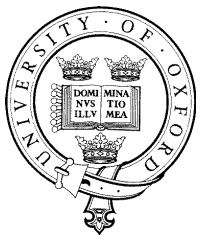
FMRI Simulator: Development and Applications

Ivana Drobnjak



Hilary Term, 2007

Computing Laboratory

and

FMRIB Centre, Department of Clinical Neurology

University of Oxford

To my family

Mami, tati, bati, seki, baki, deki i ujki ...

FMRI Simulator: Development and Applications

Ivana Drobnjak

Linacre College
University of Oxford

Abstract

Functional Magnetic Resonance Imaging (FMRI) is a non-invasive method of imaging brain function in-vivo. However, images produced in FMRI experiment almost invariably contain imperfections, known as artifacts. These artifacts can result from, for example, rigid-body motion of the head, magnetic field inhomogeneities, chemical shift and eddy currents.

To investigate these artifacts, with the eventual aim of minimising or removing them completely, a computational model of the FMR image acquisition process was built which can simulate all of the above mentioned artifacts. The simulator uses a geometric definition of the object (brain), Bloch equations (to model the behaviour of the magnetisation) and a model for the Blood Oxygen Level Dependent (BOLD) activations. Furthermore, it simulates rigid-body motion of the object by solving Bloch equations for an object moving continuously in time (as opposed to assuming movement only between the acquisition of consecutive images). This is a novel approach in the area of MRI computer simulations. With this approach it is possible, in a controlled and precise way, to simulate the full effects of various rigid-body motion artifacts in FMRI data (e.g. spin-history effects, B_0-motion interaction and within-scan motion blurring) and therefore formulate and test algorithms for their reduction. This thesis presents the development of the model for the simulator, its numerical implementation and solutions for the computational issues, and the validation of the simulator by comparing its outputs with existing theoretical and experimental results.

Finally, the simulator is applied in a number of diverse applications. These applications include: comparing different acquisition techniques for eddy-current compensation; reproducing and extending experiments in neuronal current imaging; quantifying the performance of motion correction software; quantitatively evaluating the impact of stimulus correlated motion artifacts; and investigating the performance of Independent Component Analysis (ICA) as a tool for quantifying motion-related artifacts.

Word count: 45000

Thesis submitted for the degree of Doctor of Philosophy
at the University of Oxford

Hilary Term, 2007

Acknowledgements

First of all I would like to thank my supervisor "up in FMRIB" Mark Jenkinson for his constant support, endless scientific skills, invaluable guidance, all the things I learned from him, the tireless reading of this thesis, and above all for his goodness and the "airport-bus-viva" trip to Oxford. I would also like to thank my supervisors "down in COMLAB": David Gavaghan for his constant support, endless enthusiasm, and his belief in me for a very long time which led to me starting this project in FMRIB; Endre Süli for his constant support, the constantly open door of his COMLAB office and his belief in me even before we met which led to me coming to Oxford.

I would also like to highlight the contribution made by my insightfull examiners, Professor Sir Mike Brady and Professor Richard Bowtell, whose comments have contributed greatly to the final form of this thesis.

The research summarised here was funded by an MIAS/IRC research grant. Financial support is also gratefully acknowledged from the Computing Laboratory, ISMRM society and Linacre College.

Special thanks go to my computing clusters providers: Cubric Centre in Cardiff, Oxford Supercomputing Centre and Robotics Research Laboratory in Oxford, who helped enormously with generating my results.

I would also like to thank FMRIB and its members. Never before have I worked in such a highly skilled, hard working but equally friendly and enchanting environment. Steve Smith for being a friend/boss from my first day in FMRIB, and for the food on the roof. Tim Behrens for his willingness to help at any given occasion, and for his driving skills. Heidi Johansen-Berg for helping me with my first HBM abstract, numerous advices, her thesis and for her wedding. Dave,

Duncan and Ian for their endless computing support. Christian for having an answer to everything. Karla for the MR knowledge. Marylin for being a heart. Salima for the car-support. Dan G for brainstormings. Sean for the viva help. Jen for the runs. Natalie for the soups, flowers and care. Rami, Brian, Morgan, Gwen, Saad, Katie, Antonio, Jesper, Mark W, Sue, Hannah, Stephan and all other members of FMRIB for being wonderful people to work with.

Special thanks go to Stuart Clare for his help in providing me with all the scanner information so that I can develop my own pulse sequence generator. He was also there as a constant fountain of MR physics knowledge and probably would have preferred some times if my office was kilometers away.

Thanks go to Matt Robson who gave many valuable comments, generated a FLASH sequence for me and gave me a box full of glass spheres which I used as my phantoms. Peter Jezzard always had quick and reliable answers regarding various MRI related issues. Rita Nunes gave me some nice images of eddy currents.

Many thanks goes to Sarah Jenkinson who was supplying me with books, food and most importantly with Hercule Poirot, who maintained my sanity.

I would like to thank my best friend Christina for being my source of optimism, joy and strength during this time. In some moments during our DPhil trip, for both personal and scientific reasons, only her huge optimism, the love of life, and the enormous capacity for friendship kept me going. I can truly call her my sister.

A special thank you goes to my boyfriend Richard for finding energy and time to offer help in proof reading and discussing my thesis on early Saturday mornings, and more generally for being a constant source of support, love, and music in my life.

The biggest thank you of all goes to my family. To my uncle Slavko, an ingenious engineer who was taken for cancer last year, for his love and his friendship. To my brother Igor and my sister Jelena for always being there for me. And finally, to my parents Milenko and Ljuba, for their unconditional love and for their faith in me since the day I was born. It is to my family that I dedicate this thesis to.

Related Publications

Journal Papers

I. Drobnjak, D. Gavaghan, E. Süli, J. Pitt-Francis, M. Jenkinson. Development of a FMRI simulator for modelling realistic rigid-body motion artifacts. *Magnetic Resonance in Medicine*, 56(2), 364-380, August 2006.

S.M. Smith, M. Jenkinson, P.R. Bannister, C.F. Beckmann, T.E.J. Behrens, M. De Luca, **I. Drobnjak**, D. Flitney, H. Johnasen-Berg, R. Niazy, J. Saunders, J. Vickers, M.W. Woolrich, Y. Zhang, J.M. Brady, P.M. Matthews. Advances in Functional and Structural MR Image Analysis and Implementation as FSL. *NeuroImage*, 23, 208-219, 2004.

MIDAS Consortium: F. Boada, S. Clare, D.L. Collins, **I. Drobnjak**, A.C. Evans, D. Gavaghan, M.P. Griffin, M. Jenkinson, D. Noll, R. Nunes, G.B. Pike, H. Shi, D. Schroff, V.A. Stenger, E. Süli, K. Worsley. A Realistic Physics-Based Simulation Tool for FMRI: MIDAS. *NeuroImage*, 2006 (submitted).

Conference Abstracts

I. Drobnjak, C. Beckmann, M. Jenkinson. Quantification of motion-related artifacts in simulated FMRI data using ICA *In Proceedings of the International Society for Magnetic Resonance in Medicine 14th Scientific Meeting*, (Seattle, USA), 2006.

I. Drobnjak, D. Gavaghan, E. Süli, M. Jenkinson. Application of an FMRI simulator in modelling the spin history effects *In Proceedings of the 11th International conference for Functional Mapping of the Human Brain*, (Toronto, Canada), 2005.

R. Nunes, **I. Drobnjak**, S. Clare, P. Jezzard, M. Jenkinson. Quantitative Simulation of Affine Registration for Correction of Eddy Current Distortions in Diffusion-Weighted Images. *In Proceedings of the International Society for Magnetic Resonance in Medicine 13th Scientific Meeting*, (Miami, USA), 2005.

I. Drobnjak, M. Jenkinson. Modelling FMRI Motion Artifacts Using a Simulated Scanner. *In Proceedings of the International Society for Magnetic Resonance in Medicine 12th Scientific Meeting*, (Kyoto, Japan), 2004.

I. Drobnjak, M. Jenkinson. FMRI Simulator and its application in modelling the interaction of motion and B_0 inhomogeneities. *In Proceedings of the 10th International conference for Functional Mapping of the Human Brain*, (Budapest, Hungary), 2004.

The MIDAS Consortium: Boada, Collins, **Drobnjak**, Eddy, Evans, Griffin, Jenkinson, Noll, Pike, Shi, Shroff, Stenger, Worsley MIDAS-A Multi Site fMRI Simulator Consortium. *In Proceedings of the 10th International conference for Functional Mapping of the Human Brain*, (Budapest, Hungary), 2004.

Contents

Acknowledgements **iii**

Related Publications **v**

List of Figures **xiii**

List of Tables **xv**

List of Abbreviations **1**

1 Introduction **3**
1.1 Motivation . . . 3
1.2 Scope and objectives . . . 6
1.3 The outline of the thesis . . . 7

2 Principles of FMRI **11**
2.1 Physical principles of image generation . . . 12
2.1.1 Magnetisation Vector: $\boldsymbol{\mu} \rightarrow \mathbf{M}$ . . . 14
2.1.2 RF pulse: $\mathbf{M} \rightarrow \mathbf{M}_{xy}$ . . . 22
2.1.3 Signal Detection: $\mathbf{M_{xy}} \rightarrow S(t)$ . . . 25
2.1.4 k-space: $S(t) \rightarrow S(\mathbf{k})$ . . . 27
2.1.5 Image reconstruction: $S(\mathbf{k}) \rightarrow I(\mathbf{x})$ . . . 32
2.2 Imaging system . . . 32
2.2.1 Imaging hardware . . . 32
2.2.2 Pulse sequence . . . 34
2.2.3 Scanning parameters . . . 38
2.3 Physiological principles of BOLD imaging . . . 44
2.4 Imaging artefacts . . . 46
2.4.1 Magnetic-susceptibility-induced B_0 inhomogeneities . . . 46
2.4.2 Rigid-body motion . . . 47
2.4.3 Chemical shift . . . 50
2.4.4 Eddy Currents . . . 53

2.5 Summary 54

3 Simulator model development **55**
3.1 Existing simulation strategies 56
3.2 Signal Model 60
3.3 Magnetic-susceptibility-induced B_0 inhomogeneities 65
3.4 Motion Model 67
3.5 BOLD model 70
3.6 Signal evaluation 71
3.7 Additions 72
3.7.1 Chemical Shift 72
3.7.2 Eddy Currents 73
3.7.3 Noise 73
3.8 Summary 74

4 Simulator implementation **75**
4.1 Introduction 75
4.2 Inputs to the simulator 77
4.2.1 Object 78
4.2.2 The MR parameters 79
4.2.3 Pulse sequence 80
4.2.4 Slice profile 83
4.2.5 Motion sequence 84
4.2.6 B_0 inhomogeneities 86
4.2.7 Activation 87
4.3 Bloch equation solver and the signal 87
4.3.1 Noise 95
4.4 Image reconstruction 95
4.5 Computational time 96
4.5.1 Introduction 96
4.5.2 Computational speed ups 97
4.5.3 Parallelisation 100
4.5.4 Results 101
4.6 Summary 104

5 Simulator validation **105**
5.1 Introduction 105
5.2 Quantitative comparison of the simulator output with the theoretical results 108
5.2.1 Evaluation of the simulator performance for different object voxel sizes 108
5.2.2 Image contrast 113
5.2.3 B_0 inhomogeneities 114

5.3 Quantitative comparison of the simulation results with the real scan images . . . 118
5.3.1 Translation in z . . . 118
5.3.2 Rotation about the y-axis . . . 126
5.4 Qualitative comparison of the simulation results with real scan images . . . 136
5.5 Summary . . . 141

6 Simulator applications 143
6.1 Quantifying the performance of a motion correction algorithm . . . 143
6.1.1 Introduction . . . 144
6.1.2 MCFLIRT Algorithm . . . 147
6.1.3 Methods . . . 149
6.1.4 Results and discussion . . . 158
6.1.5 Conclusions . . . 168
6.2 Quantifying motion artefacts with ICA . . . 169
6.2.1 Introduction . . . 169
6.2.2 Methods . . . 169
6.2.3 Results and discussion . . . 171
6.2.4 Conclusions . . . 173
6.3 Evaluating Stimulus-Correlated Motion Artifacts . . . 174
6.3.1 Introduction . . . 174
6.3.2 Methods . . . 175
6.3.3 Results and discussion . . . 176
6.3.4 Conclusions . . . 178
6.4 Eddy Current Effects . . . 180
6.4.1 Introduction . . . 180
6.4.2 Methods . . . 182
6.4.3 Results and discussion . . . 183
6.4.4 Conclusions . . . 186
6.5 Direct MR imaging of neuronal activity . . . 188
6.5.1 Introduction . . . 188
6.5.2 Methods . . . 190
6.5.3 Results and discussion . . . 193
6.5.4 Conclusions . . . 198
6.6 Summary . . . 198

7 Conclusions and future directions 201
7.1 Development of the simulator . . . 202
7.2 Applications of the simulator . . . 204
7.3 Future development of the simulator . . . 206
7.4 Final remarks . . . 209

A Magnetic field calculations **211**

Bibliography **215**

List of Figures

1.1 The 3T Varian/Siemens magnetic resonance imaging system . . . 4

2.1 Currently used functional brain mapping techniques. 12
2.2 Formation of the MR image. 13
2.3 Nuclear precession. 15
2.4 The solution of the Bloch Equation for a fat tissue sample. 19
2.5 Excitation. 24
2.6 Process of slice selection with sinc RF pulse. 29
2.7 Two dimensional k-space for Fourier transform samples. 31
2.8 MRI system hardware. 33
2.9 An example of a pulse sequence diagram. 35
2.10 Gradient echo. 36
2.11 Echo Planar Image. 37
2.12 Scanning parameters. 38
2.13 Different image contrasts of an axial brain slice at 3T. 39
2.14 T_1 recovery and T_2 decay curves. 40
2.15 The BOLD response. 45
2.16 Magnetic susceptibility artefacts in an EPI image. 48
2.17 Motion artefact: position displacement. 49
2.18 Motion-related artefacts in the FMRI data. 50
2.19 Spin history artefact. 51
2.20 Blurring artefact. 51
2.21 Chemical shift artefact in an EPI image. 52
2.22 The effect of eddy currents on an ideal gradient waveform. 53
2.23 Image distortions due to eddy currents. 53

3.1 Tissue templates. 58
3.2 Descriptive representation of the simulation process. 61
3.3 T_2^* variation. 70

4.1 Descriptive representation of the simulation process. 76
4.2 Descriptive representation of the pulse sequence. 81

4.3 Examples of different slice profiles. 83
4.4 Descriptive representation of the parallelisation. 102
4.5 Limitation of the parallelisation. 103

5.1 Validation graph. 106
5.2 Optimal voxel size. 110
5.3 Spin history effects. 112
5.4 Image Contrast. 114
5.5 B_0 field inhomogeneities. 117
5.6 Translation in z: Input object. 120
5.7 Translation in z: Perturbation field. 121
5.8 Translation in z: Motion. 122
5.9 Translation in z: Analysis. 123
5.10 Experimental results. 125
5.11 Spin history results. 125
5.12 Rotation about y-axis: Input object 128
5.13 Rotation about y-axis: Perturbation field. 128
5.14 Rotation about the y-axis: Motion. 129
5.15 Rotation about the y-axis: Result 1. 132
5.16 Rotation about the y-axis: Result 2. 133
5.17 Rotation about the y-axis: Result 3. 134
5.18 Rotation about the y-axis: Result 4. 135
5.19 Ghosting. 136
5.20 Chemical Shift. 137
5.21 EPI with B_0 field distortions. 139
5.22 Eddy Currents. 140
5.23 Within-scan motion. 140

6.1 Motion types. 145
6.2 Level1 motion used in the simulations. 150
6.3 Level2 motion used in the simulations. 151
6.4 Level3 motion used in the simulations. 151
6.5 RMS diff: Translations. 159
6.6 RMS diff: Rotations. 160
6.7 RMS diff: All. 161
6.8 Group4 and Group5 simulations: RMS error. 164
6.9 MCFLIRT summary. 165
6.10 ICA: Rotation about the z-axis. 170
6.11 IC with motion-B_0 effect. 171
6.12 A bar-chart with std values. 172
6.13 The correlation coefficients. 172
6.14 Stimulus correlated motion. 177
6.15 Eddy Currents: DW sequences. 181

6.16 Simulated Images displaying eddy current effects. 184
6.17 Sum of squares of the residuals after affine registration. 185
6.18 Phantom used in the experiments. 191
6.19 Input object and the B_0 inhomogeneity field. 192
6.20 Images representing the signal change when the current is on. . . 194
6.21 Sensitivity to the current amplitude. 195
6.22 Signal change in a 9 by 9 patch. 195
6.23 Results of the "walk-through" experiment. 197

List of Tables

4.1 MR parameters. 79
4.2 Computational time. 103

6.1 Accuracy of the estimated parameters. 153
6.2 Accuracy of the application of the transformation. 157
6.3 Error in applying the final transformation in MCFLIRT. 166

List of Abbreviations

BOLD	Blood Oxygenation Level Dependant
CSF	Cerebro-Spinal Fluid
DWI	Diffusion-Weighted Imaging
EPI	Echo Planar Imaging
FID	Free Induction Decay
FLASH	Fast Low-Angle SHot
FLIRT	FMRIB Linear Image Registration Technique
FMRI	Functional Magnetic Resonance Imaging
FMRIB	Oxford centre for fMRI of the Brain
FSL	FMRIB software library
FOV	Field Of View
FT	Fourier Transform
GE	Gradient-Echo
GM	Grey Matter
ICA	Independent Component Analysis
MCFLIRT	Motion Correction using FLIRT
MIDAS	MR Imaging Data Acquisition Simulator
MRI	Magnetic Resonance Imaging
NMR	Nuclear Magnetic Resonance
POSSUM	Physics Oriented Scanner Simulator Utility for MRI
RF	Radio-Frequency
ROI	Region-Of-Interest
SCM	Stimulus Correlated Motion
SE	Spin-Echo
SNR	Signal-to-Noise Ratio
TE	Echo Time
TR	Repetition Time
WM	White Matter

Chapter 1

Introduction

1.1 Motivation

Magnetic resonance imaging (MRI) is an imaging technique that, since its beginnings in the 1970's, has become one of the most important tools in understanding how the human body works. Among its many applications in the biomedical sciences, MR imaging is used in studying the anatomy, pathology and, more recently, function of the brain. The development of this most recent MR imaging application, called Functional Magnetic Resonance Imaging (FMRI), has been driven in large measure by its potential to increase our understanding of how the human brain works, both in the normal and diseased states.

The aim of FMRI data analysis is to identify small, spatially localised changes in image intensity that are induced by the performance of some experimental task. This is normally accomplished by collecting a series of images covering part or all of the brain at intervals of a few seconds and analysing the resulting time-series obtained at each image voxel. The experimentally-associated intensity changes for each voxel are frequently only a few percent of the mean value at that spatial location and are embedded in a noisy signal contaminated by electronic and physiological artefacts. These are examples of significant imperfections in the



Figure 1.1: The 3T Varian/Siemens magnetic resonance imaging system installed at the Oxford Centre for Functional Magnetic Resonance Imaging of the Brain (FMRIB) where the research described in this report was carried out. It is designed specifically for ultra-fast imaging in brain functional magnetic resonance imaging studies and is equipped with both a body gradient coil and a fast head gradient coil insert.

FMRI method. Head motion is one further, and critically important, confound. Very small movements of the head, on a scale of less than a millimetre, can be a major source of error in FMRI analysis. This is particularly the case in situations when the motion is correlated with the task being carried out by the subject.

The imperfections due to artefacts in the FMRI method can not be resolved in a straightforward way. For a large number of artefacts, reduction or elimination is carried out before and during scanning by improving the scanning system, and choosing the appropriate parameters of the scanning process. However, for some of them (head motion, magnetic susceptibility induced magnetic field inhomogeneities, physiological changes in the body etc.) this is not possible. In the case of motion artefacts, the most common approach is to tell subjects not to move during scanning, and to repeat the scanning sequence if they do move apprecia-

bly. However, very often the subjects in the scanner are patients (as opposed to volunteers) or small children where movement is inevitable and scanning time limited and not repeatable.

In order for FMRI to become a reliable and widely used clinical tool, it is essential to have techniques and tools to recognise and eliminate, or at least minimise, artefacts that contaminate the signal. There has been a considerable amount of research effort directed at this issue. However, most of the artefact correction techniques that are developed are based on analyses of empirically acquired FMRI data. As a result, the ground truth is not known which makes it difficult to assess the precise accuracy of these methods. (e.g. see [32, 38, 25, 41, 36]).

In order to be able to measure the accuracy of existing techniques a computer simulation of the whole FMRI acquisition, starting from the first-principles of signal generation and detection has been developed in this thesis. This makes it possible to simulate various artefacts, such as head motion and B_0 inhomogeneities, in a controlled way. Understanding and quantifying the artefacts in this way enables a realistic ground truth for the signal to be established. Furthermore, with the simulator it is also possible to investigate the magnitude of the artefacts and their interactions, which can then lead to the development of new algorithms for their reduction, and ideally, removal.

Although the original motivation for this project came from its applications in artefact correction in FMRI, the simulator itself has many other exciting applications. These are in the fields of image acquisition, neuronal current imaging, eddy-current modelling as well as for educational purposes, to mention just a few examples. Some of these applications form a part of the research described in this thesis and they are set out in the Application section (Chapter 6). The precise scope and the objectives of this thesis are set out in the following section.

1.2 Scope and objectives

The aim of the research presented in this thesis is to develop and then consequently use a general computer simulation program for FMRI with the following properties:

- The ability to produce realistic anatomical simulated MR images of any brain (or any other object) for a variety of different scanning conditions;

- The ability to produce realistic time-dependent sequences of simulated MR images including the effects of physiological activations due to some external stimulus;

- The ability to model realistic image related artefacts (mainly due to rigid-body motion of the object and magnetic field inhomogeneities).

It is important to note that this simulator is aimed at modelling the artefacts that are one possible cause of errors in investigating the functional response. Physiological aspects of that functional response are modelled via changes in physical parameters (mainly T_2^*) but the relations between the various physiological variables (such as the blood volume or the blood flow) are not modelled. This is why the main attention of the thesis is on generation of images and image artefacts. Less detail is given to the description of the physiology of the functional response in the brain.

Throughout this thesis this simulation program will be referred to as "the simulator" or "POSSUM" (Physics Oriented Simulated Scanner Utility for MRI). The outline of the work done in this thesis is described in the following section.

1.3 The outline of the thesis

The material in this thesis is organised as follows:

Chapter 2 (The background) describes the physical and physiological principles of FMRI. Its main focus, and the majority of the chapter, is dedicated to the theory of MRI which is the main tool exploited by FMRI. The simulator is ultimately a scanner simulator and therefore relies on the main concepts of MRI. MRI physics is given in necessary details, covering each step of the image formation, which are essential for the research work described in this thesis. Description of the commonly used pulse sequences (specifically Echo Planar Imaging - EPI) is also given together with the generation of image contrast, and various scanning parameters.

In the second part of the chapter, the physiological principles of FMRI are described. The nature and interpretation of the BOLD (blood oxygen level dependent) signal, and the spatial and temporal resolution of the signal are discussed. Finally, common FMRI artefacts (namely: motion and the magnetic field inhomogeneities) are discussed.

Chapter 3 (Simulator model) describes the development of the theoretical model which is used by the simulator. The main engine of the simulator is based on solving the Bloch equations. This chapter starts from the basic form of the Bloch equations and then discusses solving them when magnetic field inhomogeneities are present and the object is moving. In addition, a few other important parts of the simulator are discussed and these are: the eddy current artefact, RF inhomogeneities and thermal noise.

Chapter 4 (Simulator implementation) builds on where the previous chapter stopped. It describes the way the model was implemented in software. An overview of all component parts of the software is given. One of the main issues with the software has been improving its speed and therefore the main focus in this chapter is on the work devoted to reducing computational time. The chapter concludes with a presentation of the parallelisation technique implemented into the software and the relevant results.

Chapter 5 (Simulator validation) is the last step in the development process of the simulator. It is concerned with the quantitative evaluation of the software against both theoretical and experimental results. In addition, qualitative results which demonstrate the possibility of simulation of images impacted by realistic B_0 inhomogeneities, chemical shift, ghosting, eddy currents and within-scan motion are also presented.

Chapter 6 (Simulator applications) consists of five different applications of the simulator. Firstly, application of the simulator in evaluating a motion correction algorithm is presented. The accuracy of the algorithm is discussed under various scanning (simulated) conditions: with a few different types of motion paradigms, with or without the magnetic field inhomogeneities and with and without noise. Different cost functions, interpolation techniques and tolerance levels of the algorithm were compared.

The second application of the simulator was an investigation of the performance of Independent Component Analysis (ICA) as a tool for quantifying motion-related artefacts. An artefact-rich simulated data set when a) the object is not moving; b) the object is moving a minimal amount; c) the object is moving with an average amount of motion; d) the object is moving with a large amount of

motion; is analysed using an ICA method. A number of conclusions were drawn from these results.

The third application of the simulator was quantitative evaluation of the impact of stimulus correlated motion (SCM) artefacts in FMRI data. Two groups of simulated data sets, with SCM and without SCM, are presented. Both groups contained simulated data with various levels of realistic "patient" extracted motion. Data was processed using a General Linear Model (GLM) based tool for functional analysis. The influence of the magnetic field inhomogeneities on the analysis is also discussed.

The fourth application of the simulator was concerned with the eddy current effects. Two different pulse sequences were looked at: standard and optimised sequence. The simulations for both of these pulse sequences were carried out, and the respective results compared in order to choose the sequence which minimises the eddy current effects. In addition, the ability to correct for the effect of eddy currents using an affine registration method is analysed.

The fifth, and last application concerns neuronal current imaging. Various pulse sequence parameters were considered in order to investigate the magnitude of the signal change due to the neuronal currents. The results were compared to the results obtained experimentally.

Chapter 7 (Conclusions) presents a summary of the previous chapters and results, and discusses the contributions of this thesis. In addition, further developments of the simulator, together with suggestions for some interesting applications of the simulator in the future are also discussed.

Chapter 2

Principles of FMRI

Functional Magnetic Resonance Imaging (FMRI) is an imaging technique for examining brain function. Since its first appearance in 1991 (Belliveau et al.[8]) the use of FMRI in neuroscience has grown exponentially. The main reason for this is that FMRI has many advantages over other techniques for functional brain mapping. These advantages include its unique ability to combine relatively high spatial and temporal resolution, together with the ability to perform repeated measurements safely on the same subjects due to its non-invasive nature. A schematic comparison of the FMRI technique with other imaging modalities is shown in Figure 2.1.

FMRI exploits the existing Magnetic Resonance Imaging (MRI) method as its main tool for the generation of images. MRI and its properties are discussed in the first few sections of this chapter. The following sections describe the physiological principles of FMRI specific to the scope of this thesis. Finally, a set of imaging artefacts that are commonly present in FMRI data are also described.

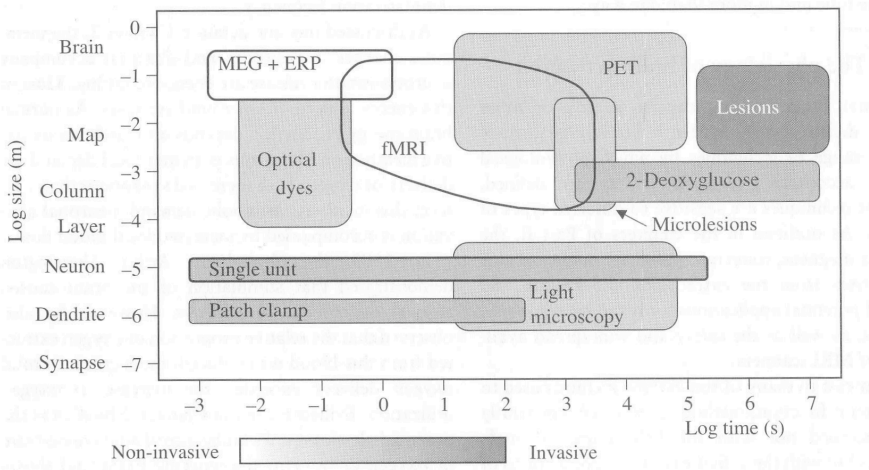


Figure 2.1: Relative spatial and temporal sensitivities of different functional brain mapping methods are shown. FMRI is unique in its ability to combine relatively high spatial and temporal resolution. An additional characteristic of FMRI, which is probably one of the most important is that it is non-invasive. Figure adapted from Cohen et al. [18].

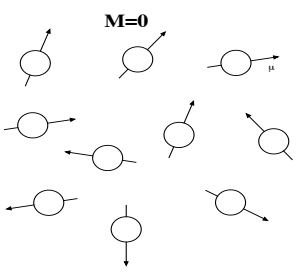
2.1 Physical principles of image generation

In this section we summarise the key physical phenomena exploited by magnetic resonance (MR) techniques in the formation of images. We introduce how MR signals are generated, manipulated, detected and processed into an image. In doing so, we start with the microscopic magnetic moments $\boldsymbol{\mu}$ in an object and then trace step-by-step how they are subsequently converted to a magnetisation $\mathbf{M}$, a transverse magnetisation $\mathbf{M_{xy}}$, and an electrical signal $S(t)$. $S(t)$ is measured and reordered into a matrix which is called k-space in such a way that it represents Fourier coefficients of the desired image $I(\mathbf{r})$. $I(\mathbf{r})$ is reconstructed computationally via the Inverse Discrete Fourier Transform (IDFT). The flowchart of the process is represented by the relations:

$$\mu \rightarrow \mathbf{M} \rightarrow \mathbf{M_{xy}} \rightarrow S(t) \rightarrow S(\mathbf{k}) \rightarrow I(\mathbf{r}). \tag{2.1}$$

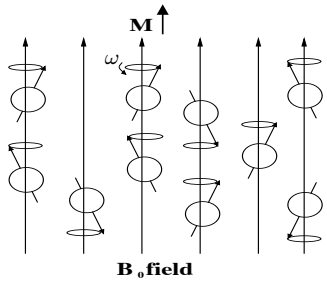
An outline descriptive summary of this process is given in Figure 2.2. More details

Each of the hydrogen protons in an object spins (in classical analogy) and produces a small magnetic field μ (magnetic moment). In the absence of an external magnetic field, these magnetic moments orient randomly, cancelling each other.



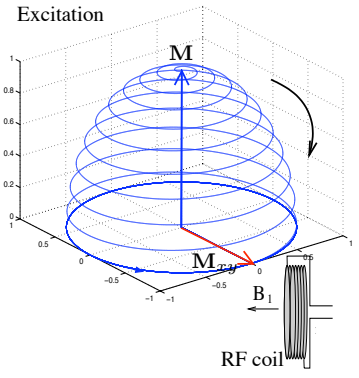
$\mu \rightarrow \mathbf{M}$

Once the object is in an external magnetic field $\mathbf{B}_0$ (e.g. the main magnet of the scanner), magnetic moments start precessing about the axis of the field. They tend to orient with the field which creates a net magnetisation $\mathbf{M}$ in the direction of the $\mathbf{B}_0$ field. They precess with the frequency ω which is directly proportional to the strength of the magnetic field.



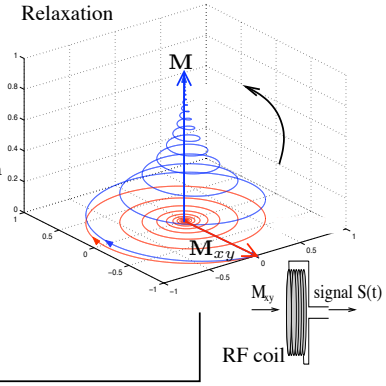
$\mathbf{M} \rightarrow \mathbf{M}_{xy}$

An additional magnetic field $\mathbf{B}_1$ (RF field) is transmitted from the RF coil (part of the MRI scanner). If the frequency of the RF pulse matches the precession frequency of the protons, then those protons get excited and the vector $\mathbf{M}$ flips in a rotating fashion away from the z-axis creating a projection in the x-y plane $\mathbf{M}_{xy}$.



$\mathbf{M}_{xy} \rightarrow S(t)$

The RF field is then turned off and $\mathbf{M}$ starts relaxing i.e. rotating back to its original position with the frequency directly proportional to the strength of the magnetic field. During the rotation, the magnetization in the x-y plane, $\mathbf{M}_{xy}$, is measured via an RF coil, producing the signal S(t).

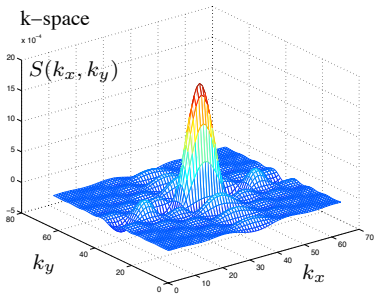


$S(t) \rightarrow S(k_x, k_y)$

During the process of signal collection, two additional, mutually independent magnetic fields are applied. They are called the x and y gradient fields and are spatially changing (linearly) in the x and y direction respectively. Controlled manipulation of these gradient fields creates spatial and time dependent changes in the magnetic field and therefore in the frequency of rotation of the magnetisation vector $\mathbf{M}$. These changes are encoded as information in the signal S(t):

$$S(k_x, k_y) = \int\int \mathbf{M}_{xy}(x, y) e^{-i2\pi(xk_x + yk_y)} \mathrm{d}x\mathrm{d}y$$

where $\mathbf{M}_{xy}(x, y)$ represents the transverse magnetisation in the object and $k_x(t)$, $k_y(t)$ describe the amount of change of the gradient fields and indirectly the frequency content of the signal. Once calculated, the signal is reordered into a 2D matrix which "lives" in an abstract space called the k-space.



$S(k_x, k_y) \rightarrow I(x, y)$

Once the signal is reordered into the k-space, it is then reconstructed into an image using the Discrete Fourier transform (DFT) The final image represents the magnetisation of hydrogen protons as a function of x and y.

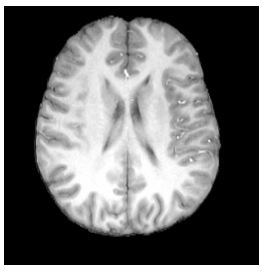


Figure 2.2: The MR image of the object I(x,y), is derived from the signal that comes from hydrogen protons. It describes the transverse magnetisation $\mathbf{M}_{xy}(x, y)$ that is created in the object. The magnetisation can be reconstructed from its Fourier coefficients $S(k_x, k_y)$ which in turn have physical interpretation in the MR signal. The goal of MR imaging is to extract these coefficients from the signal and then to use them to reconstruct the image. This figure describes descriptively the process: measuring Fourier coefficients, their ordering into the frequency space (often called the k space), and consequent image reconstruction by the means of the Discrete Fourier Transform (DFT).

can be found in [1], [14], [29], [48] and [33].

2.1.1 Magnetisation Vector: $\boldsymbol{\mu} \rightarrow \mathbf{M}$

Classical Response of a Single Nucleus to a Magnetic Field

In this section the focus is on a single proton's response to an external magnetic field $\mathbf{B}$, ignoring the interactions of the proton with its surrounding environment.

A fundamental property of a nucleus with an odd atomic number is that it possesses an angular momentum $\mathbf{J}$, often called its spin. Although nuclear spin is a property characterised by quantum mechanics, in the classical vector model, these nucleons can be visualised as spinning charged spheres that give rise to a small magnetic moment $\boldsymbol{\mu}$. In biological systems, the hydrogen nucleus ^1H (a single proton), is the most abundant nucleus of this type, generating the largest signals, and therefore is by far the most commonly used in imaging. Throughout this thesis, when we talk about magnetic moments, spins etc., we will assume that we are only looking at hydrogen nuclei.

The direct relationship between the magnetic moment $\boldsymbol{\mu}$ and the spin angular momentum vector $\mathbf{J}$ is found from experiment,

$$f\boldsymbol{\mu} = \gamma\mathbf{J} \tag{2.2}$$

where γ is a nucleus dependent constant, called the gyromagnetic ratio. For the hydrogen nucleus, it is found to be

$$\gamma = 2.675 \times 10^8 \, \frac{rad/s}{T} \tag{2.3}$$

where T is Tesla, the unit of magnetic field strength [29].

According to classical mechanics, the torque $\mathbf{N}$ that a magnetic moment $\boldsymbol{\mu}$ experiences from an external magnetic field $\mathbf{B}$ is $\mathbf{N}(t) = \boldsymbol{\mu}(t) \times \mathbf{B}(t)$. Also, nonzero torque on a system implies that the system's total angular momentum

$\mathbf{J}$ changes according to $\frac{\mathrm{d}\mathbf{J}(t)}{\mathrm{d}t} = \mathbf{N}(t)$ [29]. Using both of these equations, and including the relation between the spin and the magnetic moment expressed in equation 2.2 we have

$$\frac{\mathrm{d}\boldsymbol{\mu}}{\mathrm{d}t} = \gamma\boldsymbol{\mu} \times \mathbf{B} \tag{2.4}$$

which is the fundamental equation of motion for isolated spins in the classical treatment. The quantum description of the precession details can be found in [1] and [14].

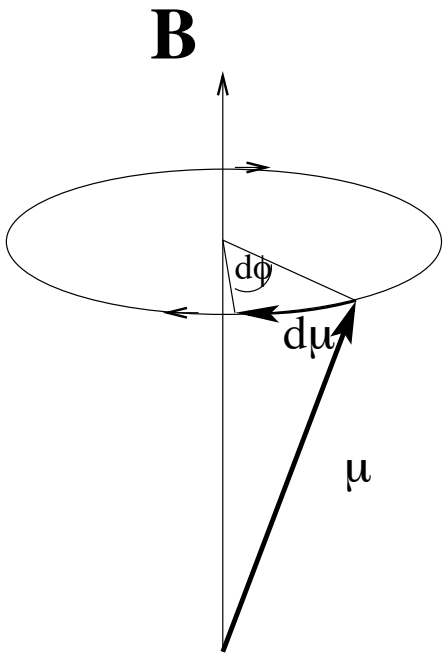


Figure 2.3: Precession of a spin about an applied external magnetic field $\mathbf{B}$

If we form the scalar product of both sides of equation 2.4 with $\boldsymbol{\mu}$ we get equation $\frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}(\boldsymbol{\mu} \cdot \boldsymbol{\mu}) = 0$ which implies that the magnitude of the magnetic moment does not change with time. On the other hand, the direction of the vector $\boldsymbol{\mu}$ changes with time. We can see in Figure 2.3, that as $\mathrm{d}\boldsymbol{\mu} = \gamma\boldsymbol{\mu} \times \mathbf{B}\mathrm{d}t$ is perpendicular to the plane defined by $\boldsymbol{\mu}$ and $\mathbf{B}$, the tip of the magnetic moment $\boldsymbol{\mu}$

will be pushed on a clockwise precession around a circular path. The angular frequency of the precession is

$$\omega \equiv \left| \frac{\mathrm{d}\phi}{\mathrm{d}t} \right| = \gamma |\mathbf{B}|. \tag{2.5}$$

where $\frac{\mathrm{d}\phi}{\mathrm{d}t}$ is the change in time of the precessional angle of the magnetic moment This equation is also know as the Larmor precession formula and it is one of the most important equations in MRI as it relates the strength of the magnetic field to the frequency that the spins precesses about it.

Magnetisation Vector and the Bloch Equation

In this section we extend the approach we used in the previous section to more than one spin, and focus on modelling the behaviour of the magnetisation of the spin system in the external magnetic field $\mathbf{B}$.

Let us consider a collection of spins placed in a small volume element, that is being exposed to a homogeneous magnetic field $\mathbf{B}$. We introduce the averaged vector sum of all the magnetic moments in the element, called the magnetisation vector

$$\mathbf{M} = \frac{1}{V} \sum_{n=1}^{N} \boldsymbol{\mu}_n \tag{2.6}$$

where N is the total number of spins in the voxel and V is the element volume. In analogy with the magnetic moment of one nucleus, $\mathbf{M}$ represents the local magnetic moment of the element.

If there were no interactions between the spins and their neighbourhood, the magnetisation vector would, according to equations 2.4 and 2.6, behave in the same fashion as the individual magnetic moments

$$\frac{\mathrm{d}\mathbf{M}}{\mathrm{d}t} = \gamma \mathbf{M} \times \mathbf{B}. \tag{2.7}$$

However, due to interactions of the spins with their environment this is not the case.

The thermal equilibrium of the magnetisation vector when exposed to a constant, static magnetic field $B_0\hat{\mathbf{z}}$ is

$$\mathbf{M} = M_0\hat{\mathbf{z}} \tag{2.8}$$

where M_0 is a value determined from quantum statistics

$$M_0 = \frac{\gamma^2\hbar^2 B_0\rho}{4KT_s}, \tag{2.9}$$

$\hbar$ is Planck's constant h ($6.6 \times 10^{-34}J$) divided by 2π, ρ is the spin density of the sample (number of spins per unit volume), K is the Boltzmann constant ($1.38 \times 10^{-23}J/K$), and T_s is the absolute temperature of the spin system. MRI experiments are often carried out at room temperature, and therefore the only controllable parameter is B_0 which depends on the imaging system and typically ranges from 0.2 to 7 Tesla (the scanner used in FMRIB, Oxford is a 3T scanner).

According to the laws of thermodynamics, a spin system that has been perturbed from its thermal equilibrium state (due to the perturbation of the external magnetic field) will return to this state, provided sufficient time is given. In MRI this process is characterised by two independent relaxation processes:

1. Recovery of the longitudinal magnetisation component M_z, called the longitudinal relaxation.

2. Cancellation of the transverse magnetisation component $\mathbf{M}_{xy} = M_x\hat{\mathbf{x}} + M_y\hat{\mathbf{y}}$, called the transverse relaxation.

Longitudinal relaxation is due to the spins giving back the energy they acquired from the perturbation of the external magnetic field to the surrounding lattice in order to go back to their equilibrium state. The time constant associated with this process is called the T_1 relaxation time, it is determined empirically and is fixed for a specific tissue (at a given static field strength).

The transverse relaxation is due to the spins de-phasing. It happens because when two spins are close to each other they slightly change the strength of the magnetic field that each is experiencing, and therefore the speed of their precession changes (according to equation 2.5), making them go out of phase relative to each other. Consequently, the vector sum of their magnetisation, in the transverse plane, tends to cancel. The time constant associated with this process is called the T_2 relaxation time, it is also determined empirically and is fixed for a specific tissue (at a given static field strength).

The process of transverse relaxation happens faster if the main magnetic field is not perfectly homogeneous. This is because the inhomogeneity in the field introduces additional differences in the speed of their precession and as a result they go out of phase relative to each other more than in the case when the field is perfectly homogeneous. In order to characterise this, another time constant known as the T_2^* relaxation time can be used instead of the T_2 relaxation time. T_2^* is not fixed for a given tissue as it depends on the inhomogeneity of the external field as well as tissue type and overall magnetic field strength.

Both the longitudinal and the transverse relaxation are modelled by including additional relaxation terms in equation 2.7. The newly formed equation describing the time dependent behaviour of the magnetisation vector is called the Bloch equation and is of the form

$$\frac{\mathrm{d}\mathbf{M}}{\mathrm{d}t} = \gamma\mathbf{M} \times \mathbf{B} - \frac{\mathbf{M}_{xy}}{T_2} + \frac{(M_0 - M_z)\hat{\mathbf{z}}}{T_1}. \tag{2.10}$$

For the case when there is only the main static magnetic field $\mathbf{B} = B_0\hat{\mathbf{z}}$ we have

$$\begin{aligned} \frac{\mathrm{d}M_z}{\mathrm{d}t} &= \frac{M_0 - M_z}{T_1} \\ \frac{\mathrm{d}M_x}{\mathrm{d}t} &= \omega_0 M_y - \frac{M_x}{T_2} \\ \frac{\mathrm{d}M_y}{\mathrm{d}t} &= -\omega_0 M_x - \frac{M_y}{T_2} \end{aligned} \tag{2.11}$$

where $\omega_0 = \gamma B_0$ is the Larmor frequency of the spin system. We can easily solve equations 2.11, with initial conditions $\mathbf{M}(0) = M_x(0)\hat{\mathbf{x}} + M_y(0)\hat{\mathbf{y}} + M_z(0)\hat{\mathbf{z}}$, and we present three different representations for the solution:

Cartesian Representation: The solution of the system in the Cartesian representation is

$$M_z(t) = M_z(0)e^{-t/T_1} + M_0(1 - e^{-t/T_1}) \quad (2.12)$$

$$M_x(t) = (M_x(0)\cos(\omega_0 t) + M_y(0)\sin(\omega_0 t))e^{-t/T_2} \quad (2.13)$$

$$M_y(t) = (M_y(0)\cos(\omega_0 t) - M_x(0)\sin(\omega_0 t))e^{-t/T_2} \quad (2.14)$$

Figure 2.4 gives the graphical representation of the solution for each of the

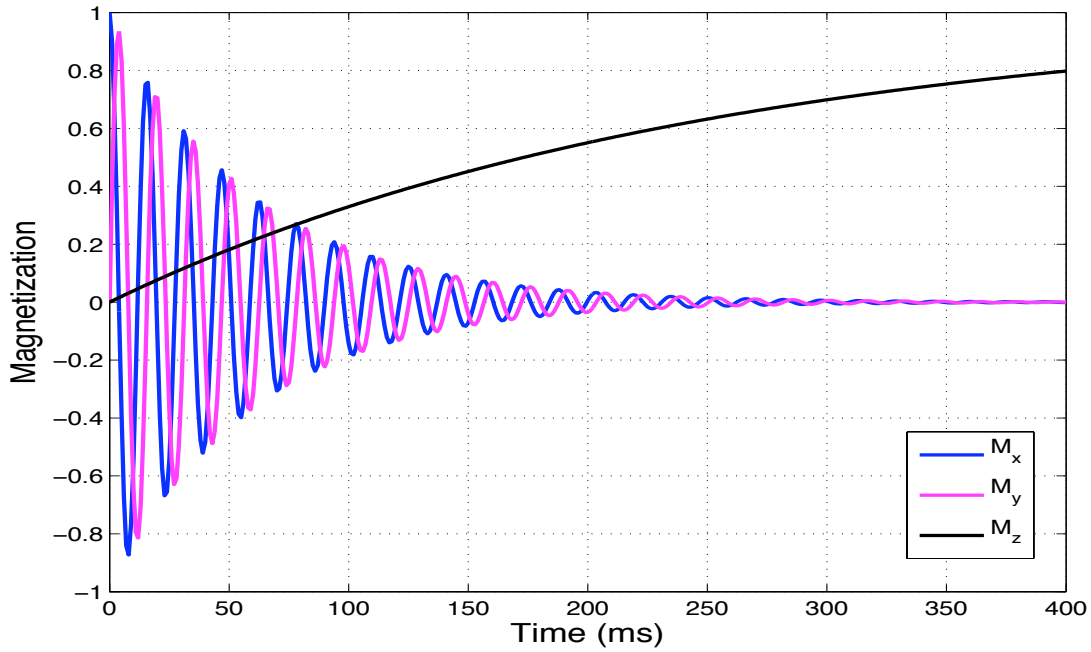


Figure 2.4: The regrowth of the longitudinal component of magnetisation from the initial value $M_z(0) = 0$ (time $t = 0$ represents the moment just after the RF pulse is finished) toward the equilibrium value $M_0 = 1$ is shown with the M_z curve. The 'oscillatory' decay of the transverse magnetisation from the initial values $M_x(0) = 1, M_y(0) = 0$ is shown with curves M_x and M_y. Note that the initial values of the magnetisation vector are illustrative.

components of the magnetisation vector for a fat tissue sample in the brain where $T_1 = 250$ms and $T_2 = 60$ms, which are typical values at $B_0 = 1.5T$. In three dimensional space, the vector $\mathbf{M}$ spirals clockwise about the z-axis, with the angular frequency ω_0, until it reaches its equilibrium state $M_0\hat{\mathbf{z}}$ (see Figure 2.2).

Matrix Representation: If we represent a clockwise rotation about the z-axis, through an angle θ by the matrix

$$R_z(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.15}$$

and denote

$$W(t) = \begin{bmatrix} e^{-t/T_2} & 0 & 0 \\ 0 & e^{-t/T_2} & 0 \\ 0 & 0 & e^{-t/T_1} \end{bmatrix} \quad \text{and} \quad Q(t) = \begin{bmatrix} 0 \\ 0 \\ M_0(1 - e^{-t/T_1}) \end{bmatrix} \tag{2.16}$$

then the solution in matrix form is

$$\mathbf{M}(t) = W(t)R_z(\omega_0 t)\mathbf{M}(0) + Q(t) \tag{2.17}$$

where

$$\mathbf{M}(t) = \begin{bmatrix} M_x(t) \\ M_y(t) \\ M_z(t) \end{bmatrix} \tag{2.18}$$

Complex Representation: Let us introduce the complex notation of the transverse magnetisation:

$$M^c_{xy} \equiv M_x + iM_y. \tag{2.19}$$

We use equations 2.13 and 2.14 and get

$$\begin{aligned} M^c_{xy}(t) &= e^{-t/T_2}(M_x(0)(\cos(\omega_0 t) - i\sin(\omega_0 t)) + M_y(0)(i\cos(\omega_0 t) + \sin(\omega_0 t)) \\ &= e^{-t/T_2} M^c_{xy}(0) e^{-i\omega_0 t} \end{aligned} \tag{2.20}$$

Magnitude-phase notation will prove more useful in describing the signal later, and is $M^c_{xy}(t) = |M^c_{xy}(t)|e^{i\phi(t)}$, giving

$$M^c_{xy}(t) = e^{-t/T_2}|M^c_{xy}(0)|e^{-i(\omega_0 t - \phi(0))} \tag{2.21}$$

which is the solution for the transverse magnetisation of the system 2.11 in complex representation.

This form of the solution is the most useful for representing the accumulated phase of the magnetisation vector due to its precession about the magnetic field.

Rotating Reference Frame

In order to visualise the rotation of a magnetisation vector about the static magnetic field in a simple manner, it is convenient to use a reference frame rotating at the Larmor precession frequency. To distinguish it from the conventional stationary frame (also called the laboratory frame), we use x', y' and z' to denote the three orthogonal axes of the rotating frame and correspondingly, $\hat{\mathbf{x}}', \hat{\mathbf{y}}'$ and $\hat{\mathbf{z}}'$ as their unit direction vectors. The two frames are related by the following transformation:

$$\begin{bmatrix} \hat{\mathbf{x}}' \\ \hat{\mathbf{y}}' \\ \hat{\mathbf{z}}' \end{bmatrix} = R_z(\omega_0 t) \begin{bmatrix} \hat{\mathbf{x}} \\ \hat{\mathbf{y}} \\ \hat{\mathbf{z}} \end{bmatrix} \tag{2.22}$$

where $R_z(\theta)$ is the rotation matrix introduced in the previous section and it describes clockwise rotation about the z-axis by an angle θ.

The Bloch equation for the magnetic field $\mathbf{B} = B_0\hat{\mathbf{z}}$ in the rotating frame can then be expressed in the form :

$$\frac{\mathrm{d}\mathbf{M}_{rot}}{\mathrm{d}t} = -\frac{M_{x'}\hat{\mathbf{x}}' + M_{y'}\hat{\mathbf{y}}'}{T_2} + \frac{(M_0 - M_{z'})\hat{\mathbf{z}}'}{T_1} \tag{2.23}$$

where $\mathbf{M}_{rot} = M_{x'}\hat{\mathbf{x}}' + M_{y'}\hat{\mathbf{y}}' + M_{z'}\hat{\mathbf{z}}'$ is the magnetisation vector in the rotating frame [48]. Its relation with the laboratory frame magnetisation vector is given

by the transformation rule between the two reference systems:

$$\mathbf{M}_{rot} = R_z(\omega_0 t)\mathbf{M}. \tag{2.24}$$

2.1.2 RF pulse: $\mathbf{M} \rightarrow \mathbf{M}_{xy}$

In this section we will introduce the main properties of an RF (radio frequency) pulse and describe the impact it has on the magnetisation of the object while exposed to the static magnetic field $B_0\hat{\mathbf{z}}$.

An RF pulse is an electro-magnetic field that oscillates with a frequency in the range 40-300 MHz, which belongs to the radio frequency window of the electromagnetic spectrum. Such pulses are usually transmitted from the RF coil which is built-in to the scanner, as seen in Figure 2.8. A typical RF pulse has its magnetic field perpendicular to the main magnetic field and is of the form

$$\mathbf{B_1}(t) = B_1^e(t)[\cos(\omega_{\text{rf}}t)\hat{\mathbf{x}} - \sin(\omega_{\text{rf}}t)\hat{\mathbf{y}}] \tag{2.25}$$

where $\mathbf{B}_1$ is the usual symbol for the RF field, ω_{rf} is the carrier frequency of the field and $B_1^e(t)$ is its envelope function. The form of the envelope function determines the shape of the RF pulse. The most common RF pulse in MRI is the sinc pulse and its envelope function is

$$B_1^e(t) = \begin{cases} B_1 sinc[\frac{\omega_{\text{rf}}}{2}(t - \frac{t_{\text{rf}}}{2})], & 0 \leq t \leq t_{\text{rf}} \\ 0, & \text{otherwise} \end{cases} \tag{2.26}$$

where $sinc(x) = \frac{sinx}{x}$.

Now we will look into the effects of an RF pulse on a spin 'isochromat' (a collection of spins with the same precessional frequency, ω_0). One very important property of this system is that when it is 'hit' by an oscillating magnetic field with the carrier frequency $\omega_{\text{rf}} = \omega_0$, resonance occurs and the system becomes 'excited' – that is, it is supplied with additional energy and is therefore perturbed

from its equilibrium state. When both the static and RF fields exist, then spins will precess about the total applied magnetic field, so that two rotations will take place. One rotation is about the static field $B_0\hat{\mathbf{z}}$ and the other one is about the RF field $\mathbf{B_1}$. In order to quantitatively describe this process we go back to the Bloch equation during the RF pulse:

$$\frac{\mathrm{d}\mathbf{M}}{\mathrm{d}t} = \gamma\mathbf{M} \times (B_0\hat{\mathbf{z}} + \mathbf{B}_1(t)) - \frac{\mathbf{M}_{xy}}{T_2} + \frac{(M_0 - M_z)\hat{\mathbf{z}}}{T_1}. \tag{2.27}$$

First of all, as the RF pulse is applied for a very short time ($t_{\mathrm{rf}} \sim 4$ms), for most of the imaging methods, compared with typical relaxation times ($T_1 \sim 1000$ms, $T_2 \sim 100$ms) we therefore have

$$t_{\mathrm{rf}} \ll T_1 \qquad t_{\mathrm{rf}} \ll T_2 \tag{2.28}$$

for most tissues. As the relaxation times describe the return of the magnetisation vector to its equilibrium, we can assume that during the RF pulse, no relaxation happens. Therefore, when solving the Bloch equation during this period (commonly called the excitation period), we can neglect the last two terms in (2.27) and get a simpler form of the equation:

$$\frac{\mathrm{d}\mathbf{M}}{\mathrm{d}t} = \gamma\mathbf{M} \times (B_0\hat{\mathbf{z}} + \mathbf{B}_1(t)) \tag{2.29}$$

where the RF field $\mathbf{B}_1(t)$ is defined as in equation 2.25.

In the rotating reference frame we introduced in the previous section, we obtain

$$\frac{\mathrm{d}\mathbf{M}_{rot}}{\mathrm{d}t} = \gamma\mathbf{M}_{rot} \times B_1^e(t)\hat{\mathbf{x}'} \tag{2.30}$$

where $\mathbf{M}_{rot} = R_z(\omega_0 t)\mathbf{M}$ as defined in equation 2.24. For more details on the derivation see [48]. The solution of this equation describes the behaviour of the magnetisation vector during an excitation period and in the rotating frame is

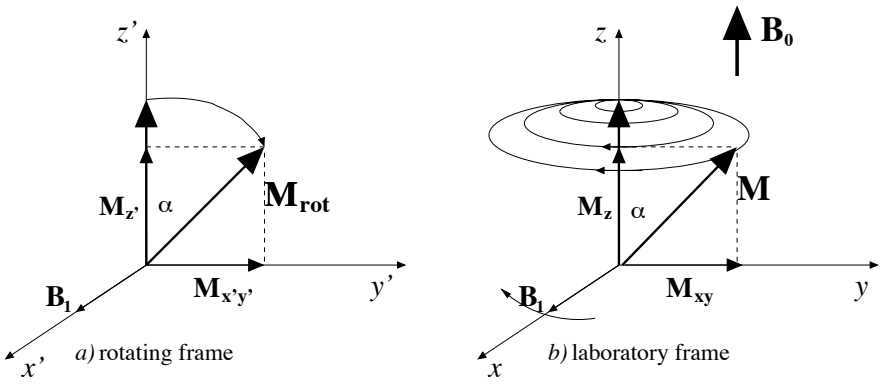


Figure 2.5: Excitation: the $\mathbf{B_1}$ field, tuned to the Larmor frequency and applied in the transverse plane, induces rotation of the magnetisation vector as it tips away from the z-axis for a flip angle α in a) the frame rotating at the Larmor frequency about the z-axis and b) the real scanner (laboratory) frame.

given by:

$$\begin{aligned} M_{x'}(t) &= M_{x'}(0) \\ M_{y'}(t) &= M_{y'}(0)\cos\alpha(t) + M_{z'}(0)\sin\alpha(t) \\ M_{z'}(t) &= -M_{y'}(0)\sin\alpha(t) + M_{z'}(0)\cos\alpha(t) \end{aligned} \tag{2.31}$$

where $0 \leqslant t \leqslant t_{\mathrm{rf}}$, $M_{x'}(0), M_{y'}(0)$ and $M_{z'}(0)$ are the values just before the RF pulse and α is the angle that the magnetisation vector $\mathbf{M}$ is tipped away from the z axis:

$$\alpha(t) = \int_0^t \omega_1(t')\mathrm{d}t' = \int_0^t \gamma B_1^e(t')\mathrm{d}t' \tag{2.32}$$

where ω_1 is the angular frequency of the rotation of the magnetisation vector about the $\mathbf{B}_1$ field. $\alpha(t_{\mathrm{rf}})$ is called the flip angle and gives the position of the magnetisation vector just after the RF pulse as seen in Figure 2.5.

