{J}$ from Ampere's law (equation 2).

Internal current density, $\qquad \boldsymbol{J} = \boldsymbol{\nabla} \times \boldsymbol{B}/\boldsymbol{\mu}_0$ $\qquad\qquad\qquad\qquad$ (2)

where μ_0 is the magnetic permeability of free space **[4]**

2.1.4 Magnetic Resonance Electrical Impedance Tomography Imaging

The basic concept of MREIT was proposed by Zhang (1992), Woo et al (1994) and Ider and Birgul (1998) by combining EIT and MRCDI. The key idea of MREIT emphasized the measurement of $\mathbf{B}$ using a current-injection MRI technique. Internal current density $\mathbf{J}$ images from magnetic flux density $\mathbf{B}$ were constructed by Ampere's law as in MRCDI. From $\mathbf{B}$ and/or $\mathbf{J}$, it is possible to understand the internal current pathways due to the conductivity distribution of the subject. In this way, Magnetic Resonance Electrical Impedance Tomography (MREIT) was pioneered to overcome the technical

difficulties in Electrical Impedance Tomography (EIT) and produce high-resolution conductivity images [4].

A serious problem in using equation 2 is the measurement of all three components of **B**. Currently available magnetic resonance scanners can only measure one component of **B** that is parallel to the main magnetic field (B_0). Despite this limitation, all three components of **B** can be measured by rotating the subject. Theoretically, this seems like a feasible solution. However, it is discouraged because it misaligns pixels and is impractical in a clinical setting [4]. Most recent MREIT techniques focus on investigating the relationship between the measured component of **B** and the current density or conductivity distribution to be imaged. Assuming B_0 is in the z-direction, Oh (2003) invented a new method to extract conductivity information from B_z known as the Harmonic-B_z algorithm. Numerous non-biological and biological phantoms, postmortem animal tissues, *invivo* animal and human experiments were conducted to validate and test the new algorithm [1]. Potential clinical applications of MREIT include Functional imaging, neuronal source localization and mapping, optimization of therapeutic treatments using electromagnetic energy.
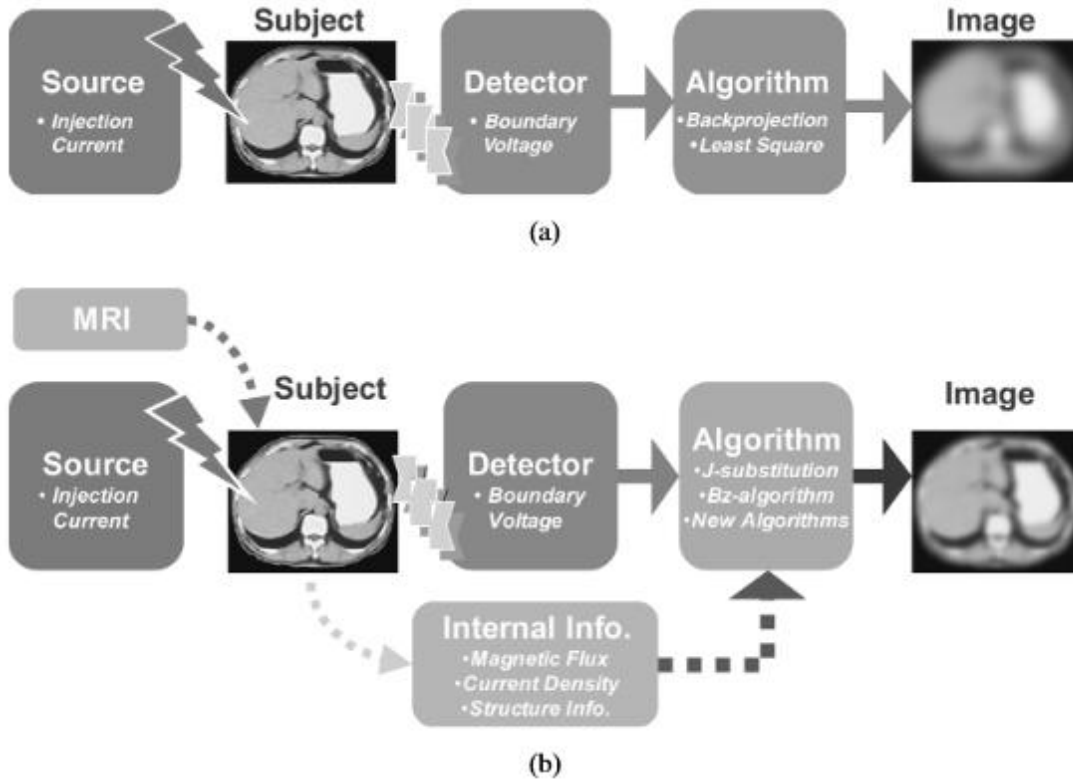
Figure 2: (a) EIT using boundary measurements (b) MREIT using both internal and boundary measurements [4]

Comparing MREIT with EIT

	MREIT	EIT
Advantages	Better spatial resolution and accuracy	High temporal resolution
	Information from MREIT can be used as apriori information in EIT reconstructions for better results.	Portability
Disadvantages	Long imaging time	Poor spatial resolution
	Lack of portability	Inaccurate
	Requirement of an expensive MR scanner	

Table 1: Comparison of the pros and cons of MREIT and EIT [4]

2.2　Theoretical considerations of MREIT

2.2.1　Influence of current on the phase of MR signals

The internal magnetic flux density induced during electrical interrogation is crucial in determining the spatial resolution and accuracy of reconstructed conductivity images in MREIT [4]. The current injected in MREIT experiments is in the form of pulses with wide pulse-width similar to LF (low frequency) - MRCDI [4]. A constant current source sequentially injects positive $I^{\pm}$ and negative $I^{\mp}$ currents through surface electrodes in synchrony with an MR pulse sequence. Injected current induces a magnetic flux density $\mathbf{B} = (B_x, B_y, B_z)$ causing inhomogeneity in $\mathbf{B_0}$ changing $\mathbf{B}$ to $(\mathbf{B} + \mathbf{B_0})$. This leads to phase accumulation proportional to the z-component of $\mathbf{B}$ i.e. B_z. Positive and negative currents with the same amplitude and width are injected sequentially to cancel out any systematic phase artifact of the MRI scanner and to increase the phase change by a factor of 2 [1]. The MR spectrometer provides complex k-space data corresponding to positive $I^{\pm}$ and negative $I^{\mp}$ currents as:

$$S^{I^{\pm}} = \iint_{-\infty}^{\infty} M(x,y)\, e^{j\delta(x,y)}\, e^{j\gamma B_z(x,y)T_c}\, e^{j(xm\Delta k_x + yn\Delta k_y)} dx\, dy \tag{3}$$

$$S^{I^{\mp}} = \iint_{-\infty}^{\infty} M(x,y)\, e^{j\delta(x,y)}\, e^{-j\gamma B_z(x,y)T_c}\, e^{j(xm\Delta k_x + yn\Delta k_y)} dx\, dy \tag{4}$$

where M is the MR magnitude image representing the transverse magnetization,

　　　δ is any systematic phase error,

　　　$\gamma = 26.75 \times 10^7$ rad T^{-1} s^{-1} is the gyromagnetic ratio of hydrogen

　　　T_c is the pulse width of the current in seconds.

Two-dimensional discrete Fourier transformations of $S^{I\pm}$ and $S^{I\mp}$ result in complex images $M_c^{\pm}(x,y)$ and $M_c^{\mp}(x,y)$ respectively as shown:

$$M_C^{\pm} = M(x,y)e^{j\delta(x,y)}e^{\pm j\gamma B_z(x,y)T_c} \tag{5}$$

Incremental phase change is calculated by dividing the imaginary part of two complex images as:

$$Arg\left(\frac{M_c^{\pm}(x,y)}{M_c^{\mp}(x,y)}\right) = Arg\left(e^{j2\gamma B_z(x,y)T_c}\right) = 2\gamma B_z(x,y)T_c = \varphi_z(x,y) \tag{6}$$

where Arg(w) denotes the argument of a complex number w.

The phase change φ_z is wrapped in $-\pi < \varphi_z < \pi$, and must be unwrapped using a phase unwrapping algorithm such as Goldstein's branch cut algorithm.

2.2.2 Phase Unwrapping

Goldstein's branch cut algorithm is based on detecting inconsistencies when summing wrapped phase gradients around every 2 x 2-sample path. The summation yields non-zero results at inconsistencies and are known as residues. Residues of opposite polarities (i.e. signs) are balanced by connection with branch cuts. The cuts are generated by a method to minimize the sum of cut lengths.

A search of size 3 is placed around a residue and searched for another within the box. If a residue of opposite polarity is found, a branch cut is placed between them and labeled "uncharged". The search for another residue continues within the box. If a residue of same polarity was found, the box is moved to a new residue until an opposite charged

residue is found or no residues can be found within the boxes. If no residues are found, the size of the box is increased by 2 and the algorithm repeats from the present starting residue.

2.2.3 Reconstruction of conductivity distribution

By sequentially injecting positive and negative currents, the systematic phase artifact δ is rejected and the phase change is doubled. B_z is related to unwrapped phase $\psi(x, y)$ by a scaling factor and can be computed by:

$$B_z(x, y) = \frac{\psi(x,y)}{2\gamma T_c} = \frac{1}{2\gamma T_c} \arg\left(\frac{M^+(x,y)}{M^-(x,y)}\right) \qquad (7)$$

Multi-slice magnetic resonance magnitude and phase images are reconstructed from *k*-space data. Magnitude images provide boundary geometry and electrode positions whereas phase images provide B_z data.

The spatial resolution of a reconstructed conductivity image is limited by the noise measured in B_z data. The standard deviation of noise in B_z, s_{B_z}, is related to the signal-to-noise ratio (SNR) of the magnitude image, γ_M and total current injection time T_c as:

$$s_{B_z} = \frac{1}{\sqrt{2}\,\gamma\,T_c\,\gamma_M} \qquad (8)$$

Incremental phase change (in equation 11) is the raw data in MREIT. This phase change is proportional to the product B_z and T_c. Since B_z is proportional to I, the incremental phase change can be increased by optimizing MREIT pulse sequences to

11

maximize the product of I and T_c. B_{z_1} and B_{z_2} due to positive $I^{\pm}$ and negative $I^{\mp}$ current injections were calculated as in equation 8. From the z-component of the curl of the Ampere's law $\nabla \times J = \nabla \times \nabla \times B / \mu_0$, the following relationship is solved for the conductivity:

$$\begin{bmatrix} \frac{\partial u_1}{\partial y}(\mathbf{r}) & -\frac{\partial u_1}{\partial x}(\mathbf{r}) \\ \frac{\partial u_2}{\partial y}(\mathbf{r}) & -\frac{\partial u_2}{\partial x}(\mathbf{r}) \end{bmatrix} \begin{bmatrix} \frac{\partial \sigma}{\partial x}(\mathbf{r}) \\ \frac{\partial \sigma}{\partial y}(\mathbf{r}) \end{bmatrix} = \frac{1}{\mu_0} \begin{bmatrix} \nabla^2 B_{z,1}(\mathbf{r}) \\ \nabla^2 B_{z,2}(\mathbf{r}) \end{bmatrix}, \qquad \mathbf{r} \in \Omega,$$

Figure 3: Inverse relationship between electric field, gradients of conductivity and laplacian of B_z.

where u_1 and u_2 are voltages satisfying boundary-value condition due to $I^{\pm}$ and $I^{\mp}$. This is iteratively solved in CoReHA software package which implements the Harmonic B_z algorithm [5].

2.2.3.1   Image Contrast

Image contrast is an important parameter to overcome the disability of the human visual system to detect differences in absolute illuminance values. It is defined as differences in image intensity. Contrast depends on a multitude of factors such as spin density, relaxation times and diffusion coefficients. This dependence is greatly influenced by the data acquisition protocol [10]. In this experiment, data acquisition parameters were chosen as described in Table 2 to enhance the T_1 effect. Generally, enhancing the effect of either the spin density, T1 or T2 on image contrast is achieved by relatively varying values of T_R and $T_{E.}$ as shown in Table 2. The resultant image is said to carry a T_1 contrast because the image contrast is exponentially dependent on the T_1 relaxation time of the sample. MR imaging of normal soft tissues have significantly different T_1 values

12

thereby making it effective for good anatomical definition. Practically, TE and TR are limited by system hardware performance and imaging time respectively[10].

Contrast	T_E	T_R
T1 - weighting	Short	Appropriate
T2 - weighting	Appropriate	Long
ρ-weighting	Short	Long

Table 2: Influence of echo and relaxation time (T_E, T_R) on image contrast.

2.2.4. Forward Problem

A forward solver is extremely useful for algorithm development, experimental design and verification. Image reconstruction in MREIT is inherently 3D, and therefore a 3D forward solver is implemented. This model provides distributions of current density **J**, and voltage V within an electrically conducting domain (i.e. subject) following current injection using recessed electrodes.

Consider Ω as an electrically conducting domain with isotropic conductivity distribution σ and boundary $\partial\Omega$. Let $\mathcal{C}$, $\mathcal{E}$ and $\mathcal{L}$ represent the area covered by plastic containers ($\mathcal{C}_1$ and $\mathcal{C}_2$), electrodes ($\mathcal{E}_1$ and $\mathcal{E}_2$) and lead wires ($\mathcal{L}_1$ and $\mathcal{L}_2$) respectively. Electrodes $\mathcal{E}$ are recessed from the surface of the object $\partial\Omega$ by plastic containers $\mathcal{C}$.

Artifacts in Magnetic Resonance images occur due to the RF shielding effect of conductive electrodes. To move these artifacts out of the domain Ω, recessed electrodes are preferred. Figure 4(b) displays the recessed electrode assembly. Use of recessed electrodes ensures artifact-free MR images of the domain, including its boundary.
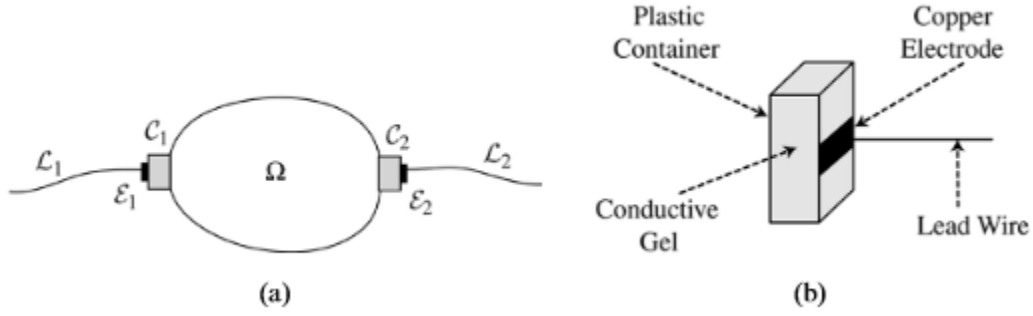
Figure 4: (a) Definition of domains and (b) recessed electrode assembly

To formulate the problem, consider D as the region comprising of the domain and two plastic containers i.e. $D = \Omega \cup C$. Assume a low-frequency current I injection through $\mathcal{E}_1 \ and \ \mathcal{E}_2$ attached on ∂D, then the induced voltage V satisfies the following boundary value problem with the Neumann boundary condition [4]:

$$\begin{cases} \nabla. \left[\sigma(r) \, \nabla V(r) \right] = 0 \quad in \ D \\ -\sigma \nabla V. \boldsymbol{n} = g \quad\quad\quad on \ \partial D \end{cases} \tag{10}$$

where **n** is the outward unit normal vector on ∂D

g is a normal component of the current density on ∂D due to I

r is a position vector in R^3.

g is zero on the portions of the boundary not in contact with the electrodes and $\int g \, ds = \pm I$ over $\mathcal{E}_j$ for j=1 or 2. To arrive at a unique solution for V in equation 10, a reference voltage $V(\mathbf{r_0}) = 0$ for $\mathbf{r_0} \in \partial D$ is chosen. Having computed the voltage distribution V, the current density **J** is given by:

$$J(r) = -\sigma(r)\nabla V(r) = \sigma(r) \, E(r) \tag{11}$$

where $E = -\nabla V$ is the electric field intensity.

14

Considering the magnetic field produced by I, the induced magnetic flux density **B** in Ω is :

$$\boldsymbol{B(r)} = \boldsymbol{B_\Omega(r)} + \boldsymbol{B_{\mathcal{C}}(r)} + \boldsymbol{B_{\mathcal{E}}(r)} + \boldsymbol{B_{\mathcal{L}}(r)}$$

where $\boldsymbol{B_\Omega}, \boldsymbol{B_{\mathcal{C}}}, \boldsymbol{B_{\mathcal{E}}}$ and $\boldsymbol{B_{\mathcal{L}}}$ are magnetic flux densities due to **J** in $\Omega, \mathcal{C}, \mathcal{E}$ and I in $\mathcal{L}$ respectively.

From the Biot-Savart law,

$$\boldsymbol{B_\Omega(r)} = \frac{\mu_0}{4\pi} \int \boldsymbol{J(r')} \times \frac{r-r'}{|r-r'|^3} \quad \text{over } \Omega \tag{12}$$

The effects of recessed electrodes and lead wires ($\boldsymbol{B_{\mathcal{C}}}, \boldsymbol{B_{\mathcal{E}}}$ and $\boldsymbol{B_{\mathcal{L}}}$) are removed based on equation (13)

$$\nabla^2 (\boldsymbol{B_{\mathcal{C}}(r)} + \boldsymbol{B_{\mathcal{E}}(r)} + \boldsymbol{B_{\mathcal{L}}(r)}) = 0 \; for \; \boldsymbol{r} \in \Omega \tag{13}$$

since $\nabla^2 \left(\frac{1}{|r-r'|} \right) = 0$ when $\mathbf{r} \neq \mathbf{r'}$.

From Ampere's law,

$$\boldsymbol{J(r)} = \nabla \times \boldsymbol{B(r)} / \mu_0 \quad \text{in } \Omega \tag{14}$$

where μ_0 is the magnetic permeability of free space

Since current is injected externally, there is no internal source or sink. This implies $\nabla . J = 0$. Equating the expressions for **J(r)** :

$$\nabla . \boldsymbol{J(r)} = \frac{1}{\mu_0} \nabla . \nabla \times \boldsymbol{B(r)} = 0 \quad \text{in } \Omega \tag{15}$$

The condition (Equation 15) was suggested to check compatibility conditions to validate numerical solutions. However, validation was performed with experimental results in this research.

