Numerical Simulation of Bloch Equations for Dynamic Magnetic Resonance Imaging

Dissertation for the award of the degree
"Doctor of Philosophy" (Ph.D.)
Division of Mathematics and Natural Sciences
of the Georg-August-Universität Göttingen

within the doctoral program
PhD School of Mathematical Sciences (SMS)
of the Georg-August University School of Science (GAUSS)

submitted by

Arijit Hazra
from Burdwan, West Bengal, India

Göttingen, 2016

This work has been done at:

Biomedizinische NMR Forschungs GmbH
am Max-Planck-Institut für Biophysikalische Chemie

Under the supervision of:

Institut für Numerische und Angewandte Mathematik
Georg-August-Universität Göttingen

Thesis Committee

Prof. Dr. Gert Lube (referee)	Institut für Numerische und Angewandte Mathematik Georg-August-Universität Göttingen
Prof. Dr. Jens Frahm (co-referee)	Biomedizinische NMR Forschungs GmbH Max-Planck-Institut für biophysikalische Chemie

Examination Board:

Prof. Dr. Gert Lube	Institut für Numerische und Angewandte Mathematik Georg-August-Universität Göttingen
Prof. Dr. Jens Frahm	Biomedizinische NMR Forschungs GmbH Max-Planck-Institut für biophysikalische Chemie
Prof. Dr. Hans Hofsaess	Institut für Physik II Georg-August-Universität Göttingen
Prof. Dr. Gerlind Plonka-Hoch	Institut für Numerische und Angewandte Mathematik Georg-August-Universität Göttingen
Jr. Prof. Dr. Christoph Lehrenfeld	Institut für Numerische und Angewandte Mathematik Georg-August-Universität Göttingen
PD. Dr. Hartje Kriete	Mathematisches Institut Georg-August-Universität Göttingen

Date of Oral Examination: 7.10.2016

Dedicated to Koninika

Acknowledgements

First of all, I would like to thank Prof. Dr. Jens Frahm, head of Biomedizinische NMR Forschungs GmbH am Max-Planck-Institut für Biophysikalische Chemie, for offering me this great opportunity to work in an excellent research facility. He has given sufficient freedom and timely input to make the journey of scientific research in his group a memorable experience. His constant encouragement and support helped me to endure difficult phases of my work.

I am highly indebted to Prof. Dr. Gert Lube from the Institut für Numerische und Angewandte Mathematik of Georg-August-Universität Göttingen for the supervision of my thesis, his continuous interest and guidance in the mathematical aspects of my work. I had many invaluable discussions with him which introduced me to a lot of different areas of mathematical research.

I am deeply grateful to Dr. Dirk Voit for introducing me to the field of MRI and numerous hours of discussions subsequently about different theoretical and experimental aspects of MRI. His amazing ability to explain difficult concepts intuitively has sharpened my understanding about this field.

I would like to thank Arun Joseph for important discussions about theoretical and experimental aspects of flow MRI. I am thankful to him for helping me to adjust with the life in Germany by informing me about several day-to-day and administrative issues from the very beginning.

I would like to express my sincere gratitude to Volkert Roeloffs, Andreas Merrem and Zhengguo Tan for sharing their academic insights, giving me their opinions on my thesis drafts. I am really grateful to Andreas and Volkert for helping me in numerous daily life and administrative issues. I would also like to thank Jost Kollmeier for helping me during the flow experiments and providing me the data for the contrast agent experiments. Also, I would like to acknowledge Xiaoqing Wang, Markus Untenberger for very fruitful academic discussions in multiple occasions.

Apart from this, a major thanks goes to Kurt Bhöm, Oleksandr Kalentev and former colleagues Sebastian Schätz and Christian Holme for sharing their skills and insights about large scale computing and Linux operating systems.

Further, I would like to thank all of the present and past groups members in Biomed NMR to make it such a comfortable place to work.

I am very much thankful to my friends in Göttingen to make last few years a truly amazing, multi-coloured experience. Life would have been hard without their friendship and cheerful presence. I would like to thank everyone of my old friends from India and abroad for their priceless friendships, shared experiences and beautiful memories.

I would specially like to acknowledge more than a decade long friendship of Swarnendu Sil. His general insights about mathematical and scientific research have really helped me to appreciate and enjoy my work more. I would also like to acknowledge the person who is my oldest friend and the first influential teacher- my brother Somjit.

Many thanks goes to my family for their emotional support, freedom and affection in each and every step of my life. I would specially like to mention my father who always believed in me and the decisions I took and provided me with as much support as possible. This important occasion reminds me of my mother who passed away long ago but her sympathetic and kind nature shaped me more than anyone else.

Last but not the least, I would like to thank Koninika for being such an understanding, loving person and staying by my side for the last decade throughout all my whims, stupidities and ventures.

Contents

List of Figures **xi**

List of Tables **xvii**

1 Introduction **1**
1.1 Organization of the Thesis 3

2 Fundamentals of Magnetic Resonance Imaging **5**
2.1 NMR Phenomena 5
2.2 Bloch Equation 7
2.3 Signal Detection 10
2.4 Signal Localization 12
2.4.1 Slice Selection 13
2.4.2 Spatial Encoding 15
2.4.3 k-space Sampling 15
2.5 Imaging Sequence 17
2.5.1 Cartesian Gradient Echo Sequence 18
2.5.2 Radial Gradient Echo Sequence 19
2.5.3 Fast Low Angle Shot (FLASH) 20
2.6 Image Reconstruction 21
2.6.1 Gridding and FFT 21
2.7 Parallel Imaging 22
2.7.1 Nonlinear Inverse Reconstruction 24
2.8 Principles of Flow MRI 26
2.8.1 Phase-Contrast MRI 26

3 Simulation of Bloch Equations for Spatially Stationary Objects **29**
3.1 Bloch Equation for Spatially Stationary Object 30

3.2 Numerical Strategies . . . 31
3.2.1 Explicit Runge-Kutta Method . . . 33
3.2.2 Operator Splitting . . . 35
3.3 Bloch Equation Simulator . . . 41
3.3.1 Computational Model . . . 41
3.3.2 Pulse sequence . . . 42
3.3.3 Implementation . . . 43
3.3.4 Parallel Computing . . . 45
3.4 Results . . . 46
3.4.1 Slice Profile . . . 46
3.4.2 Comparison between Numerical Methods . . . 48
3.4.3 Effect of the Number of Subvoxels and Isochromats . . . 48

4 Simulation of Bloch Equation for Moving Spins 51
4.1 Bloch Equation for Flowing Spins . . . 52
4.2 Numerical Strategies . . . 53
4.3 Numerical Strategies for the Solution of Advection Equation . . . 56
4.3.1 Time Discretization . . . 58
4.3.2 Spatial Discretization . . . 59
4.3.3 Boundary Conditions and Ghost Cells . . . 63
4.4 Bloch Simulator for Flowing Spins . . . 64
4.4.1 Computational Model . . . 65
4.4.2 Implementation . . . 66
4.4.3 Parallelization . . . 68
4.5 Results . . . 69

5 Comparison of Simulations with Experimental Results 73
5.1 MRI System . . . 73
5.2 Experimental Equipments . . . 74
5.2.1 Static Phantom . . . 74
5.2.2 Flow Equipments . . . 75
5.3 Validation of the Static Case . . . 76
5.3.1 Single-channel Loop Coil Experiment . . . 76
5.3.2 Experiment with Multiple Tubes . . . 77
5.4 Application of Simulation for Parameter Estimation . . . 80
5.5 Evaluation of the Simulation of MRI for Flowing Spins . . . 83
5.5.1 Proof of Concept . . . 83

5.5.2 *In Vitro* Experiments with Laminar Flow 85
5.5.3 *In Vitro* Experiments with Pulsatile Flow 88

6 Summary and Outlook 93
6.1 Summary 93
6.2 Outlook 95

Appendix A Definitions, Theorems and Results Related to ODE Systems 97
A.1 Solution of Bloch Equations by Operator Splitting 101

Appendix B Existence and Uniqueness of Bloch Equation for Flowing Spins 107

Appendix C Discontinuous Galerkin Method for Advection Equation 113

Appendix D Briefly on the Numerical Analysis of Partial Differential Equation 119
D.1 Definitions and Theorems Related to the Solution of Advection Equation 121

Abbreviations 125

References 129

List of Figures

2.1 Schematic of a pulsed NMR experiment. (Top left) In equilibrium, $\mathbf{M}$ align along the static magnetic field $\mathbf{B}_0$. (Top middle) radio-frequency (RF) excitation tilts the magnetization from the longitudinal direction. (Top right) Precession of magnetization. (Bottom left) spin-spin relaxation time (T_2) relaxation. (Bottom right) spin-lattice relaxation time (T_1) recovery. 8

2.2 Schematic diagram of the signal detection. (Left) Signal demodulation (SD). (Right) Quadrature detection. 11

2.3 Schematic diagram depicting the relations of slice selection gradient, RF sinc pulse and the slice thickness. Different gradient strengths (G_1 and G_2) create slices of different thickness at different positions ($2L_{s1}$ and $2L_{s2}$) for same envelope $B_1^e(t)$ functions of a sinc pulse. $\mathcal{F}$ refers to the Fourier transform. 13

2.4 Envelope function of the sinc pulses with different window functions for RF pulse duration of 4 ms and 4 zero-crossings 14

2.5 Schematic illustration of typical k-space trajectories in magnetic resonance imaging (MRI). (Left) Cartesian. (Right) Radial. 16

2.6 Free induction decay. 17

2.7 Generic spoiled gradient echo sequence diagram. Gradients: (a) slice selection (b) rewinder (c) phase encoding (d) prephasing in read direction (e) readout. The colored line in the phase encoding direction corresponds to the colored line in the k-space. 18

2.8 Generic spoiled gradient echo sequence with radial trajectory. Gradients: (a) slice selection (b) rewinder (c) prephasing (d) readout. The colored line in the k-space corresponds to the current repetition 19

2.9 Radial Acquistion with 3 spokes and 5 turns 21

2.10 Flow compensation and velocity encoding gradient. (Left) Flow compensation (FC) gradient waveform for compensating constant velocity which results in zero-phase for both (a) static as well as (b) moving spins with constant velocity. (Right) Velocity encoding (VENC) gradient waveform results in zero phase for the (a) static spin but a net phase for the (b) moving spins with constant velocity. 27

3.1 Schematic diagram of computational domain. 42

3.2 (Left) The envelope function of the Hanning-Windowed RF pulse with different number of side lobes. (Right) The slice profile at the end of one RF pulse duration resulting from the pulse with the corresponding envelope function. The envelope function on the left and the resulting slice profile on the right are marked with the same color. 46

3.3 (Left) The envelope function of the Hanning-Windowed RF pulse. (Right) The slice profile at the end of RF pulse. $2L_s$ represents the nominal slice thickness and $2L'_s$ represents the actual slice thickness over which the RF pulse flips the equilibrium magnetizations. 47

3.4 Single point excitation i.e. 0-dimensional case $T_1 = 1000\,\text{ms}$, $T_2 = 100\,\text{ms}$ with 101 isochromat elements having constant off-resonance from -50 to $50\,\text{Hz}$ 48

3.5 Magnitude of averaged integrated signal intensities are plotted as a function of frame number. The figures correspond to objects with following relaxation times: (Top) $T_1 = 296\,\text{ms}$, $T_2 = 113\,\text{ms}$, (middle) $T_1 = 456\,\text{ms}$, $T_2 = 113\,\text{ms}$, (bottom) $T_1 = 456\,\text{ms}$, $T_2 = 113\,\text{ms}$. (Left column) Simulated results are shown for three different objects for the computational domain of $4.8 \times 4.8 \times 18.0\,\text{mm}^3$ divided into $15 \times 15 \times 27$ subvoxels. (Right column) Simulated results are shown for three different objects for the computational domain of $4.8 \times 4.8 \times 18.0\,\text{mm}^3$ divided into $27 \times 27 \times 27$ subvoxels. For each of these cases, simulations were carried out with subvoxels consisting of 1, 21, 41 and 61 isochromats respectively. 49

4.1 Schematic representation of a 2-D grid. Ω represent the computational domain. The boundary of the domain Γ is marked with red line. i-th cell is magnified and Ω_i and Γ_{ij} represent the area of the i-th cell and the common surface between i-th and the j-th cell respectively. 57

4.2 Schematic representation of a 1-D cell-centred grid. 60

4.3 Limiter function $\psi(\theta)$. The shaded region shows the high-resolution total variation diminishing (TVD) region. Green, red, blue lines lie along the boundary of the superbee,Van Leer and the minmod limiter functions. $\psi(\theta) = 1$ and $\psi(\theta) = \theta$ represent the boundary of the Lax-Wendroff and the Beam-Warming methods. 62
4.4 Schematic illustration of ghost cell in one-dimensional computational domain. 64
4.5 Schematic diagram of computational domain. 65
4.6 A 90° slice-selective pulse was used for the studying the flow-effects. The arrows indicate the time when the data was recorded. 69
4.7 Simulated magnetization distributions of M_x, M_y, M_z for the through-plane velocity u_z along the positive z-axis in the range 0 to 10 cm s^{-1} using splitting algorithm in the present work (Right) are compared with the results in [142] (Left) . The magnetizations were recorded at the end of the post excitation rephasing gradient as marked by the arrow Figure 4.6. The length in the slice direction is from −10 to 10 mm. . . 70
4.8 Simulated magnetization distributions of M_x, M_y, M_z for the through-plane velocity u_z along the positive z-axis in the range 10 to 80 cm s^{-1} using splitting algorithm in the present work (Right) are compared with the results in [142] (Left) . The magnetizations were recorded at the end of the post excitation rephasing gradient as marked by the arrow Figure 4.6. The length in the slice direction is from −10 to 10 mm. . . 71
4.9 Simulated magnetization distributions of M_x, M_y, M_z for the through-plane velocity u_z along the positive z-axis in the range 80 to 200 cm s^{-1} using splitting algorithm in the present work (Right) are compared with the results in [142] (Left) . The magnetizations were recorded at the end of the post excitation rephasing gradient as marked by the arrow Figure 4.6. The length in the slice direction is in the range −15 to 15 mm. 72

5.1 (Top) MRI system and (Bottom) receiver coils. (Bottom left) 64-channel head coil, (bottom middle) 18-element thorax coil, (bottom right) single-channel loop coil. 74
5.2 Static phantom with tubes containing liquids with known T_1 and T_2 parameters 75
5.3 (Left) Flow tube made of glass materials. (Right top) Programmable voltage controller. (Right middle) silicone rubber hose. (Right bottom) Flow pump 75

5.4 MRI experimental set-up with a single channel loop coil and a tube containing a liquid with known T_1 and T_2. (Left) The placement of the tube inside the scanner during the experiment. (Right) The placement of tube inside the loop-coil. 77
5.5 Comparison of the normalized energy of the experiment and simulation for four separate tubes. 78
5.6 Principal setup of the experiment with the static phantom consisting of several tubes with predetermined T_1 and T_2. (Left) Frontal or *coronal* plane view of the phantom. Yellow rectangle represents the slice. (Middle) Axial or *transverse* plane view. Yellow rectangle and the central circle show the field of view (FOV) and the isocenter respectively. The marker outside the phantom is placed to locate the position of the tubes.(Right) Lateral or *sagital* plane view. 78
5.7 (Left) Image of the container. (Right) Comparison of simulation with the image for four different liquids. 79
5.8 (Left) Image with spatially inhomogeneous coil profile. (Middle) White region containing only tap water and the black region is masked out. (Right) Estimated coil profile obtained fitting data over the white region. 80
5.9 (Left) Image of the static phantom after eliminating the coil inhomogeneity effect. (Right) Comparison of simulation with the image, compensated for spatial inhomogeneity, for four different liquids. 81
5.10 Signal enhancement for two different contrast agents for four different flip angles 82
5.11 (Left) Slice profile at echo time (TE) for the first frame and in dynamic equilibrium for 0, 50, 100 and 400 $\mathrm{mm\,s^{-1}}$ through-plane velocities. (Right) Slice profile at TE in dynamic equilibrium for through-plane velocities in the range of 0 to 400 $\mathrm{mm\,s^{-1}}$. 84
5.12 (Left) Time evolution of averaged integrated pixel intensities as a function of frame for through-plane velocity range 0 to 400 $\mathrm{mm\,s^{-1}}$. (Right) Normalized steady-state integrated pixel intensities as a function of constant through-plane velocities. 85
5.13 (Left) magnetic resonance (MR) image of the flow tube. (Right) Signal comparison normalized by the signal intensity of the averaged steady-state signal in dynamic equilibrium for last ten frames for different velocities under different operating conditions of the flow pump. The experimental result is represented by the solid line and the simulation is represented by the dotted line. 86

5.14 The velocity contours in the tube for three consecutive measurements when the pump was operated at 3 V (top) and 6 V (bottom). 88
5.15 Experimental setup for the pulsatile flow experiment. Water flowed through the left tube. Middle and right tubes contain static tap water. 89
5.16 (Left) Operational voltage diagram of the pump to create pulsation. (Right) Fitted through-plane velocity profile from phase contrast imaging (PC MRI) data. 89
5.17 The effect of pulsatile flow on the signal can be observed here. Signal is normalized with the steady-state signal of the static water. 90
5.18 (From top right clockwise) Contour plots for four equidistant different time point in one pulsation period. 90

B.1 Schematic diagram of pipe flow for illustrating domain (Ω) and domain boundaries. Γ_- marked with red color represents the inflow boundary. Γ_+ marked with green color represents the outflow boundary. The blue line represents impermeable walls of the pipe which is no-flow boundaries, denoted by the symbol Γ_0. $\mathbf{u}$ is the velocity field with flow direction (marked by the arrow below $\mathbf{u}$) from the inflow boundary towards the outflow boundary. 107

C.1 Schematic representation of a 2-D grid. Ω represent the computational domain. The boundary of the domain Γ is marked with red line. i-th cell is magnified and Ω_i and E represent the area of the i-th cell and the edge between i-th and the j-th cell respectively. 114
C.2 (Left) One dimensional example of average and jump operators. (Right) The interface between the i and j-th cell where $j > i$ is depicted with the used notation. The orientation of the outward normal is from lower to higher numbered cell. 115

D.1 Schematic diagram explaining the Courant-Friedrich-Lewy (CFL) criteria for a three-point scheme. (Left) An unstable three point scheme. The shaded region shows the numerical domain of dependence which does not contain the true domain of dependence (Right) A stable three point scheme. True domain of dependence which is marked by white cone in the centre contained in the numerical domain of dependence.The extra numerical domain is shown by the surrounding shaded region. . . 122

List of Tables

2.1 Window functions 14

4.1 Limiter functions 63

5.1 Relaxation constants for the liquids in different tubes 75
5.2 Relaxavities of the contrast agents 82
5.3 Mean and standard deviation velocities and Re based on the mean velocity for different operating voltages of the flow pump at temperature 16 °C (kinematic viscosity $\nu = 1.1092 \times 10^{-2}\,\mathrm{cm}^2/\mathrm{s}$ [137]). 87

Chapter 1

Introduction

Magnetic resonance imaging (MRI) is a powerful modality for diagnostic imaging that uses a high magnetic field and nonionizing radio-frequency (RF) irradiation to create images at high spatial resolution. They are obtained by exciting and detecting a multitude of spatially encoded nuclear magnetic resonance (NMR) signals from mobile hydrogen atoms within organs and tissues. At the current stage, MRI finds applications in both clinical radiology and biomedical research as it offers several advantages over other biomedical imaging techniques such as X-ray, computerized tomography (CT), positron emission tomography (PET) and ultrasound. Firstly, due to the absence of ionizing radiation, MRI is non-invasive unlike X-ray, CT or PET and therefore may extensively be applied without harm. Secondly, MRI can be used for imaging cross-sections as well as three-dimensional volumes without being hampered by problems such as finite penetration depth or internal reflection as in the case for ultrasound. And thirdly, MR images provide excellent soft-tissue contrast and pathological sensitivity, which facilitates diagnosis and allows for efficient monitoring of disease progression and treatment in various organs including brain, heart, joints and breast.

One of the powerful features of MRI is that the image contrast can be manipulated by varying the type, order, strength and duration of the applied RF excitation pulses and magnetic gradient fields. Altering this pattern, which is commonly known as a MRI pulse sequence, it is possible to exploit a wide range of contrast mechanisms including access to physiological functions such as diffusion, flow, blood oxygenation, cellular metabolism and tissue temperature. Therefore, MRI is not restricted to a qualitative description of anatomy, but also serves as a powerful tool for interventional, functional, metabolic and quantitative studies, which have a huge significance in diagnostic imaging.

On the other hand, MRI also has some disadvantages, which include its low acquisition speed and high costs due to the requirement of a super-conducting magnet. Because of this need for a high magnetic field, MRI technology may also not be accessible to patients with metal implants. To increase the acquisition speed and accelerate MRI, a number of fast imaging techniques such as fast low angle shot (FLASH) [43, 55, 42], echo planar imaging (EPI) [86] and rapid acquisition with relaxation enhancement (RARE) [64] were devised in the past. However, these pulse sequences alone are not sufficient to generate a continuous stream of fast images at such a high speed as required for studying physiological processes such as speaking, swallowing or rapid complex fluid motion. In order to achieve even faster image acquisitions at a temporal resolution of 10 to 40 ms to observe complex physiological processes, our group developed a method which combines highly undersampled fast low angle shot (FLASH)-type acquisitions with image reconstruction by an iterative optimization of a nonlinear inverse problem [132, 131, 133].

Although imaging of rapid processes is improved considerably with this technique, there are still unsolved problems such as a quantitative understanding of the mechanisms that lead to MRI signal alterations (i.e., both enhancement and loss) when imaging flowing spins (e.g., in vessels or the heart) or other dynamic processes. In fact, apart from flow velocities and volumes, there is an increasing demand in MRI for quantitative information such as relaxation time constants. In future, access to both high-contrast imaging and quantitative parametric mapping by MRI is expected to facilitate and contribute to computer-aided diagnostic strategies.

The main purpose of this thesis is to focus on the quantitative analysis of dynamic signal changes with a special focus on flow. Numerical simulations will be applied to study the effect of flowing spins on the MRI signal evolution during dynamic imaging. Specific aims are as follows:

(i) To develop a simulator for spatially stationary objects which is based on precise mathematical modelling and numerical techniques

(ii) To compare and validate selected results obtained by the simulator with laboratory experiments

(iii) To explore the possibility to quantify parameters such as relaxation times by comparing simulation results with experimental findings

(iv) To provide a comprehensive analysis of real-time flow MRI by incorporating the flow effect into the simulator for spatially stationary objects

(v) To compare and validate the simulations for flow imaging with laboratory experiments.

(vi) To extend the simulators by parallel implementation

In order to accomplish these goals, this thesis comprises both the theoretical analysis and numerical implementations of the governing equations of MRI for spatially stationary and flowing objects. Operator splitting methods are used for the simulation of spatially stationary objects and further extended for the simulation of flowing spins. The simulation techniques are also implemented in a CUDA-enabled graphical processing unit (GPU). The simulation results for the MRI signal behavior were compared to a number of laboratory MRI experiments performed on a commercial MRI system operating at a field strength of 3 Tesla. Specific questions of increasing complexity addressed the influence of the excited slice profile, contrast agents and flow.

1.1 Organization of the Thesis

A brief overview of the basic principles of MRI which are essential for understanding the present work is given in Chapter 2. The chapter introduces the basics of nuclear magnetic resonance (NMR), gradient echo (GE) pulse sequences, image reconstruction techniques and experimental approaches to quantitative flow imaging. Theoretical analysis and numerical implementations of operator splitting techniques for the simulation of MRI for spatially stationary objects are discussed in Chapter 3. In Chapter 4, splitting techniques are further extended for studying the effect of flowing spins on MRI. A part of this chapter is devoted to discuss finite volume method (FVM) which is used for solving the magnetization transport. In addition, preliminary results are presented at the end of the previous two chapters. In the first part of Chapter 5, the simulator for spatially stationary object is evaluated against experiments with single-compartment and multi-compartment phantoms consisting of different aqueous solutions. The last part of Chapter 5 is devoted to the comparison of simulation with *in vitro* flow experiments. Finally, Chapter 6 summarizes the main achievements of this thesis and presents an outlook of prospective work in future.

Chapter 2

Fundamentals of Magnetic Resonance Imaging

This chapter gives a brief introduction to the basic magnetic resonance imaging (MRI) principles. MRI is based on the phenomena of nuclear magnetic resonance (NMR) and its effect on condensed matter discovered by Bloch [17] and Purcell [107] in 1946. Beginning with the basic physics of NMR the equations governing the macroscopic time evolution of magnetizations are derived. The signal generation and acquisition techniques in MRI and the image formation principles based on the acquired signals are also briefly reviewed. In the end, MRI techniques for flow imaging are discussed.

For detailed discussions on the topic of MRI the reader is referred to the textbooks by Haacke et al. [54], Liang et al. [83] and Bernstein et al. [13].

2.1 NMR Phenomena

The basic principles of NMR are based on the fundamental property of the *spin*. The spin is an intrinsic form of angular momentum $\mathbf{J}$ observed in elementary atomic particles and atomic nuclei. Although the spin is a quantum mechanical property, in the classical mechanical model, it can be visualized as a spinning top. MR physics can be explained satisfactorily with the classical model. However, unlike classical magnetic momentum, the spins of an elementary particle can take only some discrete magnitudes based on its *spin quantum number* I given by

$$|\mathbf{J}| = \hbar\sqrt{I(I+1)}, \tag{2.1}$$

where I can only be an integer, half-integer or zero and $\hbar$ is the Planck constant divided by 2π.

The atomic nuclei with a non-zero I induces a magnetic moment $\boldsymbol{\mu}$ and the relation between the angular momentum and the magnetic moment is given by

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

where γ is a gyromagnetic ratio.

Although the magnitudes of the spins for a specific atomic nuclei are fixed, the directions of the induced magnetic moments are completely random due to random motions in thermal equilibrium conditions which results in zero macroscopic magnetism. However, in the presence of an external magnetic field $\mathbf{B}_0$ the spins align themselves in discrete energy states given by

$$E_m = -m_I \gamma \hbar B_0, \tag{2.3}$$

where m_I which is known as *magnetic quantum number*, can take only some discrete $2I+1$ possible values from the set $\{-I, -I+1, \cdots I\}$ corresponding to a spin quantum number I. This phenomenon is known as *Zeeman splitting*.

Due to the abundant presence of the protons of hydrogen atoms (^{1}H) in all living tissues, they are primarily used in MRI. A ^{1}H has a spin quantum number of $I = 1/2$ which leads to two possible energy states given by

$$E_\uparrow = -\frac{1}{2}\gamma \hbar B_0, \qquad E_\downarrow = \frac{1}{2}\gamma \hbar B_0. \tag{2.4}$$

The energy states correspond to parallel $\uparrow$ or anti-parallel $\downarrow$ alignment of the magnetic moments with the external magnetic field. The direction of positive B_0 is chosen to be the longitudinal direction z in MRI. The plane perpendicular to the longitudinal axis is known as the transverse plane.

The energy level difference between the two spin states is given by $\Delta E = \hbar \gamma B_0$. This energy difference results in a spin population difference of the two energy states according to the Boltzmann relationship and is given by

$$\frac{N_\uparrow}{N_\downarrow} = \mathrm{e}^{\frac{\Delta E}{K_b T_a}}, \tag{2.5}$$

where k_b is Boltzmann constant and T_a is absolute temperature. Equation (2.5) implies a slightly higher number of spins in the parallel $\uparrow$ direction.

The average magnetic moment of a spin system inside a sub-volume yields a macroscopic magnetism along the direction of the external magnetic field and its magnitude for a spin-½ system is given by

$$M_0 = \frac{\gamma^2 \hbar^2 N_s}{4 k_b T_a} B_0. \tag{2.6}$$

Equation (2.6) shows that the equilibrium magnetization M_0 is directly proportional to the external magnetic field strength B_0 as well as to the number of spins N_s within the macroscopic volume. The magnetic moment $\boldsymbol{\mu}$ experiences a torque in an external magnetic field $\mathbf{B}_0 = B_0 \hat{e}_z$ and is given by

$$\frac{\mathrm{d}\boldsymbol{\mu}}{\mathrm{d}t} = \gamma \boldsymbol{\mu} \times B_0 \hat{e}_z, \tag{2.7}$$

where $\hat{e}_z$ is the unit vector in the direction of the external magnetic field. The solution of Equation (2.7) shows that magnetic moments describe a nuclear precession clockwise about the z-axis at a fixed polar angle [83]. The angular frequency of the nuclear precession, called *Larmor frequency*, is proportional to the external magnetic field and is given by

$$\omega_0 = \gamma B_0. \tag{2.8}$$

In presence of an external magnetic field, the nuclear magnetic moments are quantized along the direction of the magnetic field but due to the random phase the transversal components of the magnetic moments give a zero macroscopic transversal magnetization. Fundamentally, MRI is based on the following two steps:

(i) Manipulation of the equilibrium magnetization to create a detectable signal from the object of interest.

(ii) The reconstruction of an image of the object from the detected signal using a suitable reconstruction method.

2.2 Bloch Equation

The time evolution of the macroscopic magnetization, in presence of an external static magnetic field $\mathbf{B}_0$ can be obtained from Equation (2.7) by averaging the magnetic moments over a continuum volume [54],

$$\frac{\mathrm{d}\mathbf{M}}{\mathrm{d}t} = \gamma \mathbf{M} \times \mathbf{B}_0. \tag{2.9}$$

Equation (2.9) is based on the implicit assumption that the protons are non-interacting. To get a response from an object undergoing an NMR experiment, the orientation of the longitudinal bulk magnetization is altered by applying an oscillating magnetic field $\mathbf{B}_1(t) = B_{1x}(t)\hat{e}_x + \hat{e}_y B_{1y}(t)$ from a nearby RF transmit coil. If the *resonance condition* is fulfilled, the $\mathbf{B_1}$ field tilts the magnetization towards the transverse plane (Figure 2.1).

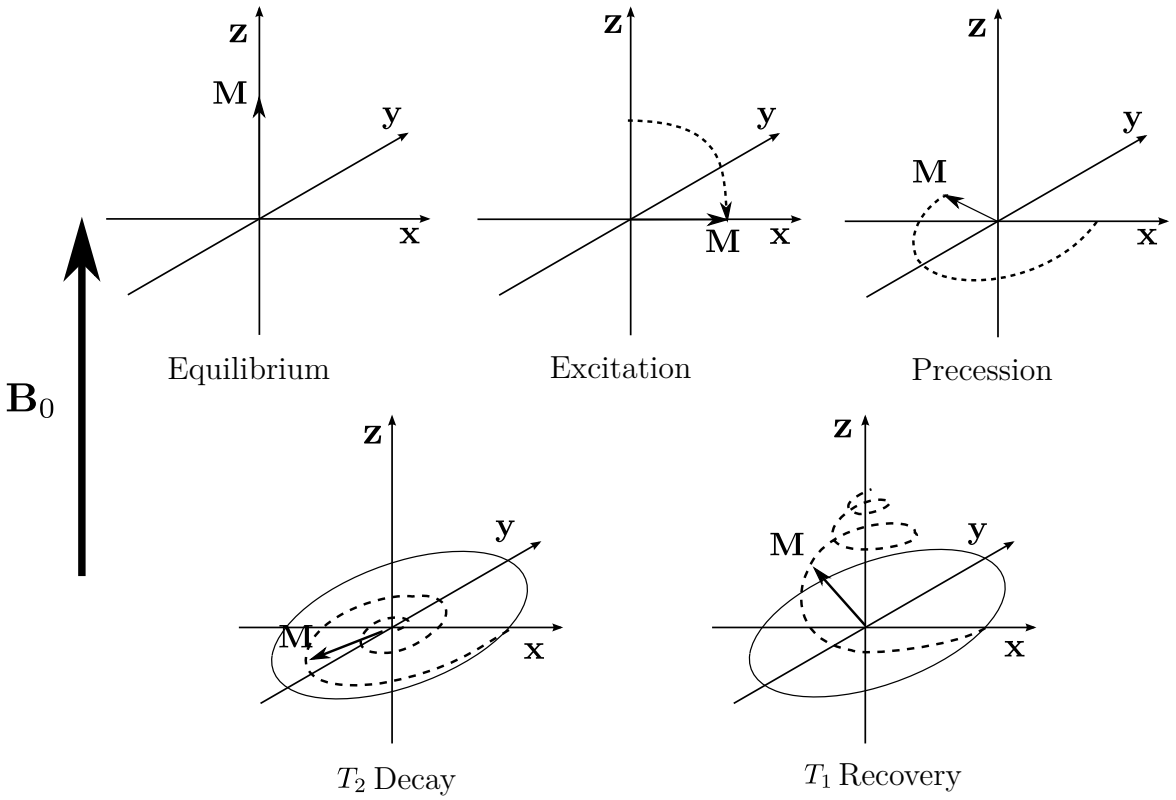

Figure 2.1: Schematic of a pulsed NMR experiment. (Top left) In equilibrium, $\mathbf{M}$ align along the static magnetic field $\mathbf{B}_0$. (Top middle) RF excitation tilts the magnetization from the longitudinal direction. (Top right) Precession of magnetization. (Bottom left) T_2 relaxation. (Bottom right) T_1 recovery.

The RF excitation field is specified by the shape and the duration τ_p of the envelope function $B_1^e(t)$, the excitation carrier frequency ω_{rf} and the initial phase ψ of the RF

pulse, expressed in the complex notation as

$$\mathbf{B_1}(t) = B_{1x}(t) + iB_{1y}(t) = B_1^e(t)\mathrm{e}^{-i(\omega_{\mathrm{rf}}t+\psi)}. \tag{2.10}$$

$B_1^e(t)$ determines the flip angle α of the magnetization due to the RF pulse by the following relation

$$\alpha = \gamma \int_0^{\tau_p} B_1^e(t)dt. \tag{2.11}$$

Immediately after $\mathbf{M}$ is tilted from its equilibrium position, the spins inside the excited volume mutually interact among themselves and with the surrounding to precess towards the equilibrium state again. The precession of the spins towards the equilibrium position, as depicted in Figure 2.1, is characterized by two phenomenologically determined intrinsic time constants:

(i) *spin-lattice relaxation time* T_1 describing the rate of the magnetization recovery in the z direction due to the energy exchange between the spin system and the surrounding chemical environment.

(ii) *spin-spin relaxation time* T_2 describing the rate of the magnetization decay in the transverse plane due to the energy exchange of spins with both the environment and among themselves.

These relaxation phenomena are governed by the following equations:

$$\frac{\mathrm{d}M_z}{\mathrm{d}t} = \frac{M_0 - M_z}{T_1}, \tag{2.12a}$$

$$\frac{\mathrm{d}M_x}{\mathrm{d}t} = -\frac{M_x}{T_2}, \tag{2.12b}$$

$$\frac{\mathrm{d}M_y}{\mathrm{d}t} = -\frac{M_y}{T_2}. \tag{2.12c}$$

The combined effect of the static magnetic field, the RF excitation field and the relaxation are given by the *Bloch equations*,

$$\frac{\mathrm{d}\mathbf{M}}{\mathrm{d}t} = \gamma\mathbf{M} \times (\mathbf{B_0} + \mathbf{B_1}) + \frac{M_0 - M_z}{T_1}\hat{e}_z - \frac{M_x}{T_2}\hat{e}_x - \frac{M_y}{T_2}\hat{e}_y. \tag{2.13}$$

The magnetization in the transverse plane is very often described using a complex notation as $M_{xy}(t) = M_x(t) + iM_y(t)$.

After the RF excitations time evolution of magnetizations are governed by the relaxation and the presence of the static magnetic field. The time evolution of transverse and longitudinal magnetizations can be expressed and solved as follows,

$$\frac{\mathrm{d}M_{xy}}{\mathrm{d}t} = -\omega_0 M_{xy} - \frac{M_{xy}}{T_2} \tag{2.14a}$$

$$\Rightarrow \quad M_{xy}(t) = M_{xy}(t_{\mathrm{rf}})\mathrm{e}^{-t/T_2}\mathrm{e}^{-\omega_0 t},$$

$$\frac{\mathrm{d}M_z}{\mathrm{d}t} = \frac{M_0 - M_z}{T_1} \tag{2.14b}$$

$$\Rightarrow \quad M_z(t) = M_0 + [M_z(t_{\mathrm{rf}}) - M_0]\mathrm{e}^{-t/T_1},$$

where t_{rf} is the time duration of the RF pulse.

2.3 Signal Detection

A pulsed NMR experiment induces a macroscopic magnetism in an object in the form of a rotating magnetization as described in Section 2.2. For detection of the rotating magnetization, the emitted energy from the rotating magnetization is converted into an electric signal.

The magnetic flux χ_m generated by magnetization $\mathbf{M}(\mathbf{r}, t)$ through a receiver coil is given by

$$\chi_m(t) = \int_\Omega \mathbf{C}_r(\mathbf{r}) \cdot \mathbf{M}(\mathbf{r}, t) d\Omega, \tag{2.15}$$

where $\mathbf{C}_r(\mathbf{r})$ is the detection sensitivity of the receiver coil. As soon as the $\mathbf{M}(\mathbf{r}, t)$ is flipped from its thermal equilibrium state, $\mathbf{M}(\mathbf{r}, t)$ precesses towards its equilibrium state, resulting in a time-varying magnetic flux $\chi_m(t)$ in the receiver coils. From Faraday's laws of electromagnetic induction, $\chi_m(t)$ induces a electromotive force (EMF) which is equal to the time-rate of change of $\chi_m(t)$ in the receiver coil and given by

$$V(t) = -\frac{\mathrm{d}}{\mathrm{d}t} \int_\Omega \mathbf{C}_r(\mathbf{r}) \cdot \mathbf{M}(\mathbf{r}, t) d\Omega. \tag{2.16}$$

The time-rate of change of M_z is negligible in comparison to the fast changing M_{xy}.

The magnetic field strength in general varies in the excited volume in an NMR experiment and thus the *Larmor frequency* is spatially dependent. Under this general

condition, Equation (2.14b) gives the following expression for the induced EMF [83].

$$V(t) = \int_{\Omega} \omega(\mathbf{r}) \mathrm{e}^{-t/T_2(\mathbf{r})} \Big| M_{xy}(\mathbf{r}, 0) \Big| \Big| C_{r,xy}(\mathbf{r}) \Big| \cos(-\omega(\mathbf{r})t + \phi_e(\mathbf{r}) - \phi_r(\mathbf{r}) + \frac{\pi}{2}) d\Omega, \tag{2.17}$$

where $C_{r,xy} = C_{r,x} + iC_{r,y}$ represents the effective detection sensitivity of the receiver coil, ϕ_r is the phase of the receiver field, ϕ_e initial phase shift introduced by the RF excitations and $\omega(\mathbf{r})$ is spatially dependent *Larmor frequency*.

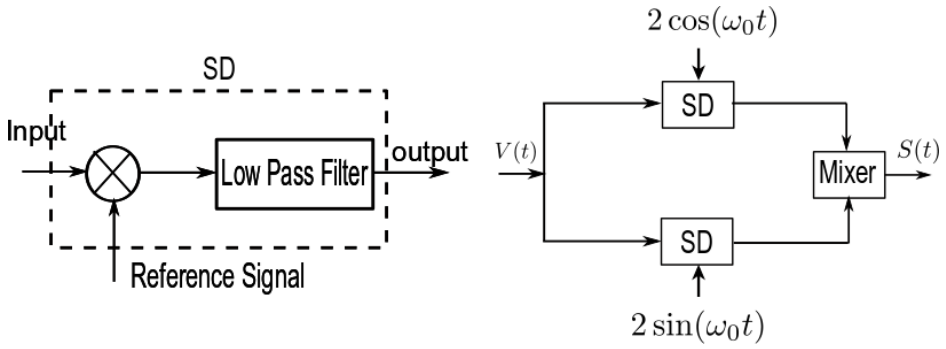

Figure 2.2: Schematic diagram of the signal detection. (Left) Signal demodulation (SD). (Right) Quadrature detection.

The high-frequency voltage signal $V(t)$ is demodulated and the demodulated output signal is detected as illustrated in Figure 2.2 which consists of the multiplication of $V(t)$ by a reference sinusoidal signal, the low-pass filtering of the resulting signal to remove the high-frequency component and the detection of this output signal. The main drawback of this detection system is that the precessing direction (clockwise (CW) or counterclockwise (CCW)) of the magnetization of a spin system can not be determined from the signal.

To overcome this problem, in modern MRI systems a quadrature detection is used as illustrated in the right part of Figure 2.2 where $V(t)$ is demodulated with two sinusoidal reference signals $2\sin(\omega_0 t)$ and $2\cos(\omega_0 t)$. The resulting demodulated signals are detected in two orthogonally placed detectors and combined in a complex signal $S(t)$,

$$S(t) = \omega_0 \mathrm{e}^{i\pi/2} \int_{\Omega} \mathrm{e}^{-i\Delta\omega(\mathbf{r})t} M_{xy}(\mathbf{r}, t) C^*_{r,xy}(\mathbf{r}) d\Omega, \tag{2.18}$$

where $\omega(\mathbf{r}) = \omega_0 + \Delta\omega(\mathbf{r})$. The scaling constant $\omega_0 \mathrm{e}^{i\pi/2}$ can be omitted without any loss of significant information.

The demodulated signal is equivalent to the signal expression obtained from the solution of the Bloch equations in a frame rotating with an angular frequency ω_0, given by

$$\frac{\mathrm{d}\mathbf{M}'}{\mathrm{d}t} = \gamma\mathbf{M}' \times (\mathbf{B}_0 + \mathbf{B}_1 - \frac{\omega_0}{\gamma}\hat{e}_z) + \frac{M_0 - M_z}{T_1}\hat{e}_z - \frac{M_{x'}}{T_2}\hat{e}_{x'} - \frac{M_{y'}}{T_2}\hat{e}_{y'}, \tag{2.19}$$

which conceptually simplifies the RF excitation effect in MRI by eliminating the effect of the static $\mathbf{B}_0$ field. Therefore, Bloch equations are generally solved in a rotating frame.

2.4 Signal Localization

There are basically two fundamental spatial localization methods: selective excitation, where only a slice of the object is excited, and spatial encoding which can be used to encode the signals from excited spins. Both of these techniques are used for 2D imaging where a slice is selected and the remaining two directions are spatially encoded.

Spatial localizations are controlled by magnetic field gradients applied using additional gradient coils. The shape and forms of these magnetic field gradients can be adjusted independently in three orthogonal directions.

The longitudinal magnetic field with an arbitrary magnetic gradient $\begin{bmatrix} G_x & G_y & G_z \end{bmatrix}^T$ and the corresponding *Larmor frequency* can be expressed as

$$B_z = B_0 + \mathbf{r} \cdot \mathbf{G} = B_0 + xG_x + yG_y + zG_z, \tag{2.20a}$$

$$\omega(\mathbf{r}) = \omega_0 + \gamma\mathbf{r} \cdot \mathbf{G}. \tag{2.20b}$$

Equation (2.20b) shows, the precession frequency ω of a spin ensemble changes with a change in the local magnetic field strength. Application of linear magnetic gradients alter the *resonance condition* of the spin ensemble from a distinct frequency to a continuous bandwidth such that signals from different spatial location can be distinguished.