A common flip angle is 90° where a magnetisation that is initially aligned with the z-axis is rotated to lie completely in the (x, y) plane, with no M_z component. In that case, the magnitude of the transverse magnetisation just after the RF pulse is $|\mathbf{M}_{xy}(t_{\mathrm{rf}}^+)| = M_0$ where t_{rf}^+ stands for the time just after the RF pulse

finished. Often we will use $t_{\mathrm{rf}}^{+} = 0$ as it also represents the initial time from which the relaxation period starts and during which, as we will describe in the following section, the signal is being collected.

2.1.3 Signal Detection: $\mathbf{M_{xy}} \rightarrow S(t)$

Once the magnetisation vector $\mathbf{M}$ is tipped away from the z-axis by the RF pulse, it will start relaxing back to its equilibrium position $M_0\hat{\mathbf{z}}$. During the process of relaxation, magnetisation has to be detected i.e. converted into an electrical signal. This is done by the receiver coil, which in traditional imaging systems also serves as the RF pulse transmitter and can be seen in Figure 2.8. In this section we will briefly describe the basics of the signal detection process. For a more detailed version see [1], [29], [48].

The process of signal detection is based on Faraday's law of electromagnetic induction which states that a time-varying magnetic flux through a conducting loop (a receiver coil) will induce in the coil an electromagnetic force (voltage) that is equal to the rate at which the magnetic flux $\Phi(t)$ in the coil is changing [29]:

$$V(t) = -\frac{\mathrm{d}\Phi(t)}{\mathrm{d}t}. \tag{2.33}$$

The relationship between the magnetic flux and the magnetisation vector can be seen in

$$\Phi(t) = \int_{object} \mathbf{B}^{receive}(\mathbf{r}) \cdot \mathbf{M}(\mathbf{r}, t)\mathrm{d}\mathbf{r} \tag{2.34}$$

where $\mathbf{B}^{receive}(\mathbf{r})$ is the laboratory frame magnetic field at location $\mathbf{r}$ produced by a hypothetical unit direct current flowing in the RF coil which describes the spatial coupling of the field to the receiver coil. The signal $S(t)$ measured by the

receiver electronics is proportional to $V(t)$ [29]:

$$S(t) \propto -\frac{\mathrm{d}}{\mathrm{d}t}\int_{object}\left[B_x^{receive}(\mathbf{r})M_x(\mathbf{r},t)+B_y^{receive}(\mathbf{r})M_y(\mathbf{r},t)+B_z^{receive}(\mathbf{r})M_z(\mathbf{r},t)\right]\mathrm{d}\mathbf{r}. \tag{2.35}$$

Consider the general solutions of the Bloch equation for each of the components of the magnetisation vector $\mathbf{M}$ during relaxation:

$$\begin{aligned} M_z(\mathbf{r},t) &= M_z(\mathbf{r},0)e^{-t/T_1(\mathbf{r})}+M_0(1-e^{-t/T_1(\mathbf{r})}) && (2.36)\\ M_{xy}^c(\mathbf{r},t) &= e^{-t/T_2(\mathbf{r})}|M_{xy}(\mathbf{r},0)|e^{-i(\omega_0 t-\phi_0(\mathbf{r}))}. && (2.37)\end{aligned}$$

Differentiation in (2.36) gives:

$$\begin{aligned} \frac{\mathrm{d}M_z(\mathbf{r},t)}{\mathrm{d}t} &= -\frac{1}{T_1}M_z(\mathbf{r},0)e^{-t/T_1(\mathbf{r})}-\frac{1}{T_1}M_0(1-e^{-t/T_1(\mathbf{r})})\\ \frac{\mathrm{d}M_{xy}^c(\mathbf{r},t)}{\mathrm{d}t} &= -\frac{1}{T_2}e^{-t/T_2(\mathbf{r})}|M_{xy}(\mathbf{r},0)|e^{-i(\omega_0 t-\phi_0(\mathbf{r}))}\\ &- i\omega_0 e^{-t/T_2(\mathbf{r})}|M_{xy}(\mathbf{r},0)|e^{-i(\omega_0 t-\phi_0(\mathbf{r}))}. && (2.38)\end{aligned}$$

The Larmor frequency, ω_0, is at least four orders of magnitude larger than typical values of $1/T_1$ and $1/T_2$, and therefore all but the last term in the second equation can be neglected, giving the signal, $S(t)$, as

$$\begin{aligned} S(t) \propto \omega_0\int_{object} e^{-t/T_2(\mathbf{r})}|M_{xy}(\mathbf{r},0)|[&B_x^{receive}(\mathbf{r})\sin(\omega_0 t-\phi_0(\mathbf{r}))\\ &+B_y^{receive}(\mathbf{r})\cos(\omega_0 t-\phi_0(\mathbf{r}))]\mathrm{d}\mathbf{r} \end{aligned} \tag{2.39}$$

where $B_x^{receive}$ and $B_y^{receive}$ can be written in terms of magnitude and phase, $B_x^{receive}(\mathbf{r}) \equiv B_\perp\cos(\theta_B(\mathbf{r}))$, $B_y^{receive}(\mathbf{r}) \equiv B_\perp\sin(\theta_B(\mathbf{r}))$ with $\theta_B(\mathbf{r})$ the reception phase angle and $B_\perp = |B_{xy}^{receive}|$. This leads to

$$S(t) \propto \omega_0\int_{object} e^{-t/T_2(\mathbf{r})}|M_{xy}(\mathbf{r},0)|B_\perp(\mathbf{r})\sin(\omega_0 t-\phi_0(\mathbf{r})+\theta_B(\mathbf{r}))\mathrm{d}\mathbf{r}. \tag{2.40}$$

The oscillations at the frequency ω_0 are removed from the signal by the electronic process of demodulation, which is equivalent to viewing the signal from a rotating

reference frame. More details can be found in [1], [14], [29]. Then, using the complex representation, the signal is given by,

$$S(t) \propto \omega_0 \int_{object} e^{-t/T_2(\mathbf{r})} |M_{xy}(\mathbf{r}, 0)| B_\perp(\mathbf{r}) e^{i(\phi_0(\mathbf{r}) + \theta_B(\mathbf{r}))} \mathrm{d}\mathbf{r}. \tag{2.41}$$

When the transmitting and receiving RF coils are sufficiently uniform, the receive field directional phase θ_B and the receive field amplitude $B_\perp$ are all independent of position [29], and therefore, together with the proportionality constant, can be absorbed into a single constant Λ. In the future, unless stated otherwise, when writing signal equations we will omit all the constants that are both spatially and temporally independent.

$$S(t) = \Lambda \int_{object} e^{-t/T_2(\mathbf{r})} |M_{xy}(\mathbf{r}, 0)| e^{i\phi_0(\mathbf{r})} \mathrm{d}\mathbf{r}. \tag{2.42}$$

This equation can also be written in the form:

$$S(t) = \int_{object} M^c_{x'y'}(\mathbf{r}, t) \mathrm{d}\mathbf{r} \tag{2.43}$$

where $M^c_{x'y'}(\mathbf{r}, t)$ is a complex representation of the transverse magnetisation in the rotating frame.

2.1.4 k-space: $S(t) \rightarrow S(\mathbf{k})$

We have seen in the previous section the process of signal detection. However, the goal of MR imaging is not only to determine the global signal coming from the object but also to determine the spatial distribution of the signal within the object. In MR this is achieved by measuring the Fourier coefficients of the image.

Since identical nuclei precess at different rates in locations where the magnetic field is different, the spatial localisation of the signal can be determined from the frequency content of the resulting MR signal, provided that a well defined spatial field variation is superimposed on the homogeneous static field $B_0\hat{\mathbf{z}}$ (the B_0 field)

during the imaging process. This variation is created by a (spatially) linearly varying magnetic field $\mathbf{B_G}$ called the gradient field.

A gradient field is a special kind of inhomogeneous field whose z-component B_{Gz} varies linearly in space along a specific direction called the gradient vector. The gradient field also has components in the x- and y- direction but these are ignored because the B_0 field is much stronger than the gradient field (three orders of magnitude stronger) and it is in the z-direction, therefore the vector sum of $B_0\hat{\mathbf{z}}$ and $\mathbf{B_G}$ only significantly changes the field magnitude when the z-component of the gradient field changes. The gradient field is defined by

$$\begin{aligned}\mathbf{G}(t) &\equiv \nabla B_{Gz}(\mathbf{r},t) \\ &= \frac{\partial B_{Gz}(\mathbf{r},t)}{\partial x}\hat{\mathbf{x}} + \frac{\partial B_{Gz}(\mathbf{r},t)}{\partial y}\hat{\mathbf{y}} + \frac{\partial B_{Gz}(\mathbf{r},t)}{\partial z}\hat{\mathbf{z}} \\ &\equiv G_x(t)\hat{\mathbf{x}} + G_y(t)\hat{\mathbf{y}} + G_z(t)\hat{\mathbf{z}} \end{aligned} \tag{2.44}$$

and therefore the total magnetic field (static plus linear) can be written as

$$\mathbf{B}(\mathbf{r},t) = (B_0 + B_{Gz}(\mathbf{r},t))\hat{\mathbf{z}} = (B_0 + G_x(t)x + G_y(t)y + G_z(t)z)\hat{\mathbf{z}} = (B_0 + \mathbf{G}(t)\cdot\mathbf{r})\hat{\mathbf{z}} \tag{2.45}$$

As we have seen in equation 2.5, the angular frequency of the precessing spins is proportional to the field strength

$$\omega(\mathbf{r},t) = \gamma|\mathbf{B}(\mathbf{r},t)| = \gamma B_0 + \gamma B_{Gz}(\mathbf{r},t) = \omega_0 + \gamma\mathbf{G}(t)\cdot\mathbf{r}. \tag{2.46}$$

This equation is of primary importance for spatial encoding of the signal as it relates the spatial coordinates with the frequency of precession of the magnetisation vector. The process of spatial encoding is done (for multi-slice imaging) using two steps:

1. Selective excitation

2. Spatial information encoding

The first step is the process of selective excitation (Figure 2.6). In order to generate an image of a two-dimensional slice, spins are selectively excited in only one plane of the brain, leaving the spins outside that plane unperturbed. In practice this is done by having the RF pulse on in the presence of a magnetic field gradient orientated in the direction normal to the desired plane [40]. The

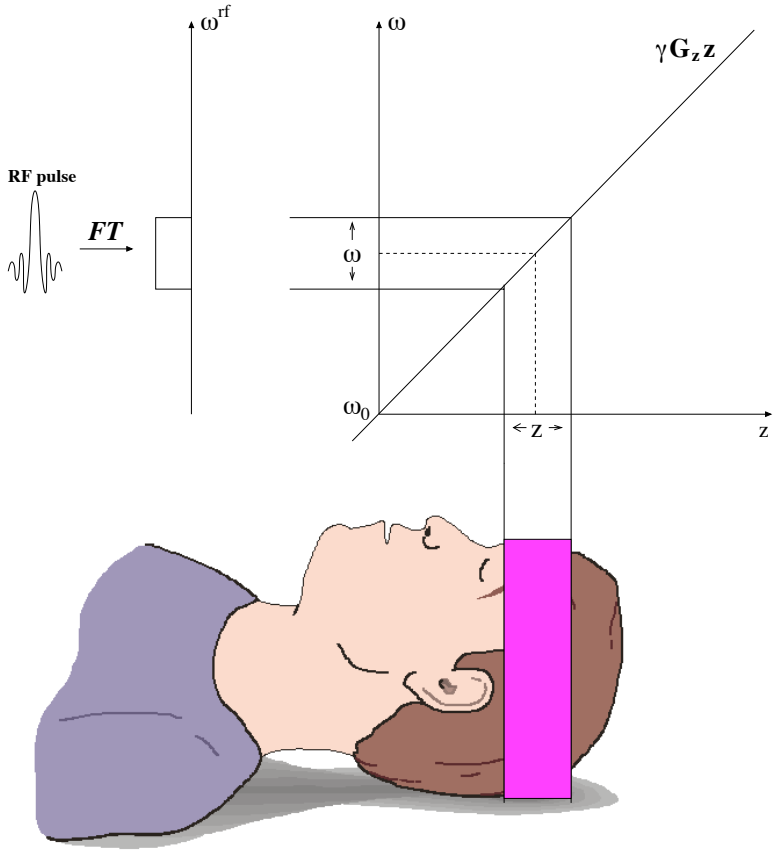


Figure 2.6: Slice selection for a slice in the (x, y) plane (z is the normal direction of the slice). FT stands for the Fourier Transform and shows the frequency profile of the sinc RF pulse. Only spins precessing with frequencies belonging to this range will get excited. As the desired frequency range only occurs in a small spatial range, then the effect is to select only this spatial 'slice'.

direction of the gradient field determines the direction of 'slicing'. Let us assume for now that it is the direction of the z-axis i.e. $\mathbf{B}_G = G_z z$ and that we have a sinc RF pulse. A property of sinc pulse is that the range of frequencies that

it selects is approximated by a 'box-car' (rectangular window function), so only those spins whose frequency lie within the range of the box-car will be excited.

The second step is the process of spatial information encoding. After spins of the chosen slice of the object are excited by a selective RF pulse, spatial information is encoded into the signal during the relaxation period. In this period the Bloch equations take the form

$$\frac{\mathrm{d}M_z}{\mathrm{d}t} = \frac{M_0 - M_z}{T_1} \tag{2.47}$$

$$\frac{\mathrm{d}M_x}{\mathrm{d}t} = (\omega_0 + \gamma \mathbf{G}(t)\cdot\mathbf{r})M_y - \frac{M_x}{T_2} \tag{2.48}$$

$$\frac{\mathrm{d}M_y}{\mathrm{d}t} = -(\omega_0 + \gamma \mathbf{G}(t)\cdot\mathbf{r})M_x - \frac{M_y}{T_2}. \tag{2.49}$$

The solution for this system will have an additional, spatially dependent, accumulation in phase, compared to the solution in the static field case (when all the spins have the same phase): $\phi(\mathbf{r},t) = -\int_0^t \omega(\mathbf{r},t')\mathrm{d}t' = -\omega_0 t - \gamma\int_0^t \mathbf{r}\cdot\mathbf{G}(t')\mathrm{d}t'$. Therefore the complex representation of the transverse magnetisation is:

$$M^c_{xy}(\mathbf{r},t) = e^{-t/T_2(\mathbf{r})}|M_{xy}(\mathbf{r},0)|e^{-i(\omega_0 t+\gamma\int_0^t \mathbf{r}\cdot\mathbf{G}(t')\mathrm{d}t')} \tag{2.50}$$

In the case when the z-gradient is used for a selective excitation, the slice is parallel to the (x,y) plane so only the x-gradient and the y-gradient are needed to encode the spatial information in the phase of the magnetisation vector. This information will be present in the spatial frequency of the detected signal:

$$S(t) = \int\int e^{-t/T_2(\mathbf{r})}|M_{xy}(\mathbf{r},0)|e^{-i\gamma\int_0^t (G_x(t')x+G_y(t')y)\mathrm{d}t'}\mathrm{d}x\mathrm{d}y. \tag{2.51}$$

The signal is commonly collected at uniformly spaced time points and then stored in a two dimensional plane called k-space with coordinates k_x and k_y defined by:

$$k_x(t) = \frac{\gamma}{2\pi}\int_0^t \mathbf{G}_x(t')\mathrm{d}t', \qquad k_y(t) = \frac{\gamma}{2\pi}\int_0^t \mathbf{G}_y(t')\mathrm{d}t'. \tag{2.52}$$

The coordinates in k-space at which the signal is stored form a regular grid (see Figure 2.7) whose size is commonly chosen to be 64×64, 128×128 or

256×256. From equation (2.52) it can be seen that a G_x gradient moves the k-space coordinate along the k_x direction and a G_y gradient moves the k-space coordinate along the k_y direction. This can be easily seen in the typical case when the gradients are constant so that:

$$k_x(t) = \frac{\gamma}{2\pi} G_x t, \qquad k_y(t) = \frac{\gamma}{2\pi} G_y t. \tag{2.53}$$

Neglecting the decay term $e^{-t/T_2(\mathbf{r})}$ and using equation (2.9), the signal in terms of k-space coordinates is:

$$S(k_x, k_y) = \int\int \rho(x, y) e^{-i2\pi(k_x x + k_y y)} \mathrm{d}x\mathrm{d}y. \tag{2.54}$$

It is clear that this represents the Fourier coefficients of the spin density $\rho(x, y)$ evaluated at specific points in k-space.

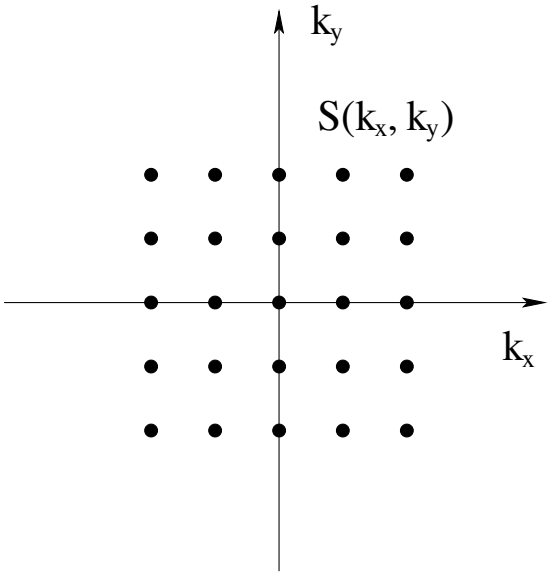


Figure 2.7: Two dimensional k-space for Fourier transform samples.

2.1.5 Image reconstruction: $S(\mathbf{k}) \rightarrow I(\mathbf{x})$

After the signal is sampled at discrete time points, corresponding to discrete locations in k-space, the reconstruction process is performed. The problem of reconstructing a function from its Fourier coefficients can be formulated as follows [48]:

determine the image $I(\mathbf{r})$ if the signal $S(\mathbf{k}_n) = \int_{object} I(\mathbf{r}) e^{-i2\pi(\mathbf{k}_n \cdot \mathbf{r})} \mathrm{d}\mathbf{r}$ is given,

where $\mathbf{k}_n \in \Psi$, and Ψ contains the set of k-space points at which measured data are collected.

The computational tool used for this type of image reconstruction when the samples are regularly spaced on a grid is the Inverse Discrete Fourier Transform (IDFT). In MRI, the common approach is to sample 2^r ($r \in N$) number of points for each k axis so that the Fast Fourier Transform (FFT) algorithm can be applied.

With this section we have completed the principles of the process of image production in MRI, starting from signal generation, through spatial encoding, to image reconstruction. In order for this whole process to be carried out, a set of instructions needs to be provided to the various coils in the magnet. The way this is done determines the properties of the image we want to generate. This is the topic of the following section.

2.2 Imaging system

2.2.1 Imaging hardware

The basic components of MRI hardware are:

- The *main magnet*. This provides a strong, uniform, static, magnetic field

referred to as the $\mathbf{B_0}$ field. The conventional coordinate system used for the scanner has the $\mathbf{B_0}$ field directed along the z-axis, so that $\mathbf{B}_0 = B_0\hat{\mathbf{z}}$, where B_0 is the constant describing the strength of the main magnetic field.

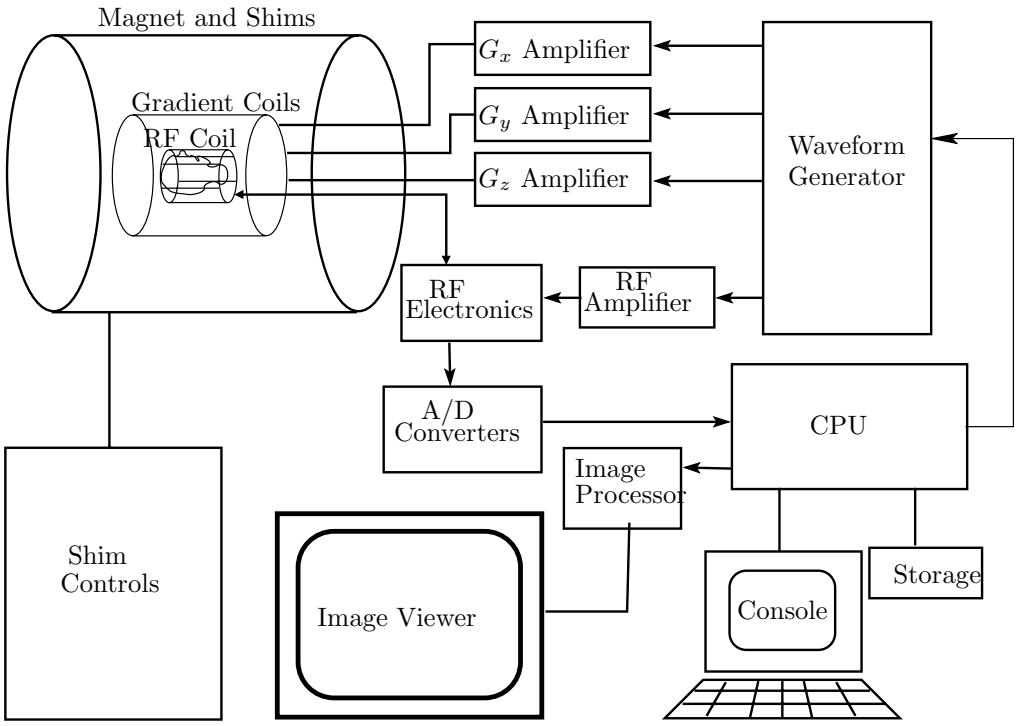


Figure 2.8: MRI system hardware. The figure courtesy of Stuart Clare.

- The *RF system.* This consists of a transmitter coil (RF coil) that is capable of generating a rotating radio-frequency electromagnetic field (known as the $\mathbf{B_1}$ field or an 'RF pulse') in the brain, and the receiver coil that is responsible for the detection of the signal, $S(t)$, emitted by the brain.

- A system of three orthogonal gradient coils called the *gradient system*, which produces a time-varying magnetic field, $\mathbf{B_G}$, of controlled spatial nonuniformity which varies linearly with position.

- A *shimming system* is a set of coils which is used to correct static inhomogeneities in the main magnetic field.

- A computer-based *imager system*, which stores, reconstructs and displays the obtained image.

The diagrammatic representation of the MRI system is given in Figure 2.8

2.2.2 Pulse sequence

In order for an MR image to be created, a set of instructions must be provided from the main console of the scanner to the gradient coils and to the RF coil which are located in the bore of the scanner (Figure 2.8). This set of instructions is known as the *pulse sequence*. The pulse sequence specifies the waveform of the RF pulses. It also specifies three waveforms, one for each of the G_x, G_y and G_z gradient fields. Furthermore, the pulse sequence specifies the timings of the signal read-outs which are then supplied to the RF coil. An accessible and widely used way of illustrating the pulse sequence is by the use of a pulse sequence diagram. A pulse sequence diagram is, as Hashemi et al. [33] nicely put it, " to an MR scientist as sheet music is to a musician". An example of a pulse sequence diagram is shown in Figure 2.9.

The first line (from top) of Figure 2.9 describes the instructions for the RF pulse: timing and the size of the flip angle. The second line describes the G_z gradient waveform. The G_z gradient together with the RF pulse is responsible for the slice selection. This is why it is often referred to as the slice-select gradient. The third line describes the G_y gradient waveform. It is this one that is used for going from one line of the k-space to the next and is known as the phase-encode gradient. Both the G_y gradient and the G_x gradient are used initially to move the position in the k-space to the starting position just before the read-out takes

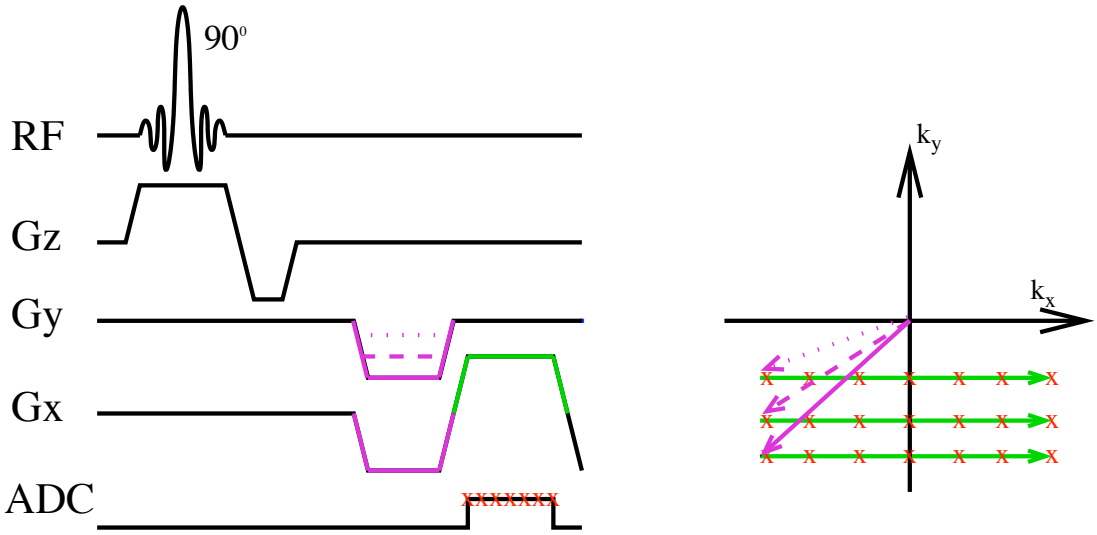


Figure 2.9: An example of a pulse sequence diagram (left) representing a sampling of a few different lines in the k-space (right). The five different lines of the pulse sequence describe five different sets of instructions for the RF coil and the gradient fields. One RF pulse is needed for each of the different lines of the k-space.

place (full magenta). After one line of the k-space is sampled, G_y gradient changes in intensity while all the other gradients and the RF pulse repeat the previous instructions in order to acquire the next line (dashed magenta). This process continues until the whole of the k-space is sampled. The fourth line is for the G_x gradient waveform. When this one is positive we are moving in the positive direction of the k_x-axis (and opposite when it is negative). It is when this gradient is non-zero that the signal read-out takes place. This is why it is known as the read-out gradient. And finally the fifth line is the Analogue-to-Digital Conversion (ADC) line and it is on this one that the signal read-out timings are specified (in red). The corresponding sampling points are shown in the k-space image on the right. Note that this is just a schematic representation of the sampling points and that during the scanning the number of these points per line is much larger, e.g. 64, 128. Also, the role of different gradients can change depending on the slice orientation during scanning (axial, sagittal or coronal) so G_z can become

the read-out gradient and G_x can become the slice-select gradient and so on.

The pulse sequence which was just described in Figure 2.9 is called a standard gradient echo (GE) pulse sequence. The term "gradient echo" comes from a signal property called an "echo". A signal echo is formed when, during the relaxation process while the spins are dephasing, the application of an RF pulse or a gradient field makes spins rephase causing a regrowth in the MR signal. Depending on whether an RF pulse or a gradient field was used for rephasing of the signal, the echo can be called a "spin" or "gradient" echo respectively. Figure 2.10 shows the collective behaviour of spins during a gradient echo.

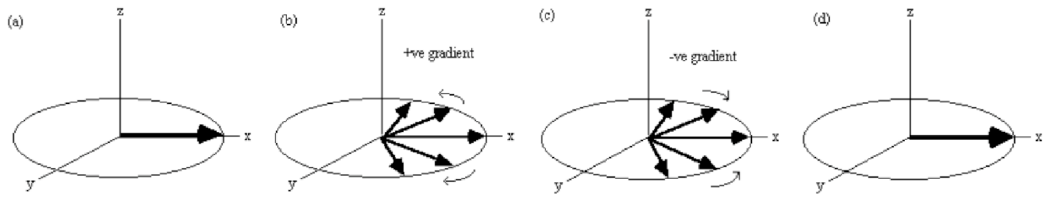


Figure 2.10: Gradient echo. Spins precess in the rotating frame and are all in phase (a). Once the gradient field is applied spins start dephasing as spins acquire different precessional speeds and go out of phase with each other (b). With a rapid switch of the gradient in the opposite direction spins refocus (c). After some time spins are again in phase (d). The time it takes from the moment the spins were excited to this moment when they are again in phase is called the echo time. Figure courtesy of Stuart Clare.

Echo-Planar Imaging (EPI)

In the following we will describe one of the most commonly used pulse sequences in FMRI: echo-planar imaging (EPI) [53]. EPI is a gradient-echo-based fast scan imaging technique as it allows whole volume images of the brain to be acquired in just a few seconds. Its great speed comes from its ability to sample an entire k-space matrix (a single slice) using only a single RF pulse (during the standard GE sequence as we saw in the previous example, one RF pulse is applied per

line). The EPI pulse sequence and the corresponding sampling of the k-space is schematically represented in Figure 2.11.

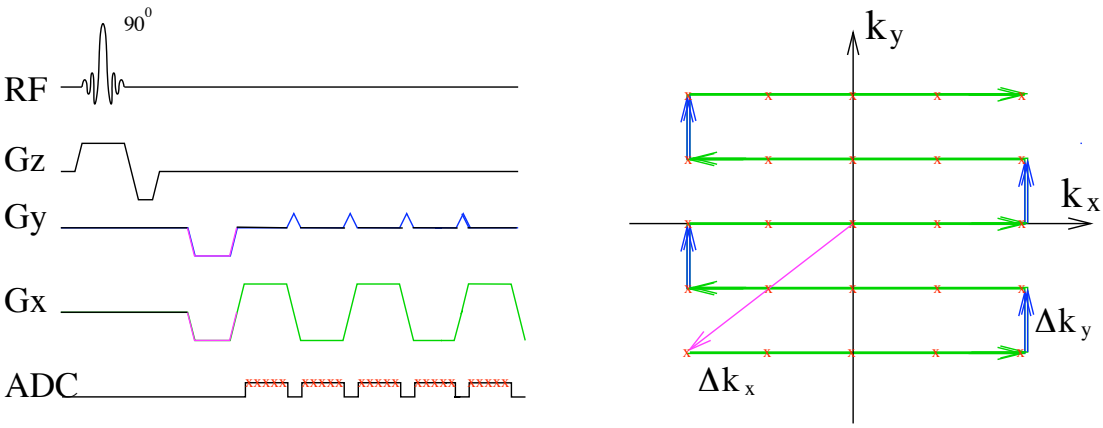


Figure 2.11: Schematic representation of the echo planar imaging (EPI) pulse sequence (on the left) and corresponding sampling of the k-space (on the right) for imaging of one slice. Only one RF pulse is used to sample the whole slice.

Read-out points in the k-space progress chronologically from the bottom left corner, following the arrows until the sampling ends at the top left corner. The sampling interval on the horizontal lines is $\Delta k_x = \frac{\gamma}{2\pi} \int_{t_n}^{t_{n+1}} G_x(t')\mathrm{d}t'$ and it corresponds to the area under the G_x gradient waveform between two successive read out time points. $\Delta k_y = \frac{\gamma}{2\pi} \int_{t_n}^{t_{n+1}} G_y(t')\mathrm{d}t'$ is the interval between two horizontal lines and is determined with the blips of the G_y gradient we see in Figure 2.11.

Once the sampling of the signal for one slice is finished, two dimensional reconstruction from the Fourier samples is performed in order to obtain the image of the slice. The carrier frequency of the RF pulse is then shifted in order to cover the range of frequencies of the next slice, and the new RF pulse is applied with the whole process of k-space sampling being repeated. The image of the whole brain (often called the MR volume) is the collection of all the slices. In a typical scanning session, the volume is acquired in a few seconds and usually 20 or more slices are imaged.

2.2.3 Scanning parameters

In this section we will discuss various scanning parameters. These are the parameters that can influence the image properties and quality and they can be controlled and adjusted by the operator. The scanning parameters can be divided into two groups: primary and secondary scanning parameters (Figure 2.12).

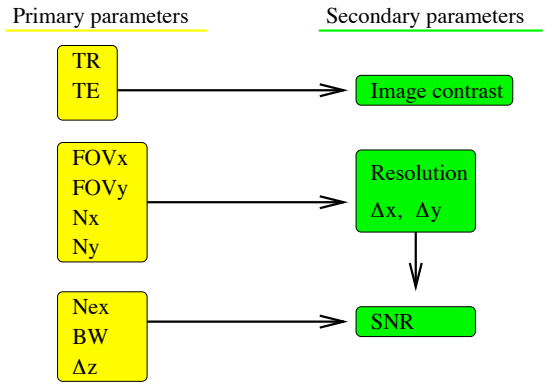


Figure 2.12: Scanning parameters. The primary parameters are set directly with the pulse sequence (left) while the secondary parameters are the ones that are derived from the primary parameters and describe the image (right). The idea for the figure was based on the division of the parameters done by Hashemi et al. [33].

The primary parameters are those parameters that are set directly into the pulse sequence. The secondary parameters are those that describe the image. They are derived from the primary parameters and are: image contrast, resolution and SNR. In this section we will describe each of these parameters.

Image contrast

Image contrast is an important imaging parameter as it helps to distinguish one brain tissue from another. It can be achieved because each of the brain tissues has different MR parameters: the spin density (often also referred to as the proton density), T_1 relaxation time and T_2 decay time. All three parameters contribute to the contrast, but usually one is emphasised. Figure 2.13 shows an example

MR image for each of these three different contrast types.

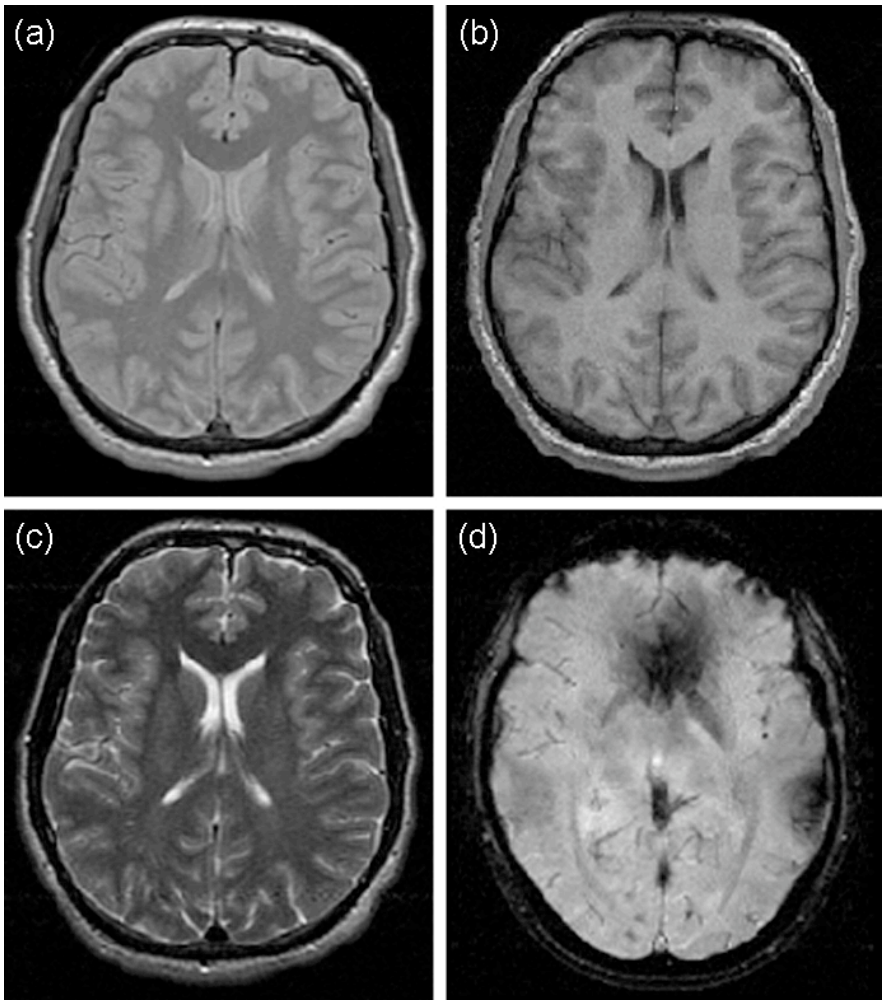


Figure 2.13: Different image contrasts of an axial brain slice at 3T: (a) PD-weighted image (SE spin-warp sequence with TR/TE = 4000/15 ms); (b) T_1-weighted image (GE FLASH sequence with TR/TE = 14/5 ms and flip angle = 20°); (c) T_2-weighted image (SE sequence with TR/TE = 4000/100 ms); (d) T_2^*-weighted image (GE FLASH sequence with TE = 30 ms). The figure courtesy of James Wilson.

Which MR parameter is emphasised depends on the primary parameters TE and TR used when acquiring the image.

TR (repetition time) is the time between two consecutive applications of an RF pulse.

TE (echo time) is referred to as the echo delay time. It describes a period in

time which passes between the RF pulse and the time when the signal is measured at the center of the read-out period.

By varying these two parameters, TR and TE, the contrast between different tissue types is altered. This is shown in Figure 2.14 where the values for T_1 and T_2 that tissues typically have are given.

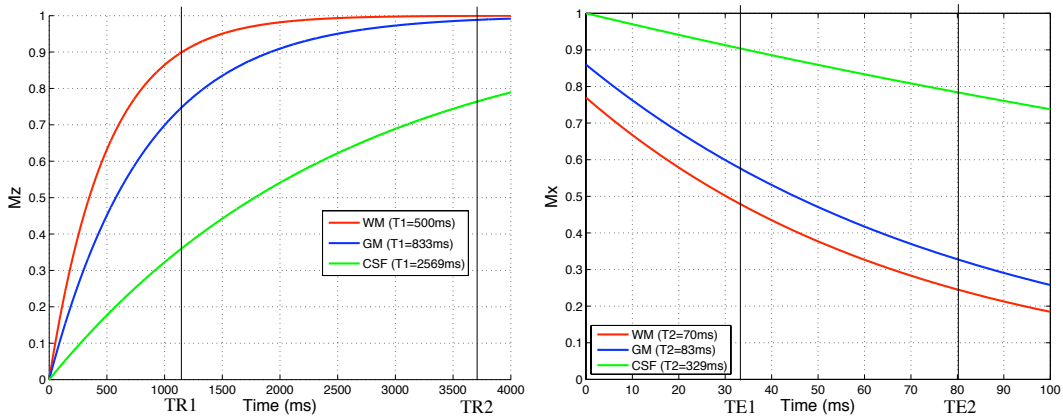


Figure 2.14: The figure shows the different rates by which signal coming from different tissues change at 1.5T. T_1 recovery and T_2 decay curves for white matter (WM), grey matter (GM) and cerebrospinal fluid (CSF) are shown. Whether the tissue types will show up bright or dark on the image is determined by the choice of TR and TE parameters.

The degree of T_1 recovery that has occurred after an RF pulse will determine how much longitudinal magnetization is available to be flipped by the following RF pulse. If nearly complete T_1 recovery is allowed to occur after an RF pulse (i.e. TR is long) and a signal is acquired very quickly after the next RF pulse (i.e. TE is short) then intensity will depend on proton density. Proton density is the starting point of the signal in the plot on the right of Figure 2.14. However, if TR is short (e.g. TR1 in this figure) and TE is short (e.g. TE1 in this figure), then intensity will depend on the T_1 recovery times of the different tissue types. Therefore, in a T_1-weighted image white matter is bright, grey matter is intermediate and CSF is dark. If TE is long (e.g TE2 in this figure) then

intensity will depend on the T_2 decay of the different tissue types. Therefore, in a T_2-weighted image CSF is bright, grey matter is intermediate and white matter is dark. There is also a third form of weighting called the T_2^*-weighting. T_2^* is often much shorter than T_2 as it describes the extra dephasing due to the external magnetic field inhomogeneities. Gradient echo sequences are used to obtain T_2^* weighting, as spin echo sequences will rephase the effects of the inhomogeneities.

Image resolution

Resolution is the minimum distance that we can distinguish between two points on an image [33]. In object-space, the spatial resolution of the image is represented by Δ_x and Δ_y. It depends on FOV_x, FOV_y, N_x, and N_y.

$\mathbf{FOV_x}, \mathbf{FOV_y}$ describe the Field Of View (FOV) of the image in the x- and y- directions respectively.

$\mathbf{N_x}, \mathbf{N_y}$ describe the size of the k-space matrix where N_x is the number of sampling points on a k-space line parallel to the k_x axis (number of samples in the k_x direction) and N_y is the number of lines (number of samples in the k_y direction).

Both field of view and the number of samples in the k-space determine the resolution of an image:

$$\Delta x = \frac{FOV_x}{N_x}, \qquad \Delta y = \frac{FOV_y}{N_y}. \tag{2.55}$$

FOV_x and FOV_y are determined by the pulse sequence. During the pulse sequence, the phase-encoding gradient, G_y, is applied for a period t_y and is incremented in N_y steps of ΔG_y from $-G_{y,max}$ to $+G_{y,max}$. During readout, the readout gradient has a fixed amplitude of G_x that is applied for time T. The sampling period of the analogue to digital converter (ADC) that samples the signal is Δt and N_x data points are sampled per readout.

In k-space, the sampling intervals are defined as Δk_x and Δk_y while the highest spatial frequencies sampled are $k_{x,max}$ and $k_{y,max}$. Using the above terms, the FOV of the image can be expressed as

$$FOV_x = \frac{1}{\Delta k_x} = \frac{1}{\frac{\gamma}{2\pi} G_x \Delta t}, \text{ and} \tag{2.56}$$

$$FOV_y = \frac{1}{\Delta k_y} = \frac{1}{\frac{\gamma}{2\pi} \Delta G_y t_y}, \tag{2.57}$$

while the spatial resolution of the image can be expressed as

$$\Delta x = \frac{1}{N_x \Delta k_x} = \frac{1}{2k_{x,max}} = \frac{1}{\frac{\gamma}{2\pi} G_x T}, \text{ and} \tag{2.58}$$

$$\Delta y = \frac{1}{N_y \Delta k_y} = \frac{1}{2k_{y,max}} = \frac{1}{\frac{\gamma}{2\pi} 2G_{y,max} t_y}. \tag{2.59}$$

The latter two equations imply that, along each axis of the image, the resolution size is the half-cycle width of the highest spatial frequency recorded.

Signal to Noise Ratio (SNR)

Signal to Noise Ratio (SNR) is a way of characterising the quality of the signal. It describes the proportion of the noise that is present in the image. The signal is primarily a function of the magnetisation M_0 available. The amount of magnetisation is proportional to the number of spins present within each voxel in the image. If the voxel size is increased (with either lower in-plane resolution or thicker slices), the voxel will contain more proton spins and therefore the signal will increase.

SNR also depends on the amount of time spent collecting the data. The signal increases linearly with the number of samples collected, whereas the noise increases only with its square root. This is because, provided that the samples are independent, the total variance when N measurements are combined σ_N^2, is equal to the sum of the N individual variances, all equal to σ^2. The noise σ_N will hence

be equal to $\sqrt{N}\sigma$, where σ is the noise correspondent to a single measurement. This can be summarised as follows:

$$SNR \propto \text{voxel volume}\sqrt{\text{acquisition time}}. \tag{2.60}$$

The acquisition time is proportional to the number of excitations and the read-out time (which is given by the number of points collected along the frequency-encode direction N_x multiplied with the sampling interval ΔT, and the number of phase-encode steps N_y), so the equation 2.60 becomes

$$SNR \propto V\sqrt{\frac{N_x N_y N_{ex}}{BW}}. \tag{2.61}$$

where V is the volume of the image voxel. A voxel is a three-dimensional volume element with dimensions Δx, Δy, and Δz and so its volume is calculated as $V = \Delta x \Delta y \Delta z$. Δx and Δy define the in-plane resolution while the Δz parameter specifies the slice thickness.

$\mathbf{N_{ex}}$ (Number of Excitations) is the number of times a scan has been acquired.

BW (Receiver Bandwidth) The receiver bandwidth is related to the sampling interval ΔT with the following equation [1]:

$$BW = \frac{1}{\Delta T}. \tag{2.62}$$

The sampling interval ΔT is the time that passes between the acquisition of the two adjacent points (in k_x direction) in k-space.

From the expression 2.61, it is possible to see what the impact of increasing image resolution is on the SNR of the images. For instance, if the voxel dimension along each axis is halved, the SNR decreases by a factor of 8. This effect can be

[1] In MR imaging two different kind of bandwidths are commonly used. One is the RF bandwidth (the transmission bandwidth). The transmission bandwidth is the range of frequencies that are "sent" to the head by the RF coil. This bandwidth determines the slice thickness. The other commonly used bandwidth is the receiver bandwidth.

compensated for by increasing the number of repeats. However, since the SNR only increases with the square root of the acquisition time, 64 averages would need to be acquired to maintain the same level of SNR. As it is in practice not feasible to prolong the scanning times in this way, other means of increasing the SNR need to be considered if high-resolution is to be attained (e.g. using a higher static magnetic field).

2.3 Physiological principles of BOLD imaging

FMRI is a method which investigates brain function in-vivo. As it typically has an in-plane resolution of 3mm or larger, it is concerned not with the behaviour of single neurons but with activity of large populations of neurons. The data produced is nonetheless highly informative, as single neurons do not work independently, but function together in large groups [40].

When the body is performing a particular task (finger tapping, listening to an audio stimulus, looking at changing colour etc.) certain parts of the brain become activated and neurons start exchanging information. Information transfer between neurons in the brain is an energy-requiring process, and because of this the places in the brain where this transfer is happening have greater demand for energy and hence a greater demand for oxygen. To meet this increased metabolic demand, neuronal activation is accompanied by increased local blood flow.

It has been observed that the additional influx of oxygen exceeds the amount extracted, therefore creating a higher concentration of oxygen in the blood locally compared to the resting state. Since oxygen is not very soluble in blood, it is transported bound to the large iron containing molecule, haemoglobin [40]. The form of the compound that contains oxygen is called oxyhaemoglobin and without oxygen is called deoxyhaemoglobin.

Deoxyhaemoglobin has different magnetic properties from the surrounding tissues and creates inhomogeneities which cause local dephasing and hence a reduction in the value of T_2^*. In a T_2^* weighted imaging experiment, this will cause a signal loss (a darkening of the image in those voxels containing the vessels). Due to activation the relative concentration of deoxyhaemoglobin decreases and therefore induces less signal loss, causing the signal intensity to increase. The parts of the images which show increased intensity should therefore correspond to the brain areas which were activated due to the task performance. This characteristic provides the basis for Blood Oxygenation Level Dependent (BOLD) FMRI (Figure 2.15).

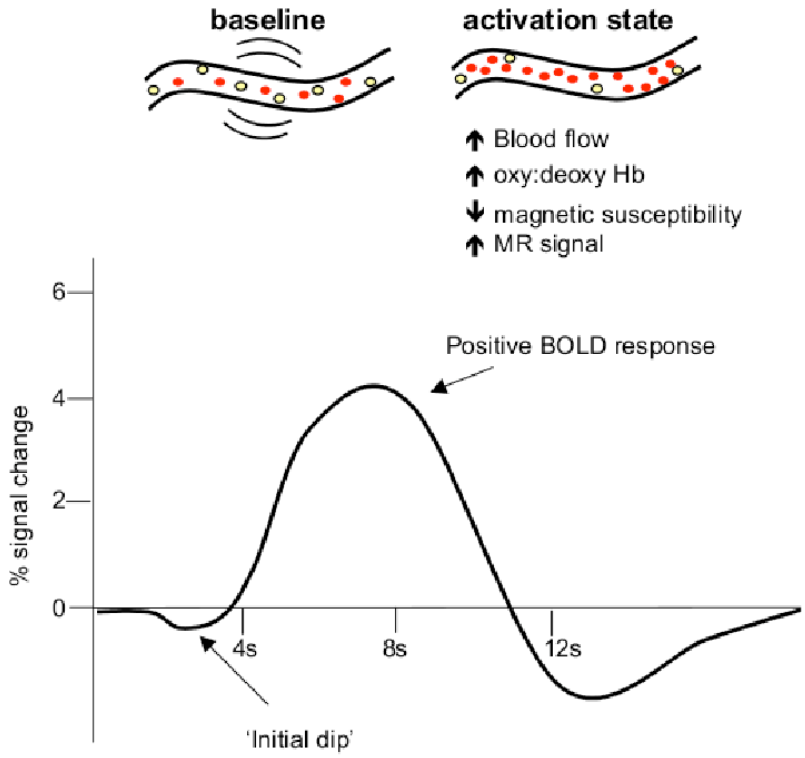


Figure 2.15: Blood Oxygenation Level Dependent (BOLD) signal. Figure courtesy of Heidi Johansen-Berg.

In order to be able to image BOLD contrast, the imaging sequences used must

produce T_2^* weighted images and therefore must be based on a gradient echo. In addition to this, BOLD FMRI needs to be performed with very fast imaging techniques to capture the temporal changes while still covering a substantial region of the brain. For these reasons, one of the most suitable, and most commonly used sequences in BOLD FMRI is EPI.

Fast imaging of BOLD-related inhomogeneities comes at the price of several artefacts. These are set out in the following section.

2.4 Imaging artefacts

2.4.1 Magnetic-susceptibility-induced B_0 inhomogeneities

Magnetic-susceptibility-induced B_0 inhomogeneities are caused by the magnetic susceptibility differences between different tissues types in a head [2]. The magnetic susceptibility is a property possessed by every substance and it describes the extent to which that substance can become magnetised when placed in a magnetic field. When an object consists of a few different substances which have different magnetic susceptibilities, magnetic field distortion will occur at the interfaces of these substances (such as tissue-air or tissue-bone interfaces).

One of the main remedies for this type of inhomogeneity is the use of magnet shimming coils which form a part of imaging hardware and are schematically represented in Figure 2.8. The currents in these coils are set at the beginning of each imaging experiment, usually based on measurements of the field, in order to counteract the inhomogeneities. Shimming, however, often can not appropriately correct for smoothly varying fields and hence does not full correct the

[2]Throughout this thesis magnetic-susceptibility-induced B_0 inhomogeneities are often just referred to as the B_0 inhomogeneities as they are the ones that this thesis is concerned the most.

inhomogeneities.

The B_0 inhomogeneity causes dephasing of spins i.e. a distribution of resonance frequencies which are different from the Larmor frequency associated with the static magnetic field. Inhomogeneities in the magnetic field are especially problematic in the EPI pulse sequence which is very sensitive to it due to the fact that an entire plane of k-space is sampled after just one RF pulse. The off-resonance effect creates a phase error and this propagates throughout the whole acquisition of the k-space without a chance to be rephased (as it is the case in spin-echo sequences) or to be nulled after acquisition of each k-space line (as is the case in standard gradient echo pulse sequences). The phase error is more visible in the phase-encode direction (k_y) because during the EPI scanning it takes much longer to sample in this direction than in the read-out direction which gives more time for the phase error to accumulate. This creates large translations (up to several voxels) along the y-axis for the affected voxels in the output image and is known as "geometric distortion artefact".

An example of the distorted area of an EPI image, due to magnetic-susceptibility-induced field inhomogeneity, is shown in Figure 2.16 (a). Another susceptibility artefact is referred to as "signal loss artefact". It is due to a local decrease in image signal in regions that exhibit large intrinsic B_0 gradients in the slice-select direction. An example of signal loss artefact is shown in Figure 2.16 (b).

2.4.2 Rigid-body motion

Rigid-body motion is an important issue in FMRI analysis as even the slightest patient motion (less than a degree of rotation or a mm translation) during a scan can mean that a voxel location does not correspond to the same physical point in the volume as the same voxel in a subsequent volume, given a spatial resolution

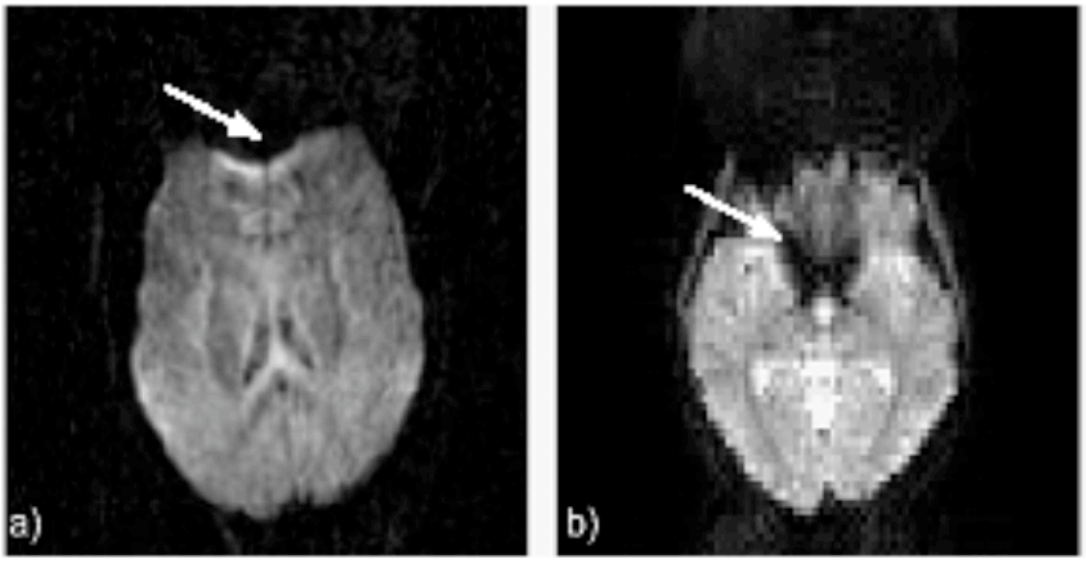


Figure 2.16: Magnetic-susceptibility-induced artefacts in an EPI image: geometric distortion (a) and signal loss (b). The phase encoding was in the y-direction.

of a few mm in the acquisition process (2.17). In addition, the change in signal intensity due to motion can be far greater than the BOLD effect itself (e.g. 5-10% change in the signal due to motion and 1-2% change in the signal due to BOLD), particularly at tissue boundaries, at the edge of the brain or near major ventricles.

In addition to position displacement, there are various artefacts that can occur in images as a consequence of motion. Figure 2.18 gives a descriptive representation of artefacts associated with each of the different motion parameters.

When a head moves in a field which has susceptibility-induced inhomogeneities two different types of B_0 distortions can occur. Type 1 distortion occurs when the magnetic field inhomogeneities move together (aligned) with the object (each voxel will continue experiencing the same magnetic field although the object is moving). This happens when the structure and the position of the tissue-air boundaries in the object does not change (with respect to the direction of the B_0

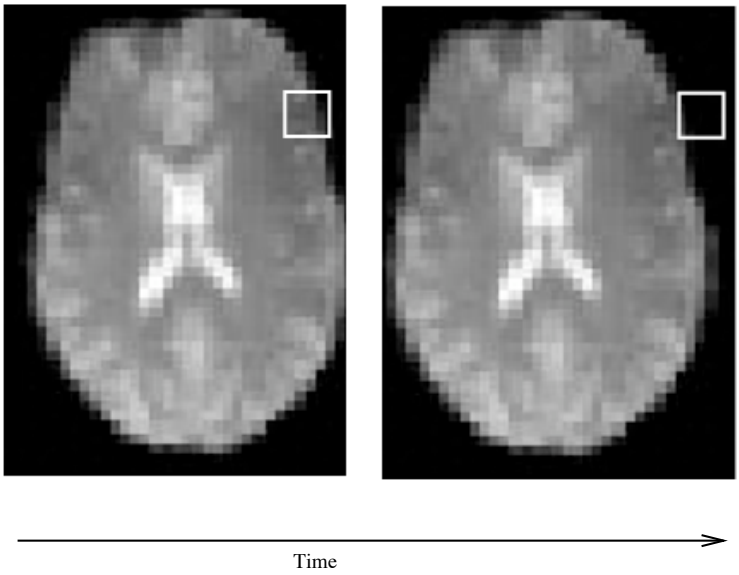


Figure 2.17: Due to the motion artefact (position displacement) a brain region of one acquired MR volume will not correspond to the same brain region in the following volume.

field) with object's motion. An example of this is when the object rotates about the z-axis of the scanner coordinate system. On the other hand when the object rotates about the x- or y-axis of the scanner coordinate system, the tissue-air boundaries within the object change with respect to the direction of the B_0 field. This creates type 2 distortion which results in the magnetic field changes as the object moves (i.e. they do not move together any more). More detail about both types of these inhomogeneities can be found in Andersson et.al.[2].