The next step includes reconstruction of an image of σ ($\sigma = \frac{1}{\rho}$) in Ω from measured **B** or B_z in Ω and V on $\partial\Omega$ for a given injection current I and electrode configuration. Two orthogonal injection currents are applied for the uniqueness of the reconstructed image [4].
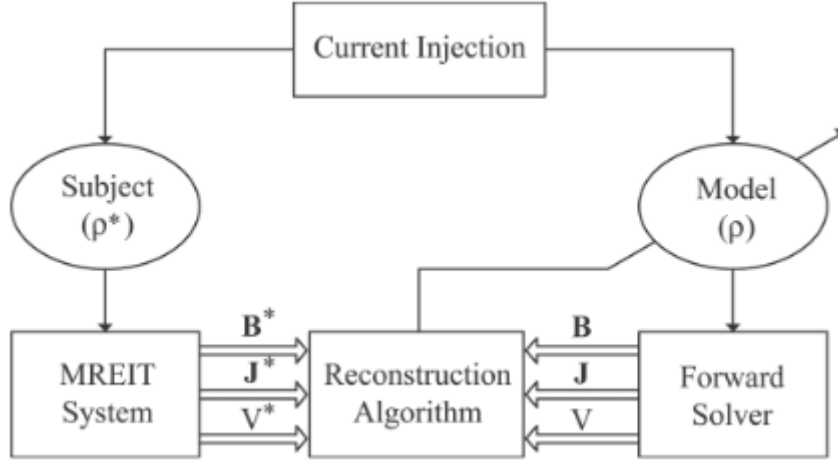
Figure 5: Forward and inverse problems in MREIT

The Finite Element Method (FEM) is used to numerically solve for V in equation 10. A 3D model of D and $\mathcal{E}$ is constructed and the thickness of each electrode is assumed to be negligible. The model is discretized into a finite element mesh and the numerical solution of V is a set of nodal voltages of the corresponding finite element mesh. The current density **J** is computed using Equation 11.

2.2.5   Inverse Problem

The inverse problem of MREIT is handled by utilizing either all three components of **J/B** or only B_z :

1. **J**-based MREIT

The imaging object is rotated twice in the magnetic resonance imaging (MRI) scanner to collect data of all three components **J/B** using Equation 11. Then, the conductivity is calculated using the voltage distribution for current injections in **J**-based MREIT conductivity reconstruction algorithms. Not much experimental work is available because rotating the object causes misalignment of pixels.

2. B_z-based MREIT

This class of reconstruction algorithms provides a practical alternative to conductivity reconstruction utilizing the information in one component of **B** i.e. B_z. Multiple injection currents are used and its corresponding B_z is recorded. This data along with at least one voltage measurement is used to reconstruct the absolute values of σ. In absence of voltage information, conductivity contrast images are reconstructed [1].

2.2.5.1 Harmonic - Bz algorithm

Under the assumption that the resistivity of a subject does not change much in the z-direction in a thin imaging slice, an approximately transversal internal current density *J* i.e. $(J_x, J_y, 0)$ can be developed using longitudinal electrodes. The internal magnetic flux density **B** is due to the internal current density **J** and external current *I* through lead wires, i.e. $\mathbf{B} = \mathbf{B^J} + \mathbf{B^I}$. Using an MR scanner with main-magnetic field in z-direction, the z-component B_z of **B** is measured. B_z changes along the z-direction in the imaging slice, even if **J** is independent of *z* in the imaging slice. Since lead wires are out of the sample, $\nabla^2 B_z^{\ I} = 0$. The relationship in Figure 3 is solved by the steps detailed in Appendix E.

17

2.3   Diffusion Tensor Imaging

Diffusion is a mass transport process resulting in molecular or particle mixing without requiring bulk motion. Fick's law explains this phenomenon through the relationship:

$$J = -D \, \nabla C \qquad (16)$$

where J is the net particle flux

C is the particle concentration

D is the Diffusion coefficient

This equation describes diffusion as the flow of particles from high to low concentration. The rate of diffusion is proportional to the concentration gradient and the diffusion coefficient. Diffusion coefficient is an intrinsic property of the medium and depend on the size of diffusion molecules, temperature and microstructural features of the environment. Dependence of D on the microstructural environment is advantageous in studying the properties of biological tissues. Diffusion is greatly influenced by the geometrical structure of the environment.

Diffusion characteristics are quantified by magnetic resonance imaging. This is achieved by applying a diffusion gradient during a standard spin-echo MR imaging pulse sequence as shown in Figure 6. The gradient is bipolar which is a positive lobe followed by a negative. A positive phase shift proportional to the position of a spin is added during the first gradient lobe. Similarly, a negative phase shift is added during the second

gradient lobe. Spins at different locations in the subject acquire different phase shifts depending on their location. The net phase shift acquired during the echo is a reflection of the motional history of the particles in the sample. Stationary particles accumulate no net phase because the gain and loss of phase is equal.
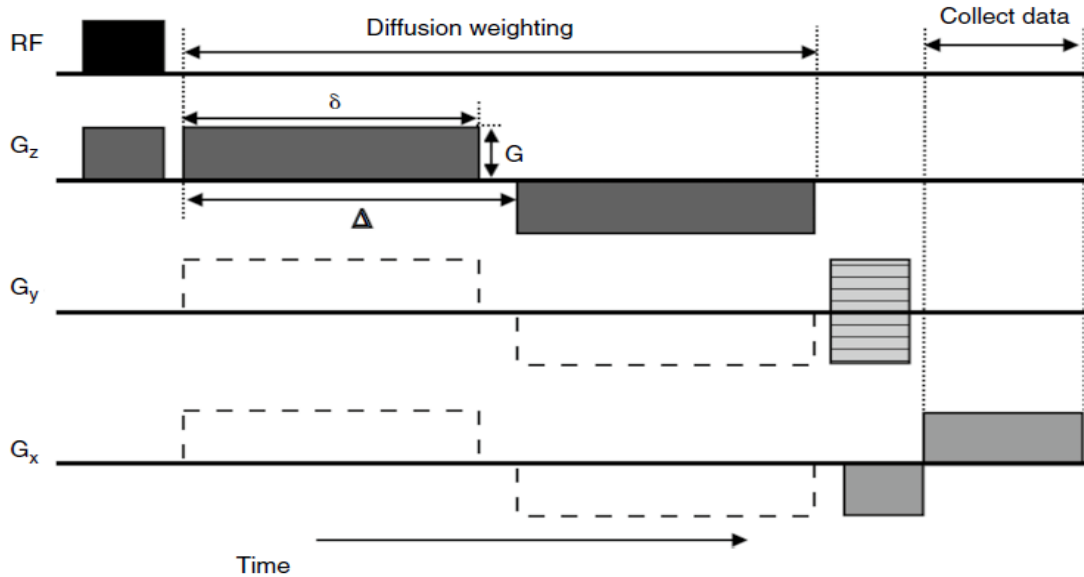
Figure 6: Simple MR pulse sequence with diffusion weighting added in one direction.

A diffusion tensor **D** is a 3 x 3 symmetric matrix of displacements in 3D useful to characterize unequal displacements per unit time in all directions.

$$\mathbf{D} = \begin{bmatrix} D_{xx} & D_{xy} & D_{xz} \\ D_{xy} & D_{yy} & D_{yz} \\ D_{xz} & D_{yz} & D_{zz} \end{bmatrix}.$$

Figure 7: Inverse relationship between electric field, gradients of conductivity and laplacian of B_z.

19

The diagonal elements of **D** correspond to diffusivities along the three orthogonal axes (i.e. Scanner frame). Off-diagonal elements correspond to correlation between displacements along those orthogonal axes. When off-diagonal elements are zero i.e. the tensor is aligned with the principal axes of the measurement frame, then the diagonal elements correspond to the eigenvalues $(\lambda_1, \lambda_2, \lambda_3)$ of **D**. The orientation of the principal axes of **D** is given by eigenvectors $(\varepsilon_1, \varepsilon_2, \varepsilon_3)$ which are mutually orthogonal. The tensor is oriented parallel to the direction of the principal eigenvector (ε_1). The principal eigenvector is recognized as the eigenvector associated with the largest eigenvalue (λ_1). The principal eigenvector is assumed to be co-linear with the dominant fiber orientation within the voxel [6].
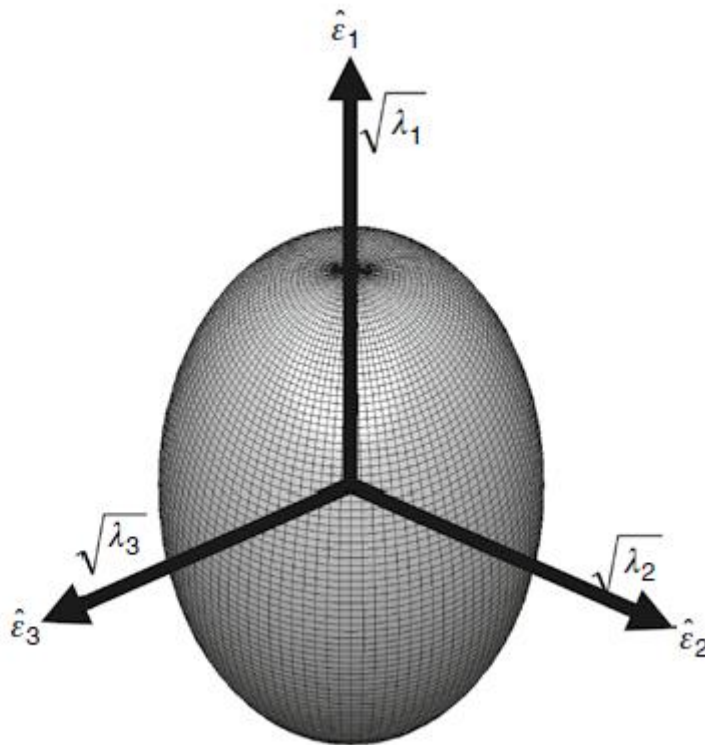
Figure 8: Schematic of the diffusion tensor ellipsoid. A spin placed at the center of the ellipsoid will diffuse with equal probability throughout the envelope.

CHAPTER 3

MATERIALS AND METHODS

3.1   Anisotropic phantom design

Novel algorithms were recently developed to reconstruct conductivity tensors in anisotropic phantoms. To validate these anisotropic reconstruction algorithms, it is imperative to develop anisotropic phantoms with a stable and reproducible composition. A criteria to develop a homogeneously anisotropic conductivity element was observed when alternating high and low isotropic conductivity layers were arranged at greater than 10 times the spatial frequency compared to the measurement scale [7].

The degree of anisotropy, also known as the anisotropy ratio (k), is defined as the ratio of longitudinal to transverse conductivity. This measure can be controlled by continuously varying the relative conductivities of the layers. The anisotropy ratio, k, depends on the total thickness of each isotropic material and is not affected by the number or arrangement of layers. The maximum value of k, is observed when the total thicknesses of the two layers are the same i.e. $\alpha_t = t_2/t_1 = 1$. Then, the maximum value of k, k_{max} depends on: $k_{max} = (\sigma_2 + \sigma_1)^2/4\sigma_2\sigma_1$

$$k_{max} = (\sigma_2 + \sigma_1)^2/4\sigma_2\sigma_1 \tag{17}$$

In the phantom composed of gel slices, the longitudinal direction was parallel to slice planes and transverse direction was orthogonal to the planes. A polysaccharide material, TX151 (The Oil Research Center, LA,USA), when mixed with water formed a

tissue equivalent gel that maintained integrity during heating. The consistency of the gel was similar to rubber after being heated. The gel was then shaped by pouring into molds and refrigerated. The conductivity and permittivity of gels were controlled by the amount of Sodium Chloride and Sucrose. The gelling time of the mixture was controlled by the temperature of the mixture and the ratio of TX-151 to water. Lower temperatures of water and reduced amounts of TX-151 lowered the gelling time. Two batches of TX-151 gels were prepared to make low and high conductivity isotropic gels respectively. These batches were sliced into layers of equal thickness and placed in alternating low and high isotropic conductivity layers.

3.1.1 Composition of gels

Structures with 1 (42.6 mm), 3 (14.2 mm), 27 (1.57 mm) and 47 (0.91 mm) layers were constructed by alternating layers of high and low conductivity gel slices. In all these cases, high conductivity layers were placed near the electrodes. The conductivity contrast σ_2/σ_1 was 6.85 with σ_2 as 1.37 S/m and σ_1 as 0.2 S/m in layered phantoms. The behavior of the layered phantom approached that of a purely anisotropic structure when ten or more alternating conductivity layers were used.

Ingredients	Purpose	High conductivity gel (1.37 S/m measured at 1kHz on HP 4192A over 4 hours)	Low conductivity gel (0.2 S/m measured at 1 kHz on HP 4192A over 4 hours)
Water	Sets electric conductivity	692 ml	692 ml
Sucrose	Sets electric permittivity	84 g	84 g

22

Agar	Solidifier	40 g	40 g
TX-151	Thickener	15 g	15 g
Copper sulphate	Reduces T_1	0.692 g	0.692 g
Sodium Chloride	Principal ingredient	5 g	0 g

Table 3: Recipe for high and low conductivity gels
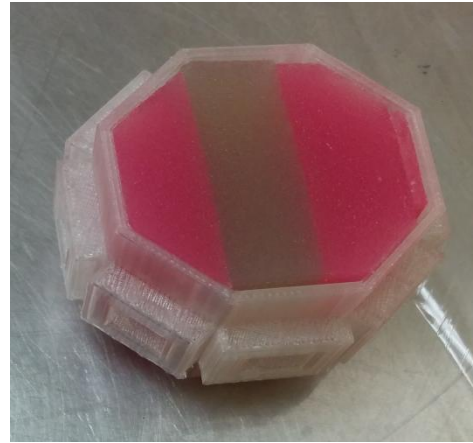
(a)   (b)
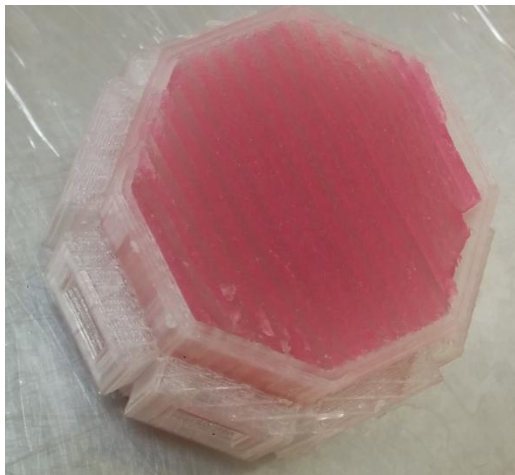
(c)   (d)

Figure 9: TX-151 gel phantoms with (a) 1 layer (b) 3 layers (c) 27 layers (d) 47 layers in custom identical sample chambers used as imaging sample in MREIT experiments.

3.2   Sample chamber and miter box design

Two pairs of orthogonal currents were injected to produce non-parallel current densities throughout the sample necessary for unique cross-sectional conductivity image

reconstruction in MREIT [8]. Care was taken to ensure current through the sample mostly resided in the XY and minimized current flow in the Z-direction. Hence, current density in the z-direction was negligibly small ($J_z = 0$). Carbon fiber electrodes were used to inject current through electrode gel at the electrode-phantom interface.

The octagonal sample chamber was designed in Solidworks (Dassault Systèmes SOLIDWORKS Corp.) with a wireframe model shown below (Will insert a Figure). Each side face contained a recessed port for current injection with dimension 10 mm x 10 mm x 5 mm. The overall size of the model was 52 mm x 52 mm x 42 mm. The design in Solidworks was exported as .STL and printed using a Makerbot Replicator 2.

To accommodate gel phantoms in the sample chamber, a miter box of dimensions identical to the cross-section of the sample chamber was designed in Solidworks. The miter box design was exported as .STL and printed by a Makerbot Replicator 2.The miter box was useful to shape gel phantom slices by sliding a cutter through the slits in the miter box.

3.3   Magnetic Resonance Imaging Experiments

3.3.1   MR Scanner

The experimental setup of MREIT includes an MRI scanner and a constant current source. Nonmagnetic conductive materials such as copper, silver and carbon ideally serve as electrodes. However, an artifact occurs at the interface of the electrode with the surface of the subject because it shields RF signals. To move this artifact out of

the region of interest, recessed carbon electrodes were used. These electrodes had a gap of conductive gel between the copper electrode and surface of the object. Recently, carbon-hydrogel electrodes with conductive adhesive is being used in *invivo* animal and human experiments [1].

The sample chamber enclosing the phantom was placed in a 70 mm bore and a birdcage RF coil was used in a 7 T MRI scanner (Bruker, BioSpec) at Barrow Neurological Institute. The main magnetic field B_0 is in the z-direction. A spin echo pulse sequence was used for imaging experiments. The imaging parameters are summarized in Table 4.

3.3.2 MREIT and DTI Imaging Parameters

Imaging Parameters	MREIT	DTI
Pulse sequence	Spin - echo	Spin - echo based DTI
TR/TE (ms)	1000/25	2094.305/210
Number of slices	11	5
Slice thickness (mm)	4	10.5
Spatial resolution (mm^2)	0.9375 x 0.9375	10.5 x 10.5
Matrix size	64 x 64	32 x 32
Field-of-view (mm^2)	60 x 60	336 x 336
NEX	2	1
Number of repetitions	1	1
Total scan time (s)	167	480
B - value	-	1000
NDiffdir (Number of diffusion directions)	-	6
NDiffExp	-	7
DwEffBval	-	7

Table 4: Imaging parameters in MREIT and DTI experiments.

3.3.3 MREIT current source

The presence of two non-parallel current densities within a conductive region has been previously shown as sufficient to recover the relative conductivity of an object. The magnetic field information due to current injection into a volume conductor is mapped onto the phase of an MRI acquisition. This mapping is in the form of a phase shift in the recorded MR signal

$$S_I(k_x(t), k_y(t), k_z(t)) = \int_{R^3} \rho(x, y, z) \exp[-i(xk_x + yk_y + zk_z + \gamma T_c B_z)] dxdydz$$

$$= \int_{R^3} \rho(x, y, z) \exp(-i\gamma T_c B_z) \exp[-i(xk_x + yk_y + zk_z)] dxdydz$$

Figure 10: MR signal recorded in k-space under current injection of duration T_c.

where T_c is the duration of the current pulse and B_z is the z-component of the current - induced magnetic field (B_0 is in the z-direction) [10].