The Bloch equations for a general MRI sequence need to take into account an arbitrary gradient field, magnetic field inhomogeneity ΔB and is given by

$$\frac{\mathrm{d}\mathbf{M}'}{\mathrm{d}t} = \gamma\mathbf{M}' \times (\mathbf{B}_1 + (B_0 + \Delta B + \mathbf{G} \cdot \mathbf{r} - \frac{\omega_0}{\gamma})\hat{e}_z) + \frac{M_0 - M_z}{T_1}\hat{e}_z - \frac{M_{x'}}{T_2}\hat{e}_{x'} - \frac{M_{y'}}{T_2}\hat{e}_{y'}. \tag{2.21}$$

2.4.1 Slice Selection

An RF excitation pulse with limited bandwidth of $\Delta\omega_p$ will only excite spins within a matching frequency range. For a slice selective excitation, a linear field gradient is applied corresponding to the limited bandwidth of the RF pulse as illustrated in Figure 2.3.

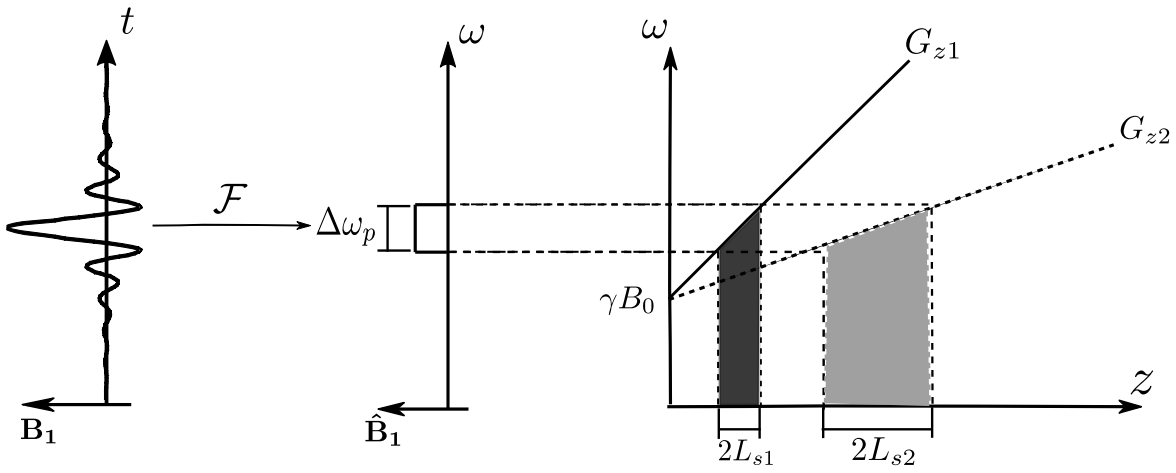

Figure 2.3: Schematic diagram depicting the relations of slice selection gradient, RF sinc pulse and the slice thickness. Different gradient strengths (G_1 and G_2) create slices of different thickness at different positions ($2L_{s1}$ and $2L_{s2}$) for same envelope $B_1^e(t)$ functions of a sinc pulse. $\mathcal{F}$ refers to the Fourier transform.

The frequency bandwidth should be a rectangular function $\Pi(\omega)$ in order to get a perfectly rectangular slice profile so that the excitation pulse will excite spins equally within the slice of the sample leaving the surrounding spins in equilibrium state.

Although the RF excitation pulse $B_1(t)$ is accurately proportional to the Fourier transform of the frequency bandwidth for small flip angles, the same relation is acceptable to a very good approximation even for high flip angles [83]. The identity $\frac{1}{a}\Pi(\frac{f}{a}) \xrightarrow{\mathcal{F}} \mathrm{sinc}(\pi a t)$ implies that a sinc function which has an unlimited support is necessary to get a perfectly rectangular slice profile. As only pulses with finite durations are feasible, a truncated sinc pulse is used which results in a distorted slice profile. Windowing functions are very often used with the truncated sinc pulse to reduce the distortion of the slice profile.

The explicit expression of the envelope function of the sinc pulse is given by [13]:

$$B_1^e(t) = \begin{cases} w(t) B_1 \mathrm{sinc}[{}^{\pi(t-Nt_0)}/_{Nt_0}] & 0 \le t \le \tau_p, \\ 0 \quad \text{otherwise}, \end{cases} \tag{2.22}$$

Table 2.1: Window functions

window function	$w(t)$
rectangular	1
Hamming	$0.50 + 0.50\cos(\pi(t-Nt_0)/Nt_0)$
Hanning	$0.54 + 0.46\cos(\pi(t-Nt_0)/Nt_0)$
Blackman	$0.42 + 0.50\cos(\pi(t-Nt_0)/Nt_0) - 0.08\cos(2\pi(t-Nt_0)/Nt_0)$

where $w(t)$ is a window function, N represents twice the zero-crossing of the sinc pulse and t_0 one half the width.

Figure 2.4 shows sinc pulses for $N = 2$ with different window functions as listed in Table 2.1.

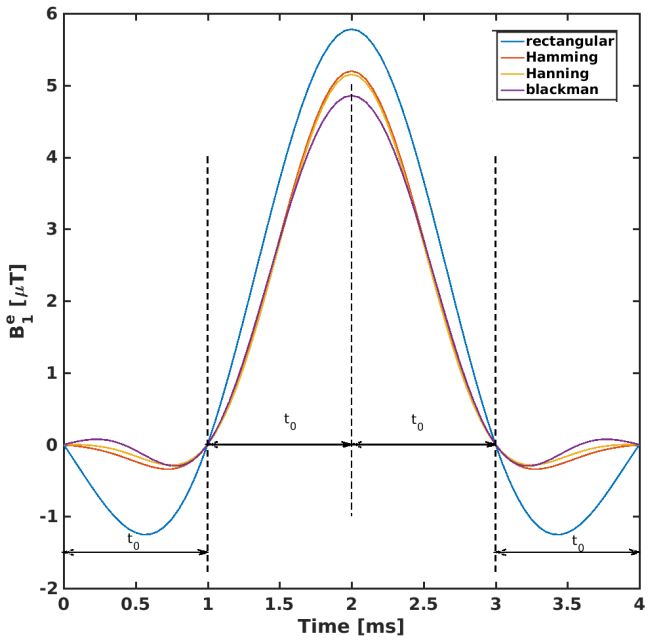

Figure 2.4: Envelope function of the sinc pulses with different window functions for RF pulse duration of 4 ms and 4 zero-crossings

To create a slice profile of thickness $2L_s$, the required slice selection gradient G_{ss} is given by :

$$G_{ss} = \frac{\pi \Delta f}{\gamma L_s} \quad \text{where } \Delta f = \frac{\Delta\omega}{2\pi}\frac{1}{t_0} \tag{2.23}$$

2.4.2 Spatial Encoding

After the slice selection, the spatial localization problem reduces from three to two dimensions and two spatial encoding gradients G_x, G_y are used to encode signals at each location within the selected slice. The magnetic field as a result of the spatial encoding is given by

$$\Delta B = G_x \cdot x + G_y \cdot y. \tag{2.24}$$

In presence of the inhomogeneous magnetic field ΔB the signal expression for spatial encoding becomes

$$S(t) = \int_\Omega C_r^*(\mathbf{r}) \cdot M_{xy}(\mathbf{r}, t_{\mathrm{RF}}) \mathrm{e}^{-i[\gamma x \int_0^t G_x dx + \gamma y \int_0^t G_y dy]} d\Omega. \tag{2.25}$$

For conceptual advantages, the spatial encoding is often expressed in a k-space formalism as

$$S(t) = \int_\Omega C_r^*(\mathbf{r}) \cdot M_{xy}(\mathbf{r}, t_{\mathrm{RF}}) \mathrm{e}^{-i2\pi(k_x \cdot x + k_y \cdot y)} d\Omega, \tag{2.26}$$

where the k-space trajectory is defined by

$$k_x := \frac{\gamma}{2\pi} \int_0^t G_x(\tau) d\tau, \tag{2.27a}$$

$$k_y := \frac{\gamma}{2\pi} \int_0^t G_y(\tau) d\tau. \tag{2.27b}$$

Equation (2.26) shows that the received signal in k-space is the Fourier transform of the dot product between transverse magnetizations and the coil sensitivity map.

2.4.3 k-space Sampling

In principle, the k-space trajectories can be arbitrary. Figure 2.5 shows only two popular k-space trajectories.

In Cartesian sampling strategy, data are collected along lines parallel to an axis one at a time and thus each sample is located on a Cartesian grid. Therefore, fully-sampled Cartesian data require only a 2D inverse fast Fourier transform (IFFT) for image reconstruction. In practice, Equation (2.26) is discretized by sampling at a certain rate. The sampling distance Δk between two neighboring discrete points is related

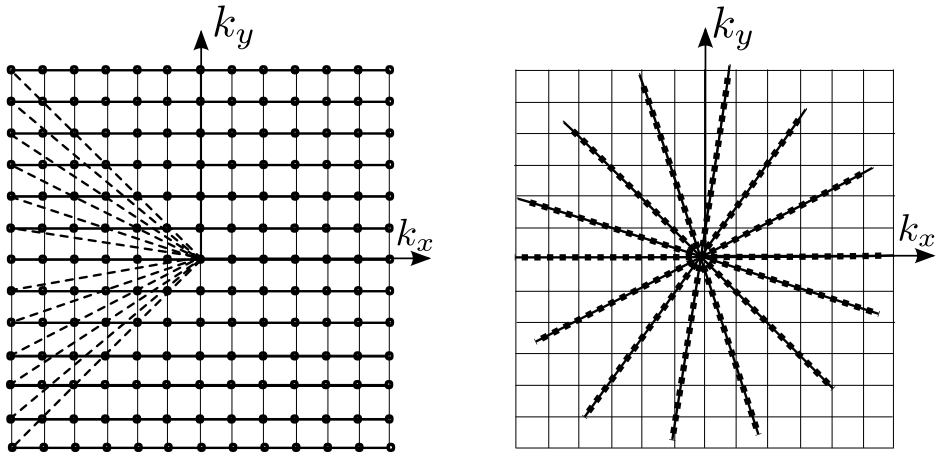

Figure 2.5: Schematic illustration of typical k-space trajectories in MRI. (Left) Cartesian. (Right) Radial.

with the image size i.e. the FOV by

$$\Delta k = \frac{1}{FOV}. \tag{2.28}$$

The discrete FOV is composed of a number of square image elements (pixels) whose characteristic size, defined as spatial resolution, is given by

$$\Delta r = \frac{FOV}{n_b} = \frac{1}{\Delta k \cdot n_b} = \frac{1}{2k_{\max}}, \tag{2.29}$$

where n_b represents the number of time sample points (base resolution) in a single data acquisition, $k_{\max}$ the maximal sampling distance from the centre in the k-space.

In order to avoid aliasing, the sampling distance has to satisfy Nyquist-Shannon criteria given by

$$\Delta k = \frac{2k_{\max}}{n_b} \leq \frac{1}{FOV}. \tag{2.30}$$

On the other hand, according to Equation (2.27), Δk can be expressed as

$$\Delta k_x = \frac{\gamma}{2\pi} G_x \Delta t_x, \quad \Delta k_y = \frac{\gamma}{2\pi} G_y \Delta t_y. \tag{2.31}$$

Sampling interval (dwell time) Δt_x and Δt_y along the G_x and G_y direction can be determined from Equation (2.31) and Equation (2.30).

Ideally an increasing dwell time prolongs the total readout duration which results in a higher signal-to-noise ratio (SNR). The reciprocal of the dwell time Δt is represented

by the receiver bandwidth (BW). Therefore, higher BW results in a faster sampling and low SNR.

In radial sampling strategy shown in Figure 2.5 data are acquired along spokes placed at an angle to an axis. Radial sampling strategy was first proposed by Lauterbur in 1973 [101] and filtered back-projection method was used for the image reconstruction. From the definition of k-space in the Equation (2.27), radial sampling can be achieved with the following gradients

$$G_x = G_{\max} \cos\theta, \tag{2.32a}$$

$$G_y = G_{\max} \sin\theta, \tag{2.32b}$$

where $G_{\max}$ is the maximal gradient amplitude and θ is the angle of the radial spoke. According to the Nyquist criteria of sampling the number of spokes to be acquired n_s should satisfy $n_s \geq \frac{\pi}{2} \cdot n_b$. Undersampling results in streaking artifacts for radial trajectory unlike aliasing as in case of Cartesian sampling [18].

2.5 Imaging Sequence

The most elementary form of signal is the free induction decay (FID) which is basically the received NMR signal immediately after the RF excitation pulse without any manipulation of the primary NMR signal, as depicted in Figure 2.6. The envelope of the FID can be approximated by an exponential function with an effective spin-spin relaxation time (T_2^*) assuming a spin system with Lorenzitian distribution

$$\frac{1}{T_2^*} = \frac{1}{T_2} + \gamma \Delta B_0. \tag{2.33}$$

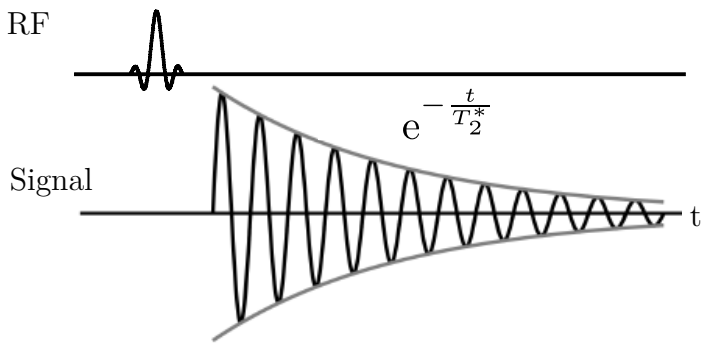

Figure 2.6: Free induction decay.

Imaging sequences relevant to this work are based on the gradient echo (GE) sequences as discussed next.

2.5.1 Cartesian Gradient Echo Sequence

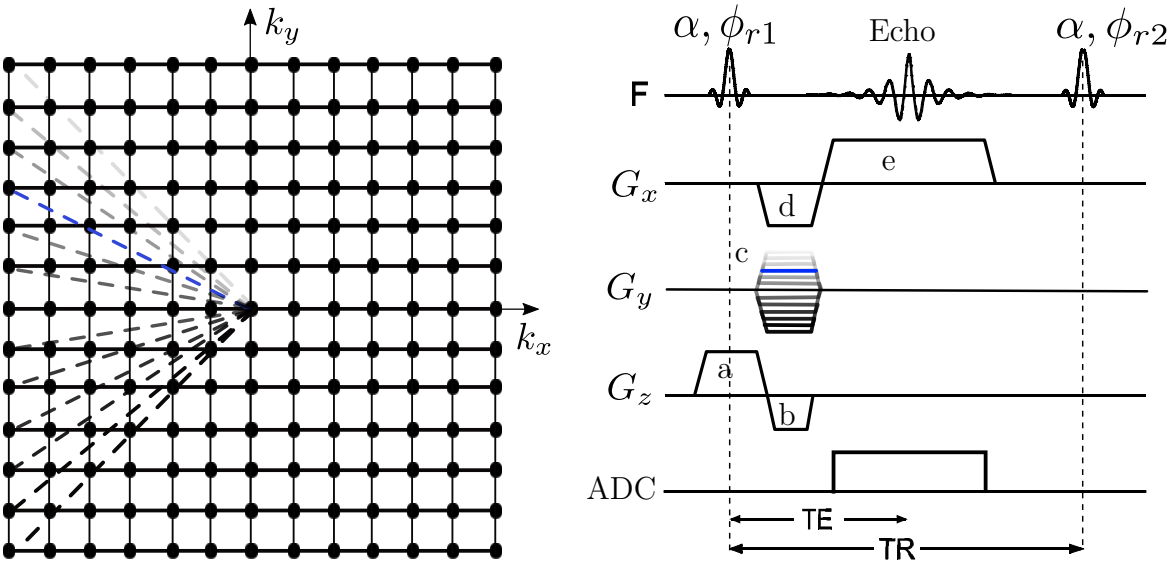

Figure 2.7: Generic spoiled gradient echo sequence diagram. Gradients: (a) slice selection (b) rewinder (c) phase encoding (d) prephasing in read direction (e) readout. The colored line in the phase encoding direction corresponds to the colored line in the k-space.

A generic spoiled GE sequence for 2D imaging is shown in Figure 2.7. A slice selection gradient is applied along with a α pulse and ϕ_{r1} RF phase for selective excitations. After that a rewinder gradient is applied in the slice direction to avoid undesirable signal loss as a result of the phase shift caused by the application of the slice selection gradient. A phase encoding gradient and a prephasing gradient are applied in the direction of phase encoding (y) and readout (x) respectively to accelerate the FID signal decay. Then the dephased spins are rephased by applying a gradient of opposite polarity in the direction of readout.

When the gradient moment of the readout gradient equals the gradient moment of the prephasing gradient in the direction of readout gradient, the spins are completely rephased and form an echo. The time between the center of the RF pulse and the peak of the signal induced is known as echo time (TE) and the time duration from RF pulse to the next RF pulse is defined as repetition time (TR).

Each GE sequence consists of a train of excitation pulses separated by a TR period. Between successive excitation pulses, the spatial encoding is performed with switched

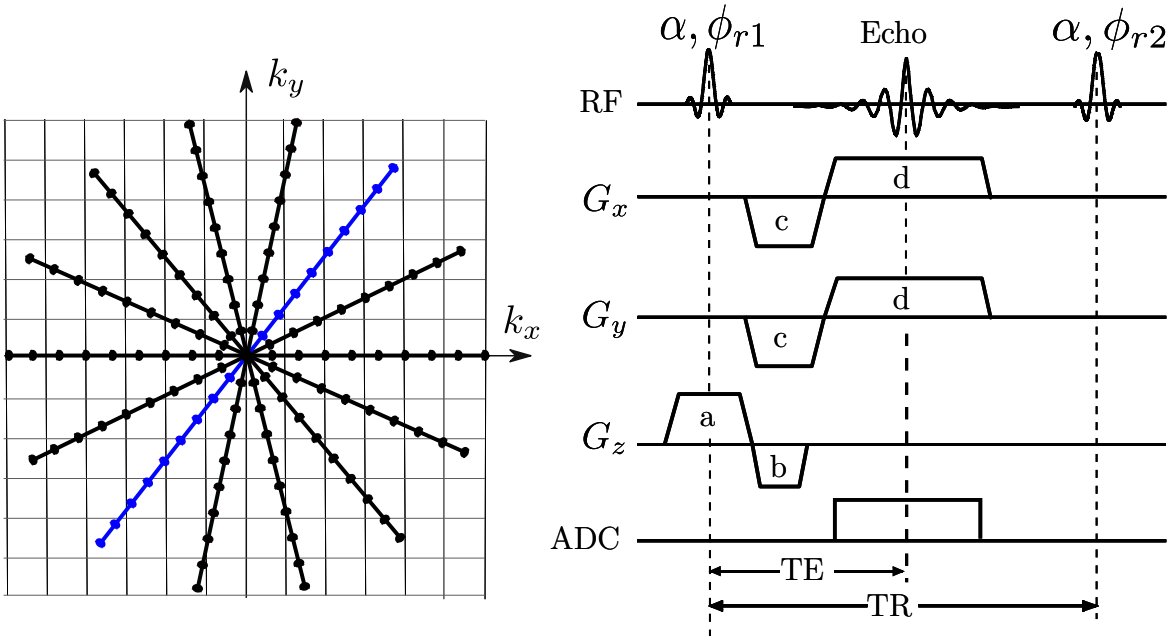

Figure 2.8: Generic spoiled gradient echo sequence with radial trajectory. Gradients: (a) slice selection (b) rewinder (c) prephasing (d) readout. The colored line in the k-space corresponds to the current repetition .

gradients in read, phase and slice direction and one line in k-space is acquired with each repetition of RF.

The effect of excitation, gradient pulses and the magnetic field inhomogeneity on the magnetization vector of each spin is described by Equation (2.21) consisting of α excitation pulse and the precession due to gradient, inhomogeneity and time-relaxation as depicted schematically in Figure 2.1. Excitation pulse in the next TR acts on the modified magnetization and the process of precession is repeated again and again. Carr showed in [23] that under constant α, and gradient moment and constant TR the magnetizations reach a state of dynamic equilibrium after several repetitions which is known as Steady-state Free Precession (SSFP). For clinical imaging, the acquisition starts only after the magnetizations reach SSFP after several preparatory TR repetition.

2.5.2 Radial Gradient Echo Sequence

Figure 2.8 illustrates a GE sequence with radial trajectories. The fundamental difference of radial with Cartesian trajectory is that radial trajectory consists of a readout gradient in two directions unlike one phase encoding and one readout as in Cartesian trajectory.

Although radial encoding scheme was not used widely in the past, it is gaining interest in last decade because of a number of interesting advantages [145]. First of all, radial encoding is relatively more resistant to undersampling than Cartesian encoding. Moreover, undersampling artifacts appear as streaks at the edge of the image while the main structure of the object is maintained. Secondly, the readout gradient in radial GE allows oversampling along both readout directions without additional measuring time. This oversampling enlarges the circular-supported FOV and hence reduces undersampling artifacts. Thirdly, radial encoding is intrinsically robust against motion. Due to the absence of phase encoding, motion-induced ghost artifacts are eliminated as seen very often in Cartesian trajectory.

2.5.3 Fast Low Angle Shot (FLASH)

Fast low angle shot (FLASH) is a specific example of a gradient echo sequence invented by Frahm et al. in 1985 [43, 55, 42] which uses short TR, TE and the low flip angle to produce T_1 weighted images [62]. Due to the low flip angle a significant amount of longitudinal magnetizations remain at the end of each repetition and thus enabling to produce higher signals in the dynamic equilibrium than conventional gradient echo imaging with high flip angles.

However, due to short TR a residual transversal magnetization generally remains after each repetition resulting in artifacts. In spoiled FLASH technique, the residual transversal magnetization is destroyed to avoid artifacts using either gradient or RF spoiling techniques.

(i) Gradient spoiling. A spoiler gradient of high magnitude is applied at the end of the repetition interval to destroy the residual transverse magnetization [44].

(ii) RF spoiling. The RF phase is quadratically incrementally with a suitable angle or changed randomly after each repetition [29]. A spoiler gradient is applied very often additionally at the end of the repetition.

The experiments, conducted in this work, used RF spoiled FLASH sequences with radial encoding schemes [145]. A quadratic increment RF phase with 117° was suggested by Crawley et al. in [29] for Cartesian FLASH sequences. However, in a recent study Volkert et.al. [111] has shown that randomized RF spoiling works better with radial FLASH. Thus, a randomized RF spoiling was used in the present work.

In radial imaging, the order in which the spokes are acquired play a significant role in dynamic imaging as discussed extensively in [145]. The ordering strategy used in this work is described next.

Each reconstructed image frame (*turn*) comprises a certain number of spokes n_s which are uniformly distributed. Afterwards, the spokes are sequentially rotated between successive turns. This pattern is also repeated after certain number of turns n_t. As the orientation of the spokes should be distinct n_t and n_s are both odd numbers. Figure 2.9 depicts schematically employed radial acquisition with $n_s = 3$ spokes and

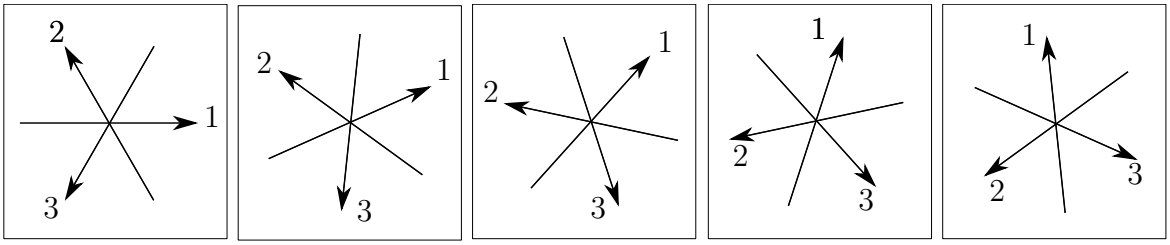

Figure 2.9: Radial Acquistion with 3 spokes and 5 turns

$n_t = 5$ turns. The angle increment between two successive spokes ($\Delta\theta_s$) and between two successive turns are given by

$$\Delta\theta_s = \frac{2\pi}{n_s}, \quad \Delta\theta_t = \frac{2\pi}{(n_s \cdot n_t)} \tag{2.34}$$

respectively. Therefore, the orientation of the i-th spoke in j-th turn is:

$$\theta(i,j) = [(j-1) \bmod n_t] \cdot \Delta\theta_t + (i-1) \cdot \Delta\theta_s. \tag{2.35}$$

2.6 Image Reconstruction

In case of Cartesian sampling, the inverse Fourier transform is applied directly to the sampled k-space data to obtain an image. In radial trajectories, however the sampled k-space data are neither on a Cartesian grid nor equidistant. Thus, the image reconstruction with radial sampled data require advance techniques like non-uniform fast Fourier transform (NUFFT) [41] or interpolation of the data onto a Cartesian grid prior to the uniform IFFT which is known as regridding or simply gridding algorithm as discussed in the next section.

2.6.1 Gridding and FFT

The gridding algorithm was first proposed by O'sulivan in 1985 [98] and detailed discussions can be found in [109, 10, 18]. The gridding algorithm includes the following steps:

(i) Density compensation. Data acquired through radial sampling have varying sample density. Therefore, a density compensation function (DCF) is required to weight the k-space data properly at every sample point in k-space.

(ii) Convolution and evaluation in a grid. The density-compensated data is convolved with a radial interpolation kernel and the convolved data are evaluated in desired k-space positions. Kaiser-Bessel function was found to be an optimal interpolation function for radial acquisitions.

(iii) IFFT is performed on gridded data, oversampled in both directions. The oversampling helps to avoid artifacts.

(iv) Roll-off correction. The convolved data in k-space is equivalent to a dot product between the object and the inverse fast Fourier transform (FFT) of the radial interpolation kernel in image domain. A final reconstructed image is obtained by dividing a 2D IFFT of the oversampled gridded data by the inverse Fourier transform of the interpolation kernel. This is known as roll-off correction.

(v) Cropping. Finally, the final image is cropped to display the original region of interest without aliasing.

2.7 Parallel Imaging

One main drawback of MRI is its relatively low imaging speed. To overcome the speed barrier of MRI, general techniques based on the idea of combining the information acquired through multiple receiver coils have been developed. They are known as parallel imaging methods in MRI. The idea of parallel imaging was first conceived with the introduction of the phased-array coils [112]. The phased-array coils consist of small surface coils, usually placed around the subject in order to simultaneously receive k-space data from localized regions. The MRI signal obtained for j-th coil from a coil system is given by Equation (2.36) which is obtained from Equation (2.26) taking into consideration that $M_{xy}(\mathbf{r}, t)$ is equivalent to the proton density (PD) ρ and replacing the suffixes of the coil sensitivity map with j to denote the complex coil sensitivity from j-th coil,

$$S_j(t) = \int_{\Omega} \rho(\mathbf{r}) \cdot c_j(\mathbf{r}) \mathrm{e}^{-2i\pi(k_x \cdot x + k_y \cdot y)} d\Omega. \tag{2.36}$$

The image reconstruction method from the acquired data with multiple coils can be mathematically posed as an inverse problem where the forward problem is given by,

$$F : x \mapsto \begin{pmatrix} P_k\mathcal{F}\{c_1 \cdot \rho\} \\ \vdots \\ P_k\mathcal{F}\{c_N \cdot \rho\} \end{pmatrix} \quad \text{with } x = \begin{pmatrix} \rho \\ c_1 \\ \vdots \\ c_N \end{pmatrix}, \tag{2.37}$$

where $\mathcal{F}$ is the 2D FFT and P_k is the projection onto the k-space trajectory.

Parallel imaging is mainly designed to use the spatial information of coil sensitivities to allow for the undersampling of k-space and substitute or replace the missing lines in k-space. Therefore, one of the most crucial aspect of the parallel image reconstruction is the calibration method of the coil sensitivities of individual receiver coils. Coil sensitivity maps can be estimated via a pre-scan, auto-calibrated signal (ACS) or jointly with image.

Irrespective of the calibration technique applied, modern parallel imaging methods can be classified roughly into two broad categories: image-domain based techniques (sensitivity encoding (SENSE) [106, 105], nonlinear inverse reconstruction (NLINV) [132, 133, 131]) and regenerative k-space method (simultaneous acquisition of spatial harmonics (SMASH) [91, 120, 70], generalized auto-calibrating partially parallel acquisition (GRAPPA) [50]).

k-space methods are based on the *k-space locality principle* [141], which postulates that k-space data points are highly correlated with their neighbours. Calibration scans or the ACSs are additionally required for determining coil-sensitivity profiles and the weights. The weights are determined by fitting undersampled k-space data to the calibration data. Undersampled data and the weights are utilized subsequently to fill the full k-space. Uncombined coil images obtained by different coils can be combined via sum of squares [112] or phase-preserving coil combination algorithm [138].

On the other hand, the generalized SENSE treats parallel imaging as a linear inverse problem where the following cost function is to be minimized

$$\Phi(x) = \underset{x}{\operatorname{argmin}} \sum_{i=1}^{N} \|S_n - F(\rho \cdot c_i)\|_2^2 + \alpha\|\rho\|_2^2 \quad \text{with } x = \rho. \tag{2.38}$$

This method relies on the accurate estimation of the coil sensitivity map calculated via either calibration scan or auto-calibration method. Once the coil sensitivity maps are obtained, the minimization problem can be solved via regular least-square technique

or iteratively. Regularization method like Tikhonov regularization [36] that penalizes the L^2 norm of the estimated image is also used to overcome the ill-conditioned nature of the inverse problem.

For the present work, NLINV reconstruction method proposed by Uecker et al. [132, 133] was often used as a parallel imaging technique. NLINV is the first method based on the joint estimation of the image content and coil sensitivity maps and is discussed briefly in the next section.

2.7.1 Nonlinear Inverse Reconstruction

Following Equation (2.37) the parallel imaging problem can be expressed as

$$y = F(x), \quad x = \begin{pmatrix} \rho \\ c_1 \\ c_2 \\ \vdots \\ c_N \end{pmatrix}, \tag{2.39}$$

where y is the measured data and the image content ρ and the coil sensitivity maps c_j combined together in a variable x.

In NLINV, iteratively regularized Gauss-Newton method (IRGNM) [36, 8] is used to solve the non-linear system of Equation (2.39) in a least-square sense. Equation (2.39) is linearised $\mathbf{y} \approx DF(\mathbf{x}_n)d\mathbf{x} + F(\mathbf{x}_n)$ about the estimation $\mathbf{x}_n$ in the n-th newton step and a term is added similar as the Tikhnov regularization term. The following minimization problem is solved for the update dx at each step,

$$\Phi(dx) = \underset{dx}{\operatorname{argmin}} \|DF(x_n)dx - (y - F(x_n))\|^2 + \alpha_n \|x_n + dx - x_0\|^2 . \tag{2.40}$$

It can be seen that Equation (2.37) is highly under-determined, as multiplication of ρ by any complex function and dividing the coil sensitivities with the same complex function gives a different solution with same measured signal y. This shift in information from the image ρ to the coil sensitivities c_j can be restricted by adding prior knowledge that coil sensitivities are generally rather smooth even if the object may contain edges. This

prior information is incorporated into NLINV by a pre-conditioning matrix W

$$\hat{x} = W^{-1}x = \begin{pmatrix} I & & & \\ & (1 + s\|k\|_2^2)^{l/2}\mathcal{F} & & \\ & & \ddots & \\ & & & (1 + s\|k\|_2^2)^{l/2}\mathcal{F} \end{pmatrix} x, \tag{2.41}$$

resulting in a new minimization problem

$$\Phi(d\hat{x}) = \underset{d\hat{x}}{\operatorname{argmin}} \|DG(\hat{x}_n)dx - (y - G(\hat{x}_n))\|^2 + \alpha_n \|\hat{x}_n + d\hat{x} - \hat{x}_0\|^2, \text{ where } G\hat{x} = FW\hat{x}. \tag{2.42}$$

Equation (2.42) is a linear least-square problem. The resulting normal equations from the linear least-square problem are solved by conjugate gradient (CG) method. The $\hat{x}_n$ is then updated with an optimized $d\hat{x}$ to give $\hat{x}_{n+1}$.

The update rule of $d\hat{x}$ can be derived from Equation (2.42):

$$d\hat{x} = (DF(\hat{x}_n)^H DF(\hat{x}_n) + \alpha_n I)^{-1}(DF(\hat{x}_n)^H(y - F(\hat{x}_n)) + \alpha_n(\hat{x}_n - \hat{x}_0)). \tag{2.43}$$

An optimal estimation of the image content and the coil sensitivity profiles can be obtained after performing the iteration for a number of newton steps e.g. 6 or 7.

For the reconstruction of a series of images in real time MRI applications, the algorithm is initialized with $\rho = 1, c_j = 0$ for the first frame and the subsequent frames take the estimate from the preceding frame as initial guess. The regularization parameter decays along Newton steps according to $\alpha_n = \alpha_0 2^{-n}, \alpha_0 = 1$.

The measure data is preprocessed before the NLINV image reconstruction via gradient delay corrections. To deal with the enormous amount of multi-channel k-space data in real time MRI applications, the data is compressed to 10 virtual channels via principal component analysis (PCA) [21] and finally gridded to a 2D Cartesian grid without density compensation and used for image reconstruction.

After the images are reconstructed using NLINV, they are post-processed with a modified version of the non-local mean denoising [20] and a temporal median filter [76].

For quantitative comparison, the image content ρ is multiplied with the root of sum of squares (RSS) of estimated coil sensitivity profiles c_j to give the final image ρ_f

$$\rho_f = \rho \sqrt{\sum_{j=1}^{n_c} \left| c_j \right|^2}, \tag{2.44}$$

where n_c is the number of coils chosen after data compression using PCA.

2.8 Principles of Flow MRI

The previous discussions were based on the implicit assumption that the object under investigation is spatially stationary. However, during MRI of human bodies, scenarios related to moving spins are frequently encountered. The imaging techniques for moving spins are called flow MRI. Flow MRI can be divided into two subgroups broadly based on the information obtained from the images. First of them, Magnetic resonance angiography (MRA) methods, provide qualitative information about fluid flow through vessels [93, 139, 5, 35], whereas quantitative flow methods such as PC MRI [94, 7, 95, 19] and Fourier flow imaging [63] are used to obtain quantitative and functional information of many physiological processes. As the quantitative calculations of flow velocities from PC MRI were used for flow studies in the present work, PC MRI is discussed briefly in the next section.

2.8.1 Phase-Contrast MRI

The basic principle of PC MRI was discovered by Hahn [56] in 1960 which states that the velocity of moving spins can be encoded into the phase by the introduction of a bipolar gradient. The position of a moving spin can be expressed as a function of time using Taylor's series

$$x(t) = x(0) + \underbrace{x^{'}(0)}_{v_0} t + \underbrace{x^{''}(0)}_{a_0} \frac{t^2}{2!} + \cdots . \tag{2.45}$$

The temporal evolution of phases due to the movement of spins can be expressed as

$$\begin{aligned}
\phi(t) &= \gamma \int_0^T G(t)x(t)dt \\
&= \gamma \int_0^T G(t)(x_0 + v_0 t + \frac{1}{2}a_0 t^2 + \cdots) \\
&= \gamma[x_0 \int_0^T G(t)dt + v_0 \int_0^T G(t)tdt + \int_0^T G(t)\frac{t^2}{2!}dt + \cdots] \\
&= \gamma[x_0 m_0(T) + v_0 m_1(T) + a_0 m_2(T) + \cdots],
\end{aligned} \tag{2.46}$$

where m_0, m_1, m_2 represent the gradient moments due to static spin at x_0, gradient moment for constant velocity v_0 and constant acceleration a_0 respectively. According

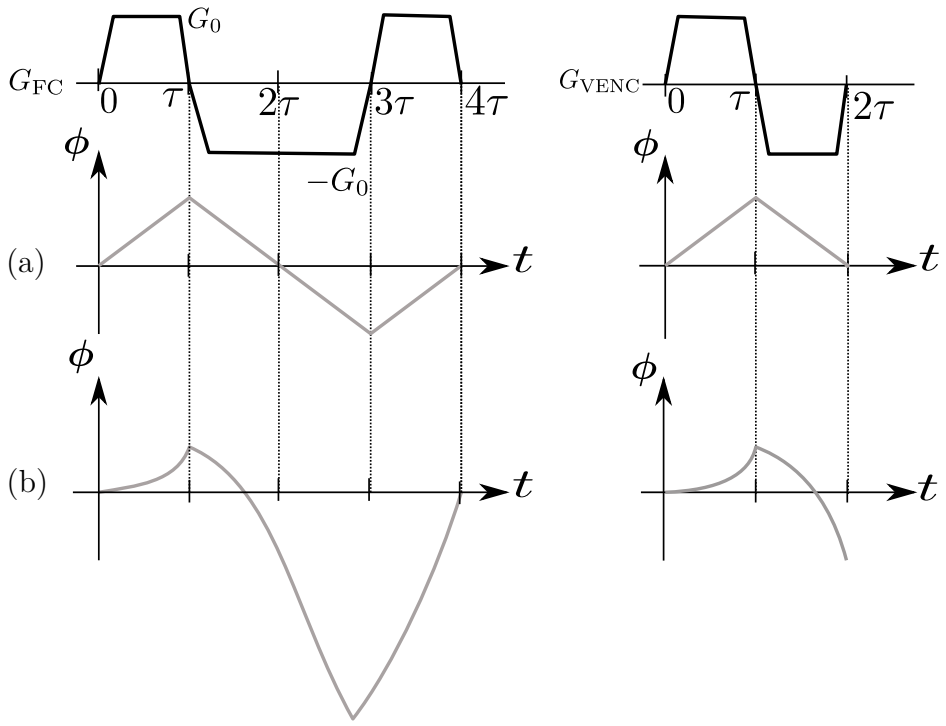

Figure 2.10: Flow compensation and velocity encoding gradient. (Left) Flow compensation (FC) gradient waveform for compensating constant velocity which results in zero-phase for both (a) static as well as (b) moving spins with constant velocity. (Right) Velocity encoding (VENC) gradient waveform results in zero phase for the (a) static spin but a net phase for the (b) moving spins with constant velocity.

to Equation (2.46), static and spins moving with constant velocity will have zero phase by the end of the application of the flow compensation (FC) gradient (G_{FC}) with waveform $1\bar{2}1$ as depicted schematically on the left part of Figure 2.10. On the other hand, application of the bipolar gradient with wave form $1\bar{1}$ (shown in the right part of Figure 2.10) will result in zero phase for static spins and a net phase for moving spins with constant velocities, which is given by,

$$\phi_v(2\tau) = \gamma v_0 \int_0^{2\tau} G(t)tdt \tag{2.47}$$

$$= -\gamma G_0 \tau^2 v_0, \tag{2.48}$$

which is linearly proportional to the velocity and is also determined by the amplitude and duration of the velocity encoding (VENC) gradient.

Although the expression looks straight forward, there are some practical considerations in order to calculate the velocity accurately. Firstly, VENC gradient and the duration must be chosen such that the velocity-encoding range should be larger than

the velocity to be measured (v_0). Otherwise, the accumulated phase exceeds the value π resulting in phase-wrap artifacts. On the other hand, VENC range can not be too large as it results in poor SNR. Without any prior idea of the velocity range, several MRI scans are often necessary to choose a proper VENC.

Secondly, MR images have various sources of phase accumulation e.g. the off-resonance induced phase. Therefore, to remove phases induced by sources other than the velocity, at least two measurements are needed. The measurements are performed in two different ways [71]. The first measurement uses a flow compensation gradient and the second measurement consists of the bipolar velocity gradient resulting in a phase difference $\Delta\phi = -\gamma G_0 \tau^2 v_0$. Subsequently, velocity can be calculated from the phase difference expression [19, 99, 14].

For PC MRI data with multiple coils, firstly the reconstructed images from two consecutive measurements and the coil sensitivities are combined to remove unwanted phase contributions from coils. The weighted image ρ_{ij} for i-th measurement and j-th coil is given by

$$\rho_{ij} = \frac{\rho_i c_{ij}}{\sqrt{\sum_{k=1}^{n_c} c_{ik}\overline{c}_{ik}}}, \quad i = 1, 2 \quad j = 1, 2, \cdots, n_c \tag{2.49}$$

where n_c are the number of chosen coils after PCA. The complex phase difference images ρ_{pc} are calculated by

$$\hat{\rho}_{pc} = \sum_{j=1}^{n_c} \rho_{0j}\overline{\rho}_{1j}, \quad \rho_{pc} = \frac{\hat{\rho}_{pc}}{\sqrt{\hat{\rho}_{pc}}}. \tag{2.50}$$

The complex phase difference map was used further to calculate the pixel-wise velocities.

The PC MRI technique explained above is applicable for one-dimensional flow. However, the same principle can be extended in two or three orthogonal directions to encode multi-dimensional constant velocities [87, 88].

Chapter 3

Simulation of Bloch Equations for Spatially Stationary Objects

Numerical simulators of Bloch equations are essential tools for a variety of different research directions in MRI. Throwing lights on important essential features of MRI, it helps in further methodological developments. For example, numerical simulations have been used previously for the pulse sequence optimizations and designs [3]. MRI pulse sequence optimization goes through several steps of parameter modifications until desired image characteristics are obtained. It is a time consuming process and Bloch equation based numerical simulators are suitable, inexpensive tools for such purposes. Numerical simulations have also been used for artifact detection and elimination [53].

Due to controlled experiments with the input data, numerical simulations can be used effectively to locate the exact sources of the artifacts, i.e. whether the artifacts are generated due to some physical phenomena (motion etc.) or hardware malfunctions [100, 34].

In prior works, MRI simulations have also been used for the design of specialized RF pulses [121]. Image reconstruction techniques can be tested using MRI simulations [121] as well. Apart from the controlled experiments with precise input data, numerical simulators can also be used to simulate various limiting experimental conditions which are either improbable or difficult to reproduce in experiments. Because of this advantage, it is a very useful pedagogical and educational tool [15]. Multiple utilities of numerical simulations have been combined to produce a few general purpose MRI simulators also [140, 12, 121].

3.1 Bloch Equation for Spatially Stationary Object

As already briefly discussed in Chapter 2, Bloch equations describe the time-dependent behavior of magnetizations in presence of an external magnetic field. Also, when expressed in a proper rotating frame Bloch equations explain MRI phenomena better conceptually. Furthermore, Bloch equations for MRI experiments contain time-dependent magnetic gradient $\mathbf{G}(t)$ fields, RF excitation fields $\mathbf{B}_{xy} = \begin{bmatrix} B_x(t) & B_y(t) & 0 \end{bmatrix}^T$ with time dependent amplitudes and phases, magnetic field inhomogeneities ΔB. In its most general form, Bloch equation in a rotating frame is expressed as follows:

$$\frac{\mathrm{d}\mathbf{M}'}{\mathrm{d}t} = \gamma \mathbf{M}' \times \mathbf{B}_{\text{eff}} + \frac{(M_0 - M_z)\hat{e}_z}{T_1} - \frac{M_{x'}\hat{e}_{x'} + M_{y'}\hat{e}_{y'}}{T_2} \tag{3.1a}$$

$$= \begin{bmatrix} -\frac{1}{T_2} & \gamma B_z & -\gamma B_y \\ -\gamma B_z & -\frac{1}{T_2} & \gamma B_x \\ \gamma B_y & -\gamma B_x & -\frac{1}{T_1} \end{bmatrix} \begin{pmatrix} M_{x'} \\ M_{y'} \\ M_z \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \frac{M_0}{T_1} \end{pmatrix}, \tag{3.1b}$$

where

$$\mathbf{B}_{\text{eff}} = B_x(t)\hat{e}_{x'} + B_y(t)\hat{e}_{y'} + (B_0 - \frac{\omega_{\text{rf}}}{\gamma} + \mathbf{G}(t) \cdot \mathbf{r} + \Delta B)\hat{e}_z.$$

Equation (3.1) governs the MRI experiments for spatially stationary objects.