Spin-history effects (Figure 2.19) occur for all motion that causes movement in the through-slice direction (translation in z, rotation about x, rotation about y). It occurs because if a spin gets excited by the RF pulse during the acquisition of one volume and then the head moves in the z-direction, the frequency of the precession of the spin will change possibly matching the frequency of the RF pulse which was planned to happen for the following slice. The following RF pulse will then excite that same spin again sooner (or later) than when the planned RF

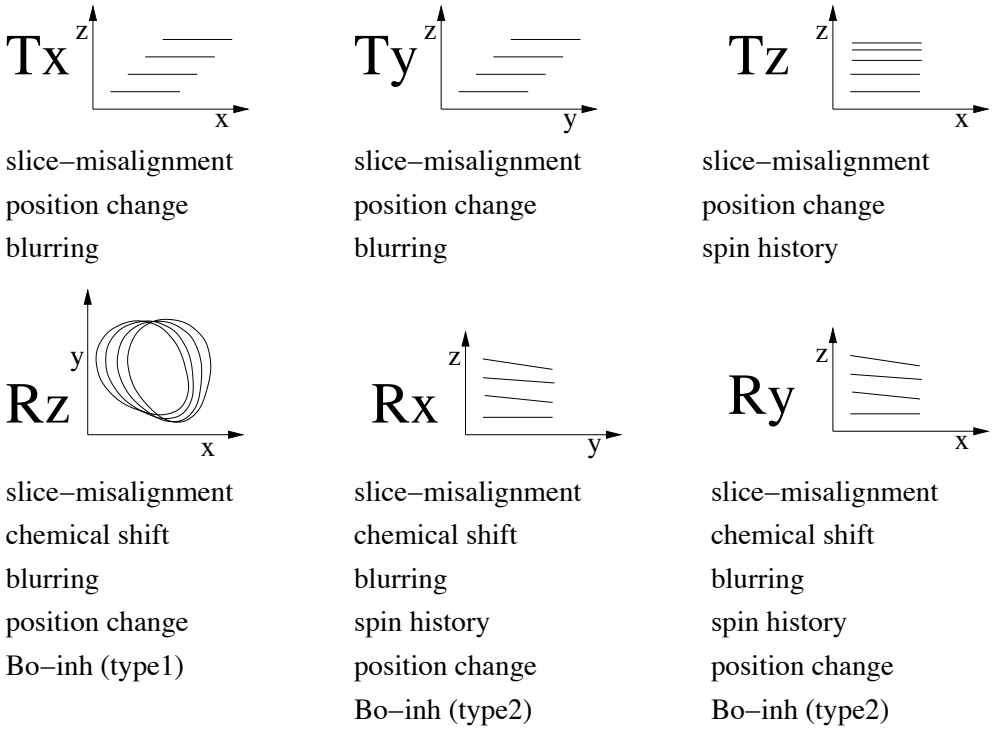


Figure 2.18: A description of the main motion-related artefacts in FMRI data. Each of the motion parameters (translations T_x, T_y, T_z and rotations R_x, R_y, R_z) has its own set of artefacts and these are shown in the figure. Plots show the type of the slice-misalignment created with different motion parameters. The artefacts are listed for the case when the phase encode is in the y-direction.

pulse was supposed to happen. This will result in a spike in the image intensity.

Another very prominent effect of motion is slice misalignment which occurs in a different fashion for each of the motion parameters. Plots in Figure 2.18 descriptively show how this occurs. In the case of motion in plane and during the read-out window, blurring also occurs (Figure 2.20).

2.4.3 Chemical shift

The chemical shift artefact is caused by differences in precessional frequencies of protons from different molecules. Two such molecules in the head are water and fat. Protons found in water precess slightly faster than those ones found in

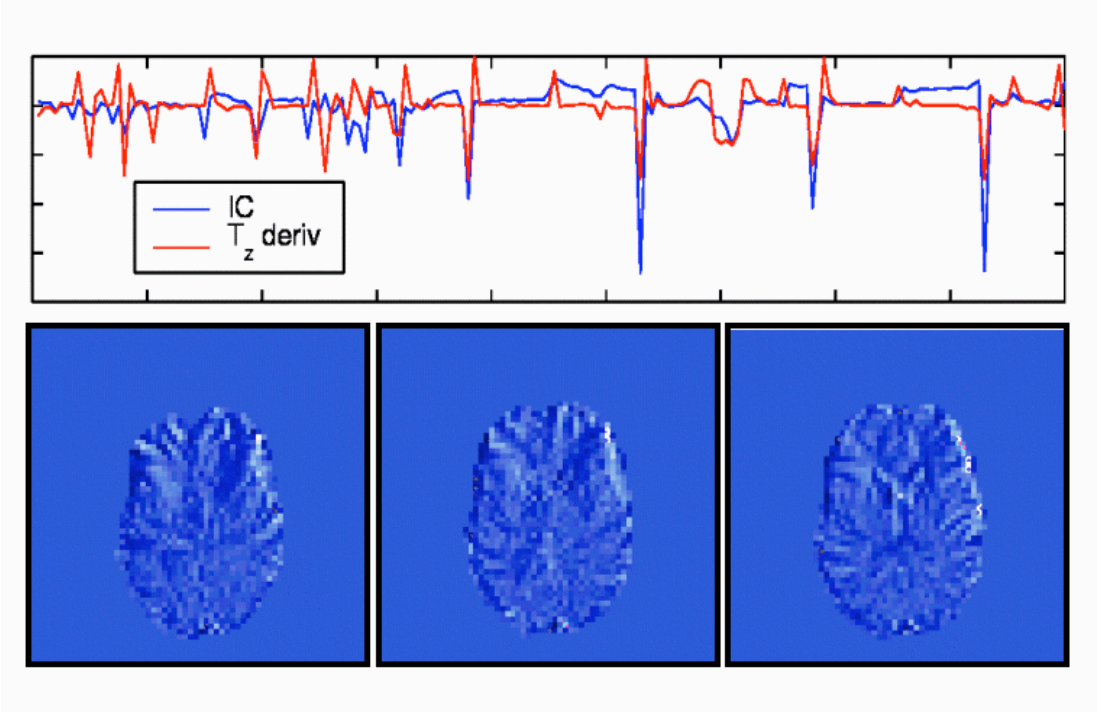


Figure 2.19: Spin history artefact identified as an independent component (IC) by the use of Independent Component Analysis (ICA). T_z represents translation in the z-direction.

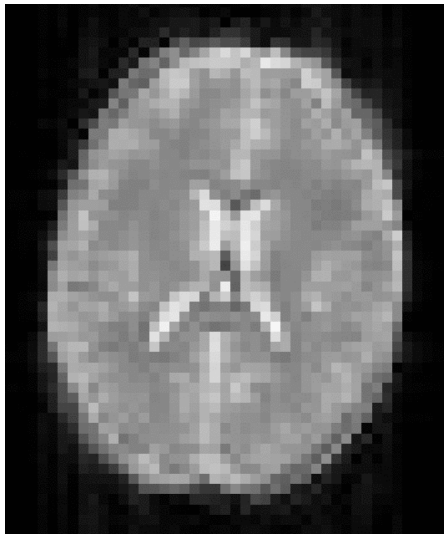


Figure 2.20: Blurring artefact caused by the rotation about the z-axis.

fat. Similarly, as in the case of B_0 inhomogeneities, this is called an off-resonance effect and it will create the same kind of artefacts as the one caused by the B_0 inhomogeneity (translation and signal loss). An example of chemical shift artefact is shown in Figure 2.21 where we can seen how the fatty tissue is shifted by many voxels from its original position.



Figure 2.21: Chemical shift artefact in an EPI image. The fatty tissue is shifted by many voxels from its original position and appears as a slightly shifted white ring around the image.

The main difference between the chemical shift artefact and the artefacts caused by the magnetic inhomogeneities is in the way they interact with the rigid-body motion of the head. While the artefacts due to the magnetic field inhomogeneities interact with motion (as described in the previous section: B_0 inhomogeneity type 2), the amount of distortion due to chemical shift does not change when the head moves.

2.4.4 Eddy Currents

Eddy currents are small electric currents that are generated when the gradient coils are rapidly switched on or off [33]. This is a result of the electromagnetic induction due to the rapid changing of the magnetic field. In the presence of eddy currents, the shape (profile) of the gradient applied will be different from the shape of the gradient waveform supplied by the gradient amplifier. The difference between the desired and the actual gradient shape is shown in Figure 2.22.

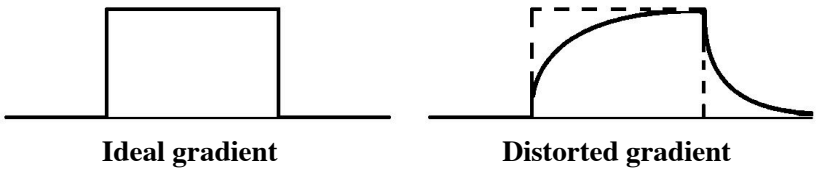


Figure 2.22: Eddy currents cause the shape of the gradient to change. The shape supplied by the gradient amplifier (left) and the actual shape of the applied gradient due to the eddy currents (right, solid line).

Examples of a few different eddy current effects are shown in Figure 2.23. The main difference between the eddy currents applied in each of these images is their spatial orientation.

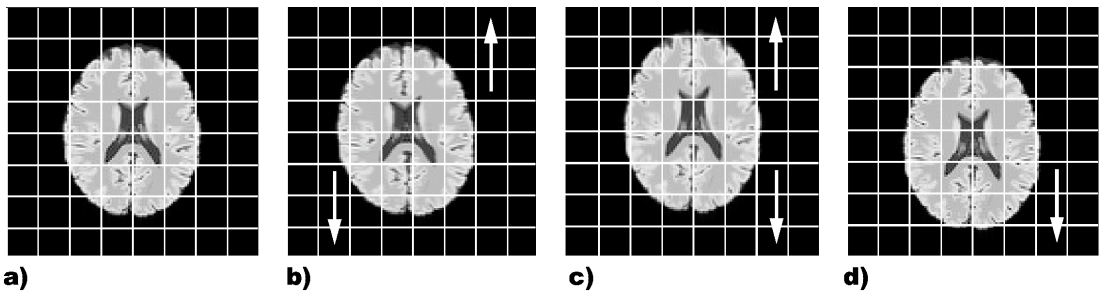


Figure 2.23: Geometric distortions induced by eddy currents in EPI images. **a)** Undistorted base image; **b)** shearing induced by an eddy current field along the read direction; **c)** scaling due to a phase-encode eddy current field; **d)** translation due to the presence of spatially invariant eddy current. The read out direction is horizontal, while the phase encode direction is vertical on these images. Figure and most of the caption courtesy of Rita Nunes.

2.5 Summary

This chapter has described the process of MR image formation. It started by introducing magnetic moments in an object and then traced step-by-step how they are converted to a magnetisation vector $\mathbf{M}$, a transverse magnetisation $\mathbf{M}_{xy}$, an electrical signal $S(t)$, a k-space signal $S(k_x, k_y)$ and then finally, via the IDFT, into the desired image.

This chapter also described the basic physiology of FMRI, especially blood oxygen level dependent (BOLD) FMRI and introduced EPI – the most commonly used pulse sequence in this type of image acquisition.

The goal of this chapter is to give the background required for the next chapter in which we describe the mathematical model which simulates the whole FMRI process.

Chapter 3

Simulator model development

The following three chapters describe the main steps in the development of the FMRI simulator including the development of the model, implementation of the model into the simulation software and ultimately validation of this software.

The model behind the simulator is based on the Bloch equations which we described in more detail in Chapter 2. There have been other simulators based on solving the Bloch equations and these, together with the other simulation strategies, are described in the first section of this chapter. However, the unique characteristics of our simulator are its capability to simulate realistic motion artefacts (such as spin-history and blurring), artefacts produced by the magnetic field inhomogeneities, and the BOLD signal in the FMRI data (these concepts are discussed in the previous chapter).

This chapter's main goal is to describe how the Bloch equations were appropriately adapted and solved so that they can include all of the mentioned characteristics and artefacts: motion-related artefacts, magnetic field inhomogeneities, eddy currents artefact, and thermal noise.

Most of the material covered in the following three chapters is published in Drobnjak et al. [22].

3.1 Existing simulation strategies

Realistic simulation of FMRI data is a difficult task because of the complex nature of the data itself. Firstly, each subject has a unique anatomy and their own levels of haemodynamic responses, physiological noise (from such sources as respiration, cardiac pulsation and low-frequency networks) and degree and range of motion. Secondly, each MR scanner has its own behaviour in terms of instrumentation drifts, performance of the shims[1] and the RF and gradient coils. Thirdly, the FMRI experimenter can choose a different functional stimulus according to what he or she wants to investigate. These different stimuli will have different presentation rates and modalities. The MR operator then needs to select a pulse sequence and protocol. Combining all of these features into a single simulation is a challenging task which, in general, has not been undertaken for functional MRI. Most of the simulators created to date cover a few of these features but none try to combine all of them. This section provides an overview of existing simulation strategies.

One approach is based on k-space formalism [62]. They use a Fourier transform to generate the ideal MR signal from a user defined object. The k-space trajectory given by a pulse sequence is calculated and used to select elements from the ideal MR signal. This selection of elements mimics the sampling of the signal in an actual MRI scan. These methods are very useful in developing new pulse sequences and are computationally highly effective. However, they are very limited when it comes to modelling more complex image artefacts (e.g. B_0 inhomogeneities, and complex motion-induced effects such as spin history and blurring).

Another group of simulators is based on the steady-state solutions of the

[1] A hardware system designed to make the magnetic field within the scanner homogeneous.

Bloch equations for the well known pulse sequences [51], [65], [69], [30]. These simulations are based on knowledge of T_1, T_2 and proton density. From this, the synthetic images are calculated pixel by pixel. The imaging parameters (i.e. TR, TE, TI, etc.) are usually provided by the user. This type of simulator is very fast and useful for teaching and training purposes. However, as in the case of the k-space formalism, they are very limited when it comes to modelling more complex artefacts.

More extensive and sophisticated approaches are based on generating the MR signal by solving the Bloch equations. These approaches were initiated by Bittoun et.al. [10] and Summers et.al. [68] who solved the Bloch equations for every selected point in the object obtaining a magnetisation vector for each, and summing all of the magnetisation vectors in order to obtain the total signal for the object. A similar approach was later applied by Olsson et.al. [59] who also introduced more complete modelling for the dephasing of spins which allowed them to model the effects of T_2^* dephasing and spin echoes. Although this approach is very advanced and allows for a variety of image artefacts to be modelled, its major drawback is that it is very intensive in terms of computation and memory. Because of this, most of the objects used at the time were simple geometric shapes, either one- or two- dimensional, and images produced were of low resolution.

Kwan et.al. [46] tried to overcome the problem of computational speed and memory by an approach that used tissue templates. These templates are three-dimensional anatomical images of distinct tissue types (e.g. one template of grey matter, another of white matter and a third of cerebro-spinal fluid (CSF) as shown on Figure 3.1). A Bloch equation is then solved for each template, providing a magnetisation vector for each tissue type. A value describing the proportion of tissue present is given to each of the cuboid volume elements (or 'voxels') into which the template is divided. The magnetisation vector for a tissue type is then

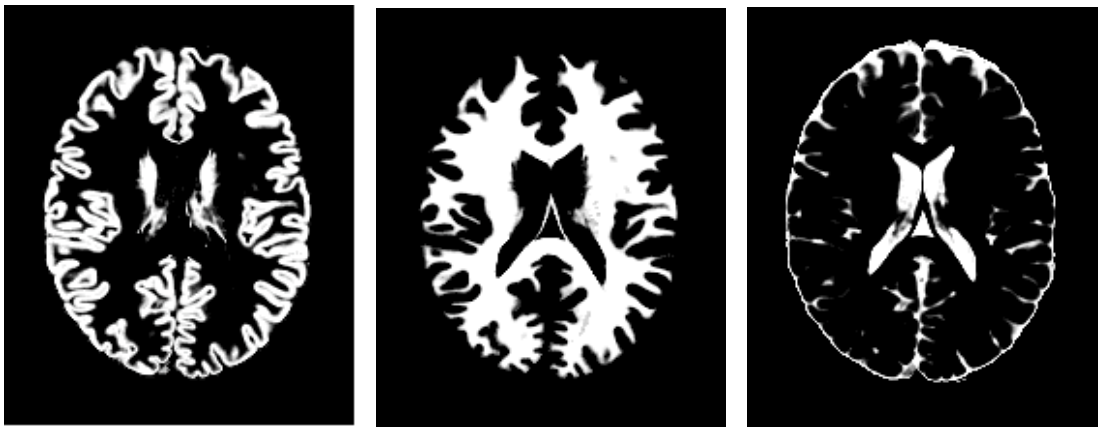


Figure 3.1: Cross-sections of the tissue templates derived from the anatomical images of different tissue types [19]. Shown are grey matter, white matter and CSF (ordered from left to right).

multiplied by the proportion of that tissue type within each object voxel. This is performed for each tissue type and the overall magnetisation vector for a given voxel is obtained by summing each of the weighted magnetisation vectors. In this way the Bloch equation is solved only as many times as there are tissue types instead of solving it for every object voxel as in previous simulators. Although computationally very fast, this approach is restrictive in terms of object voxel properties as it does not allow for different object voxels to experience different susceptibility induced magnetic field inhomogeneities so several artefacts due to rigid body motion or magnetic inhomogeneities cannot be modelled. The approach by Kwan et.al. was used to generate an on-line simulated brain database [17] which to date has been one of the most widely used resources of simulated MRI data. A collaboration has been formed with some of the original authors which is exploring ways of expanding this database to include the FMRI images generated by the FMRI simulator described in this thesis [54].

In order to be able to model B_0 inhomogeneities, Yoder et.al. [74] and Cattin et.al. [9] use an approach that is similar to the early simulators by solving the

Bloch equations for each object voxel. Yoder et.al. [74] introduced this as a way to model the B_0 susceptibility artefacts and used the Boundary Element method to estimate the perturbation field. In Cattin et.al. [9] parallelisation techniques were used to speed up the computational process, and they introduced a way to model spin echoes.

However, all of the above simulators simulate MRI images as opposed to FMRI time series, with the exception of Kwan et.al. [45] who described preliminary work in modelling BOLD FMRI, but did not take the work further. Most of the current FMRI simulations use simple techniques [23, 24, 27, 52, 72], most commonly taking existing MRI data for a given subject brain and artificially imposing on that data an activation time-series i.e. time-series describing the behavior of areas which got activated due to an external stimulus. The activation time-series are assigned to pre-specified image voxels in the image of the brain. The location of these voxels is determined either using the results of previous studies [23] or manually selected regions of interest [24, 27, 52, 72].

Furthermore, all of the above simulators are developed for producing motionless images and none simulate motion and motion artefacts such as spin-history, within-scan blurring or distortions due to B_0-movement interaction.

The main goal of the work described in this thesis was to develop a flexible and extensive FMRI simulator which is capable of simulating a realistic MR scanner environment together with realistic subject properties. These include eddy current effects, RF coil inhomogeneities, thermal noise, the anatomical model of the head, realistic rigid-body motion of the head, BOLD-induced physiological noise, and tissue-dependent magnetic field inhomogeneities. The remaining part of this chapter deals with development of the model of the simulator.

3.2 Signal Model

The model in the simulator is built around the Bloch equations. These equations were described in detail in the previous chapter, where the solutions for excitation and relaxation events in ideal scanning conditions were shown. In the next few sections we show how we further expanded on these solutions in order to simulate scanning in less ideal conditions (e.g. when the subject is moving). Furthermore, many of these conditions change in time and can influence a number of consecutive images in a time sequence (e.g. spin-history artefacts). The respective magnetization vectors of each object voxel are tracked and updated independently by solving the Bloch equations for each voxel at each time point. With this method every change in scanning conditions (e.g. motion, eddy currents etc.) is accounted for when simulating the magnetisation vector of each voxel. The total signal is calculated as the sum of all the magnetisation vectors across the object at each time point (note that the magnetisation vectors are weighted as a consequence of the inhomogeneities in the RF receiver coil which receives signal in a non-uniform fashion). This property allows the simulator to model very realistic scanning conditions and is one of the main differences between this type of simulator and the ones based on k-space formalism or the steady-state solutions of the Bloch equations. Figure 3.2 depicts the process of the simulation from inputs into the Bloch equation at the top to the outputs and subsequent modifications of the outputs at the bottom. Model for each of the stages of the simulation process are described in this chapter. Implementation details for each of the stages are described in the following chapter.

The Bloch equation

$$\frac{\mathrm{d}\mathbf{M}}{\mathrm{d}t} = \gamma\mathbf{M} \times \mathbf{B} - \frac{M_x\hat{\mathbf{x}} + M_y\hat{\mathbf{y}}}{T_2} - \frac{(M_z - M_0)\hat{\mathbf{z}}}{T_1}\,, \tag{3.1}$$

describes the time evolution of the magnetisation vector $\mathbf{M} = M_x\hat{\mathbf{x}} + M_y\hat{\mathbf{y}} + M_z\hat{\mathbf{z}}$

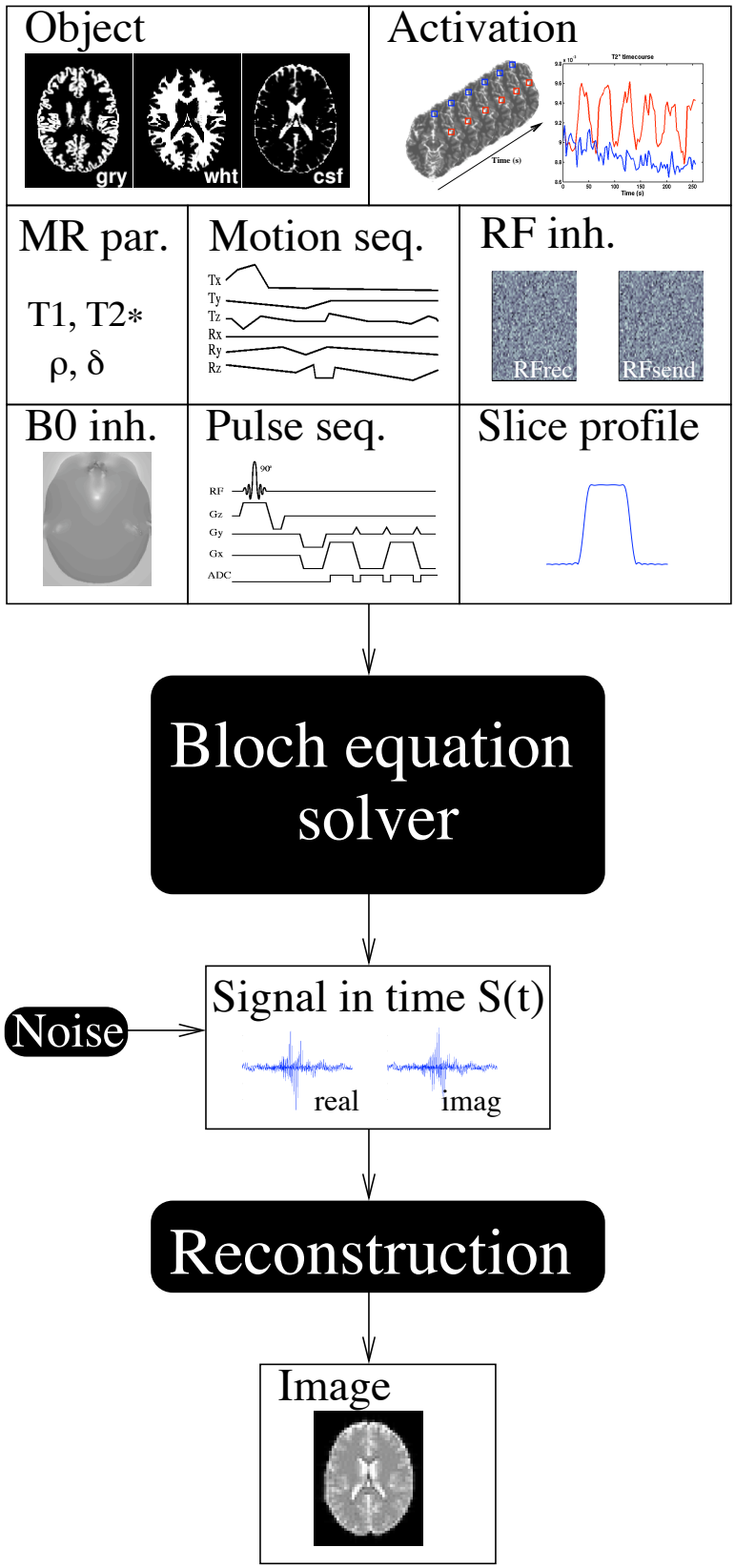


Figure 3.2: Descriptive representation of the simulation process. The black box in the middle of the image represents the 'brain' of the simulator - the Bloch equation solver. Images represented on the top (object, pulse sequence, motion sequence etc.) are inputs to the solver. The output of the solver is the signal $S(t)$ to which the thermal noise is added. The final stage is the reconstruction of the signal into the image. More details are described in the main text of this and the following chapter.

in the presence of the magnetic field $\mathbf{B} = B_z\hat{\mathbf{z}}$. The constant γ is the gyromagnetic ratio (42.6 MHz/T for hydrogen nuclei), T_1 is a relaxation time associated with the exponential recovery of the longitudinal component of the magnetisation vector $M_z\hat{\mathbf{z}}$ towards its equilibrium value $M_0\hat{\mathbf{z}}$, and T_2 is a relaxation time associated with the exponential decay of the transverse component of the magnetisation vector $M_x\hat{\mathbf{x}} + M_y\hat{\mathbf{y}}$ to its equilibrium value of zero. The magnetic field is represented in the simulator as

$$\mathbf{B}(\mathbf{r}, t) = B_0\hat{\mathbf{z}} + B_p(\mathbf{r})\hat{\mathbf{z}} + (\mathbf{G}(t) \cdot \mathbf{r})\hat{\mathbf{z}}\,, \tag{3.2}$$

where B_0 is the main static magnetic field, $B_p(\mathbf{r})$ is due to the susceptibility induced B_0 inhomogeneities, $\mathbf{G}(t)$ represents the gradient fields applied by the gradient system of the scanner, and $\mathbf{r} = x\hat{\mathbf{x}} + y\hat{\mathbf{y}} + z\hat{\mathbf{z}}$ is the spatial coordinate.

The object (brain) to be imaged is modelled by a collection of object voxels (small cuboid elements as described earlier in the chapter) with physical dimensions of (L_x, L_y, L_z) which have typical values $L_x = L_y = L_z = 1$mm. These voxels are usually much smaller (often hundreds or thousands of times smaller) than the voxels of the output image that is generated e.g. $1\times1\times1$mm object voxels are 64 times smaller than typical EPI $4\times4\times4$mm image voxels; $0.1\times0.1\times0.1$mm object voxels are 64000 times smaller than typical EPI $4\times4\times4$mm image voxels. Each object voxel is assumed to contain a mixture of tissues, each of which are uniformly distributed across the object voxel spatially. This implies that, for each tissue type, all of the properties (relaxation times T_1, T_2, and the spin density ρ), are constant over the object voxel. It is important to note here that this simulator does not model local T_2 dephasing due to the 'spin-spin' interactions separately from additional dephasing of the magnetisation introduced by field inhomogeneities (sometimes referred to as T_2' decay [29]). Therefore the loss of transverse magnetisation due to T_2' is not 'recoverable' in this model and spin-echo-based pulse sequences cannot be simulated accurately. In order to make this

clear, T_2^* (which includes both T_2 and $T_2^{'}$) will be used instead of T_2 throughout the rest of the thesis.

For the moment consider the case where the object (and hence all object voxels) are assumed to be motionless. In order to evaluate the signal during an FMRI sequence we solve the Bloch equation for each tissue type separately *and* (unlike in the tissue template method) for each of the object voxels. The total signal is calculated as the sum of all the magnetisation vectors from all the object voxels at that time. The advantage of this method over the tissue template method is that it enables magnetisation vectors for each object voxel to be evaluated even where there is motion of the object.

During the relaxation period, and the period of the signal acquisition, the RF field is not present and the magnetisation vector processes about the z-axis with frequency $\omega(\mathbf{r},t) = -\gamma(B_0 + B_p(\mathbf{r}) + (\mathbf{G}(t)\cdot\mathbf{r}))$. In the rotating reference frame the closed form solutions during this process are:

$$\begin{aligned} M_z(\mathbf{r},t) &= e^{-(t-t_0)/T_1} M_z(\mathbf{r},t_0) + (1 - e^{-(t-t_0)/T_1}) M_0 \,, \\ M_{xy}(\mathbf{r},t) &= e^{-(t-t_0)/T_2^*} |M_{xy}(\mathbf{r},t_0)| e^{-i\phi(\mathbf{r},t)} \,, \end{aligned} \tag{3.3}$$

where $M_{xy} \equiv M_x + iM_y$, the phase $\phi(\mathbf{r},t)$ is

$$\phi(\mathbf{r},t) = \gamma \int_{t_0}^{t} (\mathbf{G}(t)\cdot\mathbf{r} + B_p(\mathbf{r}))\mathrm{d}t \,, \tag{3.4}$$

t_0 is the time of the last RF pulse, and initial magnetisation at t_0: $M_z(\mathbf{r},t_0)$ and $M_{xy}(\mathbf{r},t_0)$ is solved from the previous excitation period as represented in the following.

We assume that the excitation period is significantly shorter than the relaxation times T_1 and T_2^*, as it is in most pulse sequences, and therefore model the excitation as happening instantaneously. These assumptions allow the effect of an RF pulse to be represented as a simple rotation through a flip angle α about

an axis (e.g. the x-axis) which can be conveniently represented in matrix form as:

$$\mathbf{M}(\mathbf{r}, t_0) = R_x(\alpha)\mathbf{M}(\mathbf{r}, t_{0-}) \,, \tag{3.5}$$

where t_{0-} is the time just prior to the excitation pulse and

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{bmatrix} \,, \tag{3.6}$$

is the rotation matrix about the x-axis. The choice of the axis of rotation depends on the user and the RF field specification, and the x-axis is given just as an example.

The actual flip angle of the magnetisation vector for each object voxel is calculated as a proportion of the desired flip angle. The desired flip angle is the one specified by the pulse sequence. Ideally all of the voxels in the slice should be flipped by this angle. However, due to the fact that the RF pulses are not infinite (in time) that is not the case, and some of the voxels will not experience the desired flip angle but an angle that is slightly smaller (or bigger). By solving the Bloch equation during the excitation period it is possible to determine this angle for each voxel and get the "slice profile" of the excitation. For example, slice profile for the ideal case when all of the voxels get flipped by the same angle would be reactangular while a more realistic slice profile would have side lobes and ripples (Figure 4.3 shows examples for these two different slice profiles).

The simulator uses a pre-calculated slice profile as an input but it is possible to expand the model for RF pulses within the simulator by solving the Bloch equations fully, over the excitation period. We are planning to address this in future research.

3.3 Magnetic-susceptibility-induced B_0 inhomogeneities

B_0 inhomogeneities occur at interfaces of materials with different magnetic susceptibilities, such as tissue-air interfaces. These differences lead to distortion in the local magnetic field, as Maxwell's equations need to be satisfied. This is especially problematic in the EPI pulse sequences which are one of the most common sequences used in FMRI. Due to the long read-out times, EPI sequences are very sensitive to the inhomogeneities in the magnetic field which can result in signal loss and geometric distortion in the output images.

In order to evaluate the distortion in the magnetic field due to the susceptibility differences, a perturbation method for solving Maxwell's equations is used [37]. For more details see Appendix A. This method requires that the magnetic susceptibility at each point in the object is known. Note that, due to the major magnetic susceptibility differences between air and other tissue, it is important to have a good segmentation of the air spaces within the head.

When the object is placed in the scanner, the difference in susceptibility values of the air, χ_0, and the tissues in the object, $\chi(\mathbf{r}) = \chi_0 + \delta\chi_1(\mathbf{r})$, creates a perturbation in the field and the total magnetic field can be expressed as:

$$\mathbf{B}(\mathbf{r}) = \mathbf{B}^{(0)}(\mathbf{r}) + \delta\mathbf{B}^{(1)}(\mathbf{r}) + o(\delta^2)\,, \tag{3.7}$$

where $\mathbf{B}^{(0)}(\mathbf{r})$ is the induced homogenous magnetic field if only air were present in the scanner, $\delta\mathbf{B}^{(1)}(\mathbf{r})$ is the perturbation field which was in the previous sections denoted as $B_p(\mathbf{r})$ and δ is a constant equal to the difference between the susceptibilities of tissue and air. $\chi_1(\mathbf{r})$ shows the relative amounts of tissue and air at position $\mathbf{r}$. It is equal to zero, when nothing but air is present at $\mathbf{r}$ (no tissue), or one when only tissue is present at $\mathbf{r}$ (no air). Values between zero

and one leave open the possibility of 'partial volume' susceptibility values, which are used for object voxels that are part air and part tissue. When tissues exist with different susceptibility values, χ_1 represents the susceptibility values weighted by partial volume, see [37]. The first order term $\mathbf{B}^{(1)}(\mathbf{r})$ in this expansion is calculated from Maxwell's equations which assuming the object is nonconductive can be reduced to a single equation: $\nabla \cdot \mathbf{B}(\mathbf{r}) = 0$. The solution $\mathbf{B}^{(1)}(\mathbf{r}) = (B_x^{(1)}(\mathbf{r}), B_y^{(1)}(\mathbf{r}), B_z^{(1)}(\mathbf{r}))$ is given by the following equations:

$$\begin{aligned} B_x^{(1)}(\mathbf{r}) \simeq & \frac{\chi_1}{1+\chi_0} B_x^{(0)}(\mathbf{r}) - \frac{1}{1+\chi_0} \left(\left(\frac{\partial^2 G(\mathbf{r})}{\partial x^2} \right) * \left(\chi_1 B_x^{(0)}(\mathbf{r}) \right) + \right. \\ & \left. \left(\frac{\partial^2 G(\mathbf{r})}{\partial y \partial x} \right) * \left(\chi_1 B_y^{(0)}(\mathbf{r}) \right) + \left(\frac{\partial^2 G(\mathbf{r})}{\partial z \partial x} \right) * \left(\chi_1 B_z^{(0)}(\mathbf{r}) \right) \right) , \quad (3.8) \end{aligned}$$

$$\begin{aligned} B_y^{(1)}(\mathbf{r}) \simeq & \frac{\chi_1}{1+\chi_0} B_y^{(0)}(\mathbf{r}) - \frac{1}{1+\chi_0} \left(\left(\frac{\partial^2 G(\mathbf{r})}{\partial x \partial y} \right) * \left(\chi_1 B_x^{(0)}(\mathbf{r}) \right) + \right. \\ & \left. \left(\frac{\partial^2 G(\mathbf{r})}{\partial y^2} \right) * \left(\chi_1 B_y^{(0)}(\mathbf{r}) \right) + \left(\frac{\partial^2 G(\mathbf{r})}{\partial z \partial y} \right) * \left(\chi_1 B_z^{(0)}(\mathbf{r}) \right) \right) , \quad (3.9) \end{aligned}$$

$$\begin{aligned} B_z^{(1)}(\mathbf{r}) \simeq & \frac{\chi_1}{1+\chi_0} B_z^{(0)}(\mathbf{r}) - \frac{1}{1+\chi_0} \left(\left(\frac{\partial^2 G(\mathbf{r})}{\partial x \partial z} \right) * \left(\chi_1 B_x^{(0)}(\mathbf{r}) \right) + \right. \\ & \left. \left(\frac{\partial^2 G(\mathbf{r})}{\partial y \partial z} \right) * \left(\chi_1 B_y^{(0)}(\mathbf{r}) \right) + \left(\frac{\partial^2 G(\mathbf{r})}{\partial z^2} \right) * \left(\chi_1 B_z^{(0)}(\mathbf{r}) \right) \right) \quad (3.10) \end{aligned}$$

where $G(\mathbf{r}) = (4\pi||\mathbf{r}||)^{-1}$ and $\mathbf{B}^{(0)}(\mathbf{r}) = (B_x^{(0)}(\mathbf{r}), B_y^{(0)}(\mathbf{r}), B_z^{(0)}(\mathbf{r})) = (0, 0, B_0)$ is a homogenous magnetic induction field $\mathbf{B}_0$. More details on the derivation of these equations can be found in Appendix A and Jenkinson et.al. [37].

Out of $B_x^{(1)}(\mathbf{r})$, $B_y^{(1)}(\mathbf{r})$ and $B_z^{(1)}(\mathbf{r})$ only the projection onto the z-axis, $B_z^{(1)}(\mathbf{r})$, will have a detectable impact on the overall field (because it is parallel to the $\mathbf{B}_0$ field) and therefore $B_p(\mathbf{r}) = B_z^{(1)}(\mathbf{r})$. The magnetic field values $\mathbf{B}^{(1)}(\mathbf{r})$ are obtained at the centre of each object voxel. In order to obtain a representation of the perturbation field B_p throughout the whole object voxel trilinear interpolation can be applied. However the exact form obtained by using trilinear interpolation

involves cubic spatial dependency making later integration too complex and the computation therefore too slow. To overcome this the perturbed field is approximated by a linear spatial function:

$$B_p(\mathbf{r}) \simeq B_p(\mathbf{r}_0) + \mathbf{G}^p(\mathbf{r}_0) \cdot (\mathbf{r} - \mathbf{r}_0) \,, \tag{3.11}$$

where $\mathbf{r}_0$ is the centre of the object voxel and $B_p(\mathbf{r}_0) = B_z^{(1)}(\mathbf{r}_0)$. In the expansion (3.11), the vector $\mathbf{G}^p = (G_x^p, G_y^p, G_z^p)$ is fitted to the trilinearly interpolated field, $B_p(\mathbf{r})$, by minimising the squared difference of these fields giving the closest linear approximation. This method introduces small discontinuities at the object voxel boundaries, but these can be ignored in our model as we are mainly interested in the global fit of the perturbed field over the whole voxel. The resulting least squares solution gives $\mathbf{G}^p(\mathbf{r}_0)$ as a linear combination of the values of $B_z^{(1)}$ at adjacent grid points. The resulting expression for the perturbed magnetic field is then integrated into the phase of the magnetisation vector for each object voxel independently,

$$\phi(\mathbf{r}, t) = \gamma \int_{t_0}^{t} (\mathbf{G}(t) \cdot \mathbf{r} + B_p(\mathbf{r}_0) + \mathbf{G}^p(\mathbf{r}_0) \cdot (\mathbf{r} - \mathbf{r}_0)) \mathrm{d}t \,, \tag{3.12}$$

where $\mathbf{r}_0$ is the center of the object voxel.

3.4 Motion Model

Now consider the case where the head is moving in the scanner. A slightly different approach needs to be applied in order to solve Bloch and Maxwell's equations. The main idea is to have two coordinate systems: one that is fixed to the scanner and the other one that is fixed to the object.

In the *scanner* coordinate system the $\hat{\mathbf{z}}$ direction, by convention called the "longitudinal direction", is parallel to the static magnetic field of the scanner and

points in the positive direction of that field while $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ define the "transverse plane" which is orthogonal to the longitudinal direction. The object coordinate system is fixed to the input object, which is denoted by $\mathbf{r}_{ob} = x\hat{\mathbf{x}}_{ob} + y\hat{\mathbf{y}}_{ob} + z\hat{\mathbf{z}}_{ob}$. In case the object coordinate system is not specified, by default it will be assumed that its centre is the centre of the volume and its initial position, before any motion happens, is aligned with the *scanner* coordinate system.

The two coordinate systems are related in time with the expression $\mathbf{r} = \mathbf{R}(t)\mathbf{r}_{ob} + \mathbf{T}(t)$, where $\mathbf{T}(t)$ is a translation vector and $\mathbf{R}(t)$ is a rotation matrix. There are two ways to approach solving the Bloch equation. One is to stay in the *scanner* coordinate system in which case the tissue parameters change with the motion and we can not use the closed form solutions of the Bloch equation. The other way, which we choose, is to be in the coordinate system of the object in which our object appears static and the scanner moves relative to the object. We assume that the speed of the motion is very small relative to the precessional speed of spins, so that the axis about which the spins are processing is always the direction of the main static field. Therefore the solution during the relaxation and signal acquisition process takes the form:

$$\begin{aligned} M_z(\mathbf{r}_{ob}, t) &= e^{-(t-t_0)/T_1} M_z(\mathbf{r}_{ob}, t_0) + (1 - e^{-(t-t_0)/T_1}) M_0 \,, \\ M_{xy}(\mathbf{r}_{ob}, t) &= e^{-(t-t_0)/T_2^*} |M_{xy}(\mathbf{r}_{ob}, t_0)| e^{-i\phi(\mathbf{r}_{ob}, t)} \,, \end{aligned} \tag{3.13}$$

where M_z and M_{xy} are projections onto the z-axis and transversal plane of the *scanner* coordinate system respectively and the phase, $\phi(\mathbf{r}_{ob}, t)$, is

$$\phi(\mathbf{r}_{ob}, t) = \gamma \int_{t_0}^{t} (\mathbf{G}(t) \cdot (\mathbf{R}(t)\mathbf{r}_{ob} + \mathbf{T}(t)) + \widetilde{B}_p(\mathbf{r}_{ob}, t)) \mathrm{d}t \,, \tag{3.14}$$

where $\widetilde{B}_p(\mathbf{r}_{ob})$ is the susceptibility-induced B_0 inhomogeneity and is evaluated as described below.

Inhomogeneity in the magnetic field induced by the susceptibility differences between different tissues will also be influenced by the head motion. When the

object is moving, the coordinate systems of the object and the scanner are no longer aligned therefore, in the reference frame of the object, the orientation of the main magnetic field is changing and this in turn changes the B_p values of the perturbed field. This can be seen from equations 3.8, 3.9 and 3.10. As the orientation of the main magnetic field relative to the object changes (i.e. $\mathbf{B}^{(0)}(\mathbf{r})$ change) the perturbation field changes. As shown in Jenkinson et.al. [37], this new value of the perturbation field can be calculated using a set of pre-computed basis vector functions $\mathbf{B}_{p1}(\mathbf{r}), \mathbf{B}_{p2}(\mathbf{r}), \mathbf{B}_{p3}(\mathbf{r})$. These three vector functions were calculated using equations 3.8, 3.9 and 3.10 and for three different orientations of the main magnetic field relative to the object. Specifically, $\mathbf{B}_{p1}(\mathbf{r}) = \mathbf{B}^{(1)}(\mathbf{r})$ when $\mathbf{B}^{(0)}(\mathbf{r}) = (B_0, 0, 0)$; $\mathbf{B}_{p2}(\mathbf{r}) = \mathbf{B}^{(1)}(\mathbf{r})$ when $\mathbf{B}^{(0)}(\mathbf{r}) = (0, B_0, 0)$ and $\mathbf{B}_{p3}(\mathbf{r}) = \mathbf{B}^{(1)}(\mathbf{r})$ when $\mathbf{B}^{(0)}(\mathbf{r}) = (0, 0, B_0)$. This basis set is used to obtain any new value of the perturbation field:

$$\widetilde{B}_p(\mathbf{r}, t) = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \mathbf{R}(t) \begin{bmatrix} \mathbf{B}_{p1}(\mathbf{r}) & \mathbf{B}_{p2}(\mathbf{r}) & \mathbf{B}_{p3}(\mathbf{r}) \end{bmatrix} R^{-1}(t) \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \quad (3.15)$$

Note that translation does not impact the perturbation field. This is because the orientation of the magnetic field relative to the object does not change when translation is applied and therefore the perturbation field does not change either.

The perturbed field is calculated at the centre of each object voxel according to Eq. [3.15] and can be written as $\widetilde{B}_p(\mathbf{r}_{ob}, t) = \sum_{j=1}^{3} \mathbf{c}_j(t)\mathbf{B}_{pj}(\mathbf{r}_{ob})$ where $\mathbf{c}_j$ is a row vector given by $\mathbf{c}_j(t) = R_{3,j}(t) \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \mathbf{R}(t)$ and $R_{3,j}(t)$ is the $(3, j)$th element from the matrix $\mathbf{R}(t)$. $\widetilde{\mathbf{G}}^p(\mathbf{r}_{ob}, t)$ is calculated from $\widetilde{B}_p(\mathbf{r}_{ob}, t)$, and implemented into the simulation equations in a similar fashion as described before.

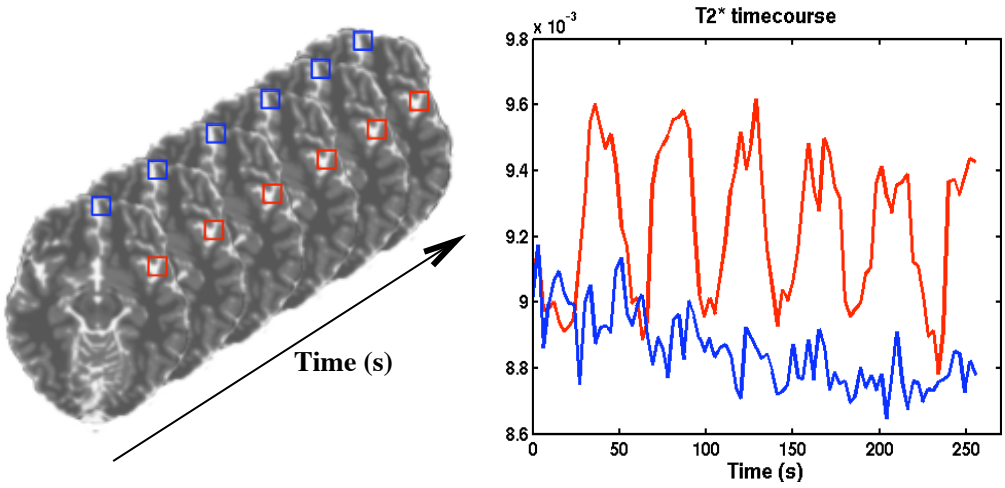


Figure 3.3: T_2^* variation is defined for every object voxel through a user specified file. This figure shows examples of T_2^* timecourses where the red and the blue lines correspond to two randomly chosen object voxels in the marked red and blue selected areas of the brain respectively.

3.5 BOLD model

In the simulator, T_2^* models the effects of blood oxygenation changes within the object. Specifically, T_2^* variation in time is used to model the effects of neuronal activation on the MR signal, via the BOLD effect. This model can also include, as additional T_2^* variation, unwanted physiological changes (e.g. low frequency networks - so called "resting states" or respiratory-induced fluctuations). The T_2^* variation is user-defined for every object voxel separately in the form shown in Figure 3.3. The Bloch equations for the transverse components, when the parameter T_2^* varies in time, are

$$\begin{aligned} \frac{dM_x(t)}{dt} &= \gamma M_y(t) B_z(t) - \frac{M_x(t)}{T_2^*(t)}\,, \\ \frac{dM_y(t)}{dt} &= -\gamma M_x(t) B_z(t) - \frac{M_y(t)}{T_2^*(t)}\,, \end{aligned} \tag{3.16}$$

where $B_z(t)$ is the magnetic field in the z-direction of the scanner. We introduce the change of variables $M_x(t) = m_x(t)\exp(D(t))$, $M_y(t) = m_y(t)\exp(D(t))$, and

in order to eliminate the relaxation terms we choose $D(t) = -\int_{t_0}^{t}[T_2^*(t)]^{-1}dt$. The resulting differential equations for m_x and m_y are

$$\begin{aligned} \frac{dm_x(t)}{dt} &= \gamma m_y(t)B_z(t)\,, \\ \frac{dm_y(t)}{dt} &= -\gamma m_x(t)B_z(t)\,, \end{aligned} \tag{3.17}$$

which are the equations representing the processional motion about the z-axis. Substituting the solution of Eq. [3.17] back into the expressions for M_x and M_y and using the complex form we get the solution of Eq. [3.16] as

$$M_{xy}(t) = |M_{xy}(t_0)| \exp\left(-\int_{t_0}^{t}\frac{1}{T_2^*(t)}dt\right)\exp\left(-i\gamma\int_{t_0}^{t}B_z(t)dt\right). \tag{3.18}$$

3.6 Signal evaluation

The MR signal of the object, received by the scanner, is given by the signal equation

$$S(t) = \iiint_{\text{object}} M_{xy}(\mathbf{r},t)d\mathbf{r}\,, \tag{3.19}$$

where $\mathbf{r}$ is in the object coordinate system and unless otherwise stated we will assume from now on that all the behavior of the magnetisation is described in the object coordinate system. As there is no analytical solution for this integral we write

$$S(t) = \sum_{j\in\Lambda}\sum_{\mathbf{r}_0\in\Omega} s_j(\mathbf{r}_0,t)\,, \tag{3.20}$$

where Ω is the collection of all of the object voxels, Λ is the collection of all the tissue types in the object and $s_j(\mathbf{r}_0,t)$ is the signal coming from the jth tissue component of the object voxel centered at $\mathbf{r}_0$. As these tissues are uniformly distributed over the object voxel the signal is calculated using Eq. [3.19] as shown

in the following equation:

$$\begin{aligned} s_j(\mathbf{r}_0,t) &= p(\mathbf{r}_0,j)\iiint_{V_{\mathbf{r}_0}} |M_{xy}(\mathbf{r}_0,t_0)| \exp\left(-\int_{t_0}^{t}\frac{1}{T_2^*(\mathbf{r}_0,t)}dt\right) \\ &\quad \exp\left(-i\gamma\int_{t_0}^{t}\mathbf{G}(t)\cdot(\mathbf{R}(t)\mathbf{r}+\mathbf{T}(t))dt\right) \\ &\quad \exp\left(-i\gamma\int_{t_0}^{t}(\widetilde{B}_p(\mathbf{r}_0,t)+\widetilde{\mathbf{G}}^p(\mathbf{r}_0,t)\cdot(\mathbf{r}-\mathbf{r}_0))dt\right)d\mathbf{r}\,, \quad (3.21) \end{aligned}$$

where j represents the tissue type (e.g. gray, white, CSF etc.), $V_{\mathbf{r}_0}$ is the object voxel with the centre at $\mathbf{r}_0$ and with dimensions (L_x, L_y, L_z) where L_x is its length in the x direction (L_y in y and L_z in z), $p(\mathbf{r}_0, j)$ represents the proportion of the jth tissue type present in the object voxel $V_{\mathbf{r}_0}$, t_0 is the time of the last RF pulse, $M_{xy}(\mathbf{r}_0, t_0)$ is the transverse magnetisation following the last RF excitation, $\mathbf{G}(t)$ is the applied gradient vector at time t, and $\widetilde{B}_p(\mathbf{r}_0,t)+\widetilde{\mathbf{G}}^p(\mathbf{r}_0,t)\cdot(\mathbf{r}-\mathbf{r}_0)$ is the perturbed B_0 field. For each tissue type T_1, T_2^*, ρ and χ are spatially constant throughout an object voxel, and so the integral can be evaluated analytically over each object voxel separately giving

$$\begin{aligned} s_j(\mathbf{r}_0,t) &= p(\mathbf{r}_0,j)L_xL_yL_z|M_{xy}(\mathbf{r}_0,t_0)| \exp\left(-\int_{t_0}^{t}\frac{1}{T_2^*(\mathbf{r}_0,t)}dt\right) \\ &\quad \operatorname{sinc}(F_x(t))\operatorname{sinc}(F_y(t))\operatorname{sinc}(F_z(t)) \\ &\quad \exp\left(-i\gamma\int_{t_0}^{t}(\mathbf{G}(t)\cdot(\mathbf{R}(t)\mathbf{r}_0+\mathbf{T}(t))+\widetilde{B}_p(\mathbf{r}_0,t))dt\right)\,, \quad (3.22) \end{aligned}$$

where $F_x(t) = \frac{1}{2}\gamma L_x\int_{t_0}^{t}(\mathbf{G}(t)R_x(t)+\widetilde{G}_x^p(\mathbf{r}_0,t))dt$, and R_x denotes the first column of the rotation matrix (similarly for the y and z directions).

3.7 Additions

3.7.1 Chemical Shift

In the case of a chemical shift, an additional B_0 inhomogeneity is introduced that is not dependent on the actual position of the object voxel but on the tissue type

(as it depends on the shielding constant δ_{cs}), so there is an additional constant $B_{cs} = \delta_{cs} B_0$ in Eq. [3.23] which does not vary when the brain is moving. That is,

$$B_p(\mathbf{r}, j) = B_p(\mathbf{r}_0) + B_{cs}(j) + \mathbf{G}^p(\mathbf{r}_0) \cdot (\mathbf{r} - \mathbf{r}_0) \,, \tag{3.23}$$

where j indicates the tissue type (white, gray, fat etc.).

3.7.2 Eddy Currents

Eddy currents are unwanted induced currents which are produced in the conducting elements of an MR scanner. They arise due to the fast changing fields that occur within the scanner, normally due to rapid switching of applied gradient fields. The effect of the eddy currents is to create additional, unwanted gradient fields which are simulated by superimposing a sum of exponentially decaying terms on to the gradient waveform.

3.7.3 Noise

MR scanners, like all measurement devices, are influenced by noise. For instance, thermal noise is present in the object and receiver electronics which results in fluctuations in the received signal that cover a wide frequency range. This is modeled as additive, independent, white Gaussian noise in both receiver channels within the simulator.

The simulator can also include physiological noise, specifically respiratory and cardiovascular by modifying the T_2^* and B_0 variations in time. This is planned as a part of the future work. Modelling and data for the physiological noise will be provided by the Montreal Neurological Institute, McGill University with whom, together with a group at the University of Hawaii (formerly at the University of Pittsburgh), we have formed a collaborative consortium named MIDAS [54].

3.8 Summary

Developing an FMRI Simulator is a very complex task because of the number of different factors that can have an impact on the FMRI data. In order for the simulations to be realistic it is crucial to have as many of these factors included in the simulations as possible. Due to the computational intensity of the task, most of the simulators that were developed in the past could not produce realistic images. Some simulators were developed in such a way as to avoid the computational difficulties but were unable to incorporate some of the very common but more complex artefacts. This section presented the main theoretical model behind the FMRI simulator developed in this thesis, together with its main features: magnetic inhomogeneities, rigid-body motion of the head, BOLD signal, eddy currents and noise. The following chapter describes the implementation of this model into a simulator environment and discusses the way the computational issues were dealt with in order to make the images realistic and simulation processes both memory- and time-efficient.

Chapter 4

Simulator implementation

Following on from the previous chapter, which considered the model of the simulation process, this chapter is concerned with how simulations are implemented in software. As mentioned in the previous chapter, the computational demands of these types of simulations are considerable. This chapter also sets out the way various techniques, including parallelisation, were used to lessen the demands on processor time and on computer memory.

4.1 Introduction

The FMRI simulator software is called Physics Oriented Simulated Scanner Utility for MRI (POSSUM) and is written in C++ for a variety of Unix-based platforms (e.g. MacOSX, Linux, Cygwin under Windows, etc.). It consists of several different modules that work together in order for a simulation to be performed. Figure 4.1 depicts the process of the simulation from inputs into the Bloch equation at the top to the outputs and subsequent modifications of the outputs at the bottom. Each of the stages of the simulation process is carried out by one or more software modules.

