UTRECHT UNIVERSITY

MASTER'S THESIS

Radiofrequency pulse design through optimal control and model order reduction of the Bloch equation

Author:
Willem VAN VALENBERG

Supervisors:
Gerard L.G. SLEIJPEN
Alessandro SBRIZZI

July 8, 2015

correspondence to: w.vanvalenberg@students.uu.nl

Notation

γ	rad/s/T	gyromagnetic ratio
r, t	cm, ms	spatial and temporal variables
L	-	number of radiofrequency transmit coils
T_1, T_2	ms	longitudinal and transversel relaxation rates
M_0	A/m	equilibrium magnetization vector
M	A/m	magnetization vector
λ	-	costate vector
G	mT/m	gradients
B_0	T	static magnetic field strength
ΔB_0	Hz	static magnetic field inhomogeneity
$b_1^{(l)}$	mT	radiofrequency waveform of coil l
s_l	-	transmit field sensitivity of coil l
B_1	mT	radiofrequency magnetic field
N_s, N_t	-	number of discretized spatial and temporal points

Contents

Introduction

Magnetic resonance imaging (MRI) is a tool used mostly in the medical world for creating images of the human body for diagnosis. An advantage of MRI over other medical imaging modalities is that it does not require the use of ionizing radiation.

Images are based on magnetization dynamics of hydrogen protons in the body. The dynamics indicate the type of biological tissue the protons are a part of. Radiofrequency (RF) pulses bring the magnetization into an excited state, where it generates a signal. The response of the magnetization to an RF pulse is described by the *Bloch equation*. *Radiofrequency pulse design* deals with the inverse problem of finding an RF pulse that transfers the magnetization to a desired state. Most pulse design methods invert the Bloch equation under some strict assumptions. The optimal control method allows for more freedom in pulse design, at the cost of requiring the repeated calculation of the magnetization dynamic. Calculating the magnetization dynamic is computationally costly, and so is the optimal control method as a consequence. Finding a method for model order reduction of the Bloch equation can greatly reduce the simulation time. Leading to a more efficient optimal control method.

The thesis is organized into three parts. In the first chapter, the basics of MRI are introduced, a mathematical framework based on the Bloch equations is set up, the magnetization response for a given RF pulse is calculated, and a standard method for RF pulse design is introduced and tested.

In the second chapter, the optimal control framework is set up, different numerical methods for reducing computation time are discussed and implemented, and the method is used for improving the RF pulses of the first chapter.

In the third chapter, both spatial and temporal domain reduction are discussed in the context of the Bloch equations, and numerical results of temporal reduction technique are shown.

Chapter 1

Basics of magnetic resonance

The focus of this first chapter is analysing the Bloch equation, which describes the magnetization dynamics that are measured in MRI. This differential equation will be the basis of the whole thesis and this chapter describes its physical background, properties of the mathematical model and the numerical solution. At the end of the chapter a standard method for radiofrequency pulses design is introduced.

1.1 Physics

In this section, the physics that explain the magnetic resonance imaging process are discussed by following the treatment of [11]. The goal is not to give a full account of all physics involved but rather to give the rationale behind the Bloch equations and the functioning of an MR scanner. Some elements are skipped or simplified. The reader can find a more thorough description in the book by Haacke [11].

1.1.1 Magnetization

The magnetic moment of an object is represented by a three-dimensional vector and is used to describe the behaviour of the object under the influence of an external magnetic field. A single hydrogen atom or proton has both a charge and an inherent spin that together lead to a magnetic moment pointed along the rotation axis.
A group of protons that spin at the same frequency is called an isochromat. The protons within the isochromat have an average magnetic moment called the *magnetization*. It is denoted by the vector M, and measured in amperes per meter (A/m).

In our description of MRI, the scanned object is discretized into voxels of size V, where each voxel is supposed isochromat and thus has a single magnetization vector. This assumption is more accurate for small V. The response to an external magnetic field of all protons in a voxel can then be described by the response of the magnetization vector.

The external magnetic field B is represented by a three-dimensional vector field with units given in tesla (T). The field is spatially variable but the voxel size V is supposed to be small enough that the magnetic field is approximately constant within it and the local discretization error is neglected. The generation of the magnetic field by the MRI scanner system is the subject of the next section, first its effect on the magnetization is explained.

When considering the magnetization vector M of a voxel subject to a constant external magnetic field B, its dynamic is described by the *Bloch equations*:

$$\frac{dM}{dt} = \gamma M \times B + \frac{1}{T_1}(M_0 - M_\parallel) - \frac{1}{T_2}M_\perp. \qquad (1.1)$$

Where $\times$ denotes the outer product between two vectors in $\mathbb{R}^3$, M_0 the equilibrium magnetization state, $M_\parallel$ and $M_\perp$ respectively the components of M parallel and orthogonal to B, T_1 and T_2 the longitudinal and transversal relaxation times in seconds, and γ the gyromagnetic ratio in radians per second per tesla. All elements will be discussed in more detail in the following sections. Here it is noted that values of the spatial-dependent variables T_1, T_2 and M_0 are coupled to the tissue type the protons are found in, allowing identification of different biological features. The value of the constant γ is the same for each voxel.