In an analytical setting, Equation (3.1) can be written as a linear initial value problem (IVP),

$$\dot{\mathbf{w}} = A(t)\mathbf{w} + \mathbf{g}(t), \quad t > 0, \quad \mathbf{w}(0) = \mathbf{w}_0, \tag{3.2}$$

where

$$A(t) \in \mathbb{R}^{n \times n}, \quad \mathbf{w}, \mathbf{g} \in \mathbb{R}^n.$$

Equation (3.2) is an inhomogeneous ordinary differential equation (ODE) with a forcing term $\mathbf{g}(t)$. The following theorem gives the condition for the existence of a unique solution for an ODE with the general form of Equation (3.2).

Theorem 3.1 (Existence, Uniqueness and Extension)**.** *If $A(t) \in \mathbb{R}^{n\times n}$ is a continuous matrix and $\mathbf{g}(t) \in \mathbb{R}^n$ is a continuous vector, then the linear initial value problem*

$$\dot{\mathbf{w}} = A(t)\mathbf{w} + \mathbf{g}(t), \qquad \mathbf{w}(t_0) = \mathbf{w}_0, \tag{3.3}$$

has a solution for every $(t_0, \mathbf{w}_0) \in \mathbb{R} \times \mathbb{R}^n$ and the solution can be extended to all times $t \in \mathbb{R}$.

Proof. The proof of this theorem can be found in [125, 116]. □

In case of MR experiments the function $A(t)$ is a piecewise constant function and $\mathbf{g}(t)$ is a piecewise constant vector. Then, Theorem 3.1 is applicable to all intervals where $A(t)$ and $\mathbf{g}(t)$ are piecewise constant and the solution can be continuously extended to $t \in \mathbb{R}$ [144].

Immediately after the discovery of MR phenomena researchers tried to find analytical solutions of Bloch equations. If the RF excitation field amplitude, phases and magnetic field gradients are constant, the Bloch equations are a system of linear ODE with constant coefficients. The first analytical solution of such equations was given by Torrey (1949) by adopting Laplace transform method [129]. Mulkern et al. [96] and Murase et al. [97] used this method further for simulating different MRI experiments.

Analytical solutions of Bloch equations are given in literature with various approximation and limiting situations such as steady-state solutions [1, 54], for short-lived RF pulse approximation [54], very weak RF fields [118]. An approximate analytical solution is also proposed for a rectangular RF pulse in [83]. Analytical solutions in these limiting situations give us more physical insight about the MR phenomena. However, no closed form analytical solutions are available for Bloch equations for MR experiments. Hence, a suitable numerical method must be chosen for a realistic simulation of MR experiments. The choice of a proper numerical strategy will be discussed in the next section.

3.2 Numerical Strategies

Prior studies have shown that for accurate numerical simulations of MR experiments, Equation (3.1a) must be solved for a huge number of mutually independent isochromats [115, 114]; an isochromat is a microscopic group of spins which resonate at the same frequency.

Moreover, many previous discussions [83, 115, 114] related to the solution of the Bloch equations for MRI are based on the assumptions that the RF pulses are short-

lived, i.e. they take a small fraction of TR. Because of the assumption of short durations of the RF pulses, relaxations of magnetizations are ignored during RF excitation. The assumption of short-lived RF pulses are reasonable as long as the durations of RF pulses are negligible compared to the TR and TE of a repetition. However, the present study aims at MRI experiments where the duration of the shortest RF pulse is generally 400 µs and the TR can be as small as 2-4 ms. Therefore, RF pulses occupy 10-20 % of the TR. Pulse sequences with same TR are also repeated multiple times during an MR experiment, which means that RF pulses occupy a considerable percentage of time during MR experiments and ignoring relaxations during RF excitations is a significant deviation from the experimental conditions. Therefore, in the present thesis, the simulations are performed without the simplified assumption of a short-lived RF pulse.

In addition, the longitudinal magnetizations are often assumed to be flipped uniformly over the slice thickness. Results from numerical simulations show in Section 3.4 that RF pulses do not flip the longitudinal magnetizations uniformly over the slice thickness. As a consequence, the resulting transverse magnetization due to RF excitation and subsequent time evolution are non-uniform as well. Furthermore, the effect of non-uniform slice profiles are also added cumulatively over multiple repetitions of the pulse sequences. Exact slice profiles are always taken into consideration for simulation in this work.

Apart from that, magnetic gradients and RF excitation pulse for real time MRI are step functions of very small time-steps which are known as raster times. Raster time can be as small as 5 µs. As a result, simulation must be carried out for a high number of time-steps if the exact pulse sequence data is to be considered.

To sum up, accurate numerical simulations of MRI require simulating Bloch equations for an ensemble of isochromats over a high number of small time steps which is a very demanding task computationally. Therefore, computational time is also a matter of concern for choosing a proper numerical strategy for Bloch equation-based MRI simulators together with numerical accuracy.

For solving an ODE system such as Equation (3.2), multiple numerical methods have been proposed and are widely available in the literature. A detailed analysis of these methods can be found in standard textbooks for numerical solution of an ODE [58, 69, 22]. In a previous study, Grivet et al. used one of these standard numerical techniques, explicit 4-5th order adaptive Runge-Kutta (RK45) for Bloch equations simulations [51]. Explicit RK45 methods are discussed and are also tested for performance and numerical accuracy in Section 3.4.

In some previous studies, MRI simulations were carried out splitting Equation (3.1a) into two subproblems. The general technique of solving a complex time-dependent problem by splitting into simpler subproblems are known as operator splitting method. In this chapter, operator splitting methods for solving a system of ODE [68, 47] are discussed. Higher order splitting methods have been proposed for solving Equation (3.1a) as well. Results related to the accuracy and performances of the operator splitting methods are discussed in Section 3.4.

The basic definitions and results for analysis of the numerical methods for an ODE are given in Appendix A.

3.2.1 Explicit Runge-Kutta Method

This section discusses Runge-Kutta (RK) methods very briefly. More detailed discussions can be found in [58, 69, 22].

Explicit RK methods are single-step multi-stage techniques where the solutions at $t = t_{n+1}$ is determined from the solutions at $t = t_n$. General s-stage RK methods are written as follows:

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \tau \sum_{i=1}^{s} b_i \mathbf{k}_i, \quad \tau = t_{n+1} - t_n, \tag{3.4a}$$

where

$$\mathbf{k}_i = \mathbf{f}(t_n + c_i\tau, \mathbf{w}_n + \tau \sum_{j=1}^{s} a_{i,j}\mathbf{k}_j), \quad \mathbf{f}(t, \mathbf{w}) = A(t)\mathbf{w} + \mathbf{g}(t). \tag{3.4b}$$

For explicit RK methods $a_{i,j} = 0, \forall j \geq i$. The matrix $[a_{ij}]_{i,j=1}^{s}$ and the coefficients $b_i, c_i,\ i = 1, 2, \cdots s$ are generally arranged in a table, known as *Butcher tableau*. For explicit RK methods, *Butcher tableau* is given by

$$\begin{array}{c|ccccc} 0 & & & & & \\ c_2 & a_{21} & & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ c_s & a_{s1} & a_{s2} & \cdots & a_{s,s-1} & \\ \hline & b_1 & b_2 & \cdots & b_{s-1} & b_s \end{array} . \tag{3.5}$$

The RK methods are consistent if

$$\sum_{j=1}^{s} b_j = 1. \tag{3.6}$$

To achieve a specific order of consistency p certain number of stages s of the explicit Runge-Kutta method must be calculated where $s \geq p$.

Embedded RK methods are used very often to control the local truncation errors with adaptive step size as will be discussed next.

Embedded Runge-Kutta

In embedded RK methods, two methods of order p and $p-1$ are calculated and the difference between them is taken to be the error estimator ($\boldsymbol{\epsilon}_{n+1}$) of the solution.

Let us assume the solution at $t = t_{n+1}$, using p-th order method is denoted by Equation (3.4a) and the solution by $p-1$-th order method is given by

$$\mathbf{w}^*_{n+1} = \mathbf{w}_n + \tau \sum_{i=1}^{s} b^*_i \mathbf{k}_i. \tag{3.7}$$

The estimated error $\boldsymbol{\epsilon}_{n+1}$, which is of the order p and given by

$$\boldsymbol{\epsilon}_{n+1} = \mathbf{w}^*_{n+1} - \mathbf{w}_{n+1} = \tau \sum_{i=1}^{s} (b_i - b^*_i)\mathbf{k}_i. \tag{3.8}$$

The estimated error $\boldsymbol{\epsilon}_{n+1}$ is used to control the time step size adaptively and thus to ensure desired accuracy.

The *Butcher tableau* for embedded RK methods are extended with b^*_i values and given by

$$\begin{array}{c|ccccc} 0 & & & & & \\ c_2 & a_{21} & & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ c_s & a_{s1} & a_{s2} & \cdots & a_{s,s-1} & \\ \hline & b_1 & b_2 & \cdots & b_{s-1} & b_s \\ & b^*_1 & b^*_2 & \cdots & b^*_{s-1} & b^*_s \end{array}. \tag{3.9}$$

Popular adaptive RK45 methods are Cash-Karp method [25], Dormand–Prince method [32], Runge–Kutta–Fehlberg method [39, 40]. Cash-Karp RK45 has been used for testing accuracy and performance in this work, discussed in Section 3.4.

3.2.2 Operator Splitting

Operator splitting is a powerful tool to decompose a complex time-dependent problem into simpler sub-problems which can be solved efficiently or more accurately with suitable analytical or numerical methods. The solution of different sub-processes are then combined by coupling the subprocess through their initial conditions.

Operator splitting are very often distinguished between two different cases. The first case is known as the differential splitting where a partial differential equation (PDE) system, describing the original multi-scale, multi-physics problem, is split into different sub-processes; operators for each sub-process usually represents a different physical phenomena (e.g., advection, diffusion, reaction etc.). The numerical strategy chosen for solving Bloch equations for flowing spins is based on differential splitting, discussed in Chapter 4.

For the second category, splitting is applied to an ODE system which is very often the semi-discretized form of a partial differential equation (PDE) system. This operator splitting technique is known as algebraic splitting. In this section, operator splitting methods are described in an analytical setting in the context of an ODE system, beginning with a linear ODE system to a general class of ODE system. The purpose here is to propose an efficient solution methodology for solving Equation (3.1a).

For detail discussion on operator splitting and its applications, the reader is suggested to consult [38, 68, 57].

Operator Splitting for Linear ODE

As an illustration of operator splitting for a linear ODE system, consider a homogeneous IVP with constant coefficient matrix $A \in \mathbb{R}^{n\times n}$ given by

$$\dot{\mathbf{w}} = A\mathbf{w}, \quad \mathbf{w}(t_0) = \mathbf{w}_0, \tag{3.10}$$

where

$$t \in \mathbb{J} = [t_0\ T] \subset \mathbb{R}, \quad \mathbf{w} : \mathbb{J} \to \mathbb{R}^n.$$

The solution of Equation (3.10) at $t = t_{n+1}$ in terms of the solution at $t = t_n$:

$$\mathbf{w}(t_{n+1}) = \mathrm{e}^{\tau A}\mathbf{w}(t_n), \quad \tau = t_{n+1} - t_n. \tag{3.11}$$

Generally calculating the exponential of the matrix at each time step, is tedious and computationally expensive. In many cases, A is split in matrices which can be solved analytically. The method is illustrated here by splitting A into two matrices

$$A = A_1 + A_2. \tag{3.12}$$

Splitting of A into more than two matrices is a straightforward extension. After splitting A, the following two subproblems are solved sequentially one after another instead of the original problem,

$$\frac{\mathrm{d}\mathbf{w}^*(t)}{\mathrm{d}t} = A_1\mathbf{w}^*(t), \quad t_n < t \leq t_{n+1}, \quad \mathbf{w}^*(t_n) = \mathbf{w}_n, \tag{3.13a}$$

$$\frac{\mathrm{d}\mathbf{w}^{**}(t)}{\mathrm{d}t} = A_2\mathbf{w}^{**}(t), \quad t_n < t \leq t_{n+1}, \quad \mathbf{w}^{**}(t_n) = \mathbf{w}^*_{n+1}. \tag{3.13b}$$

This splitting method is known as *sequential splitting*. Solving Equations (3.13a) and (3.13b), an approximate solution is obtained by putting $\mathbf{w}_{n+1} = \mathbf{w}^{**}_{n+1}$

$$\mathbf{w}_{n+1} = \mathrm{e}^{\tau A_2}\mathrm{e}^{\tau A_1}\mathbf{w}_n + \tau\boldsymbol{\epsilon}_t, \tag{3.14}$$

where $\boldsymbol{\epsilon}_t$ is the local truncation error of the splitting method. Let us assume that Equations (3.13a) and (3.13b) are solved exactly. Solving Equation (3.10) by splitting A into A_1 and A_2 introduces a splitting error $\boldsymbol{\epsilon}_s$ which can be calculated by a series expansion for the matrix exponential.

$$\mathrm{e}^{\tau A} = I + \tau(A_1 + A_2) + \frac{\tau^2({A_1}^2 + {A_2}^2)}{2!} + O\big(\tau^3\big), \tag{3.15a}$$

$$\mathrm{e}^{\tau A_2}\mathrm{e}^{\tau A_1} = I + \tau(A_2 + A_1) + \frac{\tau^2({A_1}^2 + 2A_2A_1 + {A_2}^2)}{2!} + O\big(\tau^3\big). \tag{3.15b}$$

Subtracting Equation (3.15b) from Equation (3.15a), the local splitting error $\boldsymbol{\epsilon}_s$ for one time step can be calculated as

$$\boldsymbol{\epsilon}_s = (\mathrm{e}^{\tau A} - \mathrm{e}^{\tau A_2}\mathrm{e}^{\tau A_1})\mathbf{w}(t_n) = \frac{1}{2}\tau^2[A_1, A_2]\mathbf{w}(t_n) + O\big(\tau^3\big), \tag{3.16}$$

where $[A_1, A_2]$ is called commutator and defined as:

$$[A_1, A_2] = A_1 A_2 - A_2 A_1. \tag{3.17}$$

We can also see from Equation (3.16) and Equation (3.14) that the order of $\boldsymbol{\epsilon}_t$ of a splitting method is one less than the order of $\boldsymbol{\epsilon}_s$. Also, Equation (3.16) implies that the order of accuracy and splitting error of a splitting method depends on $[A_1, A_2]$. The Baker-Campbell-Hausdorff formula [68] shows that $\boldsymbol{\epsilon}_s$ vanishes completely if $[A_1, A_2]$ vanishes. It expresses the product of two exponentials as a new exponential, given by

$$\mathrm{e}^{\tau A_2}\mathrm{e}^{\tau A_1} = \mathrm{e}^{\tau \tilde{A}}, \tag{3.18a}$$

where

$$\begin{aligned}\tilde{A} = (A_1 + A_2) + \frac{1}{2}\tau[A_1, A_2] + \frac{1}{12}\tau^2([A_2, [A_1, A_2]] + [A_1, [A_1, A_2]]) \\ + \frac{1}{24}\tau^3([A_2, [A_1, [A_1, A_2]]]) + O\big(\tau^4\big).\end{aligned} \tag{3.18b}$$

According to Equation (3.18b), if A_1 and A_2 commutes, all the higher order splitting terms vanish. However, in general, sequential splitting is only first order accurate. Some splitting methods, discussed in Section 3.2.2, give us higher order of accuracy.

Operator Splitting for General Class of Initial value problem

Splitting methods can be extended to a general class of ODE system such as Equation (3.1a). Let us consider a general ODE system

$$\frac{\mathrm{d}\mathbf{w}}{\mathrm{d}t} = \mathbf{F}(t, \mathbf{w}(t)) \quad \text{where } t \in [t_0\ T], \quad \mathbf{w}(t_0) = \mathbf{w}_0. \tag{3.19}$$

Let us further assume, the source term of Equation (3.19) is split into two terms

$$\mathbf{F}(t, \mathbf{w}) = \mathbf{F}_1(t, \mathbf{w}) + \mathbf{F}_2(t, \mathbf{w}). \tag{3.20}$$

In order to solve Equation (3.19) by splitting technique, we need to solve the following sub-problems as before,

$$\frac{\mathrm{d}\mathbf{w}^*(t)}{\mathrm{d}t} = \mathbf{F}_1(t, \mathbf{w}^*(t)), \quad t_n < t \leq t_{n+1}, \quad \mathbf{w}^*(t_n) = \mathbf{w}(t_n), \tag{3.21a}$$

$$\frac{\mathrm{d}\mathbf{w}^{**}(t)}{\mathrm{d}t} = \mathbf{F}_2(t, \mathbf{w}^{**}(t)) \quad t_n < t \leq t_{n+1}, \quad \mathbf{w}^{**}(t_n) = \mathbf{w}^*(t_{n+1}). \tag{3.21b}$$

Setting $X(t_{n+1}) = X^{**}(t_{n+1})$ gives us the approximate solution at $t = t_{n+1}$. Using Taylor's expansions of $X^{**}(t_{n+1})$ and $X^{*}(t_n)$ around $t = t_n$ splitting error $\boldsymbol{\epsilon}_s$ for n-th time step can be calculated as

$$\boldsymbol{\epsilon}_s = \frac{1}{2}\tau^2[\frac{\partial \mathbf{F}_1}{\partial \mathbf{w}}\,\mathbf{F}_2 - \frac{\partial \mathbf{F}_2}{\partial \mathbf{w}}\,\mathbf{F}_1] + O\Big(\tau^3\Big). \tag{3.22}$$

The commutator for a general ODE system can be defined as

$$[\mathbf{F}_1, \mathbf{F}_2](\mathbf{w}) = [\frac{\partial \mathbf{F}_1}{\partial \mathbf{w}}\,\mathbf{F}_2 - \frac{\partial \mathbf{F}_2}{\partial \mathbf{w}}\,\mathbf{F}_1]. \tag{3.23}$$

If the commutator vanishes then the splitting error vanishes completely like a linear ODE system as stated by Theorem 3.2.

Theorem 3.2. *If for the IVP Equation* (3.19) *the commutator vanishes, i.e., Equation* (3.23) *is zero then the splitting error* $\boldsymbol{\epsilon}_s$ *also completely vanishes.*

Proof. The proof of this theorem can be found in [68]. □

Higher Order Splitting Methods

In the previous sections, the sequential operator splitting method and its properties have been discussed. Higher-order operator-splitting methods can be constructed with a slight increase in the computational load. The method of symmetrically weighted sequential operator splitting (SWSS) is one of the ways to achieve a higher order accuracy. In this method, the differential equation (DE) is solved sequentially in a specific order and in its reverse order in every time-step. The average of the solutions by these two calculations are taken as the approximate solution of the DE. For example, for the linear ODE system, Equation (3.13a) and then Equation (3.13b) are solved first and Equation (3.13b) and then Equation (3.13a) are solved next. The average of these two solutions are taken as the final solution which is given by

$$\mathbf{w}_{n+1} = \frac{1}{2}(\mathrm{e}^{\tau A_2}\mathrm{e}^{\tau A_1} + \mathrm{e}^{\tau A_1}\mathrm{e}^{\tau A_2})\mathbf{w}_n. \tag{3.24}$$

From the series expansion of the matrix exponentials, one obtains the splitting error for SWSS:

$$\boldsymbol{\epsilon}_s(\tau) = -\frac{1}{12}\tau^3([A_1, [A_1, A_2]] + [A_2, [A_1, A_2]])\mathbf{w}(t_n) + O\Big(\tau^4\Big). \tag{3.25}$$

For the general ODE system we get the final solution following the same procedure, given by

$$\mathbf{w}_{n+1} = \frac{1}{2}(\mathrm{e}^{\tau\mathcal{F}_1}\mathrm{e}^{\tau\mathcal{F}_2} + \mathrm{e}^{\tau\mathcal{F}_2}\mathrm{e}^{\tau\mathcal{F}_1})\mathbf{w}_n, \tag{3.26}$$

and the splitting error is given by,

$$\boldsymbol{\epsilon}_s(\tau) = -\frac{1}{12}\tau^3([\mathbf{F}_1,[\mathbf{F}_1,\mathbf{F}_2]] + [\mathbf{F}_2,[\mathbf{F}_1,\mathbf{F}_2]])\mathbf{w}(t_n) + O\Big(\tau^4\Big). \tag{3.27}$$

Strang(1968) [122] proposed another higher order splitting method, known as *Strang splitting method* or sometimes *Strang-Marchuk splitting method*. In this method, the following subproblems have to be solved for Equation (3.19) if we split the right hand side of the equation in two terms.

$$\frac{\mathrm{d}\mathbf{w}^*(t)}{\mathrm{d}t} = \mathbf{F_1}(t,\mathbf{w}^*(t)), \quad t_n < t \le t_{n+\frac{1}{2}}, \quad \mathbf{w}^*(t_n) = \mathbf{w}(t_n), \tag{3.28a}$$

$$\frac{\mathrm{d}\mathbf{w}^{**}(t)}{\mathrm{d}t} = \mathbf{F_2}(t,\mathbf{w}^{**}(t)) \quad t_n < t \le t_{n+1}, \quad \mathbf{w}^{**}(t_n) = \mathbf{w}^*(t_{n+\frac{1}{2}}), \tag{3.28b}$$

$$\frac{\mathrm{d}\mathbf{w}^{***}(t)}{\mathrm{d}t} = \mathbf{F_1}(t,\mathbf{w}^{***}(t)) \quad t_{n+\frac{1}{2}} < t \le t_{n+1}, \quad \mathbf{w}^{***}(t_n) = \mathbf{w}^{**}(t_{n+1}). \tag{3.28c}$$

The solution obtained using this method:

$$\mathbf{w}_{n+1} = (\mathrm{e}^{\frac{\tau}{2}\mathcal{F}_1}\mathrm{e}^{\tau\mathcal{F}_2}\mathrm{e}^{\frac{\tau}{2}\mathcal{F}_1})\mathbf{w}_n. \tag{3.29}$$

This method gives us the following approximate solution for linear ODE system Splitting error $\boldsymbol{\epsilon}_s$ in this case:

$$\boldsymbol{\epsilon}_s(\tau) = \frac{1}{24}\tau^3([\mathbf{F}_1,[\mathbf{F}_1,\mathbf{F}_2]] + 2[\mathbf{F}_2,[\mathbf{F}_1,\mathbf{F}_2]])\mathbf{w}(t_n) + O\Big(\tau^5\Big). \tag{3.30}$$

Splitting Method for Bloch Equations

The generalized Bloch equation for MRI, given by Equation (3.1a), combines two physical phenomena,

(i) Rotation due to the RF pulse and precession of magnetizations due to the magnetic gradients and field inhomogeneities, represented by a linear system.

(ii) T_1 and T_2 time relaxations of magnetizations given by an inhomogeneous linear system.

Equation (3.1a) is solved at each time step, splitting the equation into the two operators mentioned above.

For $t_n < t \leq t_{n+1}$,

$$\frac{\mathrm{d}\mathbf{M}^*}{\mathrm{d}t} = \underbrace{\gamma \mathbf{M}^* \times \mathbf{B}_{\text{eff}}}_{\text{rotation}} = \underbrace{\begin{bmatrix} 0 & \gamma B_z & -\gamma B_y \\ -\gamma B_z & 0 & \gamma B_x \\ \gamma B_y & -\gamma B_x & 0 \end{bmatrix}}_{\text{rotation operator}(A_{\text{rot}})} \begin{pmatrix} M_x^* \\ M_y^* \\ M_z^* \end{pmatrix}, \quad \mathbf{M}^*(t_n) = \mathbf{M}'(t_n), \tag{3.31a}$$

$$\frac{\mathrm{d}\mathbf{M}^{**}}{\mathrm{d}t} = \underbrace{\begin{pmatrix} -\frac{M_x^{**}}{T_2} \\ -\frac{M_y^{**}}{T_2} \\ \frac{M_0 - M_z^{**}}{T_1} \end{pmatrix}}_{\text{time relaxation}}, \quad \mathbf{M}^{**}(t_n) = \mathbf{M}^*(t_{n+1}). \tag{3.31b}$$

Both of these sub-problems can be solved analytically. The solution of Equation (3.31a):

$$\mathbf{M}^*(t_{n+1}) = \mathrm{e}^{\tau A_{\text{rot}}} \mathbf{M}'(t_n) = R\mathbf{M}'(t_n). \tag{3.32}$$

The details of the calculation of R is discussed in Section A.1. The Equation (3.31b) can also be solved analytically and that gives us the solution of magnetizations step-by-step for sequential splitting.

$$\mathbf{M}'(t_{n+1}) = \mathbf{M}^{**}(t_{n+1}) = \begin{pmatrix} \mathrm{e}^{-\frac{\tau}{T_2}} M_x^*(t_{n+1}) \\ \mathrm{e}^{-\frac{\tau}{T_2}} M_y^*(t_{n+1}) \\ \mathrm{e}^{-\frac{\tau}{T_1}} M_z^*(t_{n+1}) + +M_0(1 - \mathrm{e}^{-\frac{\tau}{T_1}}) \end{pmatrix}. \tag{3.33}$$

The splitting error for sequential splitting can be estimated from Equations (3.22)–(3.23):

$$\boldsymbol{\epsilon}_s(\tau) = \frac{1}{2}\tau^2((DA_{\text{rot}} - A_{\text{rot}}D)\mathbf{M}' + \begin{pmatrix} \gamma B_y \frac{M_0}{T_1} \\ -\gamma B_x \frac{M_0}{T_1} \\ 0 \end{pmatrix}) + O\big(\tau^3\big), \tag{3.34a}$$

$$= \frac{1}{2}\tau^2((\frac{1}{T_2} - \frac{1}{T_1}) \begin{pmatrix} \gamma B_y M_z \\ -\gamma B_x M_z \\ \gamma B_y M_{x'} - \gamma B_x M_{y'} \end{pmatrix} + \begin{pmatrix} \gamma B_y \frac{M_0}{T_1} \\ -\gamma B_x \frac{M_0}{T_1} \\ 0 \end{pmatrix}) + O\big(\tau^3\big), \tag{3.34b}$$

where $D = \mathrm{diag}(-\frac{1}{T_2}, -\frac{1}{T_2}, -\frac{1}{T_1})$.

From Equation (3.34), it can be observed that the splitting error is minimum when T_2 is close to T_1. Furthermore, splitting error is **0** after the RF excitation as $B_x, B_y = 0$ in post-excitation period of TR.

The splitting error can be minimized even more using higher order splitting methods as discussed in Section A.1.

3.3 Bloch Equation Simulator

In the present work, the Bloch equations simulator focuses on the radial FLASH sequences. The computational kernel simulates the time evolution of magnetization for time-dependent RF and pulse sequence data for an ensemble of isochromats and it is based on the operator splitting method, described in the previous section.

The simulator takes as input exact pulse sequence data and a model of the object to be simulated as computational domain. The summed up complex signals $M_{xy} = M_x + iM_y$ from an ensemble of isochromats, recorded at some specified time points, are the output of the simulations. The number of output sample points can be freely chosen and it is adapted to the specific situation.

The computational kernel simulates the time evolution of magnetizations for an ensemble of isochromats. As the isochromats are mutually independent, the computational load can be distributed and parallelized. A parallel solver is also developed using GPU. The numerical implementation in GPU will be discussed in Section 3.3.4.

3.3.1 Computational Model

A schematic diagram of the computational domain is given in Figure 3.1. It shows that the simulation is generally carried out for a number of voxels. Each of these voxels are subdivided into a number of subvoxels. Each subvoxel represents an ensemble of isochromats. The aim of the present work is to calculate step-by-step all components of the magnetizations for all the isochromats in the ensemble which have specific and pre-determined MR properties.

Isochromats are characterized by the following MR parameters of the specific object to be simulated: T_1, T_2, off-resonance frequency ω due to field inhomogeneity and the location of the isochromats relative to isocenter (r_x, r_y, r_z). The off-resonance frequencies of the isochromats due to magnetic gradients can be calculated from the relative position from the isocenter. The isochromats along with their characterizing

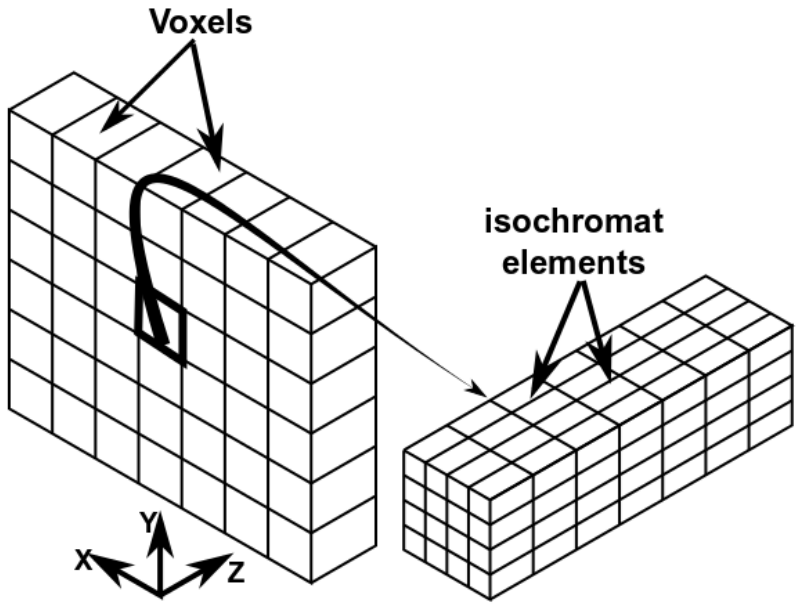

Figure 3.1: Schematic diagram of computational domain.

properties are taken as the computational model for the simulation. Computational models are created using MATLAB (MathWorks, Natwick, NA).

3.3.2 Pulse sequence

The exact pulse sequence is the second set of input data, which consists of the following components:

(i) RF excitation pulse $\begin{bmatrix} B_x & B_y & 0 \end{bmatrix}^T$

(ii) Magnetic field gradients in slice direction G_z, read-directions G_x, G_y.

(iii) A binary vector specifying the signal-acquisition time points.

In order to acquire the pulse sequence data, MR experimental protocols are taken as input in an MR vendor provided software framework to simulate the sequence diagram and acquire the sequence parameters. The acquired pulse sequence data is further used to generate the input binary file according to the raster times. For this thesis, as the experiments were conducted in Siemens Prisma system (Section 5.1), pulse sequence data were generated using the Siemens provided IDEA (Siemens AG, Erlangen, Germany) framework.

The complex signals are acquired during simulation either at the signal-acquisition time points specified by experimental protocols of MRI experiments or at TE of each

repetition. At the specified output time points during an experiment, the receiver coils become active and the analog-to-digital converter (ADC) converts the analog signal to digital for image reconstruction. However, the simulated results at time points, specified by the experimental protocol, needs to be recorded when the energy of signals are compared. In general, the simulated data are recorded at TE of each repetition for comparison.

3.3.3 Implementation

The numerical implementation procedure of the simulator is described in Algorithm 1. The MRI simulator is developed in C++ due to its multiple features, availability of a large number libraries, parallel computing interfaces and the high level of performance requirements of the problem. The serial version of the code relies on C++ Linear algebra libraries Eigen [52]. Eigen makes the reading, writing and transfers of the data simpler and also provides some readily available optimized linear algebra operations. The computationally expensive part is implemented as parallel program using CUDA-C. The details of parallel implementation are discussed in Section 3.3.4.

As described in line 1 of Algorithm 1, the following parameters are passed as inputs in the simulator:

(i) Pulse sequence data as a function of time consisting of RF pulse magnitude $\max(B_1)$, maximum read-out magnetic gradient $\max(G)$, slice selective gradient G_z, the duration of different time-steps τ, a vector ADC specifying the output time points.

(ii) Computational model, i.e. the ensemble of isochromats with their positions (r_x, r_y, r_z) and MR properties ω, T_1, T_2.

(iii) Additional parameters related to an MRI experiment, e.g. total number of repetitions(nEx) which specifies the total duration of numerical simulation, number of spokes $nspokes$ used for imaging, number of turns $nTurn$, the MR sequence type $seqtype$.

Some other parameters are also specified that gives the simulator greater flexibility. However, they are not mentioned here as they are not essential for a minimal working of the simulator and illustration purposes.

After the simulator reads the input data (line 1 in Algorithm 1), the initial magnetizations $\mathbf{m}_0$ of the isochromats are set (line 4 in Algorithm 1). The equilibrium magnetizations M_0 are assumed be 1. For the sequences considered in this work, in the

Algorithm 1 Simulating Bloch Equation for a Spatially Static Object

1: **Inputs:**
   $B_1, G, G_z, \tau, ADC \leftarrow$ Readfile($PulseSeqeunceData$)
   $isochromats = r_x, r_y, r_z, \omega, T_1, T_2 \leftarrow$ Readfile($IsochromatEnsemble$)
   $nEx, nSpoke, nTurn, baseresol, seqtype \leftarrow$
   Readfile($SequenceSpecifier$)
2:
3: **procedure** BLOCHSIMULATION
4: **Initialize:**
   $M_x^p(0), M_y^p(0), M_z^p(0) \leftarrow$ InitializeMagnetization($\mathbf{m}_0$), $p =$
   $1, 2, ...r_x.length$
5: **for** $i \in$ nEx **do**
6: $\theta_{\text{rf}} \leftarrow$ CalculateRFOrientation($seqtype$)
7: $\theta_{\text{s}} \leftarrow$ CalculateSpokeOrientation($nTurn, nSpoke$)
8: $B_x = B_1 \cdot \cos(\theta_{\text{rf}}), B_y = B_1 \cdot \sin(\theta_{\text{rf}})$
9: $G_x = G \cdot \cos(\theta_s), G_y = G \cdot \sin(\theta_s)$
10: **for** $j \in (G.length - 1)$ **do**
11: $pulsedata \leftarrow B_x(j), B_y(j), G_x(j), G_y(j), G_z(j), ADC(j)$
12: **for** $k \in r_x.length$ **do**
13: $isochromat \leftarrow r_x(k), r_y(k), r_z(k), \omega(k), T_1(k), T_2(k)$
14: $M_x^k(j+1), M_y^k(j+1), M_z^k(j+1) \leftarrow$ CalculateMagnetization($isochromat$
15: $, pulsedata, M_x^k(j), M_y^k(j), M_z^k(j))$
16: **if** $ADC(j+1) = 1$ **then**
17: $M_{xy}(i,c) \leftarrow$ CalculateSignal($M_x^p(j+1), M_y^p(j+1)$), $p = 1, 2, \cdots r_x.length, c = 1, 2, \cdots, baseresol$
18: **return** M_{xy}

beginning of a MR experiment the magnetizations along the transverse directions are assumed to be 0 and the longitudinal magnetizations are assumed to be equal to the M_0. Therefore, the initial condition of all the isochromats are given by $\mathbf{m}_0 = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T$.

After setting the initial conditions, the time evolution of the magnetizations of all the isochromats are calculated from repetition to repetition and through all the specified time points step-by-step (lines 5–16 in Algorithm 1) in the following order:

(i) For every repetition, orientation of the RF axis θ_{rf} and the spokes θ_s are calculated (lines 6–7 in Algorithm 1).

(ii) B_x, B_y, G_x, G_y are calculated for every repetition (lines 8–9 in Algorithm 1).

(iii) Pulse sequence data $B_x, B_y, G_x, G_y, G_z, ADC$ for a specific repetition are used for a step-by-step solutions for all the isochromats (lines 10–16 in Algorithm 1).

(iv) Integrated complex signals are recorded at time points specified by the ADC vector (Lines 16–17 in Algorithm 1).

3.3.4 Parallel Computing

Algorithm 2 Parallel Subroutine for Calculating Magnetizations in GPU

1: **procedure** MAGNETIZATIONCALCULATIONSTATIC($isochromats, data$)
2: $isochromatGPU \leftarrow isochromats, pulsedataGPU \leftarrow pulsedata$
3: Setup CalculateMagnetizationKernel execution configuration: $grid_1, block_1$
4: Setup CalculateSignalKernel execution configuration: $grid_2, block_2$
5: CalculateMagnetizationKernel $<<< grid_1, block_1 >>>$ $(M_x(j), M_y(j), M_z(j),$
6: $isochromatGPU, pulsedataGPU), M_x(j+1), M_y(j+1), M_z(j+1))$
7: **if** $ADC(j+1) = 1$ **then**
8: $CalculateSignalKernel <<< grid_2, block_2 >>>$ $(M_x(j+1), M_y(j+1),$
9: $M_{xy}(j+1))$
10: **return** M_{xy}

All operations in the loops specified in lines 12–14 in Algorithm 1 and the procedure specified in line 16 are mutually independent. Therefore, it is possible to parallelize them. These parts from Algorithm 1 have been implemented in CUDA-C as a GPU-based parallel procedures. A CUDA-based template library Thrust [11] has been used for data transfer from Central Processing Unit (CPU) to GPU and some elementary reduction operations.

The parallel procedure in Algorithm 2 is described in following steps:

(i) Data related to isochromats and the pulse sequence data i.e. *isochromats* and *pulsedata* are transferred from CPU to GPU global-memory (Line 1 of Algorithm 2).

(ii) CalculateMagnetizationKernel, GPU implementation of Lines 12–14 in Algorithm 1 and CalculateSignalKernel, GPU implementation of line 16 in Algorithm 1 are configured for GPU calculations (Lines 2-3 of Algorithm 2).

(iii) CalculateMagnetizationKernel (Line 5 of Algorithm 2) is executed at each time step and CalculateSignalKernel (Lines 6-7 of Algorithm 2) is executed when output data is to be recorded.

In order to assess the performance enhancement with GPU, numerical experiments were performed in a Supermicro SuperServer 4027GR-TR system with e Ubuntu 14.04 operating system, 2x Intel Xeon Ivy Bridge E5-2650 main processors. To test the parallel version of the code a NVIDIA GTX Titan Black (Kepler GK110) GPU were used.

To measure the speed up, the simulations were carried out for $27 \times 27 \times 45$ isochromats for 258 000 time points. The output data were recorded in 1000 time-points. The parallel simulator took 22.61 s and the serial code took 1813.13 s. Approximately 82 times speed up is achieved.

3.4 Results

3.4.1 Slice Profile

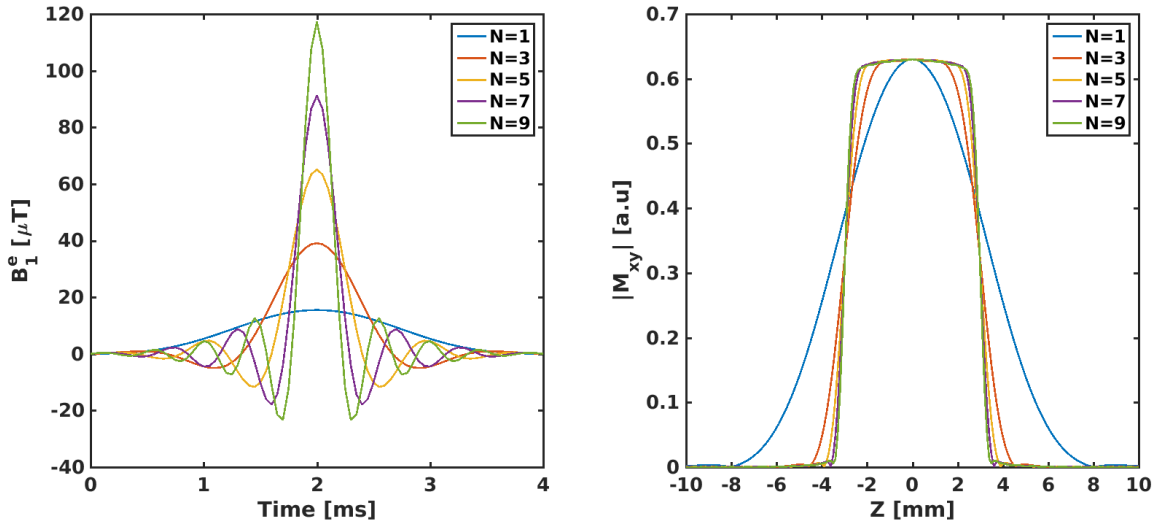

Figure 3.2: (Left) The envelope function of the Hanning-Windowed RF pulse with different number of side lobes. (Right) The slice profile at the end of one RF pulse duration resulting from the pulse with the corresponding envelope function. The envelope function on the left and the resulting slice profile on the right are marked with the same color.

As already mentioned, a truncated sinc pulses with a few side lobes results in a distorted non-rectangular slice profile. However, if the number of side lobes increase, the slice profile approaches towards a rectangular slice profile as depicted by the simulation results in Figure 3.2. To illustrate the effect of side lobes on the slice profile, the simulations were carried out for a sinc pulse with Hanning windowed apodizations over a duration of 4 ms. The nominal slice thickness was 6 mm and the corresponding slice selection gradient were calculated from Equation (2.23). The simulations were carried out for the sinc pulses for $2N$ zero-crossings with $N = 1, 3, 5, 7, 9$.

Figure 3.2 shows that the slice profile becomes more distorted with a decreasing number of side lobes. Therefore, a sinc pulse with multiple side lobes are favourable for a less distorted slice profile. However, due to the requirement of short repetition time fast truncated sinc pulses with very short durations are generally used as a compromise in dynamic MRI experiments.

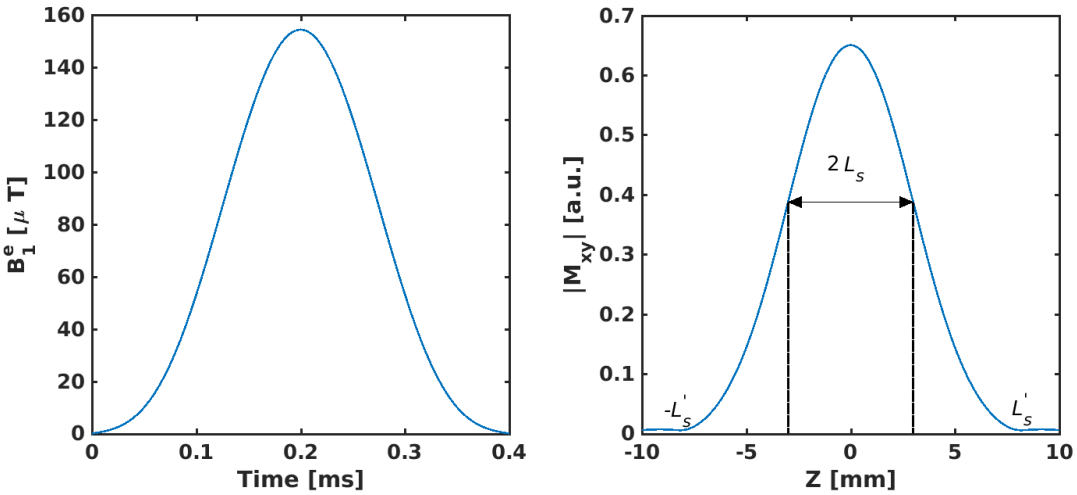

Figure 3.3: (Left) The envelope function of the Hanning-Windowed RF pulse. (Right) The slice profile at the end of RF pulse. $2L_s$ represents the nominal slice thickness and $2L_s'$ represents the actual slice thickness over which the RF pulse flips the equilibrium magnetizations.