The object is a collection of object voxels each of which is a collection of

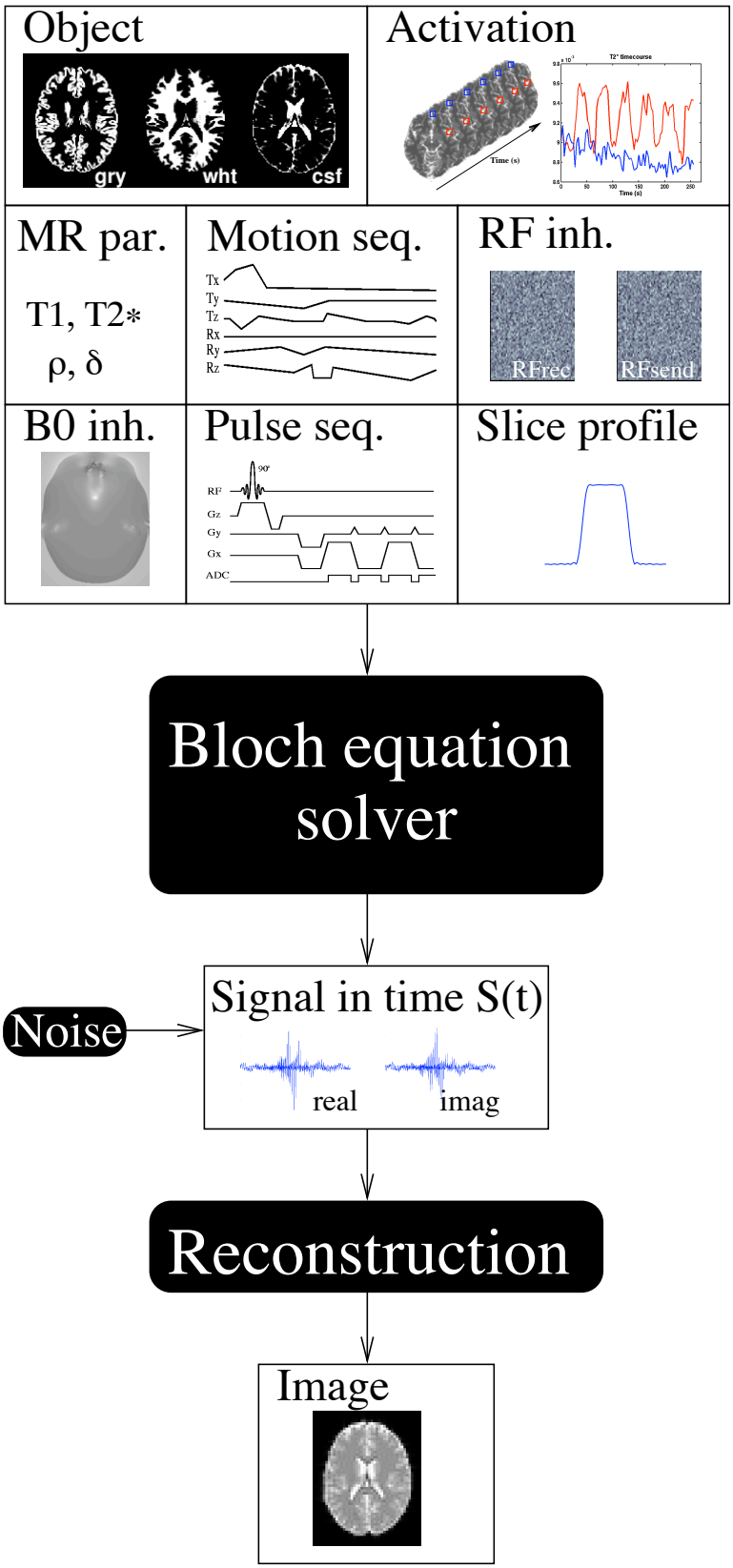


Figure 4.1: Descriptive representation of the simulation process. The black box in the middle of the image represents the 'brain' of the simulator - the Bloch equation solver. Images represented on the top (object, pulse sequence, motion sequence etc.) are inputs to the solver. The output of the solver is the signal $S(t)$ to which the thermal noise is added. The final stage is the reconstruction of the signal into the image. More details are described in the main text.

tissues with specified relaxation times T_1, T_2^*, spin density ρ, and chemical shift value δ_{cs}. The pulse sequence is defined by the values of the gradient field, RF pulse parameters (including the desired flip angle) and read-out timings. The motion sequence is defined by the values of the motion parameters (translations T_x, T_y, T_z, and rotation angles R_x, R_y, R_z characterising rotations about the center of the volume and about x-, y- and z-axis respectively). The activation is defined by variations in T_2^*. The input activation file is four-dimensional, and contains a time-series of T_2^* changes for each object voxel. The B_0 inhomogeneities are calculated at the centre of each of the object voxels as described in the previous chapter. These values are then read into the solver. The slice profile is defined through an input matrix file specifying the frequency profile of the RF pulse. RF inhomogeneities are specified for each of the voxels, one 3D image for the receive coil and one for the transmit coil.

Signal values (both real and imaginary) are calculated at the read-out points and then, after the thermal noise is added to them, passed into the image reconstruction algorithm. Here they are arranged into a k-space at sampling positions that are calculated from the pulse sequence gradients, and by means of some image reconstruction algorithm (e.g. Fast Fourier Transform (FFT)), are converted into the image. Note that the complex noise-free, non reconstructed signal $S(t)$ is also available as a direct output and can be passed into any custom reconstruction software. In the following sections each of the elements of this diagram is described in greater detail.

4.2 Inputs to the simulator

As mentioned in the previous chapter, the main goal of the FMRI simulator is to make simulations as realistic as possible. For that reason it is important to provide

as many different object- and scanner- related properties. These properties are provided to the simulator in the form of either image or matrix data. All of the properties that the simulator can take into account are described in this section.

4.2.1 Object

The object can be a segmented anatomical voxel model from any source. It should consists of three-dimensional anatomical images of distinct tissue types with specified MR parameters (relaxation times T_1, T_2^*, spin density ρ, susceptibility value χ and chemical shift value δ_{cs}) for each of the tissue types. For example, if the object is a brain, then a segmented anatomical voxel model would be a collection of tissue templates: three-dimensional anatomical images of distinct tissue types (e.g. one template of grey matter, another of white matter and a third of cerebro-spinal fluid (CSF) as shown in Figure 3.1). If the object is a breast, then a collection of these tissue templates might consists of three-dimensional anatomical images of fat tissue, skin, etc.

Most of the examples in this thesis are generated using the digital brain phantom introduced by the McConnell Brain Imaging Centre, Montreal Neurological Institute, McGill University [17, 19, 46]. The digital brain phantom consists of a set of three-dimensional tissue membership volumes, with nine tissue classes (white matter, gray matter, cerebrospinal fluid (CSF), fat, muscle, glial tissue, bone, skin and connective tissue). The voxel values in these volumes reflect the proportion of tissue present in that voxel, in the range [0,1]. The volumes are defined on a 1mm voxel grid, with dimensions $181 \times 217 \times 181$ and an origin of (90, 126, 72). Examples in this thesis are generated using three main tissue classes: white matter (WM), gray matter (GM) and cerebrospinal fluid (CSF) (except in the case of the chemical shift example where fatty tissue was also used).

The McGill phantom was created from an enhanced MRI volume, obtained from 27 T_1 weighted scans of a single subject, that were automatically co-registered, normalised and averaged. After all these steps have been taken, the resulting volume was segmented into distinct tissue templates [19]. Other input objects can be made in a similar way to this phantom and can then be used for simulations.

4.2.2 The MR parameters

The MR parameters (relaxation times T_1, T_2^*, spin density ρ, and chemical shift value δ_{cs}) are specified using a matrix. The parameter values for each of the tissue types in the brain phantom that were used in the thesis are set out in Table 4.1. These MR parameters were derived by the group at the McConnell Brain Imaging Centre which used a simulator developed by Kwan et.al. [46]. They used their tissue template model with specified MR parameters to simulate images which were then compared with the real scan from which the templates were originally derived. They minimised the difference between the real and simulated images with respect to the MR parameters. For the initial value of the MR parameters they took well known literature-based values.

Tissue type	T_1 (ms)	T_2^* (ms)	ρ	δ_{cs} (ppm)
Gray	833	83	0.86	0
White	500	70	0.77	0
CSF	2569	329	1	0
Fat	350	70	1	-3.5
Air	-	-	0	-

Table 4.1: Relaxation times, spin density, and chemical shift values (at 1.5T) used in the simulations.

4.2.3 Pulse sequence

During the real image acquisition process, each of the pieces of equipment in the scanner receives a set of commands. These commands control the behaviour of the gradient fields, RF fields and receiver electronics by specifying the required currents and voltages to the coils and circuits. The set of commands is called the pulse sequence. In the simulator the pulse sequence also controls all of the gradient fields, RF fields and receiver electronics during the scanning but instead of currents and voltages it specifies the requested fields directly (or in the case of receiver electronics it specifies 1 for the action "read the signal" and 0 otherwise).

The pulse sequence in the simulator is defined by the desired values of the gradient field, RF pulse parameters and readout timings at discrete, user-specified time-points. It is in matrix form and an example portion can be seen in Figure 4.2 (note: values in the matrix do not correspond to real values, they are just representative). The top four rows represent slice selection, the next three show signal read-out for one line of k-space, and the last three show how to go from one line of k-space to the next. Time is in seconds, and for each moment in time, a value is specified for all the parameters.

RF pulses in the simulator are defined by the start time, desired flip angle, frequency width, and the centre frequency. RF frequency bandwidth (df) defines the range of frequencies around the RF centre (carrier) frequency offset (fc): $(fc - \frac{df}{2}, fc + \frac{df}{2})$. The read-out is instantaneous and the read-out points are defined in a discrete fashion using on/off (1/0) values. The times for the read-out points are unrestricted so that either linear or nonlinear temporal sampling schemes can be specified. G_x, G_y and G_z are the desired applied gradient fields. They are specified with given values at known time-points. The values of gradients between the specified time-points are obtained by linear interpolation.

time (s)	RF angle (rad)	RF frequency bandwidth (Hz)	RF center frequency (Hz)	signal read–out 0/1	gradients (T/m)		
					G_x	G_y	G_z
0	0	0	0	0	0	0	0
0.00003	0	0	0	0	0	0	0.0072
0.00203	1.5708	303.512	−150.5	0	0	0	0.0072
0.00403	0	0	0	0	0	0	0.0072
...	...	...	...	...	...	...	...
0.02919	0	0	0	1	0.00917	0	0
0.02920	0	0	0	1	0.00917	0	0
0.02921	0	0	0	1	0.00917	0	0
...	...	...	...	...	...	...	...
0.02981	0	0	0	1	0.00917	0	0
0.02982	0	0	0	1	0.00917	0	0
0.02986	0	0	0	0	0	0.00478	0
0.02990	0	0	0	1	–0.00917	0	0
0.02991	0	0	0	1	–0.00917	0	0
0.02992	0	0	0	1	–0.00917	0	0
...	...	...	...	...	...	...	...

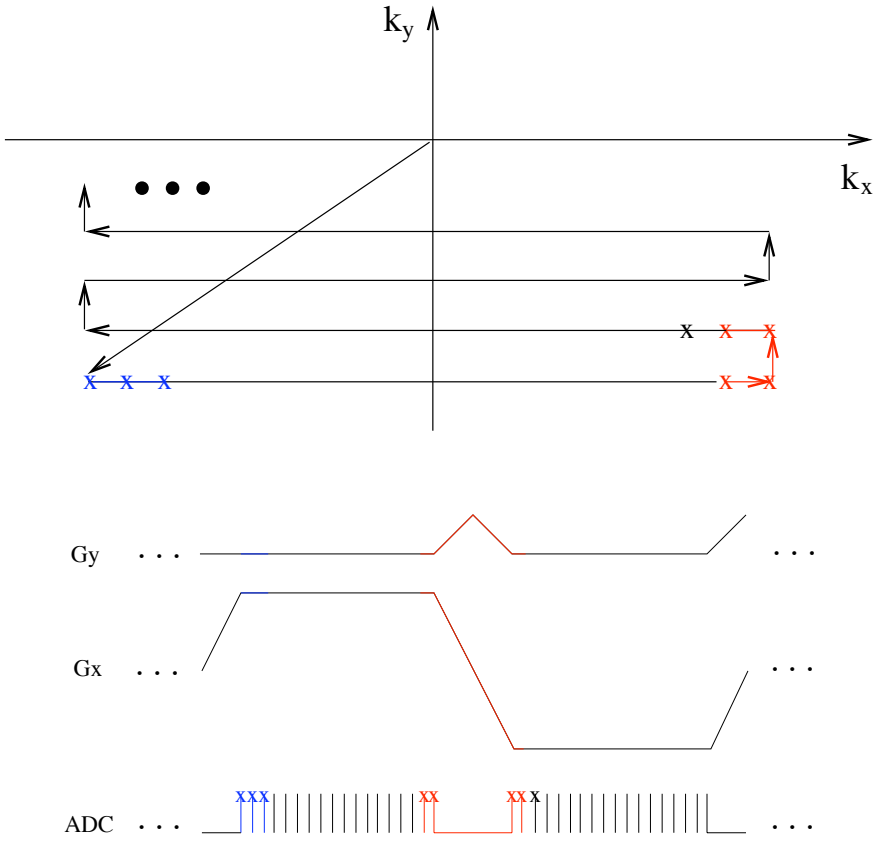


Figure 4.2: The figure shows an example portion from a user-specified pulse sequence matrix which is one of the inputs into the simulator. Below this are shown the corresponding k-space trajectory and gradient wave forms.

The matrix form presented in Figure 4.2 here is in its raw format which is used directly in the main engine of the simulator. Its size depends on the size and a resolution of an output image. In a typical FMRI pulse sequence this matrix can have approximately 20 million lines! This is one of the main reasons for the massive computational requirements of the simulator: the Bloch equation is solved and magnetisation vectors updated for all of the object voxels at each of these lines. A special module of the simulator is responsible for building the matrix. Named "the pulse sequence generator" this module has a user-friendly setup which functions in exactly the same way as a normal scanner console. This module allows more familiar quantities, such as the matrix size and field of view, as well as general sequence type (Echo Planar Imaging (EPI), standard Gradient Echo (GE), etc.) to be chosen by the user. It also allows more hardware specific characteristics of the scanner to be specified such as: the speed of the gradient coils, the maximum gradient field strength etc. From these values a raw pulse sequence matrix is then calculated and used as an input to the Bloch equation engine.

It is also possible to import the matrix directly from any other pulse sequence generators. These are often part of the scanner software system. This allows the experimenter to use exactly the same pulse sequence on the simulator as the one that was used in real experiments.

Eddy currents arise as a consequence of the rapid and powerful pulsing of the gradient coils. This is described in Chapter 1. The effect of the eddy currents is simulated by superimposing a sum of exponentially decaying terms on to the gradient waveforms defined with the pulse sequence just described.

4.2.4 Slice profile

A further property of RF pulses, in addition to those mentioned when RF pulses were defined above, is the slice profile, which is used as an input to the simulator. The slice profile describes the shape of the RF pulse in the frequency space. The shape is important as it defines the "amount" of excitation for the spins precessing at different frequencies (in Figure 2.6 the relationship between these shapes and the actual slice selection is shown).

Figure 4.3 shows examples of two different slice profiles: square slice profile which is an ideal, and not realistically achieved profile, as well as windowed-sinc profile which is more realistic and commonly used in scanning. The x-axis in the

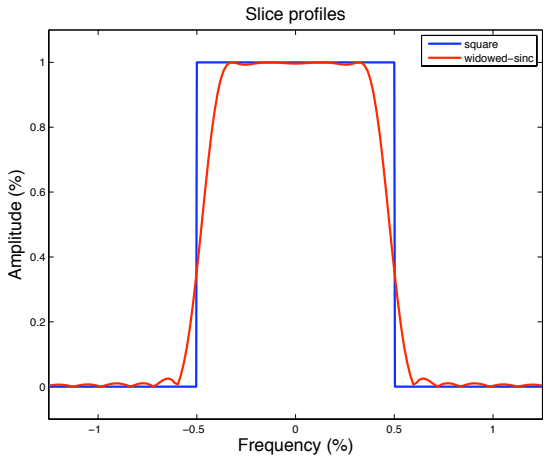


Figure 4.3: Examples of different slice profiles. These are Fourier transforms of the original RF pulses. In the case of windowed-sinc, the name refers to the shape of the RF pulse envelope and not its shape in the frequency space.

figure is given in percentages of the RF frequency bandwidth. The distance from -0.5 to 0.5 represents one whole bandwidth or equivalently the range of frequencies required for the excitation of one slice. The center of the slice profile (value 0) corresponds to the RF center frequency and this value changes from one slice to the next during the simulation. Values that are higher than 0.5 (or lower than

-0.5) represent frequencies that are outside of the ideal range of frequencies for the slice. The y-axis is given in percentages of the desired angle. The frequencies for which amplitude is 1 will get excited with a flip angle equal to the desired, user-specified, flip angle.

The square slice profile is an ideal slice profile which occurs when the RF pulse is an infinite sinc function. It is ideal because all of the targeted spins will get excited equally. However this is never the case in practice because all RF pulses are finite and imperfect, therefore ripples can occur at the edges of the perfect square meaning not just that non-desired spins will be excited but also that different spins will get different "amounts" of excitation creating a non-uniformly excited slice. The other example, windowed-sinc, is a slice profile of a windowed-sinc RF pulse (note that the windowed-sinc slice profile is named after the shape of its RF pulse and not because of its shape in the frequency space). It was calculated using real scanner system software, and is a realistic example of a slice profile. In this thesis, simulations were done for both the ideal square and the windowed-sinc slice profiles and the different impacts in the images are recorded, as shown in Figure 5.3.

4.2.5 Motion sequence

The input motion sequence is defined in the simulator by an input file which specifies the motion parameters: T_x, T_y, T_z characterising translations, and rotations R_x, R_y, R_z characterising rotations about the center of the volume. These rotations are specified using rotation angles often called pitch (rotation about the x-axis), roll (rotation about the y-axis) and yaw (rotation about the z-axis) and

can be carried out by the rotation matrices:

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{bmatrix}, R_y(\beta) = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix}, \tag{4.1}$$

$$R_z(\gamma) = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4.2}$$

The total rotation R of the object at a specific time point is then obtained by multiplying these rotation matrices using the expresion $R = R_z(\gamma) * R_y(\beta) * R_x(\alpha)$ (chosen by convention). However, working with rotation matrices in the simulator was not the most efficient and most accurate approach. One of the main reasons for this is that after each matrix multiplication, each of the columns has to be normalised in order to be a unit vector (as this is the fundamental property of all rotation matrices). This process induces an error which accumulates with the increassing number of multiplications. As there could be millions of consecutive multiplications within the simulator, this process soon becomes very numerically unstable. In order to avoid this, rotation matrices are not used in the simulator. For the representation of the rotations, quaternions are used instead.

A unit-length quaternion is an algebraic structure which can be written in the form

$$q = \cos(\frac{\theta}{2}) + \mathbf{u}\sin(\frac{\theta}{2}) \tag{4.3}$$

which conveniently represents a rotation by an angle θ and about an axis $\mathbf{u} = (u_x, u_y, u_z)$. This property of using only four coordinates to represent a rotation and also the fact that the number of constraints on these coordinates is small (the only constraint on a quaternion representing rotation is that it have a unit

length[44]) means that the unit quaternions are a very efficient representation of rotations. In addition to this they are also numerically better conditioned than rotation matrices as in order to preserve the unit-length there is only one correction that needs to be done (instead of three as is the case for the rotation matrices). More information on quaternions and their properties can be found in [44, 15, 4].

Assuming that not all users are familiar with quaternions, and in order to make things easier for them, the rotations in the input motion file are specified with yaw, pitch and roll angles. Once the simulator starts running, one of its first steps is to convert these into quaternions which from then on are solely used. In addition, the motion parameters are specified at user-specified discrete time-points (e.g. only every 3s), which can be totally different to the pulse sequence time (e.g. every 0.000001s). In order to have a representation of the object's position and orientation at every point of the pulse sequence the interpolation between the discrete time-points is done. The interpolation assumes a constant linear and angular velocity between each pair of discrete time-points and is described in more detail in the Bloch equation solver section.

4.2.6 B_0 inhomogeneities

In the previous chapter a method for calculating B_0 inhomogeneities induced by the differences in susceptibility values was presented. B_0 inhomogeneities represent difference values between the scanner-defined uniform B_0 field and the actual field B calculated by solving Maxwell's equations. B_0 inhomogeneity values are influenced by head motion, as described in the previous chapter. In order to calculate the inhomogeneities, a set of precomputed basis vector functions [37] is needed. The basis set is calculated and as a set of volumes (3D images) passed

into the Bloch equation solver.

4.2.7 Activation

Activation (BOLD signal) is defined in the simulator through the changes in the T_2^*. Each object voxel has its own time-series of T_2^* changes. These are user-defined and packed into massive four-dimensional volumes often called T_2^* maps. In order to obtain realistic T_2^* maps, T_2^* is often calculated from real artefact-corrected data sets.

A common way to calculate T_2^* values is to derive them by fitting the equation $S = S_0 \exp(-TE/T_2^*)$ to an experimentally acquired FMRI data set, where S is the measured signal intensity, TE is the echo time and S_0 is the baseline intensity (determined from the average value of S under baseline conditions). Temporal sampling of T_2^* is coarse as one T_2^* value is obtained for every one TR (which is 3s in a typical FMRI experiment). This is however acceptable for the BOLD simulations as neuronally-induced BOLD signals change slowly (approximately 6s takes from the onset of the stimulus till BOLD signal reaches its maximum value[40]).

4.3 Bloch equation solver and the signal

The Bloch equation solver is the main engine of the simulator. All of the inputs described above are processed by the solver in order to create one output: the signal. During the process of scanning, magnetisation is detected i.e. converted into an electrical signal by the RF receiver coil. In Chapter 3 the main equations for calculating this signal were given: Eq. 3.20 for calculating the total signal and Eq. 3.22 for calculating the signal coming from each individual voxel. Implemented in a straightforward way these equations create a huge demand on

computer processing time and memory. Simulations of realistic FMRI images would require months or years to be carried out. This section describes the steps taken to implement these equations and to do so in such a way that simulations can be performed within hours or days.

An outline of the Bloch equation solver is given as algorithm 1. A careful description of each of the lines of the algorithm is set out in this section. All of the lines of the algorithm that start with "Speedup" describe procedures carried out in order to speed up the simulation process.

The algorithm begins by initially setting up a coordinate system-based on the dimensions of the input object. The centre of the coordinate system is in the centre of the object volume. In addition, all of the input files, except the magnetic field perturbation volume B_p, are read into the solver. B_p volume is read into the solver at a later stage. Depending on the amount of motion and whether B_p varies in time or not the simulator will run one of the four levels Level1, Level2, Level3 or Level4. All of the levels are similar in structure and have fundamentally the same methods, however they run different functions *voxel1*, *voxel2*, *voxel3* and *voxel4*. These functions are solving the Bloch equations for a single voxel, but each of them solves them for different levels of complexity of the model (e.g. *voxel1* solves the simplest equations as they do not contain any motion terms while *voxel3* solves the most complex case when there is out of plane rotation happening which involves complex motion and B_0 interactions). The use of this kind of division into levels is purely for increasing the computational speed of the algorithm. Later in the section a more detailed description of the *voxel1-4* functions is given.

Following the level selection, a pulse sequence and a motion sequence are combined into one sequence, referred to in the rest of the text as the events sequence. Pulse sequence parameters are given at time-points which are very

Algorithm 1 POSSUM: calculates total signal S(t)

Set up a coordinate system
Read in all input files except B_p
events sequence=*sort*(pulse sequence, motion sequence)
Scale slice profile to the given slice thickness
Speedup1: Extract the level of motion and the type of B_p changes
Speedup2: Extract the parameters for the ROI in the object
Speedup3: Extract the parameters for the ROI in the event sequence
Calculate Nreadp: the number of the read-out points in the event sequence
Initialise the total signal: $S(t) = zeros(2, Nreadp)$
if Level1: no-motion AND B_p not changing in time **then**
  Read in 3D B_p volume
  Apply function *calc-grad*: Calculate 3D gradient volumes G_x^p, G_y^p, G_z^p
  Speedup4: Exclude all zero spin density voxels from the ROI
  for $v \in$ non-zero spin density voxels of ROI **do**
    Calculate $s_j(v,t)$ and add to the total signal using function *voxel1*
  end for
  Output the total signal S(t)
end if
if Level2: motion parallel to the B_0 field AND B_p not changing in time **then**
  Read in 3D B_p volume
  Apply function *calc-grad*: Calculate 3D gradient volumes G_x^p, G_y^p, G_z^p
  Speedup4: Exclude all zero spin density voxels from the ROI
  for $v \in$ non-zero spin density values of ROI **do**
    Calculate $s_j(v,t)$ and add to the total signal using function *voxel2*
  end for
  Output the total signal S(t)
end if
if Level3: motion non-parallel to the B_0 field AND B_p not changing in time **then**
  Read in basis vectors volumes: B_{p1}, B_{p2}, B_{p3}
  Apply function *calc-grad*: Calculate gradient volumes for each of the B_p's.
  Speedup4: Exclude all zero spin density voxels from the ROI
  for $v \in$ non-zero spin density values of ROI **do**
    Calculate $s_j(v,t)$ and add to the total signal using function *voxel3*
  end for
  Output the total signal S(t)
end if
if Level4: no-motion AND B_p changing in time **then**
  Read in 4D B_p volume
  Apply function *calc-grad*: Calculate 4D gradient volumes G_x^p, G_y^p, G_z^p
  Speedup4: Exclude all zero spin density voxels from the ROI
  for $v \in$ non-zero spin density values of ROI **do**
    Calculate $s_j(\mathrm{v},t)$ and add to the total signal using function *voxel4*
  end for
  Output the total signal S(t)
end if

close to one another (micro seconds) while motion parameters are usually given at much coarser intervals (seconds). In order to make the events sequence, each time point requires all of the pulse sequence and motion sequence information. For the pulse sequence parameters direct application of linear interpolation is used for estimating the missing points.

Each of the time-points where motion parameters (T_x,T_y,T_z,R_x,R_y,R_z) are specified by an user we refer to as control points. In order to interpolate between these points we need to find a curve that approximates the motion that happened in-between the control points. The interpolation assumes a constant linear and angular velocity between each pair of discrete time-points. Because of this assumption, linear interpolation is used to estimate the position (translation parameters).

In addition to this, a rotation that is required for the object to move from the orientation at one control point to the next is estimated. This is done by firstly converting pitch,roll and yaw angles (as defined in Section 4.2.5) into quaternions at each of the control points and then calculating the axis $\mathbf{u}$ and the angle θ of the difference quaternion. Interpolation is applied on the obtained angle θ. This process is represented as Algorithm 2. Once the interpolation is done, both pulse sequence parameters and the motion sequence are stored in the events sequence which is from then on used instead of the two separate input sequences.

Algorithm 2 interpolates rotation between two control points

1: Read pitch, roll and yaw angles at control point i: $\alpha_1, \beta_1, \gamma_1$
2: Read pitch, roll and yaw angles at control point $i+1$: $\alpha_2, \beta_2, \gamma_2$
3: Convert pitch, roll and yaw angles at i to a unit quaternion: q_i
4: Convert pitch, roll and yaw angles at $i+1$ to a unit quaternion: q_{i+1}
5: Calculate the difference quaternion: $q = q_{i+1} * q_i^{-1}$
6: Convert q into the axis $\mathbf{u}$ and the angle θ of the rotation using Eq. 4.3
7: Interpolate linearly the angle θ for all the points in-between the control points i and $i+1$: θ_1, θ_2,... θ_n

The axis and angle of rotation are used to derive the rotation matrix $\mathbf{R}(t)$ [44] which for implementational efficiency we write in the form

$$\mathbf{R}(t) = I + \sin(\theta(t))\mathbf{A}(t) + (1 - \cos(\theta(t)))\mathbf{A}(t)^2 \,, \tag{4.4}$$

where

$$\mathbf{A}(t) = \begin{bmatrix} 0 & -u_z(t) & u_y(t) \\ u_z(t) & 0 & -u_x(t) \\ -u_y(t) & u_x(t) & 0 \end{bmatrix} \,,$$

$\mathbf{I}$ is the identity matrix, θ is the rotation angle and $\mathbf{u} = (u_x, u_y, u_z)$ is a unit vector representing the axis of the rotation from the initial orientation to the orientation at time t.

After the sorter is applied (line 3 of the Algorithm 1), a slice profile is scaled to the required slice parameters as described in the previous section. The following three lines (line 5, 6 and 7) represent three different speed-up mechanisms that are applied. The first, Speedup1 uses the information about the motion and the B_0 inhomogeneity to determine which of the four levels are to be used in the simulation. Depending on the amount of motion and whether B_p varies in time or not the simulator will run one of the four. All of the levels are similar in structure and have fundamentally the same methods. In the case that the object is not moving and the perturbation field B_p is not changing in time, Level1 is used. From Eq.3.15 can be seen that in this case only one basis vector is needed to model the perturbations.

Similar situation happens in Level2 when the object is moving parallel to the main magnetic field. The difference between Level1 and Level2 is in the equations that are calculating the signal. The ones in Level1 do not involve motion and are therefore much simpler and faster to calculate.

Level3 is used when an object rotates about axis that is not parallel to the

field. In this case all of the basis vectors are needed which requires significantly more calculations to be performed and additional use of computer memory. In terms of the computational speed, Level3 is 50% longer in duration than Level2.

Level4 is applied when the perturbed field is changing in time. It is read into the solver as a 4D volume where the fourth dimension represents time. This feature is only done for the case when there is no motion and it is in the plans for the future work for this feature to be extended for the case when the object is moving. The main difference between the different levels is in their speed as the ones that simulate more advanced effects require more calculations and are slower. The level division is useful because applications which do not involve motion or complex B_0 inhomogeneity effects can then be performed quickly. Each of the different level structures can be seen in Algorithm 1.

Speedup2 is a process which extracts the region of interest from the object (ROI). Calculations are consequently performed only for voxels belonging to the ROI. Speedup3 is a process which extracts a region of interest from the events sequence. Detailed description of these two speed-ups is in the section Computational time. As described in the section about the pulse sequence, read-out points are time-points at which the receiver coil collects the signal. The number of read-out points in the events sequence is proportional to the number of the k-space points, the number of slices in the volume and to the number of volumes (line 8). It is used to initialise the signal. The signal is in the complex form and the real and the imaginary part of the signal are calculated separately using equations Eq.4.5 and 4.6.

Depending on the amount of motion and whether B_p varies in time or not the simulator will run one of the four levels. All of the levels are similar in structure and have fundamentally the same methods. Therefore, in order to avoid repetition we will describe Level3 as that is the one the contains the most

complex calculations.

As shown in the previous chapter, $\widetilde{B}_p(\mathbf{r},t)$ is calculated using a set of pre-computed basis vector functions $\mathbf{B}_{p1}(\mathbf{r}), \mathbf{B}_{p2}(\mathbf{r})$, and $\mathbf{B}_{p3}(\mathbf{r})$ [37]. Each of the vector basis functions is evaluated at the centre of each object voxel, and is read into the simulator as a set of three 3D volumes one for each of the x-,y- and z-coordinates. In order to obtain a representation of these basis vectors throughout the whole object voxel, the perturbed field is approximated by a linear spatial function (Eq. 3.11) which was then fitted to the trilinearly interpolated field. As described in the B_0 inhomogeneity section of the previous chapter, gradient fields $\mathbf{G}^p = (G_x^p, G_y^p, G_z^p)$ are calculated for each of the basis vectors (line 30 of Algorithm 1).

The simulator runs through the voxels of the input object and discards all those which have spin density (Speedup4, line 31). The signal is calculated using the function voxel3 which is described in more detail in Algorithm 3. The main goal of this function is to calculate the signal coming from one voxel. To do so, the function uses Eq. 3.22 which is re-arranged in such a way as to allow the most efficient calculation of the various integrals:

$$\begin{aligned}\mathrm{Real}(s_j(v,t)) &= p(v,j)L_xL_yL_z|M_{xy}(v,t_0)|\exp\left(-I_1(t)\right)\mathrm{sincT}\left(\frac{1}{2}\gamma L_x(I_{2,x}(t)+I_{5,x}(t))\right)\\ &\quad \mathrm{sincT}\left(\frac{1}{2}\gamma L_y(I_{2,y}(t)+I_{5,y}(t))\right)\mathrm{sincT}\left(\frac{1}{2}\gamma L_z(I_{2,z}(t)+I_{5,z}(t))\right)\\ &\quad \mathrm{cosT}\left(\gamma(I_{2,x}(t)x_0+I_{2,y}(t)y_0+I_{2,z}(t)z_0+I_3(t)+I_4(t))\right)\ , \qquad (4.5)\end{aligned}$$

$$\begin{aligned}\mathrm{Imag}(s_j(v,t)) &= -p(v,j)L_xL_yL_z|M_{xy}(v,t_0)|\exp\left(-I_1(t)\right)\mathrm{sincT}\left(\frac{1}{2}\gamma L_x(I_{2,x}(t)+I_{5,x}(t))\right)\\ &\quad \mathrm{sincT}\left(\frac{1}{2}\gamma L_y(I_{2,y}(t)+I_{5,y}(t))\right)\mathrm{sincT}\left(\frac{1}{2}\gamma L_z(I_{2,z}(t)+I_{5,z}(t))\right)\\ &\quad \mathrm{sinT}\left(\gamma(I_{2,x}(t)x_0+I_{2,y}(t)y_0+I_{2,z}(t)z_0+I_3(t)+I_4(t))\right)\ , \qquad (4.6)\end{aligned}$$

where sincT, sinT and cosT refer to the look-up tables of functions sinc, sin and cos respectively, $\mathbf{I}_2(t) = (I_{2,x}(t), I_{2,y}(t), I_{2,z}(t))$, $\mathbf{I}_5(t) = (I_{5,x}(t), I_{5,y}(t), I_{5,z}(t))$ and

Algorithm 3 function *voxel3*: calculates signal $s_j(v,t)$ for a voxel v

1: Read in all the input values for v including MR parameters for tissue type j
2: Initialise the signal for v: $s_j(v,t) = zeros(2, Nreadp)$
3: **if** v is the first voxel from the object to go through voxel3 **then**
4: Speedup5: Set up the look-up tables for sinc, sin and *cos*
5: **else**
6: Read in the look-up tables for sinc, sin and *cos*
7: **end if**
8: **for** $t = 1$ to Nevents **do**
9: **if** v is the first voxel from the object to go through voxel3 **then**
10: Speedup6: Calculate the integrals $\mathbf{I}_2(t)$, $I_3(t)$, $I_4(t)$, $\mathbf{I}_5(t)$ and store them in an array $I_{all}(t)$
11: **else**
12: Read the pre-calculated values of the integrals $\mathbf{I}_2(t)$, $I_3(t)$ $I_4(t)$, $\mathbf{I}_5(t)$ from the array $I_{all}(t)$.
13: **end if**
14: Calculate $I_1(v,t)$
15: **if** Events(k) is an excitation **then**
16: Calculate frequency at the center of the voxel v
17: Find the flip angle that corresponds to this frequency in the slice profile
18: Rotate the magnetisation vector of the voxel v by this angle
19: **end if**
20: **if** Events(k) is a signal read-out **then**
21: Calculate equations 4.5 and 4.6
22: Store the solutions into $s_j(v,t)$
23: **end if**
24: **end for**

v corresponds to the voxel with the center at $\mathbf{r}_0 = (x_0, y_0, z_0)$. The integrals are calculated using:

$$
\begin{aligned}
I_1(v,t) &= \int_{t_0}^{t} \frac{1}{T_2^*(v,t)} dt, \qquad & \mathbf{I}_2(t) &= \int_{t_0}^{t} \mathbf{G}(t)\mathbf{R}(t) dt, \\
I_3(t) &= \int_{t_0}^{t} \mathbf{G}(t) \cdot \mathbf{T}(t) dt, \qquad & I_4(t) &= \int_{t_0}^{t} \widetilde{B}_p(\mathbf{r}_0, t) dt, \\
\mathbf{I}_5(t) &= \int_{t_0}^{t} \widetilde{\mathbf{G}}(\mathbf{r}_0, t) dt. & & \qquad (4.7)
\end{aligned}
$$

In order to decrease the amount of calculations per voxel, integrals that are the same for all of the voxels are calculated only once ($\mathbf{I}_2(t)$, $I_3(t)$ $I_4(t)$, $\mathbf{I}_5(t)$). This is

done for the first voxel and the calculated values are stored in an array I_{all}. When the calculations need to be performed for all of the other voxels in the object, the values of the pre-calculated integrals are obtained from array I_{all}. Once the signal $s_j(v,t)$ (both its real and imaginary component) is calculated, it is added to the total signal $S(t)$.

4.3.1 Noise

In the current section noise refers to thermal noise. Thermal noise is modeled as additive, independent, white Gaussian noise in both receiver channels and at this stage is added to the signal. Mean and the standard deviation of the Gaussian are user-defined.

4.4 Image reconstruction

In order for the image reconstruction to take place, k-space data need to be constructed for each slice. The position of the sampling points in the k-space in 2D sequences is completely determined by the pulse sequence using:

$$\begin{aligned} k_x(t) &\equiv \frac{\gamma}{2\pi}\int_{t_0}^{t} G_x(t)\mathrm{d}t\,, \\ k_y(t) &\equiv \frac{\gamma}{2\pi}\int_{t_0}^{t} G_y(t)\mathrm{d}t\,. \end{aligned} \tag{4.8}$$

Signal is arranged into the k-space. Signal value collected at time T, $S(T)$, will be placed at the position $(k_x(T), k_y(T))$.

Once this rearrangement is finished, image reconstruction takes place. This is the final step of the simulation process and can be done using various image reconstruction techniques. The one that is commonly used in the simulator is a direct application of the Fast Fourier Transform (FFT). For more complicated

sequences (including 3D sequences), the raw complex signal values can be saved and passed into any custom reconstruction code.

4.5 Computational time

4.5.1 Introduction

Simulators on the Bloch equation are computationally expensive, as the magnetisation vector must be updated for every defined event which is proportional to at least the number of the event points and the number of volumes in the FMRI sequence. Furthermore as each object voxel has its own magnetisation vector, the whole process is also proportional to the number of object voxels and the number of tissue types in the object. In addition to this, a number of various calculations are done for each of the voxels and for each of the events. Taking all this into account, the total amount of computation required is:

$$Nflops \propto Nevent \times Nvoxel \times Nfev \tag{4.9}$$

where 'flops' stands for the 'floating point operation per second', $Nfev$ is the number of flops per event and per each object voxel. $Nevents$ is the total number of events (rows) in the events sequence:

$$Nevent \propto Nkps \times Nslice \times Nvolume, \tag{4.10}$$

where $Nkps$ is the number of k-space samples needed for each slice. $Nvoxel$ is the number of voxels in the input object:

$$Nvoxel = Nvpv \times Xdim \times Ydim \times Zdim \times Ntissue\,type \tag{4.11}$$

where $Nvpv$ describes the number of object voxels per $1mm^3$, and $Xdim, Ydim$, and $Zdim$ are x-, y- and z- dimensions of the input object in mm.

A typical FMRI sequence that has 64×64 k-space matrix for each slice[1], 25 slices per volume and 100 volumes in total, approximately has 20480000 events. An average input digital brain consists of $181 \times 217 \times 181$ 1mm^3 voxels. A rough estimate for number of flops per voxel is $Nfpv = 1000$. Typically three different tissue types for the brain are used (gray matter, white matter and CSF). Putting all this in the equation we get:

$$N_{flops} = 436785377280000000 \approx 4.4 \times 10^{17} \tag{4.12}$$

For a computer processor capable of 1 GigaFlop, this will take approximately 4.4×10^8 seconds which makes approximately 5000 days! In order to decrease this number, a separate approach was applied on all of the individual factors that directly influence the time: number of events per voxel, number of flops per voxel and the total number of voxels.

4.5.2 Computational speed ups

The main CPU intensive part of the code is the process of updating each magnetisation vector throughout the whole pulse sequence. All other parts of the code: pulse sequence generation, sorting a pulse sequence and a motion file into a large matrix, calculation of the perturbed B_0 values from the basis images, and image reconstruction, only take a few minutes even with the largest pulse sequences or input objects. Therefore the main factors in the computational speed are the number of input object voxels, the length of the pulse sequence and the number of operations done for each voxel.

[1]Word "slice" in the section refers to a slice of the output image.

Number of input object voxels: Speedup2 and Speedup4

When doing multi-slice imaging, in order to decrease the number of input object voxels in a simulation, the simulator uses information about the number of slices, position of the slices, slice thickness, slice profile and the motion sequence in order to choose the Region Of Interest (ROI) of the object which includes only the object voxels that belong to the area which will get excited during the scanning (in Algorithm 1 this process is referred to as Speedup2). Choosing the region of interest influences the dimensions of the input object and therefore the total number of voxels as seen in Eq. 4.11. More specifically, if a user does not want to image the whole object (say all of the image slices required to cover a volume) but needs only a few of those, the simulator adapts to this and will only calculate the signal coming from that selected region of the object. In the case where slice selection is done along the z-axis, choosing the region of interest would influence $Zdim$ in Eq. 4.11. If the choice is to scan the whole input object, the region of interest would be the whole object and $Zdim$ would be at its maximum value. In the case we need to scan only a few slices of that input object, for example two 5mm slices, $Zdim$ would drop to 10mm which decreases the number of voxels, and therefore the computational time tenfold. In the case when there is motion however, or when the slice profile extends beyond its targeted area (as it is the case with most realistic slice profiles) $Zdim$ is not obtained by simply multiplying the number of slices with their slice thickness but an extra amount is added due to this extension. The true equation for $Zdim$ is then:

$$Zdim = Nslice * SlcThk + Zmaxm + SlcP \tag{4.13}$$

where $Zmaxm$ corresponds to the maximum movement that any object voxel did in the z-direction, and $SlcP$ corresponds to the extra area of the object that gets excited due to the non-square slice profile.

Another way of decreasing the number of voxels that go through the calculations is to eliminate all of the voxels that contain no tissue (Speedup4). The proportion of eliminated voxels depends on the sparseness of the object itself. The final equation for the number of voxels is:

$$\begin{aligned} Nvoxel &= P \times Nvpv \times Xdim \times Ydim \\ &\quad \times (Nslice * SlcThk + Zmotion + SlcP) \times Ntissue\,type. \end{aligned} \tag{4.14}$$

where P represents the proportion of the non-zero spin density voxels in the object and $Nvpv$ is the number of voxels per $1mm^3$.

Number of events per voxel: Speedup3

In order to decrease the number of events per voxel, the signal is only calculated for the events which are "relevant" for the voxel in question, i.e. signal is calculated at the read-out points only if they are positioned immediately after the RF pulse which excited the magnetisation vector of that voxel. In case an RF pulse did not excite the magnetisation vector of the voxel, read-out points are skipped and the signal is not calculated but is assumed to be zero. For example, in an ideal situation when the object is not moving and the slice profile is square shaped it is expected that during an acquisition of a volume each of the voxels will get excited only once. The signal coming from the voxel is only recorded at the read-out points directly following the excitation that has happened. Once the recording is done, the signal coming from the voxel is spoiled, and will remain zero until the next time the voxel is excited. In less ideal situations when the object is moving or when the slice profile is non-square shaped and extends beyond the targeted area, voxel can get excited more times, thereby increasing the number of times its signal needs to be included in the read-outs. How this approach

influences the number of events per voxel is shown in the following equation:

$$Nevents \propto Nkps \times (\lceil (Zmotion + SlcP)/SlcThk \rceil + 1) \times Nvolumes. \quad (4.15)$$

Number of flops per event and per voxel: Speedup1, Speedup5, Speedup6

In order to decrease the number of flops per voxel, several measures are taken. They are referred to in Algorithm 1 as Speedup1, Speedup5 and Speedup6. They involve use of look-up tables, pre-calculation of some of the relevant integrals etc. and are described in details in the previous section. In addition to these speed-ups, many other steps were described through-out this and the previous chapter which were used in order to improve the computational efficiency and speed (e.g. when approximating B_p over the voxel, or choosing the efficient ways of representing the rotation matrix). It would be very difficult to calculate the exact number of flops performed by a voxel and therefore even harder to determine the exact amount of improvement due to each of these specific speed-ups.

With all the speed-ups set up, simulation of a typical FMRI sequence described at the beginning of this section: 64×64 k-space matrix for each slice, 25 slices per volume and 100 volumes in total takes 100 days which is significantly better than the originally estimated time of 5000 days.

4.5.3 Parallelisation

All the magnetisation vectors in the simulator are independent of one another 3.13. There are spatial interactions via Maxwell's equations, but these are dealt with by pre-calculated B_p fields. In addition to this, the signal acquisition process is linear as seen in Eq. [3.20] making the signal calculation independent from one voxel to the next. These conditions are correct under the assumption that there is no flow in the input object i.e. spins from one voxel can not move to

the next and so on. The independence of signal generation from one voxel to another provides a good environment for the application of parallelisation techniques, which significantly improve the computational speed. The main idea is to divide an input object into smaller parts, calculate their k-space contributions on many different individual processors, and then, with one master node, sum all of the contributions to form the k-space for the whole object. A descriptive representation of the parallelisation process is shown in Figure 4.4. This parallelisation has been done using SCALI (http://www.scali.com/Dell) version of MPI (http://www-unix.mcs.anl.gov/mpi/).

4.5.4 Results

Generation of a k-space signal for one 3mm thick slice with a 64×64 k-space matrix on an Intel x86 (64-bit) Processor, with 3.6GHz CPU, 2GB RAM running the Linux operating system takes 2min50sec and uses 130MB of RAM. Input object size is $181 \times 217 \times 181 \times 3$ input object voxels (fourth dimension is due to the three tissue types: gray, white and CSF). Using parallelisation techniques, this time is reduced as seen in Table 4.2. In order to generate a typical time series of 100 FMRI volumes for 3 slices of 3mm thickness, with an output k-space matrix of 64×64 for each slice, an 'ideal' square slice profile and no through-slice motion it takes $2\text{m}50\text{s} * 9 * 100 = 42\text{h}30\text{m}$, which is feasible on a single processor, albeit slow. Using parallelisation techniques, the computational time it takes to generate the described FMRI sequence is 6h on a cluster of 16 Intel x86-64. The results were generated on ZUSE, an Intel x86-64 MPI Cluster, which is a part of the Oxford Supercomputing Centre (http://www.osc.ox.ac.uk).

There is a limitation to the speedup achieved by parallelisation i.e. the computational time is not perfectly proportional to the number of processors. This

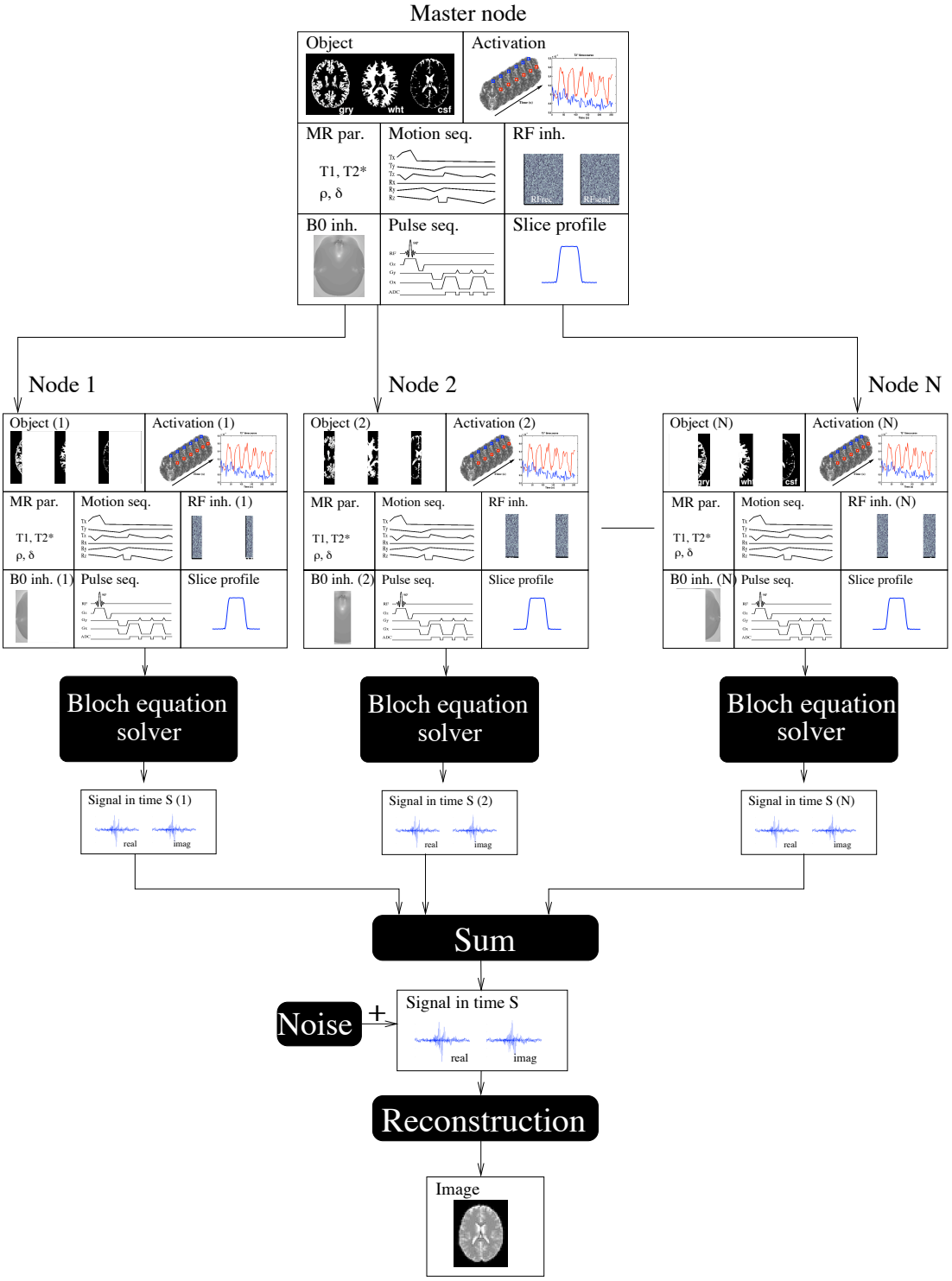


Figure 4.4: Descriptive representation of the parallelisation done in the simulation process.

Cluster Size	64×64	128×128	256×256
1	2m50s	11m5s	42m58s
2	1m37s	5m4s	21m4s
4	54s	2m54s	11m6s
8	32s	1m32s	5m37s
16	24s	58s	3m3s

Table 4.2: Computational time for simulation (k-space generation) of one 3mm slice. Input object size is $181 \times 217 \times 181 \times 3$ input object voxels (fourth dimension is due to the three tissue types: gray, white and CSF), and the object voxel size is $1 \times 1 \times 1$mm, output k-space matrix sizes are 64×64, 128×128 and 256×256 image voxels. Simulations were done on Intel x86-64 MPI Clusters.

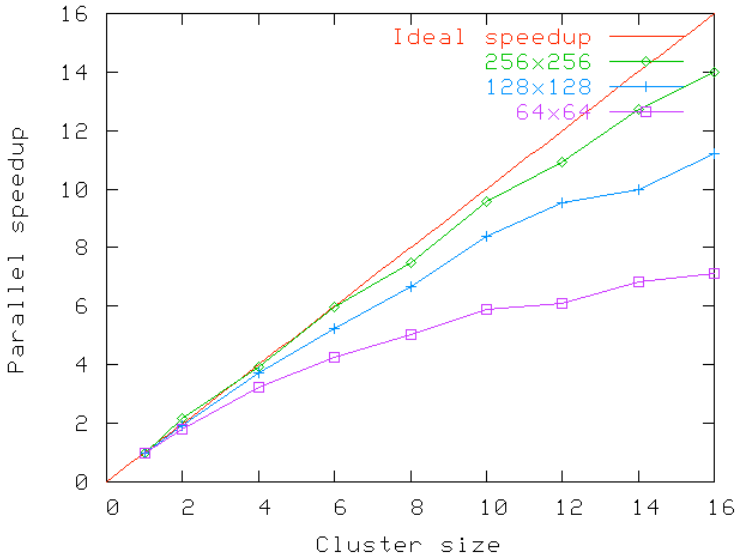


Figure 4.5: Limitation of the parallelisation. The computational time does not decrease perfectly with the increase in the number of processors. Figure courtesy of J. Pitt-Francis.

is due to the fact that there is an overhead amount of calculations that need to be processed by each of the voxels and they do not decrease in the number as the number of processors go up. These overheads, however, do increase greatly with the size of the simulations and do not take longer than a few minutes. The relationship between the number of processors and computational time can be seen more clearly in Figure 4.5.

4.6 Summary

This chapter described the implementation of the simulation model into a simulator environment and discusses the way the computational issues were dealt with in order to make the images realistic and simulation processes both memory- and time-efficient. Due to the high demands on both, FMRI simulation would not be realistically possible otherwise. Within the current framework of the simulator i.e. all of the described speed-ups and the parallelisation it is possible to do simulations of an 2D image within minutes, simulations of an 3D image within an hour and long FMRI sequences within a day. This allows the simulator to be used in practice.

The following chapter is the last chapter in the sequence of the "development" chapters and deals with the validation of the simulator.

Chapter 5

Simulator validation

Following on from the two previous chapters, which considered the model of the simulation process and how this model was implemented in software, this chapter is concerned with validation of the software.

5.1 Introduction

Validation can be performed in many different ways (see Figure 5.1). The ideal validation will have three characteristics. Firstly, it will be quantitative. Secondly, it will use a realistic object, a brain. Thirdly, it will use real scanning data for the comparison with the simulated data. However, it is not often possible to have all these three characteristics present in one validation process. This is because of the large number of scanning factors inherent in the brain scanning process (for example motion, B_0 inhomogeneity, their interactions or BOLD etc.). It is difficult to isolate these factors from one another and therefore to use them in a quantitative validation process. Because of this reason, the approach taken in this thesis has been to use a series of non-ideal types of validations using simpler objects or simple theoretical models (which were experimentally validated in the past), or having purely qualitative results (the simulated images are compared to

the experimental images visually). The series of these, although not completely ideal validations, are still quite significant and should be sufficient for the reliable use of the simulator.

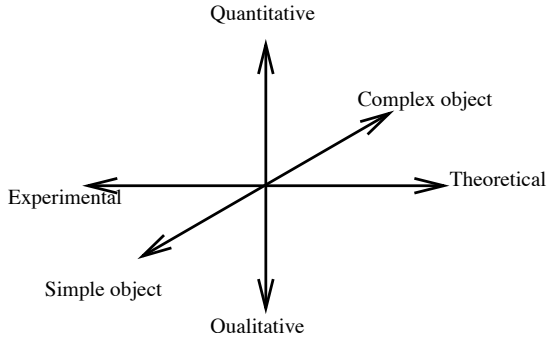


Figure 5.1: A descriptive representation of the different ways a validation can be performed. The x-axis represents whether experimental data or simple theoretical models (experimentally validated in the past) were used for the comparison with the simulated signal. The z-axis describes whether this comparison was done quantitatively or qualitatively. The y-axis describes the input object used. Complexity of the object describes how realistic object is. For example, the brain is the most realistic object. A simple geometrically shaped object (e.g. a sphere) with only one tissue type is an example of a simple object.

Each module of the simulator is validated through a series of independent process which are set out in this chapter. An investigation regarding the properties of the input object is done. More specifically, the sensitivity of the simulator regarding the object voxel size is investigated. This is done by comparing the simulated results for various voxel sizes against the analytically calculated results as if the object is continuous.

The magnetic-susceptibility-induced B_0 inhomogeneity model is validated indirectly using known theoretical models which were experimentally validated by other research groups (more details including the references are set out in the section). Some qualitative results are set out as well. The effect of the B_0 inhomogeneities was also validated through an application in neuronal current imaging which is set out in Chapter 6, Section 6.

The motion model is validated experimentally. The main focus was on the translation in the z direction, where the effects of spin-history were looked at, and the rotation about the y-axis, where the interaction of the motion and the B_0 inhomogeneities was looked at. In addition to this, a purely qualitative result showing blurring due to the rotation about the z-axis is shown. The motion model was also validated through a few applications which are set out in Chapter 6, Sections 2-4.

The chemical shift model is validated indirectly using known theoretical models. Results were generated for a realistic brain phantom and the chemical shift measure was taken for the fat tissue that was part of the brain phantom.

Implementation of the BOLD model and eddy currents model are validated through applications shown in Chapter 6, Sections 4 and 5 respectively. The validation of the physiological basis of the BOLD model was not done as it is beyond the scope of this thesis. The use of the T_2^* changes in order to simulate the BOLD signal in the simulator was based on the works published by Ogawa et al. [57], [58].

The material just described is organised into three groups and presented as three different sections in this chapter. Firstly the results for quantitative comparison of the simulator output with theoretical results are presented. Following this, results for quantitative comparison of the simulator output with the real scanning images are presented. Lastly, qualitative results which demonstrate the possibility of simulation of images impacted by realistic artefacts are presented.

5.2 Quantitative comparison of the simulator output with the theoretical results

In this section we compare the output of the simulations with the theoretical models which are well known and are well established and used in the MRI community. We start with the evaluation of the simulator with respect to the size of the object voxels, then we validate simulator contrast generation and finally we validate simulation of the simple B_0 inhomogeneity effects for which the theoretical results are well documented.