The dynamics of the magnetization M as given by equation (1.1) can be split in two effects:

1. rotation around B, due to the magnetic moment of the spins,

2. decay to an equilibrium state M_0, due to the energy loss of the protons through interaction of their magnetic moments.

The first effect is given by the first term of (1.1) and will be discussed in the following section, while the second effect is described by the other two terms and treated after.
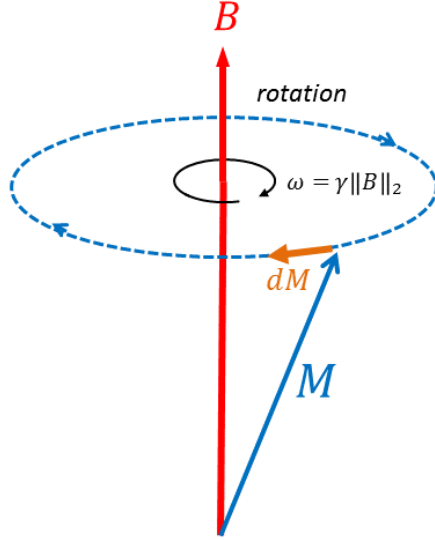
Figure 1.1: Rotation of the magnetization vector M around the external magnetic field B.

Rotation

Neglecting the decay effect for the moment and focusing only on the rotation effect, the magnetization dynamic is described by:

$$\frac{dM}{dt} = \gamma M \times B. \tag{1.2}$$

The gyromagnetic ratio, γ, relates the strength of the magnetic field with the speed of clockwise rotation given by the angular frequency ω (rad/s). For a hydrogen proton the gyromagnetic ratio is:

$$\gamma = 2.675 \times 10^8 \text{ rad/s/T}. \tag{1.3}$$

For a given magnetic field B, the angular frequency ω equals:

$$\omega \equiv |\frac{d\phi}{dt}| = \gamma \|B\|_2, \tag{1.4}$$

where ϕ denotes the phase of the magnetization in radian. Figure 1.1 illustrates the solution of equation (1.2).

To simplify notation, the vectors are represented in the Cartesian coordinate system with basis vectors $\hat{x}$, $\hat{y}$ and $\hat{z}$. Setting:

$$M = M_x\hat{x} + M_y\hat{y} + M_z\hat{z}, \qquad B = B_x\hat{x} + B_y\hat{y} + B_z\hat{z},$$

equation (1.1) can be represented in matrix form:

$$\frac{d}{dt}\begin{pmatrix} M_x \\ M_y \\ M_z \end{pmatrix} = \gamma M \times B = -\gamma B \times M$$

$$= -\gamma \begin{pmatrix} 0 & -B_z & B_y \\ B_z & 0 & -B_x \\ -B_y & B_x & 0 \end{pmatrix} \begin{pmatrix} M_x \\ M_y \\ M_z \end{pmatrix} \qquad (1.5)$$

$$= -\gamma \mathbf{B} M,$$

where the minus sign is used such that the matrix $\mathbf{B}$, defined in equation (1.5), generates a positive, counterclockwise, rotation. Note that all coordinates in (1.5) are real valued, and the matrix $\mathbf{B}$ is skew-symmetric: $\mathbf{B}^T = -\mathbf{B}$, with eigenvalues $\lambda = 0, \pm i\|B\|_2$. The matrix $\mathbf{B}$ can be decomposed as:

$$\mathbf{B} = B_x \mathbf{S}_x + B_y \mathbf{S}_y + B_z \mathbf{S}_z, \qquad (1.6)$$

with:

$$\mathbf{S}_x = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}, \quad \mathbf{S}_y = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix}, \quad \mathbf{S}_z = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

The matrices $\mathbf{S}_x, \mathbf{S}_y, \mathbf{S}_z$ form a basis for the space of real skew-symmetric 3×3 matrices, which in the following will be shown to generate rotations. The basis matrices have eigenvalues $\lambda = 0, \pm i$, and give counterclockwise rotation around respectively $\hat{x}, \hat{y}$ and $\hat{z}$. Furthermore, the matrices satisfy the special commutator rules:

$$[\mathbf{S}_x, \mathbf{S}_y] = \mathbf{S}_z, \quad [\mathbf{S}_z, \mathbf{S}_x] = \mathbf{S}_y, \quad [\mathbf{S}_y, \mathbf{S}_z] = \mathbf{S}_x, \qquad (1.7)$$

with $[\mathbf{X}, \mathbf{Y}] := \mathbf{X}\mathbf{Y} - \mathbf{Y}\mathbf{X}$. And furthermore: $\mathbf{S}_k^3 = -\mathbf{S}_k$, for $k = x, y, z$.