An MREIT data acquisition system requires an MR scanner, surface electrodes and a constant current source. Current is injected in the form of rectangular pulses synchronized with a spin echo MR pulse sequence. Earlier studies utilized a current source placed outside the shield room. However, the cables form the current source to the electrodes in the MR scanner caused numerous artifacts and noise, thereby necessitating the development of a current source to overcome these issues [9].

MREIT experiments require injection currents to be synchronized with the RF pulse of the MR system. Such synchronization is achieved by connecting the MR

spectrometer, which provides trigger signals, to the current source A new MREIT current source was developed making it possible to place it in the shield room. The new current source was connected via an optical link to the MR spectrometer for trigger signals and a separate optical link to a PC for programming current injection sequences (Appendix G). Noise elimination in the new current source improved the SNR in MREIT images by 38% [9].
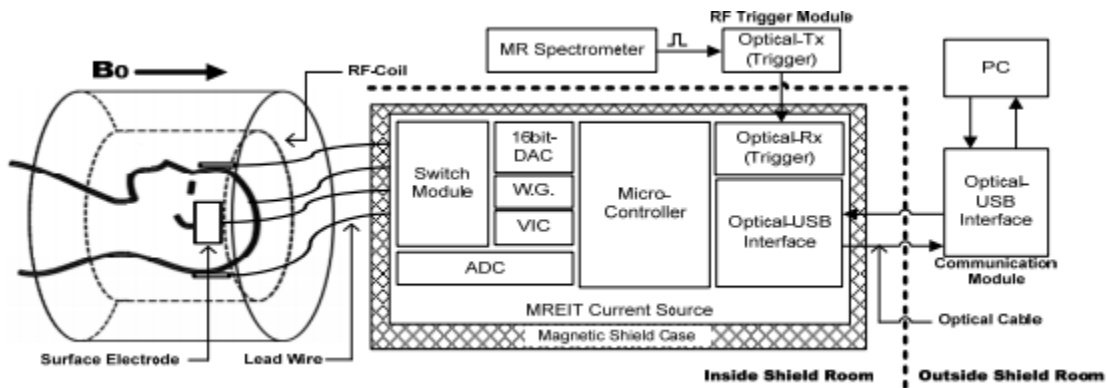
Figure 11: Structure of the new MREIT current source [9]

Current source parameters	
Current Injection	10 mA
Voltage	18/21.8 V
Resistance	3.6/4.36 Ω
T_C	16ms

Table 5: Current source parameters during MREIT experiments.

3.3.4 MREIT Pulse sequence

The spin echo pulse sequence is robust to many perturbations in phase images and so, has been widely used in MREIT experiments. Current injection is synchronized with the MR pulse sequence to generate inhomogeneity in the main magnetic field (B_0).

This is presented as a phase change with the alteration being proportional to the z-component of the magnetic field (B_z) induced by the current [10].
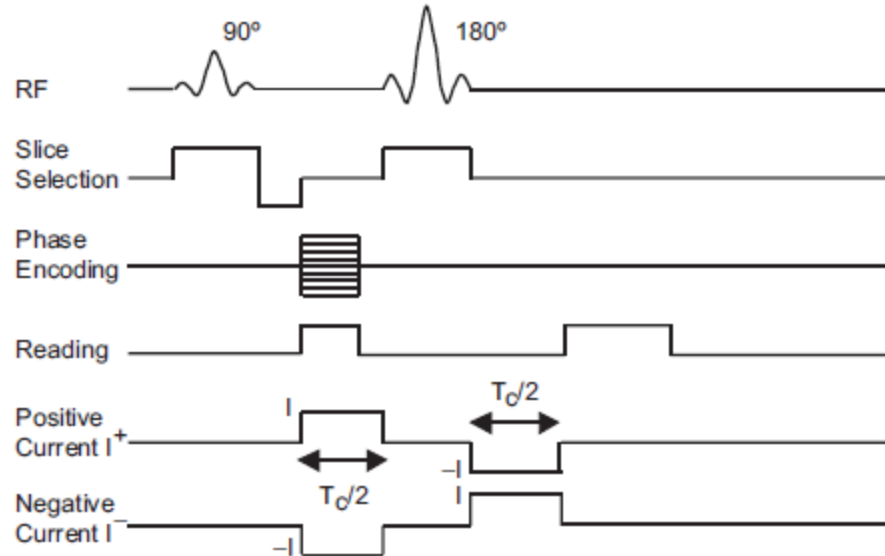
Figure 12: Standard Spin echo pulse sequence for MREIT [10]

3.4   Impedance Analyzer

Impedance is a property of any circuit made from resistors, capacitors and inductors. It is dependent on frequency and is represented as a complex number with real and imaginary parts. An Impedance Analyzer is used to determine and verify the impedance of the gel phantom (sample) between electrical ports of the sample chamber. The sliced gel phantom with alternating high and low conductivity was arranged in a 5 cm x 5 cm x 5cm rectangular box. The insides of a pair of opposite surfaces was covered with copper tape. Electrodes were placed on the outsides of the same surfaces. Current was delivered via connectors and voltage recorded from the copper tape by the

impedance analyzer. and connecting electrodes across. The gel slices were placed in a parallel combination thereby reducing the equivalent impedance.

The conductivity is estimated as:

$$R = \frac{\rho l}{A}$$

where R = Resistance

ρ = specific resistivity (Conductivity, $\sigma = \frac{1}{\rho}$)

l = length of gel layer arrangement (distance between electrodes)

A = area of cross-section the box

With l = 5 cm , A = 25 cm^2,   Conductivity, $\sigma = \frac{0.2}{|Z|}$ S/cm

HP4192A LF Impedance Analyzer was useful in measuring impedance parameters such as Absolute value of impedance (|Z|), Absolute value of admittance (|Y|), Phase angle (theta), Resistance (R), Reactance (X), Conductance (G) and Susceptance (B). The warm up of the equipment for 30 minutes was followed by setting the spot frequency at 1000 Hz [11]. The impedance analyzer was remotely controlled to measure the impedance of alternate gel layers within the rectangular box by graphical programming in LabVIEW (Appendix B).
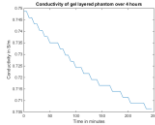
Figure 13: Conductivity of phantom with alternating high and low conductivity gel layers calculated from the impedance recorded by HP4192A.

3.5　　Finite Element Method

The Finite element method (FEM) is a mathematical method to solve complex ordinary and partial differential equations. In the FEM, a 3D domain is divided into a number of elements (example: tetrahedra, prisms, hexahedra) and the unknown potential -is represented as a polynomial of fixed order on each element. Each polynomial in the solution is represented by points known as nodes at which the FEM evaluates the solution. Finite elements intersect in whole faces, edges or at vertices, and the potential is assumed continuous across faces. Finite element method is the most used method to numerically solve linear and non-linear problems without restrictions on the geometry. The accuracy of finite element approximations to partial differential equations greatly depends on the smoothness of the analytical solution i.e. smoothness of the data [12].

30

3.5.1 COMSOL Multiphysics
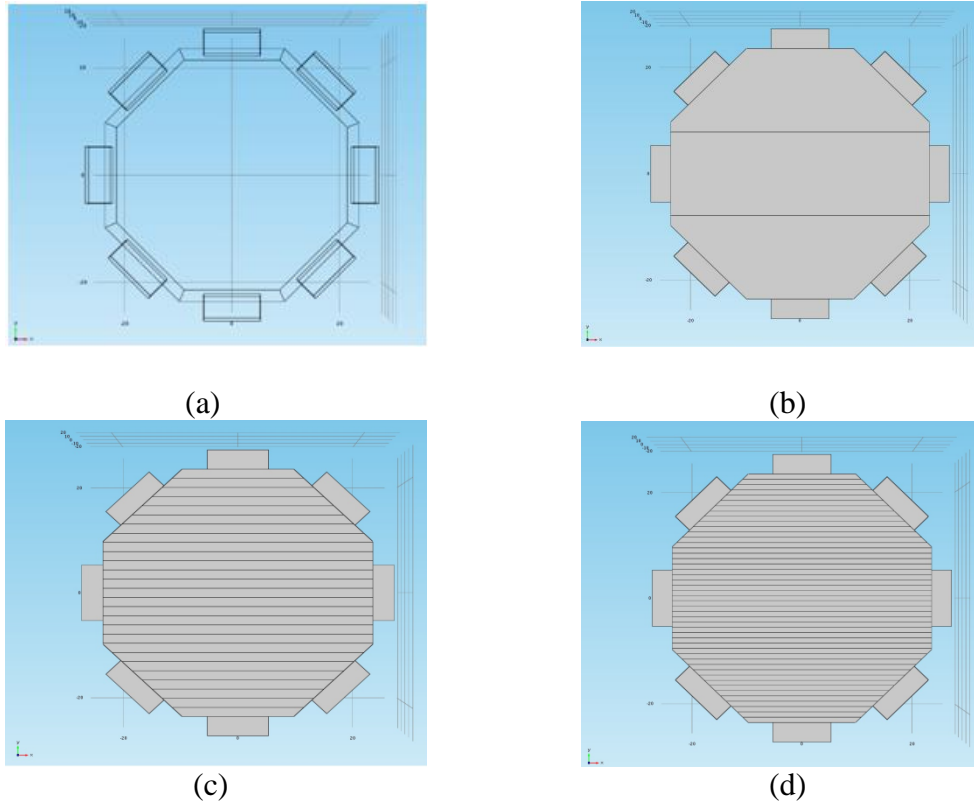
Figure 14: Cross-section of COMSOL models in the XY-plane for (a) 1 (b) 3 (c) 27 and (d) 47 gel layers respectively.

COMSOL (Comsol AB, Burlington MA) software was used to solve the forward problem by developing finite element models of MREIT experiments conducted. The Electric Currents Interface, available in COMSOL Multiphysics, was chosen to solve the steady-state current flow (i.e. electric current that does not change with time) in a conductive medium. The form of Maxwell's equations solved under a steady-state assumption for the voltage distribution (V) is :

$$\nabla.(-\sigma\nabla V) = 0$$

Other quantities derived from the voltage field V were : Electric field, $\mathbf{E} = -\nabla V$ and

Current density, $\mathbf{J} = \sigma \mathbf{E}$ where σ is the conductivity of the material.

The resultant voltage distributions were eventually used in calculating the first and second derivatives of the conductivity in phantoms. An octagonal three-dimensional model with eight recessed electrodes was constructed with overall dimensions of 52 m x 52 mm x 42 mm. The degree of anisotropy in the model was varied by increasing the number of gel layers. The first model (Figure 10a) consisted of a uniform isotropic high conductivity gel phantom of electrical conductivity 1.37 S/m and relative permittivity 80. The second model (Figure 10b) was anisotropic and composed of 3 alternating high and low conductivity gel layers of average thickness 14.2 mm/layer. The third model (Figure 10c) was anisotropic and composed of 27 alternating high and low conductivity gel layers of average thickness 1.57 mm/layer. The fourth model (Figure 10d) was anisotropic and composed of 47 alternating high and low conductivity gel layers of average thickness 0.91 mm/layer. The high and low conductivities in the second and third models are 1.37 S/m and 0.2 S/m.

The electrical conductivity and relative permittivity of electrodes in all three models was set at 1 S/m and 1 respectively. Current was injected normal to the surface of an electrode (Normal current density = 100 A/m^2 i.e. I = 10 mA) and the opposite was set as ground (Voltage = 0). The model was iteratively solved with a relative tolerance of 0.001.

3.6   MREIT Data Processing

3.6.1. Processing MREIT experimental data in MATLAB

3.6.1.1 Magnetic resonance image reconstruction

According to Bruker format, each scanning session is stored in a separate directory. Each experiment directory contains another subdirectory called 'pdata' along with other data files such as acquired parameters (acqp), method, fid, pulseprogram, spnam. Few files are described below:

(i) acqp : This text file contains base-level acquisition parameters.

(ii) fid : This data file contains raw and unreconstructed MR Free Induction Decay data, also known as "k-space" time-domain data.

(iii) method : This text file contains high-level acquisition parameters derived from acqp.

Magnetic resonance echoes stored in Free Induction Decay (.fid) and imaging parameters (acqp) files were read in MATLAB. Complex echo signals containing frequency and phase-encoded spatial information were Fourier transformed and the signals entered k-space. K-space is a 2D Fourier space with spatial frequency and amplitude information organized. A 2D Inverse Fourier Transform of the entire k-space entails magnetic resonance image reconstruction. One pixel transformation from k-space contributes a single spatial frequency to the image. Appendix C contains the code that reconstructs magnetic resonance complex data from free induction decays. The magnitude and phase components are separated to form magnitude and phase images.

3.6.1.2   Phase unwrapping and scaling

Complex MR data were decomposed into magnitude and phase components. Measured phase is technically a "wrapped phase" and must be unwrapped before further processing. This was achieved by implementing the Goldstein phase unwrapping algorithm. Once the phase was unwrapped, it was scaled to arrive at the B_z (Appendix D). Phase unwrapping algorithms are implemented to calculate the incremental phase change ψ. Rapid phase changes occur near current-injection electrodes and care must be taken in these regions.

3.6.1.3 Finite-element model

The electromagnetic field developed in MREIT experiments (as explained in section 3.5) were set up in COMSOL to simulate the current and magnetic field distributions. By solving the current density and voltage distributions for different current injections, it was possible to calculate the z-component of **B** developed using the Biot-Savart law in Equation 12. The C++ code to implement the Biot-Savart law is detailed in Appendix H.

3.6.1.3   Inverse solution

Internal magnetic flux densities B_{z_1} and B_{z_2}, due to the positive and negative injection currents were convolved to calculate the laplacian. In addition, the experimental protocol simulated in COMSOL produced voltage distributions of corresponding injections. These data were combined in the equation 9 to solve for gradient and laplacian of conductivity of the subject. (Appendix E)

3.7   DTI data processing

3.7.1   FMRIB's Software Library (FSL)

Raw DTI scans were collected from the 7T MRI scanner (Bruker, Biospec) and imaging parameters can be found in Table 4. These datasets were converted to NIfTI and processed in FSL to compute eigenvectors and eigenvalues. The first step in DTI processing is Eddy Current Correction, followed by Brain Extraction Tool and then by DTIFIT.

(i) Eddy Current Correction : Stretches and shears are induced in diffusion weighted images by eddy currents in gradient coils. These distortions differ with gradient directions and are corrected using an affine registration.

(ii) Brain extraction tool : This tool deletes non-brain tissue i.e. non-phantom part of the image of the sample chamber. Thereby creating a binary mask containing ones inside the phantom and zeros outside [13].

(iii) DTIFIT : DTIFIT models a diffusion tensor at each voxel. It is run on eddy current corrected data using additional inputs such as the binary mask, b values and gradient directions. The outputs of this operation, namely, Fractional Anisotropy, Eigenvalues and Eigenvectors were further processed in MATLAB.

3.7.2   MATLAB

3.7.2.1 Statistics of voxel parameters

Fractional Anisotropy (FA) is an index for the amount of diffusion asymmetry in a voxel calculated from eigenvalues. FA closer to zero indicates isotropic diffusion and FA closer to one indicates diffusion anisotropy. Binary masks were created from FA maps to obtain boundary information of phantoms. The average of Eigenvalues within the phantom were calculated. Average and standard errors of eigenvectors within the phantom were calculated using custom MATLAB codes (Appendix F).

CHAPTER 4

RESULTS

4.1 Diffusion Tensor Image Analysis

4.1.1 Quality of Diffusion Magnetic Resonance Imaging (DWI)

A quantitative measure of the quality of data collected by DWI is Signal-to-Noise ratio (SNR). A comparison of SNR at different isotropic voxel dimensions and diffusion gradient durations are presented in Figure 15. In Figure 15(a) the SNR was observed to be higher in acquisitions with diffusion gradients of 100 ms (blue) compared to 200 ms (maroon) duration in 10.5 mm x 10.5 mm x 10.5 mm voxels. Figure 15(b) shows higher SNR in measurements with voxel size 10.5 mm x 10.5 mm x 10.5 mm compared to 5.25 mm x 5.25 mm x 5.25 mm under the influence of 100 ms long diffusion gradients.
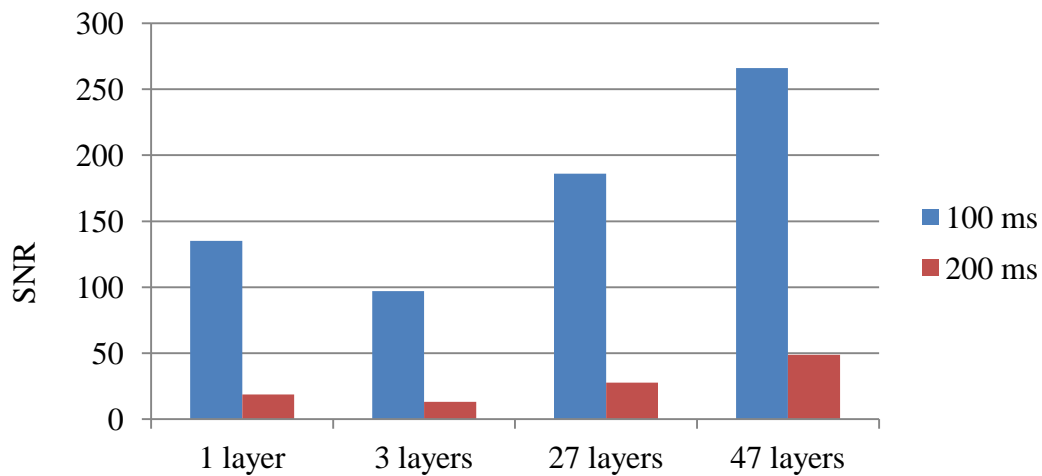
(a)
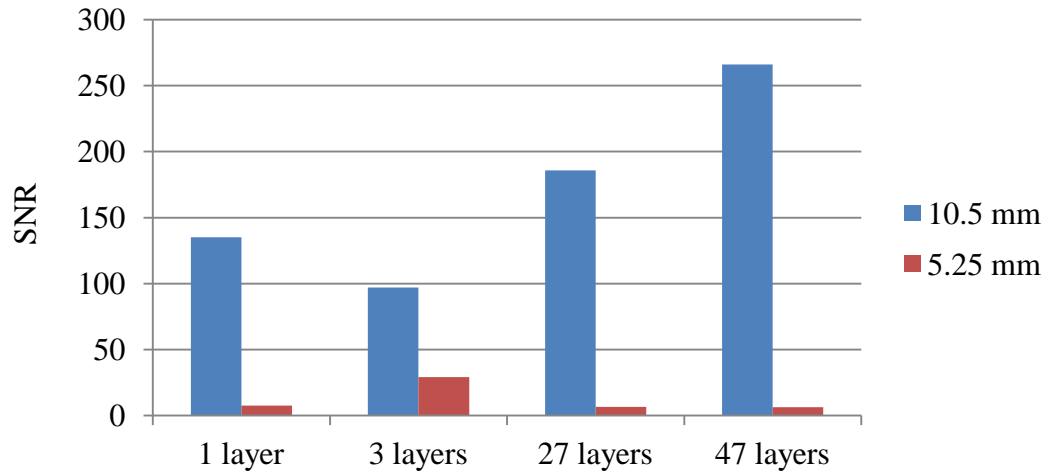
(b)

Figure 15: (a) Change in SNR with increasing length of diffusion gradients in isotropic voxels of side 10.5 mm. (b) Change in SNR with increasing isotropic voxel size under 100 ms diffusion-sensitizing gradient.