A sinc pulse of duration 400 µs with Hanning window apodizations has been commonly applied for the MRI experiments conducted for the present study. In this short duration, sinc pulses with only the center lobe can be applied due to hardware constraints which results in a distorted slice profile as illustrated in the left part of Figure 3.3.

In order to have an idea about the resulting slice profile simulation was carried out for the sinc pulse with the envelope function $B_1^e(t)$ corresponding to 40° flip angle, a slice selection gradient G_{ss} corresponding to a nominal slice thickness $2L_s = 6$ mm. The resulting slice profile at the end of the RF pulse are plotted in the right part of Figure 3.3. The figure clearly shows that the truncated sinc pulse results in a distorted slice profile of an effective length $2L_s'$ instead of a rectangular slice profile of length $2L_s$. The figure also shows that the RF pulse does not flip the magnetization uniformly over the length of the slice. Therefore, the numerical simulations were performed over an estimated effective length $2L_s'$ along with taking into account the non-uniform slice profile.

3.4.2 Comparison between Numerical Methods

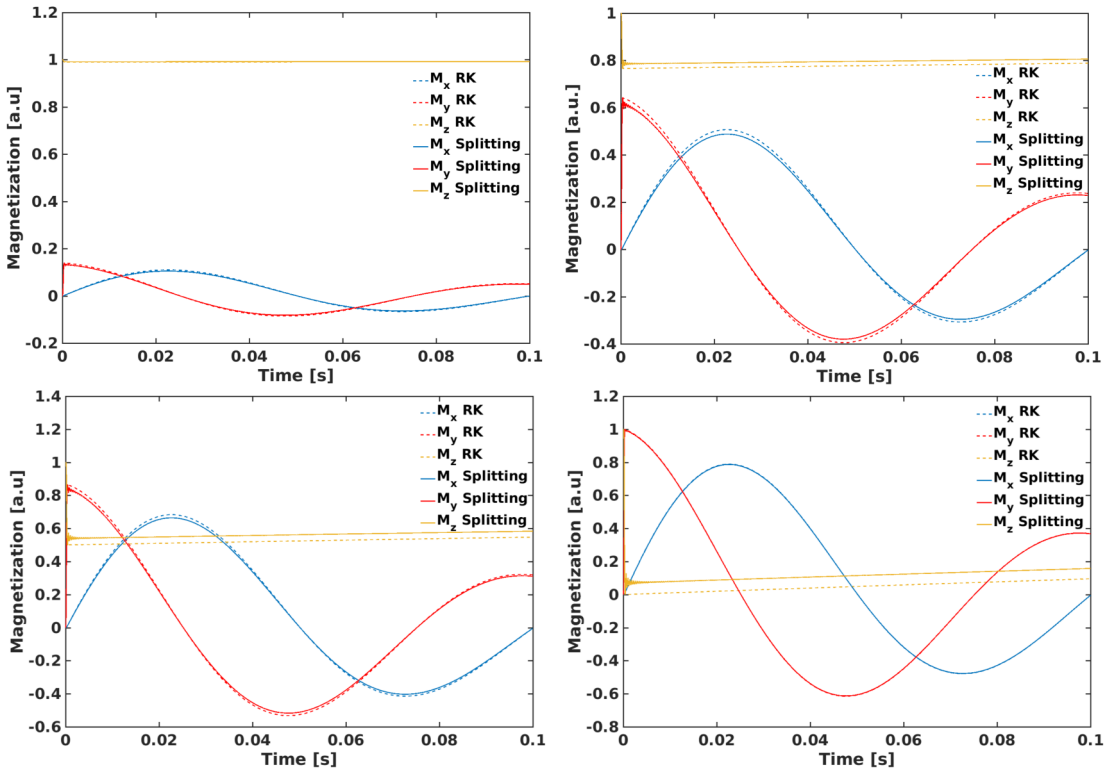

Figure 3.4: Single point excitation i.e. 0-dimensional case $T_1 = 1000\,\text{ms}$, $T_2 = 100\,\text{ms}$ with 101 isochromat elements having constant off-resonance from -50 to $50\,\text{Hz}$

In order to compare the accuracy and computational time between the embedded RK method and the operator splitting method, numerical simulations were performed for a single isochromat with the previously mentioned RF pulse and a constant off-resonance of 10 Hz. The simulations were performed for four different RF pulses for same number of time-steps. The RK simulations were performed with odeint c++ libraries [2]. Figure 3.4 shows the comparisons for the methods. Though the RK method is more accurate, it takes 4 times more computational time.

3.4.3 Effect of the Number of Subvoxels and Isochromats

To model the T_2^* effect of a gradient echo (GE) sequence, each ensemble of isochromats is assumed to be composed of an arbitrary number of isochromats with different off-resonance frequencies.

In order to study the effect of the number of isochromats in a randomly spoiled T_1 weighted radial FLASH sequence, simulations were carried out for a pulse sequence with

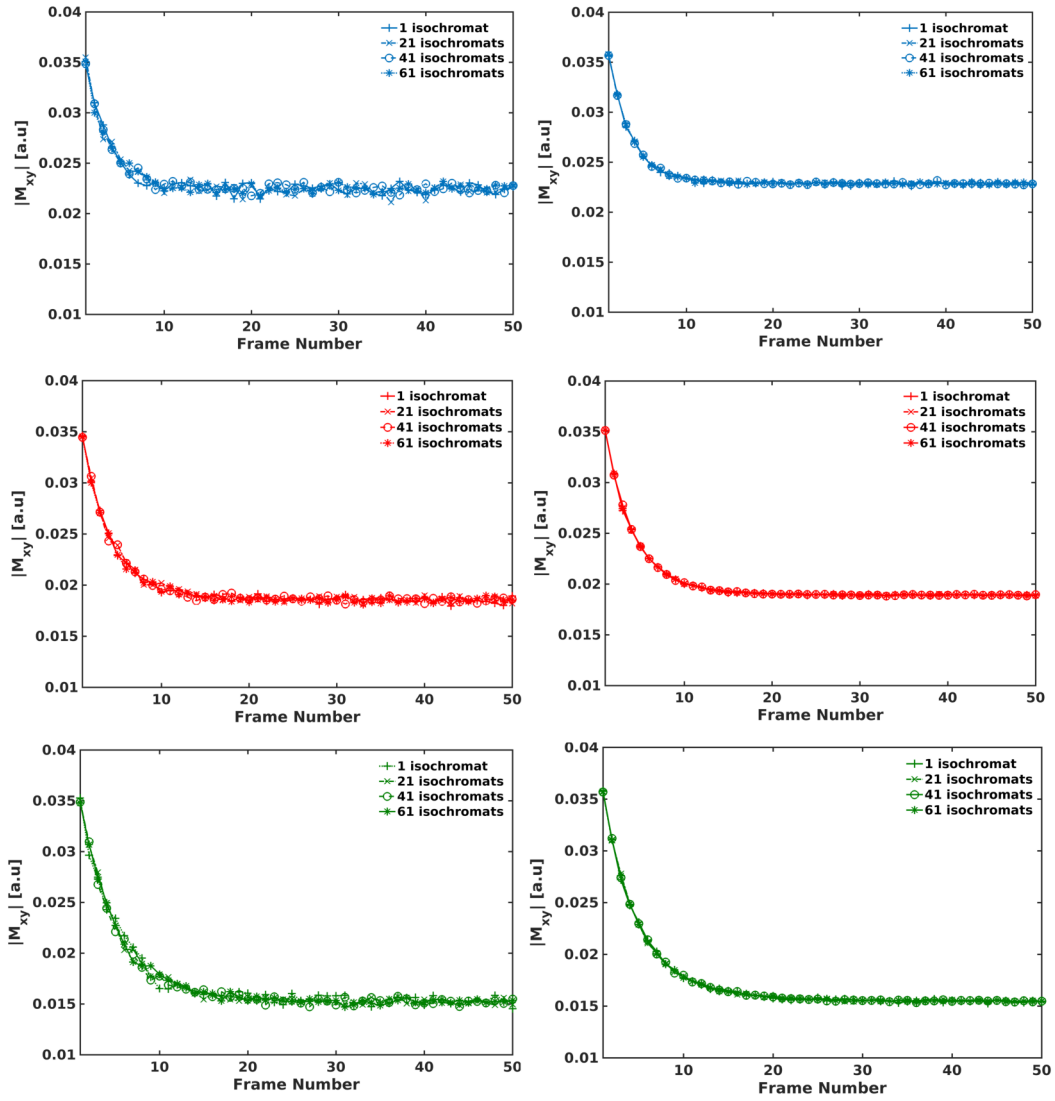

Figure 3.5: Magnitude of averaged integrated signal intensities are plotted as a function of frame number. The figures correspond to objects with following relaxation times: (Top) $T_1 = 296\,\text{ms}$, $T_2 = 113\,\text{ms}$, (middle) $T_1 = 456\,\text{ms}$, $T_2 = 113\,\text{ms}$, (bottom) $T_1 = 456\,\text{ms}$, $T_2 = 113\,\text{ms}$. (Left column) Simulated results are shown for three different objects for the computational domain of $4.8 \times 4.8 \times 18.0\,\text{mm}^3$ divided into $15 \times 15 \times 27$ subvoxels. (Right column) Simulated results are shown for three different objects for the computational domain of $4.8 \times 4.8 \times 18.0\,\text{mm}^3$ divided into $27 \times 27 \times 27$ subvoxels. For each of these cases, simulations were carried out with subvoxels consisting of 1, 21, 41 and 61 isochromats respectively.

TR/TE = 2.18/1.28 ms, flip angle 8°, in-plane resolution = $1.6 \times 1.6\,\text{mm}^2$, nominal slice thickness=6 mm, FOV = $256 \times 256\,\text{mm}^2$, number of spokes/turns = 17/5 and base resolution = 160.

The simulations were performed over a computational domain of $4.8 \times 4.8 \times 18.0\,\mathrm{mm}^3$ divided into $15 \times 15 \times 27$ and $27 \times 27 \times 27$ subvoxels respectively for three different objects. As a reference, simulations were performed taking only one isochromat in each subvoxel for both computational domains. Also, simulations were performed assuming each subvoxel to consist of 21, 41 and 61 isochromats in a Lorentzian distribution over a frequency range -20 to $20\,\mathrm{Hz}$. The magnitude of complex signals are recorded and summed up for all the isochromats at TE for all the repetitions. After that, they were averaged over the number of spokes. Magnitude of these averaged integrated signal intensities are plotted as simulation results.

On the left side of Figure 3.5, simulation results related to the computational domain with $15 \times 15 \times 27$ subvoxels is shown for 1, 21, 41 and 61 isochromat numbers within each subvoxel. Similarly, the right hand side of Figure 3.5 shows simulation results related to the computational domain $27 \times 27 \times 27$ for all four cases.

Figure 3.5 clearly depicts that the integrated magnitude signals do not depend much on the number of isochromats within a subvoxel. However, the number of subvoxels within a computational domain have an effect on the oscillations of the simulated results. The simulated results show an oscillatory behaviour with less number of subvoxels whereas the oscillations are negligible for slightly higher number of subvoxels. Thus, for further simulations each subvoxel is assumed to consist of one isochromat.

Chapter 4

Simulation of Bloch Equation for Moving Spins

Numerical simulations of flow MRI techniques have been used often as an essential tool for understanding the mechanism of signal production in previous studies. The earliest known work related to the numerical simulation for flowing spins was a study on the effect of fluid flow on slice selection [142]. The effect of flow on slice selection was also investigated later in [82].

There are also prior studies related to the simulation of magnetic resonance angiography (MRA) which has become a routine imaging modality for the clinical evaluation of diseases resulting from vascular shape distortions [75, 84]. The main purpose of simulating MRA is to increase the understanding of complex circulatory flow patterns, which redistribute the magnetic resonance (MR) signal in a complex way, generate flow artifacts and impair image quality.

Numerical simulations have also been performed for the phase contrast imaging (PC MRI). Although PC MRI provides useful qualitative information, it suffers from artifacts in highly complex and turbulent flow region. Numerical simulations can increase our understanding of artifacts and facilitate further development in PC MRI [103, 104].

However, current MRI studies still lack a comprehensive understanding of the flow-induced disturbances and the effect of flow on signal amplitude in magnitude images. In this work, mathematical modeling and numerical simulation of the Bloch equation for flowing spins has been used to increase our understanding the effect of flow on magnitude images and explore the possibility of quantitative estimation of flow properties from the T_1 weighted magnitude images.

4.1 Bloch Equation for Flowing Spins

There are fundamentally two different ways we can look at the time evolution of the magnetizations for flowing spins in presence of an external magnetic field during an MRI experiment - the Lagrangian approach and the Eulerian approach.

In the Lagrangian approach, the fluid particles are observed as they move through time and space along a flow trajectory. When the fluid particles move along flow trajectories in the presence of an external magnetic field during an MRI experiment, they also undergo the processes of the RF excitations, precession due to the magnetic gradients and time-relaxations.

In the Eulerian approach, flow of magnetized fluids is observed on specific locations in the space of a domain. The transportation of magnetizations due to fluid flow are incorporated by adding a convective term in the Bloch equations [45] and is given by

$$\frac{\partial \mathbf{M}}{\partial t} + \underbrace{(\mathbf{u}\cdot\boldsymbol{\nabla})\mathbf{M}}_{\text{transport}} = \underbrace{\gamma \mathbf{M}\times\mathbf{B}_{\text{ext}} + \frac{(M_0 - M_z)\hat{e}_z}{T_1} - \frac{M_x\hat{e}_x + M_y\hat{e}_y}{T_2}}_{\text{magnetic resonance (MR)}}. \tag{4.1}$$

Due to signal demodulation, the signal acquired from the flowing spin in an MRI experiment is actually equivalent to an equation where magnetizations and the magnetic field ($\mathbf{B}_{\text{eff}}$) are written in the rotating frame with $\mathbf{u}$ is kept in laboratory frame [84].

$$\frac{\partial \mathbf{M}'}{\partial t} + (\mathbf{u}\cdot\boldsymbol{\nabla})\mathbf{M}' = \gamma \mathbf{M}'\times\mathbf{B}_{\text{eff}} + \frac{(M_0 - M_z)\hat{e}_z}{T_1} - \frac{M_{x'}\hat{e}_{x'} + M_{y'}\hat{e}_{y'}}{T_2}. \tag{4.2}$$

Bloch Equation for flowing spins given by Equation (4.2) along with proper initial and boundary conditions is a well-posed problem as discussed elaborately in Appendix B.

Additionally, if we assume an incompressible flow field which is true for most of the experimental conditions with body fluids and water, the Eulerian Bloch equations for flowing spins can be expressed as

$$\boldsymbol{\nabla}\cdot\mathbf{u} = 0, \tag{4.3a}$$

$$\frac{\partial \mathbf{M}'}{\partial t} + \underbrace{\boldsymbol{\nabla}\cdot(\mathbf{u}\otimes\mathbf{M}')}_{\text{transport}} = \underbrace{\gamma \mathbf{M}'\times\mathbf{B}_{\text{eff}} + \frac{(M_0 - M_z)\hat{e}_z}{T_1} - \frac{M_{x'}\hat{e}_{x'} + M_{y'}\hat{e}_{y'}}{T_2}}_{\text{magnetic resonance (MR)}}. \tag{4.3b}$$

4.2 Numerical Strategies

For developing a numerical simulator for Bloch equations expressed in the Lagrangian approach, the flow trajectories of fluid particles are calculated analytically for simple flow situations or computationally for more complex flow situations. Subsequently, the Bloch equations are solved for all points along the flow trajectories. This approach is taken by Marshall et al. in [90, 89] and Tyen et al. in [135].

The advantage of the Lagrangian approach is that it can be understood intuitively and works well for simple streamlined flow patterns. However, the Bloch equations are required to be solved always for a substantial number of flow trajectories for precise simulation. As a result, this approach is computationally demanding even for simple flow patterns. Apart from that, Lagrangian approach has several disadvantages for complex flow and geometries. Firstly, determining the flow trajectories is extremely difficult for complex flow patterns. Secondly, a high level of spatial resolution near the wall is required to resolve adequately subtle changes in geometries. These disadvantages make it difficult to apply the Lagrangian approach effectively for complex flow patterns in a complex geometry.

To develop a numerical simulator in Eulerian framework, the computational domain is divided into a mesh. The magnetizations are calculated on the mesh as a function of time and space from Equation (4.3). This approach was chosen in [73, 74, 75, 84]. As this approach does not require flow pathline tracking, it is well-suited for modeling complex flow pattern.This approach is computationally less expensive compared to Lagragian approach as well [74]. Considering these advantages, Eulerian framework is chosen over the Lagrangian framework for developing the simulator for flowing spins in this work.

Previously, Equation (4.3) is solved using multiple numerical strategies. Jou et al. in [73] and Lorthois et al. in [84] solved Equation (4.3) by first solving the flow field in an Eulerian mesh in a FVM software. They expressed Equation (4.3) later in terms of longitudinal magnetizations and the magnitude and phase of transverse magnetizations [84, 73]. These equations are solved further using finite difference method (FDM). However, they took the assumptions of a short-lived pulse and the RF excitation profile is assumed to be sharp-edged which are significant deviations for dynamic MRI as will be seen from Section 4.5 and later in Section 5.5.1.

Jurczuk et al. in [75] solved Equation (4.3) by splitting the transport and the MR terms. In their work, the magnetization transport was calculated using lattice-Boltzmann method (LBM) and the magnetizations were calculated using the sequential

splitting technique as discussed in Section 3.2.2. Operator splitting approach is used in this work as well with a modification as discussed in the next subsection.

Splitting of Transport and Magnetic Resonance Terms

In general, a physical phenomena represent the combined effect of several processes with different scales. For example, the Bloch equation for flowing spins, given by Equation (4.3), has two subprocesses of magnetization transport and local magnetization transformations due to rotation and relaxation. In this case, the behaviour of the magnetization evolution is described by the sum of the two sub-operators of different physical nature: transportation and the MR phenomena. A differential splitting, described in Section 3.2.2, is used widely to solve such mutiscale, multiphysics problem.

The general advantages of the splitting process can be listed as follows [146]:

- Different numerical methods can be used to treat different sub-problems with optimal numerical methods for the each subproblems.
- It is easy to change numerical algorithms for different sub-problems in order to achieve improvements of the codes.
- Different time-scales can be used to resolve different subproblems which helps in reducing computational load. In many cases, the splitting procedure leads to better parallel implementation.

On the other hand, the operator splitting introduces splitting errors which interact with the errors of each subprocess in a very complex way. Thus, it is very difficult to control the impact of these errors on the total error. Moreover, the operator splitting typically leads to difficulties in handling the boundary conditions.

In this work, a differential splitting approach is applied to Equation (4.3) as well. Here, Equation (4.3) represents an *advection-reaction* equation where the transport term in Equation (4.3) takes care of the *advection* of the magnetizations at a specific point in time and space from the neighbouring locations. The MR term in Equation (4.3) which describes the time evolution of magnetizations at that specific time and spatial point due to RF excitations and time-relaxation is analogous to the *reaction* term. In an analytical setting, an *advection-reaction* equation of a space and time dependent vector quantity $\mathbf{w} \equiv \mathbf{w}(\mathbf{r}, t)$ is given by

$$\frac{\partial \mathbf{w}(\mathbf{r}, t)}{\partial t} + \underbrace{\nabla \cdot (\mathbf{u}\mathbf{w})}_{\text{advection}} = \underbrace{\mathbf{f}(\mathbf{w})}_{\text{reaction}} . \tag{4.4}$$

The reason behind calling $\mathbf{f}(\mathbf{w})$ in Equation (4.4) reaction, comes from the frequent occurrence of this form of equation in chemical transport, also known as the *source* term.

If we split Equation (4.4) into *advection* and *reaction*, the following two sub-problems need to be solved:

$$\frac{\mathrm{d}\mathbf{w}^*(t)}{\mathrm{d}t} = \mathbf{f}(\mathbf{w}) = \mathbf{F}_\mathbf{r}(\mathbf{w}^*(\mathbf{r},t)), \quad t_n < t \leq t_{n+1}, \quad \mathbf{w}^*(t_n) = \mathbf{w}_n, \tag{4.5a}$$

$$\frac{\partial \mathbf{w}^{**}(t)}{\partial t} = -\nabla \cdot (\mathbf{u}\mathbf{w}) = \mathbf{F}_\mathbf{a}(\mathbf{w}^{**}(\mathbf{r},t)), \quad t_n < t \leq t_{n+1}, \quad \mathbf{w}^{**}(t_n) = \mathbf{w}^*_{n+1}. \tag{4.5b}$$

Let us assume that the numerical operator of Equation (4.4) is $\mathcal{S}_\tau$ and then the solution of Equation (4.4) is given by

$$\mathbf{w}_{n+1} = \mathcal{S}_\tau(\mathbf{w}_n), \quad \tau = t_{n+1} - t_n. \tag{4.6}$$

Let us also assume that the solution operator for Equation (4.5a) is $\mathcal{S}_{r,\tau}$ and for Equation (4.5b) $\mathcal{S}_{a,\tau}$. Then the approximate solution using operator splitting is given by

$$\mathbf{w}_{n+1} = \mathcal{S}_{a,\tau}(\mathcal{S}_{r,\tau}(\mathbf{w}_n)). \tag{4.7}$$

The numerical solution using the splitting method will introduce a splitting error $\boldsymbol{\epsilon}_s$ which is given by

$$\boldsymbol{\epsilon}_s = \frac{1}{2}\tau^2[\mathbf{F}_r, \mathbf{F}_a](\mathbf{w}) + O\Big(\tau^3\Big), \tag{4.8a}$$

where

$$[\mathbf{F}_r, \mathbf{F}_a] = \mathbf{F}'_r(\mathbf{w})\mathbf{F}_a(\mathbf{w}) - \mathbf{F}'_r(\mathbf{w})\mathbf{F}_a(\mathbf{w}), \tag{4.8b}$$

and

$$\mathbf{F}'_a(\mathbf{w})\mathbf{F}_r(\mathbf{w}) = -\nabla \cdot (\mathbf{u}\mathbf{f}(\mathbf{w})) = -\mathbf{f}'(\mathbf{w})(\mathbf{u} \cdot \nabla \mathbf{w}) - (\nabla \cdot u)\mathbf{f}(\mathbf{w}) - \sum_{i=1}^{3} u_i \frac{\partial \mathbf{f}(\mathbf{w})}{\partial x_i}, \tag{4.8c}$$

$$\mathbf{F}'_r(\mathbf{w})\mathbf{F}_a(\mathbf{w}) = -\mathbf{f}'(\mathbf{w})\nabla \cdot (\mathbf{u}\mathbf{w}) = -\mathbf{f}'(\mathbf{w})(\mathbf{u} \cdot \nabla \mathbf{w}) - \mathbf{f}'(\mathbf{w})(\nabla \cdot \mathbf{u})\mathbf{w}. \tag{4.8d}$$

It can be proved that if the commutator $[\mathbf{F}_r, \mathbf{F}_a]$ vanishes then the splitting error is completely zero [68]. In order to determine the condition under which the commutator

vanishes, the expression for the commutator is evaluated using Equations (4.8c)–(4.8d) [78],

$$[\mathbf{F}_r, \mathbf{F}_a] = (\nabla \cdot \mathbf{u})(\mathbf{f}(\mathbf{w}) - \mathbf{f}'(\mathbf{w})\mathbf{w}) + \sum_{i=1}^{3} u_i \frac{\partial \mathbf{f}(\mathbf{w})}{\partial x_i}\,. \tag{4.9}$$

The following result can be obtained from the above expression [78, 68]

Theorem 4.1. *Advection commutes with reaction if : (i)* $\mathbf{f}$ *is independent of* $\mathbf{r}$ *and* $\nabla \cdot \mathbf{u} = 0$ *or (ii)* $\mathbf{f}$ *is independent of* $\mathbf{r}$ *and linear in* $\mathbf{u}$.

According to Theorem 4.1, the commutator for advection-reaction equation does not vanish as f depends on $\mathbf{r}$ which results in a truncation error due to splitting. The local truncation error $\boldsymbol{\epsilon}_t$ due to splitting is given by

$$\boldsymbol{\epsilon}_t(t_n) = \frac{\tau}{2}[(\nabla.\mathbf{u})(\mathbf{f}(\mathbf{w}) - \mathbf{f}'(\mathbf{w})\mathbf{w}) + \sum_{i=1}^{3} u_i \frac{\partial \mathbf{f}(\mathbf{w})}{\partial x_i}] + O\big(\tau^2\big). \tag{4.10}$$

The detailed numerical analysis related to the advection-reaction equation is not presented further. For an in-depth discussion on the evolution equation with bounded operators, the reader may consult the review paper [92] by McLachlan et al. and the book [38] by Farago et al. There are a few recent studies on splitting techniques [60, 59, 6] which focus on the unbounded operators as well.

In order to solve the Equation (4.3) by splitting, the order in which the operators are solved is crucial. To get a fully consistent treatment, *reaction* part Equation (4.5a) is solved first and the *advection* term Equation (4.5b) is solved next as explained by Hundsdorfer et al. in [67, 68]. Equation (4.5a) is solved using the splitting technique described in Section 3.2.2. Numerical solution strategies to solve the transport of magnetization Equation (4.5b) is discussed next.

4.3 Numerical Strategies for the Solution of Advection Equation

In an analytical setting the advection part of Equation (4.4) is given by

$$\frac{\partial \mathbf{w}(\mathbf{r},t)}{\partial t} + \nabla \cdot (\mathbf{u}(\mathbf{r},t) \otimes \mathbf{w}(\mathbf{r},t)) = \mathbf{0}, \quad \mathbf{r} \in \Omega, \quad t \in (0,T], \tag{4.11a}$$

$$\mathbf{w}(\mathbf{r},0) = \mathbf{w}_0(\mathbf{r}), \quad \mathbf{r} \in \Omega, \tag{4.11b}$$

$$\mathbf{w}(\mathbf{r},t)|_{\mathbf{u}\cdot\mathbf{n}_\Gamma<0} = \mathbf{w}_b, \quad t \in [0,T]. \tag{4.11c}$$

Equation (4.11) can be solved using various methods, viz. LBM [75], discontinuous Galerkin method (dGFEM) [117], FDM [84, 73].

It has been shown in Appendix C that for $k = 0$ a generalized dGFEM formulation reduces to standard FVM. In the present work, a high-resolution FVM is used for the simulations of the transport of magnetizations. The biggest advantage of FVM is the possibility of easy implementation of the method for a complex geometry unlike FDM and FVM can be easily parallelized in GPU. Also FVM is computationally less expensive than dGFEM.

Beginning with the semi-discretized form for FVM given by Equation (C.10) in Appendix C, techniques to determine numerical fluxes and time-discretizations are discussed in this section. Detailed discussions about FVM can be found in [65, 66, 136]. FVM for hyperbolic PDEs is discussed in [81, 130].

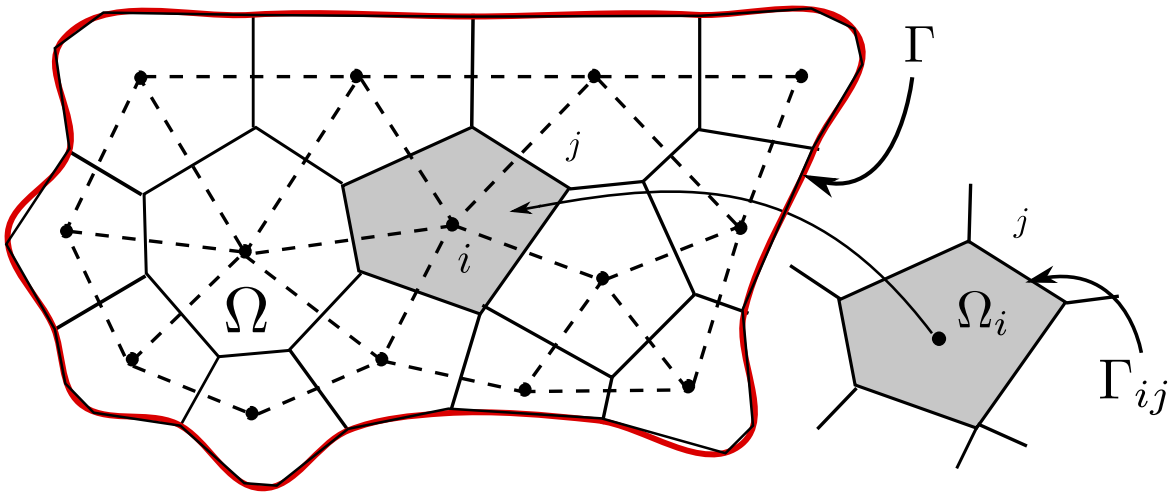

Figure 4.1: Schematic representation of a 2-D grid. Ω represent the computational domain. The boundary of the domain Γ is marked with red line. i-th cell is magnified and Ω_i and Γ_{ij} represent the area of the i-th cell and the common surface between i-th and the j-th cell respectively.

As depicted in Figure 4.1, in order to get numerical solutions the spatial domain $\Omega \in \mathbb{R}^n$ is divided into a finite number of cells where the variables are stored in the

center of each cell. The grid consists of a set of closed control volumes $\{\overline{\Omega}_i \subset \Omega : i = 1, 2, 3 \cdots I\}$ where $\mathcal{R} = \{\mathbf{r}_i \in \Omega : i = 1, 2, 3, \cdots, I\}$ represents cell centers marked with black dots in Figure 4.1. The grid has the following properties :

1. $\overline{\Omega} = \bigcup_{i=1}^{I} \overline{\Omega}_i$.
2. $\Omega_i \bigcap \Omega_j = \emptyset,\ i \neq j$.
3. $\overline{\Omega}_i \bigcap \overline{\Omega}_j = \Gamma_i \bigcap \Gamma_j = \Gamma_{ij}$.

As discussed in Appendix C, the semi-discretized equation based on the cell-average values of i-th cell $\mathbf{w}_i(\mathbf{r}, t)$ is given by

$$\frac{\mathrm{d}(|\Omega_i|\,\mathbf{w}_i)}{\mathrm{d}t} + \sum_{j \in J_i} \mathbf{F}_{ij}(\mathbf{u}, \mathbf{w}) \left|\Gamma_{ij}\right| = \mathbf{0}, \tag{4.12}$$

where $\mathbf{w}_i(\mathbf{r}_i, t),\ i = 1, 2, \cdots, I$ is assigned to the i-th point of $\mathcal{R}$. As the grid dimensions are assumed not to be changing with time, Equation (4.12) reduces to

$$\frac{\mathrm{d}\mathbf{w}_i(t)}{\mathrm{d}t} + \frac{1}{|\Omega_i|} \sum_{j \in J_i} \mathbf{F}_{ij}(\mathbf{u}, \mathbf{w}) \left|\Gamma_{ij}\right| = \mathbf{0}. \tag{4.13}$$

The cell-averaged values at i-th cell $\mathbf{w}_i(\mathbf{r}, t)$ at each time-step depends on the flux $\mathbf{F}_{ij}$ calculated at different faces of the i-th cell. Various spatial discretization schemes determine the flux function $\mathbf{F}_{ij}$ with different order of accuracy, as a function of the values of $\mathbf{w}_i$ for a number of cells. The cells together constitute the *stencil* of the numerical scheme. Along with a spatial discretization scheme, a proper time-stepping technique must be chosen to calculate the time evolution of $\mathbf{w}_i(\mathbf{r}, t)$ in Equation (4.13), which will be discussed next.

4.3.1 Time Discretization

There are two fundamental methods of time discretization - explicit and implicit time-stepping. In explicit time stepping, cell-averaged values $\mathbf{w}_i(\mathbf{r}, t_{n+1})$ at $t = t_{n+1}$ are calculated from the cell-averaged values $\mathbf{w}_i(\mathbf{r}, t_n)$ at $t = t_n$ whereas $\mathbf{w}_i(\mathbf{r}, t_{n+1})$ are calculated solving a system of algebraic equations involving both $\mathbf{w}_i(\mathbf{r}, t_n)$ and $\mathbf{w}_i(\mathbf{r}, t_{n+1})$ in implicit time stepping methods.

The choice of time-stepping method depends on the characteristic time scale of the problem. The biggest advantage of an implicit time-stepping over the explicit methods is that implicit method are unconditionally stable whereas the time-step sizes for

explicit time-stepping methods are limited by certain stability criteria. However, when the important time-scales of a hyperbolic PDE is less than or equal to the time-step for a stable explicit method, then a explicit time-stepping is favored.

Moreover, explicit methods do not require us to solve a set of algebraic equations per time-step as required for implicit methods. Therefore, the cost per time step for implicit methods are higher than for explicit methods. Implicit time-stepping methods very often need to be used with preconditioners for faster convergence of the iterative solver of the algebraic equation which is harder to implement. On the other hand, explicit methods are easy to implement and require a low amount of computer memory. Considering all the factors mentioned above an explicit method is chosen in this work.

With the simplest first order explicit time-stepping, Equation (4.13) can be approximated as,

$$\mathbf{w}_i^{n+1} - \mathbf{w}_i^n = \sum_{j \in J_i} \frac{\tau}{\Omega_i} \left(\int_{t_n}^{t_{n+1}} \mathbf{F}_{ij}(\mathbf{u}, \mathbf{w}) \left| \Gamma_{ij} \right| \, dt \right) + O(\tau), \tag{4.14}$$

where

$$\tau = t_{n+1} - t_n, \quad \mathbf{w}_i^n = \mathbf{w}_i(t_n).$$

Equation (4.14) is a general multi-dimensional discretized form of Equation (4.12).

The numerical strategy for solving the transport equations depends further on the spatial discretization of the numerical flux at each time step as discussed in the next section.

4.3.2 Spatial Discretization

To study the effect of one-dimensional flow field on magnetization transport is the obvious starting point. Not only that, one-dimensional flow field is frequently encountered in human bodies and of major practical importance. The present work is also focused on the analysis of the Bloch equations for flowing spins for one-dimensional through-plane steady and pulsatile flow field. The spatial discretization strategies of one-dimensional flow transport equation is discussed in this section beginning from the first-order upwind method to high-resolution schemes.

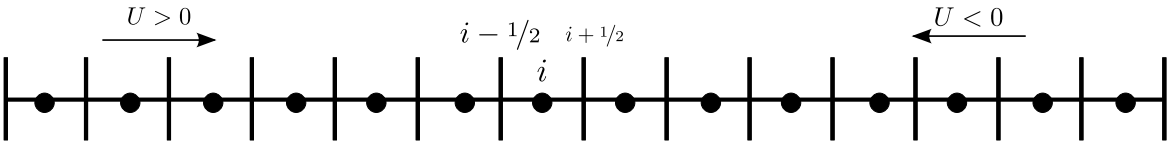

Figure 4.2: Schematic representation of a 1-D cell-centred grid.

Simple Upwind

Transport equation for a flow field in one dimension in z-direction is given by

$$\frac{\partial \mathbf{w}(\mathbf{r},t)}{\partial t} + \frac{\partial (u_z(\mathbf{r},t)\mathbf{w}(\mathbf{r},t))}{\partial z} = \mathbf{0}. \tag{4.15}$$

In upwind methods, the average flux function is determined by looking at the direction from which the information is coming, i.e. it is an one-sided method. The numerical flux is determined based on values only to the left or only to the right for positive and negative velocity as depicted in Figure 4.2.

The difference equation for simple upwind scheme can be written as,

$$\mathbf{w}_i^{n+1} = \mathbf{w}_i^n - \frac{\tau}{\Delta z}(\mathbf{F}^n_{i+1/2} - \mathbf{F}^n_{i-1/2}) + O(\tau) + O(\Delta z), \tag{4.16}$$

where $\mathbf{F}^n_{i+1/2}$ denote the flux through the right edge of the i-th cell and $\mathbf{F}^n_{i-1/2}$ is the flux through the left edge of the i-th cell and they are given by

$$\mathbf{F}^n_{i+1/2} = u_i^n \mathbf{w}_i, \qquad \mathbf{F}^n_{i-1/2} = u^n_{i-1}\mathbf{w}_{i-1}, \qquad \text{if } U > 0 \tag{4.17a}$$

$$\mathbf{F}^n_{i+1/2} = u^n_{i+1}\mathbf{w}_{i+1}, \qquad \mathbf{F}^n_{i-1/2} = u_i^n \mathbf{w}_i, \qquad \text{if } U < 0 \tag{4.17b}$$

The above equation shows that simple upwind has a two-point stencil and it is first order accurate in time and space. It can be shown from the modified equation (explained in Appendix D) that the upwind method introduces artificial diffusion which results in a lower order accuracy [81]. Nevertheless, the upwind method is monotonicity-preserving and non-oscillatory (explained in Appendix D).

The explicit upwind method is stable if it satisfies the CFL condition $u_i \frac{\tau}{\Delta z} \leq 1$.

Lax-Wendroff Method

One of the more accurate flux approximation methods is the Lax-Wendroff method which is second order accurate. This method has an extra term to correct for the artificial diffusion introduced in the upwind method [81]. This extra term introduces an extra-diffusion term which compensates for the numerical diffusion term completely.

The difference equation for Lax-Wendroff method is given by

$$\mathbf{w}_i^{n+1} = \mathbf{w}_i^n - \frac{\tau}{\Delta z}(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n) - \frac{\tau}{\Delta z}(\mathbf{F}_{h,i+\frac{1}{2}}^n - \mathbf{F}_{h,i-\frac{1}{2}}^n), \tag{4.18}$$

$$\mathbf{F}_{h,i-\frac{1}{2}}^n = \frac{1}{2}\,|s_{i-\frac{1}{2}}|\,(1 - \frac{\tau}{\Delta z}\,|s_{i-\frac{1}{2}}|)\mathbf{W}_{i-\frac{1}{2}}, \tag{4.19}$$

where

$$s_{i-1/2} = \begin{cases} u_i^n, & \text{if } U > 0, \\ u_{i-1}^n, & \text{if } U < 0, \end{cases} \qquad \mathbf{W}_{i-1/2} = \mathbf{w}_i^n - \mathbf{w}_{i-1}^n. \tag{4.20}$$

$\mathbf{F}_{i+1/2}^n$ and $\mathbf{F}_{i-1/2}^n$ have the same expression as for the upwind method. $\mathbf{F}_{h,i-\frac{1}{2}}^n$ and $\mathbf{F}_{h,i+\frac{1}{2}}^n$ represent the anti-diffusive term of Lax-Wendroff method. The above equations show that the Lax-Wendroff method has a three point stencil. One can show that it is second order accurate in space and first order accurate in time. The CFL condition for the Lax-Wendroff method is also give by $u_i \frac{\tau}{\Delta z} \leq 1$.

The major advantage of this method is that it gives a more accurate solution in comparison to upwind method in regions with smooth solutions. On the other hand, this method results in oscillations near discontinuities [81]. Also, Lax-Wendroff method introduces a dispersive term in the modified equation which causes a slight shift in smooth humps, a *phase error* as shown in [81].

High Resolution Schemes

High resolution schemes combine best features of the non-oscillatory method such as the upwind method and the higher order accurate method such as the Lax-Wendroff method. As a consequence, high resolution methods are at least second order accurate on smooth sections of the solution while preserving non-oscillatory behavior near discontinuities and smooth humps.

One of the popular approaches to find a high-resolution scheme is by using flux-limiter methods. Using this method the numerical flux at $i - \frac{1}{2}$-th edge is written as

$$\mathbf{F}_{i-\frac{1}{2}} = \mathbf{F}_{i-\frac{1}{2}}^L + \psi(\theta_{i-\frac{1}{2}}^n)(\mathbf{F}_{i-\frac{1}{2}}^H - \mathbf{F}_{i-\frac{1}{2}}^L), \tag{4.21}$$

where $\mathbf{F}_{i-\frac{1}{2}}^L$ and $\mathbf{F}_{i-\frac{1}{2}}^H$ denote the flux for a lower and a higher order scheme respectively at $i - \frac{1}{2}$-th edge. $\psi(\theta_{i-\frac{1}{2}}^n)$ is called the flux limiter and $\theta_{i-\frac{1}{2}}^n$ is known as the smoothness

parameter and is given by

$$\theta^n_{i-\frac{1}{2}} = \frac{\left|\mathbf{W}^n_{I-1/2}\right|}{\left|\mathbf{W}^n_{i-1/2}\right|} \qquad I = \begin{cases} i-1 & \text{if } u_i > 0, \\ i+1 & \text{if } u_i < 0. \end{cases} \tag{4.22}$$

We have to choose a proper $\psi(\theta)$ such that the desirable properties of a high resolution scheme can be obtained. From the discussions in Appendix D, high resolution schemes

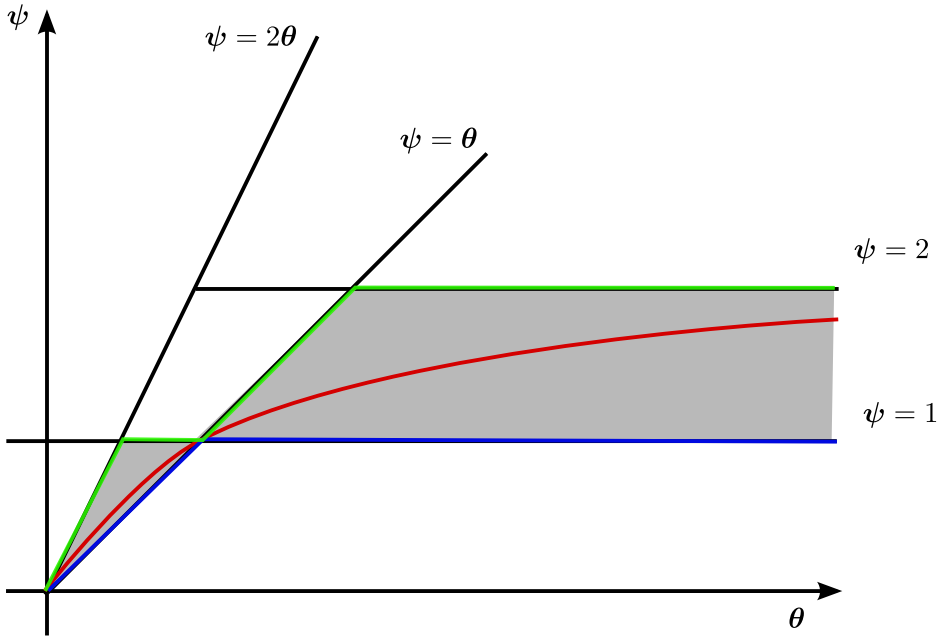

Figure 4.3: Limiter function $\psi(\theta)$. The shaded region shows the high-resolution TVD region. Green, red, blue lines lie along the boundary of the superbee,Van Leer and the minmod limiter functions. $\psi(\theta) = 1$ and $\psi(\theta) = \theta$ represent the boundary of the Lax-Wendroff and the Beam-Warming methods.

must satisfy the TVD requirements to be non-oscillatory. For a scheme to satisfy the TVD criteria the value of limiter function must satisfy the following criteria [126]:

$$0 \leq \frac{\psi(\theta)}{\theta} \leq 2, \qquad 0 \leq \psi(\theta) \leq 2. \tag{4.23}$$

Also, from the discussions in Appendix D, high resolution schemes must be nonlinear. Sweby introduced in [124] the following criteria for a scheme to be second order high resolution which is illustrated by the shaded region in Figure 4.3:

- if $0 < \theta < 1$ for high-resolution TVD schemes $\theta < \psi(\theta) < 1$
- if $\theta \geq 1$ for high-resolution TVD schemes $1 < \psi(\theta) < \theta$

Few popular limiter functions are listed in Table 4.1 and are also marked in Figure 4.3.