Supposing the magnetic field B is constant over time, the solution of the homogeneous differential equation (1.5) with initial value $M(t_0)$ is given by:

$$M(t) = e^{-\gamma(B_x \mathbf{S}_x + B_y \mathbf{S}_y + B_z \mathbf{S}_z)t} M(t_0) = e^{-\gamma \mathbf{B}t} M(t_0). \qquad (1.8)$$

To be consistent with Figure 1.1, the matrix exponential in equation (1.8) is proven to define a rotation around B. A rotation matrix is defined as a an orthonormal matrix with determinant equal to 1. The exponent of the skew-symmetric matrix $\mathbf{X}$ is orthonormal since:

$$e^{\mathbf{X}} \left(e^{\mathbf{X}}\right)^T = e^{\mathbf{X}} e^{\mathbf{X}^T} = e^{\mathbf{X}} e^{-\mathbf{X}} = I. \qquad (1.9)$$

The determinant equals:

$$\det(e^{\mathbf{X}}) = e^{\text{tr}(\mathbf{X})} = 1, \qquad \text{since } \text{tr}(\mathbf{X}) = 0. \tag{1.10}$$

Setting $\mathbf{X} = -\gamma\mathbf{B}t$, it follows that the matrix exponential $e^{-\gamma\mathbf{B}t}$ from equation (1.8) is a rotation matrix. The matrix can be described by a simple formula, by noting that $\mathbf{B}^3 = -\|B\|_2^2\mathbf{B}$, such that:

$$
\begin{aligned}
e^{-\gamma\mathbf{B}t} &= \mathbf{I} + \sum_{i=0}^{\infty}\frac{(-\gamma\mathbf{B}t)^{2i+1}}{(2i+1)!} + \sum_{i=1}^{\infty}\frac{(-\gamma\mathbf{B}t)^{2i}}{(2i)!} \\
&= \mathbf{I} + \frac{1}{\|B\|_2}\sum_{i=0}^{\infty}\frac{(-1)^i(-\gamma\|B\|_2 t)^{2i+1}}{(2i+1)!}\mathbf{B} + \frac{1}{\|B\|_2^2}\sum_{i=1}^{\infty}\frac{(-1)^{i-1}(-\gamma\|B\|_2 t)^{2i}}{(2i)!}\mathbf{B}^2 \\
&= \mathbf{I} + \frac{\sin(-\gamma\|B\|_2 t)}{\|B\|_2}\mathbf{B} + \frac{1-\cos(-\gamma\|B\|_2 t)}{\|B\|_2^2}\mathbf{B}^2.
\end{aligned}
\tag{1.11}
$$

Equation (1.11) is known as Rodrigues' rotation formula.

The matrix exponential $e^{-\gamma\mathbf{B}t}$ leaves B fixed since by equation (1.5):

$$\mathbf{B}B = B \times B = 0, \tag{1.12}$$

and therefore by equation (1.11), $e^{-\gamma\mathbf{B}t}B = B$. Furthermore, the exponent is an orthogonal matrix and hence length preserving.

Since there is some freedom in choosing the basis vectors $\hat{x}, \hat{y}, \hat{z}$, here the direction of the z-axis is chosen to be aligned with the magnetic field B:

$$B = B_z\hat{z}, \tag{1.13}$$

leading to the rotation taking place in the xy-plane. The magnetization is split into a component aligned with the magnetic field, $M_\parallel$, and a component orthogonal to it, $M_\perp$:

$$M_\parallel := M_z\hat{z}, \qquad\qquad M_\perp := M_x\hat{x} + M_y\hat{y}. \tag{1.14}$$

In this case, equation (1.2) can be decomposed into two parts:

$$\frac{dM_z}{dt} = 0, \qquad\qquad \frac{dM_\perp}{dt} = \gamma M_\perp \times B. \tag{1.15}$$

Decay

The rotation does not describe the whole magnetization dynamics, since there is also slowly decays to an equilibrium state. The equilibrium state, M_0, is given by:

$$M_0 := m_0 \hat{z}, \qquad \text{with} \ \ m_0 = cP_D \|B\|_2 / T. \tag{1.16}$$

In the equilibrium state the direction of the magnetization is aligned with B while the magnitude is a function of the voxel-dependent proton density P_D, field strength $\|B\|_2$, and temperature T (c is a physical constant). The decay can be split into two effects: longitudinal and transversal relaxation.

Through longitudinal relaxation, the aligned magnetization, $M_{\parallel} = M_z \hat{z}$, grows to its equilibrium state M_0 over time due to spin-lattice interaction. This is equivalent to M_z returning to m_0. The rate at which the magnetization M_z returns to m_0 is proportional to the inverse of the so-called *longitudinal relaxation* time T_1, given in seconds:

$$\frac{dM_z}{dt} = \frac{1}{T_1}(m_0 - M_z). \tag{1.17}$$

Note that this is equivalent to the second term in equation (1.1). The longitudinal relaxation time T_1 is measurable with experiments and depends both on the magnetic field strength and temperature, but most importantly on the tissue type the proton is found in.