5.2.1 Evaluation of the simulator performance for different object voxel sizes

Input objects for the simulator are discrete, voxel-based volumes and the accurate representation of the underlying structure (virtual brain) depends on the object voxel size, which then influences the simulator output. In order to investigate the sensitivity of the simulator to object voxel size, we have compared images simulated for the discrete objects with varying voxel sizes, with the one generated analytically for a truly continuous object. This was done separately for the in-plane object voxel size (x- and y- direction) and for the through-slice object voxel size (z- direction) as the two have a different impact on the signal.

In-plane tests

For the in-plane tests we have chosen an ellipsoid as an input object with dimensions such that it resembles a normal human brain. In order to be able to calculate an analytical solution the area inside the ellipsoid is homogeneous (with spin density equal to one) and the MRI parameters were chosen in order to ob-

tain a simulated k-space with no T_2^* decay ($T_2^* = 100$ s). One 1mm slice was simulated and simulations were done for a series of object voxel sizes ranging from 5.5×5.5mm to 0.1×0.1mm in plane. An analytic k-space S was calculated by using the analytical expression for the Fourier transform of the ellipse, and evaluating it at the k-space points (k_x, k_y) generated by the same pulse sequence as the one used in simulations:

$$\begin{aligned} S(k_x, k_y) &= (\frac{ab}{C} J_1(2\pi C)\cos(2\pi(k_x x_0 + k_y y_0)), \frac{ab}{C} J_1(2\pi C)\sin(2\pi(k_x x_0 + k_y y_0))) \\ C &= \sqrt{(ak_x\cos(\theta) + ak_y\sin(\theta))^2 + (bk_y\cos(\theta) - bk_x\sin(\theta))^2} \end{aligned} \quad (5.1)$$

where a, b are the semimajor and semiminor axes of the ellipse, x_0, y_0 is the center of the ellipse and θ is the angle between its major axis and the x-axis and J_1 is the Bessel's function of the first order.

In order to evaluate the error of the simulated k-spaces (compared to the analytical k-space) the Root Mean Square (RMS) value was calculated for the difference between analytically calculated k-space $S_a(k_x, k_y)$ and each of the simulated k-spaces $S(k_x, k_y)$. This value was divided with the RMS value of the analytically calculated k-space and in addition multiplied with 100, giving the accuracy as a percentage of the analytically calculated k-space (Figure 5.2):

$$Error = 100 \frac{\sqrt{\frac{1}{4096}\sum_{k_x,k_y=1}^{64}(S_a(k_x, k_y) - S(k_x, k_y))^2)}}{\sqrt{\frac{1}{4096}\sum_{k_x,k_y=1}^{64}(S_a(k_x, k_y))^2)}}. \quad (5.2)$$

The results show that the error in the simulated signal is less than 0.5% of the analytical solution for object voxel sizes that are smaller than 1×1mm in-plane. This is likely to be an acceptable error for FMRI sequences, since typical noise levels are around 1-2% of the total signal mean. With the decrease in object voxel size, the computational time for the simulation increases, which is shown with the red line in Figure 5.2. In order to spend the least amount of computational time but also not compromise the simulated data, we choose the object voxel

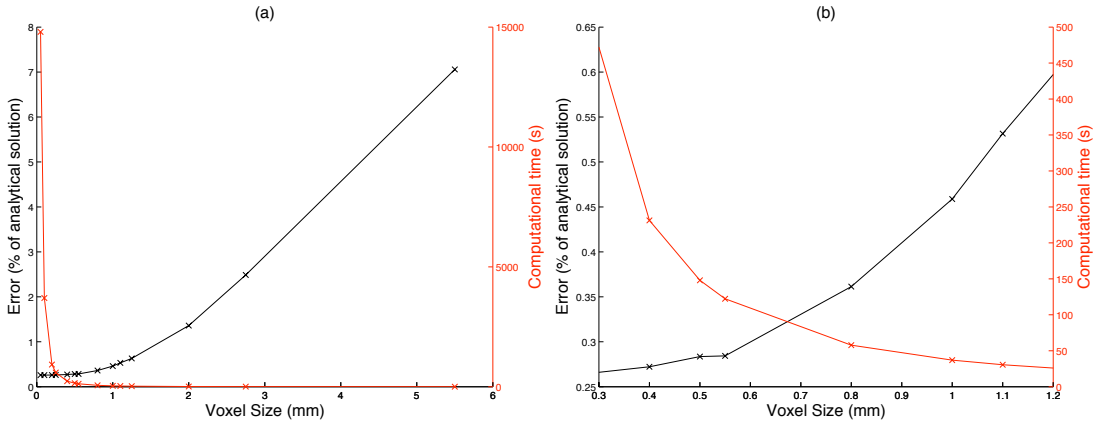


Figure 5.2: The black line and black y-axis in both of the plots show the RMS-based difference between the analytically calculated k-space and simulated k-spaces expressed as a percentage of the analytically calculated k-space signal (plot (b) is an enlarged inset of plot (a)). The red line and the red y-axis show the computational time which it took for the generation of the simulations. It can be seen from the plot (b) that for voxel sizes which are less than 1×1mm in plane, the error in the simulations is less than 0.5% which is likely to be an acceptable error for FMRI sequences, since typical noise levels are around 1-2% of the total signal mean. However, the computational time is increasing as the voxel size goes down, therefore the optimal voxel size depends on the specific simulation and whether the accuracy or the computational speed is a priority.

size according to the simulations we are running, which is 1×1mm in plane for most applications. When decreasing the object voxel size n times for each in-plane dimension, the computational time increases n^2 times. For example, for a typical FMRI sequence described in Section 4.5.3, and for an object voxel of 0.5×0.5mm in plane, the error decreases to 0.2%. However, the computational time increases by 400% (from 42h30m to 170h) which would often be prohibitive for one processor and makes an additional 0.2% error an acceptable tradeoff.

Through-slice tests

For the through-slice tests we have chosen to do tests with a simple homogeneous cubic object (with spin density equal to one) for which we can derive an analytical solution. The main sensitivity of the signal with respect to the object voxel size in

the through-slice direction is due to the excitation mechanism. The excitation of an object voxel is determined by the frequency value at its center. If the center of an object voxel is excited with a flip angle α, the whole object voxel gets excited with that same flip angle, creating a piecewise-constant-shaped approximation of the given slice profile. In order to investigate the situation which will create the largest error, the case when the object is moving in the through-slice direction is investigated.

Parameters used for the simulations are: EPI pulse sequences with flip angles of 30°/60°/90°; square/windowed-sinc slice profiles); in-plane matrix 64×64; 1mm slice thickness; 3 sequential slices per volume; TR of 1.15s; $T_1 = 1.9$s, $T_2^* = 0.3$s and $\rho = 1$. Motion sequence consisted of two back and forth motions: the 20th volume was displaced by +3mm and the 30th volume was displaced by -1mm in the z-direction. The values for the experiment were taken from the paper by Muresan et.al. [55]. The object's total length in the z-direction was fixed and simulations were done for various object voxel sizes ranging from 1mm to 0.001mm in the z-direction.

Motion in the through-slice direction creates changes in the signal due to spin history effects. In a similar fashion as Muresan et.al. [55] these changes are represented using the relative intensity $\triangle I_t = 100\frac{I_t - I_{18}}{I_{18}}, t = (1, 2, ..40)$ of an image voxel situated in the middle slice. By the 18th volume the signal has reached its steady state and therefore the 18th volume is chosen as a reference volume. The results are shown in Figure 5.3 (a) and (b). The signal changes in the first few volumes are due to saturation effects, which in a case of 90° flip angle and an 'ideal' square slice profile is reached in one TR (all the simulations were done with perfectly homogeneous RF and B_0 fields). Spin history effects can be seen in the areas of the 20th and 30th volume. For an 'ideal' square slice profile the steady state was re-established much quicker than for the windowed-sinc slice

profile confirming the more complex situation when a variety of object voxels with different flip angles are present.

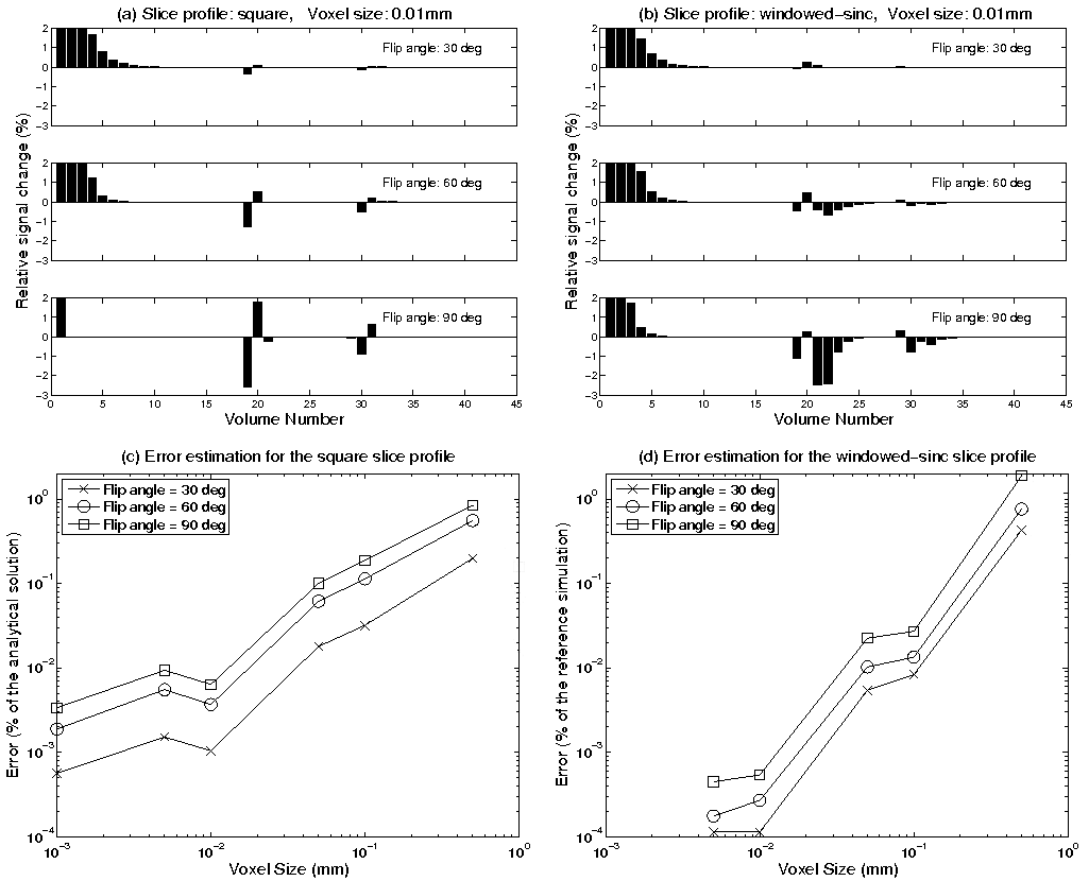


Figure 5.3: (a) and (b) show relative change in the simulated signal due to saturation effects (first few volumes) and the spin history effects caused by the motion in through-slice direction (areas of 20th and 30th volume in the sequence). Different extents to which the spin history influences the signal for different slice profiles can be seen when comparing plots (a), generated with an 'ideal' square slice profile, and (b), generated with a realistic windowed-sinc slice profile. Simulations in (a) and (b) were done with a 0.01mm object voxel size which is generated with a mean error of less than (a) 0.01% of the analytical solution; (b) 0.001% of the simulation for an object voxel size of 0.001mm (we refer to it in plots as the 'reference simulation'). Mean error for various other voxel sizes in range from 0.001mm to 1mm was calculated and is represented in the plots (c) and (d).

The accuracy of the simulations was tested separately for the case of an 'ideal' square slice profile and the case of the realistic windowed-sinc slice profile. For an 'ideal' square slice profile a solution based on theory was calculated using Eq. [3.3]

and Eq. [3.5]. This solution was compared against simulations generated with different object voxel sizes using equation $error(v) = \text{mean}_t(|\triangle I_t(v) - \triangle A_t|), t = (1, 2, ..40)$ where v is the object voxel size, $\triangle I_t$ is the relative signal intensity of the simulations and $\triangle A_t$ is the relative signal intensity of the analytical solution at the time point t (intensities are in k-space).

The results can be seen in Figure 5.3 (c). They show that in order to obtain a reasonable and acceptable simulation accuracy for an 'ideal' square slice profile (given noise levels of 1-2%), an object voxel size in through-slice direction needs to be 1mm or below. Error calculations for the windowed-sinc slice profile were calculated in the same way as for an 'ideal' square slice profile case, with the difference that instead of the analytical solution, simulation with an object voxel size 0.001mm was used as the reference. These results are shown in Figure 5.3 (d). For this situation, in order to achive acceptable errors of less than 1% of the reference signal, the object voxel size also needs to be 1mm or less. Note that all of the the simulations were done for the slice thickness of 1mm which is the worst possible case for the error estimation. In a realistic FMRI sequence slice thickness is usually 4mm or more, which considerably reduces the size of the error.

5.2.2 Image contrast

In order to test the simulation of image contrast in the software, the simulations were compared with the results derived from theoretical expressions for image contrast. These expressions can be found in Sections 15.4.3 and 15.4.4 of Haacke et.al. [29], where they are also plotted for $TE = 0$s (Figure 15.7) and long (infinite) TR (Figure 15.8). These plots were recreated using the same theoretical expressions and parameters as in Haacke et.al., and are shown in Figure 5.4 (dashed lines). In the same figure, plots that use the same theoretical expressions

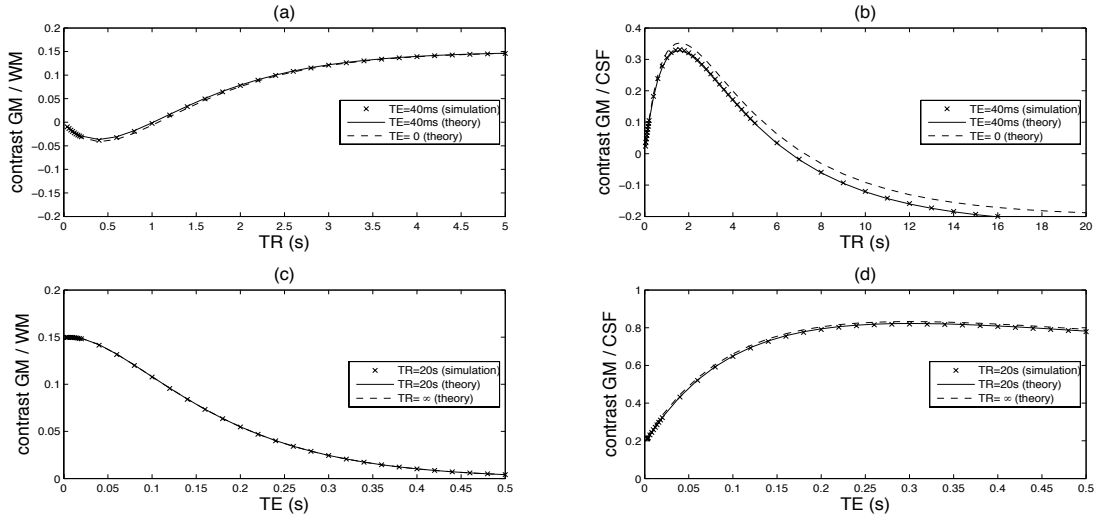


Figure 5.4: Figure shows contrast for (a), (c) gray matter and white matter; (b), (d) gray matter and CSF obtained for a 90° flip angle gradient echo experiment. Contrasts in (a) and (b) are shown as a function of TR for a T_1-weighted scan simulation with $TE = 40$ ms. Contrasts in (c) and (d) are shown as a function of TE for a T_2^*-weighted scan simulation with $TR = 20$ s. Simulation results are marked with crosses and they agree very well with the results derived from the theory (solid and dashed lines - described in the text).

and tissue parameters but more realistic sequence parameters, $TE = 40$ms and $TR = 20$s, are also shown (solid lines). Simulation results were generated using the latter, realistic values for TE and TR and are marked as crosses. From Figure 5.4 it can be seen that the simulation results agree very well with the theoretical results, implying the accuracy of the software implementation. These simple contrast models have been extensively tested in practice [29], [46]

5.2.3 B_0 inhomogeneities

In order to test the simulation of the B_0 inhomogeneity in the software, we independently investigate two main effects of the B_0 inhomogeneity in gradient-echo EPI: translation in the phase encode direction, and signal loss.

Input objects and the perturbed field were chosen such that the theoretical

solution can be calculated. The input object was an ellipsoid (described at the beginning of this section), with the area inside the ellipsoid homogeneous (with spin density equal to one) and MRI parameters chosen in order to obtain a simulated k-space with minimal T_2^* decay ($T_2^* = 100$ s). A perturbed B_0 field was chosen to be spatially constant for the case of testing the translation in the phase encode direction. However, for measuring the signal loss it was chosen to vary linearly, in the positive z direction (through-slice). In this way, we were able to use theoretical results in order to calculate the expected amount of translation and signal loss and compare them to the simulation results.

Simulation results were generated for one 1mm thick slice, and an EPI pulse sequence with an 'ideal' square slice profile and flip angle of 90°. Theoretical results for the amount of translation and the signal loss were obtained by using Eqs. [5.3] which are derived in the same way as in Jezzard et.al. [38] and Deichmann et.al. [20]:

$$\begin{aligned} T_{PE} &= \frac{\gamma}{2\pi} B_p N_y \triangle y \left(\sqrt{\frac{8\ \tau\ \pi}{\gamma\ G_{max}\ N_y \triangle y}} + \frac{N_x - 1}{BW} \right) \\ \frac{\triangle S}{S} &= 1 - \mid \mathrm{sinc}(\frac{\gamma}{2\pi} \triangle z\ \frac{\mathrm{d}B_p}{\mathrm{d}z}\ TE) \mid \end{aligned} \tag{5.3}$$

where T_{PE} is the amount of expected translation for each image voxel in phase encode direction, B_p is the strength of the perturbed magnetic field, BW is the bandwidth, (N_x, N_y) is the size of the k-space matrix, $\triangle y$ is the size of the image voxel in the phase encode direction, τ is the rise time of the gradient coils and G_{max} is the maximum strength of the gradient coils. In the equation for the signal loss, S is the 'ideal' signal, i.e. not affected with B_0 inhomogeneities, $\triangle S$ is the expected loss of the signal S due to B_0 inhomogeneities, $\triangle z$ is the slice thickness, TE is the echo time. In order to obtain the distorted images, an analytic k-space was calculated by using the analytical expression for the Fourier transform of the ellipse (see Eq.[5.1]) and taking into account the theoretically calculated values

for expected translation and signal loss (see Eq.[5.3]). Signal was evaluated using this analytical expression at the k-space points generated by the pulse sequence used in the simulations.

In order to show the amount of translation in the images affected with the offset in B_0 field, 1D profiles of the simulated 2D images are plotted in Figure 5.5 (a), (b) and compared with the 1D profile in a magnetic field with no field offset. Simulations were done for different pulse sequence parameters and perturbed field strengths. Two plots on the left hand side of the figure represent 1D image profiles for two different bandwidths $BW = 85KHz$ (a) and $BW = 100KHz$ (b), and three different values of constant perturbed B_0 field, $B_p = 0T$ (green line), $B_p = 1.5\mu T$ (black line), $B_p = 3\mu T$ (blue line). It can be seen in the plots that the amount of translation (which can be seen as a difference between the green line and the line in question) increases with the strength of the perturbed B_p field. The analytically calculated translations for all of the simulated cases are: 1) $BW = 85KHz$, $B_p = 1.5\mu T$: 0.0127m (3.18 image voxels); 2) $BW = 85KHz$, $B_p = 3\mu T$: 0.0254m (6.36 image voxels); 3) $BW = 100KHz$, $B_p = 1.5\mu T$: 0.0109m (2.73 image voxels); 4) $BW = 100KHz$, $B_p = 3\mu T$: 0.0218m (5.46 image voxels). As expected, the higher bandwidth induced a smaller amount of translation than the lower bandwidth, which is due to the reduced k-space acquisition time. The dashed lines are the 1D image profiles of the analytic results and they agree very closely with the simulation results (often the difference can not be seen).

The plots also show Gibbs ringing effects due to the finiteness of the Fourier series in the process of image reconstruction (this can be more clearly seen in plot (c) which is an enlarged inset of plot (a)). The level of ringing changes as the object shifts: blue line - 0mm shift (0 image voxels); black line - 0.0127mm shift (3.18 image voxels); green line - 0.0254mm (6.36 image voxels). The case

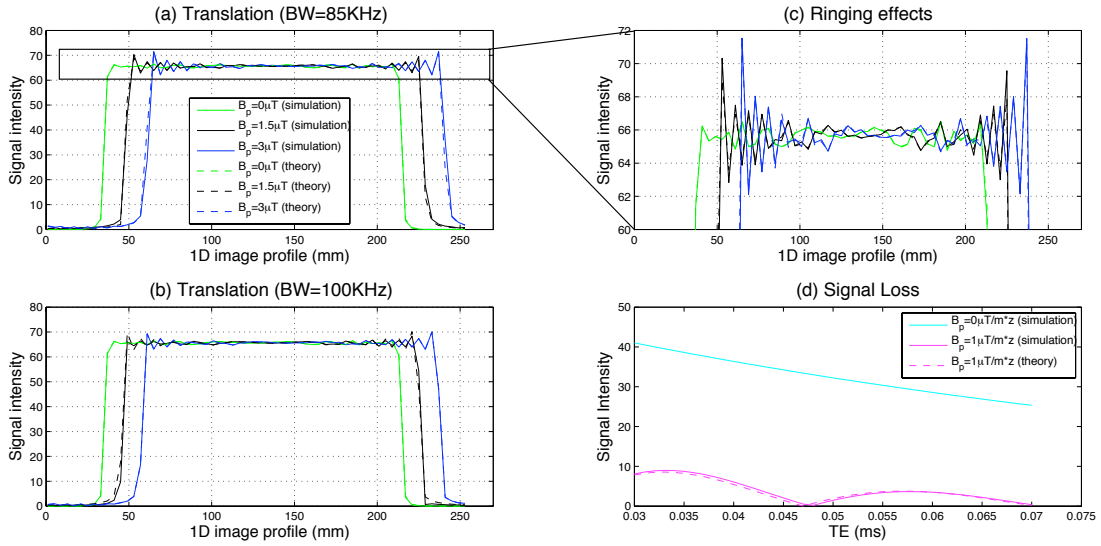


Figure 5.5: Figure shows effects of the B_0 field inhomogeneities: translation in the phase-encode direction ((a), (b) and (c)) and signal loss (d). Plots in (a), (b) represent 1D profiles (along the phase-encode direction) of the simulated 2D images representing one slice of the ellipsoid-shaped object. The plots also show Gibbs ringing effects due to the finiteness of the Fourier series in the process of image reconstruction (this can be more clearly seen in plot (c) which is an enlarged inset of plot (a)). Dashed lines in plots (a), (b) and (c) are 1D profiles (along the phase-encode direction) of 2D images that are the theoretical ('ideal') solutions which we obtained by analytically calculating the k-spaces (described in the text) and they show very close agreement with the simulated results. In some of the cases the difference between the theory and the simulations cannot be seen. Plot (d) is generated for a magnetic field with a linearly increasing inhomogeneity in direction of the positive z-axis $B_p = (\mu T/m) * z$ (full magenta line) while the light blue line shows the signal intensity for a magnetic field with no inhomogeneity. The dashed magenta line represents the analytically calculated signal loss using Eq. [5.3].

which shows the smallest amount of ringing (blue line) is due to an accidental, perfect alignment of the input object and the sampling matrix which happens very rarely in practice. No apodization of the k-space data was included in the image reconstruction. The capability to model ringing artefact demonstrates the usefulness of k-space-based simulations as it can not be generated with image-space-based simulations.

In order to show the amount of signal loss, the signal intensity is plotted (Figure 5.5) (d) for varying TE values in the case when there is no perturbed

magnetic field (light blue line on the right hand plot), and in the case when the perturbed field linearly increases in the z-direction with a slope of $1\mu T/m$ (solid magenta line). Due to its dependence on TE, the dependence of the signal intensity in the distorted image is in the form of sinc function, which agrees with the theoretical results presented in Eq. [5.3] (magenta, dashed line).

5.3 Quantitative comparison of the simulation results with the real scan images

This section presents the work done on the experimental validation of the simulator. Experiments were devised for two different kinds of motion: translation along the z-axis and rotation about the x-axis. These two kinds were chosen as representatives because they respectively cover some of the most difficult motion effects to correct: spin history effects and interaction of the B_0 field inhomogeneities and motion.

Each of the experiments consisted of four steps:

1. Acquiring the images in the scanner
2. Running the simulator using the input values estimated from the scanned images
3. Analysing the scanner-acquired and the simulator-acquired data
4. Comparing the results of the simulator to the experimental results.

5.3.1 Translation in z

This section focuses on work done in validating the performance of the simulator when an object is moving along the z-axis (in either the + or - direction). One

of the most complex effects of this type of motion is the spin history effect and validating that effect was the primary consideration of this section.

Scanning

In order to test the performance of the simulator one of the most important factors was the choice of the input object. For an ideal validation a human subject would be used. However it is almost impossible for a human subject to move in a controllable way. Also, more importantly, there is lots of physiological signal in humans masking other changes. Therefore a simpler object was used: a grapefruit, which has a very high water content making the signal strong, and also which has very long T_1 time ($T_1 = 2531$ms) making spin history effects larger and therefore easier to analyse. This object was attached to a long stiff rod which was placed on the main bed of the scanner. One end of the rod (the one with the attached object) was placed inside the scanner bore, while the other end of the rod was outside the scanner bore and was held by the experimenter. The rod could be manually moved alongside a measurement tape either into or out of the bore which corresponds to +z or -z direction respectively.

An Echo Planar Imaging (EPI) series of 30 object volumes (3D images of the object) was acquired. The pulse sequence parameters used for the experiments were: $TR = 3$s, $TE = 30$ms, a matrix size of 64×64, 25 slices, an image voxel size $4 \times 4 \times 4$mm and a flip angle of $90°$. During the first 20 volumes the object was motionless. During the 21st volume the object was moved into the bore by 5mm (corresponds to 5mm in +z direction). The remaining 10 volumes were again motionless.

In addition to the motion images, additional images of the object were obtained: a T_1- weighted structural image, a field-map image and 12 spin-echo images. These extra images were acquired in order for some of the simulator

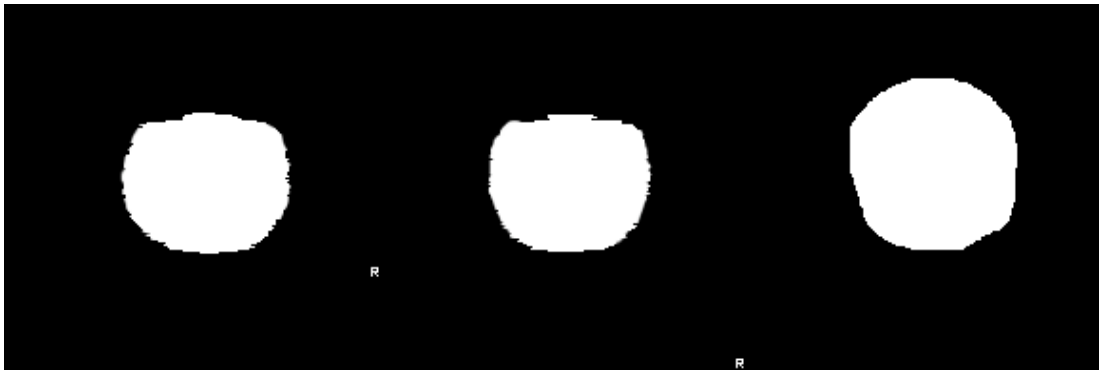


Figure 5.6: Mid-sagittal, -coronal and -axial slices of the input object (grapefruit) are showen from left to right. The size of the input object voxels were $1 \times 1 \times 1$mm.

inputs to be extracted as described in the following section.

Simulating

In order for the simulation to represent the real experiment as close as possible, each of the simulator inputs were derived from the experimentally obtained images of the grapefruit.

Object: In order to have the appropriate anatomical input into the simulator, a T_1-weighted structural image of the object is used. Due to the simplicity of the grapefruit, only one template (matter type) is extracted. The object used for the simulation is presented in Figure 5.6.

B_0 *perturbation field:* The amount of perturbation in the magnetic field is also estimated for each of the voxels. In Chapter 3 a description of the model for calculating B_0 inhomogeneities using known susceptibility values was given. Here a different approach was used. A field-map was firstly acquired (a couple of complex images scanned with different echo times). The field-map is then used to calculate the values of the perturbed field using FUGUE (FMRIB's Utility for Geometrically Unwarping EPIs [67]) . The perturbation images are showen in

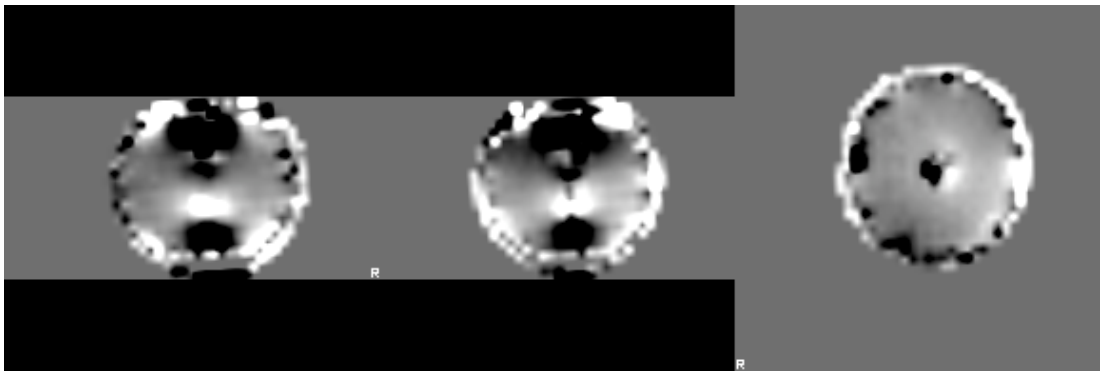


Figure 5.7: Mid-sagittal, -coronal and -axial slices of the pertubation field are shown from left to right. Magnetic field inhomogeneities evaluated for every voxel.

Figure 5.7.

MR parameters: The object was scanned using spin-echo acquisitions with different repetition times in order to measure T_1 and T_2 values [16]. The resulting values were obtained for each image voxel in the volume, which after averaging were: $T_1 = 2531$ms, $T_2 = 444$ms.

Slice profile: The Windowed-Sinc slice profile was estimated from a known RF waveform used for the scanning, and is shown in Figure 4.3.

Pulse sequence: The pulse sequence was generated in the pulse sequence generator of the simulator using the same parameters as the one used in the experiment (described above).

Motion sequence: The object was moved 5mm in +z direction during the acquisition of the 21st volume. However, the exact velocity and timing of motion, the precise times at which the motion began and ended within the 3s that took for the volume to be acquire, are not known. In order to learn more about the motion, the resulting images from the scanner were also examined. Several different options were tried, three of which are shown in Figure 5.8.

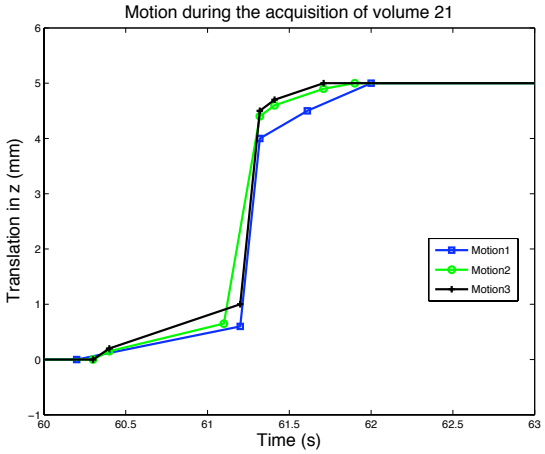


Figure 5.8: The object was moved 5mm during the acquisition of the 21st volume. The exact velocity and timing of this motion are unknown. Therefore, a number of different paradigms which could represent the motion that have occurred, are used in simulations.

Analysing the data

In order to investigate how well the spin history effects were simulated spin history data was analysed in such a way as to emphasise only the spin history effect and not the predominant position displacement effect. The description of the analysis process is shown in Figure 5.9.

The top row represents the data set acquired either by real scanning or by simulation. The first 20 volumes were acquired while the object was motionless. During the acquisition of the 21st volume the object has moved to the new position in the scanner which is 5mm shifted in the +z direction from the initial position. The 22nd to 30th volumes were acquired while the object was motionless and situated in this new position. This is why in the figure these volumes appear slightly shifted compared with the initial volumes. The first four volumes of the data are erased to avoid the saturation effects (they usually occur in the first few volumes until the signal reaches the steady state).

The motion during the acquisition of the 21st volume was in the +z direction.

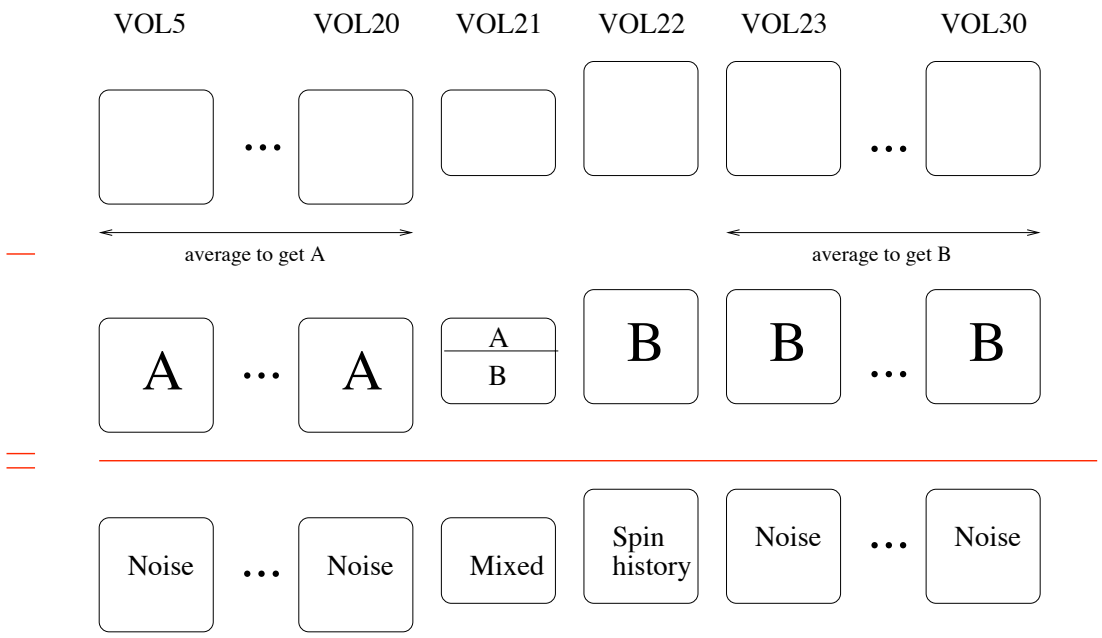


Figure 5.9: Descriptive representation of the analysis process. The bottom row is the first row minus the second row.

However, the slice selection order in this experiment ran in the -z direction. The motion and slice selections therefore took place at the same time but in opposite directions. As a result of this, some of the areas of the object were not scanned and the image of the 21st volume appears as if the bottom section has been lost (Figure 5.9) with mixed effects of position displacement, image distortion and spin history. However, the 22nd volume is affected only by the spin-history artefact. As a result of the movement the spins will "relax" shorter or longer than the fixed time-interval between the volumes (TR). The strongest impact in the signal is created by the spins which did not get excited during the 21st volume, and therefore will "relax" for 6s which is much longer than the fixed 3s time-interval.

In order to capture only the spin history effects, all of the other effects are subtracted from the data. This is done by averaging the first 16 volumes to

obtain volume "A", averaging the last 7 volumes to obtain volume "B" and then connecting all of them into the AB set. The 21st volume of the AB set is obtained by combining the A and B volumes as shown in Figure 5.9. The newly created AB set is subtracted from the data and the required spin history effects are found isolated in the 22nd volume.

Motion effects vary from voxel to voxel within each of the slices. In order to obtain a single value for each of the slices, each slice of the original data was masked to exclude the background values and then averaged. This is done before the analysis process described above.

Results and Discussion

Firstly, the images obtained from the experiment are shown in Figure 5.10. The images acquired by the scanner are shown in the top row and the images acquired by the simulator are shown in the bottom row. Mid-sagittal, -coronal and -axial slices of volume 21 (from left to right) are shown. Motion occurred *during* the acquisition of volume 21. It can be seen that the experimental and the simulation results match well regarding the image distortions due to B_0 inhomogeneities.

The results of the analysis process which investigates the spin-history artefact are shown in Figure 5.11. This figure shows the percentage of the signal change due to the translation in +z direction. The motion happened during the acquisition of the 21st volume and the resulting effects for both the real and simulated data are shown on the left. The spin-history artefact is present in the 22nd volume and this is shown in the plot on the right hand side of the figure. It can be seen from the figure that within a very small range of motion paradigms, the range of spin-history effects can be seen that include the real one (red line). It can be seen also that the results are very sensitive to the particular the motion that occurred during the acquisition of the 21st volume. This is the most likely

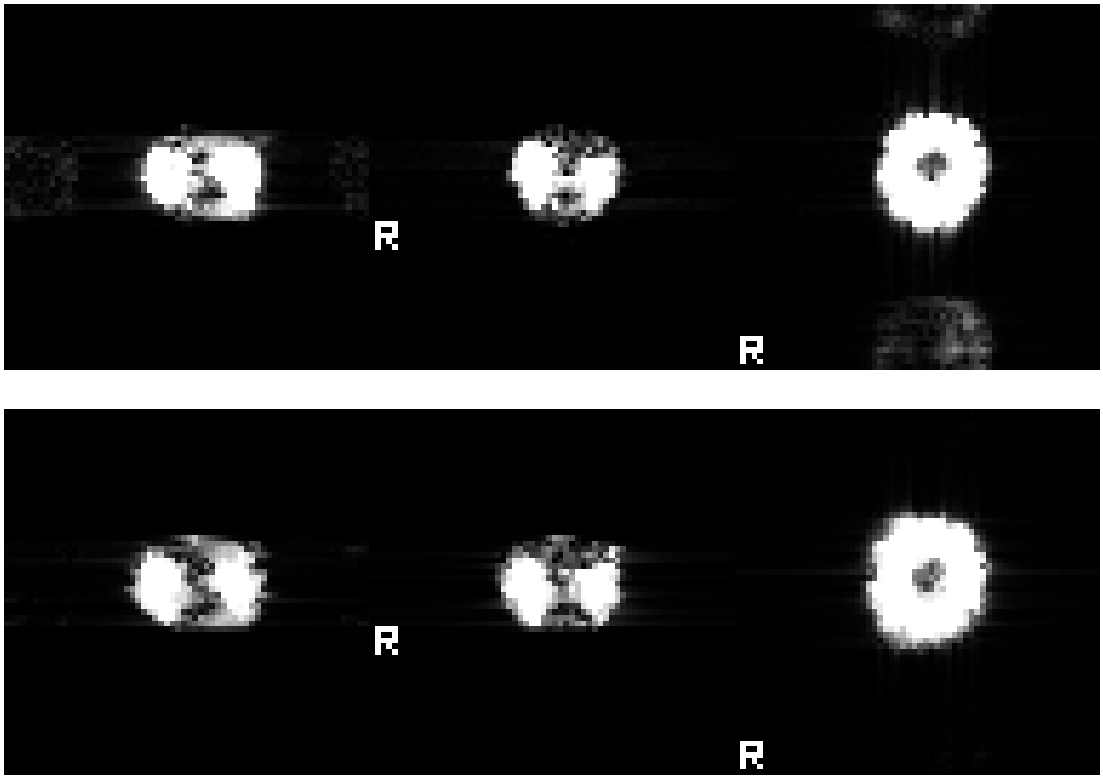


Figure 5.10: Experimental results (top), simulation results (bottom). Mid-sagittal, -coronal and -axial slices of volume 21 are shown (from left to right).

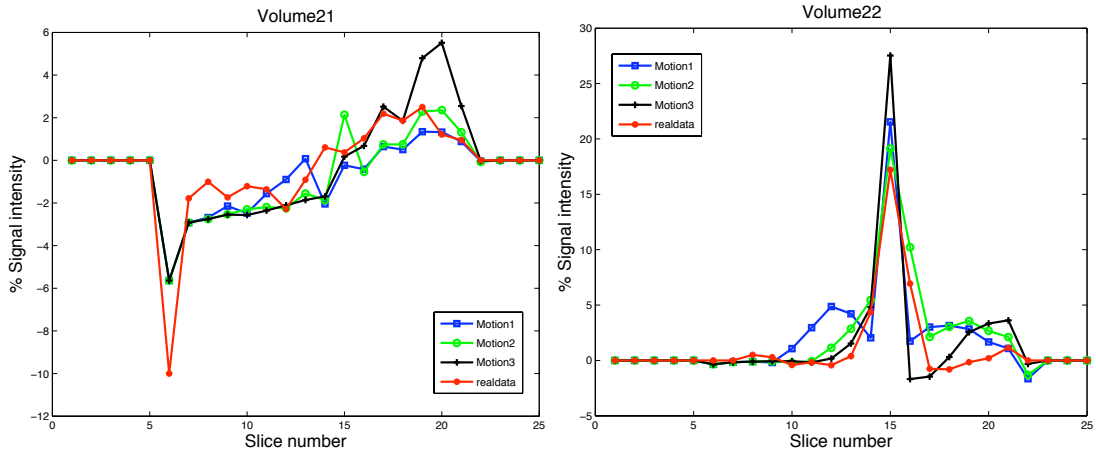


Figure 5.11: The percentage of the signal change due to the translation in +z direction is shown. The x-axis in the plots represents the z-axis of the object and is marked with the slice numbers. Progression from 1 to 25 slices is equivalent to progression in z+ direction in the scanner.

reason for their imperfect match. If it were possible to produce the exact type of motion as the one that happened during the real scanning process we could expect to see a closer match between the real and simulated data than the ones shown in Figure 5.11.

5.3.2 Rotation about the y-axis

This section focuses on work done in validating the performance of the simulator when an object is rotating about the y-axis. One of the most complex effects of this type of motion is the effect of interaction of B_0 inhomogeneities (due to susceptibility) and motion and validating that effect was the primary consideration of this section.

Scanning

As in the case of translation described in the previous section, a simple object was used for the experiments. This object was a glass sphere filled with gelatin. In addition to gelatin, an extra structure built of wood and a plastic air bubble was inserted into the sphere with a goal of creating susceptibility-induced B_0 inhomogeneity artefacts. The object was attached to a long stiff rod which was placed on the main bed of the scanner. One end of the rod (the one with the attached object) was placed inside the scanner bore, while the other end of the rod was outside the scanner bore and was held by the experimenter. The rod could be manually moved alongside a measurement tape either left or right which corresponds to rotation about y-axis of the scanner.

An Echo Planar Imaging (EPI) series of 30 object volumes (3D images of the object) was acquired. The pulse sequence parameters used for the experiments were: $TR = 3$s, $TE = 30$ms, a matrix size of 64×64, 36 slices (with the slice

acquisition order that ran in -z direction), an image voxel size $4 \times 4 \times 4$mm and a flip angle of 90°. During the first 19 volumes the object was motionless. During the 20th volume the object was rotated by 15 degrees about y-axis. The remaining 11 volumes were again motionless.

In addition to the motion images, additional images of the object were obtained: a T_1-weighted structural image, a field-map image and 12 spin-echo images. These extra images were acquired in order for some of the simulator inputs to be extracted as described in the following section.

Simulating

In order for the simulation to represent the real experiment as close as possible, each of the simulator inputs were derived from the experimentally obtained images of the sphere.

Object: In order to have the appropriate anatomical input into the simulator, a T_1-weighted structural image of the object is used. The object used for the simulation is presented in Figure 5.12. The wooden part inside the object was assumed not to give any signal (as it has considerably less water molecules than the rest of the object) and was given a spin density of zero.

B_0 *perturbation field:* As described in Chapter 4, Section 2 a set of precomputed basis functions is needed as an input to the simulator. In order to do this, a perturbation method for solving Maxwell's equations was used [37] (as described in more detail in Chapter 3, Section 3). Two different basis sets were generated, one for the sphere with gelatin and the other for the wooden stick inside the sphere. These two basis sets were then linearly combined in order to form a final basis set for the whole object. In order to calculate the correct coefficients in the linear model of the two basis sets, the appropriate pair was fitted to the experimentally derived perturbed field. This perturbed field was

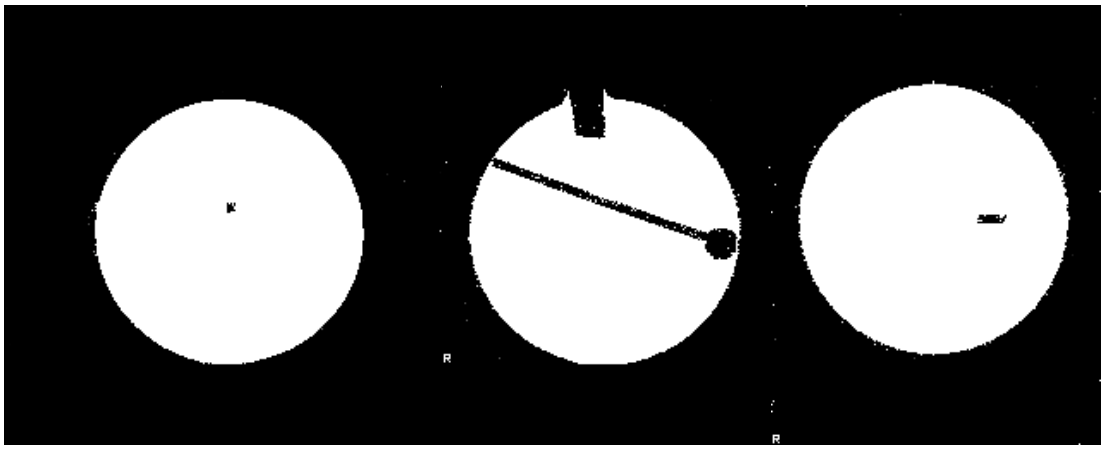


Figure 5.12: The input object. The size of the input object voxels were $1 \times 1 \times 1$mm.

calculated from a field-map acquired during the scanning (using PRELUDE and FUGUE [67]). An example vector of the final basis set is shown in Figure 5.13.

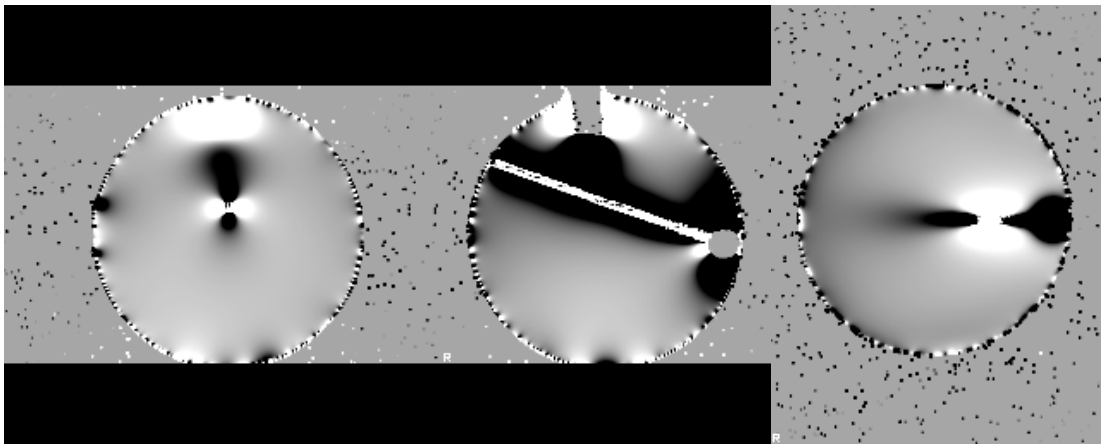


Figure 5.13: An example basis vector of the basis set of the B_0 perturbation field.

MR parameters: The object was scanned using spin-echo acquisitions with different repetition times in order to measure T_1 and T_2 values [16]. The resulting values were obtained for each image voxel in the volume, which after averaging were: $T_1 = 1125$ms, $T_2 = 100$ms.

Slice profile: The Windowed-Sinc slice profile was estimated from a known RF waveform used for the scanning, and is shown in Figure 4.3.

Pulse sequence: The pulse sequence was generated in the pulse sequence generator of the simulator using the same parameters as the ones used in the experiment (described above).

Motion sequence: The object was rotated by 15 degrees about the y-axis during the acquisition of the 20th volume. However, as in the case of translation which was described in the previous section, the exact velocity and timing of motion are not known. In order to learn more about the motion, the resulting images from the scanner were examined to determine the timing by subjective assessment. The resulting motion paradigm was used in the simulations and is shown in Figure 5.14.

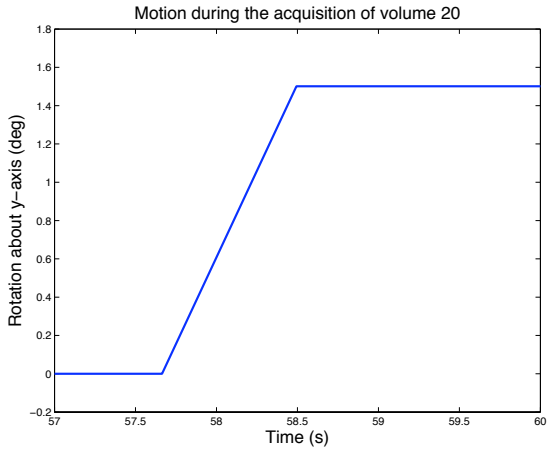


Figure 5.14: The object was rotated by 15 degrees about the y-axis and during the acquisition of the 20th volume. The exact velocity and timing of motion were not known and therefore in order to choose the correct motion paradigm, the resulting images from the scanner were also examined.

Analysing the data

In order to investigate how well the motion interaction with the B_0 inhomogeneities was simulated, data was analysed in such a way as to emphasise only this effect. Firstly part of the analysis was done in the same way as for the previous experiment (Translation in z).

The first four volumes of the data are erased to avoid the saturation effects (they usually occur in the first few volumes until the signal reaches the steady state). The following 16 volumes are averaged to obtain volume "A". The motion happened during the acquisition of volume 20, and this volume (and a subsequent one) will therefore contain mixed effects of spin history artefact, and the interaction between the motion and the B_0 inhomogeneities. In order to avoid both of the effects, these couple of images are also erased from the data and the following 8 are averaged to obtain volume "B". The averaged volumes are then subtracted from the original data set. The resulting data set (specifically volume 20) should contain just the spin-history artefact and the interaction between the motion and the B_0 inhomogeneities. More details about this analysis can be found in the previous section (see Figure 5.9).

The second part of the analysis was done in order to investigate how well the change in B_0 inhomogeneities due to motion was simulated. In order to do this averaged volumes before ("A") and after ("B") the motion took place were compared. Volumes "A" and "B" (which were explained in the previous paragraph) are motion corrected in order to match. A plane that contains the wooden stick (inside the object) and is parallel to y-axis, was extracted from both of the images. The plane from volume "A" was then subtracted from the plane from volume "B". The differences between the two planes (the area underneath the wooden stick) should be only due to the voxels experiencing different amounts

of B_0 inhomogeneities. The direction of the translation in y-direction (phase-encode direction) due to the B_0 inhomogeneities should remain the same between the two. This is because the motion was purely rotation about the y-axis and therefore the two planes remain parallel to the y-axis before and after the motion happened. In addition to these results, a set of data was also simulated without the B_0 inhomogeneities. The same kind of analysis as the one described earlier was applied to this data set.

Results and Discussion

Firstly, the images obtained from the experiment are shown in Figure 5.10. The images acquired by the scanner are shown on the left hand side and the images acquired by the simulator are shown on the right hand side. Mid-sagittal, -coronal and -axial slices of volumes 19, 20 and 21 (from top to bottom) are shown. Motion occurred *during* volume 20 and this can be seen in the images by looking at the wooden stick (which acts as a useful reference point) which is considerably bent in this volume. It can also be seen that the position of the wooden stick is different in volumes 19 and 21 reflecting the time points just before and after the motion took place. It can also be seen that the experimental and the simulation results match well both regarding the gross motion effects and the effect B_0 inhomogeneities.

Results of the first part of the analysis are shown in Figure 5.16. Mid-sagittal, -coronal and -axial slices of volume 20 are shown for both the experimental (top) and simulated (bottom) data. Both sets of data were analysed in the same way which is described in the Methods section. Volume 20 should only contain the interaction between the motion and B_0 inhomogeneities (together with some effect of the spin history). The patterns in the simulated and the experimental data showed a good level of agreement, specifically in the areas around the wooden stick where the largest interactions between the motion and the B_0 inhomogeneities

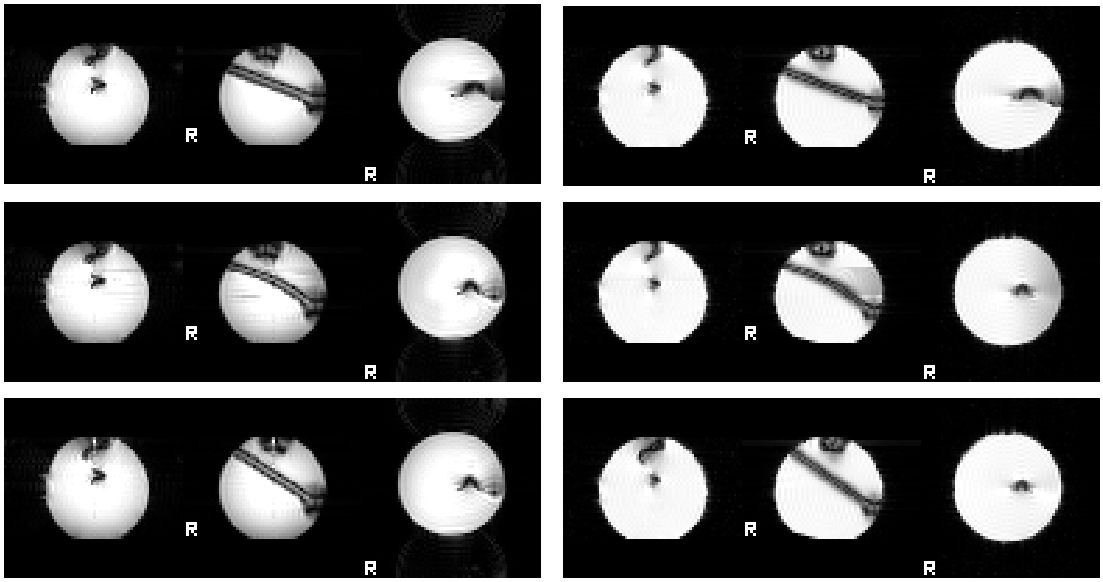


Figure 5.15: Experimental results (left), simulation results (right). Mid-sagittal, -coronal and -axial slices of volumes 19, 20, and 21 (from top to bottom). Motion occurred during the acquisition of volume 20 and this can be seen by the look of the bent wooden stick inside the object.

are expected due to the susceptibility differences between the stick and the surrounding gelatin. These can be seen mainly in the middle and the right hand images.

The results of the second part of the analysis process are shown in Figures 5.17 and 5.18. Figure 5.17 shows the plane from volume "A" containing the wooden stick and is parallel to the y-axis (left), the plane from volume "B" containing the wooden stick and is parallel to the y-axis (middle) and the subtraction of the planes (right). The experimental results are shown in the top row, the simulation results which were generated with the presence of B_0 inhomogeneities are shown in the middle row, and the simulation results which were generated without the presence of B_0 inhomogeneities are shown in the bottom row.