Table 4.1: Limiter functions

Name	$\psi(\theta)$
minmod [110]	$\text{minmod}(1,\theta)$
superbee [110]	$\max(0,\min(1,2\theta),\min(2,\theta)$
Van Leer [134]	$\frac{\theta+\lvert\theta\rvert}{1+\theta}$

$$\text{where minmod}(a,b) = \begin{cases} a, \text{ if } |a| < |b| \text{ and } ab > 0, \\ b, \text{ if } |b| < |a| \text{ and } ab > 0, \\ 0, \text{ if } ab < 0. \end{cases}$$

Choosing the upwind method as the lower order method and the Lax-Wendroff method as the higher order method, the high resolution numerical discretization for Equation (4.15) can be written as

$$\mathbf{w}_i^{n+1} = \mathbf{w}_i^n - \frac{\tau}{\Delta z}(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n) - \frac{\tau}{\Delta z}(\widetilde{\mathbf{F}_{h,i+\frac{1}{2}}^n} - \widetilde{\mathbf{F}_{h,i-\frac{1}{2}}^n}), \tag{4.24}$$

$$\widetilde{\mathbf{F}_{h,i-\frac{1}{2}}^n} = \psi(\theta_{i-\frac{1}{2}}^n)\mathbf{F}_{h,i-\frac{1}{2}}^n, \tag{4.25}$$

where $\widetilde{(\cdot)}$ is the flux-limited term of the anti-diffusive term $(\cdot)$ in Lax-Wendroff scheme. It is obvious from Equation (4.24) that $\psi(\theta) = 0$ and $\psi(\theta) = 1$ for upwind and Lax-Wendroff scheme respectively.

4.3.3 Boundary Conditions and Ghost Cells

In the transport equation, the magnetizations in the inner cells are calculated by adding the effect of the magnetization transport from the neighboring cells. However, the magnetizations on the boundary cells need to be calculated from the imposed conditions on the boundaries. Therefore, application of accurate boundary conditions is crucial for accuracy of the numerical result.

In order to solve Equation (4.3) the inflow condition needs to be specified as boundary conditions. In this work, the computational domain is chosen such that a fully saturated incoming magnetizations can be assumed as the inflow conditions. That implies the boundary condition is given by

$$M_x = 0, \quad M_y = 0, \quad M_z = M_0, \mathbf{u} \cdot \mathbf{n}_\Gamma < 0. \tag{4.26}$$

For numerical implementation very often the computational domain is extended to include a few additional cells on every end. These additional cells are known as *ghost cells* as illustrated schematically in Figure 4.4. For one-sided methods like upwind

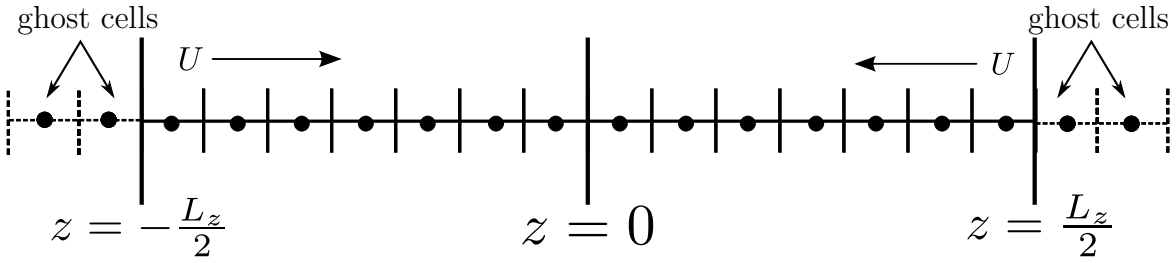

Figure 4.4: Schematic illustration of ghost cell in one-dimensional computational domain.

method implementing the boundary condition requires implementing only the inflow boundary conditions. On the other hand, for numerical implementation of boundary condition in higher order methods such as Lax-Wendroff or high resolution methods an outflow boundary condition needs to be specified. However, for a hyperbolic PDE such as transport equation the outflow boundary condition has no physical meaning and a compatible numerical strategy is necessary.

One way to solve this problem is to use upwind methods in the outflow boundary cell and more accurate method on inner cells. Another approach is to impose the boundary condition in the ghost cells in the outflow directions is by extrapolating data from the interior cell. Zero-order extrapolation and the first order extrapolation from the interior data are very frequently used. Though first-order extrapolation is more accurate, it very often leads to stability problems [81]. Therefore, zero-order extrapolation is used here which means assigning the inner-boundary cell values to ghost cells when required which means

$$\mathbf{w}_{l+1}^n = \mathbf{w}_l^n, \qquad \mathbf{w}_{l+2}^n = \mathbf{w}_l^n, \qquad \text{if } U > 0 \tag{4.27a}$$

$$\mathbf{w}_{l-1}^n = \mathbf{w}_l^n, \qquad \mathbf{w}_{l-2}^n = \mathbf{w}_l^n, \qquad \text{if } U < 0 \tag{4.27b}$$

where the outer boundary cell is denoted by l.

4.4 Bloch Simulator for Flowing Spins

The Bloch equations simulator for flowing spins is based on the splitting method described in Section 4.2. The simulator is extended from the simulator for spatially

stationary object (Section 3.3) to include the magnetization transport. The computational kernel simulates the magnetization transformation under the influence of the RF pulse and magnetic gradient and then the magnetization transport due to flow sequentially at each time step.

Using the simulator, the *transport* part of Equation (4.3) can be solved using upwind, Lax-Wendroff and high-resolution flux-limited FVM methods. The MR term of Equation (4.3) can be solved using sequential, Strang or SWSS splitting techniques for solving Bloch equations for spatially stationary objects as described in Section 3.2.2. This simulator takes the velocity field as an additional input along with the pulse sequence data and the model of the computational domain. Although the simulator in the present stage can take any one-dimensional time and space dependent velocity field, the present discussion is limited to through-plane flow with constant and time dependent velocity field. Regarding the output, the integrated pixel intensities are recorded at previously specified time points similarly as the simulator for spatially stationary objects. The magnetization transport is also parallelized in GPU like the Bloch phenomena.

4.4.1 Computational Model

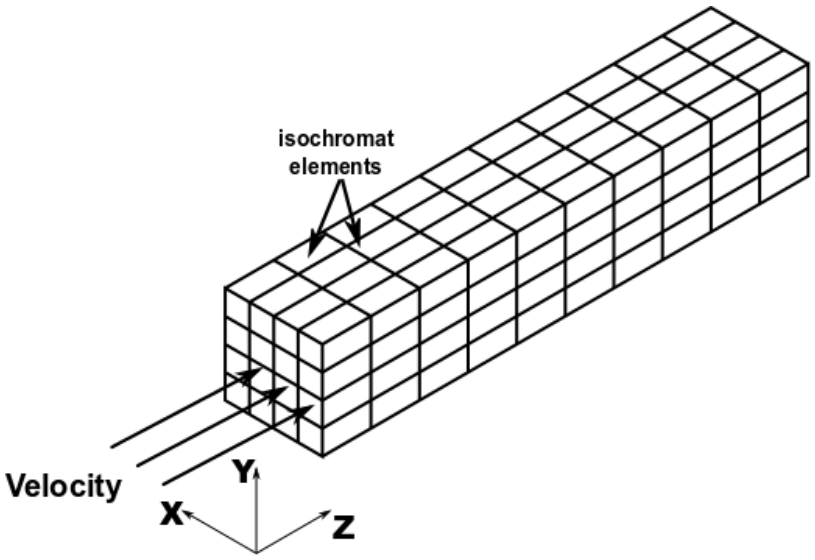

Figure 4.5: Schematic diagram of computational domain.

The computational model is depicted schematically in Figure 4.5. A regular rectangular cell-centered grid is used for the simulation. Each cell in the grid represents one subvoxel. One isochromat is assumed to reside in the centre of each subvoxel. u_z is

associated for each of these cell-centers at every time-step of the simulation to calculate the effect of flow on time evolution of magnetization.

4.4.2 Implementation

The numerical implementation procedure is described in Algorithm 3. The simulator for flowing spins takes pulse sequence data, the computational model and additional MR sequence parameter as inputs like Algorithm 3. The simulator takes also the magnitude of a constant through-plane velocity as an additional input to study the effect of the through-plane plug flow. On the other hand, sum of sinusoids are used to study the effect of pulsatile flow and given by

$$u_z(t) = \sum_i a_i \sin(b_i t + c_i). \tag{4.28}$$

For the evaluation of the pulsatile flow experiments the amplitudes a_i, the angular frequencies b_i and the initial phases c_i were obtained after fitting the pulsatile flow data using curve-fitting as discussed later in Chapter 4.

The initial condition of Equation (4.3) at each cell-centered grid in the computational domain is taken to be $\mathbf{m}_0 = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T$. Computational domain is also chosen in such a way that the magnetizations at inflow boundary also can be assumed to be fully unsaturated i.e. we can take $\mathbf{m}_{\mathbf{u}\cdot\mathbf{n}_\Gamma<0} = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T$. The length of computational domain in the through-plane flow direction can be roughly estimated from the slice profile of the magnetization due to through-plane flow.

After setting the initial and boundary conditions, the time evolution of the magnetizations at all cell-centres are calculated from repetition to repetition and through all the specified time points step-by-step (lines 6–27 in Algorithm 3) in the following order:

(i) For every repetition, orientation of the RF axis θ_{rf} and the spokes θ_s are calculated (lines 7–8 like Algorithm 3.

(ii) B_x, B_y, G_x, G_y are calculated for a repetition (lines 9–10 in Algorithm 3).

(iii) Pulse sequence data $B_x, B_y, G_x, G_y, G_z, ADC$ and input velocity u are calculated for a specific time point (lines 12–13 in Algorithm 3).

(iv) Magnetization transformation are calculated for all the isochromats (lines 14–17 in Algorithm 3).

Algorithm 3 Bloch Equation Simulation for Flowing Spins for One-dimensional Velocity Field

1: **Inputs:**
   $B_1, G, G_z, \tau, ADC \leftarrow$ Readfile($PulseSeqeunceData$)
   $isochromats = r_x, r_y, r_z, \omega, T_1, T_2 \leftarrow$ Readfile($isochromatEnsemble$)
   $nEx, nSpoke, nTurn, baseresol, seqtype \leftarrow$
   Readfile($SequenceSpecifier$)
   $velocitySpecifier \leftarrow$ Readfile($VelocityProfile$)
2:
3: **procedure** BlochFlowSimulation
4: **Initialize:**
   $M_x^p(0), M_y^p(0), M_z^p(0) \leftarrow$ InitializeMagnetization($\mathbf{m}_0$) $p = 1, 2, \cdots, r_x.length$
5: $M_x^p(0), M_y^p(0), M_z^p(0) \leftarrow$ ImplementBoundaryCondition($\mathbf{m}_0$) $p = -l+1, -l+2, \cdots 0$
6: **for** $i \in$ nEx **do**
7: $\theta_{\mathrm{rf}} \leftarrow$ CalculateRFOrientation($seqtype$)
8: $\theta_{\mathrm{s}} \leftarrow$ CalculateSpokeOrientation($nTurn, nSpoke$)
9: $B_x = B_1 \cdot \cos(\theta_{\mathrm{rf}}), B_y = B_1 \cdot \sin(\theta_{\mathrm{rf}})$
10: $G_x = G \cdot \cos(\theta_s), G_y = G \cdot \sin(\theta_s)$
11: **for** $j \in (G.length - 1)$ **do**
12: $pulsedata \leftarrow B_x(j), B_y(j), G_x(j), G_y(j), G_z(j), ADC(j)$
13: $u \leftarrow CalculateVelocity(velocitySpecifier)$
14: **for** $k \in r_x.length$ **do**
15: $isochromat \leftarrow r_x(k), r_y(k), r_z(k), \omega(k), T_1(k), T_2(k)$
16: $M_x^{*k}(j+1), M_y^{*k}(j+1), M_z^{*k}(j+1) \leftarrow$ CalculateSource($isochromat$,
17: $pulsedata, M_x^k(j), M_y^k(j), M_z^k(j)$)
18: **for** $l \in ng$ **do**
19: $M_x^{*l}(j+1), M_y^{*l}(j+1), M_z^{*l}(j+1) \leftarrow$ UpdateBoundary($M_x^{*ibc}(j+1)$,
20: $M_y^{*ibc}(j+1), M_z^{*ibc}(j+1)$)
21: **for** $k \in r_x.length$ **do**
22: $M_x^k(j+1), M_y^k(j+1), M_z^k(j+1) \leftarrow$ CalculateAdvection(u,
23: $M_x^*(j+1), M_y^*(j+1), M_z^*(j+1)$)
24: **if** $ADC(j+1) = 1$ **then**
25: $M_{xy}(i, c) \leftarrow$ CalculateSignal($M_x^p(j+1), M_y^p(j+1)$),
26: $p = 1, 2, \cdots r_x.length, c = 1, 2, \cdots, baseresol$
27: **return** M_{xy}

(v) The boundary conditions are updated extrapolating the in flow boundary conditions to ghost cells (lines 18–20 in Algorithm 3).

(vi) Magnetization transport is calculated at all the cell-centers in the computational domain at the same time step taking the calculated magnetization due to Bloch phenomena as the initial value lines 21–23 in Algorithm 3.

(vii) Integrated complex signals are recorded at specified time points (Lines 24–26 in Algorithm 3).

4.4.3 Parallelization

Algorithm 4 Parallel Subroutine for Calculating Magnetizations in GPU for Flowing Spins

1: **procedure** MAGNETIZATIONCALCULATIONFLOW($isochromat, data$)
2: $isochromatGPU \leftarrow isochromats, pulsedataGPU \leftarrow pulsedata, uGPU \leftarrow u$
3: Setup CalculateMagnetizationKernel execution configuration: $grid_1, block_1$
4: Setup UpdateBoundaryKernel execution configuration: $grid_2, block_2$
5: Setup CalculateAdvectionKernel execution configuration: $grid_3, block_3$
6: Setup CalculateSignalKernel execution configuration: $grid_4, block_4$
7: CalculateMagnetizationKernel $<<< grid_1, block_1 >>> (M_x(j), M_y(j), M_z(j),$
8: $isochromatGPU, pulsedataGPU, M_x^*(j+1), M_y*(j+1), M_z*(j+1))$
9: UpdateBoundaryKernel $<<< grid_2, block_2 >>> (M_x^{*ibc}(j+1), M_y^{*ibc}(j+1),$
10: $M_z^{*ibc}(j+1), M_x^{*gc}(j+1), M_y^{*gc}(j+1), M_z^{*gc}(j+1)$
11: CalculateAdvectionKernel $<<< grid_3, block_3 >>> (M_x^*(j+1), M_y^*(j+1), M_z^*(j+1),$
12: $u, M_x(j+1), M_y(j+1), M_z(j+1)$
13: **if** $ADC(j+1) = 1$ **then**
14: CalculateSignalKernel $<<< grid_4, block_4 >>> (M_x(j+1), M_y(j+1),$
15: $M_{xy}(j+1))$
16: **return** M_{xy}

All the calculations in the loops given in lines 14–17, 18–20, 21–23 and 24–26 in Algorithm 3 can be parallelized. These parts are implemented in CUDA-C subroutines for parallel computation as illustrated in Algorithm 4.

The parallel procedure in Algorithm 4 is described in following steps:

(i) Data related to isochromats and the pulse sequence data i.e. *isochromats*, *pulsedata* and u are transferred from CPU to GPU global-memory (Line 2 of Algorithm 4).

(ii) CalculateMagnetizationKernel, GPU implementation of Lines 16–17 in Algorithm 3

(iii) UpdateBoundaryKernel (Lines 9–10 of Algorithm 4) and CalculateAdvectionKernel (Lines 11–12 of Algorithm 4) are executed at each time step and CalculateSignalKernel (Lines 13–14 of Algorithm 4) is executed when output data has to be recorded.

In order to measure the speed up with GPU, numerical experiments were performed in the same hardware as described in Section 3.3.4, the simulations were carried out for $27 \times 27 \times 45$ isochromats for 258 000 time points and a constant velocity of $20\,\mathrm{mm\,s^{-1}}$ were taken as input. The output data were recorded in 1000 time-points. The run-time was 37.51 s and 2363.13 s for the parallel and the serial versions of the code. Hence, 63 times speed up is achieved approximately.

In order to measure the ratio of execution times of the kernels of magnetization transport and MR term the numerical experiment performance were profiled using GNU profiler. Profiling showed that the execution time for magnetization transport was 75 % of the execution time of the MR term.

4.5 Results

To test the numerical algorithm and the implementation, simulation method is compared with the results in [142] where Yuan et al. studied the effect of RF pulse on the magnetization for through-plane flow of different velocities in the range of 0 to $200\,\mathrm{cm\,s^{-1}}$ using FDM.

Simulations were performed [142] for the pulse sequence depicted in Figure 4.6 with the following parameters: RF pulse of Blackman-windowed sinc pulse (Section 2.4.1) with an amplitude of 0.1750 G and duration of 2.6794 ms, slice selection gradient $G_z = 1.0\,\mathrm{G\,cm^{-1}}$, nominal slice thickness of $2L_s = 7\,\mathrm{mm}$.

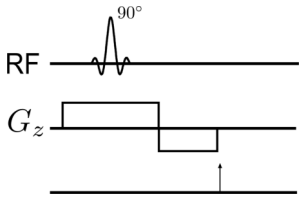

Figure 4.6: A 90° slice-selective pulse was used for the studying the flow-effects. The arrows indicate the time when the data was recorded.

The simulations were carried out for lengths of 20 mm and 30 mm in the slice direction, divided into 800 grid cells of size 0.025 mm and 0.0375 mm respectively. The magnetizations are calculated at the end of the post excitation rephasing gradient. The time duration of the simulations were divided into 4500 time steps with each time step is equal to 8.9313×10^{-4} ms.

In order to evaluate the simulation method used in this thesis, simulations were carried out with same grid size and time-steps. Lax-Wendroff method is used for the simulation of magnetization transport.

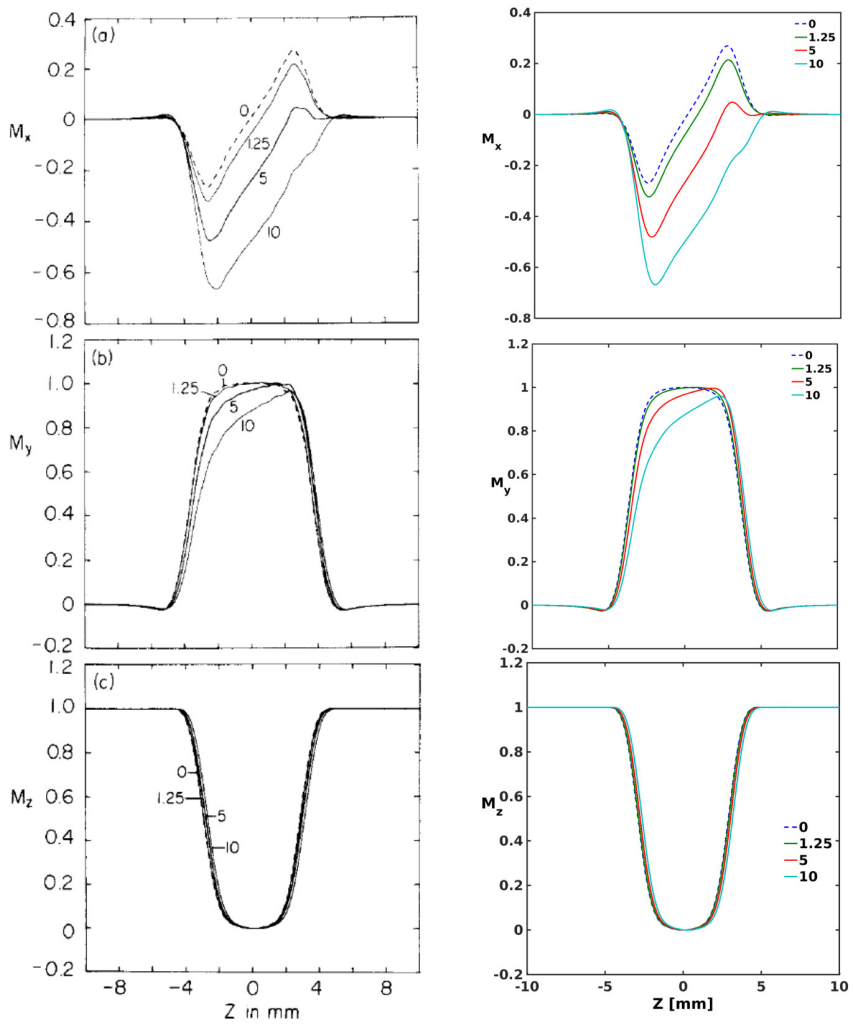

Figure 4.7: Simulated magnetization distributions of M_x, M_y, M_z for the through-plane velocity u_z along the positive z-axis in the range 0 to 10 cm s^{-1} using splitting algorithm in the present work (Right) are compared with the results in [142] (Left) . The magnetizations were recorded at the end of the post excitation rephasing gradient as marked by the arrow Figure 4.6. The length in the slice direction is from −10 to 10 mm.

Figures 4.7–4.9 show an excellent agreement between the results obtained using the Leap-frog method [127, 126] in [142] and the results using the splitting method used in the present thesis.

The plots show a shift for magnetizations along the direction of flow. The effective slice length also increases with increasing velocity. Symmetry of M_x and M_y break with increasing flow velocity as well. Therefore, a proper estimation of slice profile is necessary for choosing the length of computational domain in slice direction, which is

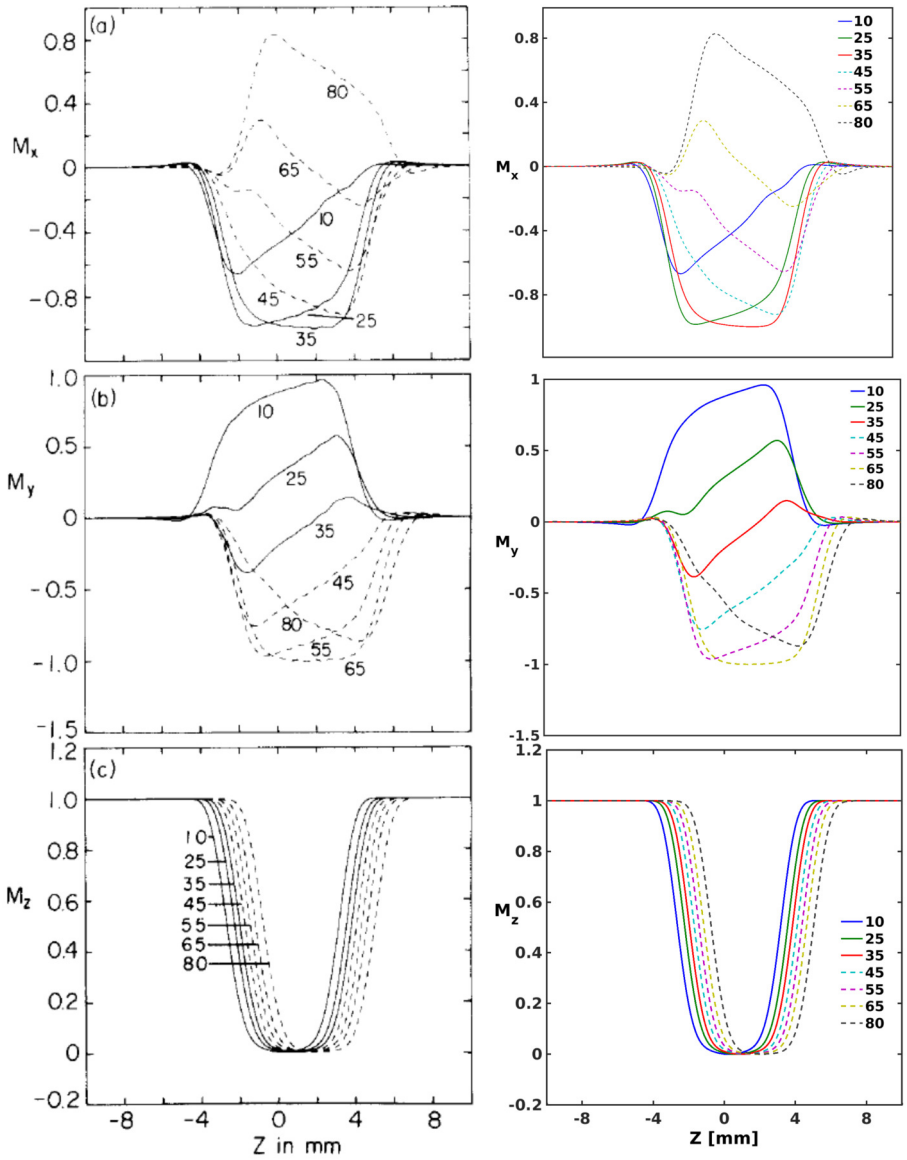

Figure 4.8: Simulated magnetization distributions of M_x, M_y, M_z for the through-plane velocity u_z along the positive z-axis in the range 10 to 80 cm s^{-1} using splitting algorithm in the present work (Right) are compared with the results in [142] (Left) . The magnetizations were recorded at the end of the post excitation rephasing gradient as marked by the arrow Figure 4.6. The length in the slice direction is from −10 to 10 mm.

elaborated and taken into consideration for comparison of simulation with experiments in Chapter 5.

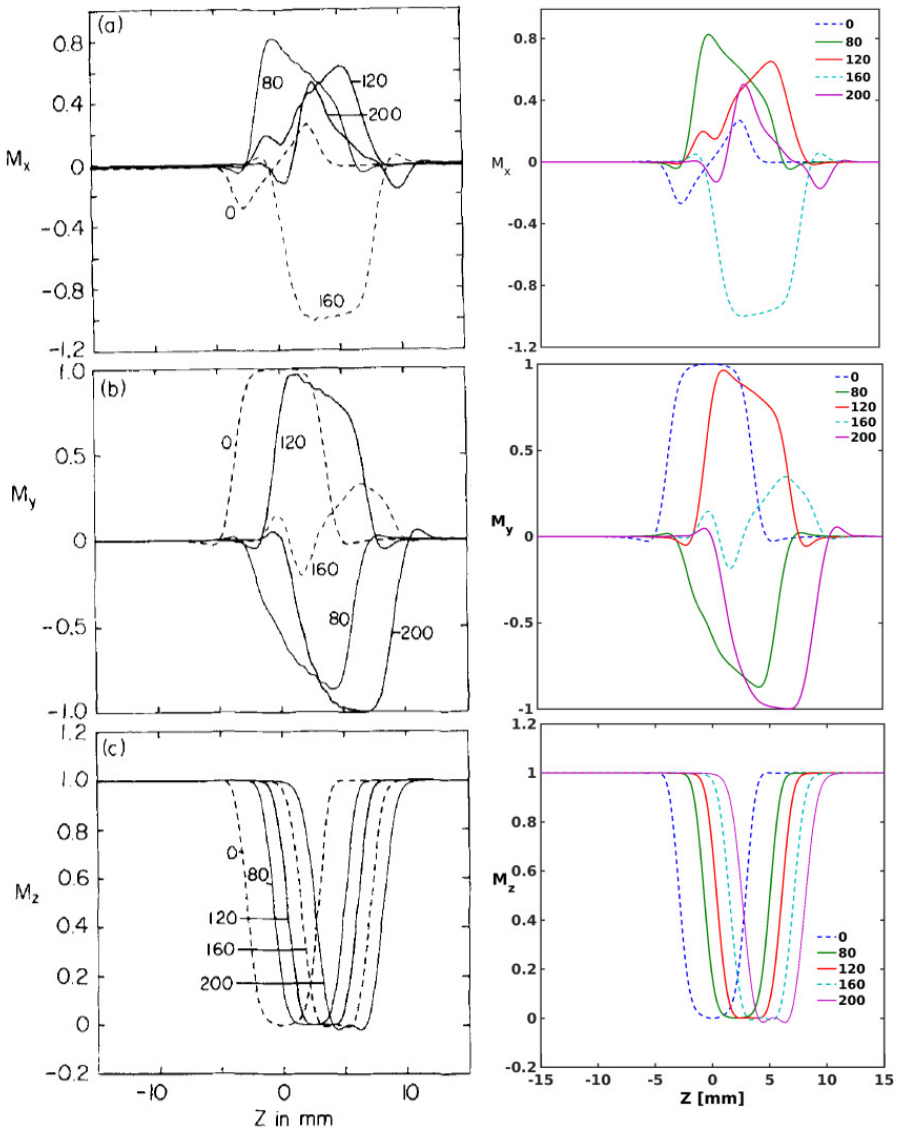

Figure 4.9: Simulated magnetization distributions of M_x, M_y, M_z for the through-plane velocity u_z along the positive z-axis in the range 80 to 200 cm s^{-1} using splitting algorithm in the present work (Right) are compared with the results in [142] (Left) . The magnetizations were recorded at the end of the post excitation rephasing gradient as marked by the arrow Figure 4.6. The length in the slice direction is in the range -15 to 15 mm.

Chapter 5

Comparison of Simulations with Experimental Results

In this chapter, the accuracy of the simulation methods for spatially stationary as well as flowing spin ensembles were evaluated against MRI experiments. In the beginning, the MRI system and phantoms used for the experiments are described. The simulated and measured MR signals are compared in later sections.

5.1 MRI System

The MRI system used in this work is a commercially available Prisma (MAGNETOM, Prisma System, Siemens AG, Erlangen, Germany) as shown in Figure 5.1. It has a bore length of 142 cm and an inner diameter of 60 cm with possible field of view of 50 cm. A superconducting magnet, cooled with liquid helium is used in the MRI system to create a static magnetic field of $B_0 = 2.89\,\mathrm{T}$. It has a two-channel transmit and receiver body coil and a gradient system with maximum gradient strength of $G_{\max} = 80\,\mathrm{mT\,m^{-1}}$. The raster time of the gradients is 10 µs and maximum slew rate is $200\,\mathrm{T\,m^{-1}\,s^{-1}}$. The body coil is built into the structure of the magnet. Apart from that, various receiver coils specialized for imaging different body parts are available. For the experiments in the present work, 64-element head coil, 18-element thorax coil and a single channel loop coil displayed in the bottom part of Figure 5.1 were used.

Figure 5.1: (Top) MRI system and (Bottom) receiver coils. (Bottom left) 64-channel head coil, (bottom middle) 18-element thorax coil, (bottom right) single-channel loop coil.

5.2 Experimental Equipments

5.2.1 Static Phantom

Multiple tubes containing liquids with known T_1 and T_2 (Diagnostic Sonar Ltd., Scotland, UK) were used in the experiments for the validation of the simulation methods for spatially stationary objects. Experiments were conducted both taking the tubes one at a time and placing all the tubes inside a container to mimic more realistic situations as shown in Figure 5.2. The tube numbers and the relaxation times of the liquids contained in the corresponding tubes are listed in Table 5.1.

Figure 5.2: Static phantom with tubes containing liquids with known T_1 and T_2 parameters

Tube #	T_1 [ms]	T_2 [ms]
3	296	113
4	463	53
7	604	95
10	745	157
14	1034	167
16	1276	204
water	2700	2100

Table 5.1: Relaxation constants for the liquids in different tubes

5.2.2 Flow Equipments

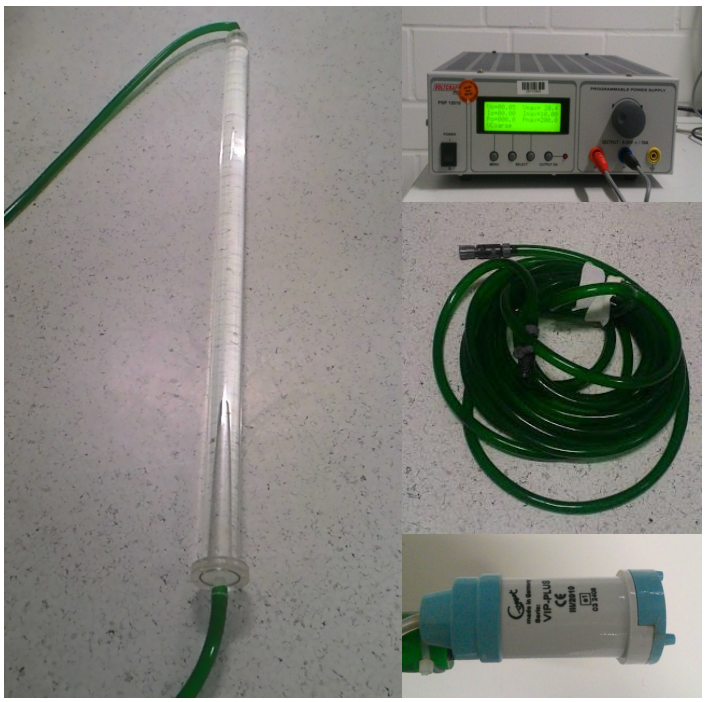

Figure 5.3: (Left) Flow tube made of glass materials. (Right top) Programmable voltage controller. (Right middle) silicone rubber hose. (Right bottom) Flow pump

An in-house acrylic glass tube shown in the left part of the Figure 5.3 was used for the flow experiments. The tube has 5 cm inner diameter and 150 cm length. The flow through the tube was controlled by an electrical pump (Lux Plus KTW270,

Herzog, Göttingen, Germany) which is connected to a computer controlled power supply (Voltcraft 12010, Hirschau, Germany) for adjusting the flow rate. Apart from constant flow of different velocities, pulsatile flow with adjustable profiles could be generated. A thin silicone rubber hose of 20 mm diameter shown in the figure was used to carry the water in the system.

5.3 Validation of the Static Case

In an MRI experiment with a constant flip angle α, constant total gradient moment and constant TR, the magnetizations reach a state of dynamic equilibrium after several TR periods. In order to validate the simulation methods, the transient signal evolution of the experimental results are compared with equivalent simulated results.

5.3.1 Single-channel Loop Coil Experiment

To begin with, the simulation method for spatially stationary objects was tested with the simplest experiment where a single tube containing only a liquid substance underwent an MRI experiment with a single-channel loop coil.

It is important to emphasize that the magnetizations are calculated as a function of time and space in simulations. On the other hand, the discrete time signals are acquired from the object in an MRI experiment. However, the simulation and the experiment can be directly related when the coil sensitivity profile is homogeneous. Equation (2.26) shows under such condition the signal expression reduces to

$$S(t) = \int_{\Omega} M_{xy}(\mathbf{r}, t_{\mathrm{RF}}) \mathrm{e}^{-i2\pi(k_x \cdot x + k_y \cdot y)} d\Omega, \tag{5.1}$$

which implies that the acquired signal has a discrete Fourier transform relation with the transverse magnetizations. Consequently, the relation between the energy of the sampled discrete signals and energy of the discrete transverse magnetizations is given by the following theorem,

Theorem 5.1 (The Plancherel formula for discrete Fourier transform (DFT))**.** *If* $x(m,n)$ *and* $X(k,l)$ *are DFT pairs then*

$$\sum_{m=0}^{M-1} \sum_{n=0}^{N-1} |x(m,n)|^2 = \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} |X(k,l)|^2 . \tag{5.2}$$

Proof. The Proof can be found [83]. □

Theorem 5.1 gives a condition for the validation of the simulation i.e., with a homogeneous coil profile, the energy of the transversal magnetization at discrete sample points should be proportional to the energy of the time discrete signals. This experimental condition is approximately fulfilled when the experiments are performed with a single-channel loop coil of sufficiently small diameter.

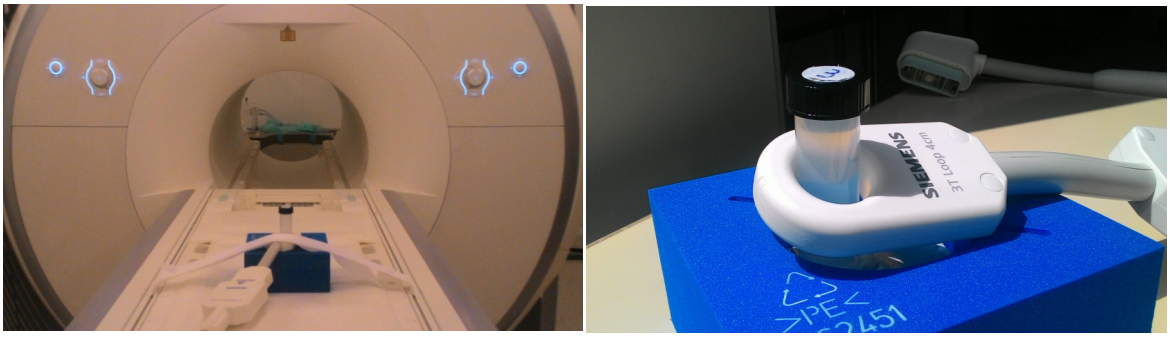

Figure 5.4: MRI experimental set-up with a single channel loop coil and a tube containing a liquid with known T_1 and T_2. (Left) The placement of the tube inside the scanner during the experiment. (Right) The placement of tube inside the loop-coil.

The experiments were conducted taking tubes one at a time from the list in Table 5.1 and placing them inside a loop coil of diameter 4 cm as shown in Figure 5.4. A randomly spoiled radial FLASH sequence with TR/TE = 2.27/1.47 ms, flip angle 8°, in-plane resolution = $1.0 \times 1.0\,\mathrm{mm}^2$, nominal slice thickness = 6 mm, FOV = $32 \times 32\,\mathrm{mm}^2$, number of spokes/turns = 27/5 and base resolution = 32 was used for the experiments.

Simulations were performed for the same pulse sequence data over a static object of $3.0 \times 3.0 \times 18.0\,\mathrm{mm}^3$ divided into $27 \times 27 \times 45$ isochromats taking the relaxation times similar to the specific tube corresponding to each separate run of the experiment. The data were recorded for all the discrete time points as in the experiment.

The experimental and simulated results are plotted for four tubes in Figure 5.5. The simulated results and the experimental results represent the energy of the transverse magnetizations averaged over a frame and the energy of the raw signal per frame respectively. The results were normalized with the corresponding value of the first frame. The figure shows that the results are in a good agreement.

5.3.2 Experiment with Multiple Tubes

In order to apply the simulator to more realistic scenarios, the setup is extended to a phantom with multiple compartments containing different substances. To this

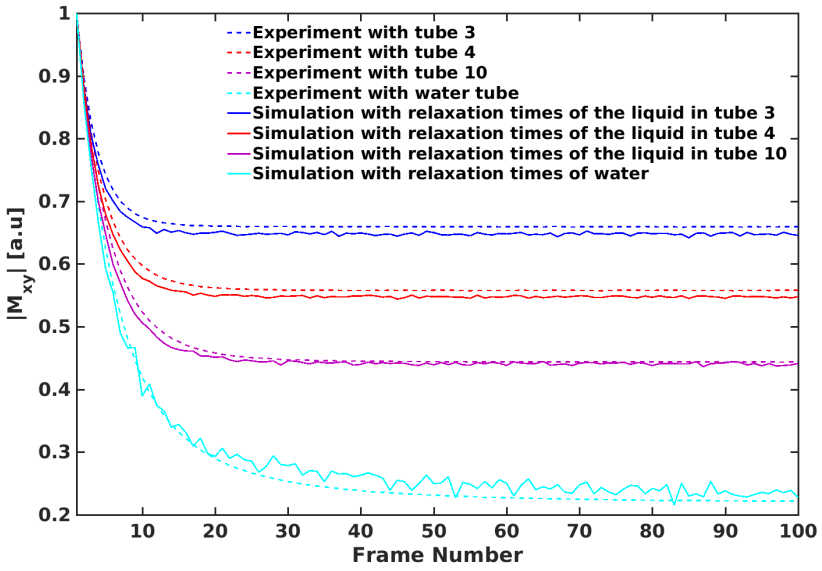

Figure 5.5: Comparison of the normalized energy of the experiment and simulation for four separate tubes.

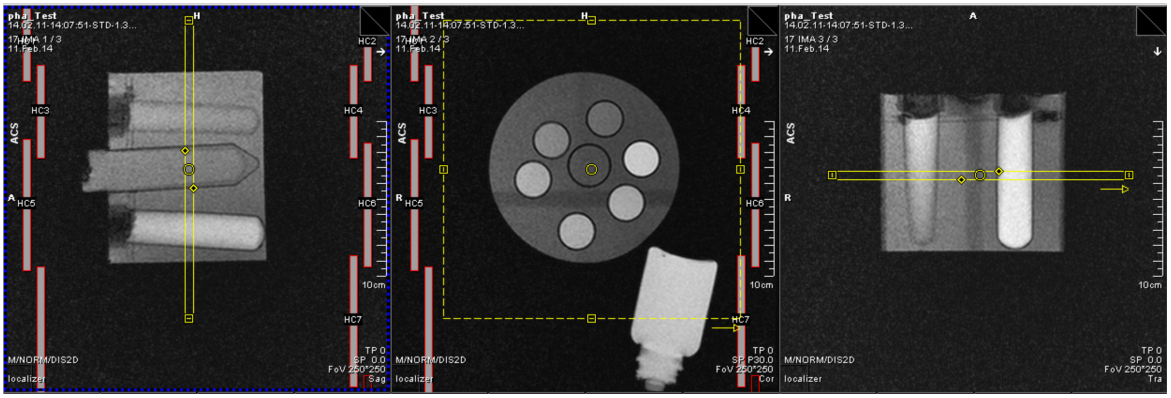

Figure 5.6: Principal setup of the experiment with the static phantom consisting of several tubes with predetermined T_1 and T_2. (Left) Frontal or *coronal* plane view of the phantom. Yellow rectangle represents the slice. (Middle) Axial or *transverse* plane view. Yellow rectangle and the central circle show the FOV and the isocenter respectively. The marker outside the phantom is placed to locate the position of the tubes.(Right) Lateral or *sagital* plane view.

end, an experiment was performed with a container with multiple tubes as shown in Figure 5.2 where each tube contains a separate liquid with known T_1 and T_2 as listed in Table 5.1. The principal setup of the experiments is shown in Figure 5.6. The MRI experiment was conducted with a randomly spoiled radial FLASH sequence with TR/TE = 2.18/1.28 ms, flip angle 8°, in-plane resolution = $1.6 \times 1.6\,\mathrm{mm}^2$, nominal

slice thickness=6 mm, FOV = $256 \times 256\,\text{mm}^2$, number of spokes/turns = 27/5 and base resolution = 160. A 64 channel head coil was used to acquire the signal.

Simulations were performed with the same MRI sequence data as in the experiment over a static object of $4.8 \times 4.8 \times 18.0\,\text{mm}^3$ divided into $27 \times 27 \times 45$ isochromats taking the T_1 and T_2 corresponding to different liquids contained in the tubes of the container. The simulated data is recorded at the TE because the isochromats are completely rephased at TE which implies that the simulated data at TE represent the proton density with zero cumulative phase. The complex signal intensities from the simulated data are averaged over all the isochromats. Then the magnitude of the integrated pixel intensities are averaged over the number of spokes used per frame in the MRI scans. In principle, the averaged integrated pixel intensity is equivalent as the averaged proton density obtained by averaging over a region of interest (ROI) in the image.

The magnitude image of the static phantom and the comparison of simulation and experiment are shown in the left and the right part of Figure 5.7 respectively. For comparison, the magnitude of the signal intensities of different liquids were normalized by the steady-state signal of the tube with maximum signal (i.e., the signal from tube 3).