The transversal relaxation describes the loss in magnetization in the xy-plane and it is due to relaxation to the equilibrium state and dephasing of the protons within the voxel V. The magnetization vector is the sum of a large number of magnetic moments. The moments will have a slight difference in rotation frequency. As a consequence, the protons accumulate a phase difference over time, or dephase, reducing the length of the average transversal magnetization as shown in Figure 1.2. The loss of transversal magnetization is described by:

$$\frac{dM_{\perp}}{dt} = -\frac{1}{T_2}M_{\perp}. \tag{1.18}$$

There are two reasons for the dephasing. First there is spin-spin interaction which means that the magnetic field generated by one proton influences the rotation of another proton causing them to dephase. The rate at which this happens equals the inverse of T_2 as shown in equation (1.18) and it is tissue dependent. It is supposed that the spin-spin interaction between voxels is
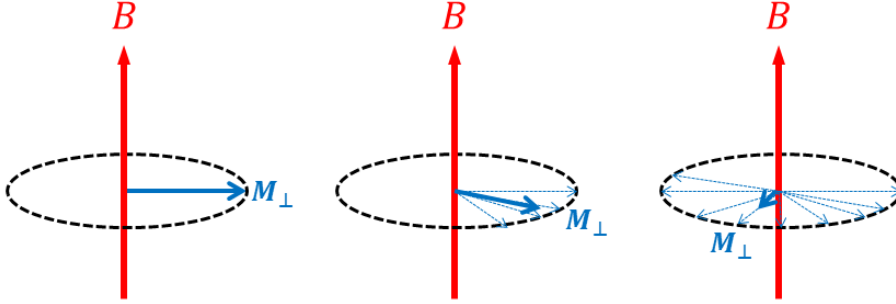
Figure 1.2: Illustration of loss of transversal magnetization due to dephasing of spins.

negligible, such that the T_2 relaxation can be considered at a voxel level. The second effect is due to small inhomogeneities in the external magnetic field within V. This last effect will be ignored for now, but it is useful to know that it can be removed by using appropriate pulse sequences. In general it is the case that $T_2 \leq T_1$.

The Bloch equations (1.1) are a combination of the rotation from equation (1.2) with the relaxation effects in equations (1.17) and (1.18). In the following, the Bloch equations will be described in matrix form:

$$\frac{d}{dt}\begin{pmatrix} M_x \\ M_y \\ M_z \end{pmatrix} = \begin{pmatrix} -1/T_2 & \gamma B_z & -\gamma B_y \\ -\gamma B_z & -1/T_2 & \gamma B_x \\ \gamma B_y & -\gamma B_x & -1/T_1 \end{pmatrix} \begin{pmatrix} M_x \\ M_y \\ M_z \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ m_0/T_1 \end{pmatrix},$$
$$(1.19)$$

which is shortened to:

$$\dot{M}(t) = \mathbf{\Omega}(t)M(t) + M_0/T_1. \qquad (1.20)$$

As mentioned at the beginning of the section, equation (1.19) forms the basis of MRI and this thesis. Images can be created based on the different values of T_1, T_2 and the proton density which scales with m_0. These values can be determined by measuring the transverse magnetization dynamic for a given external magnetic field B. Generating the magnetic field using the scanner system is the topic of the next section.

1.1.2   Generating the magnetic field

A schematic overview of an MRI scanner is given in Figure 1.3. The magnetic field is controlled using a superposition of three different types of magnetic
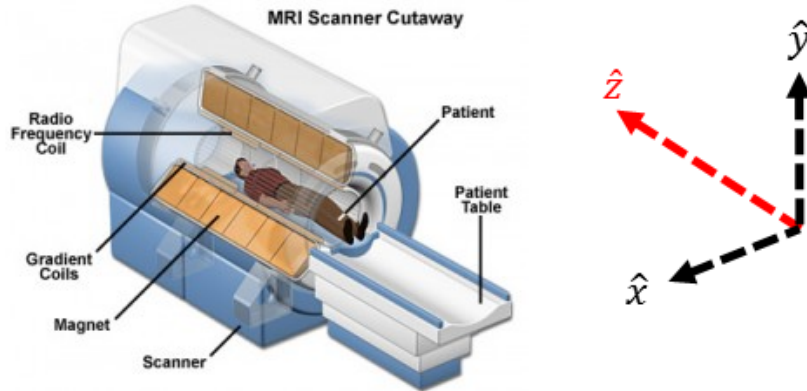
Figure 1.3: Schematic of MRI scanner

field whose directions are either aligned with the z-axis or orthogonal to it. Aligned with $\hat{z}$ and part of B_z are both the main magnetic and the so-called *gradient* field, while the radiofrequency pulses are played out orthogonally to the main field and form B_x and B_y. The main magnet generates a time-independent magnetic field, while both the gradient and radiofrequency field are generated using smaller coils that can be varied over time.