The two top rows in the figure agree well in the way B_0 inhomogeneities affected the images before and after the motion occurred. This can be more clearly

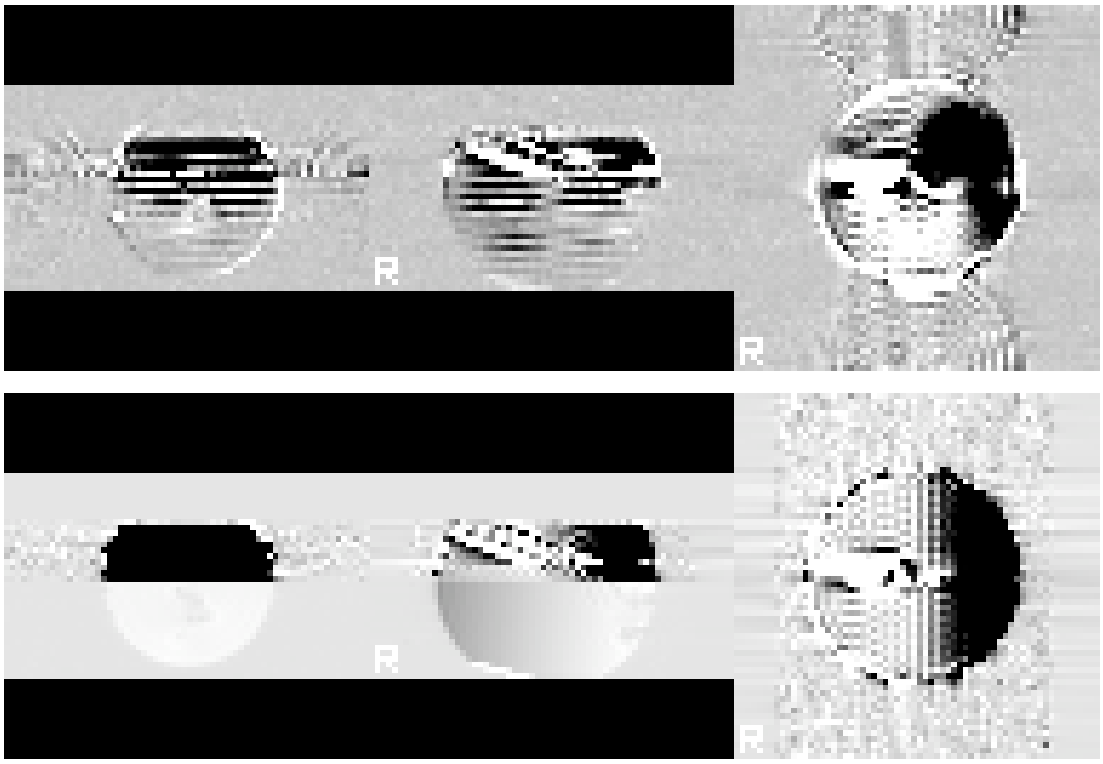


Figure 5.16: Intensity patterns in the motion-affected volume 20 are shown for both the experimental (top) and simulated (bottom) analysed data. The images contain the effects of interaction of the motion and B_0 inhomogeneities, (though some spin-history effects are also present).

seen in the difference images on the right hand side of the figure which are showing the difference in the image intensity due to B_0 inhomogeneity-based distortion which was changing with the positional displacement of the object. On the other hand, the difference image on the bottom, generated as if there is no B_0 inhomogeneity, shows that the position displacement did not greatly affect the intensity of the images. In order to see these results more clearly, an averaged 1D profile of the images in Figure 5.17 is shown in Figure 5.18. It can be seen in Figure 5.18 how 1D profiles of the experimental and the simulated data (with B_0 inhomogeneities) match well. The simulated data generated without B_0 inhomogeneities confirms that the change in the profiles is due to B_0 inhomogeneities.

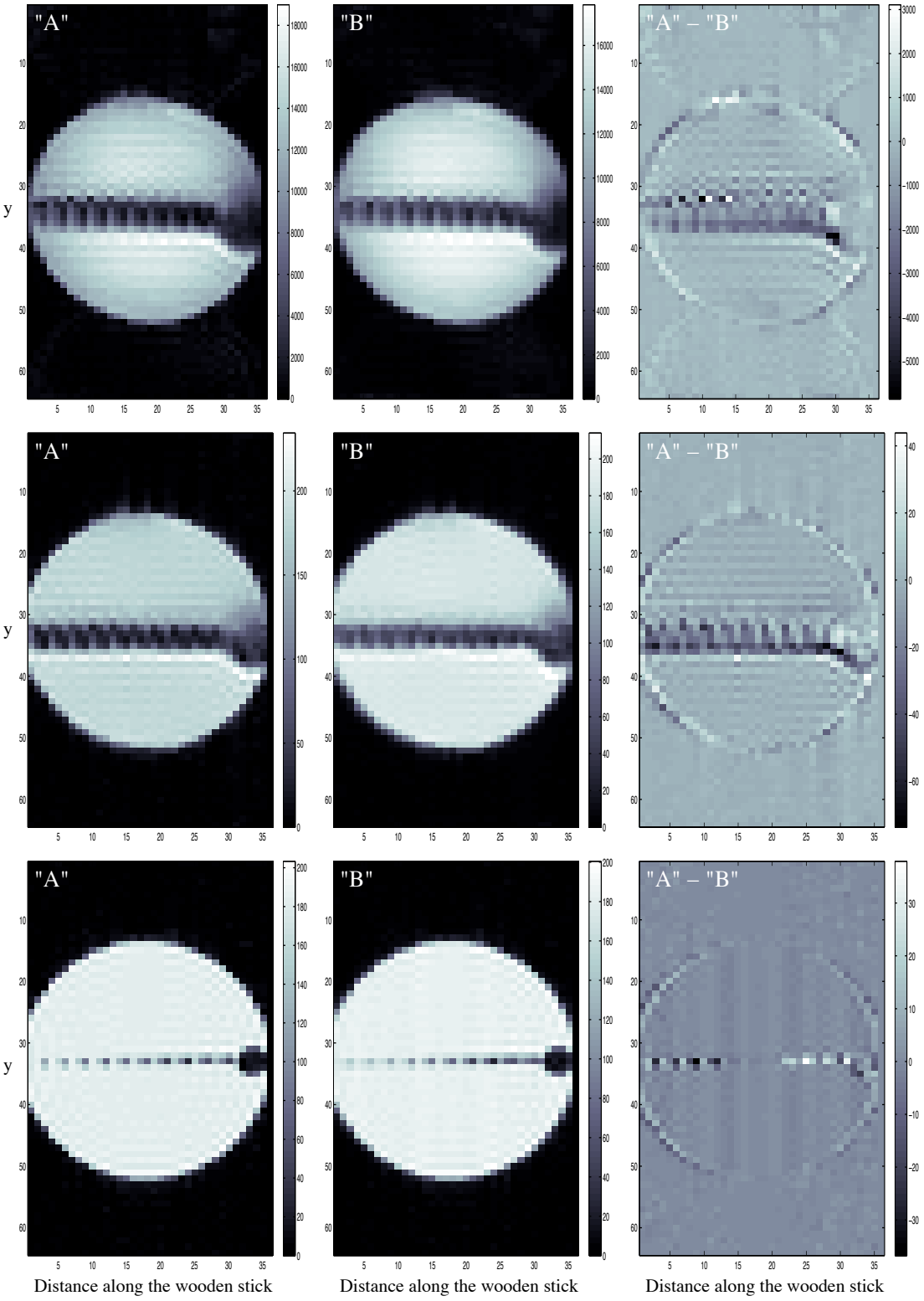


Figure 5.17: The plane from volume "A" containing the wooden stick and is parallel to the y-axis (left), the plane from volume "B" containing the wooden stick and is parallel to the y-axis (middle) and the subtraction of the planes (right). Note that the vertical axis in this figure is the y-axis, and the horizontal axis is the axis along the wooden stick (both in voxels). The experimental results (top), the simulation results with B_0 inhomogeneities (middle), and the simulation results without the presence of B_0 inhomogeneities (bottom).

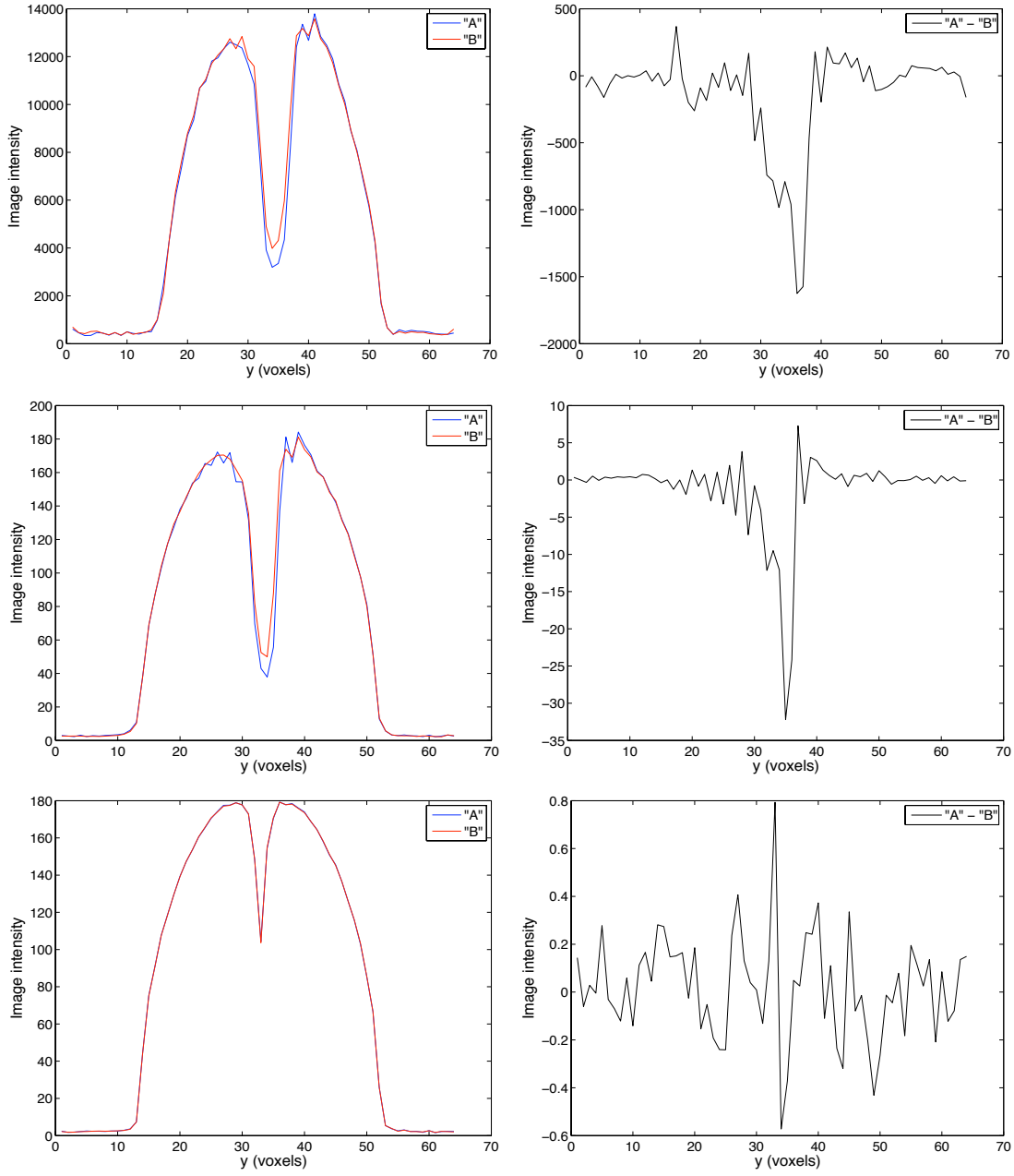


Figure 5.18: Averaged 1D profiles of "A" (the blue line on the left) and "B" (the red line on the left) images shown in Figure 5.17 and their subtraction "A" - "B" (the black line on the right). The averaging took place along the direction of the wooden stick. The experimental data are shown in the top row, the simulated data (with B_0 inhomogeneities) are shown in the middle row, and the simulated data (without B_0 inhomogeneities) are shown in the bottom row.

5.4 Qualitative comparison of the simulation results with real scan images

In this section a small selection of examples of simulated datasets are shown which are using a realistic brain phantom as an object. Most of these examples of these examples are qualitative in nature, (with an exception of the chemical shift simulation) and we present them in order to illustrate the similarity between the simulated and real images. Artifacts presented in this section are: ghosting, chemical shift, B_0 inhomogeneity, eddy currents and within-scan motion.

In all of the examples we used the McGill virtual brain [19] with the parameter values given in Table 4.1, at 3T magnetic field strength, and an EPI pulse sequence which we generated with parameters: TE = 30ms, TR = 3s, flip angle 90°, maximum gradient strength 35mT/m, rise time 0.2ms, and crushers of maximum magnitude. It is important to note that even very small mistakes in generating the pulse sequence cause various artefacts in the simulated images, as is the case with a real scanner. An example of this is ghosting (Figure 5.19) that we obtained

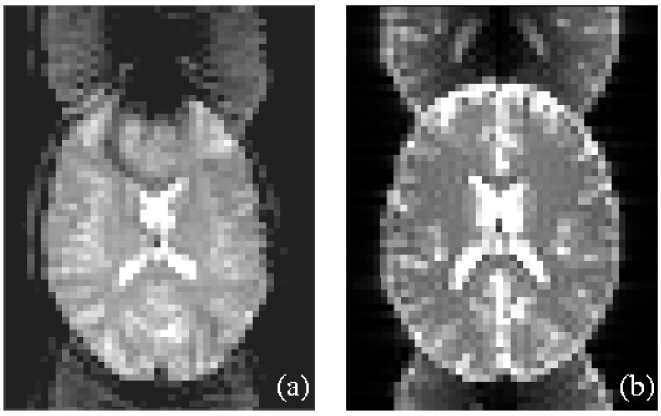


Figure 5.19: Figures show ghosting due to a user-gradient mis-specification both in a real image (a) and a simulated image (b). The additional distortion in the frontal area of the brain in the real image is due to the susceptibility effect which was not included in this simulation. The resolution in the cross-sectional plane in both images is 4×4mm.

by using a pulse sequence with a very small (less than one millisecond) delay in the echo timing.

Chemical shift artefact is shown in Figure 5.20. The figure is generated using grey, white, CSF and fat tissues and represents the effect of difference in the chemical shift between fat and other tissues in the brain. It also shows the

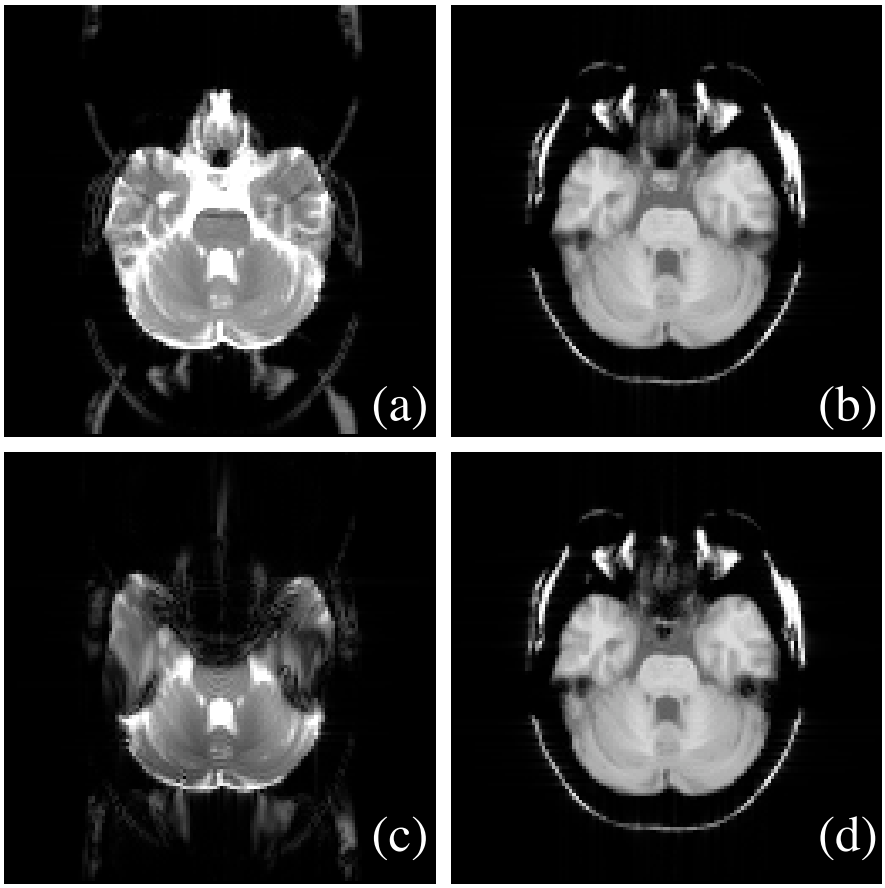


Figure 5.20: Figures showing the effect of chemical shift at 3T in (a) an EPI image without B_0 inhomogeneities; (b) T_1-weighted standard gradient echo image without B_0 inhomogeneities; (c) an EPI image with B_0 inhomogeneities; and (d) T_1-weighted standard gradient echo image with B_0 inhomogeneities. All images are simulated with a resolution in the cross-sectional plane of 2×2mm. The shift of the fat tissue due to the chemical shift is: (a) and (c) 178mm in phase encode direction (along negative y-axis) which at this resolution is 79 image voxels; (b) and (d) 1.1mm in the frequency encode direction (along positive x-axis) which at this resolution is 0.55 image voxels. These shifts were calculated from the theory and they match closely with the simulations. It can also be seen in the figure how the effects of the artefacts are much stronger in the EPI image than in the standard gradient echo image.

different intensity and characteristics of the chemical shift artefact in two different pulse sequences: an EPI imaging sequence (resolution 2×2mm, TE = 30ms, TR = 3s) and a T_1-weighted standard gradient-echo imaging sequence (resolution 2×2mm, TE = 8ms, TR = 300ms). Both imaging sequences used 90° flip angle, maximum gradient strength 35mT/m, rise time 0.2ms, and crushers of maximum magnitude. Simulations were generated without B_0 inhomogeneities as shown in Figures 5.20 (a) and (b), and with B_0 inhomogeneities in Figures 5.20 (c) and (d). Chemical shift values that were used are shown in Table 4.1.

The expected amount and the direction of the resulting shift of the fat tissue in the output images that is due to the chemical shift differences is analytically calculated for both sequences using Eq. [5.3] (EPI sequence) and equation $\triangle x = \delta B_0/G$ [48] where $\triangle x$ is the amount of the translation in the x direction, δ is the shielding constant and G is the frequency encoding gradient (standard gradient echo sequence). For the EPI sequence it is estimated to be 178mm (out of 256mm in total) in the phase encode direction (along the y axis) and that is seen in Figures 5.20 (a) and (c); and for the standard gradient echo sequence, it is 1.1mm (out of 256mm in total) in the frequency encode direction (along the x-axis) which is shown in Figures 5.20 (b) and (d) but is so small that can not be seen simply by eye. In Figures 5.20 (a) and (c) we can also see some effects of ghosting of the fat signal which is also part of the chemical shift artefact.

Figure 5.21 represents the effect of B_0 field distortions in the simulation, shown in (b), versus the case with no B_0 field distortions, shown in (a). We used an EPI pulse sequence and a resolution of $4 \times 4 \times 6$mm in this simulation. As expected, the biggest distortions and signal loss we obtained were in the areas close to the air pockets in the skull, which look very similar to a real scanning case shown in (c).

We also simulated the effects of the eddy currents. An example of the simula-

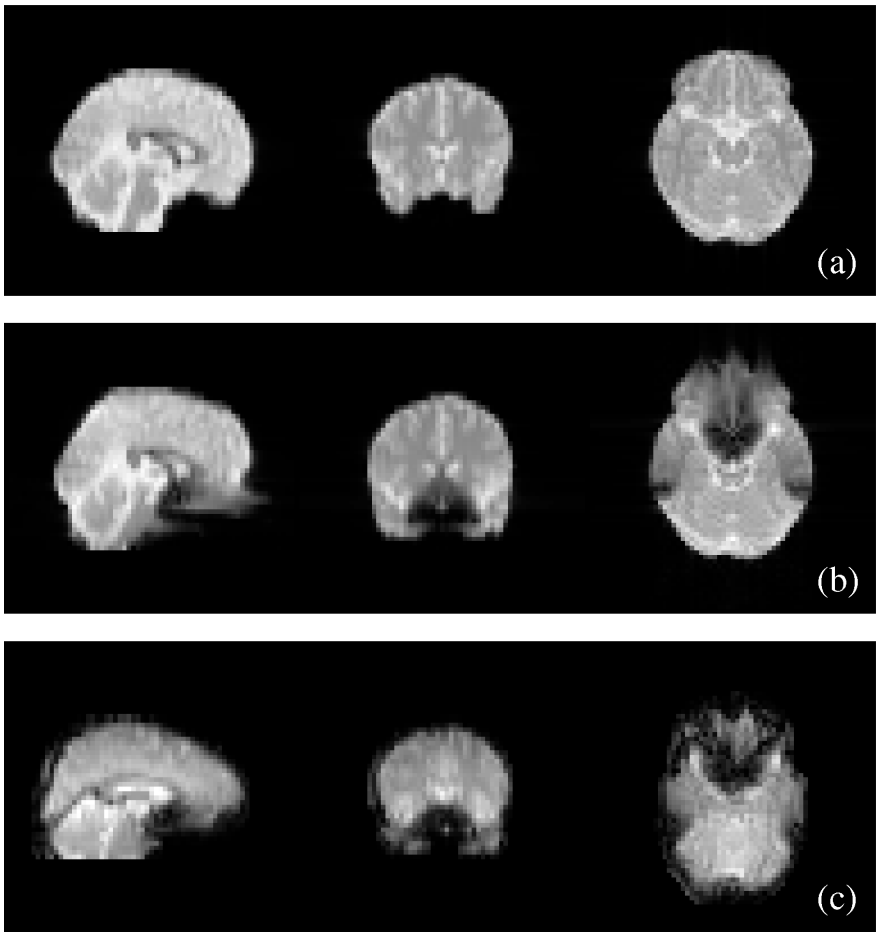


Figure 5.21: An example of an EPI scan of sagittal, coronal and axial slices generated at 3T magnetic field and with resolution $4 \times 4 \times 6$mm, TE = 30ms, flip angle 90°, maximum gradient strength 35mT/m and rise time 0.2ms, matrix size 64×64 where (a) is a simulated scan with no artefacts, (b) is a simulated scan with B_0 field distortion and (c) is a real scan (different subject) with B_0 field distortions. Note how similar the real (c) and the simulated (b) distortions look like, specially distortions in the frontal lobe and around the ears.

tion is presented in Figure 5.22 with a resolution of 2×2mm in the cross-sectional plane. The eddy currents in this example were generated by gradients applied in the read-out direction before the actual signal acquisition took place. More details about the simulation of eddy currents are given in Chapter 6.

Blurring due to the object's movement during the read-out period was also simulated. Figure 5.23 shows the effects of a rigid-body rotation about the z-axis

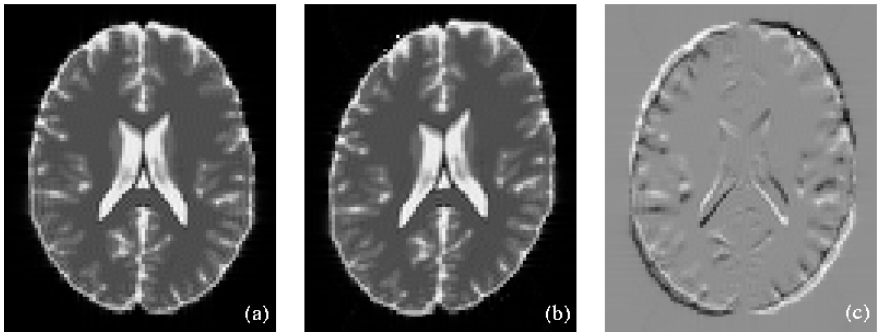


Figure 5.22: Example images showing the effects of eddy currents on an EPI image: (a) T_2-weighted image; (b) T_2-weighted image with eddy currents; (c) Subtraction of image (a) from image (b). All images have a resolution of 2×2mm in plane and TE = 30ms.

that happened during the signal acquisition time i.e. during the read-out period. Note that the motion is exaggerated here (9 degrees) to make the artefact clearly visible although much smaller movements still induce artefacts which, although not easily visible to the eye, are potentially large enough to significantly impact the FMRI statistics.

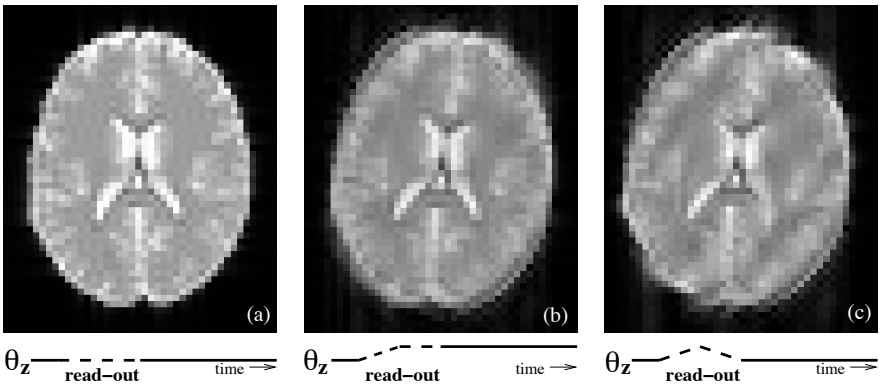


Figure 5.23: Example images showing the effects of within-scan motion (rigid-body rotation about the z-axis) on an EPI image: (a) no motion; (b) constant rotation; (c) back and forth rotation. Note that the degree of motion is exaggerated here (9 degrees) to illustrate the effect visually. The line underneath the image represents the angle of rotation as a function of time with the read-out portion of the sequence represented with a dashed line.

5.5 Summary

This chapter described the steps taken in order to validate the simulator. The sensitivity of the simulator regarding the object voxel size was investigated. The magnetic-susceptibility-induced B_0 inhomogeneity model was validated. The simulated results, the amount of translation in phase encode direction, and the amount of signal loss, showed excellent agreement with the known theoretical results. The motion model was validated experimentally using simple objects for which MR relaxation times were measured and field maps were calculated. The scanner-acquired data showed very good agreement with simulator-acquired data for all of the cases of motion. Chemical shift results were generated for a realistic brain phantom. The translation observed in the fat tissue due to the chemical shift agreed excellently with the shift calculated using the theoretical equations.

So far in this thesis the whole process of the development of the simulator has been described: the simulation model, the implementation of that model and the validation of the simulator. The next chapter describes the applications of the simulator, demonstrating its usefulness in various aspects of MRI-related research.

Chapter 6

Simulator applications

The simulator can be applied in a number of diverse applications which span both MRI and FMRI fields. These applications include the simulation and removal of various imaging artefacts (such as B_0 inhomogeneities, rigid-body motion etc.), development of new pulse sequences, evaluation and validation of software tools for FMRI analysis methods (e.g. motion correction, registration, statistical analysis of images, etc.), or it can be used as an educational tool.

In this chapter five different applications of the simulator are presented. These are: quantifying the performance of motion correction software; investigating the performance of Independent Component Analysis (ICA) as a tool for quantifying motion-related artefacts; evaluating the impact of stimulus correlated motion artefacts; comparing different acquisition techniques for eddy-current compensation; and reproducing and extending experiments in neuronal current imaging.

6.1 Quantifying the performance of a motion correction algorithm

In this section we focus on quantifying the performance of a motion correction algorithm. The motion correction algorithm in question is called MCFLIRT [6], [36] and is a part of FSL (FMRIB's Software Library, www.fmrib.ox.ac.uk/fsl) [67].

6.1.1 Introduction

Motion correction is an important issue in FMRI analysis as even the slightest patient motion (1 - 2 degrees of rotation or a few mm translation) during a scan can mean that a voxel location does not correspond to the same physical point in the volume as the same voxel in a subsequent volume, given a spatial resolution of a few mm in the acquisition process. In addition, the change in signal intensity due to motion can be far greater than the BOLD effect itself, particularly at tissue boundaries, at the edge of the brain or near major ventricles. Before analysis can take place to get the activation patterns of different parts of the brain, the acquired images must be motion corrected in order for the physical and image coordinate systems to be coincident, and for the artefacts to be removed.

At present, most FMRI motion correction methods assume a simple model of rigid-body image transformation. In order for this approach to be accurate two assumptions are necessary. First is that the motion happened instantaneously and between the acquisition of the two neighbouring volumes (3D images) i.e. just after the acquisition of one volume finished and before the acquisition of the next volume started. This is descriptively shown in Figure 6.1, Level1. With this assumption, artefacts such as blurring and slice misalignment do not occur.

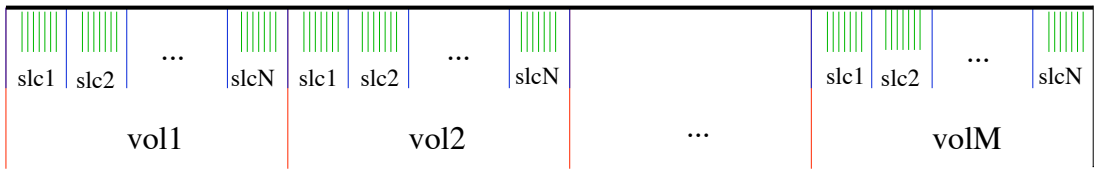


Figure 6.1: Figure describes at which points in the pulse sequence we can assume that the motion has happened. The black line represents the image acquisition process in time. Red lines describe the motion occurring instantaneously between the acquisition of the volumes (Level1), blue lines describe the motion occurring instantaneously between the acquisition of the slices (Level2), and the green lines represent that motion is happening continuously throughout the whole image acquisition, including the acquisition of each of the slices (Level3).

The second assumption is that there is no interaction between motion and B_0-inhomogeneities. These motion correction methods typically compute the image transformation that will match a 3D image at time point t to a reference template 3D image by optimising some cost function constructed to express image similarity. Commonly these type of methods are classified as image-similarity-based motion correction algorithms. Some commonly used ones are the one of Friston et. al. ([25], [26]) in the Statistical Parametric Mapping (SPM) package, the one described by Woods et. al. ([70]) in the Automated Image Registration (AIR) package and the one described by Jenkinson et.al [36], [6] in the FMRIB's Software Library (FSL) package [67]

Motion correction algorithms are one of the necessary parts of every FMRI data analysis package. However, most of the motion correction techniques that are developed are based on analysis of empirically acquired FMRI data, so that the ground truth is not known and therefore the precise accuracy of the methods difficult to estimate (e.g. see [32, 25, 41, 36]).

Accuracy testing of motion correction algorithms is usually done by the use of simulated data. These data are usually constructed by taking a real image from the scanner and applying 3D transformations to it in order to generate

the motion corrupted time-series. Due to this approach, some of the important motion artefacts are missed out and as a result the data is often simple and unrealistic making the full evaluation of the performance of the motion correction methods very difficult.

The simulator has the capability to simulate realistic images with complex motion-related artefacts such as spin history or blurring. In addition, the simulator does this in a controlled way, i.e. various artefacts can be simply turned "on" or "off". Due to these properties, the simulator can be used for testing image analysis softwares. In this section we focus on quantifying the performance of the motion correction algorithm MCFLIRT [6], [36].

An accuracy assessment of this software was done by Bannister [6], however the simulated data used for the assessment was empirically derived. A high-resolution EPI volume ($2 \times 2 \times 2$mm voxel dimensions) was duplicated 180 times and each volume was transformed by an affine matrix corresponding to real motion estimates taken from an experimental study. The images were then sub-sampled to $4 \times 4 \times 6$mm voxels. It is expected that the effect of interpolation when applying the transformations to the images will be minimised with this approach of applying the transformations to a high-resolution images first and then sub-sampling them. However the effects of the interpolation can never be completely avoided.

Furthermore, the images were generated by applying one affine transformation for each of the time points, as if motion happened instantaneously between the acquisition of volumes acquired at the neighbouring time-points (Level1 motion). This approach is sufficient for testing the accuracy of the MCFLIRT algorithm in ideal conditions but is not sufficient to evaluate the performance of the algorithm in realistic scanning conditions and with data that has realistic and more complex motion effects.

By using the simulator it is possible to address both of these issues. Firstly, the problem of interpolation would be avoided. Secondly, the simulator is capable of simulating all of the motion artefacts represented in Figure 2.18 and on any of the levels represented in Figure 6.1. This provides excellent grounds for evaluation of the performance of the MCFLIRT algorithm in realistic, but still controllable situation. In order to show the approach used to do this, a brief description of the MCFLIRT algorithm is needed and is given in the next section.

6.1.2 MCFLIRT Algorithm

The MCFLIRT algorithm is an image-similarity-based motion correction algorithm. In order to correct the motion-corrupted images, first a reference image is chosen from the time-series. Which volume is taken to be the reference image is a user-defined option. The two most common options are the middle image (default option in MCFLIRT) and the first image. All of the other images are then registered to the reference image. The registration process is computed by constructing a cost function which quantifies the dissimilarity between the two images and then searches for the transformation which gives the minimum cost value.

The method used in MCFLIRT for the process of minimisation is known as golden section search [6]. Golden section search is performed on each of the 6 rigid-body parameters in turn. It is performed for the rotational parameters first as these have a greater impact on the overall movement of the object than the translational parameters (e.g. 2 degrees rotation centered in the middle of the brain results in approximately 4mm translation at the edge of the brain) [6]. Searching in each of the 6 parameter directions stops when a specified tolerance has been reached. The default tolerances in MCFLIRT are 0.057 degrees for the

rotations and 0.02mm for the translations.

In the MCFLIRT algorithm a cost function is selected by the user and can be: mutual information, correlation ratio, normalised correlation, normalised mutual information or least squares. Each cost function is regularised via the introduction of geometric smoothing. This is done in order to avoid small discontinuities as the transformation parameters are varied smoothly which can create local minima "traps". The default cost function used in MCFLIRT is normalised correlation.

When calculating the dissimilarity between the reference image (I_r) and the image we want to register to the reference image (I), interpolation needs to be applied to I. This is a process that calculates the intensity in the image I at points between the original image voxel points. The interpolation choices in MCFLIRT are: trilinear, sinc and nearest neighbour. Different choices of the interpolation function are used for different stages of the algorithm.

The registration process is done in a few different stages. The first stage is performed on data which is sub-sampled to 8mm isotropic voxels to allow the gross intensity patterns in the images to drive the registration. The optimisation during this stage is performed using trilinear interpolation and the stopping tolerance for the golden search process is specified to be relatively large (0.45 degrees for the rotations, and 0.16mm for the translations). The second stage is performed on 4mm isotropic voxels, using trilinear interpolation and the same tolerance level as the one defined during the first stage. During the third stage the size of the voxels is maintained at 4mm, the interpolation is still trilinear but the tolerance is reduced to the default values (0.057 degrees for the rotations and 0.02mm for the translations). These three stages are the default stages of the MCFLIRT algorithm. An extra fourth stage is optional, and is the same as the third stage except that the sinc interpolation method is used instead of the trilinear interpolation method.

Once the minimisation process is finished and all of the transformation parameters are estimated, a transformation is applied to the image. This transformation can be applied using either trilinear or sinc or the nearest neighbour interpolation method (the default in MCFLIRT is trilinear interpolation).

6.1.3 Methods

In this work, two different aspects of MCFLIRT performance are evaluated:

- Accuracy of the estimated motion parameters,
- Accuracy of the application of the final transformation.

Methods which were used to do so are described in this section. Firstly, simulated data used for the evaluation purposes is described. Following from there, various MCFLIRT options and test measures are described for both of the two different aspects of MCFLIRT performance.

Simulated data

Data was simulated in such a way as to incorporate various levels of complexity at different stages. More specifically, the five different groups of simulated data were:

Group1: Level1 motion, no B_0 inhomogeneities, without noise. This group of simulations will be affected by the position displacement and also by the spin history artefact (which is unavoidable when using realistic motion sequences). Motion parameters (rotations and translations) were extracted from a patient study done by Hill et al. [34] and are shown in Figure 6.2.

The step-wise look of the motion parameters is due to the fact that the motion is defined, for this simulation, to happen instantaneously and between the

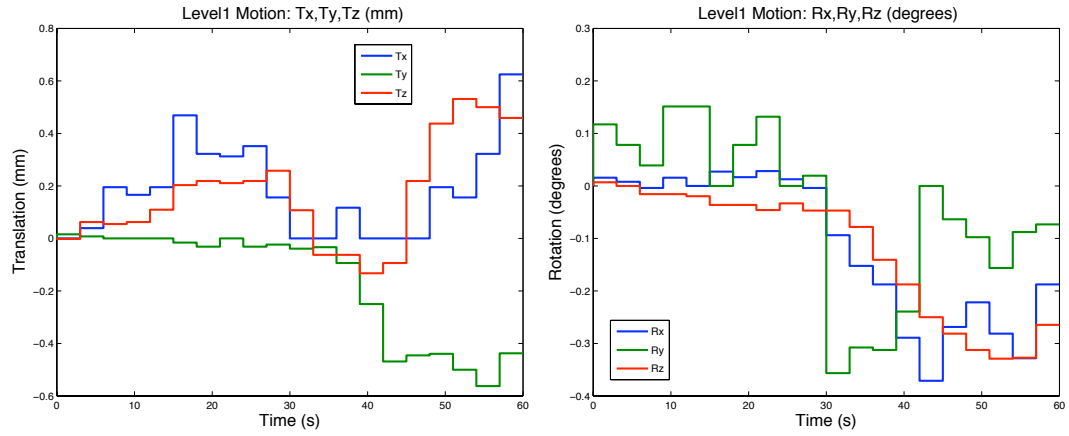


Figure 6.2: Level1 motion used in the simulations for Group1 and Group2.

acquisition of neighbouring volumes. Simulations were firstly done for motion paradigms where only one parameter is changing and all others are zero. The waveform of the motion parameter that is changing was taken from Figure 6.2. In addition to this, simulations were done when all of the motion parameters are changing together (following the pattern represented in 6.2).

Group2: Level1 motion, with B_0 inhomogeneities, without noise. This group of simulations will, in addition to all of the artefacts of Group1, also add B_0 inhomogeneities. Motion parameters and the simulation method is the same as the one described for the previous group.

Group3: Level2 motion, with B_0 inhomogeneities, without noise. This group will, in addition to all of the artefacts of Group2, also be affected by the slice misalignment. The motion parameters are shown in Figure 6.3. The difference between this and the motion file used for Group1 and Group2 is that this motion is also happening instantaneously, but between the acquisition of the consecutive slices. The motion parameters for each of the slices were interpolated

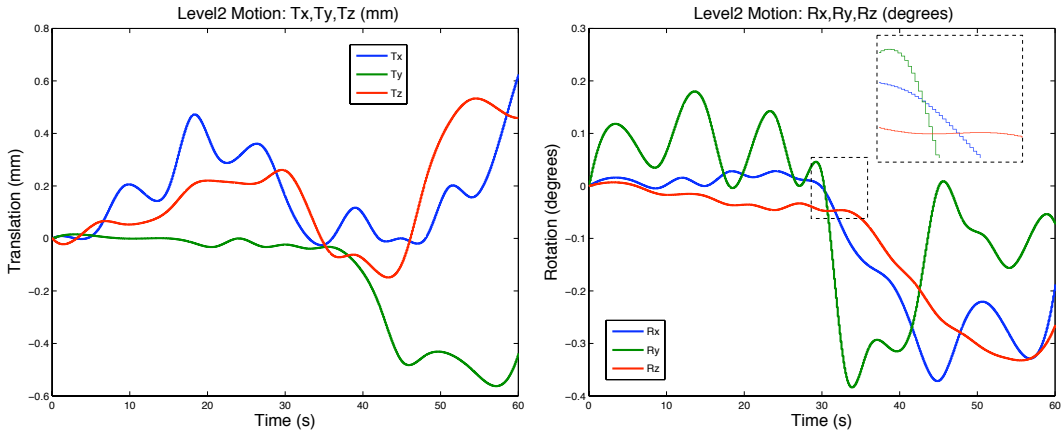


Figure 6.3: Level2 motion used in the simulations for Group3.

using piecewise cubic spline interpolation from motion parameters used in the Group1.

Group4: Level3 motion, with B_0 inhomogeneities, without noise. This group will, in addition to all of the artefacts of Group3, also be affected by blurring in the images. The motion parameters used for this group are shown in Figure 6.4. They represent continuous motion. The motion parameters for each of the time points were interpolated using piecewise cubic spline interpolation from motion parameters used in Group1. Simulations were done for small, medium and large motion. Medium motion is the one represented in Figure 6.4 (with the maximal motion of 0.8mm for the translations and 0.8 degrees for the rotations), small motion is the medium motion divided by 5 (maximal motion of 0.16mm for the translations and 0.16 degrees for the rotations), and the large motion is the medium motion multiplied by 3 (maximal motion of 2.4mm for the translations and 2.4 degrees for the rotations).

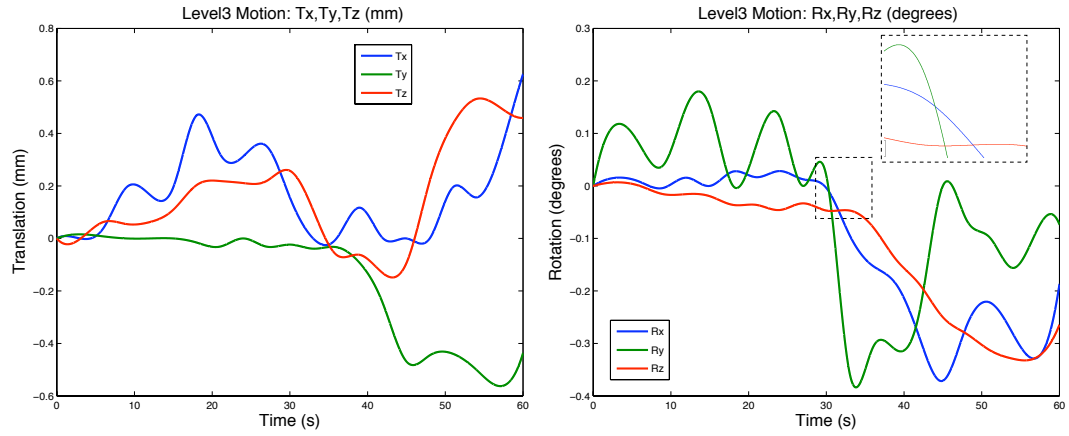


Figure 6.4: Level3 motion used in the simulations for Group4 and Group5.

Group5: Level3 motion, with B_0 inhomogeneities, with noise. This group will, in addition to all of the artefacts of Group4, be affected by the noise in the images. The only difference in the simulations from the previous group is that the various noise levels are added to the image. Standard Signal to Noise (SNR) values for single shot EPI images range from 100 to 200 [13]. Simulations were done for SNR=100, SNR=150 and SNR=200.

By separating the groups of simulations like this we hope that we can measure the influence that each of the motion effects has on MCFLIRT performance.

All of the simulations were performed using the McGill virtual brain with the parameter values given in Table 4.1, at 3T magnetic field strength, and an EPI pulse sequence which we generated with parameters: TE = 30ms, TR = 3s, flip angle 90°, maximum gradient strength 35mT/m, rise times 0.2ms, BW = $100KHz$ crushers of maximum magnitude, output voxel size $4\times4\times7$mm, 25 slices and 20 volumes. In the cases when the B_0 inhomogeneity field was simulated, a set of 9 basis volumes are passed into the simulator. Slice acquisition order ran in the -z direction.

Once the data sets were generated, they were motion-corrected using various

options of MCLFIRT. As mentioned at the beginning of this section, two different aspects of motion correction are looked at: Accuracy of the estimated motion parameters; and Accuracy of the application of the final transformation. These are, together with test measures used to assess the accuracy of the results, described in the two following sections.

Accuracy of the estimated motion parameters

The accuracy of the transformation estimation was tested for a specific choice of cost functions, interpolation functions and other options as described in the following.

Two different cost functions are investigated: normalised correlation (the current default option) and correlation ratio. These two were chosen as they were found to be the two most successful in the accuracy assessment done by Bannister [6] (with a slight preference towards the normalised correlation).

Regarding the number of stages in the optimisation process, the current default one (three stage process using trilinear interpolation) was compared to the four stage process with an extra optimisation step using the sinc interpolation (as described in the previous section). The tolerance levels chosen for the golden sec-

	Stages	Ref Image	Tolerance	Cost function
TrilF	3	First	default	Norm corr
TrilM	3	Middle	default	Norm corr
TrilMTol	3	Middle	reduced	Norm corr
TrilMCr	3	Middle	default	Corr ratio
SincF	4	First	default	Norm corr
SincM	4	Middle	default	Norm corr
SincMTol	4	Middle	reduced	Norm corr
SincMCr	4	Middle	default	Corr ratio

Table 6.1: Eight different combinations of MCFLIRT options that were tested in the experiments for testing the accuracy of the estimated parameters.

tion search were also looked at. The results generated by the current default one (0.057 degrees for the rotations and 0.002mm for the translations) were compared to results obtained using the new reduced one (0.0057 degrees for the rotations and 0.0002mm for the translations).

In addition to this, a comparison was done for two different choices of the reference image. The choice of the middle image in the time-series (the current default option) is compared to the choice of the first image in the time-series.

In summary, results were generated for eight different choices of motion correction options which are shown in Table 6.1. In order to assess the accuracy of these options two different test measures are used.

Test measure for Group1 and Group2: For the first two groups of the simulated data sets motion was instantaneous and happened *between* the acquisition of neighbouring volumes. From this it follows that the volumes themselves are only affected by the affine transformations and therefore can be completely corrected by applying the MCFLIRT algorithm. As the motion parameters for the simulations are known, it is possible to compare them with those estimated by the MCFLIRT algorithm. In order to calculate the difference between these two affine transformations (the true one, and the one estimated by MCFLIRT) a root-mean-square (RMS) average metric is employed [35].

The RMS average metric is calculated on the transformed coordinates of the volume. More specifically, when comparing two transformations, T_1 and T_2, each mapping volume A to volume B (the reference volume), if the transformations are identical then the points in volume B will be the same. However, in general the transformations differ and so map x_A to two different points: x_{B1} and x_{B2}. The vector difference between these points, $\Delta x = x_{B2} - x_{B1}$, represents the error in the transformation. It is the average magnitude of this error that is of interest.

This method of calculating the error is more suited than the other commonly used method of calculating the error directly on the transformation parameters. The reason for this is that when calculating RMS on rotation parameters directly we get the difference in degrees which is the same everywhere no matter how big the brain is. When calculating the difference in transformed coordinates over the whole brain we take that into account (for example for some brains 2 degrees in the centre is 4mm on the edge, and for some it is 3mm or 5mm etc.). In addition, for measuring the average error over the brain volume a spherical volume is chosen. This is done because it is the simplest and the closest approximation of the brain volume. Using cubic volume generates larger errors as the rotational error increases with increseing the distance from the centre of the cube (specially affected are the corners as they are the furthest).

In general, the error vector can be written as $\Delta x = Mx$ and then can be used to give the squared error:

$$\begin{aligned} E^2 &= |\Delta x|^2 && (6.1)\\ &= (\Delta x)^\top (\Delta x) && (6.2)\\ &= x^\top M^\top M x. && (6.3) \end{aligned}$$

The normalised RMS error is then given by:

$$E^2_{RMS} = \frac{\int_V E^2\, dx}{\int_V dx}. \tag{6.4}$$

Changing the variables into the spherical coordinates, and integrating over the whole spherical volume the RMS error becomes:

$$E_{RMS} = \sqrt{\frac{1}{5}R^2 \mathrm{Trace}(\mathbf{A}^\top \mathbf{A}) + (\mathbf{t} + \mathbf{A}\mathbf{x}_c)^\top (\mathbf{t} + \mathbf{A}\mathbf{x}_c)} \tag{6.5}$$

where $\mathbf{x}_c$ is the center of the volume of interest, R is the radius of the spherical

volume encompassing the brain,

$$\mathbf{M} = \mathbf{T}_2\mathbf{T}_1^{-1} - I = \begin{bmatrix} \mathbf{A} & \mathbf{t} \\ 0\ 0\ 0 & 0 \end{bmatrix} \tag{6.6}$$

and $\mathbf{T}_1$ and $\mathbf{T}_2$ are the transformations that are being compared i.e. MCFLIRT estimated transformation and the true transformation used to create the simulations. This measure gives an estimate in real units (mm) of the difference in transformations. More details on solving this equation are described in [35]. Once the RMS difference for each of the volumes is known, a mean value and a standard deviation for all of the volumes is calculated.

Test measure for Group3, Group4 and Group5: Simulations in Group3, Group4 and Group5 need to be treated differently to the simulations in Group1 and Group1. The object is moving *during* the acquisition of the volumes, creating effects in 3D images which cannot be fully compensated for by a pure affine transformation. This is because the slices will not be aligned anymore and no single 3D transformation can fix that. MCFLIRT only handles affine transformations on 3D images in its correction process, ignoring the other effects which are due to, say, slice-misalignment or blurring. Therefore validating MCFLIRT with respect to this kind of motion is unfair as MCFLIRT does not claim to be doing these corrections anyway. What is, however, possible to do is to evaluate its performance in these kind of realistic situations.

RMS difference is calculated in a very similar fashion as described above for the previous test measure. The first important difference here is that there is no one affine transformation that can correct the image (this is due to the slice-misalignment). Therefore, the true motion parameters are used to construct a separate transformation for each of the slices in the volume, and RMS difference is calculated for each of these slices individually. The second important difference

is that the RMS is calculated numerically and not analytically like in the previous case. One MCFLIRT and one true transformation were applied on each voxel of slice and the distance between the transformed voxels (in mm) was calculated. This distance is then squared, and the sum over all of the squared distances calculated. The mean of this sum is found and its squared root taken:

$$E_{RMS} = \sqrt{\frac{1}{N}\sum_{\mathbf{x}\in S} \|\mathbf{T}_1\mathbf{x} - \mathbf{T}_2\mathbf{x}\|^2} \tag{6.7}$$

where S stands for a slice, N is the number of voxels in the slice and $\mathbf{T}_1$ and $\mathbf{T}_2$ are the transformations that are being compared. Once the RMS difference for each of the slices is known, a mean value and a standard deviation for all of the slices over all of the volumes is calculated.

Accuracy of the application of the final transformation

The accuracy of application of the transformation matrix was tested for different interpolation techniques: trilinear (the current default in MCFLIRT), nearest neighbour and sinc interpolation. Results were generated for three different choices of motion correction options which are shown in Table 6.2. In order to

	Stages	Ref Image	Tolerance	Cost function	Final interp
NearN	4	First	default	Norm corr	Nearest neighbour
Tril	4	First	default	Norm corr	Trilinear
Sinc	4	First	default	Norm corr	Sinc

Table 6.2: Three different combinations of MCFLIRT options that were tested in the experiments for testing the accuracy of the application of the transformation.

assess the accuracy of MCFLIRT the following test measure is used:

Test measure: This is a measure applied directly on the corrected images and not on the parameter estimates as it was the case with the measures used when

accuracy of the estimated motion parameters was tested. A simple difference between the motion corrected image I_1 and the motionless image I_2 was calculated and divided by the averaged motionless image as shown in the equation:

$$E = 100\frac{(I_1 - I_2)}{I_2^{av}} \tag{6.8}$$

where 100 transforms the result into percentages. In order to get further insight into the characteristics of this error, the minimum, 10th percentile, mean, median, 90th percentile and the maximum value of the error over all of the volumes are summarised.

6.1.4 Results and discussion

Accuracy of the estimated motion parameters

Figures 6.5, 6.6 and 6.7 show the results of motion correction for Group1 (Level1 motion, no B_0 inhomogeneities) and Group2 (Level1 motion, with B_0 inhomogeneities) of simulated data. On the left hand side of the figures, the calculated RMS difference between the true and MCFLIRT estimated transformations is shown. Green lines represent the results for the case with no B_0 inhomogeneities (Group1), while black lines represent the case with B_0 inhomogeneities (Group2). On the right hand side of the figures, the value of the estimated motion parameter is shown. The true value which was used in the simulations is in blue, while each of the other lines represents motion correction done with a combination of options described in the legend.

It can be observed from figures 6.5, 6.6 and 6.7 that the overall accuracy of the MCFLIRT estimation of the motion parameters in the case of single motion parameter change for Group1 and Group2 (the motion happens instantaneously and between the acquisition of the neighbouring volumes) is good. The mean error does not exceed 0.08mm (for translations), and 0.3mm (for rotations).

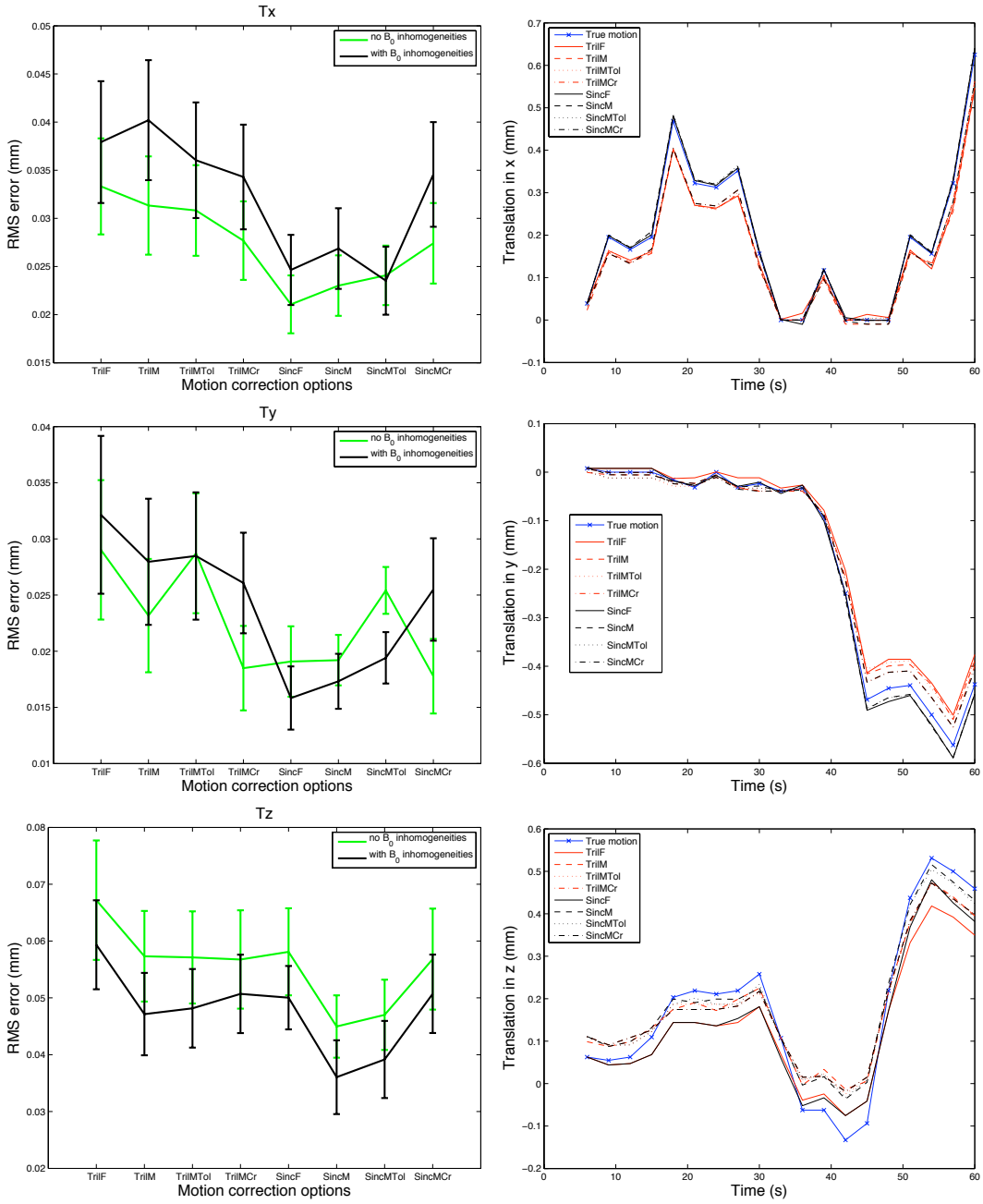


Figure 6.5: RMS mean error results for the MCFLIRT motion correction algorithm when applied to data simulated with only one motion parameter changing: Translation in x (top row); Translation in y (middle row); Translation in z (bottom row). The data belongs to Group1 (green lines) and Group2 (black lines). The left hand side plots show the error bars which indicate the standard error. The right hand side shows the MCFLIRT estimation of the motion parameter that is changing. The true values of that motion parameter (the ones used in the simulations) are shown with a blue line in each of the plots.