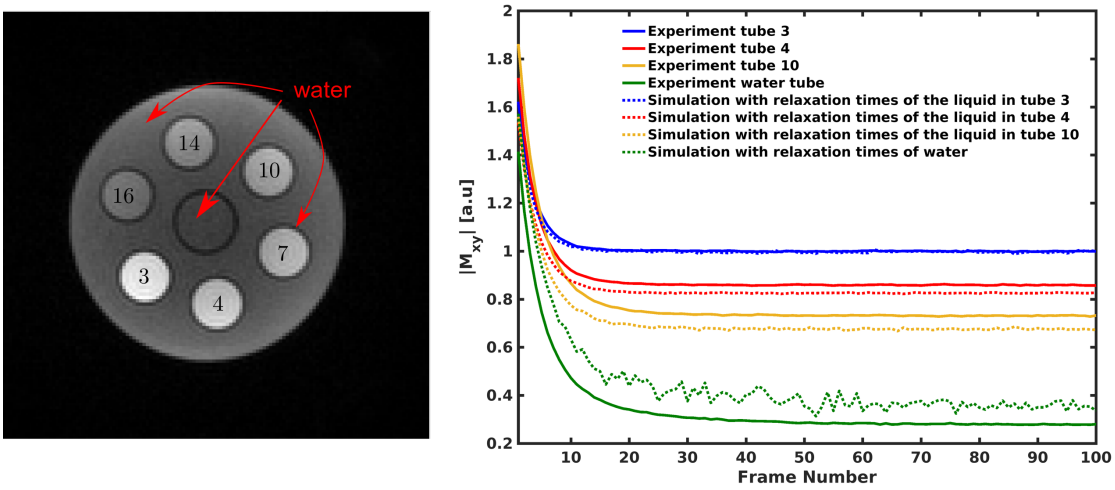

Figure 5.7: (Left) Image of the container. (Right) Comparison of simulation with the image for four different liquids.

The right part of Figure 5.7 shows that there is a deviation between the relative intensities in the simulation and the experiment. The reason for this deviation can be attributed to the inhomogeneous coil profile as can be observed from the image in the

left part of Figure 5.7. However, the simulations were performed under the assumption that the coil profiles are homogeneous. Therefore, for more accurate evaluation of the simulation methods, the effect of the spatial inhomogeneity of the coil profiles needs to be eliminated.

The spatial inhomogeneity of the coil profile is eliminated following the strategy illustrated in Figure 5.8. Following this method, the region which does not contain water is masked out from the image. A cubic polynomial surface is fitted to the signal intensity of the masked out region as depicted in the right part of Figure 5.8. The estimated smooth surface gives approximately the spatial coil sensitivity distribution.

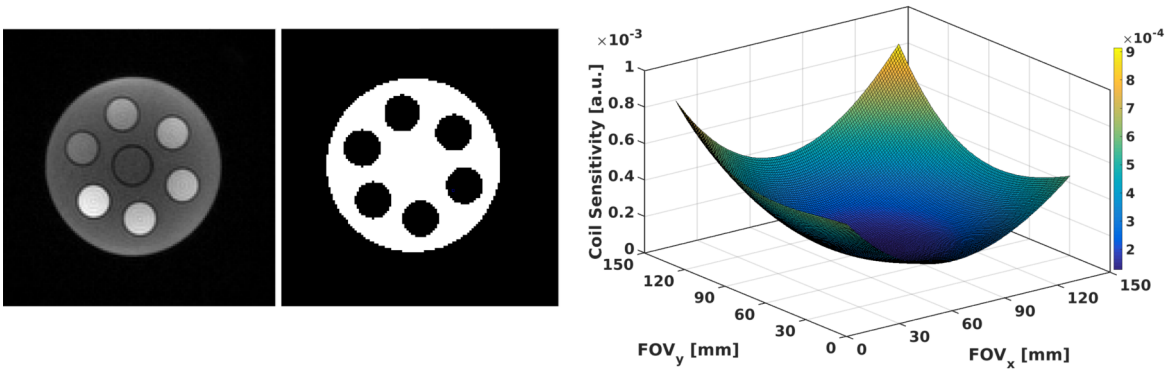

Figure 5.8: (Left) Image with spatially inhomogeneous coil profile. (Middle) White region containing only tap water and the black region is masked out. (Right) Estimated coil profile obtained fitting data over the white region.

The image is divided by the coil sensitivity distribution pixel wise to obtain a compensated image as shown in the left part of Figure 5.9. The signal intensities from different ROIs in the image are normalized like the case with uncorrected coil profile. The simulated and the experimental results show very good agreement after the coil inhomogeneity correction.

5.4 Application of Simulation for Parameter Estimation

The simulation can be applied to estimate parameters. This will be illustrated in this section taking as an example MRI experiments where the effect of the concentration of the contrast agent on the signal enhancement was studied. The experiments were performed by Kollmeier and the detailed discussion can be found in [77].

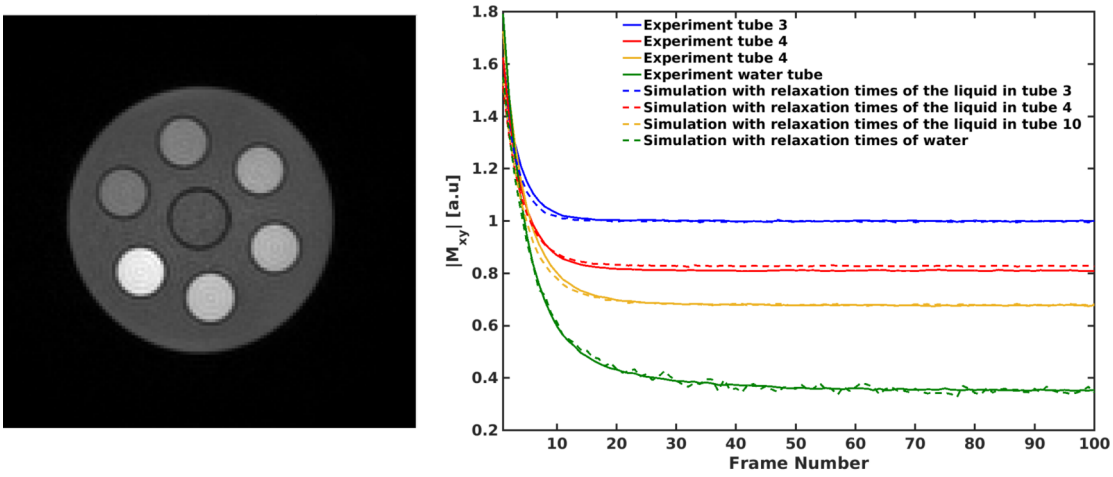

Figure 5.9: (Left) Image of the static phantom after eliminating the coil inhomogeneity effect. (Right) Comparison of simulation with the image, compensated for spatial inhomogeneity, for four different liquids.

The primary aim of a contrast agent is to improve the visibility of the internal structures by radically altering T_1 or T_2^* or both. This can be normally achieved by introducing a tiny amount of the transition metals (Cr, Mn, Fe) and rare earth metals (Gd) based chemical structures.

The effects of contrast agents on the relaxation times are determined by a property of the substance known as relaxivity. The relaxivity of an MRI contrast agent reflects how the relaxation time of a solution changes with the concentration of the contrast agent inside the object. Longitudinal relaxivity r_1 reflects the effect of contrast agents on the T_1 parameter and transverse relaxivity r_2 reflects the effect of the contrast agent on T_2 parameter. The relation between the concentration of the contrast agent and the time relaxations are given by

$$\frac{1}{T_1} = \frac{1}{T_{1r}} + r_1 \cdot [C], \tag{5.3}$$

$$\frac{1}{T_2} = \frac{1}{T_{2r}} + r_2 \cdot [C], \tag{5.4}$$

where T_{1r} and T_{2r} are relaxation times at zero concentration of the contrast agents.

The experiments were performed with *Gadovist* which contains Gadolinium-based Gadobutrol ($C_{18}H_{31}GdN_4O_9$) (Bayer Healthcare, Berlin, Germany) and saline solution of Manganese Chloride ($MnCl_2$). The relaxivities were calculated by spin-echo based relaxometry in the experiments with a 95 %-confidential interval as listed in Table 5.2. $1\,\mathrm{M} = 1\,\mathrm{mol\,l^{-1}}$.

Table 5.2: Relaxavities of the contrast agents

Contrast Agent	$r_1[\mathrm{mM}^{-1}\mathrm{s}^{-1}]$	$r_2[\mathrm{mM}^{-1}\mathrm{s}^{-1}]$
Manganese Chloride	5.6 ± 0.2	86.0 ± 7.0
Gadobutrol	4.6 ± 0.2	5.4 ± 0.2

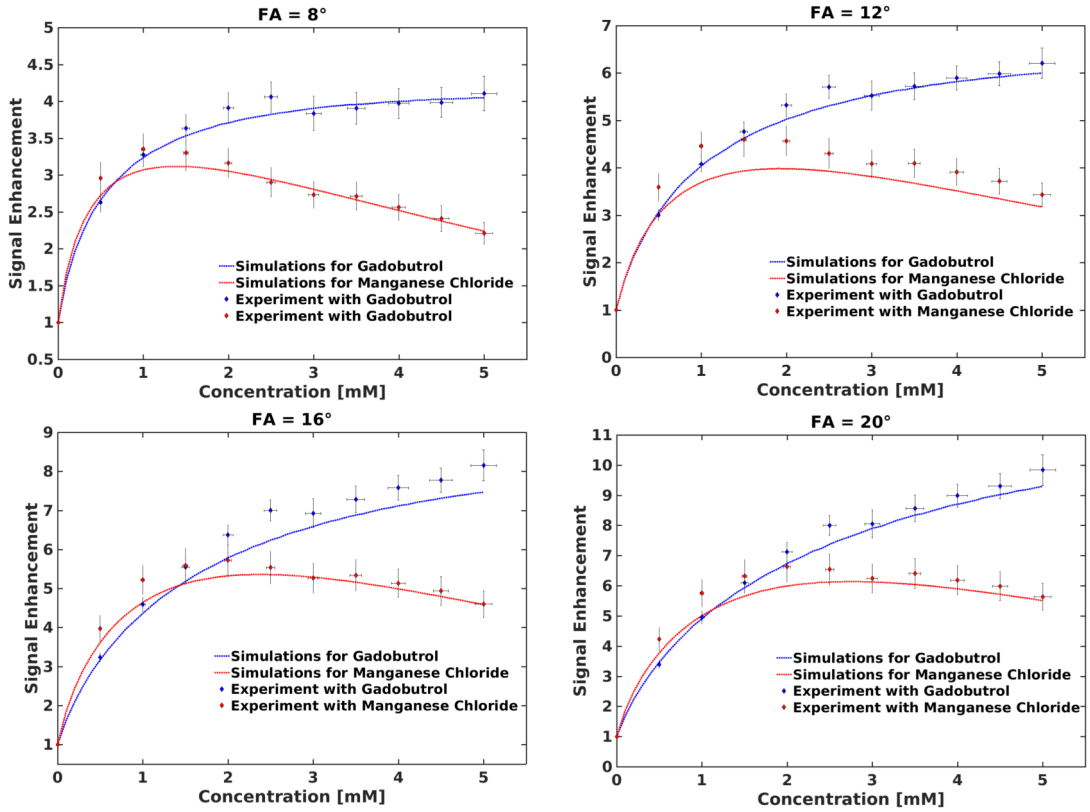

Figure 5.10: Signal enhancement for two different contrast agents for four different flip angles

In the experiments, different concentration of contrast agents were mixed up with water in different water tubes. To study the effect of the concentrations on the signal change, experiments were performed with a randomly spoiled radial FLASH sequence with TR/TE = 2.52/1.78 ms, in-plane resolution = $1.2 \times 1.2\,\mathrm{mm}^2$, FOV = $192 \times 192\,\mathrm{mm}^2$, number of spokes/turns = 51/5 and base resolution = 160, nominal slice thickness = 8 mm. The same MRI sequence was used for four different flip angles 8°,12°,16°,20°.

Simulations were carried out with identical pulse sequence data over a static object of $3.6 \times 3.6 \times 24.0\,\mathrm{mm}^3$ divided into $27 \times 27 \times 45$ isochromats with T_1 and T_2 calculated for specific concentrations of the contrast agent from Equations (5.3)–(5.4) and Table 5.2.

The images obtained from the experiments were first corrected for coil inhomogeneity as discussed in the previous section. The magnitude of the signals in dynamic equilibrium were plotted as a function of concentration for four different flip angles in Figure 5.10. The good agreement between the experimental and the simulation results suggests the possible use of the simulator for prediction as well as for parameter estimation purposes.

5.5 Evaluation of the Simulation of MRI for Flowing Spins

As mentioned earlier, one of the main goals of the present work is to develop a simulator to study the effect of flowing spins on the MRI signal evolution with a focus on the quantitative analysis of flow induced dynamic signal changes. The flow situations typically encountered in human bodies such as blood flow through heart, arteries and veins, cerebrospinal fluid (CSF) flow through brain and spine are generally pulsatile in nature. Also, the flow velocity vary over a wide range of magnitude and complexity. However, in this introductory work, the simulator for flowing spins was evaluated against *in vitro* experiments, performed for relatively simple laminar and pulsatile flow situations. For evaluation of the simulation methods, effects of plug and pulsatile flow on the magnitude of MR signals were compared. The focus of the present work was also limited to the flow along longitudinal directions i.e. through-plane flow.

5.5.1 Proof of Concept

For constant through-plane flow velocities, the MR signal from the spins in the excited slice get replaced by fresh unsaturated spins before the next RF excitation, resulting in a signal-enhancement. Depending on the velocity of the through-plane flow u_z, the saturated spins are replaced partially or fully. The magnitude of the signal enhancement increases with an increment in through-plane velocity in the partially saturated region, given by $u_z < \frac{TH}{TR}$. Beyond this region of velocity, the excited spins are replaced fully by unsaturated spins. However, this calculation is based on the simplified assumption that the flow does not have any influence on the location and shape of the slice profile [54] whereas the flow influences the location and shape of the slice significantly.

In order to illustrate the effect of through-plane flow on the location and shape of the slice profile, simulations were performed with a randomly spoiled radial FLASH sequence with TR/TE = 3.33/2.10 ms, flip angle = 8°, in-plane resolution= $0.75 \times 0.75\,\mathrm{mm}^2$,

FOV = $192 \times 192\,\mathrm{mm}^2$, number of spokes/turns = 15/5 and base resolution = 256, nominal slice thickness = 5 mm. A computational domain of $11.25 \times 11.25 \times 15.0\,\mathrm{mm}^3$ divided into $75 \times 75 \times 45$ subvoxels was chosen for the simulation with fluids having relaxation times similar as CSF i.e., $T_1 = 2000\,\mathrm{ms}$, $T_2 = 300\,\mathrm{ms}$ [24]. The simulation is carried out for a range of through-plane velocities u_z from 0 to $400\,\mathrm{mm\,s^{-1}}$.

The integrated pixel intensities averaged per frame at TE are plotted as a function of length in slice direction in Figure 5.11. The left part of the figure shows how the magnitude signals evolve from the first frame to dynamic equilibrium for velocities 0, 50, 100 and $400\,\mathrm{mm\,s^{-1}}$. The right part of the figure shows slice profiles at dynamic equilibrium for a range of velocities from 0 to $400\,\mathrm{mm\,s^{-1}}$. The left figure shows that with increasing magnitude the flow velocity dominates the slice profile and there is no significant change of slice profile with time. It can be observed from the right hand figure that the slice profile shifts and expands with increasing magnitude of velocities.

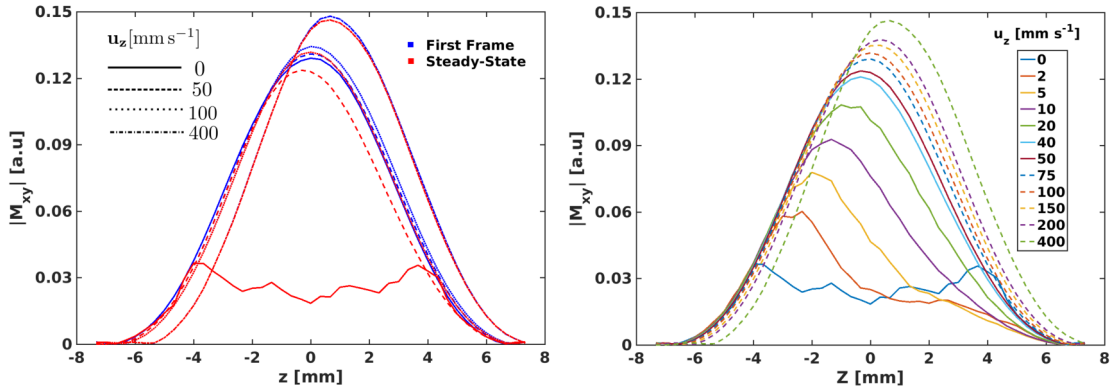

Figure 5.11: (Left) Slice profile at TE for the first frame and in dynamic equilibrium for 0, 50, 100 and $400\,\mathrm{mm\,s^{-1}}$ through-plane velocities. (Right) Slice profile at TE in dynamic equilibrium for through-plane velocities in the range of 0 to $400\,\mathrm{mm\,s^{-1}}$.

Apart from that, in order to estimate velocities from the magnitude of MR signal in dynamic equilibrium, the magnitude of MR signal should be clearly distinguishable from each other for different velocities. To estimate a velocity range for u_z where the signal-enhancement is sensitive to an increase in velocity, simulated results are analysed in Figure 5.12.

The left part of Figure 5.12 shows the transient evolution of the magnitude of signal for different velocities and the right part shows the steady-state signal intensities normalized by the steady-state signal of the static fluid. The figure depicts that the steady-state signal increases very rapidly in a very slow flow range and the rate of signal enhancement decreases with an increase in the through-plane velocity beyond an

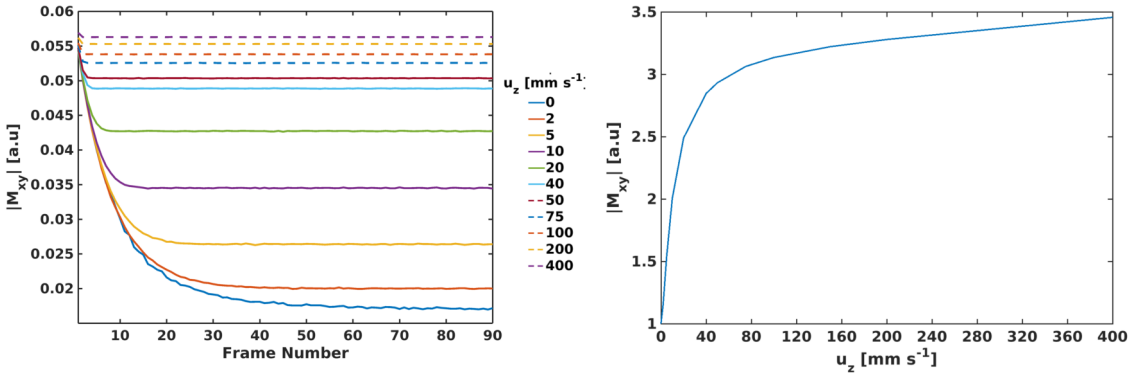

Figure 5.12: (Left) Time evolution of averaged integrated pixel intensities as a function of frame for through-plane velocity range 0 to 400 mm s^{-1}. (Right) Normalized steady-state integrated pixel intensities as a function of constant through-plane velocities.

initial range. In that initial range, the signal enhancement induced by through-plane flow can be estimated reliably from the magnitude signal.

The remarkable sensitivity of the magnitude of the MR signal to slow flow is supported by a recent study in CSF flow [33] where a rapid increase in the magnitude of the signal could be observed for a very small increase in the inspiration-induced velocities. Therefore, the present work focused further on *in vitro* experiments with slow flow in the range of 0-100 mm s^{-1}.

5.5.2 *In Vitro* Experiments with Laminar Flow

In the beginning, an attempt was made to evaluate the simulator against a laminar flow which can be expressed as a parabolic function of the length in radial coordinate for flow in a circular tube. To this end a laboratory experiment was performed with the aim to create a fully developed laminar flow which can be calculated with relative ease and can be implemented easily as well. The MR image of the experimental setup can be seen in the left part of Figure 5.13. The bright circle in the middle and the small bright circle in the right top corner in the figure are the images of the acrylic glass tube and the guiding rubber hose as illustrated in Figure 5.3.

The relative signal intensities in dynamic equilibrium resulting from different velocities are compared with simulations in the right part of Figure 5.13. From a previous calibration the operating range of the flow pump was chosen such that the through-plane velocities were expected to be in the laminar flow region (i.e, Reynolds number $Re \leq 2300$ [113]).

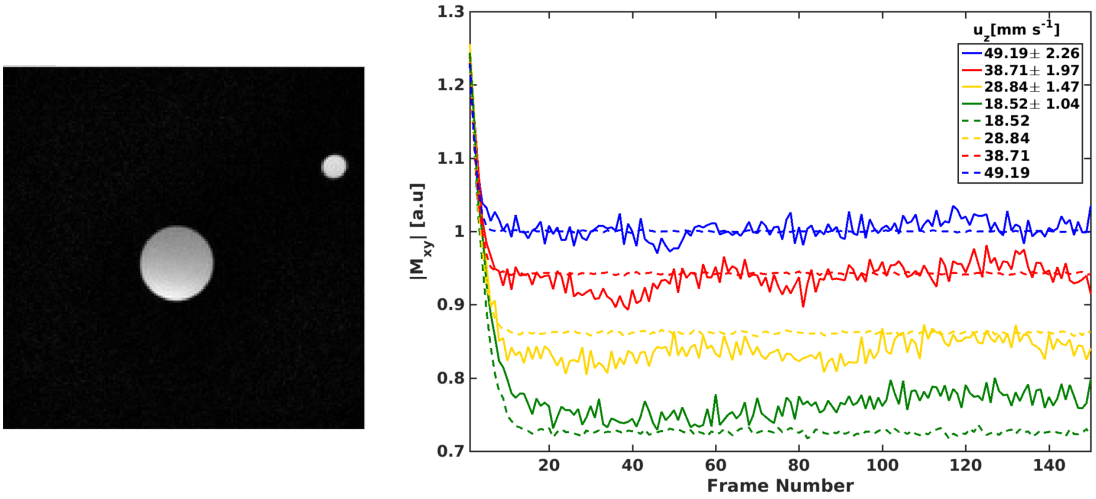

Figure 5.13: (Left) MR image of the flow tube. (Right) Signal comparison normalized by the signal intensity of the averaged steady-state signal in dynamic equilibrium for last ten frames for different velocities under different operating conditions of the flow pump. The experimental result is represented by the solid line and the simulation is represented by the dotted line.

At each listed operating voltage in Table 5.3, measurements were performed with a randomly spoiled radial FLASH sequence with TR/TE = 1.96/1.22 ms, flip angle = 8°, in-plane resolution = $1.6 \times 1.6\,\mathrm{mm}^2$, FOV = $256 \times 256\,\mathrm{mm}^2$, number of spokes/turns = 17/5 and base resolution = 160 and nominal slice thickness = 6 mm. A very small ROI was chosen in the image such that the coil sensitivity profile over that region can be assumed to be homogeneous.

A PC MRI sequence [72] was used for pixelwise estimation of the input velocities for the simulation. The sequence parameters used for estimating the flow velocities were: TR/TE = 5/4.32 ms, in-plane resolution = $1.6 \times 1.6\,\mathrm{mm}^2$, FOV = $256 \times 256\,\mathrm{mm}^2$, number of spokes/turns = 7/5 and base resolution = 160, nominal slice thickness = 6 mm, flip angle 10°. At each listed operating voltage in Table 5.3, the mean through-plane flow velocity was calculated over the chosen ROI.

The mean velocities and the standard deviation for different operating voltages are listed in the middle and the right column of Table 5.3. The standard deviation suggested a possible unsteady flow profile.

The simulation was carried out taking the mean velocity as input constant velocity for the simulation. To evaluate the simulation methods the magnitudes of relative signal intensities were compared. A computational domain of $4.8 \times 4.8 \times 18.0\,\mathrm{mm}^3$ divided into $45 \times 45 \times 45$ subvoxels was chosen for the simulation with the pulse sequence

Table 5.3: Mean and standard deviation velocities and *Re* based on the mean velocity for different operating voltages of the flow pump at temperature 16 °C (kinematic viscosity $\nu = 1.1092 \times 10^{-2}\,\mathrm{cm^2/s}$ [137]).

Voltage [V]	Mean Velocity [$\mathrm{mm\,s^{-1}}$]	Standard Deviation [$\mathrm{mm\,s^{-1}}$]	*Re*
6	49.19	2.26	2217
5	38.71	1.97	1744
4	28.84	1.47	1300
3	18.52	1.04	834

data of the randomly spoiled radial FLASH sequence. A high resolution method with superbee limiter was used for the calculation of the magnetization transport at each time discretization step which corresponds to $0.5 \times 10^{-5}\,\mathrm{s}$ during the RF pulse and $1.0 \times 10^{-5}\,\mathrm{s}$ for rest of the duration of a TR period. For the calculation of magnetization evolution at the same time step one isochromat was assumed to reside at the centre of each subvoxel.

The experimental and the simulated data are compared in the right part of Figure 5.13. The figure shows that the magnitude of signal from the experimental data never reaches a dynamic equilibrium in the experiment due to the unsteady velocity profile as already hinted by PC MRI calculation. The experimental data was normalized by the average magnitude of signal of last 20 frames for the measurement with the operating voltage of 6 V in Table 5.3. The simulated data was normalized similarly taking the corresponding velocity.

The reason for the deviation of experimental results from simulation depicted in the plot could be attributed to the unsteady flow profile which was already hinted by the standard deviation in the velocity calculation using PC MRI. A more elaborate picture of unsteady flow profile can be observed in the contour plots of the flow velocities. Figure 5.14 depicts the velocity contours inside the tube in three consecutive velocity measurements in dynamic equilibrium. The through-plane velocities were assumed to be constant over the time duration of each measurement (i.e. 70 ms). The contour plots in Figure 5.14 clearly show that the flow field never became a fully developed laminar flow in the existing setup in spite of the fact that the average velocities were in the laminar range because the laminar flow would have produced concentric circular contours. The reason of not obtaining a fully developed laminar flow can be attributed to the fact that a sufficiently big entry length could not be provided in the existing setup. Empirical studies showed that to guarantee a laminar flow profile an entry length $l_e \approx 0.05ReD$ must be provided [26], which would result in approximately 2 to 5.5 m length in the used velocity range. The contour plots in the figure also suggest

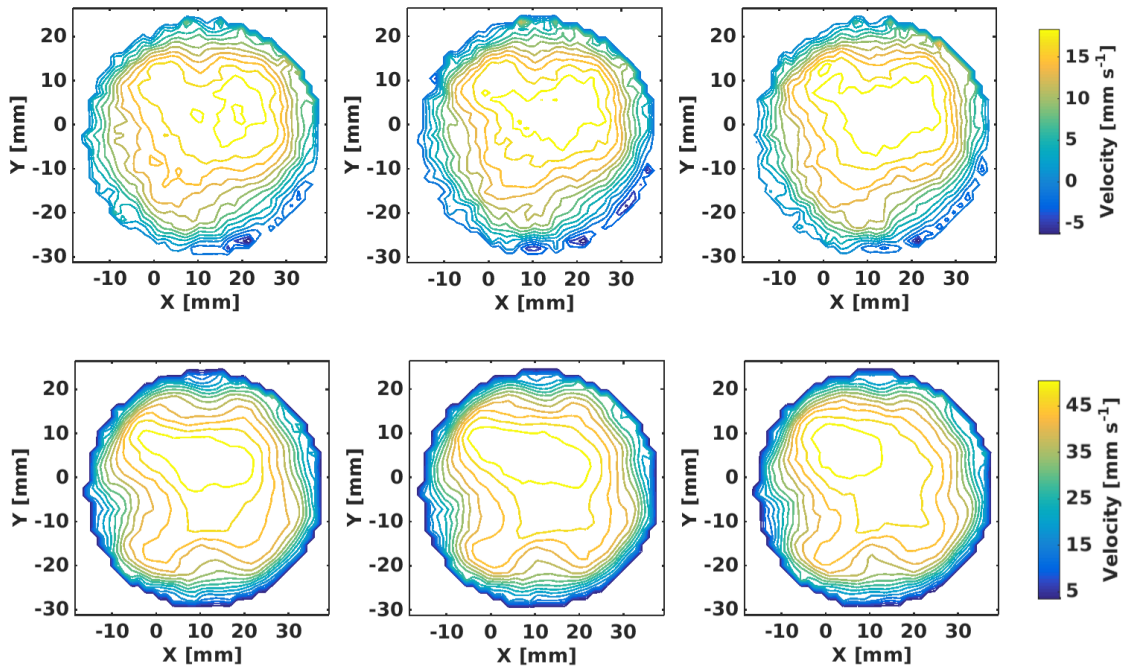

Figure 5.14: The velocity contours in the tube for three consecutive measurements when the pump was operated at 3 V (top) and 6 V (bottom).

the presence of transverse flow components during the experiments. Therefore, the assumptions of pixelwise constant through-plane velocity and one-dimensional flow on which the simulations were based, were marginally deviating from the experiments.

5.5.3 *In Vitro* Experiments with Pulsatile Flow

The simulation method was also evaluated against more relevant pulsatile flow using a laboratory experiment. The pulsatile flow experiment setup is depicted in Figure 5.15. In order to create a pulsatile flow profile in the flow tube the flow pump was operated with a periodic voltage time diagram as shown in the left part of Figure 5.16.

A randomly spoiled FLASH sequence with exactly the same parameters as used for the similar experiments in the previous section was used. A very small ROI was chosen to calculate the mean magnitude signal and later to calculate the velocities with PC MRI such that coil sensitivity profile could be assumed homogeneous. Sum of 4 sinusoids given by Equation (4.28) were fitted to the estimated mean through-plane flow velocity data using Matlab (Mathworks, Natick, MA, IUSA) curve-fitting toolbox. The calculated flow velocity profile and the fitted curve are depicted in the right part of Figure 5.16. The simulation is performed with same domain and subvoxels as used

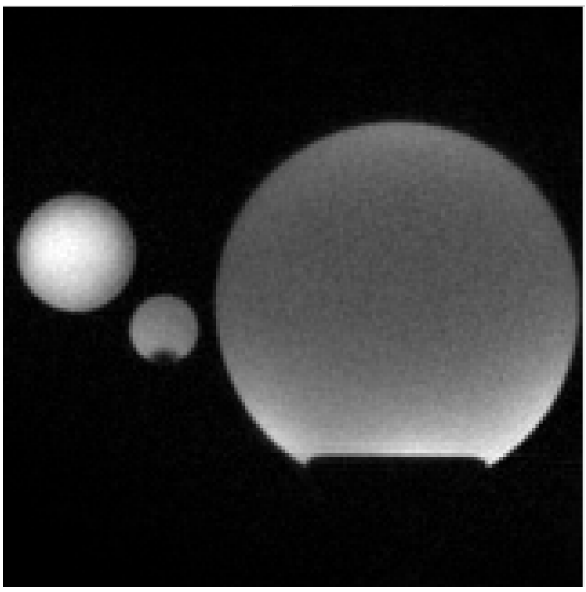

Figure 5.15: Experimental setup for the pulsatile flow experiment. Water flowed through the left tube. Middle and right tubes contain static tap water.

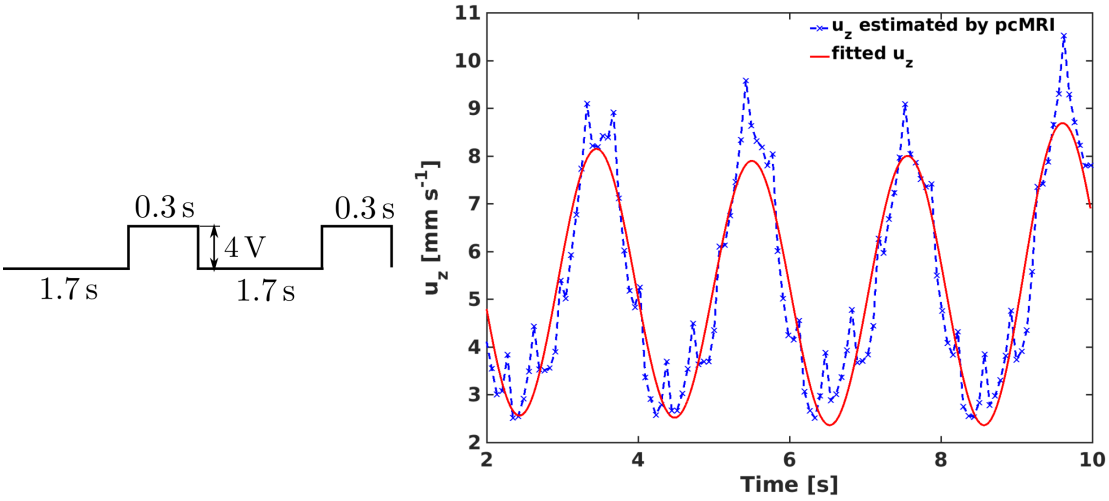

Figure 5.16: (Left) Operational voltage diagram of the pump to create pulsation. (Right) Fitted through-plane velocity profile from PC MRI data.

for the previous section taking the pulsatile velocity profile as the input through-plane velocity. Experimental data was normalized with the steady-state signal from the spatially stationary tube under the same experimental conditions. Simulation were normalized also by the steady-state signal of static water.

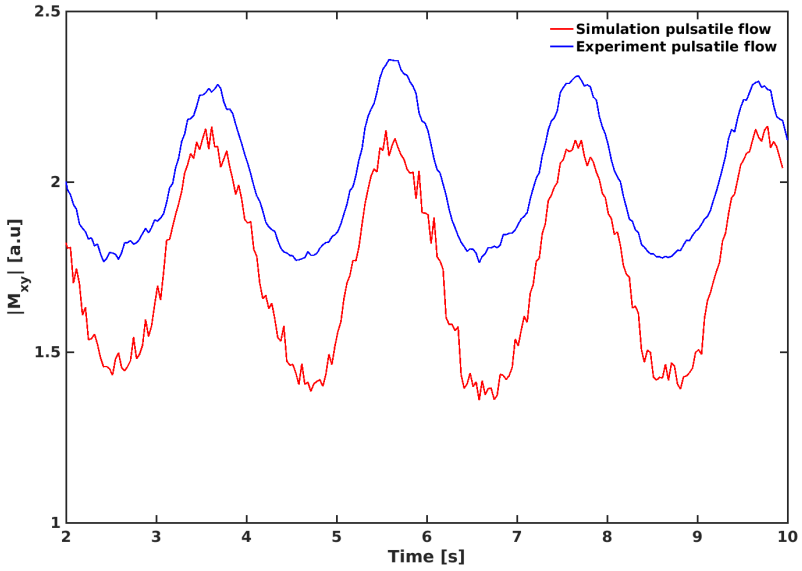

Figure 5.17: The effect of pulsatile flow on the signal can be observed here. Signal is normalized with the steady-state signal of the static water.

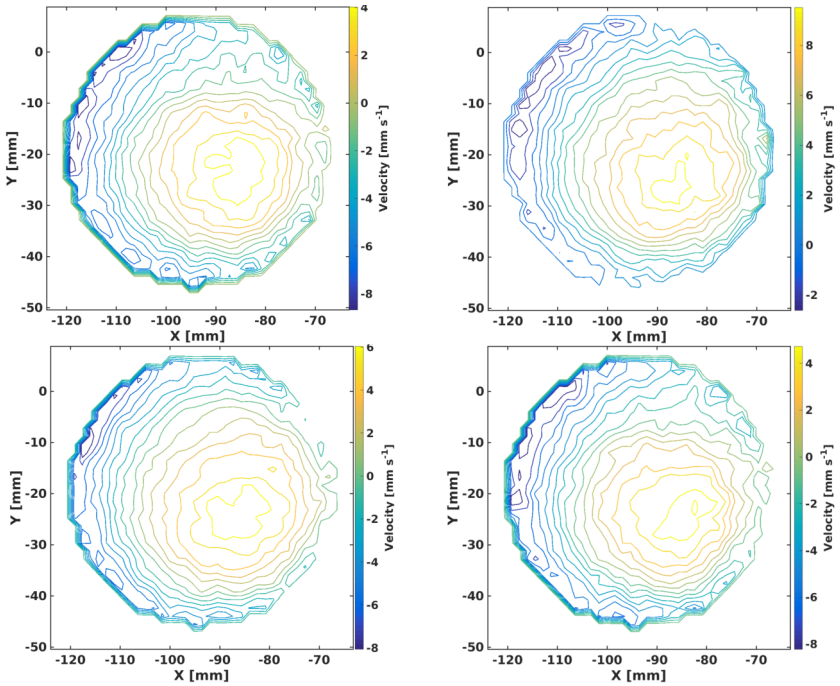

Figure 5.18: (From top right clockwise) Contour plots for four equidistant different time point in one pulsation period.

Figure 5.17 shows that although the periodicity in the magnitude signals from the experiment and the simulation agree well, the amplitude of experimental results deviate marginally from the simulation.

The deviation could be due to the flow profile implemented in the simulation were assumed to be a function of time only. The assumption implies that fluid at different positions along the tube must respond simultaneously to the changing pressure at all positions at every specific point of time in the direction of through-plane flow such that through-plane velocity at every position in the longitudinal direction are same. In order to fulfil the condition, the fluid is assumed to be moving in bulk which is artifical and unphysical [143]. Nevertheless, this assumption provides a starting point for understanding more realistic form of pulsatile flow. Moreover, the flow profile was never a one-dimensional pulsatile flow profile as can be observed from Figure 5.18 which shows the velocity contour in the flow tube in four time points with 700 ms time duration difference.

Chapter 6

Summary and Outlook

6.1 Summary

In this thesis, a numerical simulator has been developed for a quantitative description of the MRI signal of spatially stationary and flowing spins. The approach is based on a numerical solution of the Bloch equation. The simulation methods were validated with laboratory experiments. The simulated results hint at the possibility of computer-aided estimation of experimental parameters like flow velocity or NMR relaxation time constants as well as offer support and potential for further improvements.

Technically, a splitting method was used in order to solve the Bloch equation for spatially stationary objects. the method splits the equation into two sub-operators representing the rotation and relaxation of the transverse and longitudinal magnetizations involved in an MRI experiment. Subsequently, these two subproblems were solved analytically and combined step-by-step in certain order to give solutions with different degrees of accuracy. Another advantage of the operator splitting technique is that it reduces the computation time in comparison to other numerical methods like Runge-Kutta (RK).

The splitting technique was further extended to solve the Bloch equation for flowing spins which represents an advection-reaction equation. The time evolution of the relevant magnetizations due to the MR experiment and transport due to the presence of a flow field were calculated sequentially. Therefore, the transport of magnetization was added to the simulator for static objects to develop the simulator for flowing spins. High-resolution FVM methods were used to solve the magnetization transport. One of the major advantages of FVM is that the simulator can be easily extended to complex and moving geometries which often refer to the situation for flow under

in vivo conditions. In this initial study, the effect of a one-dimensional flow field with either constant or pulsatile velocities on the temporal evolution of the MRI-recorded magnitude signal was investigated. However, the present work already lays the foundation for more realistic simulation of complex flow dynamics as, for example, encountered in the ascending aorta of patients with aortic valve insufficiency and/or partial stenosis. The simulators were further parallelized for CUDA-enabled GPU to reduce the simulation time significantly. The computational domain for the simulation methods was chosen such that realistic slice profiles for real-time MRI acquisitions were taken into consideration and the time steps were chosen to comply with the raster time of the used MRI pulse sequence.

The simulation methods were validated for a randomly spoiled radial FLASH sequence and experiments in a 3 Tesla MRI system. The simulator for spatially stationary objects was tested with use of a single-compartment phantom filled with doped water to achieve defined relaxation times. The simulations were then extended to a multi-compartment phantom containing several aqueous solutions with different relaxation times. For both of these cases the simulated data agree well with the experimental results.

The use of numerical simulation for estimating experimental parameters underlying a certain MRI signal strength (or its change) was explored for the effect of a paramagnetic contrast agent and its related signal enhancement. The simulation was performed with relaxation times estimated using T_1 and T_2 mapping sequences and subsequently compared with the experimental results. The generally good agreement indicates the possibility of using the numerical simulator for parameter estimation.

In a final step, laboratory experiments were conducted for testing the simulator for slow constant and pulsatile flow. The experimentally estimated flow velocity via PC MRI was taken as input for the simulation. Even though the experimental conditions were not sufficient to produce a situation of perfectly laminar flow, the simulated results show reasonable agreement with the experiments. For looking at a pulsatile flow pattern, the experimental velocities obtained by PC MRI, fitted to a periodic flow pattern as a function of time, were taken as input. Again, the simulated and experimental results agree well in a comparable range regardless of the simplified assumption of temporal periodicity in the one-dimensional flow profile. However, the simulator already provides potential for further improvement towards the treatment of more complex flow.

6.2 Outlook

In this thesis the simulation focused on a one-dimensional ensemble of pixels with homogeneous MR properties. This offers the advantage of direct comparisons of integrated pixel intensities from the simulations with experimental MRI results. In general, however, the simulator can be extended to model the time evolution for series of two-dimensional MR images with spatially stationary as well as flowing spins. For such situations the spatially variable coil sensitivity profiles which are factors to the actual MR image must be taken care of, either by elimination as possible for a single-coil acquisition or by the incorporation of analytically or experimentally determined coil sensitivities. Moreover, when studying compartments with irregular geometries, a reliable treatment should consider partial volumes inside pixels.

Furthermore, the time scales for resolving the magnetization transport and MR phenomena are different. The splitting technique can be tested by taking different time scales for solving Equation (4.3) which may reduce the simulation time significantly. Accuracy in time discretization can also be increased by using higher-order time stepping such as TVD RK time stepping [48] or strong stability preserving linear RK methods [49].

To cater for more realistic scenarios involving pulsatile and turbulent flow as seen for vascular and cardiac flow in humans, the flow field must be extended to two and three dimensions. Equation (4.13) in Section 4.3 discusses a general formulation to study multi-dimensional flow fields. The easiest way to investigate the effect of multi-dimensional flow on the MRI signal evolution is to employ dimensional splitting techniques [68, 81]. In dimensional splitting, the effect of flow components of all directions are calculated separately and subsequently combined. Another possibility is to solve the transport of magnetization from the semidiscrete form of Equation (4.13) evaluating the fluxes by some multi-dimensional interpolation [81]. Higher-order time stepping techniques can simultaneously be used along with a multi-dimensional flow-field implementation.

As a first extension of the current work, accurate pulsatile flow fields in time and space should be incorporated to alleviate the simplified assumption of temporal periodicity used so far. In the next step, turbulence modelling in time and space needs to be employed for more complex flows. In addition, contributions from vessel movements should be incorporated for more precise simulation. Readily available commercial or open-source computational fluid dynamics (CFD) softwares can be used for simulating complex flows. Later, the temporally and spatially dependent flow fields

can be taken as input and integrated in the simulator for flowing spins to study the effects on MRI.

With increasing complexity of the flow field, the computational task will be more demanding. As a result massive parallelism will be essential. In this work, single-GPU parallelism was used to get a reasonable speed up. However, further improvements are possible in this direction. Precise optimization techniques and multiple implementation can be employed to reduce the simulation time significantly.