Main magnetic field

The main or static magnetic field is denoted by B_0 and has a strength of 1.0-3.0 T for clinical scanners while experimental scanners can go up to 11 T. Since B_0 has a constant magnetic field strength which is much higher than the other magnetic fields, the z-axis is associated with the direction of B_0 and in the equilibrium state, M_0, the magnetization is aligned with B_0. The rotation frequency of the magnetization due to the static magnetic field as given by equation (1.4) is called the *Larmor frequency* ω_0:

$$\omega_0 = \gamma \|B_0\|_2. \tag{1.21}$$

In practice, there is an uncontrollable spatial variation in the main magnetic field called *field inhomogeneity* ΔB_0 . These variations are generated by both imperfections of the scanner system as physical properties of the object, and are usually measured at the start of scanning. An example inhomogeneity map of a spherical phantom at 7T is given in Figure 1.4. The field inhomogeneity is converted from tesla to frequency offset in hertz through equation (1.21).
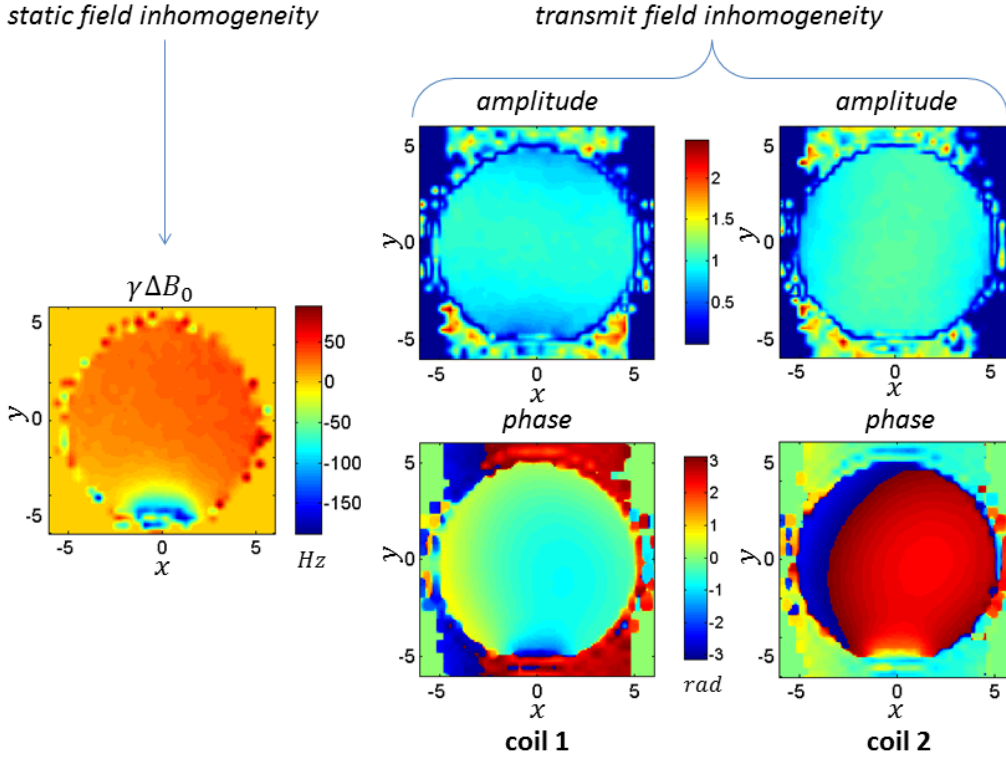
Figure 1.4: Example of field inhomogeneities for a slice of a disk-shaped phantom, in case of a two-coil 7T system. Spatial resolution in centimetres.

The gradient magnetic field is generated by the so-called *gradient waveforms* which are defined by a vector-valued function G with units in tesla per meter:

$$G(t) := (G_x(t), G_y(t), G_z(t))^T \in \mathbb{R}^3. \tag{1.22}$$

The time derivative of the gradient waveform is called the *slew rate*. The gradient magnetic field lays in the $\hat{z}$-direction and varies linear in strength along each spatial direction. Denoting the spatial position $r := (x, y, z)^T$, the gradient magnetic field at this position at time t equals:

$$(G(t) \cdot r)\hat{z} \in \mathbb{R}^3, \tag{1.23}$$

where $\cdot$ denotes the inner product of two vectors in $\mathbb{R}^3$. Due to system limitations, the gradients have a maximal amplitude: $|G(t)| \leq G_{max}$, and slew rate: $|\frac{dG}{dt}(t)| \leq S_{max}$.