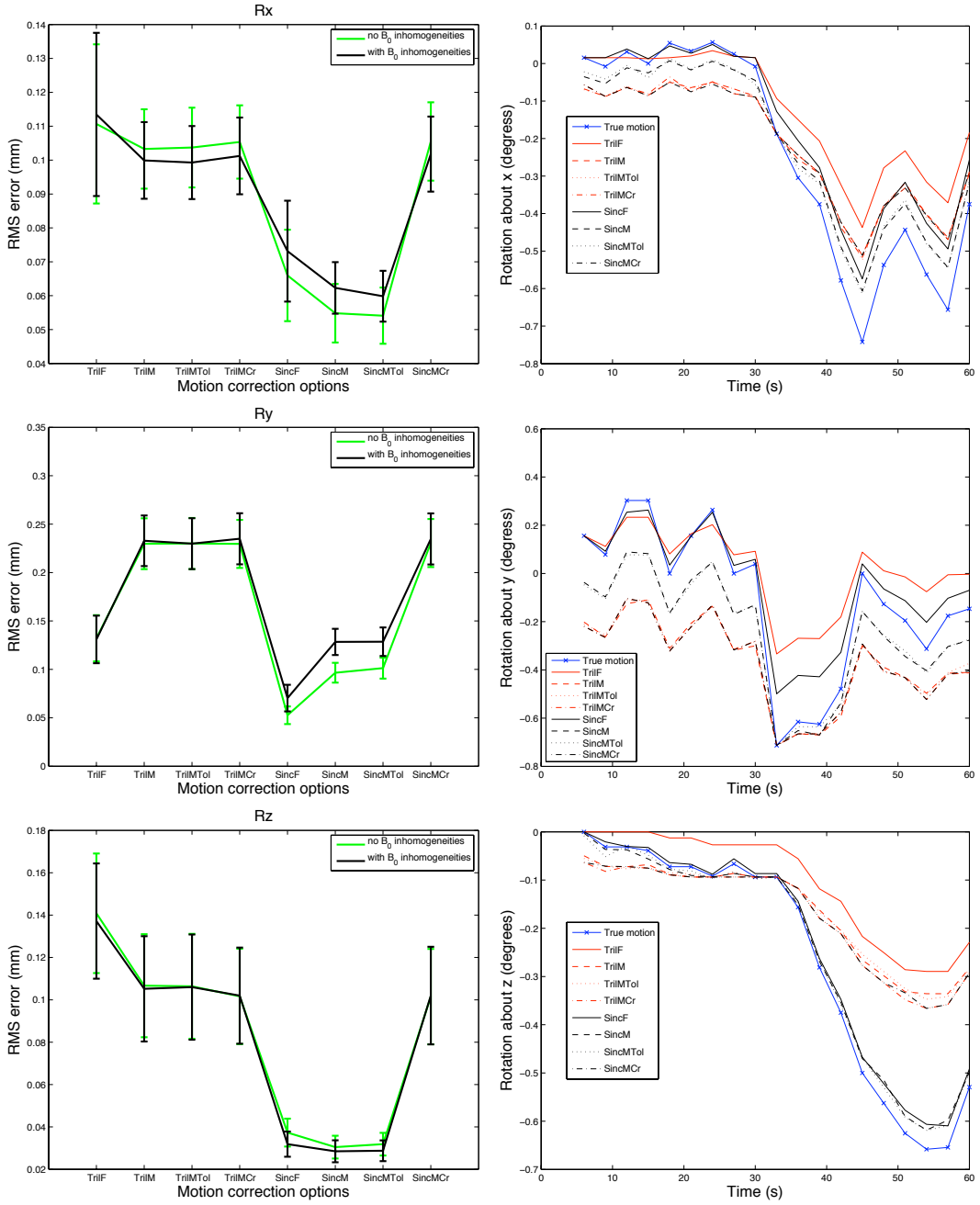


Figure 6.6: RMS mean error results for the MCFLIRT motion correction algorithm when applied to data simulated with only one motion parameter changing: Rotation about x (top row); Rotation about y (middle row); Rotation about z (bottom row). The data belongs to Group1 (full lines) and Group2 (dashed lines). The left hand side plots show the error bars which indicate the standard error. The right hand side shows the MCFLIRT estimation of the motion parameter that is changing. The true values of that motion parameter (the ones used in the simulations) are shown with a blue line in each of the plots. Note that the scale for the error plots is larger than the scale used in the translation error plots shown in Figure 6.5.

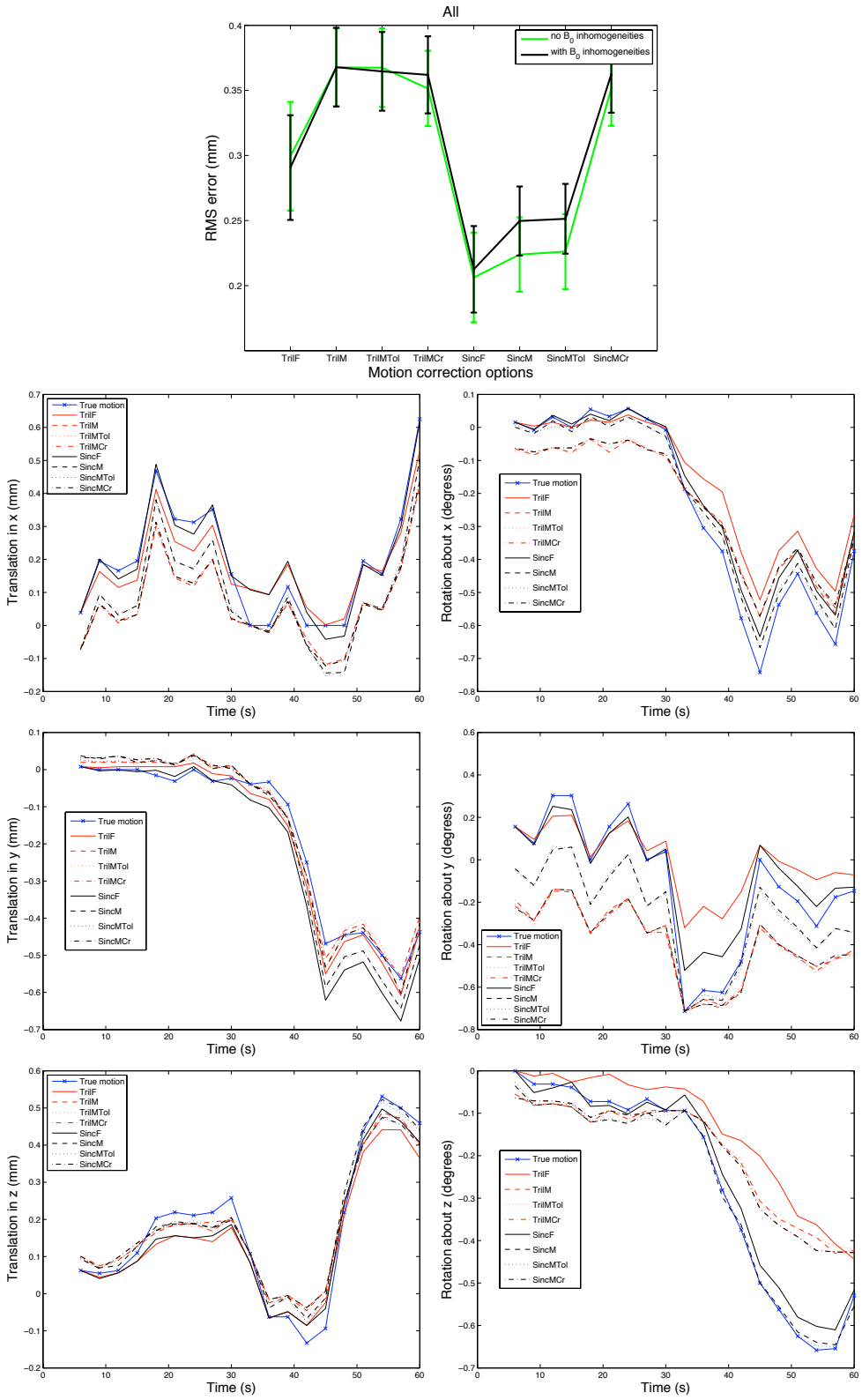


Figure 6.7: RMS mean error results for the MCFLIRT motion correction algorithm when applied to data simulated with *all* of the motion parameters changing. The top plot shows the error bars which indicate the standard error for Group1 (full lines) and Group2 (dashed lines). Other plots show MCFLIRT estimation of the motion parameters. The true values (the ones used in the simulations) are shown with a blue line in each of the plots.

It can also be seen from Figure 6.7 that each individual estimated motion parameter from the simulated data when all of the motion parameters are changing, was matched very well with the estimated motion parameters from the data when only one motion parameter is changing (shown in figures 6.5, 6.6).

A few differences can be seen between different motion correction options. Firstly, regarding the choice of the cost function, it was shown that the motion correction using the corelation ratio cost function did not perform as well as the one using the normalised correlation. This confirms the results of validation done by Bannister [6], and further justifies the use of the normalised correlation cost function in the MCFLIRT algorithm.

Regarding the number of stages in the optimisation process, the current default one (three stage process using trilinear interpolation) was compared to the four stage process with an extra optimisation step using the sinc interpolation. The results suggest considerable improvement when using four stages in the motion correction process. This improvement was stable across all of the different parameters, including the situation when the motion was quite complex, involving a change in all of the parameters.

The results regarding the tolerance levels chosen for the golden section search, suggests that the current default tolerance level (0.057 degrees for the rotations and 0.002mm for the translations) performed as well as the reduced tolerance level (0.0057 degrees for the rotations and 0.0002mm for the translations). This suggests that the current default tolerance level in the MCFLIRT is sufficient for the parameter estimation.

In addition to this, results regarding the different choices of the reference image (first image versus the middle image in the time series) are also shown. The choice of a reference image did not seem to impact results significantly or in any predictable way. It is possible that the shape of the motion paradigm influences

the preferable reference image. It would be interesting to further analyse the relationship between the two.

In addition to the conclusions drawn about the various motion correction options, a significant observation was made regarding the B_0 inhomogeneities. It can be seen in all of the figures (Figures 6.5, 6.6 and 6.7) that the presence of the magnetic-susceptibility-induced B_0 inhomogeneities (black lines) did not make a significant difference in the estimation of the motion parameters. This is quite an interesting and new observation. The error did increase in most of the cases, but this increase was not as big as what might be intuitively expected.

The results for the simulated data of Group4 and Group5 are shown in Figure 6.8, on the left and right hand side respectively. For the left plot, RMS error was calculated for the simulations of different motion intensities: red line representing the error for the large motion, blue line representing the error for the average motion, and black line representing the error for the small motion. It can be seen from the plot that as the motion level increases the size of the error also increases as expected. An interesting observation is that the amount of this increase is (almost) linearly proportional to the increase in the motion level. More specifically, the Medium motion and the Small motion are different by a factor of 5 while their respective mean RMS errors are (on average) different by a factor of 4.19. Also, the Medium motion and the Large motion are different by a factor of 3 while their respective mean RMS errors are (on average) different by a factor of 2.45.

The right hand side plot in Figure 6.8 shows the RMS error calculated for the simulations of Group5 with different noise levels: SNR=100 (red line), SNR=150 (black line) and SNR=200 (green line). The results show that the difference in noise level did not significantly influence the accuracy of estimated parameters.

A summary of the impact of different levels of complexity of the data on the

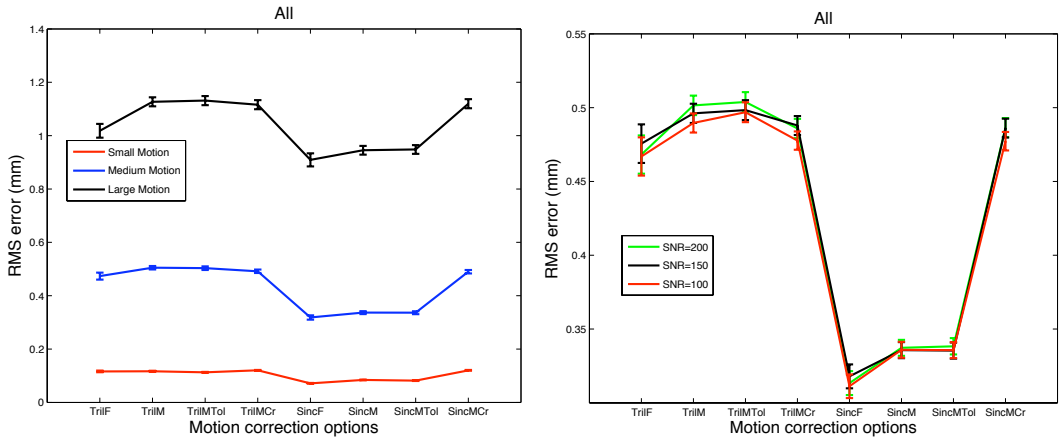


Figure 6.8: RMS mean error results for the MCFLIRT motion correction algorithm when applied to: simulated data of Group4, Large, Medium and Small motion, no noise (left); simulated data of Group5, Medium motion, SNR=100, SNR=150, SNR=200 (right). Both Group4 and 5 are simulated with the motion that varies continuously throughout the image acquisition and with B_0 inhomogeneities.

RMS error results is shown in Figure 6.9.

Different levels of motion had different impacts on the accuracy of parameter estimation. Motion that happens instantaneously between the volume acquisitions (Level1), was the easiest to correct and the error was the smallest. The presence of B_0 inhomogeneities did create a difference but this difference was not prominent.

Motion that happens instantaneously between the slice acquisitions (Level2), adds an extra motion artefact to the images: slice misalignment. From the results it is concluded that the slice misalignment created a big difference (by a factor of 1.7) in parameter estimation.

Motion that happens continuously throughout the image acquisition (Level3), adds (on the top of Level2 motion) an extra motion artefact to the images: blurring. From the results however it is concluded that blurring does not considerably change the accuracy of parameter estimation. The accuracy does not change much either when noise is added to the images (SNR=100).

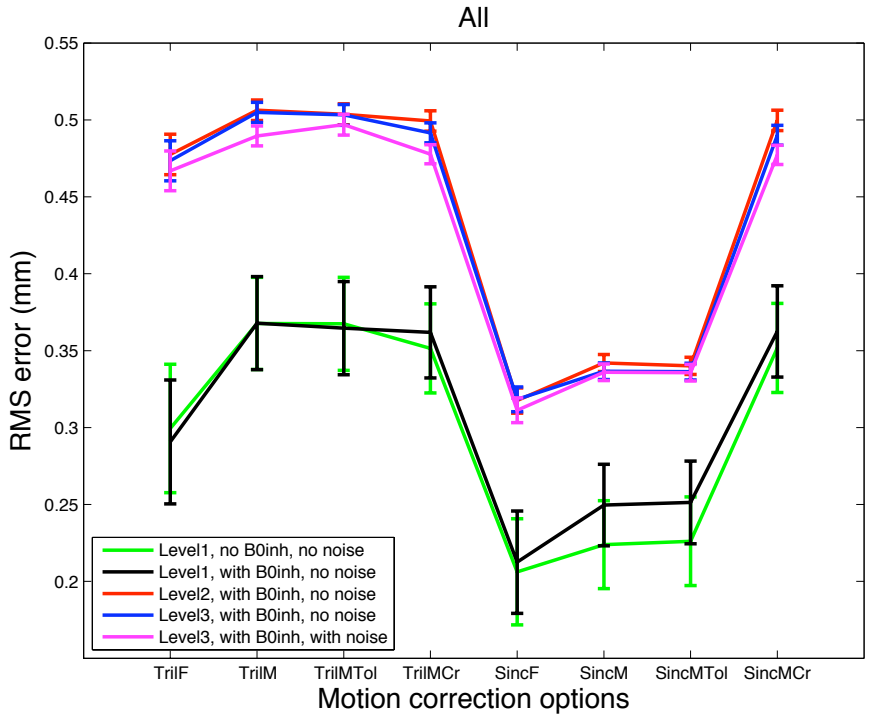


Figure 6.9: RMS mean error results for the MCFLIRT motion correction algorithm when applied to simulated data of Group1 (green line), Group2 (black line), Group3 (red line), Group4 (blue line), Group5 (magenta line).

Accuracy of the application of the final transformation

Results of the experiment testing the accuracy of MCFLIRT algorithm regarding the application of the estimated transformation at the final stage of the the algorithm are represented in Table 6.3. The error was calculated for Group 1 and 2 simulations (one representative for the translation data, one for the rotation data and one for "All", when all the parameters are changing) and is represented in the first half of the table. The error calculated for the group 3, 4 and 5 simulations is shown at the bottom of the table. The error was calculated using the test measure described in the Methods section.

Results presented in Table 6.3 show that the choice of the interpolation function when resampling the image (in the final stage of the MCFLIRT algorithm)

	min	10th perc	50th perc	mean	90th perc	max
Group1, Tx						
NearN	0	0.000178	0.014675	0.192475	0.448458	18.131998
Tril	0	0.000138	0.014644	0.166135	0.429214	17.592768
Sinc	0	0.000014	0.010066	0.112660	0.306298	14.459094
Group2, Tx						
NearN	0	0.000209	0.019885	0.194751	0.444200	19.272722
Tril	0	0.000185	0.019118	0.165766	0.417982	18.684446
Sinc	0	0.000015	0.013760	0.115542	0.306869	14.970655
Group1, Rx						
NearN	0	0.188789	0.010270	0.188789	0.429271	25.985754
Tril	0	0.000231	0.010820	0.190849	0.469642	15.738885
Sinc	0	0.000036	0.007971	0.141556	0.370623	13.398987
Group2, Rx						
NearN	0	0.000236	0.012326	0.197366	0.447132	27.397284
Tril	0	0.000226	0.013160	0.187879	0.460936	24.340437
Sinc	0	0.000034	0.009653	0.145426	0.379373	20.820518
Group1, All						
NearN	0	0.000937	0.044183	0.490567	1.121255	50.736828
Tril	0	0.000960	0.046705	0.454624	1.160782	27.538780
Sinc	0	0.000151	0.033376	0.308924	0.834392	19.972572
Group2, All						
NearN	0	0.001182	0.059673	0.516598	1.190839	51.670368
Tril	0	0.001290	0.062503	0.456827	1.164064	29.679621
Sinc	0	0.000179	0.046147	0.325090	0.870691	28.005516
Group3, All						
NearN	0	0.000970	0.060904	0.507177	1.149266	48.778713
Tril	0	0.000981	0.063039	0.449183	1.119420	31.585869
Sinc	0	0.000138	0.047097	0.317496	0.828243	30.682161
Group4, All						
NearN	0	0.000970	0.060969	0.507427	1.149931	48.751251
Tril	0	0.000988	0.063085	0.449275	1.119950	31.431250
Sinc	0	0.000138	0.047163	0.317567	0.827249	30.543518
Group5, All						
NearN	0	0.074081	0.413926	0.760446	1.396072	49.125244
Tril	0	0.068711	0.380343	0.681543	1.336535	31.405405
Sinc	0	0.059400	0.386406	0.599571	1.225205	30.497026

Table 6.3: Representing the error (in percentages of the average image intensity) which occurred when the final transformation was applied to the images. Results for three different interpolation techniques and for different levels of simulated data complexity (increasing from the top to the bottom) are given.

was important. Use of sinc interpolation was the best option out of the three interpolation techniques (trilinear interpolation, nearest neighbour, and sinc). The second best option was the use of the trilinear interpolation (which is the current default option in MCFLIRT) while the use of the nearest neighbour proved to be the worst.

It can also be observed from Table 6.3 that the overall accuracy of the MCFLIRT application of the final transformation decreases as the simulations become more realistic. When only one motion parameter is changing, and the motion is happening instantaneously between the volume acquisitions (Group1, top of the table) correction of translations was done more accurately than correction of rotations, which was expected. For the case of a single parameter changing, the error levels go even as high as 19% but the minimum, mean, maximum and the 90th percentile are overall less than 0.3% (when using sinc interpolation) of the average image intensity. The situation is worse when all of the parameters are changing with the error increasing more than twice compared to the previous cases. In this case, the 90th percentile of the error is 0.83% (when using sinc interpolation) of the average image intensity. As the simulations become more realistic the error increasses. In the most realistic case of continuous motion (Level3 motion) with B_0 inhomogeneities and noise (SNR 100), 90th percentile of the error increases to 1.22%.

It can also be observed from Table 6.3 that the magnetic-susceptibility-induced B_0 inhomogeneity did not make a big impact when evaluating the accuracy of the resampling techniques in the final stage of the MCFLIRT algorithm (difference between the groups 1 and 2 in the table). This is slightly suprising and very similar to what is found in the previous section.

6.1.5 Conclusions

The MCFLIRT algorithm showed good performance when tested for various types of simulated data. This was specifically true for estimating the motion parameters. It was found that four stages of optimisation when estimating the motion parameters, with the fourth stage using sinc interpolation, performed significantly better than the three stage option which is the current default in MCFLIRT. In addition to this, it was found that the final transformation is significantly more accurate when applied using the sinc interpolation than when applied using the trilinear interpolation which is the current default option in MCFLIRT.

It was found that by increasing the complexity levels of the simulated data (with non-rigid deformations due to motion) the MCFLIRT performance became worse which is logical and expected as MCLFIRT assumes simple rigid-body transformation of the object. It was found that slice misalignment strongly impacts the parameter estimation while, B_0 inhomogeneity, blurring and noise do not.

In addition to finding more accurate choices of some of the existing MCFLIRT options, the results suggest that the main step forward in improving the MCFLIRT algorithm is including the model for correction of the distortion due to slice misalignment.

6.2 Quantifying motion artefacts with ICA

This section describes an application of the simulator in investigating the performance of ICA as an aid to motion correction. The results presented in this section were done in collaboration with Christian Beckmann and are published in Drobnjak et al. [21].

6.2.1 Introduction

FMRI data contain many motion-related artefacts which can limit its usefulness. Many approaches exist to correct for rigid-body motion effects but they struggle to do so in cases when motion has more complex impact on the data (e.g. spin history, slice mis-alignment, or interaction of motion and the B_0 field inhomogeneities). Independent Component Analysis (ICA) on FMRI data can be used to spatially and temporally identify these artefacts, and then by doing ICA denoising (or some other advanced pre-processing method) remove the artefacts from the data.

However, in order for ICA-based tools for motion correction to be used, the accuracy and usefulness of ICA for motion artefact identification and quantification needs to be tested with data which contains realistic artefacts and for which the ground truth is known. This section shows the performance of ICA on simulated data which includes rigid-body motion effects for in-plane rotations, including the interactions with B_0 inhomogeneities.

6.2.2 Methods

The data was simulated using an EPI pulse sequence with 4mm in-plane resolution (64×64 voxels), nine 3mm slices, TR=3s, TE=30ms for 98 volumes. The BrainWeb partial volume tissue estimates were used as the object model. T_2^*

time courses were derived by fitting the equation $S = S_0 \exp(-TE/T_2^*)$ to an experimentally acquired FMRI data set (from Montreal Neurological Institute, McGill University), where S is the measured signal intensity, TE is the echo time and S_0 is the baseline intensity (determined from the average value of S under baseline conditions). The BOLD changes modelled by this T_2^* change include stimulus-related activations as well as those of no interest ("physiological noise").

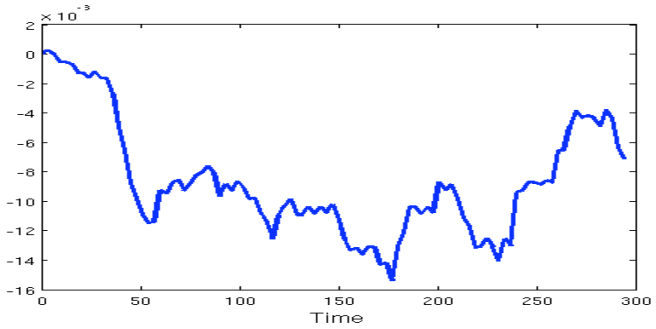


Figure 6.10: Rotation about the z-axis (degrees). Time is in seconds.

Figure 6.10 shows the change in motion parameter (rotation about the z-axis up to 0.8 degrees) over time. Four separate simulations were generated: (1) no motion; (2) motion represented in Figure 6.10 which is an average of 0.046 degrees per TR; (3) three times this motion which is 0.138 degrees per TR; and (4) five times this motion which is 0.23 degrees per TR. Rician noise (SNR=100) was added to all of the simulation outputs and a motion correction algorithm (MCFLIRT), was applied to all of them. ICA was carried out using MELODIC [7].

6.2.3 Results and discussion

There were 14 different IC components estimated by MELODIC in each of the simulations. Figure 6.11 shows two example components due to motion. Both of the components show effects of the global rotational motion as well as the motion-B_0 field interaction, which can be seen around the edges of the brain and the B_0-effected areas. The first component shows more localization in the

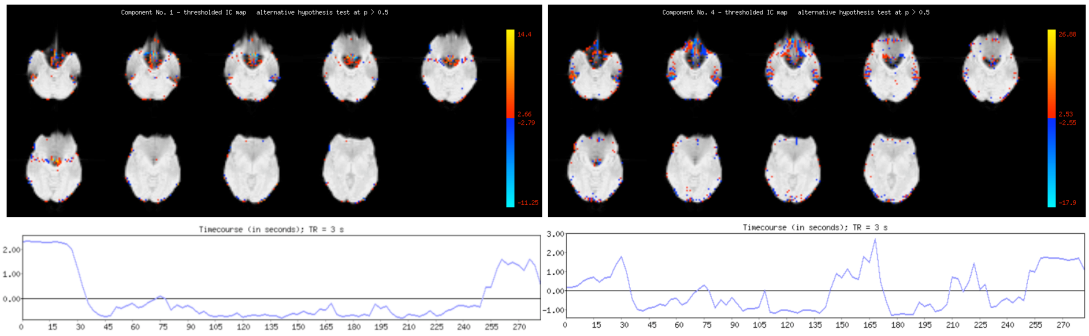


Figure 6.11: An Independent Component (IC) with predominantly motion-B_0 effect (selected due to the localisation of the effect in the B_0-affected areas, in red and blue) on the left hand side. An IC predominantly showing the effects of global rotational motion on the right hand side (selected due to the localisation of the effects around the edges, in red and blue).

B_0-affected areas and therefore is predominantly associated with the motion-B_0 interactions. Furthermore, the time courses of the two components are different, as the first closely resembles the original input motion, while the second one does not. This suggests that the influence of the motion on the data is much more complex and does not have a simple relationship with the motion parameter time-course.

In Figure 6.12 the total standard deviation explained by the ICA components in the 4 datasets can be seen. The results in this figure show that the non-motion related components had a very similar standard deviation in each case, while the level of residual motion artefact (after motion correction) increased steadily with

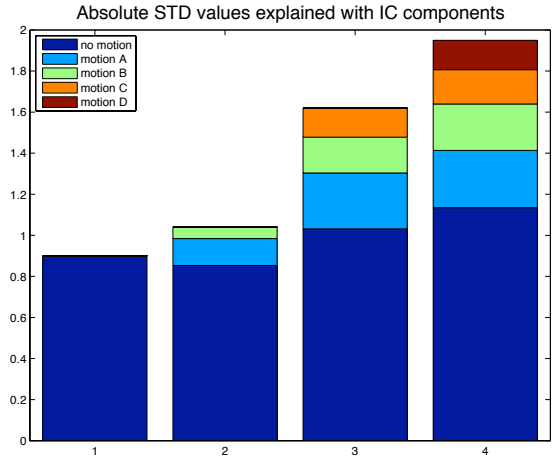


Figure 6.12: A bar-chart which shows the absolute value of the total standard deviation (std) in the simulated data. The four bars represent std explained with ICs found from the four simulations (average motion levels of: 0 deg/TR; 0.046 deg/TR; 0.138 deg/TR; and 0.23 deg/TR). Dark blue bars represent the std associated with all the non-motion-related IC components, while light blue, green, orange and brown bars represent the std of individual motion-related ICs.

IC1	IC2	IC3	IC4	IC5	IC6	IC7	IC8	IC9	IC10	IC11	IC12	IC13	IC14
0.0	0.8	0.83	0.79	0.72	0.77	0.64	0.62	0.47	0.75	0.68	0.21	0.29	0.58
0.2	0.7	0.77	0.60	0.43	0.60	0.49	0.58	0.46	0.61	0.45	0.2	0.37	0.33
0.3	0.5	0.69	0.39	0.39	0.51	0.26	0.5	0.24	0.52	0.33	0.11	0.38	0.34

Figure 6.13: The table is showing the correlation coefficients between thresholded IC spatial maps for the non-motion data set and the 3 motion-affected data sets. Only IC1 shows poor correlation since it had similar spatial and temporal characteristics as the motion components and was not effectively separated from the motion components. Note that a correlation value of 0.2 or higher has less than a 0.01 probability of occurring by chance (based on Fishers r-to-z transformation).

increasing motion levels. Similarly the number of estimated ICA components was increasing with increasing motion.

Furthermore, the non-motion-related ICs are consistently estimated in each case, as shown by the correlations between them in Table 6.13. Rows 2, 3, 4 of this table show correlation coefficients of the non-motion data with the three motion-corrupted datasets (ascending motion levels).

6.2.4 Conclusions

Degraded quality of FMR images due to motion is often observed in real experiments [31] (e.g. in patients) and is hard to correct, making FMRI less reliable than often required in clinical practice. We have shown quantitatively how ICA can be used to identify motion-related components, how the variation (std) of these components is related to amount of motion, and how the temporal characteristics of some of the motion components is not linearly related to the motion parameter changes. Furthermore, we have shown that the remaining, non-motion-related components are reliably estimated even when large amounts of motion are present. The complicated relationship between motion-artefact and motion parameters can only be explored because of the use the simulator. This complicated relationship also demonstrates that simple motion-artefact-reduction (e.g. using motion parameters as regresses) is not sufficient to correct for such artefacts.

6.3 Evaluating Stimulus-Correlated Motion Artifacts

In this section we show an application of the use of the simulator in modelling the impact of stimulus-correlated motion in FMRI data. The results represented in this section are published in Drobnjak et al. [22].

6.3.1 Introduction

Degraded quality of FMR images due to stimulus-correlated motion (SCM) is often observed in real experiments [31], [34], specifically in patients, and is hard to correct with existing motion correction methods. Rigid body motion correction algorithms, such as MCFLIRT, that were described in Section 2 cannot remove fully the SCM effects.

Most methods use rigid-body motion correction algorithms as the first step. During the statistical analysis of the FMRI data they apply a second step that uses the estimated motion parameters from the first step (or from some other method such as ICA) to remove the motion-induced fluctuation from the voxels by regressing the estimated motion parameters from the FMRI signal [26]. This method has been proved to be very good with the motion in general, but struggles to cope with the case when the motion is correlated with the experimental stimulus because when regressing the estimated motion parameter lots of the signal gets regressed as well.

In this section the simulator is applied in order to qualitatively show the effects that stimulus-correlated motion has on the FMRI data. We present simulations of two FMRI datasets, each corrupted with a different kind of head motion: motion correlated with the experimental paradigm; and motion uncorrelated with the

experimental paradigm. We show the different effects of these motion sequences on activation maps and how rigid-body motion correction algorithms that do not account for artefacts cannot recover the true activation maps.

6.3.2 Methods

In order to simulate BOLD activation (Figure 6.14), realistic T_2^* values that contain activation, resting state networks and neuronal noise were used. They were derived by fitting the equation $S = S_0 \exp(-TE/T_2^*)$ to an experimentally acquired FMRI data set (from Montreal Neurological Institute, McGill University), where S is the measured signal intensity, TE is the echo time and S_0 is the baseline intensity (determined from the average value of S under baseline conditions). The experiment that the data was generated from involved auditory naming with the paradigm consisting of six blocks 'on', and six blocks 'off'. During 'on' blocks, subjects hear a description of an object and had to think of the name of the object.

Simulations were generated with and without the presence of the B_0 field distortions (top two versus bottom two rows in both (a) and (b) of the Figure 6.14); and with (i) no motion, (ii) motion that is uncorrelated with the experiment paradigm and (iii) motion that is correlated with the experiment paradigm (described with values under each image). The two motion sequences shown in the figure were used to generate seven different motion sequences each, by scaling the rotation angles by 0, 1/50, 1/20, 1/10, 1/5, 1/2, and 1. For each of these sequences the amount of motion experienced by the object voxel that undergoes the maximum motion (expressed in percentages of the image voxel) is evaluated and is shown under each activation image.

Thermal noise was included with Contrast to Noise Ratio $CNR \sim 1$. Analysis

was carried out using FEAT (FMRI Expert Analysis Tool) Version 5.42, part of FSL (FMRIB's Software Library, www.fmrib.ox.ac.uk/fsl). The following pre-statistics processing was applied; spatial smoothing using a Gaussian kernel of FWHM 5mm; mean-based intensity normalisation of all volumes by the same factor; highpass temporal filtering. Time-series statistical analysis was carried out using FILM with local autocorrelation correction [71]. Z (Gaussianised T/F) statistic images were thresholded using clusters determined by $Z > 2.3$ and a (corrected) cluster significance threshold of $P = 0.01$ determined by the Gaussian Random Field theory [72].

Motion correction was done separately, prior to this processing, by applying the inverse of the known input rotation and translation to each image, using trilinear interpolation. Note that this corresponds to the 'ideal' case of rigid-body parameter estimation, and is not affected by the image artefacts.

6.3.3 Results and discussion

The BOLD changes modelled in these images includes changes of interest but also those of no interest ("physiological noise"). In Figure 6.14 (a), it can be seen as expected, uncorrelated motion has acted as extra random noise and, depending on the strength of motion, the activations shrank or were completely lost. On the other hand, motion that is correlated with the experimental paradigm created many false activations which can be seen in the images in Figure 6.14 (b). As the motion became larger, the maximum rotation ranging from 0.08° to 4°, the size and the number of false activations increased as well. This is particularly emphasised in the cases with the B_0 distortion in which case false activations are also seen at the edges of the areas affected by the B_0 artefact.

As the motion amplitude increases the number of erroneous activation clusters

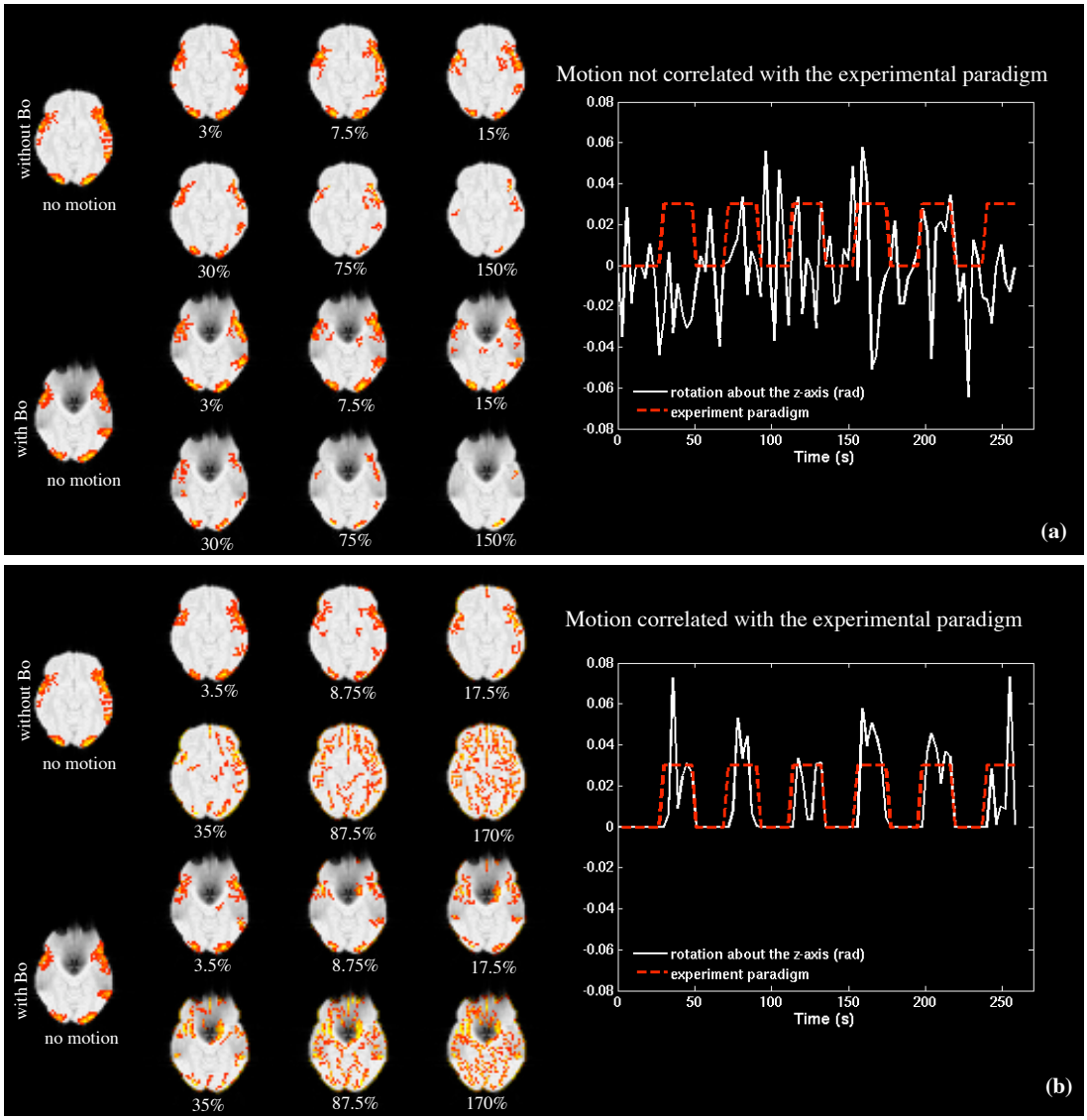


Figure 6.14: Images showing the effects of motion of different magnitudes on BOLD changes. On the right hand side of both (a) and (b) are figures describing the experimental paradigms together with the varying motion parameter (rotation about z-axis). Simulations generated with the average amount of motion (ranging between -0.7 and 0.7 degrees) experienced maximum 30% and 35% (in image voxels) movements, in the case (a) and (b) respectively. Figure (a) shows the resulting activation images when the motion is uncorrelated with the experimental paradigm in which case it acts like added noise and therefore, as expected, with increase in motion some of the activations are lost inspite of the motion correction. Figure (b) shows the resulting activation images when the motion is correlated with the experimental paradigm. As expected, some false activations are seen on the edges of the images and also on the edges of the B_0 affected areas (in the case of simulations with B_0 perturbation - bottom two rows).

increases as well. Some of the erroneous activation clusters look quite realistic and like real activations (e.g. the activation site just next to the B_0 distorted area). Such real looking activations in a real experiment, which are solely due to motion, often get confused with the real activations. In order to avoid any danger, researchers tend to model out all of the images that look suspicious. This causes a loss of valuable data, and is costly regarding the time and the resources.

A follow up application after the one described in this section would be to use the simulator in order to predict these erroneous activation sites so that they could be differentiated from the real activation sites. The simulator would use the motion parameters estimated by the use of an existing motion correction algorithm and, together with all of the subject-, scanner-, and experiment-specific properties, predict where the false activations (if any) would be in the images. If the approach proves to be successful and commonly used in practice, it could possibly save quite a lot of data that would otherwise be lost and would help in increasing the chance of the accurate and precise results in the FMRI. It would specifically be helpful in individual clinical FMRI experiments when there is only one chance to do the experiment.

6.3.4 Conclusions

These simulations demonstrated that stimulus uncorrelated motion reduces the number and the size of activations but does not induce erroneous activations. This behavior was the same with or without the presence of the B_0 magnetic field inhomogeneities. It was shown that in the case of average motion the number of the activation sites and their size was not significantly different from the true one.

On the other hand, it was concluded that stimulus-correlated motion both

reduces the number and the size of true activations and creates erroneous activations. The number, position and extent of the erroneous activations was dependent on the presence of the magnetic field inhomogeneities. It was also shown that in the case of average motion the number of the activation sites and their size was significantly different from the true one. This conclusion confirms the results that stimulus-correlated motion affects the data significantly more than the stimulus-uncorrelated motion.

Possible extensions of this application were also suggested. These would serve as a tool to predict the erroneous activations, and eliminate them from the real data. The proposed method would possibly be able to save large amounts of motion corrupted data which would otherwise be lost.

6.4 Eddy Current Effects

This work was performed in collaboration with Rita G. Nunes and Stuart Clare from FMRIB's Physics Group. It was published in [56].

A study where the eddy current effects were simulated for the standard and optimised sequences is presented. The ability to correct for the effect of eddy currents using an affine registration method is analysed.

6.4.1 Introduction

In order to be able to infer on fibre structure in the brain, a set of diffusion-weighted (DW) images need to be collected under a set of different diffusion gradient directions. The presence of the eddy current fields induced by the diffusion gradients during the readout window can lead to significant geometric distortions in echo planar images (EPI) [39]. Misalignment of the DW images will, in turn, lead to errors in the estimates of the diffusivities and fibre orientations [39].

In order to minimise the effect of eddy currents at source, modifications to the standard single spin-echo sequence, shown in Figure 6.15 (a), have been suggested. One of the modifications is the doubly-refocused sequence, developed by Reese et al.[64], and is shown in Figure 6.15 (b). When gradients are switched on and off, they induce eddy current fields of opposite polarity. In the case when the time constant of the eddy current fields is much longer than the duration of the gradients, the opposite eddy current fields will cancel out. By applying diffusion gradients with a shorter duration than the ones in the standard sequence, the cancellation of eddy currents having lower time constants can be achieved, enabling the minimisation of the eddy currents at source [64].

Despite its advantages regarding the minimisation of eddy currents, the doubly-refocused sequence is still not often used. This is because it is generally assumed

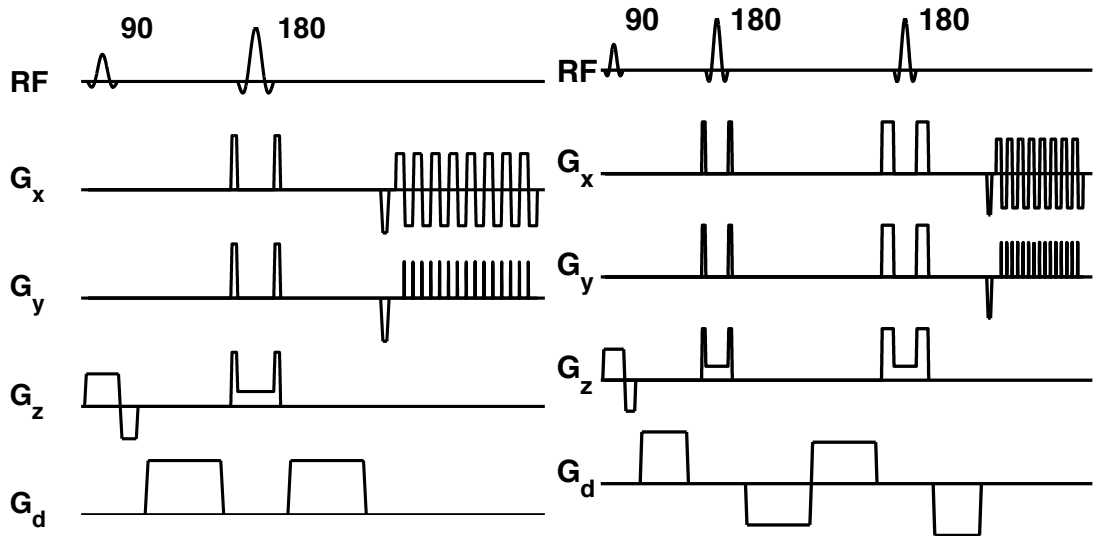


Figure 6.15: Figure courtesy of Rita Nunes. Standard, single spin-echo sequence on the left. Doubly-refocused sequence on the right. In order to minimise eddy currents, each diffusion gradient in the single spin-echo sequence is split into two.

that the geometric distortions observed with the single spin-echo DW images should be correctable using affine registration methods. This would be true provided that the eddy current effects could be considered both time invariant throughout the acquisition window and spatially invariant from slice to slice. This assumption, together with concerns regarding potential signal-to-noise loss due to the presence of an extra imperfect 180° pulse have prevented the wider implementation of the doubly-refocused sequence. The standard single spin-echo sequence still remains, therefore, the most commonly used.

While the presence of higher order eddy current terms has been experimentally demonstrated [66], the intrinsically low signal-to-noise of DW images, together with the presence of subject motion complicates their experimental quantification. In order to be able to isolate the aspects related only to eddy current fields and to remove all other confounds, the simulator was employed. The simulator has made it possible to quantitatively evaluate the extent to which affine registration

can correct for eddy current distortions.

6.4.2 Methods

Gradient-echo EPI images were generated. The matrix size was 128×128 corresponding to an in-plane resolution of 2×2 mm^2. A slice thickness of 3 mm was chosen. The bandwidth was 100 kHz, the readout window 163 ms and the echo time 90 ms. The effect of the eddy currents produced by each of the two sequences (single spin-echo and doubly-refocused) was simulated by superimposing a sum of exponentially decaying terms to the EPI readout gradients:

$$G_{eddies}(t) = \sum_i \pm\varepsilon\, G_{diff}\, \exp[-(t - t_i)/\tau] \tag{6.9}$$

each of these terms corresponds to the switching either on (-) or off (+) of each of the positive or negative diffusion gradients at time t_i. Two diffusion gradients were considered for the standard sequence each with a duration of 28 ms. For the doubly-refocused sequence four diffusion gradients with a duration of 14 ms were considered. The amplitude of the diffusion gradients G_{diff} was 21.3 mT/m while the maximum amplitudes achieved by the read and phase-encode EPI gradients were 9.1 mT/m and 4.0 mT/m respectively.

A set of eddy currents were simulated with different relative amplitudes ε and time constant τ. Time constants of 5, 10, 30, 50, 70, 90 and 100 ms were considered. For the read out direction, eddy currents with amplitudes of 0.01, 0.04, 0.07, 0.1% relative to those of the diffusion gradients were used. These amplitudes were rescaled for the phase-encode direction accounting for the lower strength of the phase-encode blips in comparison to the read out gradients.

The diffusion gradients could be applied either along the read (xx) axis, producing skews in the images or along the phase-encode direction (yy) producing scaling [39].

Although the difference in contrast displayed by both the T_2-weighted image and the DW images should also have an influence in the efficiency of the registration step, no attempt was made to model anisotropic diffusion. To provide the distorted images with a realistic contrast, the mean attenuation produced in each tissue was instead considered. The k-space data corresponding to CSF, white and grey matter were separately generated and an appropriate attenuation factor applied to each of them before summation. These were based on a b-value of 1000 smm^{-2} and a simple exponential decay was assumed: $S = S_0 \exp[-b < D >]$. The mean diffusivities $< D >$ for each tissue were: 3.2×10^{-3} mm^2/s (CSF), 0.8×10^{-3} mm^2/s (grey matter), 0.7×10^{-3} mm^2/s (white matter) [47].

The distorted attenuated images were then registered to the undistorted T_2-weighted image using an affine registration method [36]. Five degrees of freedom were allowed for: translation along xx and yy, rotations around the zz axis, skews in xy and scaling along yy. If eddy current distortions could be fully corrected by performing affine registration, no difference should exist between the registered images and the undistorted attenuated image. To evaluate the ability to correct for eddy-current-induced distortions, the sum of squares of the difference image between each registered image and the undistorted DW image was therefore calculated.

6.4.3 Results and discussion

The undistorted T_2-weighted (a) and DW (b) simulated images are shown in Figure 6.16. Images generated for the single spin-echo sequence with eddy currents along the read (c) and phase encode direction (d) are also displayed.

The distortions are clearly visible when looking at the difference images (e)

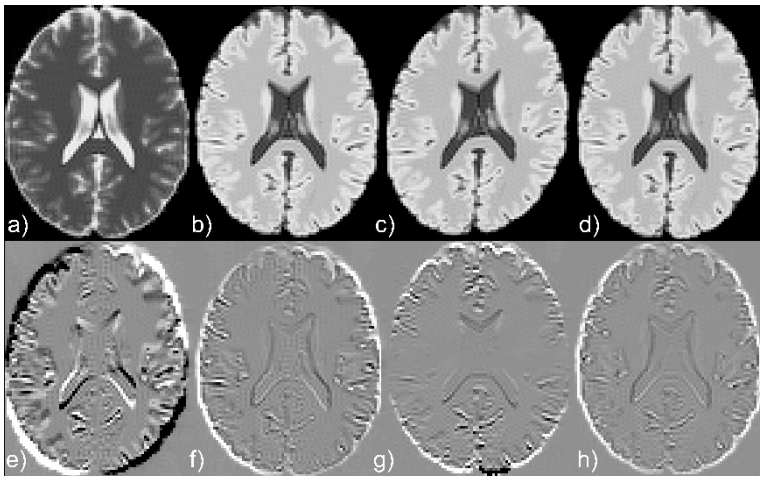


Figure 6.16: Simulated images displaying eddy current effects. (a) T_2-weighted image; (b) DW image; (c) DW image with an eddy current along the read direction (standard sequence, ε= 0.01%, τ= 10 ms); (d) DW image with an eddy current along the phase-encode direction (standard sequence, ε= 0.01%, τ= 10 ms); (e) Subtraction of image (b) from (c); (f) Subtraction of the image obtained after registering (c) to (a); (g) Subtraction of image (b) from (d); (h) Subtraction of the image obtained after registering (d) to (a).

and (g), where (b) was subtracted from each of the distorted images. Although registration did enable a reduction in these differences, residual distortions are still visible in both (f) and (h).

From the residuals plotted in Figure 6.17, it can be noted that the affine registration method does not seem to be able to cope effectively with eddy currents of large amplitudes and time constants much lower than the duration of the gradients. Unless a more effective correction method is used, minimising eddy currents at source should therefore be preferable. The simulation results also confirm previous experimental observations [63] as the residuals corresponding to the doubly-refocused sequence are significantly lower. There can be several causes for the residuals observed. One factor could be the presence of higher order eddy currents, not corrected using affine registration. On the other hand, the presence of either skews or scaling, even if affine, in the distorted images would

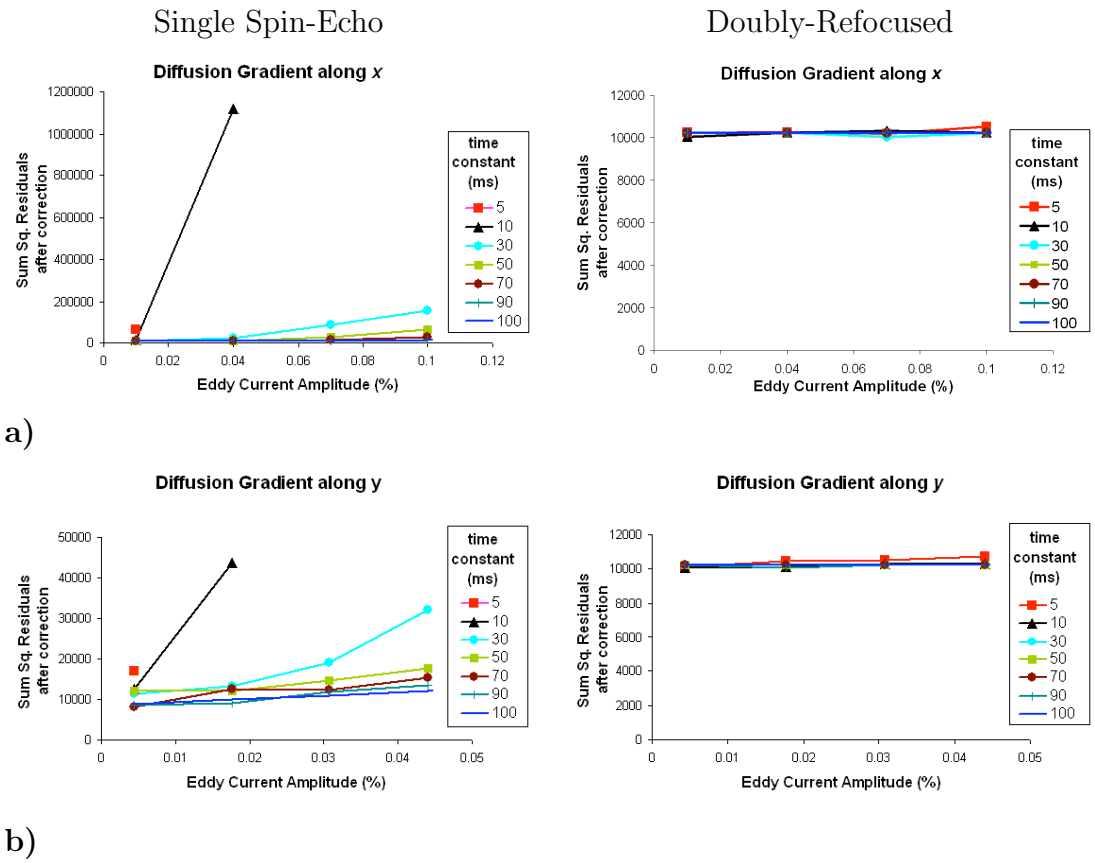


Figure 6.17: Sum of squares of the residuals for both sequences after affine registration, for eddy currents along the read (a) and the phase encode (b) directions. The sum of square of the residuals was obtained when comparing each registered image with the undistorted DW image. Each curve corresponds to a different time constant in ms as indicated in the legend. Note that the vertical scales are quite different in the left hand and the right hand figures.

distribute the overall signal over a wider number of pixels. This signal cannot be totally recovered by performing registration even if the images become perfectly re-aligned and should lead to errors when estimating diffusivities and/or fibre orientations. Another factor that can influence the quality of the registration is the presence of eddy-current-induced ghosting.

As stated in Nunes [56] "In order to be able to improve these simulations, in the future the gradient waveforms used could be altered so as to resemble more closely those used when acquiring diffusion images on the actual scanner. An-

other aspect which was not taken into account was that DW images are normally acquired using a spin-echo sequence. At the moment POSSUM does not simulate the effect of 180° pulses, but this feature is expected to be added in the future. A further step of the simulation which could be improved, is the way in which the effect of the eddy currents is added to the gradient waveforms. Instead of simply adding a sum of exponential decays, the actual gradient waveforms could be filtered so as to remove the high-frequency components. In this way the ramping up and down of the gradients could be made to be smoother as observed in practice.

Under these conditions the scanner simulator employed here could be extremely useful, as different aspects of the problem could be studied in isolation. In a first stage, this simulator could be used to consider the problem of correcting for high levels of distortion in the images without the additional difficulty of also having a low image SNR. Increased levels of noise could then be added so as to provide better tests for the correction methods developed. The problem could then be made to be even more complex, so as to better replicate the experimental conditions, by progressively introducing the effects of subject motion and field inhomogeneities."

6.4.4 Conclusions

With the use of the simulator it was demonstrated that using a standard affine registration method is not sufficient to correct for strong eddy current effects. By using the doubly-refocused sequence, the level of eddy currents is substantially reduced compared to the more widely used single spin-echo approach. The same results were achieved experimentally by Nunes [56]. Furthermore, it was also demonstrated that for the doubly-refocused sequence the registration step per-

forms equally well over a wide range of amplitudes and time constants. Possible additions to the simulator together with the novel areas of future work in which this simulator could be useful in this application have also been suggested.

6.5 Direct MR imaging of neuronal activity

The work in this section is done in collaboration with Dr Gaby Pell from the Brain Research Institute, Melbourne, Australia. Dr Pell is investigating the characteristics of an MRI technique that could potentially be used for direct detection of neuronal activity [61, 60]. The use of the simulator is likely to be of considerable use in this area. The main focus of the work presented in this section is to show the ways that the simulator could assist work in the area of direct imaging. In order to do that, two of the experiments carried out by Pell et al. and presented in their paper on "Optimisation of MR sensitivity to transient and weak currents in a conductor" [60] have been simulated. The experimental results obtained by Pell et al. are compared with the simulation.

6.5.1 Introduction

As described in Chapter 2, blood oxygenation level dependent (BOLD) FMRI has become one of the most widely used methods for the non-invasive imaging of neuronal activity in the human brain. However, this method does not measure the neuronal activity directly from the neurons themselves. It measures the haemodynamic changes that occur in the blood vessels that surround the neurons. The changes that occur in the blood vessels are coupled with the neuronal activity, but the exact nature of the coupling is complex and still not fully understood [49, 3]. The accepted view is that the haemodynamic changes are relatively sluggish and variable and therefore an imprecise measure of neuronal activity [5].

In addition, due to its dependence on the vasculature, the BOLD response is much slower than the actual neuronal response thus limiting the temporal resolution to seconds rather than milliseconds which are needed in order to image the transient activity of the neurons. Coarse temporal resolution also results

in the inability of FMRI to temporally resolve cascaded communication between sequentially activated brain regions. Furthermore, the spatial extent of the BOLD response is significantly larger than the presumed localised area of active neurons responsible for the change in the blood flow [60]. In the imaging voxels with large vessel effects this can result not only in the enlargement of the activation area but also in its slight shift (up to a centimeter) from the true region of activation [5].

Due to the above mentioned limitations of BOLD FMRI techniques, a need for new, more sophisticated methods to image brain activity has emerged. In the past few years, there has been significant research done in the field of direct neuronal imaging [12, 11, 73, 42, 43, 5, 60] which investigate new MRI techniques in directly detecting the localised magnetic effects arising from neuronal currents. These techniques rely on transient electrical activity perturbing the local magnetic field during a portion of the acquisition sequence and altering the phase and magnitude of the MR signal. The development of these techniques could potentially provide a neuronal mapping method with a direct access to the neuronal activity and with an excellent spatial (millimeters) and temporal (milliseconds) resolution.