Regarding putative applications, clinical scenarios which increasingly rely on quantitative information should be further explored, in particular for a computer-aided parameter estimation in T_1, T_2, T_2^* mapping. The current achievement of estimating flow velocities and volume rates from experimental MRI data with use of a simulator for moving spins warrants more extensive scientific and clinical trials ranging from artifical phantoms to normal and pathological flow in the large vessels of patients.

Appendix A

Definitions, Theorems and Results Related to ODE Systems

In this chapter, the focus is on the analysis of numerical methods for first-order initial value problem (IVP) of the form:

$$\mathbf{w}'(t) = \mathbf{f}(t, \mathbf{w}(t)), \quad t \in (t_0, t_0 + T], \tag{A.1a}$$

$$\mathbf{w}(t_0) = \mathbf{w}_0, \quad \mathbf{w}_0 \in \mathbb{R}^d. \tag{A.1b}$$

We are seeking a vector-valued function $\mathbf{w}(t) \in C^1[t_0, t_0 + T]$ with the initial value Equation (A.1b). The d-dimensional Eucledian space $\mathbb{R}^d$ is equipped with norm $\|\cdot\|$.

The general existence and uniqueness result of the equation is given by the following theorem.

Theorem A.1 (Picard–Lindelöf theorem)**.** *Suppose that the vector-valued function* $(t, \mathbf{w}) \mapsto f(t, \mathbf{w})$ *is continuous in the domain* D *defined by* $t_0 \leq t \leq t_0 + T$, $\|\mathbf{w} - \mathbf{w}_0\| \leq M$ *such that* $\|f(t, \mathbf{w}_0)\| \leq K$ *when* $t_0 \leq t \leq t_0 + T$ *and that* f *satisfies the Lipschitz condition:*

$\exists L > 0$ *such that* $\|\mathbf{f}(t, \mathbf{w}) - \mathbf{f}(t, \mathbf{w}^*)\| \leq L\|\mathbf{w} - \mathbf{w}^*\| \, \forall (t, \mathbf{w}), (t, \mathbf{w}^*) \in D$.

Assume further that

$$M \geq \frac{K}{L}(\mathrm{e}^{LT} - 1). \tag{A.2}$$

Then, there exists a unique function $\mathbf{w} \in C^1[t_0, t_0 + T]$ *such that* $\mathbf{w}(t_0) = \mathbf{w}_0$ *and* $\mathbf{w} = f(t, \mathbf{w})$, $t \in [t_0, t_0 + T]$.

Moreover,

$$\|(\mathbf{w}(t) - \mathbf{w}_0)\| \leq M, \quad t_0 \leq t \leq t_0 + T. \tag{A.3}$$

Proof. The proof of this theorem can be found in textbooks on theory and numerical methods on ordinary differential equation [119, 102, 125, 116, 22, 58] and textbooks on numerical analysis [46, 108]. □

Frequently, there is no analytical solution available for an ODE or the analytical solution do not give much useful qualitative information. Therefore, numerical solution of an ODE occupies an important role. The numerical solution $\mathbf{v}^n \in \mathbb{R}^d$ of Equation (A.1a) are generally calculated at discrete time points $t_n \in [t_0, t_0 + T]$. In one-step numerical methods like Euler method or Runge-Kutta (RK), the numerical approximation $\mathbf{v}^{n+1}$ at $n+1$-th time step is determined as a function of $t_n, \mathbf{v}^n$ and the time step τ_n and can be expressed in general form as,

$$\mathbf{v}^{n+1} = \mathbf{v}^n + \tau_n \mathbf{F}(t_n, \mathbf{v}^n; \tau_n), \quad \tau = t_{n+1} - t_n, \tag{A.4}$$

where the *incremental function* $\mathbf{F} : [t_0, t_0 + T] \times \mathbb{R}^d \times \mathbb{R}_+ \to \mathbb{R}^d$ is the approximate difference quotient defined by the numerical method.

To analyse the local behaviour of the numerical method, let $\mathbf{w}(t)$ be the reference solution of the following local initial value problem,

$$\mathbf{w}'(t) = \mathbf{f}(t, \mathbf{w}(t)), \quad t \in [t_n, t_n + \tau], \ \mathbf{w}^n = \mathbf{v}^n. \tag{A.5}$$

The local description is concerned with the estimation of *local truncation error* and *consistency* of the numerical method.

Definition (local truncation error)**.** The local truncation error $\boldsymbol{\epsilon}^n$ at $(t_n, \mathbf{v}^n)$ is defined as follows:

$$\boldsymbol{\epsilon}^n = \frac{1}{\tau}[\mathbf{w}^{n+1} - \mathbf{v}^{n+1}]. \tag{A.6}$$

From Equation (A.4) and Equation (A.5), local truncation error can be alternatively described as

$$\boldsymbol{\epsilon}^n = \frac{1}{\tau}[\mathbf{w}^{n+1} - \mathbf{w}^n] - \mathbf{F}(t_n, \mathbf{v}^n; \tau). \tag{A.7}$$

which is basically the difference between the exact and approximate increment per step.

The consistency of the numerical method can be defined based on the definition of local truncation error.

Definition (Consistency)**.** The numerical method is consistent with the Equation (A.1a) if the truncation error, defined by Equation (A.7), satisfies the following,

$$\sup_{n:n\tau\leq T}\|\boldsymbol{\epsilon}^n\| \to 0 \quad \text{as } \tau \to 0. \tag{A.8}$$

Using Equation (A.7) and Equation (A.5), the numerical method is consistent iff

$$\mathbf{F}(t_n, \mathbf{w}^n; 0) \equiv f(t_n, \mathbf{w}(t_n)) \quad t \in [t_0, t_0 + T],\ \mathbf{w} \in \mathbb{R}^d. \tag{A.9}$$

In order to determine the order of accuracy of a numerical scheme, the following definition is necessary.

Definition (order of magnitude)**.** $f(s)$ is of the order of $\phi(s)$ if there exist a constant A such that $|f(s)| \leq A|\phi(s)|$ for all $s \in S$ and it is generally written as $f(s) = O(\phi(s))$. $O()$ is called the *Landau notation.*

A numerical scheme is said to be accurate of order p if the local truncation error $\boldsymbol{\epsilon}^n = O(\tau^p)$ *Global description* of the numerical method is concerned with the behaviour of the *global error*; in particular stability, convergence and the step size τ_n required to ensure convergence. In order to find an approximate solution $\mathbf{v} = \mathbf{v}^n$ (grid function) a grid on the interval $[t_0, t_0 + T]$ is defined by the following set of points

$$t_0 < t_1 < t_2 < \cdots < t_N = t_0 + T, \quad \tau_n = t_{n+1} - t_n,\ n = 0, 1, 2, \cdots, N \tag{A.10}$$

where τ_n is the step size. The global behaviour is analysed for grid functions over a certain grid.

Stability of a numerical scheme which characterizes the robustness of a numerical scheme with respect to small perturbations, is defined as follows:

Definition (Stability)**.** The numerical method Equation (A.4) is said to be stable if there exists a constant $0 < C_0 < \infty$ which does not depend on τ_n such that for an arbitrary grid τ on $[t_0, t_0 + T]$ and for two arbitrary grid functions $\mathbf{v}, \mathbf{v}^* \in \Gamma_g$ on $[t_0, t_0 + T]$, there holds,

$$\|\mathbf{v} - \mathbf{v}^*\| \leq C_0(\|\mathbf{v}_0 - \mathbf{v}_0^*\| + \|\boldsymbol{\epsilon}^n(\mathbf{v}) - \boldsymbol{\epsilon}^n(\mathbf{v}^*)\|), \quad \mathbf{v}, \mathbf{v}^* \in \Gamma_g[t_0, t_0 + T]. \tag{A.11}$$

The condition for stability is given by the following theorem,

Theorem A.2. *If* $\mathbf{F}(t, \mathbf{w}; \tau_n)$ *satisfies a Lipschitz condition with respect to the* $\mathbf{w}$ *variables i.e.,*

$$\|\mathbf{F}(t, \mathbf{v}; \tau_n) - \mathbf{F}(t, \mathbf{v}^*; \tau_n)\| \leq L\|\mathbf{v} - \mathbf{v}^*\| \tag{A.12}$$

then the method Equation (A.4) *is stable.*

Proof. The proof of the above theorem can be found in [46]. □

Convergence is the property of the numerical scheme which gives us total assurance that the numerical solutions obtained is valid approximation of the exact solution.

Definition. Let $\mathbf{v} = \{\mathbf{v}^n\}$ is a grid function calculated by the numerical method Equation (A.4) on the following time discretization,

$$t_0 < t_1 < t_2 < \cdots < t_N = t_0 + T, \quad \tau_n = t_{n+1} - t_n,\ n = 0, 1, 2, \cdots, N. \tag{A.13}$$

Moreover, $\mathbf{w} = \{\mathbf{w}^n\}$ is the grid function induced by the exact solution of Equation (A.1a) on the same grid. The method is said to converge on $[t_0, t_0 + T]$ if

$$\|\mathbf{w} - \mathbf{v}\| \to 0 \quad \text{as } \tau_{\max} \to 0, \tag{A.14}$$

where $\tau_{\max} = \max\limits_{1 \leq n \leq N}(t_n - t_{n-1})$.

The global error can be defined in this connection.

Definition (global error)**.** The global error at the n-th time step is defined as follows:

$$\boldsymbol{\epsilon}_g^n = \mathbf{w}^n - \mathbf{v}^n, \quad \mathbf{w}(t_n) = \mathbf{w}^n. \tag{A.15}$$

The connection between Consistency, stability and convergence for a linear Q is given by the following theorem:

Theorem A.3 (Lax equivalence theorem)**.** *For a well-posed initial value problem and a linear two level consistent discretization scheme, stability is the necessary and sufficient condition for convergence.*

Proof. The proof of this theorem can be found [79]. □

However, it is not plausible to estimate the *global error* directly. The bound on the magnitude of *global error* can be defined in terms of local error using the following theorem.

Theorem A.4. *Consider the general one-step method where, in addition to being a continuous function of its arguments,* $\mathbf{F}$ *is assumed to satisfy a Lipschitz condition with respect to its second argument, that is, there exists a positive constant* L *such that, for* $0 \leq \tau_n \geq T$ *and* $\forall(t, \mathbf{w}_1), (t, \mathbf{w}_2)$ *in the domain*

$$D = \{(t, \mathbf{w}) : t_0 \leq t \leq T, \left\|\mathbf{w} - \mathbf{w}^0\right\| \leq C\}, \tag{A.16}$$

we have

$$\|\mathbf{F}(t, \mathbf{w}_1, \mathbf{f}(t, \mathbf{w}_1(t)); \tau_n) - \mathbf{F}(t, \mathbf{w}_2, \mathbf{f}(t, \mathbf{w}_2(t)); \tau_n)\| \leq L\|\mathbf{w}_1 - \mathbf{w}_2\|. \tag{A.17}$$

Then, assuming that $\left\|\mathbf{w}^n - \mathbf{w}^0\right\| \leq C, \quad n = 1, 2, \cdots, N$, *it follows that,*

$$\left\|\boldsymbol{\epsilon}_g^n\right\| \leq \frac{\boldsymbol{\epsilon}_{\max}}{L}(\mathrm{e}^{t_n - t_0} - 1), \quad n = 1, 2, \cdots, N, \tag{A.18}$$

where $\boldsymbol{\epsilon}_{max} = \sup_{n:n\tau \leq T}\|\boldsymbol{\epsilon}^n\|$.

Proof. The proof can be found in [123]. □

A.1 Solution of Bloch Equations by Operator Splitting

The Bloch equations for MRI in rotating frame read as

$$\frac{\mathrm{d}\mathbf{M}'}{\mathrm{d}t} = \gamma\mathbf{M}' \times \mathbf{B}_{\text{eff}} + \frac{(M_0 - M_z)\hat{e}_z}{T_1} - \frac{M_{x'}\hat{e}_{x'} + M_{y'}\hat{e}_{y'}}{T_2}. \tag{A.19}$$

To solve equation (A.19) by the sequential splitting technique the following two subproblems need to be solved at each time-step $t_n < t \leq t_{n+1}$

$$\frac{\mathrm{d}\mathbf{M}^*}{\mathrm{d}t} = \underbrace{\gamma \mathbf{M}^* \times \mathbf{B}_{\text{eff}}}_{\text{rotation}} = \underbrace{\begin{bmatrix} 0 & \gamma B_z & -\gamma B_y \\ -\gamma B_z & 0 & \gamma B_x \\ \gamma B_y & -\gamma B_x & 0 \end{bmatrix}}_{A_1} \begin{pmatrix} M_x^* \\ M_y^* \\ M_z^* \end{pmatrix} \quad \text{with } \mathbf{M}^*(t_n) = \mathbf{M}'(t_n), \tag{A.20a}$$

$$\frac{\mathrm{d}\mathbf{M}^{**}}{\mathrm{d}t} = \underbrace{\begin{pmatrix} -\frac{M_x^{**}}{T_2} \\ -\frac{M_y^{**}}{T_2} \\ \frac{M_0 - M_z^{**}}{T_1} \end{pmatrix}}_{\text{time relaxation}} = \underbrace{\begin{bmatrix} \frac{-1}{T_2} & 0 & 0 \\ 0 & \frac{-1}{T_2} & 0 \\ 0 & 0 & \frac{-1}{T_1} \end{bmatrix}}_{A_2} \begin{pmatrix} M_x^{**} \\ M_y^{**} \\ M_z^{**} \end{pmatrix} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ \frac{M_0}{T_1} \end{pmatrix}}_{g} \quad \text{with } \mathbf{M}^{**}(t_n) = \mathbf{M}^*(t_{n+1}). \tag{A.20b}$$

The solution of equation (A.20a) is given by

$$\mathbf{M}^*(t_{n+1}) = \mathrm{e}^A \mathbf{M}'(t_n), \quad \tau = t_{n+1} - t_n, \quad \tau A_1 = A, \tag{A.21}$$

and

$$\mathrm{e}^A = I + A + \frac{A^2}{2!} + \frac{A^3}{3!} + \cdots.$$

We can derive a formula for e^A using the following theorems [4]:

(i) Cayley-Hamilton theorem states that every square matrix satisfies its own characteristic polynomial i.e., if $p(\lambda) = det(A - \lambda I)$ is the characteristic polynomial of A then $p(A) = 0$.

(ii) From division algorithm, if $f(\lambda)$ is any polynomial then there exist two unique polynomials $g(\lambda)$ and $r(\lambda)$ such that

$$f(\lambda) = p(\lambda)g(\lambda) + r(\lambda), \tag{A.22}$$

where the degree of $r(\lambda) \leq n - 1$

As e^A is an infinite degree polynomial, it can be uniquely expressed as

$$\mathrm{e}^A = p(A)g(A) + r(A), \tag{A.23}$$

where $p(A)$ is the characteristic polynomial of A and $r(A)$ is a quadratic function of A and $p(A) = 0$ implies that

$$\mathrm{e}^A = r(A) = \sum_{i=0}^{2} \alpha_i A^i, \tag{A.24}$$

where α_i are coefficients of $r(A)$.

Again, if $\lambda_i, i = 1, 2, 3$ are eigenvalues of A then $p(\lambda_i) = 0$. Therefore, from the above equation

$$\mathrm{e}^{\lambda_i} = r(\lambda_i) = \sum_{i=0}^{2} \alpha_i \lambda_{i+1}^i \tag{A.25}$$

From Equation (A.25) the coefficients of the polynomial r can be calculated.

The eigenvalues of A are $0, \pm i\tau\gamma\|B\|$ where $\|B\|^2 = (B_x^2 + B_y^2 + B_z^2)$. Putting λ_is in Equation (A.25) the coefficients α_i of r and thus e^A can be determined. Taking $R = \mathrm{e}^A, n_x = \frac{B_x}{\|B\|}, \quad n_y = \frac{B_y}{\|B\|}, \quad n_z = \frac{B_z}{\|B\|}, \quad \phi = \tau\gamma\|B\|$, elements of R can be written as

$$\begin{aligned}
R_{11} &= n_x^2 + (1 - n_x^2)\cos(\phi), \\
R_{12} &= n_x n_y(1 - \cos(\phi)) + n_z \sin(\phi), \\
R_{13} &= n_x n_z(1 - \cos(\phi)) - n_y \sin(\phi), \\
R_{21} &= n_x n_y(1 - \cos(\phi)) - n_z \sin(\phi), \\
R_{22} &= n_y^2 + (1 - n_y^2)\cos(\phi), \\
R_{23} &= n_y n_z(1 - \cos(\phi)) + n_x \sin(\phi), \\
R_{31} &= n_x n_z(1 - \cos(\phi)) + n_y \sin(\phi), \\
R_{32} &= n_y n_z(1 - \cos(\phi)) - n_x \sin(\phi), \\
R_{33} &= n_z^2 + (1 - n_z^2)\cos(\phi).
\end{aligned} \tag{A.26}$$

Let us assume that $\mathcal{R}_2$ is the solution operator of Equation (A.20b) and $e_2 = \mathrm{e}^{-\tau/T_2}$, $e_1 = \mathrm{e}^{-\tau/T_1}$. Then solving Equation (A.20b), we obtain

$$\mathbf{M}^{**}(t_{n+1}) = \mathcal{R}_2\mathbf{M}^{**}(t_n) = \mathrm{diag}(e_2, e_2, e_1)\mathbf{M}^{**}(t_n) + \begin{pmatrix} 0 \\ 0 \\ M_0(1 - e_1) \end{pmatrix}. \tag{A.27}$$

The numerical solution by sequential splitting method is given by

$$\mathbf{M'}_{n+1} = \mathcal{R}_2 R \mathbf{M'}_n \tag{A.28a}$$

$$= \begin{bmatrix} e_2R_{11} & e_2R_{12} & e_1R_{13} \\ e_2R_{21} & e_2R_{22} & e_1R_{23} \\ e_2R_{31} & e_2R_{32} & e_1R_{33} \end{bmatrix} \mathbf{M'}_n + \begin{pmatrix} 0 \\ 0 \\ M_0(1-e_1) \end{pmatrix}. \tag{A.28b}$$

Similarly, the numerical solution for SWSS is

$$\mathbf{M'}_{n+1} = \frac{1}{2}(\mathcal{R}_2 R + R\mathcal{R}_2)\mathbf{M'}_n \tag{A.29a}$$

$$= \begin{bmatrix} e_2R_{11} & e_2R_{12} & \frac{(e_1+e_2)}{2}R_{13} \\ e_2R_{21} & e_2R_{22} & \frac{(e_1+e_2)}{2}R_{23} \\ \frac{(e_1+e_2)}{2}R_{31} & \frac{(e_1+e_2)}{2}R_{32} & e_1R_{33} \end{bmatrix} \mathbf{M'}_n + \begin{pmatrix} \frac{R_{13}M_0(1-e_1)}{2} \\ \frac{R_{23}M_0(1-e_1)}{2} \\ \frac{(1+R_{33})}{2}M_0(1-e_1) \end{pmatrix}. \tag{A.29b}$$

To determine the solution using the Strang splitting discussed in chapter 3, we first need to modify the relaxation operator for a time step of size $\frac{\tau}{2}$. Let us call it $\hat{\mathcal{R}}_2$ and also denote the corresponding exponential expressions with a $\hat{(\cdot)}$ and we obtain

$$\mathbf{M'}_{n+1} = \hat{\mathcal{R}}_2 R \hat{\mathcal{R}}_2 \mathbf{M'}_n \tag{A.30a}$$

$$= \begin{bmatrix} \hat{e_2}^2R_{11} & \hat{e_2}^2R_{12} & \hat{e_1}\hat{e_2}R_{13} \\ \hat{e_2}^2R_{21} & \hat{e_2}^2R_{22} & \hat{e_1}\hat{e_2}R_{23} \\ \hat{e_1}\hat{e_2}R_{31} & \hat{e_1}\hat{e_2}R_{32} & \hat{e_1}^2R_{33} \end{bmatrix} \mathbf{M'}_n + M_0(1-\hat{e_1}) \begin{pmatrix} R_{13}\hat{e_2} \\ R_{23}\hat{e_2} \\ (1+R_{33}\hat{e_1}) \end{pmatrix}. \tag{A.30b}$$

Consistency, Stability, and Convergence of the Splitting method

The Bloch equations Equation (A.19) can be expressed with the splitted operators as

$$\frac{\mathrm{d}\mathbf{M'}}{\mathrm{d}t} = A_1\mathbf{M'} + A_2\mathbf{M'} + \mathbf{g}. \tag{A.31}$$

Equation (3.22) shows that local truncation error of the operator splitting method for an ODE is at least of the order $O(\tau_n)$. Therefore, the operator splitting method is consistent.

With respect to a suitable operator norm $\|\cdot\|_M$ and a compatible vector norm $\|\cdot\|_V$ the sufficient condition for stability of Equation (A.31) in a finite interval $[t_0, t_0+T]$ is given by $\left\|\mathrm{e}^{\tau A_k}\right\|_M \leq \mathrm{e}^{\tau\omega_k}, k = 1, 2$ [16, 9].

With respect to Euclidean matrix and vector norm [108],

$$\left\|\mathrm{e}^{\tau A_1}\right\|_2 \leq \mathrm{e}^{\tau\|A_1\|_2} = \mathrm{e}^{\tau\sigma_1(A_1)} = \mathrm{e}^{\gamma\|B\|\tau} \tag{A.32}$$

as

$$\|A_1\|_2 = \sigma_1(A_1) = \sqrt{\rho(A_1^H A_1)} = \gamma\|B\|\,, \tag{A.33}$$

where $\sigma_1(\cdot), \rho(\cdot)$ represents the maximum singular value and eigenvalue of the matrix. Similarly,

$$\left\|\mathrm{e}^{\tau A_2}\right\|_2 \leq \mathrm{e}^{\tau\|A_2\|_2} = \mathrm{e}^{\frac{\tau}{T_2}} \tag{A.34}$$

as

$$\|A_2\|_2 = \sigma_1(A_2) = \sqrt{\rho(A_2^H A_2)} = \frac{1}{T_2}. \tag{A.35}$$

Therefore, according to Theorem A.3, the operator splitting method used for the solution of Bloch equation is convergent.

Appendix B

Existence and Uniqueness of Bloch Equation for Flowing Spins

In this chapter, well-posedness of Bloch Equations for flowing spins will be discussed.

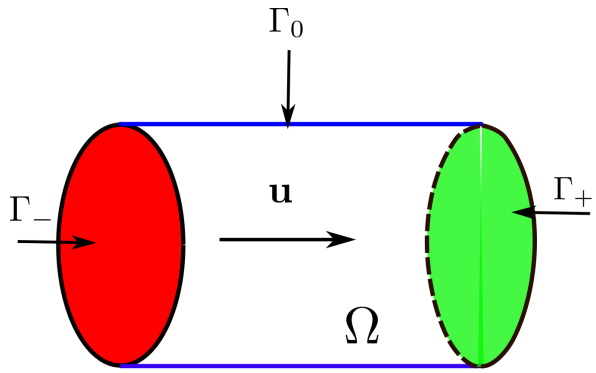

Figure B.1: Schematic diagram of pipe flow for illustrating domain (Ω) and domain boundaries. Γ_- marked with red color represents the inflow boundary. Γ_+ marked with green color represents the outflow boundary. The blue line represents impermeable walls of the pipe which is no-flow boundaries, denoted by the symbol Γ_0. $\mathbf{u}$ is the velocity field with flow direction (marked by the arrow below $\mathbf{u}$) from the inflow boundary towards the outflow boundary.

To this purpose, let $\Omega \in \mathbb{R}^d, d = 3$ be the flow domain with a piecewise smooth Lipschitz boundary $\Gamma = \Gamma_- \bigcup \Gamma_+ \bigcup \Gamma_0$. The Bloch equation for flowing spins along with

suitable boundary and initial conditions are given as follows:

$$\frac{\partial \mathbf{M}'}{\partial t} + (\mathbf{u}\cdot\boldsymbol{\nabla})\mathbf{M}' = \gamma \mathbf{M}' \times \mathbf{B}_{\text{eff}} + \frac{(M_0 - M_z)\hat{e}_z}{T_1} - \frac{M_{x'}\hat{e}_{x'} + M_{y'}\hat{e}_{y'}}{T_2} \quad (t,\mathbf{r}) \in (0,T]\times\Omega, \tag{B.1a}$$

$$\mathbf{M}' = \mathbf{M}_\Gamma, \quad (t,\mathbf{r}) \in [0,T]\times\Gamma_-, \tag{B.1b}$$

$$\mathbf{M}' = \mathbf{M}^0, \quad (t,\mathbf{r}) \in \{0\}\times\Omega, \tag{B.1c}$$

where $\Gamma_- = \{\mathbf{w}\in\Gamma | \mathbf{u}\cdot\mathbf{n} < 0\}$, $\Gamma_+ = \{\mathbf{w}\in\Gamma|\mathbf{u}\cdot\mathbf{n} > 0\}$ and $\Gamma_0 = \{\mathbf{w}\in\Gamma|\mathbf{u}\cdot\mathbf{n} = 0\}$ represent the inflow, outflow boundary and solid wall respectively; $\mathbf{u}: \Omega\times[0,T] \mapsto \mathbb{R}^d$ be a given incompressible flow field, i.e., $\nabla\cdot\mathbf{u} = 0$; $\mathbf{n}$ is the outward unit normal. A typical flow domain is illustrated with an example of pipe flow in Figure B.1.

Theorem B.1 (Well-posedness)**.** *There exists a unique solution to Equation* (B.1) *for sufficiently smooth* $\mathbf{u}$ *and* $\mathbf{B}_{\text{eff}}$*.*

Proof. Equation (B.1a) can be written in the following form using the applied magnetic field $\mathbf{B}_{\text{eff}} = \mathbf{B} = \begin{pmatrix} B_x & B_y & B_z \end{pmatrix}^T$, the relaxation time diagonal matrix $D = \text{diag}(\frac{1}{T_2}, \frac{1}{T_2}, \frac{1}{T_1})$ and the constant additional source term $\mathbf{f} = \begin{pmatrix} 0 & 0 & \frac{M_0}{T_1} \end{pmatrix}$:

$$\frac{\partial \mathbf{M}'}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{M}' - \gamma\mathbf{M}'\times\mathbf{B} + D\mathbf{M}' = \mathbf{f}, \quad (t,\mathbf{r})\in[0,T]\times\Omega. \tag{B.2}$$

Let us assume for simplicity that the Dirichlet boundary condition be $\mathbf{M}_\Gamma = \mathbf{0}$.

In order to derive a generalized solution of Equation (B.1), consider the space $L^2(\Omega)$ of square-integrable functions and the following space

$$\mathbf{X} = \{\mathbf{N}\in[L^2(\Omega)^d], \quad (\mathbf{u}\cdot\nabla)\mathbf{N}\in[L^2(\Omega)^d], \quad \mathbf{N}|_{\Gamma_-} = \mathbf{0}\}. \tag{B.3}$$

Multiplying Equation (B.2) by an arbitrary test vector function $\mathbf{N}\in\mathbf{X}$ and integrating over domain Ω, we obtain

$$\int_\Omega \partial_t\mathbf{M}'\cdot\mathbf{N}d\mathbf{r} + \int_\Omega(\mathbf{u}\cdot\nabla)\mathbf{M}'\cdot\mathbf{N}d\mathbf{r} + \gamma\int_\Omega(\mathbf{B}\times\mathbf{M}')\cdot\mathbf{N}d\mathbf{r} + \int_\Omega D\mathbf{M}'\cdot\mathbf{N}d\mathbf{r} = \int_\Omega \mathbf{f}\cdot\mathbf{N}d\mathbf{r} \tag{B.4}$$

as $\mathbf{M}'\times\mathbf{B} = -\mathbf{B}\times\mathbf{M}'$.

Let us define the bilinear and linear forms as follows:

$$a(t;\mathbf{M}',\mathbf{N}) := \int_\Omega (\mathbf{u}\cdot\nabla)\mathbf{M}'\cdot\mathbf{N}d\mathbf{r} + \gamma\int_\Omega(\mathbf{B}\times\mathbf{M})\cdot\mathbf{N}d\mathbf{r} + \int_\Omega D\mathbf{M}'\cdot\mathbf{N}d\mathbf{r}, \quad \text{(B.5a)}$$

$$l(\mathbf{N}) := \int_\Omega \mathbf{f}\cdot\mathbf{N}d\mathbf{r}. \quad \text{(B.5b)}$$

Let us use also the inner product $(\mathbf{u},\mathbf{v})_{L^2(\Omega)} := \int_\Omega \mathbf{u}\cdot\mathbf{v}dr$ and norm $\|\mathbf{v}\|_{L^2(\Omega)} := \sqrt{(v,v)}$ on $L^2(\Omega)$. With these definitions, we obtain the linear evolution problem of first order: find $\mathbf{M}' : [0,T] \mapsto \mathbf{X}$ such that

$$(\partial_t\mathbf{M}',\mathbf{N})_{L^2(\Omega)} + a(t;\mathbf{M}',\mathbf{N}) = l(\mathbf{N}), \quad \forall\,\mathbf{N}\in\mathbf{X}, \quad \text{(B.6)}$$

with initial condition

$$\mathbf{M}'|_{t=0} = \mathbf{M}^0. \quad \text{(B.7)}$$

We want to apply the main existence theorem by J.L. Lions, given by Theorem 6.6 in [37]. To this end, we define the graph norm

$$\left\|\mathbf{M}'\right\|_{\mathbf{X}} = \left\|\mathbf{M}'\right\|_{L^2(\Omega)} + \left\|(\mathbf{u}\cdot\nabla)\mathbf{M}'\right\|_{L^2(\Omega)}. \quad \text{(B.8)}$$

In order to apply the theorem, the following conditions must be satisfied:

(**P1**) The time-dependent bilinear form $t \mapsto a(t;\mathbf{M},\mathbf{N})$ is measurable $\forall\ \mathbf{M},\mathbf{N}\in\mathbf{X}$ provided the vector field $\mathbf{u}$ and $\mathbf{B}$ are sufficiently smooth.

(**P2**) The bilinear form $a(t,\cdot,\cdot)$ is bounded for $t\in[0,T],\ \forall\ \mathbf{M},\mathbf{N}\in\mathbf{X}$.

(**P3**) Finally, the bilinear form fulfills the coercivity condition, given by,

$$a(t,\mathbf{N},\mathbf{N}) = (D^{1/2}\mathbf{N}, D^{1/2}\mathbf{N})_{L^2(\Omega)} = \left\|D^{1/2}\mathbf{N}\right\|^2_{L^2(\Omega)} \geq \sigma\|\mathbf{N}\|^2_{L^2(\Omega)} \quad \text{(B.9)}$$

with

$$D^{1/2} := \mathrm{diag}(\frac{1}{\sqrt{T_2}},\frac{1}{\sqrt{T_2}},\frac{1}{\sqrt{T_1}}), \quad \sigma := \min(\frac{1}{T_1},\frac{1}{T_2}) = \frac{1}{T_1}\ (\text{as } T_1\geq T_2), \quad \text{(B.10)}$$

as from the skew-symmetry property it follows that

$$((\mathbf{u}\cdot\nabla\mathbf{N}),\mathbf{N})^2_{L^2(\Omega)} + \gamma((\mathbf{B}\times\mathbf{N}),\mathbf{N})^2_{L^2(\Omega)} = 0 \quad \text{(B.11)}$$

when $\nabla \cdot \mathbf{u} = 0$.

Now we can apply the Lions theorem giving existence and uniqueness of a generalized solution $\mathbf{M} : [0, T] \mapsto \mathbf{X}$ of Equation (B.1). □

Moreover, we obtain the a-priori estimate for the kinetic energy of the magnetic field,

$$\frac{1}{2}\left\|\mathbf{M}'(t)\right\|^2_{L^2(\Omega)} \leq \frac{1}{2}\left\|\mathbf{M}'(0)\right\|^2_{L^2(\Omega)} \mathrm{e}^{-\sigma t} + \int_0^t \|\mathbf{f}(s)\|^2_{L^2(\Omega)} \mathrm{e}^{\sigma(s-t)} ds. \tag{B.12}$$

In order to obtain the energy estimate as described in Equation (B.12), we set $\mathbf{N} = \mathbf{M}'$ in Equation (B.4) and obtain,

$$(\partial_t \mathbf{M}', \mathbf{M}') + a(t; \mathbf{M}', \mathbf{M}') = l(\mathbf{M}') \tag{B.13a}$$

$$\Rightarrow \frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\left\|\mathbf{M}'\right\|^2_{L^2(\Omega)} + a(t; \mathbf{M}', \mathbf{M}') = (\mathbf{f}, \mathbf{M}')_{L^2(\Omega)}. \tag{B.13b}$$

Equation (B.9), Using Cauchy-Schwarz and Young's inequalities respectively we obtain,

$$l(\mathbf{M}') \leq \|\mathbf{f}\|_{L^2(\Omega)} \left\|\mathbf{M}'\right\|_{L^2(\Omega)} \quad \text{(Cauchy-Schwarz's inequality)} \tag{B.14a}$$

$$\leq \frac{1}{2\sigma}\|\mathbf{f}\|^2_{L^2(\Omega)} + \frac{\sigma}{2}\left\|\mathbf{M}'\right\|^2_{L^2(\Omega)} \quad \text{(Young's inequality)}. \tag{B.14b}$$

Using (**P3**) and Equation (B.14b), we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{1}{2}\left\|\mathbf{M}'\right\|^2_{L^2(\Omega)}\right) + \frac{\sigma}{2}\left\|\mathbf{M}'\right\|^2_{L^2(\Omega)} \leq \frac{1}{2\sigma}\|\mathbf{f}\|^2_{L^2(\Omega)}\,. \tag{B.15}$$

Now, applying the Gronwall Lemma, as given by Lemma 6.9 in [37] implies

$$\frac{1}{2}\left\|\mathbf{M}'\right\|^2_{L^2(\Omega)} \leq \frac{1}{2}\|\mathbf{M}(0)\|^2_{L^2(\Omega)} \mathrm{e}^{-\sigma t} + \frac{1}{2\sigma}\int_0^t \|\mathbf{f}(s)\|^2_{L^2(\Omega)} \mathrm{e}^{\sigma(s-t)} ds. \tag{B.16}$$

We have constant source $\mathbf{f}$ giving

$$\|\mathbf{f}(s)\|^2_{L^2(\Omega)} = \|\mathbf{f}\|^2_{L^2(\Omega)} = \int_\Omega \left(\frac{M_0}{T_1}\right)^2 dr = \left(\frac{M_0}{T_1}\right)^2 |\Omega| \quad \text{as } |\Omega| = \int_\Omega dr. \tag{B.17}$$

Moreover,

$$\int_0^t \|\mathbf{f}(s)\|^2_{L^2(\Omega)} \mathrm{e}^{\sigma(s-t)} ds = \left(\frac{M_0}{T_1}\right)^2 |\Omega| \, \frac{1 - \mathrm{e}^{-\sigma t}}{\sigma} \tag{B.18}$$

and we finally obtain

$$\frac{1}{2}\left\|\mathbf{M}'\right\|^2_{L^2(\Omega)} \leq \frac{1}{2}\left\|\mathbf{M}'(0)\right\|^2_{L^2(\Omega)} \mathrm{e}^{-\sigma t} + \frac{1}{2\sigma^2}(\frac{M_0}{T_1})^2(1-\mathrm{e}^{-\sigma t})|\Omega|\,, \tag{B.19a}$$

$$\leq \frac{1}{2}\left\|\mathbf{M}'(0)\right\|^2_{L^2(\Omega)} \mathrm{e}^{-\frac{t}{T_1}} + \frac{M_0^2}{2}(1-\mathrm{e}^{-\frac{t}{T_1}})|\Omega|\,, \tag{B.19b}$$

because $\sigma = \frac{1}{T_1}$.

Remark. The result of Theorem B.1 and a-priori estimate Equation (B.19) remain valid for the special case $\mathbf{u} = \mathbf{0}$ i.e. Bloch equations for spatially stationary objects.

Appendix C

Discontinuous Galerkin Method for Advection Equation

In this chapter special discretization of advection equation using discontinuous Galerkin method (dGFEM) method is discussed. In the last section, it is shown that finite volume method (FVM) is a special case of dGFEM.

To this end, let us consider a general advection or transport equation in a conservative form in a computational domain $\Omega \in \mathbb{R}^d$ with a piecewise smooth Lipschitz boundary $\Gamma = \Gamma_- \bigcup \Gamma_+ \bigcup \Gamma_0$:

$$\frac{\partial \mathbf{w}}{\partial t} + \nabla \cdot f(\mathbf{w}) = \mathbf{0}, \quad f(\mathbf{w}) = \mathbf{u} \otimes \mathbf{w}, \quad (t, \mathbf{r}) \in (0, T] \times \Omega, \tag{C.1a}$$

$$\mathbf{w}(\mathbf{r}, 0) = \mathbf{w}_0(\mathbf{r}), \quad \{0\} \times \Omega, \tag{C.1b}$$

$$\mathbf{w}(\mathbf{r}, t)|_{\Gamma_- \times [0,T]} = \mathbf{w}_b, \tag{C.1c}$$

where $\Gamma_- = \{\mathbf{w} \in \Gamma | \mathbf{u} \cdot \mathbf{n} < 0\}$, $\Gamma_+ = \{\mathbf{w} \in \Gamma | \mathbf{u} \cdot \mathbf{n} > 0\}$ and $\Gamma_0 = \{\mathbf{w} \in \Gamma | \mathbf{u} \cdot \mathbf{n} = 0\}$ represent the inflow, outflow boundary and solid wall respectively; $\mathbf{u} : \Omega \times [0, T] \mapsto \mathbb{R}^d$ is a given transport velocity and $\mathbf{n}$ is the outward unit normal.

Multiplying Equation (C.1) by an arbitrary test vector function $\mathbf{N} \in \mathbf{X}$ (as defined by Equation (B.3)) and integrating by parts we obtain the weak formulation of Equation (C.1), given by

$$\int_\Omega \mathbf{N} \cdot \frac{\partial \mathbf{w}}{\partial t} \, d\mathbf{r} - \int_\Omega \sum_{s=1}^{d} \mathbf{f}_s(\mathbf{w}) \cdot \frac{\partial \mathbf{N}}{\partial r_s} \, d\mathbf{r} + \int_\Gamma \sum_{s=1}^{d} \mathbf{f}_s(\mathbf{w}) n_s \cdot \mathbf{N} d\Gamma = 0, \tag{C.2}$$

where $\mathbf{n} = \begin{pmatrix} n_1, & n_2, & \cdots, n_d \end{pmatrix}$ is the outer normal in the boundary.

In the following we deal with the discretization of Equation (C.2) by the dGFEM [31]. Consider a non-overlapping decomposition $\mathcal{T}_h := \{\Omega\}_{i=1}^{I}$ into convex simplical subdomains $\Omega_i, i = 1, 2, \cdots I$ as depicted in Figure C.1. We define the discontinuous finite element space

$$[\mathbb{P}_k(\mathcal{T}_h)]^d := \{\mathbf{N}_h \in [L^2(\Omega)]^d;\ \mathbf{N}_h|_{\Omega_i} \in [\mathbb{P}_k(\Omega_i)]^d \quad \forall\, \Omega_i,\ i = 1, 2, \cdots, I\}, \tag{C.3}$$

where $\mathbb{P}_k$ denotes the set of polynomials of degree $k \in \mathbb{N}$. Moreover, let $\mathbf{X}_h = [\mathbb{P}_k(\mathcal{T}_h)]^d \bigcap \mathbf{X}$ (for simplicity only homogeneous Dirichlet boundary conditions).

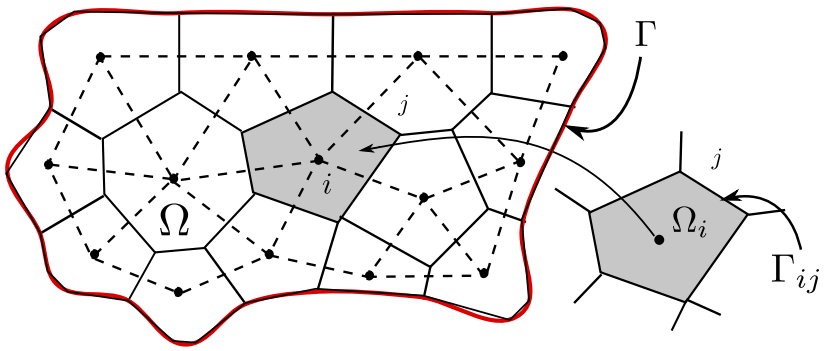

Figure C.1: Schematic representation of a 2-D grid. Ω represent the computational domain. The boundary of the domain Γ is marked with red line. i-th cell is magnified and Ω_i and E represent the area of the i-th cell and the edge between i-th and the j-th cell respectively.

For adjacent subdomains Ω_i, Ω_j with interface $\Gamma_{ij} = \overline{\Omega}_i \bigcap \overline{\Omega}_j$ and unit normal vector $\mathbf{n}_{ij}$ (directed from Ω_i to Ω_j), we define the average and jump of $\mathbf{N}_h \in \mathbf{X}_h$ across Γ_{ij} by

$$\langle \mathbf{N}_h \rangle_{\Gamma_{ij}}(r) := \frac{1}{2}(\mathbf{N}_h|_{\Omega_i}(r) + \mathbf{N}_h|_{\Omega_j}(r)), \tag{C.4a}$$

$$[\mathbf{N}_h]_{\Gamma_{ij}}(r) := \mathbf{N}_h|_{\Omega_i}(r) - \mathbf{N}_h|_{\Omega_j}(r). \tag{C.4b}$$

To derive the discrete formulation we assume that there exists an exact solution $\mathbf{w} \in C^1([0,T]; \mathbf{X}_h)$ and Equation (C.2) is applied for all the elements $\Omega_i \in \mathcal{T}_h$ with test function $\mathbf{N} \in \mathbf{X}_h$ and then summed over all the elements $\Omega_i \in \mathcal{T}_h$ to obtain the following:

$$\sum_{\Omega_i \in \mathcal{T}_h} \int_{\Omega_i} \frac{\partial \mathbf{w}}{\partial t} \cdot \mathbf{N} d\mathbf{r} - \sum_{\Omega_i \in \mathcal{T}_h} \int_{\Omega_i} \sum_{s=1}^{d} \mathbf{f}_s(\mathbf{w}) \cdot \frac{\partial \mathbf{N}}{\partial r_s} d\mathbf{r} + \sum_{\Omega_i \in \mathcal{T}_h} \int_{\partial \Omega_i} \sum_{s=1}^{d} \mathbf{f}_s(\mathbf{w}) n_{\Gamma,s} \cdot \mathbf{N} d\Gamma = 0. \tag{C.5}$$

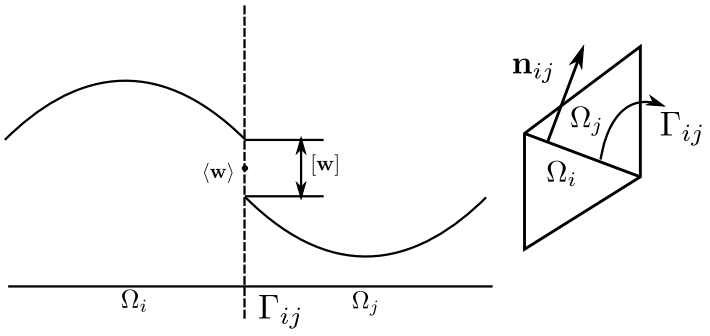

Figure C.2: (Left) One dimensional example of average and jump operators. (Right) The interface between the i and j-th cell where $j > i$ is depicted with the used notation. The orientation of the outward normal is from lower to higher numbered cell.