Superimposing the field inhomogeneities ΔB_0 and gradient field on the main magnetic field, gives the following formula for the z component of the external

magnetic field at position r and time t:

$$B_z(r,t) = \|B_0\|_2 + \Delta B_0(r) + G(t) \cdot r. \qquad (1.24)$$

Radio-frequency magnetic field

The magnetic field generated by the radiofrequency (RF) coils is orthogonal to the B_0 field. Interpreting $\mathbb{R}^2$ as the complex plane, with values in the $\hat{x}$-direction as real part and in the $\hat{y}$-direction as imaginary part, allows the convenient representation of the vector $B_x \hat{x} + B_y \hat{y}$, as complex number B_1:

$$B_1(r,t) := B_x(r,t) + iB_y(r,t). \qquad (1.25)$$

The value of B_1 is a weighted average of the magnetic fields produced by the different coils:

$$B_1(r,t) = \sum_{l=1}^{L} s_l(r) b_1^{(l)}(t). \qquad (1.26)$$

Here s_l is the transmit field sensitivity of the lth coil and $b_1^{(l)}$ is the corresponding RF waveform; both are complex numbers. The transmit field sensitivities at high-field MR are due to interference and reflection of the radiofrequency fields within the object, and can be measured. The RF pulses are essential for manipulating the magnetization which is required for imaging. High RF pulse amplitude should be avoided since it leads to heating of the tissue.

1.2 Solving the Bloch equation

Equation (1.19) allows the calculation of the magnetization through time for given radiofrequency and gradient waveforms. This solution is called the *response* of the magnetization and the subject of this section is how to explicitly determine it. This is done in three steps: first a change of variables is performed, followed by a numerical solution, and since this solution is computationally expensive, an efficient approximate solution is given.

1.2.1 The rotating reference frame

The static magnetic field B_0 is large in comparison with the other variables and can lead to relatively large numerical (round-off) errors when evaluating the magnetization response. As explained in the previous section, the static magnetic field induces a clockwise rotation of the magnetization around the
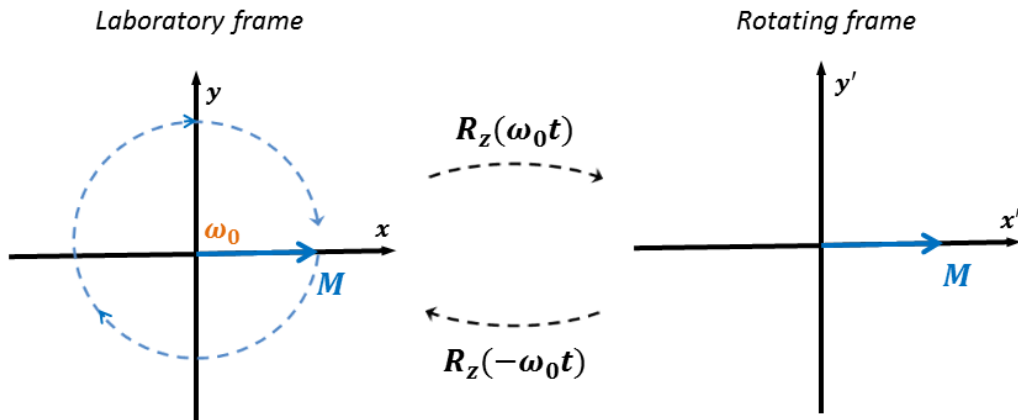
Figure 1.5: Magnetization vector in laboratory and rotating reference frame.

z-axis at the Larmor frequency $\omega_0 = \gamma\|B_0\|_2$. In a basis that rotates at the same frequency, the effect of B_0 on the magnetization is not perceived. A vector is rotated around the z-axis at the Larmor frequency using the following rotation matrix, calculated using equation (1.11):

$$R_z(\omega_0 t) = e^{\omega_0 \mathbf{S}_z t} = \begin{pmatrix} \cos(\omega_0 t) & -\sin(\omega_0 t) & 0 \\ \sin(\omega_0 t) & \cos(\omega_0 t) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \qquad (1.27)$$

Note that a positive rotation is counter-clockwise. A new basis is defined, denoted by primes, by applying R_z to the original basis:

$$\begin{pmatrix} \hat{x}' \\ \hat{y}' \\ \hat{z}' \end{pmatrix} := R_z(\omega_0 t) \begin{pmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{pmatrix} = \begin{pmatrix} \cos(\omega_0 t)\hat{x} - \sin(\omega_0 t)\hat{y} \\ \sin(\omega_0 t)\hat{x} + \cos(\omega_0 t)\hat{y} \\ \hat{z} \end{pmatrix}. \qquad (1.28)$$

The system in the original basis will be called the *laboratory reference frame*, in contrast to the *rotating reference frame* with basis defined in equation (1.28). A magnetization vector in the laboratory frame is expressed in the rotating reference frame:

$$M'(t) := R_z(\omega_0 t)M(t) = e^{\omega_0 \mathbf{S}_z t} M(t). \qquad (1.29)$$