The percentage decrease in SNR between 100 ms and 200 ms DWI acquisitions is summarized in Table 6(a). The average percentage decrease is SNR among all four phantoms is 85%. Table 6(b) displays the percent decrease of SNR in voxels of side 10.5 mm and 5.25 mm. An average decrease of 90% was observed when isotropic voxels of size 5.25 mm were used instead of 10.5 mm.

Phantom	Isotropic voxel of side (mm)	Diffusion gradient duration (ms)	NEX	Echo time (ms)	SNR	Percent decrease in SNR
1 layer	10.5	200	2	410	18.65	86.1944
	10.5	100	1	210	135.09	
3 layer	10.5	200	2	410	13.08	86.5175
	10.5	100	1	210	97.015	
27 layers	10.5	200	2	410	27.71	85.0957
	10.5	100	1	210	185.92	

	10.5	200	2	410	48.76	
47 layers	10.5	100	1	210	266.05	81.6726

(a)

Phantom	Isotropic voxel of side (mm)	Diffusion gradient duration (ms)	NEX	Echo time (ms)	SNR	Percent decrease in SNR
1 layer	10.5	200	2	410	18.65	
	10.5	100	1	210	135.09	86.1944
3 layer	10.5	200	2	410	13.08	
	10.5	100	1	210	97.015	86.5175
27 layers	10.5	200	2	410	27.71	
	10.5	100	1	210	185.92	85.0957
47 layers	10.5	200	2	410	48.76	
	10.5	100	1	210	266.05	81.6726

(a)

Phantom	Isotropic voxel of side (mm)	Diffusion gradient duration (ms)	NEX	Echo time (ms)	SNR	Percent decrease in SNR
1 layer	5.25	100	1	210	7.4	
	10.5	100	1	210	135.09	94.52217
3 layer	5.25	100	1	210	29.03	
	10.5	100	1	210	97.015	70.076792
27 layers	5.25	100	1	210	6.63	
	10.5	100	1	210	185.92	96.43395
47 layers	5.25	100	1	210	6.26	
	10.5	100	1	210	266.05	97.647059

(b)

Table 6: (a) Percent decrease in SNR with increase in length of diffusion-sensitizing magnetic field gradients in isotropic voxels of side 10.5 mm. (b) Percent decrease in SNR with increase size of isotropic voxels under diffusion-sensitizing gradients of 100 ms duration.

4.1.2   Properties of the Diffusion Tensor with increasing degree of anisotropy

The Diffusion Tensor is used to model local diffusion within a voxel based on the assumption that local diffusion is characterized by a 3D Gaussian distribution, whose covariance matrix is proportional to the diffusion tensor, ***D***. Six elements of the Diffusion Tensor are estimated by solving six independent equations resulting from the Stejskal-Tanner equation with six diffusion gradients. The ADCs from D are along the scanner's coordinate system. The diffusion tensor ***D*** is parameterized to depend on eigenvalues and eigenvectors that determine the shape and orientation of the tensor.   Eigenvalues and eigenvectors are calculated from ***D*** using FMRIB software library FSL[18].

4.1.2.1 Eigenvalues of Diffusion Tensor

The degree of anisotropy in TX-151 phantoms was controlled by the number of gel layers. The characteristics of diffusion of water molecules is understood from the eigenvalues and eigenvectors of the diffusion tensor in each voxel. Table 7 summarizes the fractional anisotropy, eigenvalues and mean diffusivity of TX-151 phantoms. The SNR in scans collected over isotropic voxels of side 10.5 mm under the influence of diffusion-encoding gradients over 100 ms was high. Though the SNR in 27 and 47 layer phantoms were high (i.e. 186 and 266 respectively), the third eigenvalue was negative. The accuracy of fractional anisotropy (FA) and mean diffusivity (MD) in the presence of negative eigenvalues was uncertain. Table 7 shows the 1 layer phantom to be anisotropic in terms of fractional anisotropy(FA= 0.6) and the 47 layer phantom was highly anisotropic with FA exceeding 1 (FA = 1.04). Mean diffusivity (MD) in 1 and 3 layer phantoms were high in comparison with 27 and 47 layers. High MD indicates isotropic

diffusion in 2 and 3 layers whereas low MD implies anisotropic diffusion in 27 and 47 layers.

Phantom	SNR	FA	λ_1	λ_2	λ_3	MD
1 layer	135	0.6190	8.3e-4	4.5e-4	1.6e-4	4.78e-4
3 layers	97	0.3629	7e-4	5e-4	3.2e-4	5.13e-4
27 layers	186	0.9329	8.2e-4	2e-4	-4.5e-4	1.92e-4
47 layers	266	1.0456	13e-4	2.3e-4	-8.5e-4	2.1e-4

Table 7: Fractional anisotropy (FA), eigenvalues ($\lambda_1, \lambda_2, \lambda_3$) of diffusion tensor and mean diffusivity (MD) of all four TX-151 phantoms imaged over 10.5 mm x 10.5 mm x 10.5 mm voxels and diffusion gradients of 200ms duration.

An alternative method to characterize the nature of diffusion is to compare the ratio of two largest eigenvalues among phantoms with varying anisotropy. Table 8 shows the ratio to be greater than 2 in case of 27 and 47 layers. This indicates greater diffusion along the principal eigenvector (V1) compared to V2. An additional ratio between the largest eigenvalue and mean diffusivity is calculated as shown in Table 8. Similar to λ_1/λ_2, the ratio of $\lambda_1/$ MD was less than 2 in isotropic phantoms. However, the integrity of MD maybe compromised by the presence of negative eigenvalues.

Phantom	SNR	λ_1	λ_2	λ_3	λ_1/λ_2	MD	λ_1/MD
1 layer	135	8.3e-4	4.5e-4	1.6e-4	1.84	4.78e-4	1.74
3 layers	97	7e-4	5e-4	3.2e-4	1.40	5.13e-4	1.36
27 layers	186	8.2e-4	2e-4	-4.5e-4	4.10	1.92e-4	4.27
47 layers	266	13e-4	2.3e-4	-8.5e-4	5.65	2.1e-4	6.19

Table 8: Estimates to measure diffusion along V1 in terms of the largest eigenvalue compared to diffusion along V2 and the mean diffusivity.

4.1.2.2 Eigenvectors of Diffusion Tensor

Eigenvectors of a diffusion tensor provide directional information. Figure 16 is a 3D plot of the first eigenvector. The first eigenvector is associated with the largest eigenvalue and is considered to indicate the direction of preferred diffusion in anisotropic samples. Phantoms comprising of 1 and 47 layers had much smaller y-components in comparison to x- and z-components. The x-component of V1 in 27 layer phantom is larger than y- and z-components.

TX-151 phantom arrangement	Principal eigenvector (V1)
1 layer	0.6137±0.1189
3 layers	0.1721±0.1320
27 layers	0.1651±0.1534
47 layers	0.4002±0.1061

Table 9: Mean and standard error of the principal eigenvector in TX-151 phantoms of increasing degree of anisotropy.
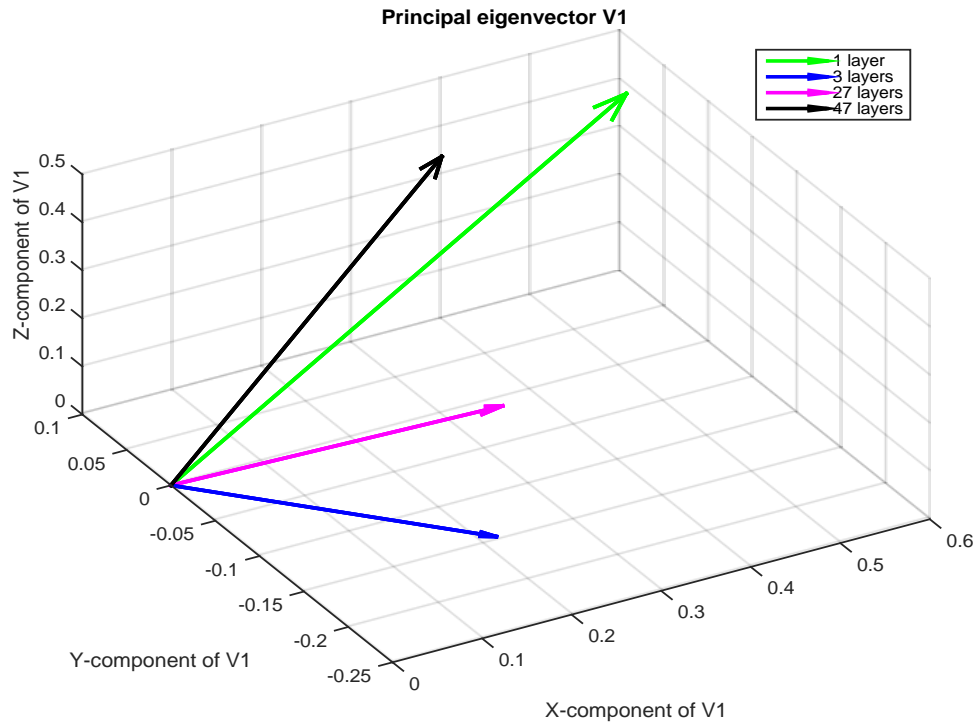
Figure 16: 3D plot of the mean of principal eigenvector (V1) in all four TX-151 gel phantoms.

4.2 Magnetic Resonance Electrical Imaging Tomography (MREIT) Data Processing

4.2.1 Quality of Magnetic Resonance Electrical Imaging Tomography (MREIT)

The Signal-to-noise ratio (SNR) in magnitude images injected by 10 mA vertical current is noted to decrease with increase in the size of a square ROI mask in all TX-151 phantoms. In Figure 17, the magnitude of change in SNR with ROI was large, however, it remained fairly stable within size range of 6-8 pixels (i.e. 5.625 mm - 7.5 mm). The SNR in 1 and 3 layer phantoms sharply decreased in square ROIs of side 12. Similar reductions in SNR were observed in 27 and 47 layer phantoms in ROIs of side 11 and 10 respectively. ROIs of size 7 pixels (6.5695 mm) was chosen for the analysis.

43
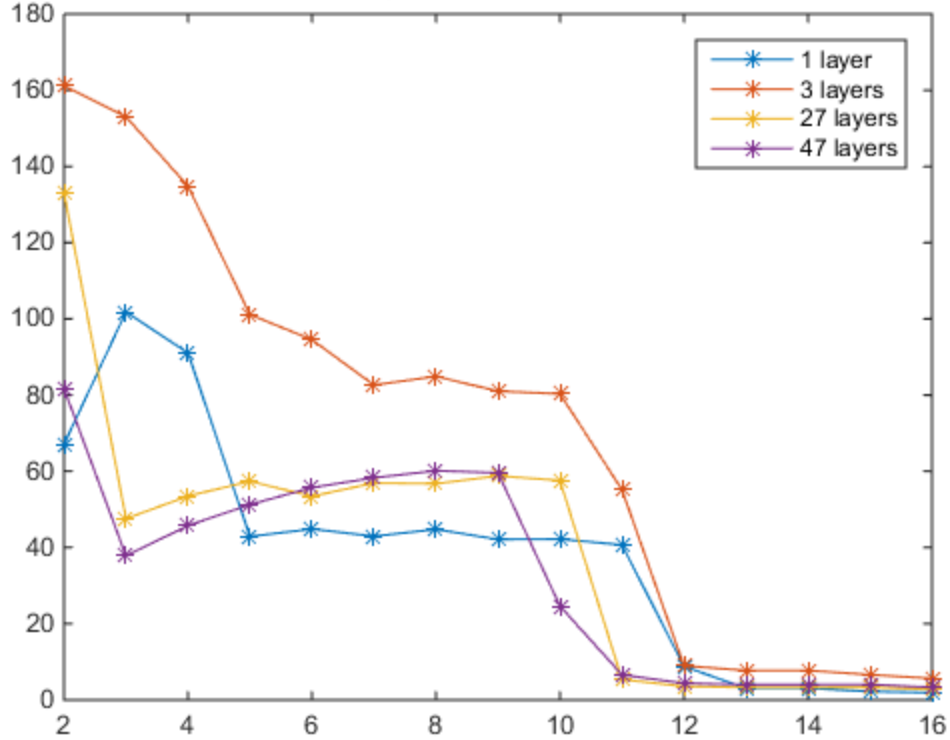
Figure 17: SNR on y-axis and square ROI of sides in pixels (1 pixel = 10.5 mm)

4.2.2 Complex MREIT data to spatial derivative of conductivity distribution in TX-151 phantoms

The raw data collected in MREIT experiments are complex in nature. The imaginary component contains phase information and is essential in MREIT. The MR phase change due to current injection in MREIT is proportional to B_z. MR phase images were unwrapped and scaled to calculate B_z as detailed in Section 3.6.1.2. B_z images in TX-151 phantoms due to a horizontal current injection shows spatial deflections at the boundary of alternating high and low gel layers. Conductivity contrast exists at each boundary between gel layers of different conductivities[22].

44

(a)   (b)

(c)   (d)

(e)   (f)
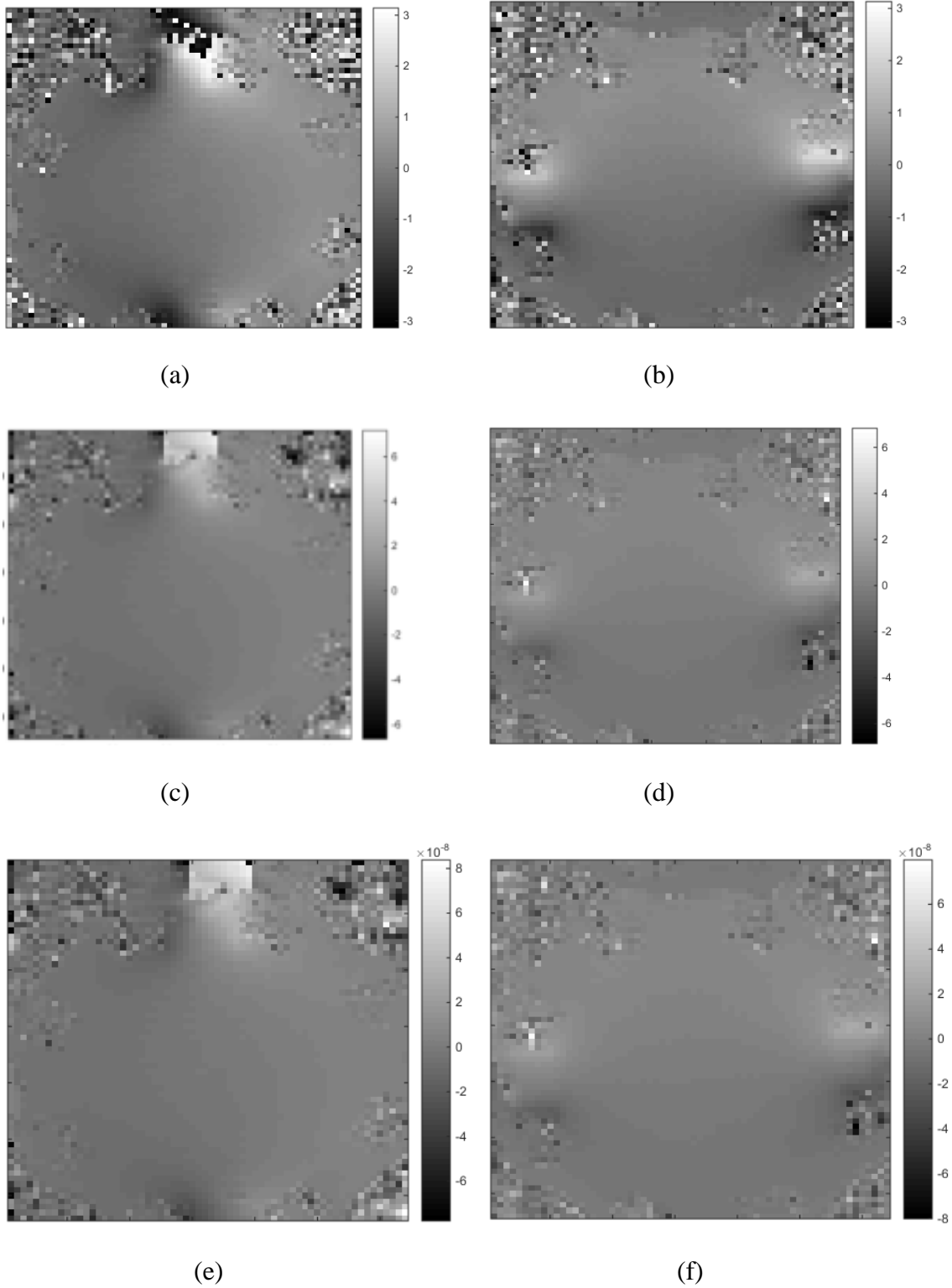
Figure 18: 47 layer TX-151 phantom is subjected to 10 mA vertical (a,c,e) and horizontal
(b, d, f) AC current. Wrapped phase images (a, b), unwrapped phase images (c, d) and
Bz (e, f) were displayed for vertical and horizontal current injections respectively.

In Figure 19, the ramps in B_z due to a horizontal positive current injection indicated the presence of a conductivity contrast. The B_z profiles of 1, 27 and 47 gel layers were similar. The 3 layer phantom has a thickness of approximately 12 mm per layer and is reflected in the profile. In case of 27 and 47 layers, the layer deflections are much smaller because each voxel has multiple layers.