Taking into consideration the boundary conditions and using dGFEM formulation [31, 30] Equation (C.5) reduces to

$$\sum_{\Omega_i \in \mathcal{T}_h} \int_{\Omega_i} \frac{\partial \mathbf{w}}{\partial t} \cdot \mathbf{N} d\mathbf{r} - \sum_{\Omega_i \in \mathcal{T}_h} \int_{\Omega_i} \sum_{s=1}^{d} \mathbf{f}_s(\mathbf{w}) \cdot \frac{\partial \mathbf{N}}{\partial r_s} \, d\mathbf{r} \tag{C.6}$$
$$+ \sum_{\Omega_i \in \mathcal{T}_h^I} \int_{\partial\Omega_i} \sum_{s=1}^{d} \mathbf{f}_s(\mathbf{w}) n_{\Gamma,s} \cdot [\mathbf{N}] d\Gamma + \sum_{\Omega_i \in \mathcal{T}_h^B} \int_{\partial\Omega_i} \sum_{s=1}^{d} \mathbf{f}_s(\mathbf{w}) n_{\Gamma,s} \cdot \mathbf{N} d\Gamma = 0,$$

where $\Omega_i \in \mathcal{T}_h^I$ and $\Omega_i \in \mathcal{T}_h^B$ denote the inner and the boundary cells respectively.

The crucial point of dGFEM is the evaluation of the integrals over $\partial\Omega_i$, approximated with the aid of numerical flux $\mathbf{F} : \mathbf{X}_h \times \mathbf{X}_h \times \mathbb{R}^d \mapsto \mathbb{R}^d$ by

$$\int_{\Gamma_{ij}} \sum_{s=1}^{d} \mathbf{f}_s(\mathbf{w}) n_{\Gamma,s} \cdot \mathbf{N} d\Gamma \approx \int_{\Gamma_{ij}} \mathbf{F}(\mathbf{w}_i, \mathbf{w}_j, \mathbf{n}_{ij}) \cdot \mathbf{N} d\Gamma, \tag{C.7}$$

where i and j denotes the so-called left and right states as depicted in Figure C.2.

The numerical flux must satisfy some basic conditions:

- continuity: $\mathbf{F}(\mathbf{w}_i, \mathbf{w}_j, \mathbf{n})$ is locally Lipschitz-continuous with respect to variables $\mathbf{w}_i$ and $\mathbf{w}_j$.
- consistency: $\mathbf{F}(\mathbf{w}, \mathbf{w}, \mathbf{n}) = \sum_{s=1}^{d} \mathbf{f}_s(\mathbf{w}) n_s, \quad \mathbf{n} = \begin{pmatrix} n_1 & n_2 & \cdots n_d \end{pmatrix}$.
- conservativity: $\mathbf{F}(\mathbf{w}_i, \mathbf{w}_j, \mathbf{n}) = -\mathbf{F}(\mathbf{w}_j, \mathbf{w}_i, -\mathbf{n}), \quad \mathbf{w}_i, \mathbf{w}_j \in \mathbf{X}_h$.

Approximating the face integrals in Equation (C.6) by Equation (C.7) and interchanging the derivative and the integral in first term, we get the discontinuous-Galerkin

space semi-discretization of Equation (C.1a) as follows:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{w}(t),\mathbf{N}\right)+b_h(\mathbf{w}(t),\mathbf{N})=0,\quad \forall\,\mathbf{N}_h\in\mathbf{X}_h,\ t\in(0,T), \tag{C.8a}$$

where

$$\left(\mathbf{w}(t),\mathbf{N}\right)=\int_\Omega \mathbf{w}(t)\cdot\mathbf{N}d\mathbf{r}, \tag{C.8b}$$

$$\begin{aligned} b_h(\mathbf{w}(t),\mathbf{N}) = &\sum_{\Gamma\in T_h^I}\int_\Gamma \mathbf{F}(\mathbf{w}_i,\mathbf{w}_j,\mathbf{n}_\Gamma)\cdot[\mathbf{N}]d\Gamma \\ &+\sum_{\Gamma\in T_h^B}\int_\Gamma \mathbf{F}(\mathbf{w}_i,\mathbf{w}_j,\mathbf{n}_\Gamma)\cdot\mathbf{N}d\Gamma-\int_{\Omega_i}\sum_{s=1}^{d}\mathbf{f}_s(\mathbf{w})\cdot\frac{\partial\mathbf{N}}{\partial r_s}\,d\mathbf{r}, \end{aligned} \tag{C.8c}$$

where Equation (C.8) make sense for $\mathbf{w},\mathbf{N}\in\mathbf{X}_h$.

The approximation of the exact solution $\mathbf{w}(t)$ will be sought in the finite-dimensional spaces $[\mathbb{P}_k(\mathcal{T}_h)]^d=\mathbf{S}_{hk}\subset\mathbf{X}_h$ for each $t\in(0,T]$.

We say that a function $\mathbf{w}_h:\Omega\times(0,T]\mapsto\mathbb{R}^d$ is the semi-discrete solution of the transport equation Equation (C.1a), if the following conditions are satisfied:

$$\mathbf{w}_h\in C^1([0,T];\mathbf{S}_{hk}), \tag{C.9a}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{w}_h(t),\mathbf{N}_h\right)+b_h(\mathbf{w}_h(t),\mathbf{N}_h)=0,\quad \forall\,\mathbf{N}_h\in\mathbf{S}_{hk},\ t\in(0,T), \tag{C.9b}$$

$$\mathbf{w}_h(0)=\Pi_h\mathbf{w}^0, \tag{C.9c}$$

where $\Pi_h\mathbf{w}^0$ is the $\mathbf{S}_{hk}$-approximation of the function $\mathbf{w}^0$ from the initial condition and usually defined as the L^2-projection of $\mathbf{w}^0$ on the space $\mathbf{S}_{hk}$.

Remark. For $k=0$ the basis functions of $\mathbf{S}_{h0}$ are chosen to be the characteristic functions χ_i of $\Omega_i\in T_h$ i.e. $\chi_k=1$ on Ω_i and $\chi_i=0$ elsewhere, Equation (C.9) reduces to standard finite volume method (FVM) (i.e. the approximate solution is piecewise constant on T_h). Putting $\mathbf{N}_h=\chi_i,\Omega_i\in\mathcal{T}_h$ we obtain the following semi-discretized equation:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(|\Omega_i|\,\mathbf{w}_i(t)\right)+\sum_{j\in J_i}\mathbf{F}(\mathbf{w}_i,\mathbf{w}_j,\mathbf{n}_{ij})=0, \tag{C.10}$$

where

$$\mathbf{w}_i = \frac{1}{|\Omega_i|} \int_{\Omega_i} \mathbf{w}_h d\Omega, \quad \Omega_i \in T_h \tag{C.11}$$

and J_i is set of all elements with a common face with Ω_i. For implementation of boundary conditions, the set J_i is assumed to contain some fictitious elements having a common face $\partial\Omega_i \bigcap \Omega$, known as ghost cell. In that case, the numerical flux is determined assigning compatible boundary conditions in the ghost cells. For higher-resolution FVM numerical flux $\mathbf{F}$ is approximated with values of $\mathbf{w}$ from several neighbouring cells depending on the order of accuracy as discussed in Chapter 4.

Appendix D

Briefly on the Numerical Analysis of Partial Differential Equation

In this chapter, a set of relevant mathematical concepts and theorems which is required for analyzing the numerical solution of the partial differential equation (PDE) in the present thesis will be discussed.

For a detailed discussion the reader is referred to the books by Thomas [126, 127], LeVque [81], Toro [128].

Consider a PDE representing an initial-boundary value problem:

$$\mathcal{L}(\mathbf{w}(\mathbf{r},t)) = \mathbf{f}(\mathbf{r},t), \quad \mathbf{r} \in \Omega, \quad t \in (0,T], \tag{D.1a}$$

$$\mathbf{w}(\mathbf{r},0) = \mathbf{w}_0(\mathbf{r}), \quad \mathbf{r} \in \overline{\Omega}, \tag{D.1b}$$

$$\mathbf{w}(\mathbf{r},t)|_{\partial\Omega} = \mathbf{w}_b(r,t), \quad t \in (0,T], \tag{D.1c}$$

where $\mathbf{f}$ and $\mathbf{w}_0$ are given and $\mathcal{L}$ is a partial differential operator of first-order.

Numerical discretization of Equations (D.1a)–(D.1c) using a suitable spatial discretization (e.g. FVM) and two level time discretization gives us the following general difference equation:

$$\mathbf{v}^{n+1} = Q^n(\mathbf{v}^n) + \tau\mathbf{F}^n, \quad \tau = t_{n+1} - t_n, \tag{D.2a}$$

$$\mathbf{v}^0 = \mathbf{w}^0, \tag{D.2b}$$

where

$$\mathbf{w}^0 = \begin{bmatrix} \mathbf{w}_0(\mathbf{r}_1) & \mathbf{w}_0(\mathbf{r}_2) & \cdots & \mathbf{w}_0(\mathbf{r}_{ng}) \end{bmatrix}^T. \tag{D.2c}$$

Here superscript corresponds to the time step and $\mathbf{v}^n$ represent the numerical solution at grid with ng grid points and the matrix Q may depend on $\tau,\ \Delta\mathbf{r}$.

Let us assume

$$\mathbf{w}^n = \begin{bmatrix} \mathbf{w}^n(\mathbf{r}_1) & \mathbf{w}^n(\mathbf{r}_2) & \cdots & \mathbf{w}^n(\mathbf{r}_{ng}) \end{bmatrix}^T \tag{D.3}$$

be the exact solution of Equations (D.1a)–(D.1c).

Definition (Consistency)**.** The numerical scheme, given by Equations (D.2a)–(D.2b), is consistent with Equations (D.1a)–(D.1c) in a norm $\|\cdot\|$ if the solution of the partial differential equation, $\mathbf{w}$ satisfies

$$\mathbf{w}^{n+1} = Q^n(\mathbf{w}^n) + \tau\mathbf{F}^n + \tau\boldsymbol{\epsilon}^n \tag{D.4}$$

such that

$$\sup_{n:n\tau\leq T} \|\boldsymbol{\epsilon}^n\| \to 0 \tag{D.5}$$

as $\|\Delta\mathbf{r}\|, \Delta t \to 0$. The quantity $\boldsymbol{\epsilon}^n$ is called the *local truncation error* of the numerical scheme.

Definition. The numerical scheme is said to be accurate of order (p, q) to the given partial differential equation Equations (D.1a)–(D.1c) if

$$\|\boldsymbol{\epsilon}^n\| = O\Big(\|\Delta\mathbf{r}\|^p\Big) + O(\tau^q). \tag{D.6}$$

Definition (Stability)**.** The numerical method is said to be stable if for some constant $C_0 < \infty$

$$\sup_{n:n\tau\leq T} \|Q^n\| \leq C_0, \tag{D.7}$$

where C_0 may depend on T.

Stability property is solely related with the numerical scheme and it does not have any relation with the differential equation.

Definition (Convergence)**.** The numerical method is said to be convergent if

$$\sup_{n:n\tau\leq T} \|\mathbf{w}^n - \mathbf{v}^n\| \to 0 \quad \text{as } \tau,\ \|\Delta\mathbf{r}\| \to 0. \tag{D.8}$$

Convergence is the property of the numerical scheme which gives us total assurance that the numerical solutions obtained is valid approximation of the exact solution.

The condition for convergence of two level linear methods like simple upwind or Lax-Wendroff method are given by Theorem A.3.

D.1 Definitions and Theorems Related to the Solution of Advection Equation

The local truncation error of a method shows how well the true solution of a differential equation satisfies the difference equation. However, trying to find out a PDE for which the numerical approximation is an exact solution reveals significant qualitative features of the numerical scheme as discussed below.

Definition (Modified equation)**.** The solution of a numerical scheme approximately satisfies a PDE which is generally different from the original PDE and it is known as the modified equation. Modified equation can be obtained by Taylor series expansion of the difference equation. By truncating the infinite series at some point, we obtain a PDE which gives a good indication of the behaviour of the numerical scheme [61, 85].

For example a simple upwind method adds an artificial diffusive term in the equation which explains the diffusivity properties in the numerical solution of simple upwind method. On the other hand, the Lax-Wendroff method adds a third-order dispersive term which leads to dispersive behaviour rather than diffusion [81].

Regarding stability of the numerical scheme, one of necessary conditions is given by the Courant-Friedrich-Lewy (CFL) criteria as defined below.

Definition (Courant-Friedrich-Lewy (CFL) condition)**.** A partial differential equation and an associated numerical scheme is said to satisfy the CFL condition if the true domain of dependence is contained in the numerical domain of dependence [28, 27].

CFL condition is illustrated in Figure D.1. In the left part of of Figure D.1, the true domain of dependence for $\mathbf{w}_i = \mathbf{w}(x_i, T)$ depicted by the non-shaded cone lies outside the numerical domain of dependence denoted by the shaded region. Therefore, the scheme is unstable. On the other hand, in the right part of Figure D.1 a finer time discretization is used where the physical domain of dependence for $\mathbf{w}(x_i, T)$ is contained in the numerical domain of dependence. Therefore the scheme is stable.

The following definitions and theorems are concerned with the construction of higher order high-resolution numerical methods. The first order linear methods are

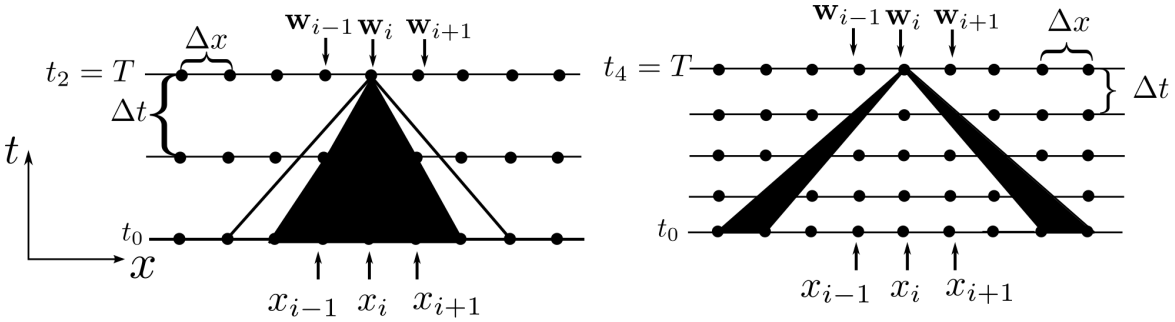

Figure D.1: Schematic diagram explaining the CFL criteria for a three-point scheme. (Left) An unstable three point scheme. The shaded region shows the numerical domain of dependence which does not contain the true domain of dependence (Right) A stable three point scheme. True domain of dependence which is marked by white cone in the centre contained in the numerical domain of dependence.The extra numerical domain is shown by the surrounding shaded region.

highly diffusive and results in much lower order solution. Whereas second-order linear methods like Lax-Wendroff method fail near discontinuities and oscillations appear due to the dispersive nature of these methods. Higher-order high resolution method combines the non-oscillatory nature of the upwind method with higher order accuracy.

In order to eliminate the numerical oscillation one natural requirement for a numerical scheme is that it must be monotonicity preserving. Everything hereafter will be defined for a scalar equations for simplicity.

Definition (Monotonicity preserving method)**.** A difference scheme of the form

$$v_i^{n+1} = Q^n(v_{i+p}^n, \cdots, v_{i+q}^n) \tag{D.9}$$

is said to be monotonicity preserving if

$$v_i^n \geq v_{i+1}^n, \quad \forall\, i \tag{D.10}$$

implies that

$$v_i^{n+1} \geq v_{i+1}^{n+1}, \quad \forall\, i \tag{D.11}$$

To construct a monotonicity preserving method numerical schemes must satisfy the total variation diminishing (TVD) property as defined below.

Definition. The numerical method

$$\mathbf{w}_i^{n+1} = Q^n(\mathbf{w}_{i+p}^n, \cdots, \mathbf{w}_{i+q}^n) \tag{D.12}$$

is called TVD if it satisfies the following criteria

$$TV(\mathbf{w}_i^{n+1}) \leq TV(\mathbf{w}_i^n), \quad \forall \mathbf{w}_i^n, \tag{D.13}$$

where

$$TV(\mathbf{w}_i^n) = \sum_i \left\| \mathbf{w}_{i+1}^n - \mathbf{w}_i^n \right\|. \tag{D.14}$$

The following two theorems give us the criteria to construct the higher order non-oscillatory schemes,

Theorem D.1. *Any TVD method is monotonicity preserving.*

Proof. The proof can be found in [80]. □

Theorem D.2. *A linear TVD difference scheme is at most of first order.*

Proof. The proof can be found in [126]. □

Therefore, the numerical schemes must be nonlinear and must satisfy the TVD property to be higher order non-oscillatory i.e., high resolution schemes. The details about constructing the high resolution schemes can be found in [126, 81, 80, 128].

Abbreviations

T_1 spin-lattice relaxation time.

T_2 spin-spin relaxation time.

T_2^* effective spin-spin relaxation time.

ACS auto-calibrated signal.

ADC analog-to-digital converter.

BW bandwidth.

CCW counterclockwise.

CFD computational fluid dynamics.

CFL Courant-Friedrich-Lewy.

CG conjugate gradient.

CPU Central Processing Unit.

CSF cerebrospinal fluid.

CT computerized tomography.

CW clockwise.

DCF density compensation function.

DE differential equation.

DFT discrete Fourier transform.

dGFEM discontinuous Galerkin method.

EMF electromotive force.

EPI echo planar imaging.

FC flow compensation.

FDM finite difference method.

FFT fast Fourier transform.

FID free induction decay.

FLASH fast low angle shot.

FOV field of view.

FVM finite volume method.

GE gradient echo.

GPU graphical processing unit.

GRAPPA generalized auto-calibrating partially parallel acquisition.

IFFT inverse fast Fourier transform.

IRGNM iteratively regularized Gauss-Newton method.

IVP initial value problem.

LBM lattice-Boltzmann method.

MR magnetic resonance.

MRA magnetic resonance angiography.

MRI magnetic resonance imaging.

NLINV nonlinear inverse reconstruction.

NMR nuclear magnetic resonance.

NUFFT non-uniform fast Fourier transform.

ODE ordinary differential equation.

PC MRI phase contrast imaging.

PCA principal component analysis.

PD proton density.

PDE partial differential equation.

PET positron emission tomography.

RARE rapid acquisition with relaxation enhancement.

RF radio-frequency.

RK Runge-Kutta.

RK45 4-5th order adaptive Runge-Kutta.

ROI region of interest.

RSS root of sum of squares.

SENSE sensitivity encoding.

SMASH simultaneous acquisition of spatial harmonics.

SNR signal-to-noise ratio.

SSFP Steady-state Free Precession.

SWSS symmetrically weighted sequential operator splitting.

TE echo time.

TR repetition time.

TVD total variation diminishing.

VENC velocity encoding.

References

1. Abragam, A. *The principles of nuclear magnetic resonance* (Oxford University Press, 1961).
2. Ahnert, K. & Mulansky, M. Odeint-Solving ordinary differential equations in C++. *arXiv preprint arXiv:1110.3397* (2011).
3. Andria, G., Attivissimo, F., Cavone, G. & Lanzolla, A. M. L. Acquisition times in magnetic resonance imaging: Optimization in clinical use. *IEEE Transactions on Instrumentation and Measurement* **58,** 3140–3148 (2009).
4. Antsaklis, P. J. & Michel, A. N. *A linear systems primer* 1–517 (Springer, 2007).
5. Atkinson, D., Brant-Zawadzki, M., Gillan, G., Purdy, D. & Laub, G. Improved MR angiography: magnetization transfer suppression with variable flip angle excitation and increased resolution. en. *Radiology* **190,** 890–4 (Mar. 1994).
6. Auzinger, W., Herfort, W., Hofstätter, H. & Koch, O. Setup of Order Conditions for Splitting Methods, 1–12 (2016).
7. Axel, L. Blood flow effects in magnetic resonance imaging. *Journal of Chemical Information and Modeling* **53,** 1689–1699 (1984).
8. Bakushinsky, A. & Kokurin, M. *Iterative methods for approximate solution of inverse problems* (Springer Science & Business Media, 2005).
9. Bátkai, A., Csomós, P. & Nickel, G. Operator splittings and spatial approximations for evolution equations. *Journal of Evolution Equations* **9,** 613–636 (2009).
10. Beatty, P. J., Nishimura, D. G. & Pauly, J. M. Rapid gridding reconstruction with a minimal oversampling ratio. *IEEE Transactions on Medical Imaging* **24,** 799–808 (2005).
11. Bell, N. & Hoberock, J. in *GPU Computing Gems: Jade Edition* 359–371 (Elsevier, 2011).

12. Benoit-Cattin, H., Collewet, G., Belaroussi, B., Saint-Jalmes, H. & Odet, C. The SIMRI project: A versatile and interactive MRI simulator. *Journal of Magnetic Resonance* **173,** 97–115 (2005).

13. Bernstein, M. A., King, K. F. & Zhou, X. J. *Handbook of MRI Pulse Sequences* 1040 (Elsevier, 2004).

14. Bernstein, M. A., Mladen, G., Brosnan, T. & Pelc, N. J. Reconstructions of phase contrast, phased Array multicoil data. *Magnetic Resonance in Medicine* **32,** 330–334 (1994).

15. Bittoun, J., Taquin, J. & Sauzade, M. A computer algorithm for the simulation of any Nuclear Magnetic Resonance (NMR) imaging method. *Magnetic Resonance Imaging* **2,** 113–120 (1984).

16. Bjørhus, M. Operator splitting for abstract Cauchy problems. *IMA journal of Numerical Analysis* **18,** 419–443 (1998).

17. Bloch, F. Nuclear induction. *Physical Review* **70,** 460–474 (1946).

18. Block, K. T. *Advanced Methods for Radial Data Sampling in Magnetic Resonance Imaging Dissertation* PhD thesis (Georg-August-Universität Göttingen, 2008).

19. Bryant, D. & Payne, J. Measurement of flow with NMR imaging using a gradient pulse and phase difference technique. *Journal of computer Assisted Tomography* **8,** 588–593 (1984).

20. Buades, A., Coll, B. & Morel, J.-M. A non-local algorithm for image denoising. *Computer Vision and Pattern* **2,** 60–65 (2005).

21. Buehrer, M., Pruessmann, K. P., Boesiger, P. & Kozerke, S. Array compression for MRI with large coil arrays. *Magnetic Resonance in Medicine* **57,** 1131–1139 (2007).

22. Butcher, J. C. *Numerical Methods for Ordinary Differential Equations* (John Wiley & Sons, 2008).

23. Carr, H. Y. Steady-state free precession in nuclear magnetic resonance. *Physical Review* **112,** 1693–1701 (1958).

24. Carr, J. & Carroll, T. *Magnetic Resonance Angiography: Principles and Applications* (Springer, 2011).

25. Cash, J. R. & Karp, A. H. A variable order Runge-Kutta method for initial value problems with rapidly varying right-hand sides. *ACM Transactions on Mathematical Software* **16,** 201–222 (Sept. 1990).

26. Cengel, Y., Turner, R. & Smith, R. *Fundamentals of Thermal-Fluid Sciences* 5th ed. (McGraw-Hill Education, 2016).

27. Courant, R., Friedrichs, K. & Lewy, H. On the partial difference equations of mathematical physics. *IBM journal* **11,** 215–234 (1967).

28. Courant, R., Friedrichs, K. & Lewy, H. Über die partiellen Differenzengleichungen der mathematischen Physik. *Mathematische annalen* **100,** 32–74 (1928).

29. Crawley, A. P., Wood, M. L. & Henkelman, R. M. Elimination of transverse coherences in FLASH MRI. *Magnetic resonance in medicine* **8,** 248–260 (1988).

30. Di Pietro, D. A. & Ern, A. *Mathematical aspects of discontinuous Galerkin methods* (Springer Science & Business Media, 2011).

31. Dolejší, V. & Feistauer, M. *Discontinuous Galerkin Method* (Springer, 2015).

32. Dormand, J. R. & Prince, P. J. A reconsideration of some embedded Runge-Kutta formulae. *Journal of Computational and Applied Mathematics* **15,** 203–211 (1986).

33. Dreha-Kulaczewski, S. *et al.* Inspiration Is the Major Regulator of Human CSF Flow. *Journal of neuroscience* **35,** 2485–91 (2015).

34. Drobnjak, I., Gavaghan, D., Süli, E., Pitt-Francis, J. & Jenkinson, M. Development of a functional magnetic resonance imaging simulator for modeling realistic rigid-body motion artifacts. *Magnetic Resonance in Medicine* **56,** 364–380 (2006).

35. Edelman, R. R. *et al.* Quiescent-interval single-shot unenhanced magnetic resonance angiography of peripheral vascular disease: Technical considerations and clinical feasibility. *Magnetic Resonance in Medicine* **63,** 951–958 (2010).

36. Engl, H. W., Hanke, M. & Neubauer, A. *Regularization of Inverse Problems* 322 (Springer, 2000).

37. Ern, A. & Guermond, J.-L. *Theory and practice of finite elements* (Springer Science & Business Media, 2013).

38. Faragó, I. & Havasiy, A. *Operator splittings and their applications* (Nova Science Publishers, Inc., 2009).

39. Fehlberg, E. *Classical fifth, sixth, seventh and eighth order Runge–Kutta formulas with stepsize control.* tech. rep. (1968).

40. Fehlberg, E. Klassische runge-kutta-formeln vierter und niedrigerer ordnung mit schrittweiten-kontrolle und ihre anwendung auf waermeleitungsprobleme. *Computing* **6,** 61–71 (1970).

41. Fessler, J. A. & Sutton, B. P. Nonuniform fast Fourier transforms using min-max interpolation. *IEEE Transactions on Signal Processing* **51,** 560–574 (2003).

42. Frahm, J., Haase, A. & Matthaei, D. Rapid NMR imaging of dynamic processes using the FLASH technique. *Magnetic Resonance in Medicine* **3,** 321–7 (1986).

43. Frahm, J., Haase, A., Matthaei, D., Hänicke, W. & Merboldt, K.-D. *Verfahren und Einrichtung zur schnellen Akquisition von Spinresonanzdaten für eine ortsaufgelöste Untersuchung eines Objekts* 1985.

44. Frahm, J., Hänicke, W. & Merboldt, K. D. Transverse coherence in rapid FLASH NMR imaging. *Journal of Magnetic Resonance (1969)* **72,** 307–314 (1987).

45. Gao, J. H. & Gore, J. C. A numerical investigation of the dependence of NMR signal from pulsatile blood flow in CINE pulse sequences. *Medical physics* **18,** 342–349 (1991).

46. Gautschi, W. *Numerical Analysis* (Birkhäuser Boston, Boston, 2012).

47. Geiser, J. *Decomposition Methods for Differential Equations Teory and Applications* (CRC Press, 2009).

48. Gottlieb, S. & Shu, C.-W. Total Variation Diminishing Runge-Kutta Schemes. *Mathematics of Computation* **67,** 73–85 (1998).

49. Gottlieb, S., Shu, C.-W. & Tadmor, E. Strong Stability-Preserving High-Order Time Discretization Methods. *SIAM Review* **19,** 786–789 (2001).

50. Griswold, M. A. *et al.* Generalized Autocalibrating Partially Parallel Acquisitions (GRAPPA). *Magnetic Resonance in Medicine* **47,** 1202–1210 (2002).

51. Grivet, J.-P. Simulation of magnetic resonance experiments. *American Journal of Physics* **61,** 1133–1139 (1993).

52. Guennebaud, G., Jacob, B., Avery, P., Bachrach, A., Barthelemy, S., *et al. Eigen v3* 2010.

53. Guo, Y. & Jiang, X. Simulations of the stent artifacts in magnetic resonance imaging. *IEEE Transactions on Magnetics* **48,** 659–662 (2012).

54. Haacke, E. M., Brown, R. W., Thompson, M. R. & Venkatesan, R. *Magnetic Resonance Imaging: Physical Principles and Sequence Design* (Wiley Blackwell, 1999).

55. Haase, A., Frahm, J., Matthaei, D., Hanicke, W. & Merboldt, K.-D. FLASH imaging. Rapid NMR imaging using low flip-angle pulses. *Journal of Magnetic Resonance (1969)* **67,** 258–266 (1986).

56. Hahn, E. L. Detection of sea-water motion by nuclear precession. *Journal of geophysical research* **65,** 776–777 (1960).

57. Hairer, E., Lubich, C. & Wanner, G. *Geometric numerical integration structure-preserving algorithms for ordinary differential equations* 644 (Springer, 2006).

58. Hairer, E., Nørset, S. & Wanner, G. *Solving Ordinary Differential Equations I* (Springer, 1993).

59. Hansen, E. & Ostermann, A. Dimension splitting for evolution equations. *Numerische Mathematik* **108,** 557–570 (Feb. 2008).

60. Hansen, E. *et al.* Exponential splitting for unbounded operators. *Mathematics of Computation* **78,** 1485–1496 (Sept. 2009).

61. Hedstrom, G. W. Models of difference schemes for $u_t + u_x = 0$ by partial differential equations. *Mathematics of Computation* **29,** 969–977 (1975).

62. Hendrick, R. E., Kneeland, J. B. & Stark, D. D. Maximizing signal-to-noise and contrast-to-noise ratios in FLASH imaging. *Magnetic Resonance Imaging* **5,** 117–127 (1987).

63. Hennig, J., Müri, M., Brunner, P. & Friedburg, H. Quantitative flow measurement with the fast Fourier flow technique. *Radiology* (1988).

64. Hennig, J., Nauerth, A. & Friedburg, H. RARE imaging: a fast imaging method for clinical MR. *Magnetic Resonance in Medicine* **3,** 823–833 (1986).

65. Hirsch, C. *Computational methods for inviscid and viscous flows* (Wiley, 1990).

66. Hirsch, C. *Numerical Computation of Internal and External Flows: The Fundamentals of Computational Fluid Dynamics* (Butterworth-Heinemann, 2007).

67. Hundsdorfer, W. & Verwer, J. G. A note on splitting errors for advection-reaction equations. *Applied Numerical Mathematics* **18,** 191–199 (1995).

68. Hundsdorfer, W. & Verwer, J. G. *Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations* (Springer Science & Business Media, 2003).

69. Iserles, A. *A First Course in the Numerical Analysis of Differential Equations* 400 (Cambridge University Press, 1996).

70. Jakob, P. M., Griswold, M. A., Edelman, R. R. & Sodickson, D. K. AUTO-SMASH: A self-calibrating technique for SMASH imaging. *Magnetic Resonance Materials in Physics, Biology and Medicine* **7,** 42–54 (1998).

71. Joseph, A. A. *Real-time MRI of Moving Spins Using Undersampled Radial FLASH* PhD thesis (Julius-Maximilians-Universität Würzburg, 2013).

72. Joseph, A. A. *et al.* Real-time phase-contrast MRI of cardiovascular blood flow using undersampled radial fast low-angle shot and nonlinear inverse reconstruction. *NMR in Biomedicine* **25,** 917–924 (2012).

73. Jou, L. D. & Saloner, D. A numerical study of magnetic resonance images of pulsatile flow in a two dimensional carotid bifurcation a numerical study of MR images. *Medical Engineering and Physics* **20,** 643–652 (1998).

74. Jou, L. D., van Tyen, R., Berger, S. A. & Saloner, D. Calculation of the magnetization distribution for fluid flow in curved vessels. *Magnetic Resonance in Medicine* **35,** 577–584 (1996).

75. Jurczuk, K. *et al.* Computational modeling of MR flow imaging by the lattice Boltzmann method and Bloch equation. *Magnetic Resonance Imaging* **31,** 1163–1173 (2013).

76. Klosowski, J. & Frahm, J. Image denoising for real-time MRI. *Magnetic resonance in medicine* **00,** 1–13 (2016).

77. Kollmeier, J. *Perfusion Phantom Studies using Real-time Magnetic Resonance Imaging* MA thesis (Georg-August Universität Göttingen, 2016).

78. Lanser, D. & Verwer, J. G. Analysis of operator splitting for advection-diffusion-reaction problems from air pollution modelling. *Journal of Computational and Applied Mathematics* **111,** 201–216 (1999).

79. Lax, P. D. & Richtmyer, R. D. Survey of the stability of linear finite difference equation. *Communications on Pure and Applied Mathematics* **2,** 267–293 (1956).

80. LeVeque, R. J. *Numerical Methods for Conservation Laws* (Birkhäuser Basel, Basel, 1992).

81. LeVeque, R. *Finite Volume Methods for Hyperbolic Problems* (Cambridge University Press, 2002).

82. Lewis, D. P., Tsui, B. M. W. & Moran, P. R. Velocity sensitivity of slice-selective excitation. *Magnetic Resonance Imaging* **16,** 907–916 (1998).

83. Liang, Z.-P. & Lauterbur, P. C. *Principles of Magnetic Resonance Imaging: A Signal Processing Perspective* (IEEE Press, 2000).

84. Lorthois, S., Stroud-Rossman, J., Berger, S., Jou, L. D. & Saloner, D. Numerical simulation of magnetic resonance angiographies of an anatomically realistic stenotic carotid bifurcation. *Annals of Biomedical Engineering* **33,** 270–283 (2005).

85. Majda, A. & Ralston, J. Discrete shock profiles for systems of conservation laws. *Communications on Pure and Applied Mathematics* **32,** 445–482 (1979).

86. Mansfield, P. Multi-planar image formation using NMR spin echoes. *Journal of Physics C: Solid State Physics* **10,** 55–58 (1977).

87. Markl, M., Frydrychowicz, A., Kozerke, S., Hope, M. & Wieben, O. 4D flow MRI. *Journal of Magnetic Resonance Imaging* **36,** 1015–1036 (2012).

88. Markl, M. *et al.* Time-resolved three-dimensional phase-contrast MRI. *Journal of Magnetic Resonance Imaging* **17,** 499–506 (2003).

89. Marshall, I. Computational simulations and experimental studies of 3D phase-contrast imaging of fluid flow in carotid bifurcation geometries. *Journal of Magnetic Resonance Imaging* **31,** 928–34 (2010).

90. Marshall, I. Simulation of in-plane flow imaging. *Concepts in Magnetic Resonance* **11,** 379–392 (1999).

91. McKenzie, C. A., Ohliger, M. A., Yeh, E. N., Price, M. D. & Sodickson, D. K. Coil-by-coil image reconstruction with SMASH. *Magnetic Resonance in Medicine* **46,** 619–623 (2001).

92. McLachlan, R. I. & Quispel, G. R. W. Splitting methods. *Acta Numerica* **11,** 341–434 (2002).

93. Miyazaki, M. & Akahane, M. Non-contrast enhanced MR angiography: Established techniques. *Journal of Magnetic Resonance Imaging* **35,** 1–19 (2012).

94. Moran, P. R., Saloner, D. & Tsui, B. W. NMR velocity-selective excitation composites for flow and motion imaging and suppression of static tissue signal. *IEEE Trans. Med. Imaging* **6,** 141–147 (1987).

95. Moran, P. R. A flow velocity zeugmatographic interlace for NMR imaging in humans. *Magnetic Resonance Imaging* **1,** 197–203 (1982).

96. Mulkern, R. V. & Williams, M. L. The general solution to the Bloch equation with constant rf and relaxation terms: application to saturation and slice selection. *Medical physics* **20,** 5–13 (1993).

97. Murase, K. & Tanki, N. Numerical solutions to the time-dependent Bloch equations revisited. *Magnetic Resonance Imaging* **29,** 126–131 (2011).

98. O 'sullivan, J. D. A Fast Sinc Function Gridding Algorithm for Fourier Inversion in Computer Tomography. *IEEE Transactions on Medical Imaging* **4,** 200–207 (1985).

99. O'Donnell, M. NMR blood flow imaging using multiecho, phase contrast sequences. *Medical physics* **12,** 59–64 (1985).

100. Olsson, M. B. E., Wirestam, R. & Persson, B. R. R. A computer simulation program for MR imaging: Application to RF and static magnetic field imperfections. *Magnetic resonance in medicine* **34,** 612–617 (1995).

101. Paul Lauterbur. Image formation by induced local interactions: Examples employing nuclear magnetic resonance. *Nature* (1973).

102. Perko, L. *Differential Equations and Dynamical Systems* 568 (Springer New York, 2001).

103. Petersson, S. *Simulation of Phase Contrast MRI Measurements from Numerical Flow Data* tech. rep. (Linköpings universitet, Linköping, Sweden, 2008), 45.

104. Petersson, S., Dyverfeldt, P., Gårdhagen, R., Karlsson, M. & Ebbers, T. Simulation of phase contrast MRI of turbulent flow. *Magnetic Resonance in Medicine* **64,** 1039–1046 (2010).

105. Pruessmann, K. P., Weiger, M., Börnet, P. & Boesiger, P. Advances in Sensitivity Encoding With Arbitrary k-Space Trajectories. *Magnetic resonance in medicine* **651,** 638–651 (2001).

106. Pruessmann, K. P., Weiger, M., Scheidegger, M. B. & Boesiger, P. SENSE: Sensitivity encoding for fast MRI. *Magnetic Resonance in Medicine* **42,** 952–962 (1999).

107. Purcell, E., Torrey, H. & Pound, R. Resonance Absorption by Nuclear Magnetic Moments in a Solid. *Physical Review* **69,** 37–38 (1946).

108. Quarteroni, A., Sacco, R. & Saleri, F. *Numerical Mathematics* (Springer Science & Business Media, 2010).

109. Rasche, V., De Boer, R. W., Holz, D. & Proksa, R. Continuous radial data acquisition for dynamic MRI. *Magnetic Resonance in Medicine* **34,** 754–761 (1995).

110. Roe, P. Some contributions to the modelling of discontinuous flows. *Large-scale computations in fluid mechanics* (1985).

111. Roeloffs, V., Voit, D. & Frahm, J. Spoiling without additional gradients: Radial FLASH MRI with randomized radiofrequency phases. *Magnetic resonance in medicine* (2015).

112. Roemer, P. B., Edelstein, W. A. & Hayes, C. E. The NMR phased array. *Magnetic Resonance in Medicine* **225,** 192–225 (1990).

113. Rott, N. Note on the History of Reynolds Number. *Annual Review of Fluid Mechanics* **22,** 1–11 (1990).

114. Shkarin, P. & Spencer, R. G. S. Direct simulation of spin echoes by summation of isochromats. *Concepts in Magnetic Resonance* **8,** 253–268 (1995).

115. Shkarin, P. & Spencer, R. G. S. Time domain simulation of Fourier imaging by summation of isochromats. *International journal of imaging systems and technology* **8,** 419–426 (1997).

116. Sideris, T. *Ordinary Differential Equations and Dynamical Systems* 557 (Atlantis Press, 2013).

117. Sigges, F. *Numerical Solution of Bloch-Equations for Incompressible Fluids* MA thesis (Institute of Computer Science, Göttingen, 2016).

118. Slichter, C. P. in *Principles of Magnetic Resonance* (Springer-Verlag, 1990).

119. Smale, S., Hirsch, M. W. & Devaney, R. L. *Differential Equations, Dynamical Systems, and an Introduction to Chaos, Second Edition (Pure and Applied Mathematics)* (Elsevier Academic Press, 2003).

120. Sodickson, D. K. & Manning, W. J. Simultaneous acquisition of spatial harmonics (SMASH): Fast imaging with radiofrequency coil arrays. *Magnetic Resonance in Medicine* **38,** 591–603 (1997).

121. Stöcker, T., Vahedipour, K., Pflugfelder, D. & Shah, N. J. High-performance computing MRI simulations. *Magnetic Resonance in Medicine* **64,** 186–193 (2010).

122. Strang, G. On the construction and comparison of difference schemes. *SIAM Journal on Numerical Analysis* **5,** 506–517 (1968).

123. Süli, E. & Mayers, D. *An Introduction to Numerical Analysis* (Cambridge University Press, 2003).

124. Sweby, P. High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM Journal on Numerical Analysis* (1984).

125. Teschl, G. *Ordinary Differential Equations and Dynamical Systems* 297 (American Mathematical Society, 2004).

126. Thomas, J. W. *Numerical Partial Differential Equations: Conservation Laws and Elliptic Equations* (Springer, 1999).

127. Thomas, J. W. *Numerical Partial Differential Equations: Finite Difference Methods* (Springer, 1995).

128. Toro, E. F. *Riemann solvers and numerical methods for fluid dynamics: a practical introduction* (Springer Science & Business Media, 2013).

129. Torrey, H. C. Transient nutations in nuclear magnetic resonance. *Physical Review* **76,** 1059–1068 (1949).

130. Trangenstein, J. *Numerical Solution of Hyperbolic Partial Differential Equations* (Cambridge University Press, 2009).

131. Uecker, M. *Nonlinear Reconstruction Methods for Parallel Magnetic Resonance Imaging* PhD thesis (Georg-August-Universität Göttingen, 2009).

132. Uecker, M., Hohage, T., Block, K. T. & Frahm, J. Image reconstruction by regularized nonlinear inversion - Joint estimation of coil sensitivities and image content. *Magnetic Resonance in Medicine* **60,** 674–682 (2008).

133. Uecker, M., Zhang, S. & Frahm, J. Nonlinear inverse reconstruction for real-time MRI of the human heart using undersampled radial FLASH. *Magnetic Resonance in Medicine* **63,** 1456–1462 (2010).

134. Van Leer, B. Towards the ultimate conservative difference scheme. II. Monotonicity and conservation combined in a second-order scheme. *Journal of Computational Physics* **14,** 361–370 (1974).

135. Van Tyen, R., Saloner, D., Jou, L. D. & Berger, S. MR imaging of flow through tortuous vessels: a numerical simulation. *Magnetic Resonance in Medicine* **31,** 184–195 (1994).

136. Versteeg, H. K. & Malaskekera, W. *An Introduction to Computational Fluid Dynamics* (Prentice Hall, 2007).

137. Viscopedia. *Viscopedia | A free encyclopedia for viscosity* 2016.

138. Walsh, D. O., Gmitro, A. F. & Marcellin, M. W. Adaptive reconstruction of phased array MR imagery. *Magnetic Resonance in Medicine* **43,** 682–690 (2000).

139. Wheaton, A. J. & Miyazaki, M. Non-contrast enhanced MR angiography: Physical principles. *Journal of Magnetic Resonance Imaging* **36,** 286–304 (2012).

140. Xanthis, C. G., Venetis, I. E., Chalkias, A. V. & Aletras, A. H. MRISIMUL: A GPU-based parallel approach to MRI simulations. *IEEE Transactions on Medical Imaging* **33,** 607–617 (2014).

141. Yeh, E. N., McKenzie, C. A., Ohliger, M. A. & Sodickson, D. K. Parallel magnetic resonance imaging with adaptive radius in k-space (PARS): Constrained image reconstruction using k-space locality in radiofrequency coil encoded data. *Magnetic Resonance in Medicine* **53,** 1383–1392 (2005).

142. Yuan, C. The solution of Bloch equations for flowing spins during selective pulse using a finite difference method. *Medical physics* **14,** 914–921 (1987).

143. Zamir, M. & Budwig, R. *Physics of Pulsatile Flow* **2** (Springer, 2000).

144. Zeidler, E. *Nonlinear Functional Analysis and Its Applications I: Fixed-Point Theorems* (Springer-Verlag, 1986).

145. Zhang, S. *Real-time Magnetic Resonance Imaging* PhD thesis (Georg-August-Universität Göttingen, 2009).

146. Zlatev, Z. & Dimov, I. Different splitting techniques with application to air pollution models. *International Journal of Environment and Pollution,* 1–23 (2008).