By inverting the equation, a fixed vector M' in the rotating reference frame is shown to be rotating clockwise at the Larmor frequency in the laboratory frame, as shown in Figure 1.5. Hence a vector rotating at the Larmor frequency due to the B_0 field is fixed in the rotating reference frame. This can

also be seen by describing the dynamic of the magnetization in the rotating frame due to the Bloch equation:

$$
\begin{aligned}
\frac{dM'}{dt}(t) &= \frac{d}{dt}\left[e^{\omega_0 \mathbf{S}_z t} M(t)\right] \\
&= e^{\omega_0 \mathbf{S}_z t}\left[\mathbf{\Omega}(t) M(t) + M_0/T_1\right] + \omega_0 \mathbf{S}_z e^{\omega_0 \mathbf{S}_z t} M(t) \\
&= e^{\omega_0 \mathbf{S}_z t}\left[\mathbf{\Omega}(t) + \omega_0 \mathbf{S}_z\right] e^{-\omega_0 \mathbf{S}_z t} M'(t) + M_0/T_1.
\end{aligned}
\tag{1.30}
$$

The rotation matrix $e^{\omega_0 \mathbf{S}_z t}$ commutes with $\mathbf{S}_z$ and leaves M_0 invariant. Here the Bloch equation in the laboratory frame (1.19) is used.

Equation (1.30) can be simplified by decomposing the matrix $\mathbf{\Omega}(t)$:

$$
\mathbf{\Omega}(t) = \mathbf{D} - \gamma(B_x(t)\mathbf{S}_x + B_y(t)\mathbf{S}_y + B_z(t)\mathbf{S}_z),
\tag{1.31}
$$

with $\mathbf{D} := -\mathrm{diag}(\frac{1}{T_2}, \frac{1}{T_2}, \frac{1}{T_1})$. The rotation matrix $e^{\omega_0 \mathbf{S}_z t}$ commutes with both $\mathbf{D}$ and $\mathbf{S}_z$, but not with $\mathbf{S}_x$ and $\mathbf{S}_y$. It follows that:

$$
\begin{aligned}
\mathbf{\Omega}'(t) &:= e^{\omega_0 \mathbf{S}_z t}\left[\mathbf{\Omega}(t) + \omega_0 \mathbf{S}_z\right] e^{-\omega_0 \mathbf{S}_z t} \\
&= \mathbf{D} - \gamma e^{\omega_0 \mathbf{S}_z t}\left[B_x(t)\mathbf{S}_x + B_y(t)\mathbf{S}_y\right] e^{-\omega_0 \mathbf{S}_z t} + (\omega_0 - \gamma B_z(t))\mathbf{S}_z.
\end{aligned}
\tag{1.32}
$$

For the $\mathbf{S}_z$ component, ω_0 cancels against $\gamma\|B_0\|_2$, which annihilates the B_0. And the $\mathbf{S}_x$ and $\mathbf{S}_y$ components can be simplified:

$$
\begin{aligned}
&e^{\omega_0 \mathbf{S}_z t}\left[B_x(t)\mathbf{S}_x + B_y(t)\mathbf{S}_y\right] e^{-\omega_0 \mathbf{S}_z t} \\
&\quad = e^{\omega_0 \mathbf{S}_z t}\left[(-B_y(t), B_x(t), 0)^T e_3^T - e_3(-B_y(t), B_x(t), 0)\right] e^{-\omega_0 \mathbf{S}_z t} \\
&\quad = B_x'(t)\mathbf{S}_x + B_y'(t)\mathbf{S}_y,
\end{aligned}
\tag{1.33}
$$

where the following definition is used:

$$
\begin{pmatrix} B_x'(t) \\ B_y'(t) \end{pmatrix} := \begin{pmatrix} \cos(\omega_0 t) & -\sin(\omega_0 t) \\ \sin(\omega_0 t) & \cos(\omega_0 t) \end{pmatrix} \begin{pmatrix} B_x(t) \\ B_y(t) \end{pmatrix},
\tag{1.34}
$$

Identifying $\mathbb{R}^2$ with the complex plane, as in equation (1.25), allows us to express equation (1.34) in complex notation: $B_1'(t) := e^{i\omega_0 t} B_1(t)$.

Equation (1.33) allows equation (1.32) to be written in matrix form:

$$
\mathbf{\Omega}'(t) = \begin{pmatrix} -1/T_2 & \gamma(G(t)\cdot r + \Delta B_0) & -\gamma B_y'(t) \\ -\gamma(G(t)\cdot r + \Delta B_0) & -1/T_2 & \gamma B_x'(t) \\ \gamma B_y'(t) & -\gamma B_x'(t) & -1/T_1 \end{pmatrix}.
\tag{1.35}
$$
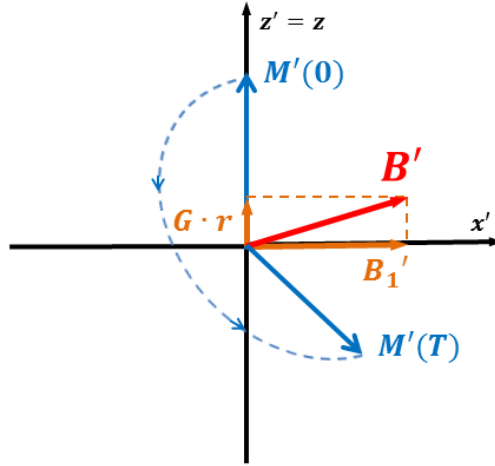
Figure 1.6: Magnetization response to radiofrequency and gradient waveforms in the rotating reference frame. View along the y'-axis. Static field inhomogeneities are neglected.