Conductivity contrast in 1 layer phantom is zero because only high conductivity gel is used. In other slice phantoms, the conductivity contrast is constant because the absolute values of conductivity are the same in all phantoms. Only the thickness per gel layer is changed among phantoms. Hence, the slope of all slice phantoms must be the same. However, this is not the case. 1, 27, 47 layer phantoms have very similar slopes..
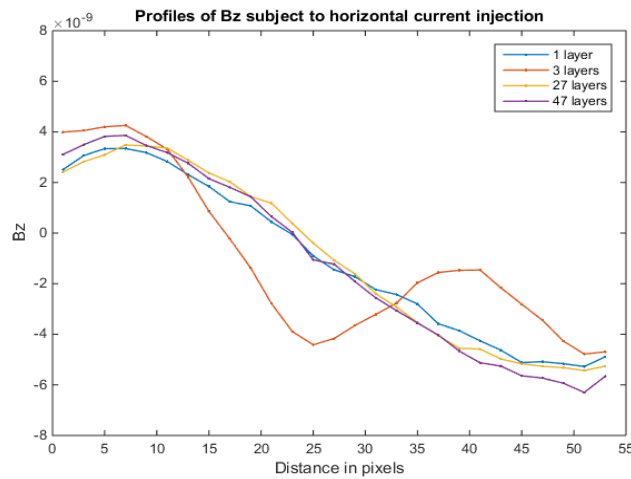
(a)
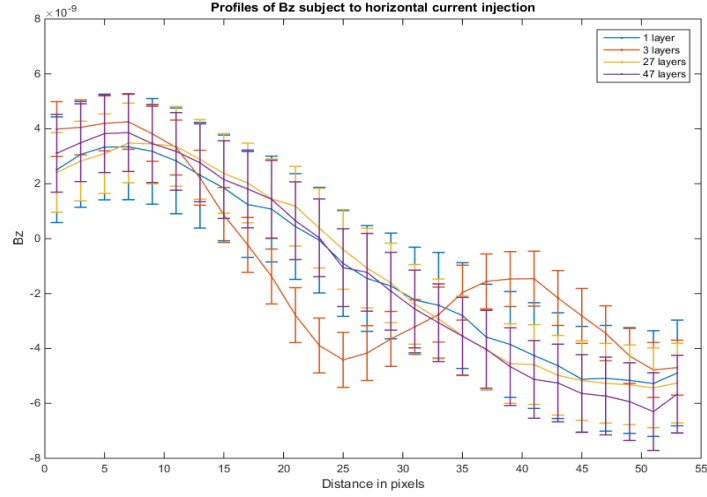
(b)

Figure 19: Spatial profiles of the (a) z-component of internal magnetic flux density (**B**) and (b) standard deviation of **B** in TX-151 gel phantoms subjected to horizontal current injection pair.

Phantom	SNR	Standard deviation of B_z
1 layer	42.96	3.8457e-9
3 layers	82.63	1.9994e-9
27 layers	56.91	2.9030e-9
47 layers	58.30	2.8338e-9

Table 10: Standard deviation of Bz in TX-151 phantoms subjected to horizontal current injection.
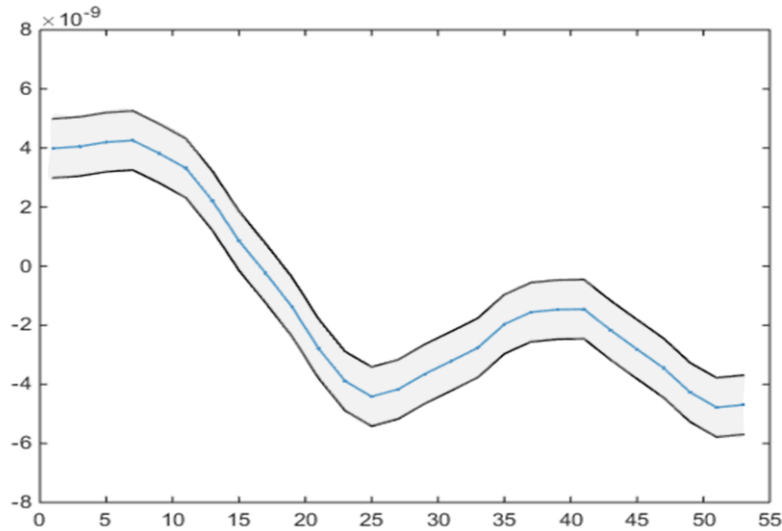
Figure 20: Average and standard deviation (shaded area) of B_z in 3 layer TX-151 gel phantom.

B_z from unwrapped phase was combined with voltage distributions due to orthogonal current injections from COMSOL as detailed in Section 3.6.1.3. Voltage distributions from COMSOL are displayed in Figure 21.

(a)

(b)
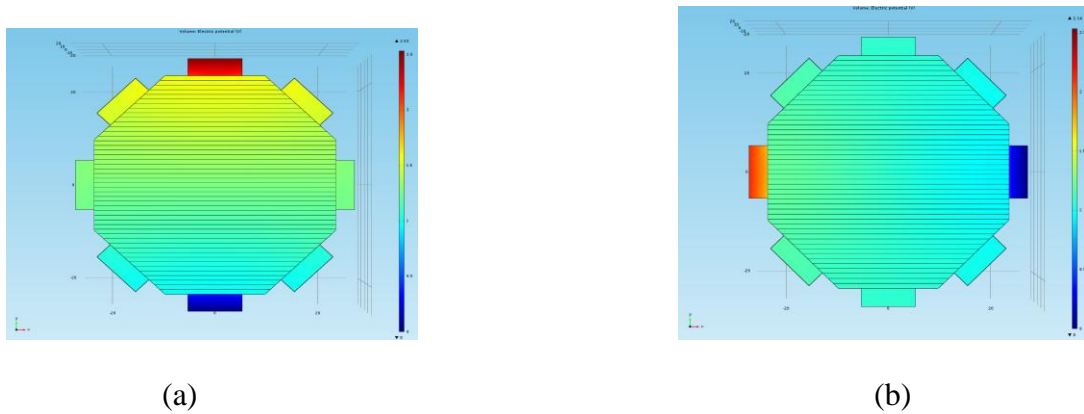
Figure 21: Voltage distribution in 47 layer TX - 151 gel phantom arrangement subjected to vertical and horizontal current injections.

Laplacian of conductivity due to horizontal and diagonal current injection pairs can be seen in Figures 22 and 23. The magnitude of laplacian of conductivity ($\nabla^2\sigma$) is

observed to be higher in regions near current-injection electrodes. By visual inspection, the magnitude of $\nabla^2\sigma$ in 1 and 3 layer phantoms is similar in Horizontal (HV) and Diagonal current injection pairs. The magnitudes are very low in high conductivity gel regions and high in low conductivity gel regions at the boundary of conductivity contrast. However, in 27and 47 gel layer phantoms the magnitude of $\nabla^2\sigma$ is different in horizontal and diagonal current injection pairs. In the case of 27 layers, gel layers are visible throughout the phantom under a horizontal (HV) current injection pair. In contrast, the magnitude of $\nabla^2\sigma$ in 27 layers phantom decreases with distance from diagonally injecting current electrodes. Similar yet more pronounced observations are made in the 47 layers phantom. The visibility of gel layers change from visible throughout the phantom to invisible as current injection is changed from horizontal to diagonal current injection pair.
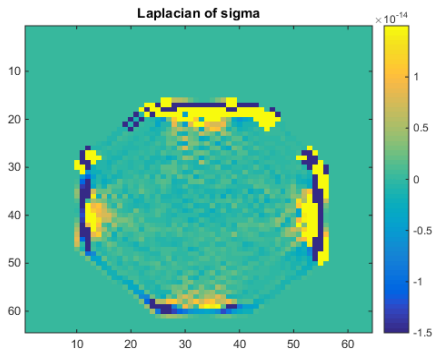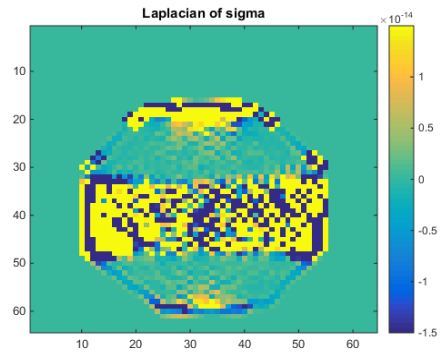
(a)   (b)
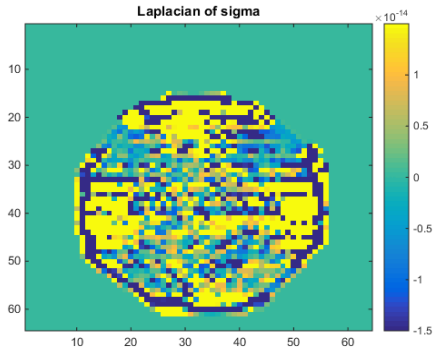
(c)
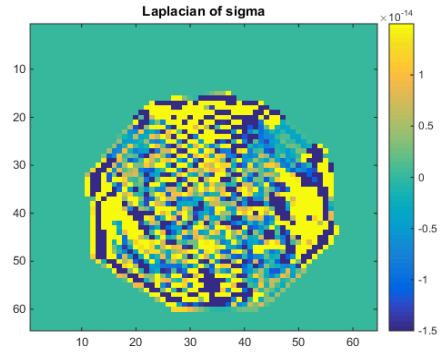
(d)

Figure 22: Laplacian of sigma in (a) 1 layer (b)3 layers (c) 27 layers and (d) 47 layers TX-151 phantoms subject to horizontal and vertical current injection pair. Scale = [-1.5e-14, 1.5e-14]
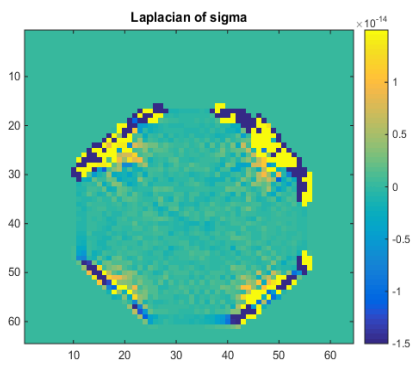
(a)
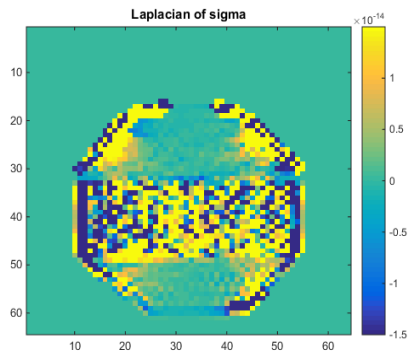
(b)
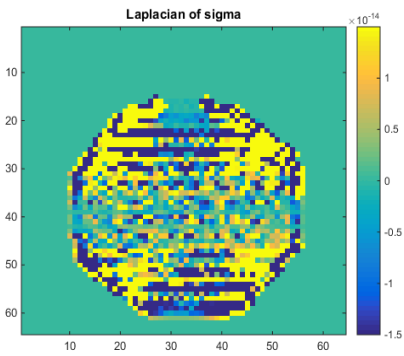
(c)
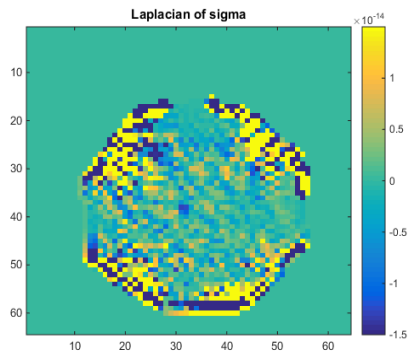
(d)

Figure 23: Laplacian of sigma in (a) 1 layer (b)3 layers (c) 27 layers and (d) 47 layers
TX-151 phantoms subject to diagonal current injection pair. Scale = [-1.5e-14, 1.5e-14].

Phantom	Horizontal current injection pair		
	Top	Middle	Bottom
1 layer	1.97e-16±1.58e-15	1.50e-16±9.49e-16	4.78e-16±1.06e-15
3 layers	5.69e-16±1.40e-15	2.12e-15±4.24e-14	5.05e-16±1.52e-15
27 layers	2.19e-15±1.61e-14	2.14e-15±1.90e-14	5.23e-16±1.90e-14
47 layers	1.09e-15±1.41e-14	-4.09e-16±1.19e-14	-1.34e-15±2.63e-14

(a)

Phantom	Diagonal current injection pair		
	Top	Middle	Bottom
1 layer	6.75e-16±2.87e-15	3.15e-17±7.91e-16	1.01e-15±2.15e-15
3 layers	1.20e-15±2.04e-15	6.31e-15±2.43e-14	1.67e-15±2.93e-15
27 layers	2e-15±3.38e-14	1.92e-16±1.02e-14	1.93e-15±4.75e-14
47 layers	-2.15e-16±1.23e-14	-2.12e-16±4.34e-15	5.05e-15±2.93e-14

(b)

Table 11: Local spatial averages of laplacian of conductivity in all four phantoms subject
to (a) Horizontal and (b) Diagonal current injection pairs

CHAPTER 5

DISCUSSION

5.1   Diffusion-Weighted Magnetic Resonance Imaging (DWI)

Diffusion of water molecules in living tissues depends on the structure of the medium. Diffusion weighted magnetic resonance imaging (DWI) measures the diffusion of water molecules and is useful in the *in vivo* determination of orientation of white matter tracts. Diffusion is isotropic (i.e. equal in all directions) if the medium is homogeneous and anisotropic (i.e. not equal in all directions) if the medium is inhomogeneous. In other words, diffusion is described as isotropic in the absence of any restriction to the mobility of water molecules. However, diffusion is anisotropic if there is restricted mobility of water molecules in any direction. The presence of parallel axonal membranes within white matter is primarily responsible in restricting the perpendicular motion of water molecules and generating anisotropy[19]. TX-151 gel phantoms were substituted for white matter tracts with the purpose of evaluating diffusion anisotropy. Water molecules follow the structure of TX-151 gel layers and move freely along rather than across each layer.

The quality of data acquired by DWI is measured by Signal-to-Noise ratio (SNR). The most important factor known to affect the SNR of diffusion weighted images is echo time (TE). The loss of signal due to T_2 decay must be as small as possible because the signal is further attenuated in the presence of diffusion gradients. TE depends on the duration and separation between diffusion-sensitizing magnetic field gradients. T_2 decay

is minimized by using the smallest possible TE. The signal in baseline images (b-value = 0) is affected by T_2 signal decay whereas directional data is further attenuated by diffusion. Therefore, SNR in baseline images is higher compared to diffusion weighted images.

The influence of imaging parameters such as voxel size and duration of diffusion gradients on the quality of DWI acquisitions is summarized in Table 6 and Figure 15. Reducing the voxel size and/or increasing the duration of diffusion gradients had a profound impact on the SNR. Decreasing the voxel size by a factor of 2 resulted in 85% decrease in SNR. Similarly, increasing the duration of diffusion gradients by a factor of 2 resulted in 90% decrease in SNR. Based on these observations, Diffusion Tensor Imaging (DTI) analysis was performed on DWI data collected with 10.5 mm x 10.5 mm x 10.5 mm voxels and 100 ms diffusion-sensitizing magnetic field gradients. Einstein's law of diffusion describes the relationship between diffusion distance and diffusion time. With increase in the diffusion time, the mean squared distance traveled by a water molecule is increased. The longer diffusion is allowed, the more likely it is to identify the presence of a preferred diffusion direction. If in fact, a preferred diffusion direction is present, then the tensor is anisotropic.

Diffusion properties in TX-151 phantoms were studied based on the average and standard error of eigenvalues and eigenvectors of diffusion tensors. Common measures to describe the overall diffusion are fractional anisotropy (FA) and mean diffusivity (MD). Both these measures are based solely on eigenvalues, thereby necessitating eigenvalues to

be real and positive. However, table 7 displays a negative eigenvalue in 27 and 47 gel layers. A previous study observed an increase in the probability of negative eigenvalues with increase in anisotropy and noise. As the SNR of both 27 and 47 phantoms were greater than 150, the occurrence of negative eigenvalues may be attributed to increase in the level of anisotropy.

A previous study performed Monte Carlo simulations and isotropic water phantom experiments to evaluate the accuracy of fractional anisotropy (FA) over increasing levels of anisotropy. The bias and standard deviation of FA was high in the low anisotropy range and reduces with increase in degree of anisotropy[24]. This instability in FA could be the reason for overestimating FA in 1 layer phantom. FA exceeds 1 in the 47 layer phantom and this could be due to the presence of a negative eigenvalue. These observations render FA as an unreliable measure of anisotropy in this study.

Inappropriate sorting of negative eigenvalues contributes to an estimation bias in diffusion anisotropy. This sorting bias leads to an overestimation of the largest eigenvalue and underestimation of the smallest eigenvalue. Measures adversely affected by the sorting bias are axial and radial diffusivity. This could be the reason for high FA in 1 and 47 gel layers. A better measure for diffusion anisotropy would be a lattice index based on spatial averaging of eigenvalues and eigenvectors. Eigenvectors are inherently robust to noise, thereby rendering the lattice index to be an accurate estimate of anisotropy[24].

Sorting bias in eigenvalues leads to overestimation of λ_1 and underestimation of λ_3. This renders ratios such as axial diffusivity ($\lambda_{//}$ i.e. λ_1 because λ_1 is parallel to fibers) and radial diffusivity (λ_3) to be unreliable measures. Mean diffusivity characterizes the overall diffusion. Higher values of Mean diffusivity were seen in 1 and 3 layer phantoms indicates isotropic behavior. A decrease of MD in 27 and 47 layer phantoms indicates anisotropic behavior. While the MD in 1 layer was expected to be higher than in 3 layers, this was not observed. Similarly, MD in 27 layers was expected to be higher than in 47 layers. However, the presence of negative eigenvalues may have affected the measure of MD [24].

The ratio of first and second eigenvalues sorted in descending order as well as the ratio of largest eigenvalue with mean diffusivity are displayed in Table 8. Both these ratios were less than 2 in 1 and 3 layer phantoms indicating isotropic diffusion. However, these ratios were greater than 2 in 27 and 47 layer phantoms indicating anisotropic diffusion. The trend in the ratio of the largest eigenvalue to mean diffusivity was the same as that observed in another study. However, the smallest eigenvalues in Table 8 were negative, thereby making ratios in eigenvalues more reliable than a ratio with the mean diffusivity. Another study reported the ratio of largest eigenvalue to the mean diffusivity more than 2 as indicative of diffusion anisotropy. This relationship holds true for 27 and 47 gel layers. The probability of obtaining negative eigenvalues increases with the degree of diffusion anisotropy and noise level.