In order for the neuronal current imaging to be used in practice, more research needs to be done regarding the sensitivity of the MR signal to the neuronal currents. In order to address this issue, Pell et al. investigate the sensitivity of MR magnitude signal change to: 1) the location of the transient magnetic field within the imaging voxel; 2) the current amplitude; 3) the timing of the current pulse within the imaging sequence; and 4) imaging parameters [60].

By using the simulator it is possible to analyse the sensitivity of MR magnitude signal change to any of the above mentioned factors. Physical experiments are time- and material- consuming and very often it is quite difficult to eliminate or minimise the artefacts that can appear during scanning. By using the simula-

tor as an aid when running the experiments it is possible to avoid those issues. This section demonstrates how the simulator was used to analyse the sensitivity of MR magnitude signal change to: 1) the 0.1mA current amplitude and the phase encode direction; and 2) the timing of current pulse within the imaging sequence.

Simulations are performed in such a way as to closely follow the experimental set up described in Pell et al. [60]. This was done in order to be able to compare whether the two independently acquired set of results (simulated results and the scanner acquired results) match well. In order to present the work fully, the relevant elements of the experimental set-up used for acquiring the scanner results by Pell et al. [60] are described throughout the section.

6.5.2 Methods

The phantom for the scanner-based experiments was constructed from a hollow glass sphere, filled with SF96/50 silicone oil. A carbon fibre conductor (diameter 0.015mm) was placed inside the glass sphere and a Perspex support structure was constructed to hold it in its place [60]. Passing current through the conductor is equivalent to the activity in a neuron (or a bundle of neurons aligned in parallel and oriented in the same way). The T_1 and T_2 relaxation times of the oil are 1420ms and 530ms respectively. These were measured by Pell et al.[60].

The phantom (object) used for the simulations was a simple sphere which is presented in the top row of Figure 6.19. The sphere had a very thin conductor going through its centre. This sphere represents an area around one of the two legs of the conductor in the phantom used in the experiments of Pell et al.. When the current goes through a conductor, a magnetic field is created around the conductor, perturbing the magnetic field of the scanner. The newly created

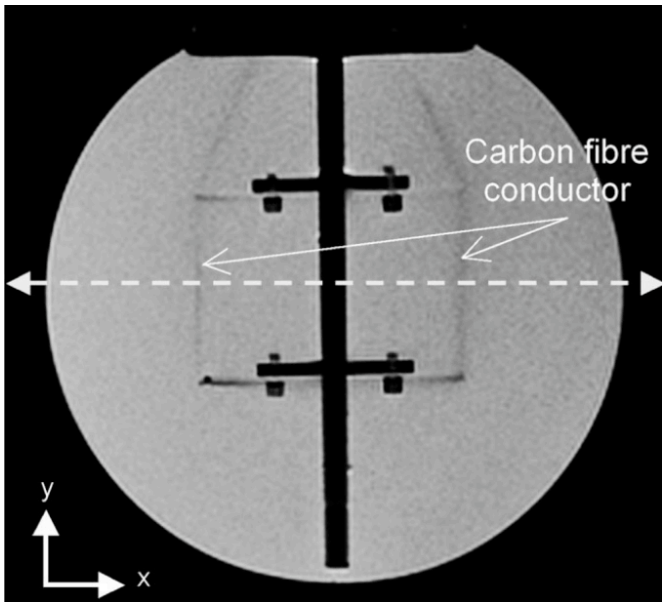


Figure 6.18: Figure courtesy of Pell et al. Spin-echo axial MR image through the phantom taken to illustrate its structural features. Figure shows the silicon oil-filled container. The Perspex mounting framework it contains can be seen as a dark shape. The loop of the carbon fibre conductor is indicated by the arrows and is faintly visible. The white dashed line indicates the coronal imaging plane used for subsequent experiments, cutting through the wire in two positions. The FOV is 20 cm.

changes in the magnetic field are calculated using the Biot-Savart law. The perturbed field is presented in the bottom row of Figure 6.19. T_1 and T_2 values used in the simulator were the same as the ones estimated by Pell et al. for the silicone oil.

The experiments by Pell et al. were performed on a 3T system with gradient slew rate of 150T/m/s and a maximum gradient strength 40mT/m. These values were incorporated into the pulse sequence generator of the simulator. Pell et al. used a hardware- system to generate current in the conductor with accurate and precise control of the amplitude and duration. An EPI sequence was used in the experiments (Pell et al. also used other sequences in their paper) and it had the following parameters: 128×128 matrix, 6mm slice thickness, 90° excitation flip

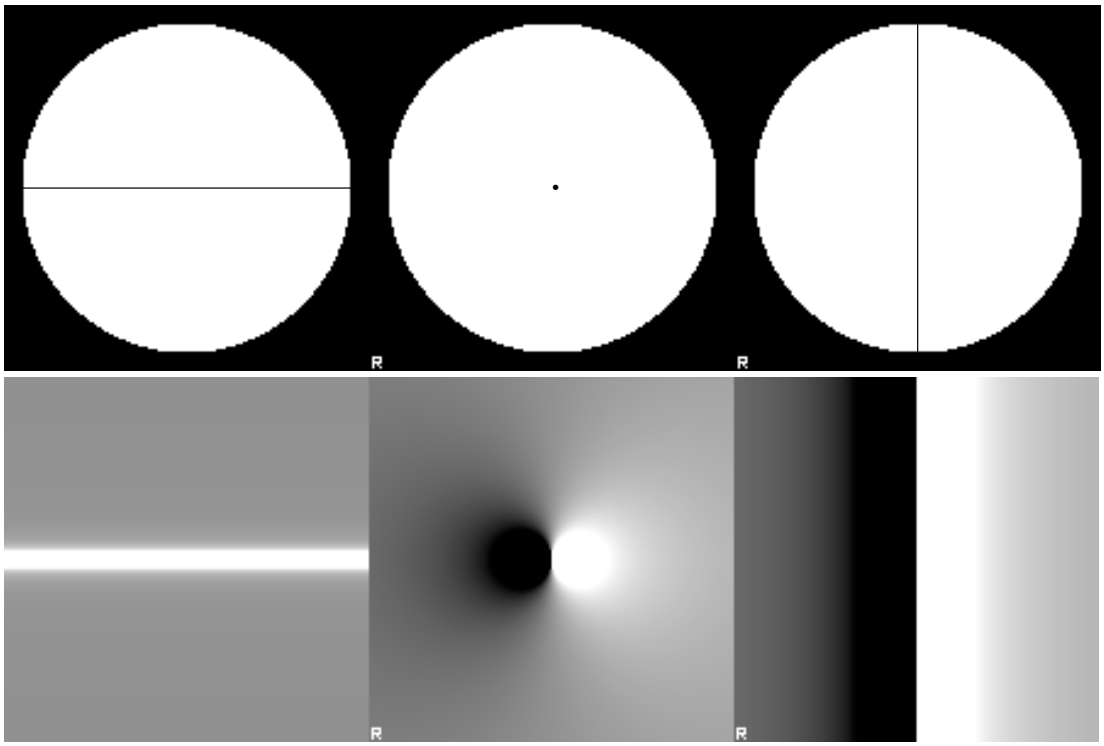


Figure 6.19: Mid-coronal, -sagittal and -axial slices of the input object for the simulation (a sphere) are shown in the top row of the figure. A thin wire goes through the centre of the sphere. The bottom row shows the B_0 inhomogeneities that occur when the current is on as calculated using the Biot-Savart law.

angle, sampling time of 8μs, imaging voxel size of $1.56 \times 1.56 \times 6$mm, $TE = 85$ms, $TR = 1$s, slice select gradient is G_y (coronal slices).

Sensitivity to the current and the phase encode direction

This experiment was designed to determine the change in the signal magnitude and the spatial characteristics of this change when a current pulse of amplitude 0.1mA is applied. The current pulse duration was 20ms, and it was placed immediately after the 90° RF pulse. In addition to the image when the current pulse is on, an image when the current pulse is off was also generated. The image representing the signal change is obtained by subtraction of these two images.

Sensitivity to the timing of the current pulse within the imaging sequence

This experiment was designed to investigate the change in the signal magnitude when a current pulse is "walked through" an EPI pulse sequence. The current pulse amplitude was 7.8mA, its duration was 5ms, and its position within the pulse sequence was changed in each of the 260 volumes that were acquired in the following fashion: during the acquisition of the first volume, the current was appllied immediately after the 90° RF pulse; during the acquisition of the second volume no current pulse was applied; during the acquisition of the third volume, the current pulse was placed 1ms later than that during the acquisition of the first volume; during the acquisition of the fourth volume, again, no current pulse was applied; during the acquisition of the fifth volume, the current pulse was placed 1ms later than that during the acquisition of the third volume; and so on. The timing of the onset of the current pulse was increased in 1ms increments for every alternate volume acquisition. A time-series is then extracted from the region of maximal signal change within the image. The experiment is discussed in more detail in Pell et al. [60].

6.5.3 Results and discussion

Sensitivity to the current and the phase encode direction

Figure 6.20 shows the subtraction of the image simulated with the current pulse on, and the image simulated with the current pulse off. The subtraction represents the signal change that occurred due to the current pulse. The image is shown with extreme colour contrast, mainly to point out the overall areas of positive (white) and negative (black) signal change. These show the "splitting" of the signal along the x-axis (in the case of phase encode in the x-direction) and along

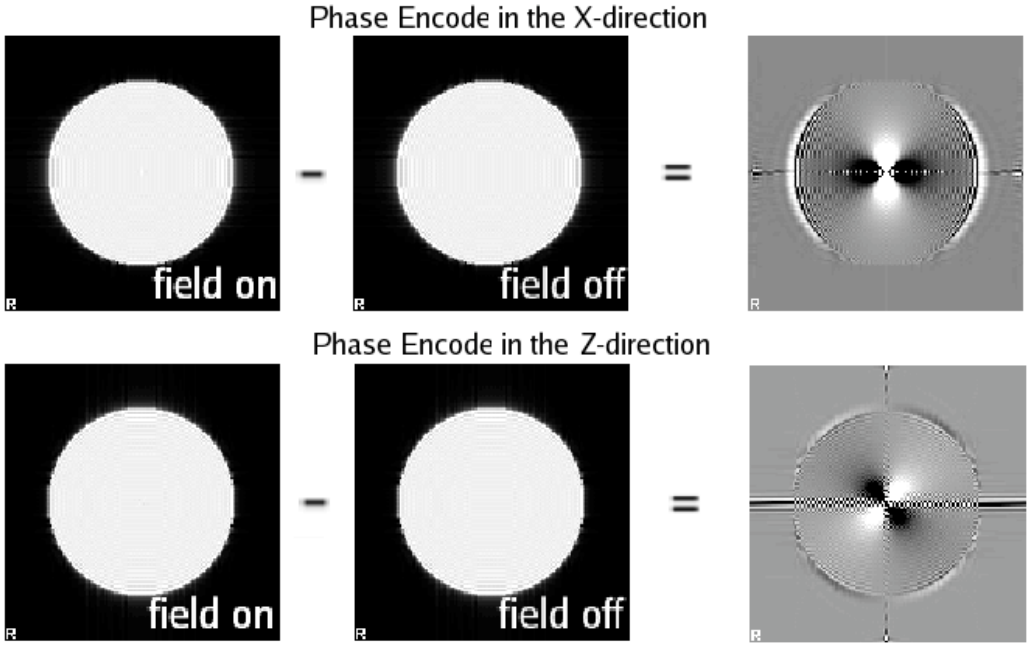


Figure 6.20: Image on the right (of both top and bottom row) represents the signal change when the current pulse is on. It was obtained by subtraction of two images: the one on the left is simulated with the current pulse on, and the one in the middle is simulated with the current pulse off. The top row represents the images simulated with the phase encode in the x-direction. The bottom row represents the images simulated with the phase encode in the z-direction.

the x-z diagonal (in the case of phase encode in the z-direction).

This kind of signal splitting has been noticed also by Pell et al. [60] and the comparison of their result to the result obtained by the simulator is shown in Figure 6.21. Note that the results of Pell et al. are statistical results showing all of the statistically significant voxels, which is not the case with the results of the simulator which were generated free of any noise or other artefacts.

In order to get more insight into the magnitude of the signal change and its spatial distribution, percentages of the change were calculated for each voxel individually (for one coronal slice) and shown in Figure 6.22. Results are shown for the case when the phase encode is in the x-direction. The voxel at the position

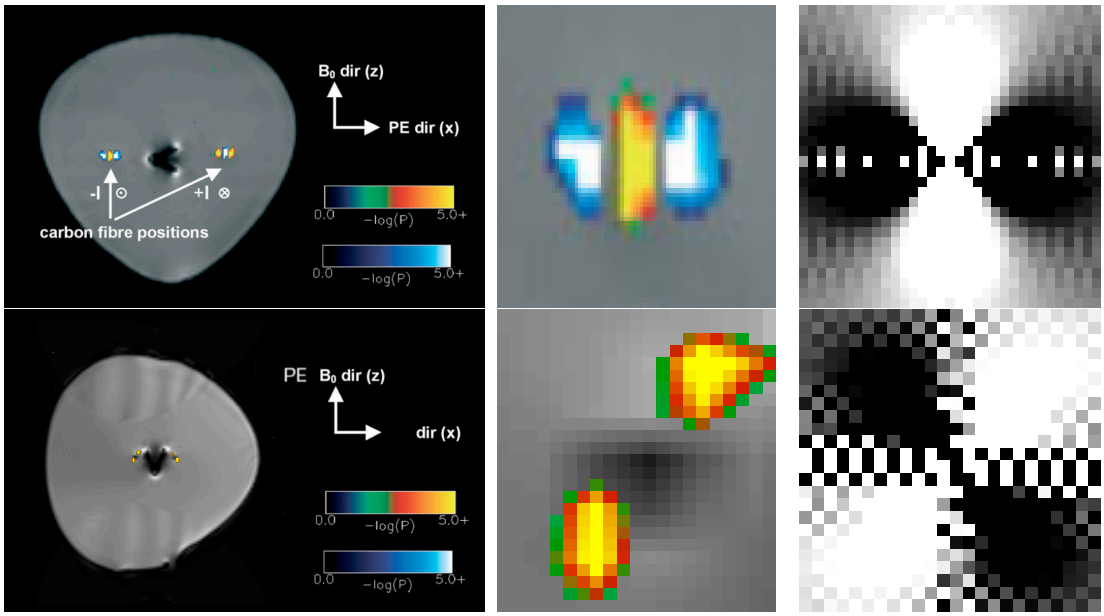


Figure 6.21: A comparison between Pell et al. results (the left hand side) and the simulator results (the right hand side) are shown. The middle column is a 24×24voxels patch around the left carbon fibre position extracted from the images on the left hand side. Phase encode in the x-direction (top row): Signal is seen to "split" along the phase encoding direction (x). Phase encode in the z-direction (bottom row): Signal is seen to "split" along the x-z diagonal.

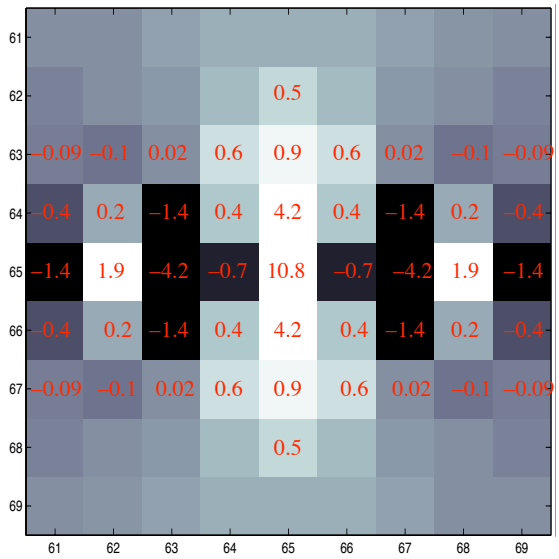


Figure 6.22: Signal change image in a 9 by 9 patch surrounding the central image voxel (the one containing the conductor) for phase encode in the x-direction. Red values represent intensity change (as percentages) for each of the image voxels independently.

(65,65) is the voxel through which contain the wire with the current goes. This voxel experienced the maximum of the signal change (10.8%). The signal change rapidly decreases as we are moving away from the wire with maximum signal change of 4.9% in the next layer and 1.9% change two layers of voxels away from the central voxel. This is equivalent to 1.56mm and 3.12mm respectivelly (as the in-plane voxel size is 1.56×1.56mm). As expected, the decay happens in a symetrical way which is due to the symmetry in the magnetic field inhomogeneities induced by the wire.

Sensitivity to the timing of the current pulse within the imaging sequence

Figure 6.23 shows the results of the "walk-through" experiment which was described in more detail in the Methods section. Sensitivity of the signal to the timing of the current pulse within the imaging sequence is demonstrated. The figure shows the signal in percentages: 100% is the value when no change occurred, 80% is the value when a 20% signal change occurred etc. The large signal change between any two consecutive points represents the interleaving of the on and off states of the current pulse. In the figure, signal values (bottom row) and the pulse sequence events for the acquisition of one slice (top row) are aligned in such way as to represent that each signal value in the bottom row corresponds to a position in the top row which is where the current pulse was centered. For example, the 120th volume is acquired during the 120th second of the total imaging time (as each TR is 1s) and during its acquisition the onset of the current pulse was at 60ms in the pulse sequence.

The results obtained by the simulator showed very similar behaviour to the experimental results. The signal rapidly increases after the application of the RF pulse. It then maintains a steady level until the pulse enters the read-out

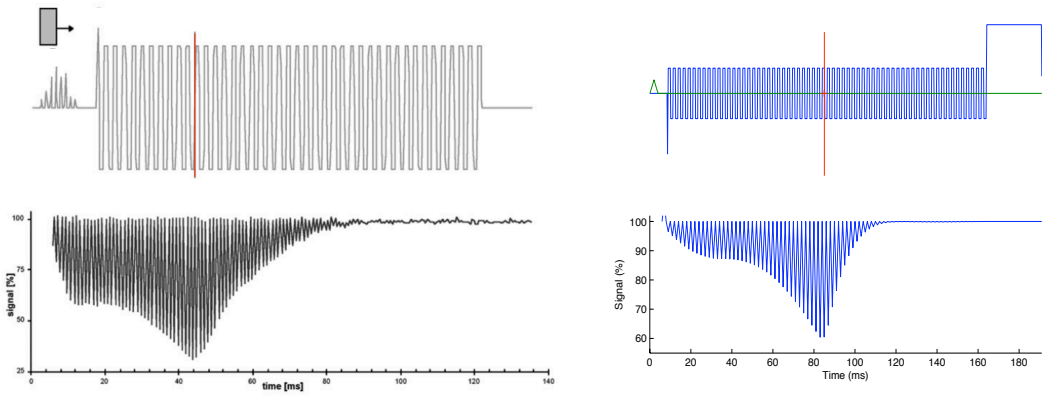


Figure 6.23: Results of the "walk-through" experiment. On the left hand side is the experimental result acquired experimentally by Pell et al..(Figure courtesy of Pell et al.). On the right hand side of the figure is the simulator result. The top row shows the time-progression of the pulse sequence (one for each of the experiments). The red line indicates the center of the k-space. The bottom row shows the normalised signal (in %). Signal values at 100% correspond to volumes acquired when no current pulse was on. The rest of the signal values correspond to the volumes when the current pulse was on. The position of the value corresponds to the time the current pulse was on within the pulse sequence (top row). The key observation in both of the left and right hand plots is that the contrast is maximal when the current pulse is timed to overlap the readout of the center of k-space.

gradient. The signal change subsequently rises in a linear fashion. The largest signal change occurs when the current pulse coincides with the acquisition of the central line of the k-space. The signal change subsequently decreases and reaches very small levels before the end of the read-out window.

The magnitude of the signal change is, however, different in the two results. This can be possibly attributed to the fact that the Pell et al. result was generated using a partial k-space reconstruction whereas the simulator results were acquired with the full k-space. It was observed by Pell et al. [60] that the results for the full k-space showed lesser change in signal magnitude.

6.5.4 Conclusions

Two experiments done by Pell et al. [60] (investigating the sensitivity of the signal to the current pulse of 0.1mA amplitude for different phase encode directions, and sensitivity of the signal to the timing of the current pulse) were replicated by the use of the simulator. The results obtained were a very close match with the results obtained by Pell et al. The work on applying the simulator in direct neuronal imaging is still on-going, but the preliminary experiments that were described in this section are very encouraging.

The application of the simulator can be very useful as an aid in doing experiments in direct current imaging. One of the main reason is that it is easier, less costly and potentially more accurate (as some of the scanner artefacts can be avoided) to do simulations than real experiments which involve physical phantoms or human subjects. The simulator can be particularly useful when investigating the effect that more than one neuronal current would have on the signal (say randomly oriented in the object). This is one of the potential future experiments and directions for the use of the simulator in this field.

6.6 Summary

In this chapter five different applications of the simulator were investigated.

Firstly, the simulator was applied in testing a motion correction algorithm - MCFLIRT. It was shown that the overall performance of MCFLIRT algorithm was good. Different default options to the ones that are currently used in the software are suggested.

Secondly, the simulator was applied to investigate the performance of ICA as a tool for quantifying motion-related artefact. It was shown that ICA could make reasonable estimation of motion-related independent components, however more

investigation is needed in order to make more general conclusions.

Thirdly, the simulator was applied in evaluating stimulus-correlated motion artefacts in FMRI data. The results demonstrated that stimulus correlated motion affects the data significantly more than stimulus uncorrelated motion. The results also demonstrated that the areas of distortion due to the B_0 inhomogeneities were particularly badly influenced in SCM-affected images.

The fourth application of the simulator presented in this thesis was in simulating eddy currents artefacts. With the use of the simulator it was demonstrated that using a standard affine registration method is not sufficient to correct for strong eddy current effects. It was also shown that by using the doubly-refocused pulse sequence, the level of eddy currents is substantially reduced compared to the more widely used single spin-echo approach.

The fifth application of the simulator presented in this thesis is in direct neuronal current imaging. The influence of a neuronal current on the image generation was simulated. Results were a very close match with the results obtained experimentally by Pell et.al.[60].

Chapter 7

Conclusions and future directions

The first and primary aim of this thesis was to develop a general computer simulation program for FMRI with the following properties:

- The ability to produce realistic anatomical simulated MR images of any brain (or any other object) for a variety of different scanning conditions;
- The ability to produce realistic time-dependent sequences of simulated MR images including the effects of physiological activations due to some external stimulus;
- The ability to model realistic image related artefacts (mainly due to rigid-body motion of the object and magnetic field inhomogeneities).

The development of the simulator was done in three distinct stages: the model development, software implementation and the evaluation of this software, each of which was described in chapters 3-5 respectively.

The second aim of this thesis was to use the newly developed simulator and apply it in investigation of some of the motion-related issues in FMRI as well as in some interesting applications in image acquisitions.

The conclusions regarding the simulator and its use, together with suggestions for future research, are discussed below.

7.1 Development of the simulator

The development of a computer program that can simulate generation of functional MRI images is a very complex task. The first reason for the complexity of the task is the difficulty of *capturing all of the realistic scanner effects into one model*. There are many factors that influence the generation of MR images: scanner environment, subject's brain structure and MR characteristics, subject's behavior which can change from experiment to experiment. Chapter 3 presents how a range of these factors have been incorporated into a single model. The model of the simulator is based on Bloch equations which are adapted and solved in order to generate MR images and also to capture some of the most prominent features in MRI and FMRI data such as: magnetic field inhomogeneities, rigid-body motion of the head, BOLD signal, eddy currents and noise.

The second reason for the complexity of the simulator development task is the *computational demand* of simulators of this kind. In order for the simulations to be realistic it is crucial to have as many of the factors influencing the FMRI data included in the simulations as possible. Incorporating these factors creates a huge demand on the computational memory and time. This problem has prevented many of the simulators developed in the past from simulating realistic 3D images and also modelling some of the very common but more complex artefacts (e.g. spin history). Chapter 4 describes the method used in this research to implement the previously described model (in Chapter 3) in a computationally efficient way. It is shown how various computational speed ups are used (including the use of semi-analytical results and parallelisation techniques) in order to make the simulations memory- and time-efficient. With the described way of implementing the model, it is possible to do simulations of a 2D image within minutes, simulations of a 3D image within an hour and long FMRI sequences within a day. The sim-

ulator now is several orders of magnitude faster than the version that was not computationally optimised. This is one of the biggest achievements of the work described in this thesis.

The third reason for the complexity of the simulator development task is the difficulty in *evaluating software* of this kind. In order to evaluate each of the software features it is crucial to have experimental data to compare the simulated data with. However, it is impossible to fully control the experimental data, which contains many un-planned factors that are part of the scanning process. The validation process done in this thesis is presented in Chapter 5 and is done so that both the implementation and the modelling side of the software are evaluated. Experiments were devised which were focused specifically on validating the motion model of the software.

Currently the simulations do contain some limitations. BOLD activation is modelled purely through changes in T_2^*, which neglects any other changes induced by varying Cerebral Blood Volume and Cerebral Blood Flow, such as flow-induced signal artefacts [28] or changes in vascular occupancy [50]. Also, the simulator can not fully model spin-echo sequences (including stimulated echoes) or implement general RF pulses. Thus, areas for future improvement of the current simulation would be to incorporate true spin-echo modelling and allow for more general RF pulses, including fully numerical Bloch simulation during the RF pulse.

The software presented in this thesis, named POSSUM (Physics-Oriented Simulated Scanner for Understanding MRI), is planned to be released as part of the FSL software package (FMRIB Software Library, http://www.fmrib.ox.ac.uk/fsl/) in the forthcoming year. In addition, POSSUM is part of an international collaboration (between the University of Oxford, the Montreal Neurological Institute at McGill University and the University of Hawaii) which aims to build a very comprehensive FMRI simulation environment named MIDAS (MR Imaging Data

Acquisition Simulator) [54]. More details can be found at
http://www.fmrib.ox.ac.uk/analysis/research/possum/.

7.2 Applications of the simulator

In this thesis five different applications of the simulator were investigated.

Firstly, the simulator was applied in testing a motion correction algorithm - MCFLIRT (Section 6.2). The MCFLIRT algorithm showed very good performance when tested with various types of simulated data. It was found that four stages of optimisation when estimating the motion parameters, with the fourth stage using sinc interpolation, performed significantly better than the three stage option which is the current default in MCFLIRT. In addition to this, it was found that the final transformation is significantly more accurate when applied using the sinc interpolation than when applied using the trilinear interpolation which is the current default option in MCFLIRT. This application of the simulator, together with some future tests, will result in an improved motion correction tool.

In Section 6.3 the simulator was applied used to investigate the performance of ICA as a tool for quantifying motion-related artefact. The results found that the ICA method was reliable in estimating some of the effects of motion (rotation about z-axis) for various levels of motion. These results are, however, not quite enough to make a more general conclusion about the performance of ICA. It would be interesting to take this research further and test the performance of ICA with simulated data that contains various other effects (such as B_0 inhomogeneities – motion interaction of second type i.e. when the motion involves rotation about the x or y-axis).

The third application was presented in Section 6.4 and it involved evaluation of the impact of stimulus correlated motion (SCM) artefacts in FMRI data.

The results demonstrated that stimulus correlated motion affects the data significantly more than stimulus uncorrelated motion. The results also demonstrated that the areas of distortion due to the B_0 inhomogeneities were particularly badly influenced in SCM-affected images. In these areas many of the erroneous activation appeared. These results are a confirmation of the results so often seen in practice and are well know. However, what would possibly be an interesting application of the simulator is to be implemented as a first step in motion correction of FMRI data as a kind of "predictor" of where the erroneous activation might occur (for a given brain, field map and a motion sequence) which would then serve to distinguish between "real" and "false" activations.

The fourth application (Section 6.5) was in the field of eddy currents artefacts. With the use of the simulator it was demonstrated that using a standard affine registration method is not sufficient to correct for strong eddy current effects. It was also shown that by using the doubly-refocused pulse sequence, the level of eddy currents is substantially reduced compared to the more widely used single spin-echo approach. These results are very promising, and based on them it is possible to say that further simulation work using the simulator may help to devise better schemes to align DW images in the presence of higher order eddy current fields.

Finally, the last application done in this thesis (Section 6.6) presents the application of the simulator in direct neuronal current imaging. The results in this application were obtained by considering the influence of one neuronal current and they found that: the maximum signal change due to the neural current happens when the timing of this current overlaps the center of the k-space; the signal splitting when the phase encode direction is along x-axis (coronal slices) is seen to be along x-axis; the signal splitting when the phase encode direction is along z-axis (coronal slices) is seen to be along the x-z diagonal. These results were

a very close match with the results obtained experimentally by Pell et.al.[60]. These are promising results, lending support to the belief that further simulation work using this software may help when investigating complex neuronal current models (e.g. the effect when a number of randomly oriented neuronal currents would have on the signal when they act together). This would be very useful for the further experiments in direct neuronal current imaging, as building phantoms is a highly difficult and costly task which could be helped by the use of simulations as an extra tool.

7.3 Future development of the simulator

The research presented in the previous chapters leaves many promising avenues open for future research. This section looks at some of the future directions in the development of the simulator.

Spin-echo modelling An obvious limitation of the simulator is that it can not fully model spin-echo sequences (including stimulated echoes). This limitation exists because, de-phasing of the spins is averaged over each voxel i.e. each voxel is characterised by only one magnetisation vector. This magnetisation vector decays exponentially following the T_2^* decay and can not exhibit "re-phasing". In order to be able to model re-phasing, the model of the voxel magnetisation needs to be more complex. For example, one way would be to model each voxel as a collection of isochromats. Magnetisation vector for each of the isochromats satisfies Bloch equation and the total sum of these vectors would in this case be able to simulate re-phasing which is necessary to obtain the spin-echo and the T_2 decay. A downfall with this approach is that it is very computationally expensive as each of these magnetisation vectors also needs to be followed through

the pulse sequence (i.e. Bloch equation needs to be solved for at every point for all of them). The computational demands of the simulator are already quite high and with this extra work the simulations quickly become un-realistic. A way to overcome this and still be able to model the spin-echo is an open problem and an obvious useful future direction for the simulator development.

General RF pulses Another possibility for future work is to implement general RF pulses. At the moment RF pulses are implemented with a use of a slice profile and are assumed to happen instantaneously. This approach prevents investigation of the motion effects during the RF pulse. It can be improved by including fully numerical Bloch simulation during the RF pulse.

RF inhomogeneities An additional improvement to the simulator is modelling RF inhomogeneities. This is a part of an on-going work which has not yet been fully implemented and validated within the simulator. However, the model has already been developed. Inhomogeneities in the RF field are modelled separately in the simulator for the receiving RF coil and the transmitting RF coil. Inhomogeneity in the receiving RF coil impacts received signal strength and is modelled with an extra multiplier $\psi_r(\mathbf{r}_0)$ in Eq. [3.19]:

$$S(t) = \sum_{j\in\Lambda} \sum_{\mathbf{r}_0\in\Omega} \psi_r(\mathbf{r}_0) s_j(\mathbf{r}_0, t) \,. \tag{7.1}$$

Inhomogeneity in the transmitting RF coil impacts the strength of the flip angle and is modelled with an extra multiplier $\psi_t(\mathbf{r}_0)$ in the equation for the flip angle:

$$\alpha(\mathbf{r}_0, t_{0-}) = \psi_t(\mathbf{r}_0) W(\omega(\mathbf{r}_0, t_{0-}) - \omega_{RF}) \,. \tag{7.2}$$

Values of $\psi_r(\mathbf{r}_0)$ and $\psi_t(\mathbf{r}_0)$ range from zero (maximal inhomogeneity) to one (no inhomogeneity). They are specified for every object voxel through user specified files. RF inhomogeneities are currently being implemented into the simulator.

More advanced pulse sequences Theoretically, the simulator can simulate any gradient-echo sequence. However, the current implementation has been done for the standard 2D gradient-echo sequence and the standard EPI sequence. A further improvement of the simulator would be to extend the implementation of the simulator so that it can simulate other gradient-echo type sequences such as spiral EPI, partial k-space EPI, 3D EPI, FLASH and SSFP. The work would mainly involve extending the pulse sequence generator and reconstruction algorithms.

Parallel imaging An additional area of improvement is including the possibility of parallel imaging in the simulator. Besides the image contrast, imaging speed is probably the most important consideration in clinical magnetic resonance imaging (MRI). Unfortunately, current MRI scanners already operate at the limits of potential imaging speed. In recent years, the greatest progress in further increasing imaging speed has been the development of parallel MRI (pMRI).

Parallel MRI works by taking advantage of spatial sensitivity information inherent in an array of multiple receiver surface-coils to partially replace time-consuming spatial encoding, which is normally performed by switching magnetic field gradients. In this way, only a fraction of phase-encoding steps have to be acquired, directly resulting in an accelerated image acquisition while maintaining full spatial resolution and image contrast. Besides increased temporal resolution at a given spatial resolution, the time savings due to pMRI can also be used to improve the spatial resolution in a given imaging time. Furthermore, pMRI can diminish susceptibility-caused artifacts by reducing the echo train length of single- and multi-shot pulse sequences.

The way the parallel MR imaging can be done in the simulator is by including the coil model in the simulator. Each coil has their own sensitivity map which

needs to be incorporated into the existing RF model of the simulator. The simulator can then be run for each of the coils independently and the output needs to be reconstructed using algorithms developed specially for parallel MR imaging.

Respiratory and cardiovascular changes In addition to the above mentioned improvements which are mainly concerned with improving the scanning properties, it would also be useful to look at the ways of improving the physiological noise model by adding the respiratory and cardiovascular induced changes. However, the exact way of how respiratory and cardiovascular induced changes influence MR parameters is not completely known and is part of current research by a few world groups e.g. Montreal Neurological Institute, McGill University. Some suggestions exist that the main changes occur in the T_2^* and the phase of the magnetic field and this could be easily modeled in the simulator by modifying the T_2^* and B_0 variations in time.

7.4 Final remarks

In this thesis, a novel FMRI simulator is presented. The simulator, for a given gradient echo pulse sequence, a segmented object with known tissue parameters, and a motion sequence produces realistic simulated images and FMRI time series. The way the Bloch equations are solved, by tracking and updating the magnetisation vector through time for every object voxel, allows the changes that occur during the acquisition of one TR to be modelled and then carried through to the acquisition of the next TR, making it possible to simulate the BOLD response (through T_2^* changes), the spin history effects, motion during the readout periods and interactions that motion-related artefacts have with B_0 inhomogeneities.

The simulator has the capability to turn on or off various subject- and scanner-

related effects, which is not possible in real scanning and therefore has a wide range of applications. These applications include the simulation and removal of various artefacts, both in MRI and FMRI. Furthermore, by generating the "ground truth", the FMRI simulator can be used for evaluation and validation of software tools for FMRI analysis methods (e.g. motion correction, registration, statistical analysis of images, etc.). Specifically, the simulator can be used to model the full effects of rigid-body motion artefacts on FMRI data and on FMRI statistics.

Using simulation-based methods is a very cost-effective method of evaluating complex systems. However, until recently this was not possible due to the computational requirements. With the development of processing power and with the development of methods such as the one presented in this thesis, simulation-methods have the potential of becoming a very important tool in the field of neuroimaging.

Appendix A

Magnetic field calculations

When an object is placed in a homogenous magnetic induction field $\mathbf{B}_0$, a distortion of this field will occur, depending on the susceptibility distribution $\chi(\mathbf{r})$ of the object. Assuming the object is non-conductive (so $J = 0$), the following Maxwell's equations will be satisfied

$$\nabla \times \mathbf{H} = 0 \tag{A.1}$$

$$\nabla \cdot \mathbf{B} = 0 \tag{A.2}$$

where $\mathbf{H}$ is the magnetic field, $\mathbf{B} = \mu\mathbf{H}$ is the total magnetic induction and the permeability, μ, is related to the susceptibility, χ, by

$$\mu = \mu_0(1 + \chi) \tag{A.3}$$

where μ_0 is the permeability of free-space.

Due to time-invariance and the absence of stationary currents the curl of $\mathbf{H}$ vanishes and $\mathbf{H}$ can be expressed as the gradient of the magnetic scalar potential $H = \nabla\phi$. This gives

$$\mathbf{B} = \mu_0(1 + \chi)\nabla\phi \tag{A.4}$$

and hence

$$\mu_0 \, \nabla \cdot (\, (1 + \chi)\nabla\phi) = 0. \tag{A.5}$$

Let the susceptibility, χ, be expanded as

$$\chi = \chi_0 + \delta\chi_1 \tag{A.6}$$

where χ_0 is the susceptibility of air (i.e. $\mu_{air} = \mu_0(1+\chi_0)$ with $\chi_0 = 4 \times 10^{-7}$), and δ is a constant that represents the average difference in the susceptibility of tissue and air (e.g. -9.5×10^{-6} for brain tissues), such that the typical range of χ_1 is from 0 to 1. $\chi_1 = 0$ when only air present in the voxel, $\chi_1 = 1$ when only tissue present in the voxel. The values in between reflect the partial volume susceptibility.

Similarly, ϕ can be expanded in a series

$$\phi = \phi_0 + \delta\phi_1 + \delta^2\phi_2 + \ldots \tag{A.7}$$

This perturbation expansion in δ can be substituted back into equation A.5 to give

$$\mu_0(1+\chi_0)\Delta\phi_0 = 0 \tag{A.8}$$

$$(1+\chi_0)\Delta\phi_1 + \nabla\cdot(\chi_1\nabla\phi_0) = 0 \tag{A.9}$$

for the zeroth and first order terms in δ.

Using the zeroth order equation together with standard vector calculus identities gives a 3D Poisson equation

$$\Delta\phi_1 = \frac{-1}{1+\chi_0}\left(\nabla\cdot(\chi_1\nabla\phi_0)\right). \tag{A.10}$$

The Green's function for this equation is

$$G(\mathbf{r}) = \frac{-1}{4\pi\|\mathbf{r}\|}. \tag{A.11}$$

This allows the solution of the Poisson equation to be written as a convolution

$$\phi_1(\mathbf{r}) = \iiint G(\mathbf{r}-\mathbf{r}')f(\mathbf{r}')\,d\mathbf{r}' \tag{A.12}$$

or more consisely as, $\phi_1 = G * f$, where $f = \frac{-1}{1+\chi_0}(\nabla \cdot (\chi_1 \nabla \phi_0))$.

From this the z-component of the B field can be written as

$$\begin{aligned} B_z &= \mu H_z = \mu \frac{\partial \phi}{\partial z} \\ &= \mu_0(1+\chi_0)\frac{\partial \phi_0}{\partial z} + \delta\, \mu_0 \left(\chi_1 \frac{\partial \phi_0}{\partial z} + (1+\chi_0)\frac{\partial \phi_1}{\partial z} \right) + O(\delta^2). \end{aligned} \tag{A.13}$$

As the zeroth order term is $B_z^{(0)} = \mu_0(1+\chi_0)\,\partial\phi_0/\partial z$, then the first order term is

$$B_z^{(1)} = \frac{\chi_1}{1+\chi_0} B_z^{(0)} + \mu_0(1+\chi_0)\frac{\partial \phi_1}{\partial z}. \tag{A.14}$$

Using the fact that

$$\frac{\partial}{\partial x}(G * f) = G * \frac{\partial f}{\partial x} = \frac{\partial G}{\partial x} * f$$

holds for any G and f, together with equations A.8, A.9, A.12 and A.14, gives

$$\begin{aligned} B_z^{(1)} \simeq\ & \frac{\chi_1}{1+\chi_0} B_z^{(0)} - \frac{1}{1+\chi_0} \left(\left(\frac{\partial^2 G}{\partial x \partial z} \right) * \left(\chi_1 B_x^{(0)} \right) + \right. \\ & \left. \left(\frac{\partial^2 G}{\partial y \partial z} \right) * \left(\chi_1 B_y^{(0)} \right) + \left(\frac{\partial^2 G}{\partial z^2} \right) * \left(\chi_1 B_z^{(0)} \right) \right) . \end{aligned} \tag{A.15}$$

More detailed explanation and additional calculations are described in [37].

Bibliography

[1] A Abragam. *The principles of nuclear magnetism*. Clarendon Press, Oxford, 1961.

[2] J.L.R. Andersson, C. Hutton, J. Ashburner, R. Turner, and K. Friston. Modeling geometric deformations in EPI time series. *NeuroImage*, 13:903–919, 2001.

[3] O.J. Arthurs and S. Boniface. How well do we understand the neural origins of the fMRI bold signal? *Trends Neuroscience*, 25:27–31, 2002.

[4] M. Baker. Euclideanspace - building a 3d world. On the World-Wide Web at: http://www.euclideanspace.com/.

[5] P.A. Bandettini, J. Bodurka, and N. Petridou. Direct detection of neuronal activity with MRI: fantasy, possibility or reality? *Applied Magnetic Resonance*, 28, 2005.

[6] P.R. Bannister. *Motion correction for functional magnetic resonance images*. PhD thesis, University of Oxford, 2003.

[7] C.F. Beckmann and S.M. Smith. Probabilistic independent component analysis for functional magnetic resonance imaging. *IEEE Trans. on Medical Imaging*, 23(2):137–152, 2004.

[8] J. W. Belliveau, D. N. Kennedy, R. C. Mckinstry, B. R. Buchbinder, R. M. Weisskoff, M. S. Cohen, J. M. Vevea, T. J. Brady, and B. R. Rosen. Functional mapping of the human visual cortex by magnetic resonance imaging. *SCIENCE*, 254(5032):716–719, 1991.

[9] H. Benoit-Cattin, G. Collewet, B. Belaroussi, H. Saint-Jalmes, and C. Odet. The SIMRI project: a versatile and interactive MRI simulator. *Journal of Magnetic Resonance*, 173:97–115, 2005.

[10] J. Bittoun, J. Taquin, and M. Sauzade. A computer algorithm for the simulation of any nuclear magentic resonance (NMR) imaging method. *Magnetic Resonance Imaging*, 2:113–120, 1984.

[11] J. Bodurka and P.A. Bandettini. Toward direct mapping of neuronal activity: MRI detection of ultraweak, transient magnetic field changes. *Magnetic Resonance in Medicine*, 47:1052–1058, 2002.

[12] J. Bodurka, A. Jesmanowicz, J.S. Hyde, H. Xu, L. Estkowski, and S.J. Li. Current-induced magnetic resonance phase imaging. *Journal of Magnetic Resonance Imaging*, 137:265–271, 1999.

[13] R.B Buxton. *Introduction to Functional Magnetic Resonance Imaging: Principles and Techniques*. Cambridge University Press, 2002.

[14] P.T Callaghan. *Principles of Nuclear Magnetic Resonance Microscopy*. Oxford University Press, 1991.

[15] F.S. Chaudhry. *Smooth Motion of a Rigid Body in* $\Re^2$ *and* $\Re^3$. PhD thesis, University of Oxford, 1999.

[16] S. Clare and P. Jezzard. Rapid t1 mapping using multislice echo planar imaging. *Magnetic Resonance in Medicine*, 45:630–634, 2001.

[17] C.A Cocosco, V. Kollokian, R.K.-S. Kwan, and A.C. Evans. BrainWeb: online interface to a 3D MRI simulated brain database. In *Third Int. Conf. on Functional Mapping of the Human Brain*, 1997.

[18] M. S. Cohen and S. Y. Bookheimer. Localization of brain function using magnetic resonance imaging. *Trends in Neurosciences*, 17(7):268–277, 1994.

[19] D.L. Collins, A.P. Zijdenbos, V. Kollokian, J.G. Sled, N.J. Kabani, C.J. Holmes, and A.C. Evans. Design and construction of a realistic digital brain phantom. *IEEE Trans. on Medical Imaging*, 17(3):463–468, 1998.

[20] R. Deichmann, O. Josephs, C. Hutton, D. R. Corfield, and R. Turner. Compensation of susceptibility-induced bold sensitivity losses in echo-planar fMRI imaging. *NeuroImage*, 15:120–135, 2002.

[21] I. Drobnjak, C.F. Beckmann, and M. Jenkinson. Quantification of motion-related artifacts in simulated fMRI data using ICA. In *Proc. Int. Soc. of Magnetic Resonance in Medicine*, page 2840, 2006.

[22] I. Drobnjak, D. Gavaghan, E. Süli, J. Pitt-Francis, and M. Jenkinson. Development of a fmri simulator for modelling realistic rigid-body motion artifacts. *Magnetic Resonance in Medicine*, 56(2):364–380, 2006.

[23] F. Esposito, E. Formisano, E. Seifritz, R. Goebel, R. Morrone, G. Tedeschi, and F. Di Sale. Spatial independent component analysis of functional MRI time series: to what extent do results depend on the algorithm used? *Human Brain Mapping*, 16:146–157, 2002.

[24] L. Freire and J.-F. Mangin. Motion correction algorithms of the brain mapping community create spurious functional activations. In M.F. Insana and R.M. Leahy, editors, *17th Int. Conf. on Information Processing in Medical Imaging*, pages 246–258. Springer, June 2001.

[25] K.J. Friston, A.P. Holmes, K.J. Worsley, J.-B. Poline, C.D. Frith, and R.S.J. Frackowiak. Statistical parametric maps in functional imaging: A general linear approach. *Human Brain Mapping*, 2:189–210, 1995.

[26] K.J. Friston, S.R. Williams, R. Howard, R.S.J. Frackowiak, and R. Turner. Movement-related effects in fMRI time-series. *Magnetic Resonance in Medicine*, 35:346–355, 1996.

[27] M. Gavrilescu, M E. Shaw, G W. Stuart, P. Eckersley, I D. Svalbe, and G F. Egan. Simulation of the effects of global normalization procedures in functional MRI. *NeuroImage*, 18:532–542, 2002.

[28] G H. Glover and A T. Lee. Motion artifacts in fMRI: comparison of 2DFT with PR and spiral scan methods. *Magnetic Resonance in Medicine*, 33:624–635, 1995.

[29] E.M. Haacke, R.W. Brown, M.R. Thompson, and R Venkatesan. *Magnetic resonance imaging, physical principles and sequence design.* Willey-Liss, 1999.

[30] T Hacklnder and Mertens H. Virtual MRI: a pc-based simulation of a clinical mr scanner. *Academic Radiology*, 12(1):85–96, 2005.

[31] J.V. Hajnal, R. Myers, A. Oatridge, J.E. Schwieso, I.R. Young, and G.M. Bydder. Artefacts due to stimulus correlated motion in functional imaging of the brain. *Magnetic Resonance in Medicine*, 31:283–291, 1994.

[32] J.V. Hajnal, N. Saeed, E.J. Soar, A. Oatridge, I.R. Young, and G.M. Bydder. A registration and interpolation procedure for subvoxel matching of serially acquired MR images. *Journal of Computer Assisted Tomography*, 19(2):289–296, 1995.

[33] R.H. Hashemi and W.G. Bradley. *MRI: the basics.* Lippincott Williams and Wilkins, 1997.

[34] D.L.G. Hill, A.D.C. Smith, A. Simmons, C.R.Jr. Maurer, T.C.S. Cox, R. Elwes, M. Brammer, D.J. Hawkes, and C.E. Polkey. Sources of error in comparing functional magnetic resonance imaging and invasive electrophysological recordings. *Journal of Neurosurgery*, 93:214–223, 2000.

[35] M. Jenkinson. Measuring transformation error by RMS deviation. Internal Technical Report TR99MJ1, Oxford Centre for Functional Magnetic Resonance Imaging of the Brain, Department of Clinical Neurology, Oxford University, Oxford, UK, 1999. Available at www.fmrib.ox.ac.uk/analysis/techrep for downloading.

[36] M. Jenkinson, P.R. Bannister, J.M. Brady, and S.M. Smith. Improved optimisation for the robust and accurate linear registration and motion correction of brain images. *NeuroImage*, 17(2):825–841, 2002.

[37] M. Jenkinson, J. Wilson, and P. Jezzard. A perturbation method for magnetic field calculations of non-conductive objects. *Magnetic Resonance in Medicine*, 52(3):471–477, 2004.

[38] P. Jezzard and R.S. Balaban. Correction for geometric distortion in echo planar images from B0 field variations. *Magnetic Resonance in Medicine*, 34:65–73, 1995.

[39] P Jezzard, AS Barnett, and C Pierpaoli. Characterization of and correction for eddy current artifacts in echo planar diffusion imaging. *Magnetic Resonance in Medicine*, 39(5):801–12, 1998.

[40] P. Jezzard, P.M. Matthews, and S.M. Smith. *Functional MRI: An Introduction to Methods*. OUP, Oxford, 2001.

[41] B. Kim, J.L. Boes, P.H. Bland, T.L. Chenevert, and C.R. Meyer. Motion correction in fMRI via registration of individual slices into an anatomical volume. *Magnetic Resonance in Medicine*, 41:964–972, 1999.

[42] D. Konn, P. Gowland, and R. Bowtell. MRI detection of weak magnetic fields due to an extended current dipole in a conducting sphere: A model for direct detection of neuronal currents in the brain. *Magnetic Resonance in Medicine*, 50(1):40–49, 2003.

[43] D. Konn, S. Leach, P. Gowland, and R. Bowtell. Initial attempts at directly detecting alpha wave activity in the brain using MRI. *Magnetic Resonance Imaging*, 22(10):1413–1427, 2004.

[44] J.B. Kuipers. *Quaternions and Rotation Sequences*. Princeton University Press, 2002.

[45] R.K.S. Kwan, A.C. Evans, and G.B. Pike. fMRI simulation for quantitative evaluation of analysis methods. *Third Int. Conf. on Functional Mapping of the Human Brain*, 3(4):S449, 1997.

[46] R.K.S. Kwan, A.C. Evans, and G.B. Pike. MRI simulation-based evaluation of image-processing and classification methods. *IEEE Trans. on Medical Imaging*, 18(11):1085–1097, 1999.

[47] D Le Bihan, J F Mangin, C Poupon, C A Clark, S Pappata, N Molko, and H Chabriat. Diffusion tensor imaging: concepts and applications. *Journal of Magnetic Resonance Imaging*, 13(4):534–46, 2001.

[48] Z. Liang and P.C. Lauterbur. *Principles of magnetic resonance imaging: a signal processing perspective*. IEEE Press Series in Biomedical Engineering. IEEE Press, Inc., New York, 2000.

[49] N.K. Logothetis, J. Pauls, M. Augath, T. Trinath, and A. Oeltermann. Neurophysiological investigation of the basis of the fMRI signal. *Nature*, 412:150–157, 2001.

[50] H. Lu, X. Golay, J.J. Pekar, and P.C.M. van Zijl. Functional magnetic resonance imaging based on changes in vascular space occupancy. *Magnetic Resonance in Medicine*, 50:263–274, 2003.

[51] R.B. Lufkin, R. Keen, M. Rhodes, J. Quinn, W. Glenn, and W. Hanafee. MRI simulator for instruction in pulse-sequence selection. *American Journal of Roentgenology*, 147(1):199–202, 1986.

[52] A S. Lukic, M N. Wernick, and S C. Strother. An evaluation of methods for detecting brain activations from functional neuroimages. *Art Intel Med*, 25:69–88, 2002.

[53] P. Mansfield. Multi-planar image formation using NMR spin echos. *Journal of Physics C*, 10:L55–L58, 1977.

[54] F. MIDAS-Consortium:Boada, D.L. Collins, I. Drobnjak, W. Eddy, A.C. Evans, M. Griffin, M. Jenkinson, D. Noll, B. Pike, H. Shi, D. Shroff, V.A. Stenger, and K. Worsley. MIDAS - a multi-site fMRI simulator consortium. In *Tenth Annual Meeting of the Organization for Human Brain Mapping*, 2004.

[55] L. Muresan, R. Renken, J.B.T.M. Roerdink, and H. Duifhuis. Position-history and spin-history artifacts in fMRI time-series. *Proc. Medical Imaging 2002: Physiology and Function from Multidimensional Image*, 4683:444–451, 2002.

[56] R.G. Nunes, I. Drobnjak, S. Clare, P. Jezzard, and M. Jenkinson. Quantitative simulation of affine registration for correction of eddy current distortions in diffusion-weighted images. In *Proc. Int. Soc. of Magnetic Resonance in Medicine*, 2005.

[57] S. Ogawa, T.-M. Lee, A. Nayak, and P. Glynn. Oxygenation-sensitive contrast in magnetic resonance image of rodent brain at high magnetic fields. *Magnetic Resonance in Medicine*, 14:68–78, 1990A.

[58] S. Ogawa, D.W. Tank, R. Menon, J.M. Ellermann, S.G. Kim, H. Merkle, and K. Ugurbil. Intrinsic signal changes accompanying sensory stimulation: Functional brain mapping with magnetic resonance imaging. *Proc. Natl. Acad. Sci. USA*, 89(13):5951–5, 1992.

[59] M.B.E. Olsson, R. Wirestam, and B.R.R. Persson. A computer simulation program for MR imaging: Application to RF and static magnetic field imperfection. *Magnetic Resonance in Medicine*, 34:612–617, 1995.

[60] G.S. Pell, D.F. Abbott, S.W. Fleming, J.W. Prichard, and G.D. Jackson. Further steps toward direct magnetic resonance (mr) imaging detection of neural action currents: Optimization of mr sensitivity to transient and weak currents in a conductor. *Magnetic Resonance in Medicine*, 55(5):1038–1046, 2006.

[61] G.S. Pell, D.F. Abbott, G.D. Jackson, S.W. Fleming, and J.W. Prichard. Neurogenic inhomogeneity localisation for detection of activity (nilda). In *Proc. ISMRM 13th Annual Meeting*, page 123, 2005.

[62] J.S. Petersson, J.-O. Christoffersson, and K. Golman. MRI simulation using the k-space formalism. *Magnetic Resonance Imaging*, 11:557–568, 1993.

[63] TG Reese, O Heid, RM Weisskoff, and VJ Wedeen. Reduction of eddy-current-induced distortion in diffusion MRI using a twice-refocused spin echo. *Magnetic Resonance in Medicine*, 49(1):177–82, 2003.

[64] TG Reese, RM Weisskof, and VJ Wedeen. Diffusion NMR facilitated by a refocuse eddy-current EPI pulse sequence. In *Proc. Int. Soc. of Magnetic Resonance in Medicine*, page 663, 1998.

[65] D.A. Rundle, S Kishore, S.B. Seshadri, and F.W Wehrli. MRI simulator: a teaching tool for radiology. In *SPIE Proceedings*, volume 1234, pages 60–65, 1990.

[66] Y Shen, DJ Larkman, S Counsell, IM Pu, D Edwards, and JV Hajnal. Correction of high-order eddy current induced geometric distortion in diffusion-weighted echo-planar images. *Magnetic Resonance in Medicine*, 52(5):1184–9, 2004.

[67] S.M. Smith, M. Jenkinson, M.W. Woolrich, C.F. Beckmann, T.E.J. Behrens, H. Johansen-Berg, P.R. Bannister, M. De Luca, I. Drobnjak, D.E. Flitney, R. Niazy, J. Saunders, J. Vickers, Y. Zhang, N. De Stefano, J.M. Brady, and

P.M. Matthews. Advances in functional and structural MR image analysis and implementation as FSL. *NeuroImage*, 23(S1):208–219, 2004.

[68] R.M. Summers, L. Axel, and S. Israel. A computer simulation of nuclear magnetic resonance imaging. *Magnetic Resonance in Medicine*, 3:363–376, 1986.

[69] G. Torheim, P.A. Rinck, R.A. Jones, and J. Kvaerness. A simulator for teaching mr image contrast behavior. *Magnetic Resonance Materials in Physics, Biology and Medicine*, 2(4), 1994.

[70] R.P. Woods, S.R. Cherry, and J.C. Mazziotta. Rapid automated algorithm for aligning and reslicing PET images. *Journal of Computer Assisted Tomography*, 16(4):620–633, 1992.

[71] M.W. Woolrich, B.D. Ripley, J.M. Brady, and S.M. Smith. Temporal autocorrelation in univariate linear modelling of FMRI data. *NeuroImage*, 14(6):1370–1386, 2001.

[72] K.J. Worsley, A.C. Evans, S. Marrett, and P. Neelin. A three-dimensional statistical analysis for CBF activation studies in human brain. *Journal of Cerebral Blood Flow and Metabolism*, 12:900–918, 1992.

[73] J. Xiong, P.T. Fox, and J.H. Gao. Directly mapping magnetic field effects of neuronal activity by magnetic resonance imaging. *Human Brain Mapping*, 20(1):41–49, 2003.

[74] D.A. Yoder, Y. Zhao, C.B. Paschal, and M Fitzpatrick. MRI simulator with object-specific field map calculations. *Magnetic Resonance Imaging*, 22:315–328, 2004.