Other studies show the MD in gray and white matter in the brain are very similar, however, the degree of anisotropy is very different as a result of their unique structure. A similar observation can be made : phantoms with very close MD values such as 1 and 3 layers and 27 and 47 layers have very different FA values.

5.2 Magnetic Resonance Electrical Impedance Tomography (MREIT)

Conductivity images acquired in previous studies in previous studies by passing current in the horizontal direction can recognize layers. However, when current is in the vertical direction, but layers in horizontal, then layers are not recognizable.

5.2.1 Influence of the orientation of current-injection electrode pair

Within each current injection pair, the magnitude of $\nabla^2\sigma$ changes with conductivity contrast and number of gel layers (anisotropy ratio). In phantoms under a horizontal current injection pair, gel layers were visible and increase in accordance with the arrangement of TX-151 gel layers. However, under a diagonal current injection pair, gel layers were not visible particularly in the middle region of the phantom in 27 and throughout the 47 layer phantom. This may happen due to the difference in orientation of electrode pairs to gel layers.

In slice phantoms (3, 27 and 47 layer phantoms), TX-151gel layers of high and low conductivity were alternately stacked with long edges in x- and z-directions. The gel layer arrangement appears as a parallel circuit to current injected in the horizontal direction. The effective resistance of a parallel circuit (R_p) is a sum of the reciprocal of

individual gel layer resistances.   The same gel layer arrangement appeared as a series circuit to current injected in the vertical direction. The effective resistance of a series circuit (R_s) is a sum of individual gel layer resistances. The utilization of a constant current source ensured the amount of injection current was 10 mA. Based on Ohm's law, the difference in effective resistance under a horizontal and vertical current injection influences the electric field. The electric field established by a horizontal current injection is non-symmetric with a field due to a vertical current injection. However, an orthogonal current injection pair through electrodes on diagonal surfaces of the octagonal sample chamber ensured the orientation of electrodes with gel layers was identical. This ensured equal effective resistance towards each of the two diagonal current injections. Equal effective resistance in both diagonal current injections established two symmetric electric fields. The symmetry property of electric fields developed when subject to two orthogonal current injections facilitates the capture of layers information only in horizontal current injection pairs.

CHAPTER 6

CONCLUSION

The second spatial derivative of conductivity ($\nabla^2\sigma$) was found to be influenced by electrode orientation and degree of anisotropy. The contrast in conductivity of TX-151 gel layers was visible in $\nabla^2\sigma$ under current injections parallel and perpendicular to TX-151 gel layers.

Tensors from diffusion magnetic resonance imaging (DWI) contained negative eigenvalues. The accuracy of common measures in DTI such as fractional anisotropy and mean diffusivity were affected by negative eigenvalues. However, the ratio of two largest eigenvalues exceeded 2 in anisotropic phantoms (27 and 47 layers).

REFERENCES

[1] E. Degirmenci and B. M. Eyuboglu, "Anisotropic conductivity imaging with MREIT using equipotential projection algorithm," Physics in Medicine and Biology, vol. 52, no. 24, pp. 7229-7242, 2007.

[2] D. Miklavcic, N. Pavselj and F. X. Hart, Electric properties of tissues, Wiley Online Library, 2006.

[3] Electrical Impedance Tomography, London: IOP Publishing Ltd, 2005.

[4] D. S. Holder, Electrical Impedance Tomography, London: Institute of Physics , 2005.

[5] K. Jeon and C.-O. Lee, "CoReHA 2.0: A Software Package for In Vivo MREIT Experiments," Computational and Mathematical Methods in Medicine, vol. 2013, pp. 1-8, 2013.

[6] H. Johansen-Berg and T. Behrens, Diffusion MRI, San Diego: Academic Press, 2009.

[7] R. J. Sadleir, F. Neralwal, T. Te and A. Tucker, "A Controllably Anisotropic Conductivity or Diffusion Phantom Constructed from Isotropic Layers," Annals of Biomedical Engineering, pp. 2522-2531, 2009.

[8] R. J. Sadleir, S. C. Grant and E. J. Woo, "Can High-Field MREIT be used to directly detect neural activity? Theoretical Considerations," Neuroimage, pp. 205-216, 2010.

[9] Y. T. Kim, P. J. Yoo, T. I. Oh and E. J. Woo, "Development of a low noise MREIT current source," in Journal of Physics, 2010.

[10] E. J. Woo and J. K. Seo, "Magnetic resonance electrical impedance tomography (MREIT) for high-resolution conductivity imaging," Physiological measurement, vol. 29, no. 10, pp. 1-26, 2008.

[11] Model 4192A LF Impedance Analyzer, Tokyo: Yokogawa Hewlett Packard, 1984.

[12] P. P. Silvester and R. L. Ferrari, Finite Elements for Electrical Engineers, Cambridge University Press, 1996.

[13] S. M. Smith, "Fast robust auomated brain extraction.," *Human Brain Mapping,* vol. 17, no. 3, pp. 143-155, 2002.

[14] E. Degirmenci and B. M. Eyuboglu, "Anisotropic conductivity imaging with MREIT using equipotential projection algorithm," *Physics in Medicine and Biology,* pp. 7229-7242, 2007.

[15] E. Degirmenci and B. M. Eyuboglu, "Anisotropic conductivity imaging with MREIT using equipotential projection algorithm," *Physics in Medicine and Biology,* pp. 7229-7242, 2007.

[16] Y. T. Kim, T. I. Oh and E. J. Woo, "Experimental verification of contrast mechanism in Magnetic Resonance Electrical Impedance Tomography (MREIT)," *IEEE,* pp. 4987-4990, 2010.

[17] J. K. Seo, F. C. Pyo and C. Park, "Image reconstruction of anisotropic conductivity tensor distribution in MREIT : A computer simulation study," *Physics in Medicine and Biology,* pp. 4371-4382, 2004.

[18]   T.E.J. Behrens, M.W. Woolrich, M. Jenkinson, H. Johansen-Berg, R.G. Nunes, S. Clare, P.M. Matthews, J.M. Brady, S. M. Smith, "Characterization and propagation of uncertainity in diffusion-weighted MR imaging," *Magnetic Resonance in Imaging*, pp. 1077-1088, 2003.

APPENDIX A

GLOSSARY OF TERMS

Common MR imaging terms

Repetition time (TR) is the time between the application successive RF pulses applied to the same slice. TR affects the total scan time and varying TR has a significant effect on the characteristics of image contrast. TR values are short for T1 contrast and long for T2 contrast.

Echo time (TE) is the time in between the 90^0 pulse and the peak of the echo signal in Spin echo and Inversion Recovery pulse sequences.

Number of averages (NEX) indicates the number of times a line is acquired in k-space.

Longitudinal relaxation time (T_1) is a time constant measuring the rate at which the longitudinal magnetization returns to the equilibrium value after an excitation pulse is administered to the sample slice. In other words, T_1 is the rate at which excited protons return to equilibrium within the lattice. The longitudinal magnetization is expected to grow from zero to 63% of its final value in T_1 time.

Field of view (FOV) is the size of the spatial encoding area.

Slice thickness: Thickness of an imaging slice in Z-direction.

Scan time: The total time required to acquire all the data needed to produce the programmed image.

Spatial resolution: Ability to define minute adjacent objects or points in an image.

Acquisition matrix: The total number of independent data samples in the frequency and phase directions.

Mean diffusivity (MD) : A measure of the bulk diffusivity ignoring directional preference and is calculated by averaging the three eigenvalues.

Files created during a MR experiment by a Bruker Biospec machine

(i) acqp : This text file contains base-level acquisition parameters.

(ii) fid : This data file contains raw and unreconstructed MR Free Induction Decay data, also known as "k-space" time-domain data.

(iii) method : This text file contains high-level acquisition parameters derived from acqp.

(iv) pulseprogram : Text file containing the MR sequence

(v) spnam (spnam0, spnam1) : Shape pulse definition during acquisition.

The pdata sub-directory contains one subdirectory numbered as "1" which contains the reconstruction of the raw data into images.

(i) 2dseq : Processed image data expressed in a raw binary format without header.

(ii) d3proc : Description of the image data contained in the 2dseq file

(iii) id : Unique dataset identification

(iv) meta: Used for backward compatibility between different Bruker software.

(v) procs: Used for backward compatibility between different Bruker software.

(vi) reco : Text file including input and output parameters for the reconstruction process

(vii) visu_pairs Parameters for postprocessing , conversion and data display

APPENDIX B

IMPEDANCE MEASUREMENT USING HP4192A

HP4192A LF Impedance Analyzer was useful in measuring impedance parameters. Current was injected through TX-151 gel layer arrangements via surface electrodes and voltage was recorded by the four-probe method as shown in Figures 24 - 25[3]. Metal plates were used as current injection electrodes and copper tape as voltage recording electrodes as can be seen in Figure 25. The LabVIEW code in Figures 26 - 28 remotely controlled the impedance analyzer HP4192A. The impedance was recorded over 4 hours and the conductivity was calculated based on the equation in Section 3.4.
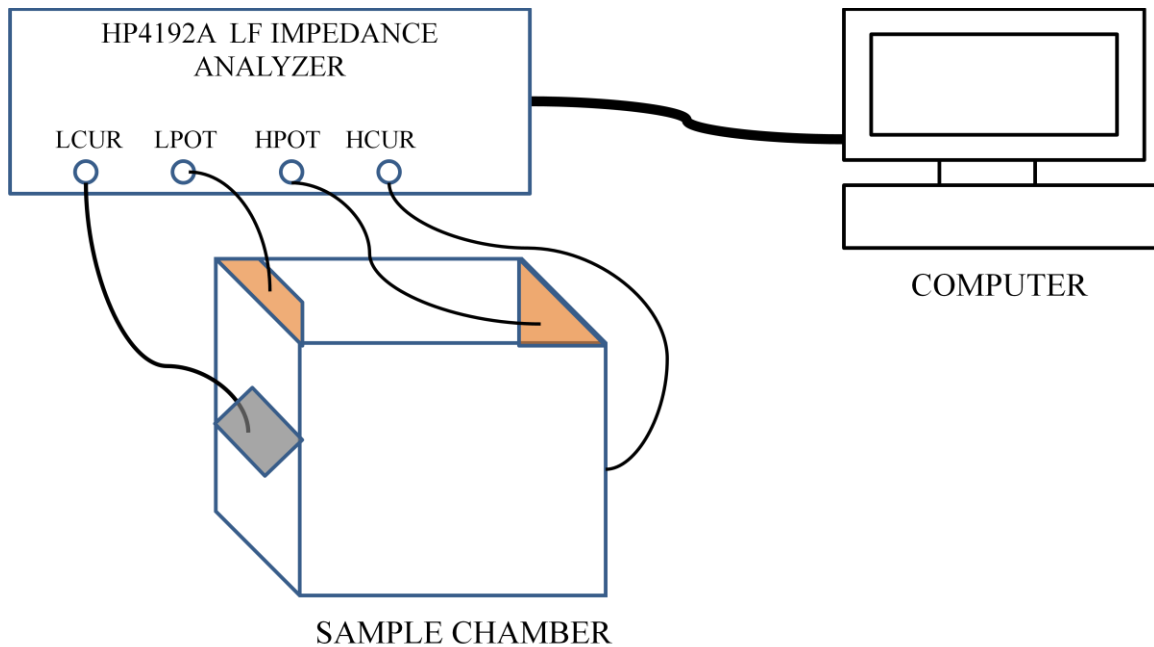
Figure 24: Schematic diagram to measure the impedance in a rectangular sample chamber (5 cm x 5 cm x 5 cm) containing TX-151 gels. The LF impedance analyzer HP4192A was remotely controlled by a LabVIEW code executed on the computer.
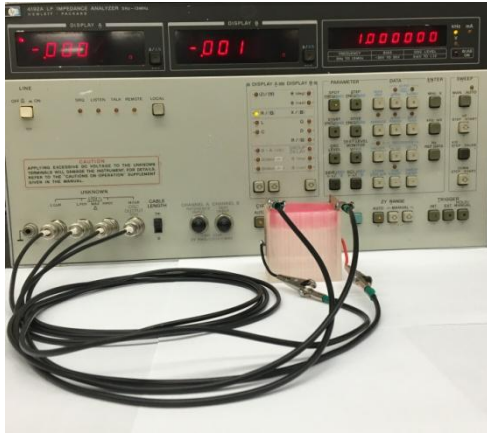
Figure 25: Pictorial representation of the measurement of impedance in high conductivity TX-151 gel in a sample chamber (5 cm x 5 cm x 5 cm) using four-probe electrode method.
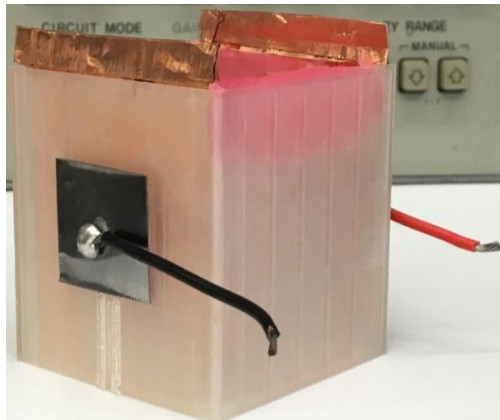
Figure 26: Rectangular sample chamber (5 cm x 5 cm x 5 cm) with two metal plates as surface electrodes for current injection and copper tape adhered to opposite walls of the chamber for voltage recording.
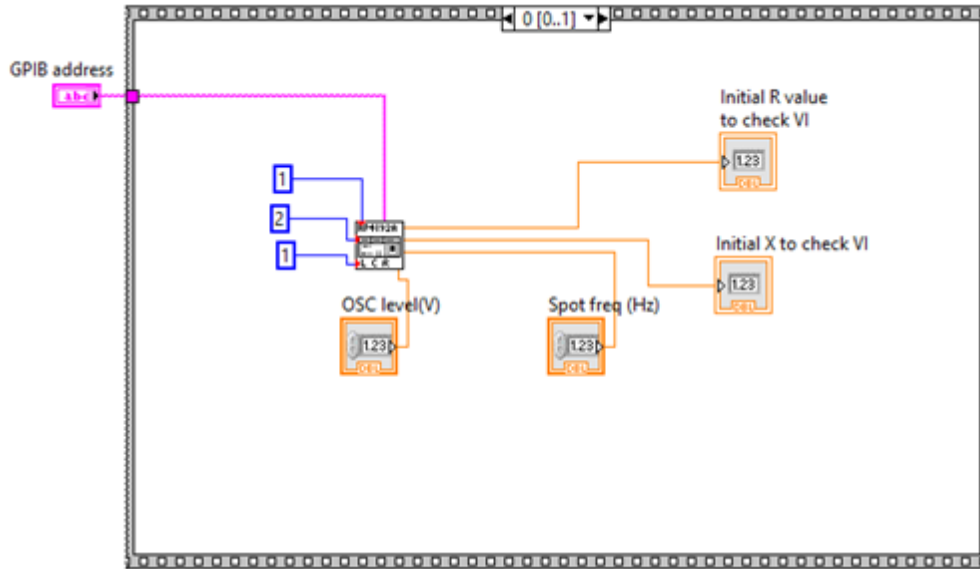
Figure 247: LabVIEW Code designed to communicate with Impedance Analyzer HP4192A and record the initial resistance value.
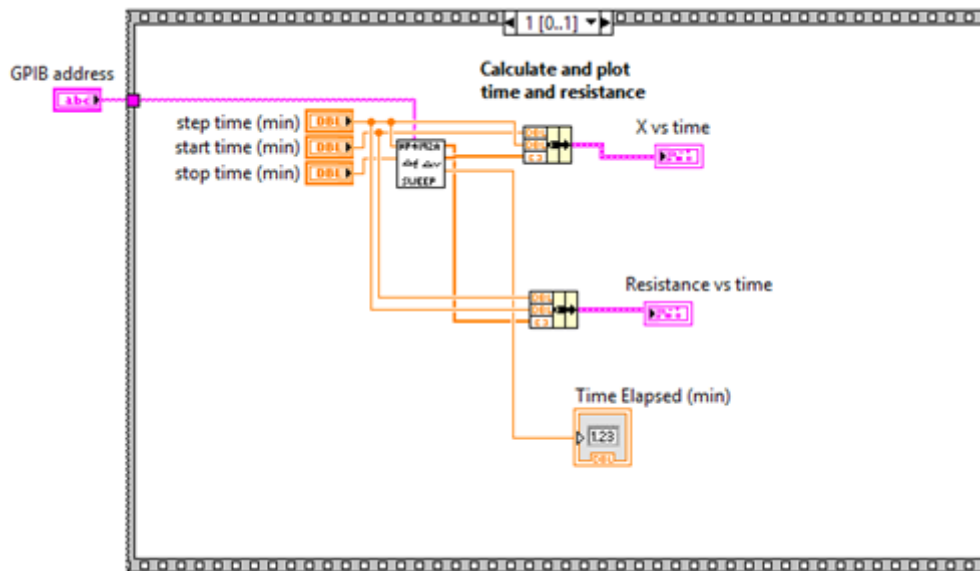
Figure 28: LabVIEW Code to display the time course of resistance property in TX-151 gel phantoms.
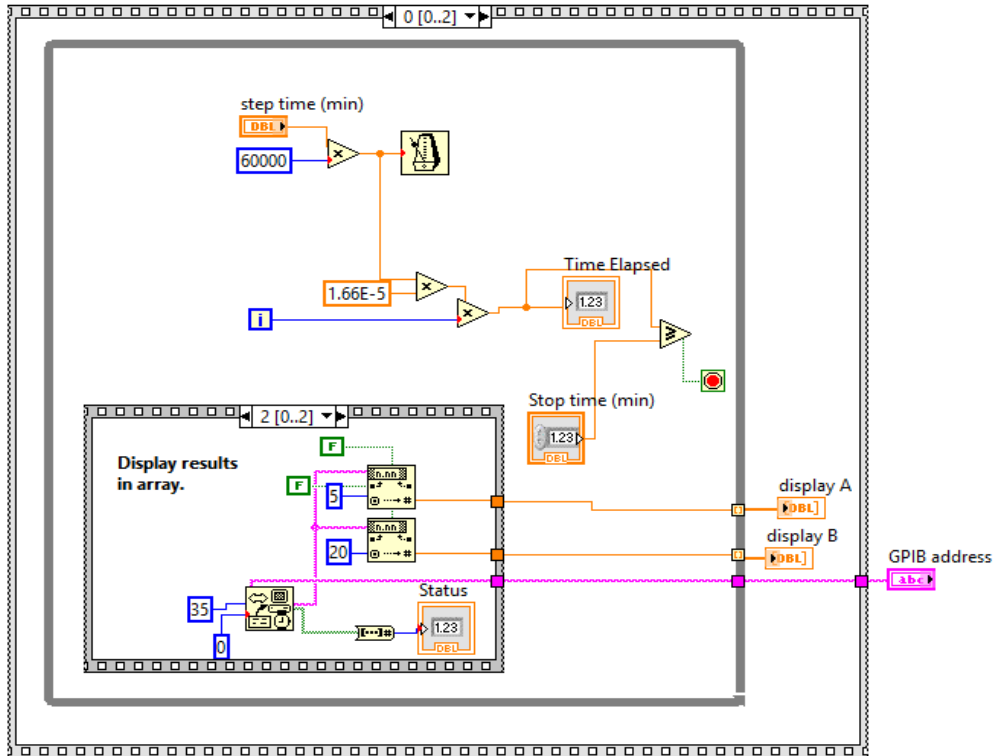
Figure 29: LabVIEW Code to read the resistance of TX-151 phantoms at time intervals of 5 minutes over a total duration of 4 hours.
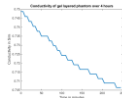
Figure 30: Conductivity of phantom with alternating high and low conductivity gel layers arranged parallel to the orientation of electrodes calculated from the impedance recorded by HP4192A over 4 hours.

APPENDIX C

RAW DATA COLLECTED FROM BRUKER, BIOSPIN 7 T

```matlab
% Enter the names of scan folders collected during horizontal and vertical positive and
negative current injections synchronized with a spin-echo sequence.

% [Horizontal+, Horizontal-, Vertical+, Vertical-]
foldername=[10 11 8 9];

% Read the acquired parameters (acqp)file
for n=1:4
    fname=foldername(n);
filename=([num2str(fname),'/acqp']);
param = fopen(filename,'rb');
if param == -1, error('File Read Error'), end

% Read acqp file line-by-line
tline = fgetl(param);
i = 1;

while ischar(tline)
    temp = strfind(tline,'=');
    hdr{i,1} = tline(3:temp-1);
    hdr{i,2} = tline(temp+1:end);

    if isempty(temp) == 1
        hdr{i,1} = tline;
        hdr{i,2} = [];
    end

    i = i+1;
    tline = fgetl(param);
end
fclose(param);
for index = 1:size(hdr,1),
    if(length(hdr{index,1}) == length('$ACQ_time_points'))
        if(hdr{index,1} == '$ACQ_time_points')
            nRepetitions = str2num(hdr{index,2})
        end
    end
    if(length(hdr{index,1}) == length('$ACQ_size'))
        if(hdr{index,1} == '$ACQ_size')
            xy_dim = str2num(hdr{index+1,1})
        end
    end
    if(length(hdr{index,1}) == length('$NSLICES'))
        if(hdr{index,1} == '$NSLICES')
            n_slices = str2num(hdr{index,2})
```

71

```
        end
    end
    if(length(hdr{index,1}) == length('$ACQ_obj_order'))
        if(hdr{index,1} == '$ACQ_obj_order')
            slice_order = str2num(hdr{index+1,1})+1
        end
    end
    if(length(hdr{index,1}) == length('$ACQ_echo_time'))
        if(hdr{index,1} == '$ACQ_echo_time')
            TE = str2num(hdr{index+1,1})
        end
    end
    if(length(hdr{index,1}) == length('$ACQ_slice_thick'))
        if(hdr{index,1} == '$ACQ_slice_thick')
            slice_thick = str2num(hdr{index,2})
        end
    end
    if(length(hdr{index,1}) == length('$ACQ_fov'))
        if(hdr{index,1} == '$ACQ_fov')
            FOV = str2num(hdr{index+1,1});
            FOV = FOV([2,1])
        end
    end
end

%% Read FID file
% Linux workstation uses little-endian byte ordering
fname=foldername(n);
fileid=([num2str(fname),'/fid']);
fid = fopen(fileid, 'r', 'ieee-le');
if fid == -1,
    fid = fopen('ser', 'r', 'ieee-le');
end

temp_d = fread(fid,'int32');
fclose(fid);

%% Make kspace
% Bruker automatically interleaves real & imaginary channels
temp = temp_d(1:2:end) + sqrt(-1)*temp_d(2:2:end);
kspace = reshape(temp,xy_dim(1)/2,n_slices, xy_dim(2),nRepetitions);
kspace = permute(kspace, [1 3 2 4]);
im = zeros(xy_dim(1)/2, xy_dim(2), n_slices, nRepetitions );

% FFTs
```

```
for rep = 1:nRepetitions,
    for slice = 1:n_slices
        im(:,:,slice,rep) = fftshift(ifftn(fftshift(squeeze(kspace(:,:,slice,rep))))).';
    end
end
im_f(:,:,:,n)=im(:,:,:);
end
Nslices=slice;
PE=xy_dim(1)/2;
clear kspace temp fileid filename fname foldername FOV hdr i n index n_slices
nRepetitions param rep slice_order slice_thick TE temp_d
clear tline xy_dim fid ans slice im;
```

APPENDIX D

PHASE UNWRAPPING AND Z-COMPONENT OF B_Z

```matlab
% Enter the experiment parameters
% im_f: Complex data
Exp=2; % Output folder (Exp_n)
CA=10;  % current amplitude [mA]
TC=16; % current injection time[ms]
c_d=[1 2];%%[H V]

for n=1:2

if (n<=1)
data_p(:,:,:)=im_f(:,:,:,1);
data_n(:,:,:)=im_f(:,:,:,2);
else
data_p(:,:,:)=im_f(:,:,:,3);
data_n(:,:,:)=im_f(:,:,:,4);
end
h_p=data_p;
h_n=data_n;
clear data_p data_n;
curr_direction=c_d(n);%Direction of current injection: 1 = Horizontal, 2 = Vertical

MR_img=abs(h_p);
slice_seq=[2 4 6 8 10 12 14 16 18 20 1 3 5 7 9 11 13 15 17 19 21];

%% PHASE SUBTRACTION
CurrentPhase=(h_p)./(h_n);

WPD=zeros(PE,PE,Nslices); UWPD=zeros(PE,PE,Nslices);
WBzD=zeros(PE,PE,Nslices); UWBzD=zeros(PE,PE,Nslices);

 for Ns=1:Nslices
 [WPhase,UWPhase,WBdata,UWBdata] =
fx_PhaseUnwrapping(CurrentPhase(:,:,Ns),PE,TC,curr_direction,CA);
 WPD(:,:,Ns)=WPhase;
 UWPD(:,:,Ns)=UWPhase;
 WBzD(:,:,Ns)=WBdata;
 UWBzD(:,:,Ns)=UWBdata;
 end
 clear WPhase; clear UWPhase; clear WBdata; clear UWBdata;

 for Ns=1:Nslices
       i=slice_seq(Ns);
       mag=MR_img(:,:,Ns);
       WP=WPD(:,:,Ns);
       UWP=UWPD(:,:,Ns);
```

```matlab
        WBz=WBzD(:,:,Ns);
        UWBz=UWBzD(:,:,Ns);


  end
clear WP; clear UWP; clear WBz; clear UWBz;
clear WPD; clear UWPD; clear WBzD; clear UWBzD;
end

%% Display the results
for i=1:21
   if (i<10)
   MRi=(['EXP_',num2str(Exp),'\Data\MR\00',num2str(i),'.mri']);
   MR(:,:,:,i)=load(MRi);
   else
   MRi=(['EXP_',num2str(Exp),'\Data\MR\0',num2str(i),'.mri']);
   MR(:,:,:,i)=load(MRi);
   end
end
figure;montage(MR,[0 10],'size',[5 5]);colorbar;title(['Magnitude']);

for i=1:21
   if i<10
   Bz1i=(['EXP_',num2str(Exp),'\Data\UWBz\00',num2str(i),'.bz1']);
   Bz1(:,:,:,i)=load(Bz1i);
   else
   Bz1i=(['EXP_',num2str(Exp),'\Data\UWBz\0',num2str(i),'.bz1']);
   Bz1(:,:,:,i)=load(Bz1i);
   end
end

figure;montage(Bz1,0.1*[-1e-6  1e-6],'size',[5 5]); colorbar;title(['Horizontal Bz']);

for i=1:21
   if i<10
   Bz2i=(['EXP_',num2str(Exp),'\Data\UWBz\00',num2str(i),'.bz2']);
   Bz2(:,:,:,i)=load(Bz2i);
   else
   Bz2i=(['EXP_',num2str(Exp),'\Data\UWBz\0',num2str(i),'.bz2']);
   Bz2(:,:,:,i)=load(Bz2i);
   end
end
figure; montage(Bz2,0.1*[-1e-6  1e-6],'size',[5 5]);colorbar;title(['Vertical Bz']);
```

```
function [WPhase,UWPhase,WBdata,UWBdata] =
fx_PhaseUnwrapping(phaseTemp,imSize,TC,curr,CA)


beforePhaseUnwrap=angle(phaseTemp); %wrapping phase data


fid=fopen('phaseWrap.bin','wb');
fwrite(fid,beforePhaseUnwrap,'float32');
fclose(fid);

if imSize==64;
   doscmd=['gold -input phaseWrap.bin -format '  ...
      'float -output phaseUnwrap.bin -xsize 64 -ysize 64 -dipole yes'];
   dos(doscmd,'-echo');
   fid=fopen('phaseUnwrap.bin','rb');
   [phaseData,cnt]=fread(fid,[64,64],'float32');
   fclose(fid);
elseif imSize==128;
   doscmd=['gold -input phaseWrap.bin -format '  ...
      'float -output phaseUnwrap.bin -xsize 128 -ysize 128 -dipole yes'];
   dos(doscmd,'-echo');
   fid=fopen('phaseUnwrap.bin','rb');
   [phaseData,cnt]=fread(fid,[128,128],'float32');
   fclose(fid);
elseif imSize==256;
   doscmd=['gold -input phaseWrap.bin -format '  ...
      'float -output phaseUnwrap.bin -xsize 256 -ysize 256 -dipole yes'];
   dos(doscmd,'-echo');
   fid=fopen('phaseUnwrap.bin','rb');
   [phaseData,cnt]=fread(fid,[256,256],'float32');
   fclose(fid);
end

delete('phasewrap.bin');
delete('phaseUnwrap.bin');
delete('phaseUnwrap.bin.brc');
delete('phaseUnwrap.bin.res');

% Change of offset
if (curr == 1)
   g1 = phaseData((imSize/2-imSize/16)+1:(imSize/2+imSize/16)-1,imSize/2);
else
```

```
    g1 = phaseData(imSize/2,(imSize/2-imSize/16)+1:(imSize/2+imSize/16)-1);
end
g2 = round(2*g1./pi); % get the integer multiple of "pi"
g3 = mean(g2);
phaseData = phaseData - g3*(pi/2);

WPhase=beforePhaseUnwrap;
UWPhase = phaseData;
WBdata=WPhase./(2*2*pi*42.57*10^6*TC*0.001*CA);
UWBdata=UWPhase./(2*2*pi*42.57*10^6*TC*0.001*CA);
```

APPENDIX E

SPATIAL DERIVATIVES OF CONDUCTIVITY PROFILE

```matlab
%% This code computes the first and second derivatives of sigma using the inverse
problem.

%Inputs: Bz1 - Bz during the horizontal current injection pair
%Bz2 - Bz during the vertical current injection pair
%BW - Binary mask of the phantom from magnitude MR image

load 'Bz1.mat'
load 'Bz2.mat'
load 'BW.mat'

BW = double(BW);

bz99_1=Bz1(:,:,1,11).*BW; bz100_1=Bz1(:,:,1,12).*BW; bz101_1=Bz1(:,:,1,13).*BW;

bz99_2  = Bz2(:,:,1,11).*BW; bz100_2 = Bz2(:,:,1,12).*BW;
bz101_2 = Bz2(:,:,1,13).*BW;

%% Image Parameters - imSize = Matrix size
fov_xy = 60; fov_z  = 42; imSize = 128; mu_0 = 4*pi*1E-7;
px_sz_x = fov_xy/imSize;

fov_xy = 0.001*fov_xy; ( % FOV in mm )
fov_z = 0.001*fov_z;

%% Find Laplacian of Bz
[lap_bz1_2d,lap_bz1_3d] =
laplacian_bz(fov_xy,fov_z,imSize,bz99_1,bz100_1,bz101_1);
[lap_bz2_2d,lap_bz2_3d] =
laplacian_bz(fov_xy,fov_z,imSize,bz99_2,bz100_2,bz101_2);

%% Find gradient of voltages
% Voltage distributions from COMSOL simulations
load('Voltage_center_H.mat')
load('Voltage_center_V.mat')

V_H = Voltage_center_H;
V_V = Voltage_center_V;

VR_H= imrotate(V_H,-3,'crop');
VR_V= imrotate(V_V,-3,'crop');
VDown_H = [zeros(12,128,3); VR_H];
VDown_V = [zeros(12,128,3); VR_V];
V_H = VDown_H(1:128,1:128,:);
V_V = VDown_V(1:128,1:128,:);
```

```
[u1_x, u1_y] = gradient(V_H);
u1_x = u1_x*(1\px_sz_x);
u1_y = u1_y*(1\px_sz_x);
[u2_x, u2_y] = gradient(V_V);
u2_x = u2_x*(1\px_sz_x);
u2_y = u2_y*(1\px_sz_x);

%% Find gradient conductivity
for k1=1:imSize
    k1
    for k2=1:imSize
        U = [ u1_y(k1,k2) -u1_x(k1,k2); u2_y(k1,k2) -u2_x(k1,k2) ];
        b = (1/mu_0)*[lap_bz1_3d(k1,k2); lap_bz2_3d(k1,k2)];
        if (det(U)==0)
            grad_sigma_x(k1,k2) = 0;
            grad_sigma_y(k1,k2) = 0;
        else
            lambda = 1/abs(det(U));

            grad_sigma = inv(U'*U + lambda*eye(size(U)))*U'*b;
            grad_sigma_x(k1,k2) = grad_sigma(1);
            grad_sigma_y(k1,k2) = grad_sigma(2);
        end
        check_cond((k1-1)*imSize+k2) = cond(U);
    end
end

%% Find Laplacian of "sigma"
lap_sigma_2d = laplacian_sigma(fov_xy,imSize,grad_sigma_x,grad_sigma_y);

% Laplacian sigma
clims=[-15e-16,15e-16];
figure;imagesc(lap_sigma_2d,clims);colorbar;title('Laplacian of sigma');
xlabel('Index: 1 to 128'); ylabel('Index: 1 to 128');

% Grad sigma
clims=[-15e-13,15e-13];
figure;imagesc(grad_sigma_x,clims);colorbar;title('Gradient of sigma in X');
xlabel('Index: 1 to 128'); ylabel('Index: 1 to 128');

clims=[-15e-13,15e-13];
figure;imagesc(grad_sigma_y,clims);colorbar;title('Gradient of sigma in Y');
xlabel('Index: 1 to 128'); ylabel('Index: 1 to 128');
```

```matlab
function [lap_bz_2d,lap_bz_3d] = laplacian_bz(fov_xy,fov_z,imSize,bz99,bz100,bz101)

%function [lap_bz_2d,lap_bz_3d] =
laplacian_bz(fov_xy,fov_z,imSize,bz99,bz100,bz101,cond100)

px_sz_x = fov_xy/imSize
px_sz_z = fov_z/200;

%% Three-dimensional Bz
bz1 = bz99;
bz2 = bz100;
bz3 = bz101;

%% Three-dimensional Laplacian of Bz
hx = [1 -2 1];
hy = hx';
lap_bz_x = conv2(bz2,hx,'same');
lap_bz_y = conv2(bz2,hy,'same');
lap_bz_z = bz1 + bz3 - 2*bz2;
lap_bz_xy = lap_bz_x + lap_bz_y;
lap_bz_xyz = lap_bz_xy + lap_bz_z;
lap_bz_2d = (1/px_sz_x^2)*lap_bz_xy;
lap_bz_3d = lap_bz_2d + (1/px_sz_z^2)*lap_bz_z;

figure(),imagesc(lap_bz_2d),colormap(gray);title('Laplacian of Bz in 2d for vertical
current injection');
figure(),imagesc(lap_bz_3d),colormap(gray);title('Laplacian of Bz in 3d for horizontal
current injection');
```

APPENDIX   F

DIFFUSION TENSOR ANALYSIS

```
%dti_FA : Fractional anisotropy
%dti_V1, dti_V2, dti_V3 : First, second and third eigenvectors
%dti_L1, dti_L2, dti_L3 : First, second and third eigenvalues
I = load_nii('dti_FA.nii.gz');
v1 = load_nii('dti_V1.nii.gz');
v2 = load_nii('dti_V2.nii.gz');
v3 = load_nii('dti_V3.nii.gz');
l1 = load_nii('dti_L1.nii.gz');
l2 = load_nii('dti_L2.nii.gz');
l3 = load_nii('dti_L3.nii.gz');

load('BinaryMask.mat')   % Binary mask from FA map
BW = double(BW);


IM = I.img; V1 = v1.img; V2 = v2.img; V3 = v3.img; L1 = l1.img; L2 = l2.img;
L3=l3.img;


%Fractional Anisotropy map of the central slice
FA_midSlice = IM(:,:,3);
figure, imagesc(FA_midSlice);title('FA'); colorbar;

L1_midSlice = L1(:,:,3).*BW; [L1_Idx1, L1_Idx2] = find(L1_midSlice);
L2_midSlice = L2(:,:,3).*BW; [L2_Idx1, L2_Idx2] = find(L2_midSlice);
L3_midSlice = L3(:,:,3).*BW; [L3_Idx1, L3_Idx2] = find(L3_midSlice);

BW_rep = repmat(BW,[1,1,5]);
BW_equal = isequal(BW_rep(:,:,1),BW_rep(:,:,3));

V1_midSlice = squeeze(double(V1(:,:,3,:))).*BW_rep(:,:,1:3);
V2_midSlice = squeeze(double(V2(:,:,3,:))).*BW_rep(:,:,1:3);
V3_midSlice = squeeze(double(V3(:,:,3,:))).*BW_rep(:,:,1:3);

%% Comparing each component  of V1 with V2

V1red = V1_midSlice(:,:,1);
V1green = V1_midSlice(:,:,2);
V1blue = V1_midSlice(:,:,3);

V2red = V2_midSlice(:,:,1);
V2green = V2_midSlice(:,:,2);
V2blue = V2_midSlice(:,:,3);

V3red = V3_midSlice(:,:,1);
V3green = V3_midSlice(:,:,2);
V3blue = V3_midSlice(:,:,3);
```

```
% Eigenvalues
figure, imagesc(L1_midSlice,[-3e-3,3e-3]); title('Lambda1'); colorbar;
figure, imagesc(L2_midSlice,[-3e-3,3e-3]); title('Lambda2');colorbar;
figure, imagesc(L3_midSlice,[-3e-3,3e-3]); title('Lambda3');colorbar;


% Eigenvectors
figure, imagesc(V1_midSlice);title('V1');
figure, imagesc(V2_midSlice);title('V2');
figure, imagesc(V3_midSlice);title('V3');

% Components of V1
figure, imagesc(V1red); title('x component of V1');colorbar;
figure, imagesc(V1green); title('y component of V1');colorbar;
figure, imagesc(V1blue); title('z component of V1');colorbar;

% Components of V2
figure, imagesc(V2red); title('x component of V2');colorbar;
figure, imagesc(V2green); title('y component of V2');colorbar;
figure, imagesc(V2blue); title('z component of V2');colorbar;

% Components of V3
figure, imagesc(V3red); title('x component of V3');colorbar;
figure, imagesc(V3green); title('y component of V3');colorbar;
figure, imagesc(V3blue); title('z component of V3');colorbar;

% Eigenvalues*Components of V1  (lambda1*V1)

LV1red = double(L1_midSlice).*V1red; figure, imagesc(LV1red.*BW,[-2.5e-3,2.5e-3]);
title('Lambda1*V1red');colorbar;

LV1green = double(L1_midSlice).*V1green; figure, imagesc(LV1green.*BW,[-2.5e-
3,2.5e-3]); title('Lambda1*V1green'); colorbar;

LV1blue = double(L1_midSlice).*V1blue; figure, imagesc(LV1blue.*BW,[-2.5e-3,2.5e-
3]); title('Lambda1*V1blue'); colorbar;

% Eigenvalues*Components of V2  (lambda2*V2)

LV2red = double(L2_midSlice).*V2red; figure, imagesc(LV2red.*BW,[-2.5e-3,2.5e-3]);
title('Lambda2*V2red');colorbar;

LV2green = double(L2_midSlice).*V2green; figure, imagesc(LV2green.*BW,[-2.5e-
3,2.5e-3]); title('Lambda2*V2green'); colorbar;
```

```
LV2blue = double(L2_midSlice).*V2blue; figure, imagesc(LV2blue.*BW,[-2.5e-3,2.5e-
3]); title('Lambda2*V2blue'); colorbar;

% Eigenvalues * Components of V3  (lambda3*V3)

LV3red = double(L3_midSlice).*V3red; figure, imagesc(LV3red.*BW,[-2.5e-3,2.5e-3]);
title('Lambda3*V3red');colorbar;

LV3green = double(L3_midSlice).*V3green; figure, imagesc(LV3green.*BW,[-2.5e-
3,2.5e-3]); title('Lambda3*V3green'); colorbar;

LV3blue = double(L3_midSlice).*V3blue; figure, imagesc(LV3blue.*BW,[-2.5e-3,2.5e-
3]); title('Lambda3*V3blue'); colorbar;

% Ratio and average of eigenvalues

L12 = (L1_midSlice./L2_midSlice).*BW;
figure, imagesc(L12,[0,5]); title('lambda1/lambda2');colorbar;

L23 = (L2_midSlice./L3_midSlice).*BW;
figure, imagesc(L23,[0,5]); title('lambda2/lambda3');colorbar;

L13 = (L1_midSlice./L3_midSlice).*BW;
figure, imagesc(L13,[0,5]); title('lambda1/lambda3');colorbar;

L123 = (L1_midSlice./((L2_midSlice+L3_midSlice)/2)).*BW;
L123(find(isnan(L123)))=0;
figure, imagesc(L123,[0,5]); title('lambda1/((lambda2+lambda3)/2)'); colorbar;

L1_Avg = sum(sum(L1_midSlice))/size(L1_Idx1,1)
L2_Avg = sum(sum(L2_midSlice))/size(L2_Idx1,1)
L3_Avg = sum(sum(L3_midSlice))/size(L3_Idx1,1)
L123_Avg = L1_Avg/((L2_Avg+L3_Avg)/2)

% Mean Standard errors of components of V1

V1red_avg = sum(sum(V1red))/size(L1_Idx1,1)
V1green_avg = sum(sum(V1green))/size(L1_Idx1,1)
V1blue_avg = sum(sum(V1blue))/size(L1_Idx1,1)

V1Dev = (V1red-V1red_avg).*BW;
V1red_sd = ((sum(sum(V1Dev.^2)))/(size(L1_Idx1,1)-1))^0.5;
V1red_SE = V1red_sd/sqrt(size(L1_Idx1,1))
```

V1Dev = (V1green-V1green_avg).*BW;
V1green_sd = ((sum(sum((V1Dev.^2))))/(size(L1_Idx1,1)-1))^0.5;
V1green_SE = V1green_sd/sqrt(size(L1_Idx1,1))

V1Dev = (V1blue-V1blue_avg).*BW;
V1blue_sd = ((sum(sum((V1Dev.^2))))/(size(L1_Idx1,1)-1))^0.5;
V1blue_SE = V1blue_sd/sqrt(size(L1_Idx1,1))

%Mean Standard errors of components of V2

V2red_avg = sum(sum(V2red))/size(L2_Idx1,1)
V2green_avg = sum(sum(V2green))/size(L2_Idx1,1)
V2blue_avg = sum(sum(V2blue))/size(L2_Idx1,1)

V2Dev = (V2red-V2red_avg).*BW;
V2red_sd = ((sum(sum((V2Dev.^2))))/(size(L2_Idx1,1)-1))^0.5;
V2red_SE = V2red_sd/sqrt(size(L2_Idx1,1))

V2Dev = (V2green-V2green_avg).*BW;
V2green_sd = ((sum(sum((V2Dev.^2))))/(size(L2_Idx1,1)-1))^0.5;
V2green_SE = V2green_sd/sqrt(size(L2_Idx1,1))

V2Dev = (V2blue-V2blue_avg).*BW;
V2blue_sd = ((sum(sum((V2Dev.^2))))/(size(L2_Idx1,1)-1))^0.5;
V2blue_SE = V2blue_sd/sqrt(size(L2_Idx1,1))

%Mean Standard errors of components of V3

V3red_avg = sum(sum(V3red))/size(L3_Idx1,1)
V3green_avg = sum(sum(V3green))/size(L3_Idx1,1)
V3blue_avg = sum(sum(V3blue))/size(L3_Idx1,1)

V3Dev = (V3red-V3red_avg).*BW;
V3red_sd = ((sum(sum((V3Dev.^2))))/(size(L3_Idx1,1)-1))^0.5;
V3red_SE = V3red_sd/sqrt(size(L3_Idx1,1))

V3Dev = (V3green-V3green_avg).*BW;
V3green_sd = ((sum(sum((V3Dev.^2))))/(size(L3_Idx1,1)-1))^0.5;
V3green_SE = V3green_sd/sqrt(size(L3_Idx1,1))

V3Dev = (V3blue-V3blue_avg).*BW;
V3blue_sd = ((sum(sum((V3Dev.^2))))/(size(L3_Idx1,1)-1))^0.5;
V3blue_SE = V3blue_sd/sqrt(size(L3_Idx1,1))

APPENDIX G

CONSTANT CURRENT SOURCE - POSITIVE AND NEGATIVE INJECTIONS

Positive script

```
sequence
loop    10000
dummy       0
      loop  1
%             Trig   delay  TC      Amp  Ch1    Ch2   Ch3    Ch4
              FALL  0       8       10mA SOURCE     SINK  NONE NONE
              FALL  0       8       -10mA SOURCE    SINK  NONE NONE
      stop

stop

endÿ
```

Negative script

```
sequence
loop    10000
dummy       0
      loop  1
%             Trig   delay  TC      Amp  Ch1    Ch2   Ch3    Ch4
              FALL  0       8       -10mA SOURCE    SINK  NONE NONE
              FALL  0       8       10mA SOURCE     SINK  NONE NONE
      stop

stop

endÿ
```

APPENDIX H

B$_Z$ FROM TRANSVERSAL CURRENT DENSITY BY BIOT-SAVART LAW

```c
#include <matrix.h>
#include <math.h>
#include <mex.h>
#include "/sw/include/fftw3.h"

#define PI (4.0*atan(1.0))

/* Definitions to keep compatibility with earlier versions of ML */
#ifndef MWSIZE_MAX
typedef int mwSize;
typedef int mwIndex;
typedef int mwSignedIndex;

/// gradient of fundamental solution
double grad_Fundamental_Solution(double x, double y, double z, int opt);

#if (defined(_LP64) || defined(_WIN64)) && !defined(MX_COMPAT_32)
/* Currently 2^48 based on hardware limitations */
# define MWSIZE_MAX    281474976710655UL
# define MWINDEX_MAX   281474976710655UL
# define MWSINDEX_MAX  281474976710655L
# define MWSINDEX_MIN -281474976710655L
#else
# define MWSIZE_MAX    2147483647UL
# define MWINDEX_MAX   2147483647UL
# define MWSINDEX_MAX  2147483647L
# define MWSINDEX_MIN -2147483647L
#endif
#define MWSIZE_MIN    0UL
#define MWINDEX_MIN   0UL
#endif

void mexFunction(int nlhs, mxArray *plhs[], int nrhs, const mxArray *prhs[])
{

void Kernel_Convolution(double *i_data, double *o_data, int cx, double fov_x, int cy,
double fov_y, int cz, double fov_z, int opt);

   double perm=4.0*PI*1.0e-7;

//declare variables
   mxArray *a_in_m, *b_in_m, *fov_in, *fov_z_in, *opt_in, *Bz_out_m;
   double *conv_x, *conv_y;
   double fov, fov_z;
   int opt;
```

```
    const mwSize *dims;
    double *Jx, *Jy, *Bz;
    int dimx, dimy, dimz, numdims;
    int sz, sz_z;
    double sumx, sumy,sumBz;
    int size[3]={0, 0, 0};
    int p,q,r;

//associate inputs
    mexPrintf("Hello from Bzconv\n");
    a_in_m = mxDuplicateArray(prhs[0]);
    b_in_m = mxDuplicateArray(prhs[1]);
    fov_in = mxDuplicateArray(prhs[2]);
    fov_z_in = mxDuplicateArray(prhs[3]);

//figure out dimensions
    dims = mxGetDimensions(prhs[0]);
    numdims = mxGetNumberOfDimensions(prhs[0]);
    dimy = (int)dims[0]; dimx = (int)dims[1]; dimz=(int)dims[2];
    mexPrintf("dimx is %d, dimy is %d, dimz is %d\n",dimx, dimy, dimz);
    sz=dimx;sz_z=dimz;
    size[0]=dimx;size[1]=dimy;size[2]=dimz;
//associate outputs
    Bz_out_m = plhs[0] = mxCreateNumericArray(3, size, mxDOUBLE_CLASS,
mxREAL);
    conv_x = (double *)mxCalloc(sz*sz*sz_z,sizeof(double));
    conv_y = (double *)mxCalloc(sz*sz*sz_z,sizeof(double));

//associate pointers
    Jx = mxGetPr(a_in_m);
    Jy = mxGetPr(b_in_m);
    Bz = mxGetPr(Bz_out_m);
    fov = (double)mxGetScalar(fov_in);
    fov_z = (double)mxGetScalar(fov_z_in);
    mexPrintf("fov is %f\t, fov_z is %f\n",fov, fov_z);

    mexPrintf("Doing FFT convolution\n");

    Kernel_Convolution(Jx, conv_x, sz, fov, sz, fov, sz_z, fov_z,2);
    Kernel_Convolution(Jy, conv_y, sz, fov, sz, fov, sz_z, fov_z,1);

    mexPrintf("FFT convolution done\n");
    sumx=0.0;sumy=0.0;sumBz=0.0;

  for (r=0;r<sz_z;r++)
```

```
  {
   for (q=0;q<sz;q++)  {
      for (p=0;p<sz;p++)  {
      Bz[r*sz*sz+p*sz+q]=(-conv_y[sz*sz*r+p*sz+(sz-1-q)] +
conv_x[r*sz*sz+p*sz+(sz-1-q)])*perm;
      //Bz[r*sz*sz+p*sz+q]=1.0;
      }
   }
   }


  for (r=0;r<sz_z;r++)
  {
   for (q=0;q<sz;q++)  {
      for (p=0;p<sz;p++)  {
      sumx+=conv_x[r*sz*sz+p*sz+q];
      sumy+=conv_y[r*sz*sz+p*sz+q];
      sumBz+=Bz[r*sz*sz+p*sz+q];
      }
   }
   }

  //mexPrintf("sumx is %e\t, sumy is %e\tsumBz is %e\n");
  mxDestroyArray(a_in_m);
  mxDestroyArray(b_in_m);
  mxDestroyArray(fov_in);
  mxDestroyArray(fov_z_in);

  return;
}

double grad_Fundamental_Solution(double x, double y, double z, int opt)
{
     double rad = pow(sqrt(x*x + y*y + z*z), 3);
     double rv = 0.0;

     if(rad != 0.0)
     {
          switch(opt)
          {
               case 1: rv = x/rad; break;
               case 2: rv = y/rad; break;
               case 3: rv = z/rad; break;
          }
     }
```

```
        return rv/(4.0*PI);
}

void Kernel_Convolution(double *i_data, double *o_data, int cx, double fov_x, int cy,
double fov_y, int cz, double fov_z, int opt)
{

        fftw_complex *in, *out_K, *out_D;
        fftw_plan plan;

        int winx, winy, winz;
        winx = cx*2;
        winy = cy*2;
        winz = cz*2;


        double dh_x = fov_x/(double)cx;
        double dh_y = fov_y/(double)cx;
        double dh_z = fov_z/(double)cz;

        long int msz = sizeof(fftw_complex)*winx*winy*winz;
        int pos;

        in  = (fftw_complex*) fftw_malloc(msz);
        if (in==NULL)   {
           mexPrintf("Stuffed, not enough memory\n");
           mexPrintf("size requested is %li, memory available is \n",msz);
           return;
        }

        /// kernel data ...
        int p, q, r;
        double px, qy, rz;

        for(p=0 ; p<winx ; p++){
             for(q=0 ; q<winy ; q++){
                   for(r=0 ; r<winz ; r++){

                        /// physical position
                        px = (p-winx/2.0)*dh_x;
                        qy = (q-winy/2.0)*dh_y;
                        rz = (r-winz/2.0)*dh_z;

                        pos = r + winz*(q + winy*p);
```

```
                in[pos][0] = grad_Fundamental_Solution(px, qy, rz, opt);
          //mexPrintf("in[%i] is %e\n",pos,in[pos][0]);
                in[pos][1] = 0.0;
            }
        }
    }

    /// out allocation
    out_K = (fftw_complex*) fftw_malloc(msz);
    plan = fftw_plan_dft_3d(winx, winy, winz, in, out_K, -1, FFTW_ESTIMATE);

    fftw_execute(plan);
    fftw_destroy_plan(plan);


    /// data fft ...
     /// initialize
   for(p=0 ; p<winx ; p++){
        for(q=0 ; q<winy ; q++){
            for(r=0 ; r<winz ; r++){

                pos = r + winz*(q + winy*p);

                in[pos][0] = 0.0;
                in[pos][1] = 0.0;
            }
        }
    }

    for(p=0 ; p<cx ; p++){
        for(q=0 ; q<cy ; q++){
            for(r=0 ; r<cz ; r++){

                in[r+cz/2 + winz*(q+cy/2 + winy*(p+cx/2))][0] =
i_data[p*cy+q+r*cx*cy];
            }
        }
    }

    /// out allocation
    out_D = (fftw_complex*) fftw_malloc(msz);

    plan = fftw_plan_dft_3d(winx, winy, winz, in, out_D, -1, FFTW_ESTIMATE);
    fftw_execute(plan);
```

95

```
        fftw_destroy_plan(plan);


    /// multiply signal data and inverse fft ..

    for(p=0 ; p<winx ; p++){
        for(q=0 ; q<winy ; q++){
            for(r=0 ; r<winz ; r++){

                pos = r + winz*(q + winy*p);
                /// real part
                in[pos][0] = (out_K[pos][0]*out_D[pos][0] -
out_K[pos][1]*out_D[pos][1])*pow(-1.0, p+q+r) ;
        //mexPrintf("in[%i] is %e\n",pos,in[pos][0]);
                /// imaginary part
                in[pos][1] = (out_K[pos][0]*out_D[pos][1] +
out_K[pos][1]*out_D[pos][0])*pow(-1.0, p+q+r);
        //mexPrintf("in[%i] is %e\n",pos,in[pos][1]);
            }
        }
    }

    /// data de-allocate
    fftw_free(out_K);

   plan = fftw_plan_dft_3d(winx, winy, winz, in, out_D, 1, FFTW_ESTIMATE);
    fftw_execute(plan);
    fftw_destroy_plan(plan);


    double scale_constant = 1.0/(double)(winx*winy*winz);
    double sc = dh_x*dh_y*dh_z;
  double sumo=0.0;

    for(p=0 ; p<cx ; p++){
        for(q=0 ; q<cy ; q++){
            for(r=0 ; r<cz ; r++){

                o_data[p*cy+q+r*cx*cy] = out_D[r+cz/2 + winz*(q+cy/2 +
winy*(p+cx/2))][0]*scale_constant*sc;
        sumo+=out_D[r+cz/2+winz*(q+cy/2+winy*(p+cx/2))][0]*scale_constant*sc;
        //mexPrintf("o_data[%i] is %e\n",p+cy*q+r*cx*cy,o_data[p+cy*q+r*cx*cy]);
            }
        }
    }
```

```
    mexPrintf("o_data sum is %e\n",sumo);

        fftw_free(out_D);
        fftw_free(in);
}
```