Substituting equations (1.32) into equation (1.30), the Bloch equation in the rotating frame is given by:

$$\frac{dM'}{dt}(t) = \mathbf{\Omega}'(t)M'(t) + M_0/T_1 \tag{1.36}$$

Note that the gradient field and the field inhomogeneity are in the $\hat{z}$-direction and therefore not affected by the change of basis.

In order to rotate the magnetization out of equilibrium using a radiofrequency pulse, it is most efficient to have a fixed rotation axis B_1' in the rotating reference frame. Inverting equation (1.34), it follows that a constant rotation axis can be achieved by giving the RF pulses the frequency ω_0. This case is called *on-resonance*.

The combined effect of a constant B_1' vector and the spatially varying gradient in the rotating reference frame is shown in Figure 1.6. Note that the view is along the y'-axis and that in the laboratory frame the illustrated magnetization trajectory is rotated around the z-axis at the Larmor frequency. Since it is always possible to move from one basis to the other, in the following the magnetization and its dynamics are modelled in the rotating frame, and to ease notation the apostrophes for this case will be dropped.

1.2.2 Discretized solution

Equation (1.36) is the Bloch equation in the rotating reference frame and in this section the equation is solved to give a numerical expression of the magnetization response. The duration of the radiofrequency pulses is given by T and the initial magnetization $M(0)$ is fixed. In most cases the initial magnetization $M(0)$ will equal the equilibrium magnetization M_0, but this is not a necessary condition. Equation (1.36) defines a system of differential equations which can be classified as inhomogeneous and linear with time-dependent coefficients.

A numerical solution requires the discretization of the given time-dependent RF and gradient waveforms. The time domain is discretized into N_t intervals of length Δt on which the time-dependent RF and gradient waveforms are piecewise constant, which means for $i \in \{0, 1, \ldots, N_t - 1\}$:

$$t_i = i\Delta t, \text{ with } \Delta t = \frac{T}{N_t}, \tag{1.37}$$

$$G(t) = G(t_i) \text{ for } t \in [t_i, t_{i+1}), \tag{1.38}$$

and, for $l \in \{1, 2, \ldots, L\}$:

$$b_1^{(l)}(t) = b_1^{(l)}(t_i) \text{ for } t \in [t_i, t_{i+1}) \tag{1.39}$$

It follows from equation (1.35) that in this case $\mathbf{\Omega}(t) = \mathbf{\Omega}(t_i)$ for $t \in [t_i, t_{i+1})$. Differential equation (1.36) becomes:

$$\dot{M}(t) = \mathbf{\Omega}(t_i)M(t) + M_0/T_1 \text{ for } t \in [t_i, t_{i+1}) \tag{1.40}$$

Now that the differential equation has constant coefficients it is solvable on each interval if the matrices $\mathbf{\Omega}(t_i)$ are assumed invertible. The solution for the homogeneous part is given by:

$$M(t) = e^{\mathbf{\Omega}(t_i)(t-t_i)}M(t_i) \text{ for } t \in [t_i, t_{i+1}) \tag{1.41}$$

The magnetization is required to have a continuous response, thus the end points of the intervals are connected:

$$M(t_{i+1}) = e^{\mathbf{\Omega}(t_i)\Delta t}M(t_i), \text{ for } i \in \{0, 1, 2, \ldots, N_t - 1\} \tag{1.42}$$

This defines the homogeneous solution for all $t \in [0, T]$. To solve for the inhomogeneous part, equation (1.40) is multiplied with $e^{-\mathbf{\Omega}(t_i)(t-t_i)}$ and some terms are shifted:

$$
\begin{aligned}
e^{-\mathbf{\Omega}(t-t_i)}M_0/T_1 &= e^{-\mathbf{\Omega}(t_i)(t-t_i)}\dot{M}(t) - e^{-\mathbf{\Omega}(t_i)(t-t_i)}\mathbf{\Omega}(t_i)M(t) \\
&= \frac{d}{dt}\left[e^{-\mathbf{\Omega}(t_i)(t-t_i)}M(t)\right] \quad \text{ for } t \in [t_i, t_{i+1})
\end{aligned}
\tag{1.43}
$$

Switching and integrating both sides and noting that initial value $M(t_i)$ is given:

$$e^{-\mathbf{\Omega}(t_i)(t-t_i)}M(t) = M(t_i) + \left[\int_{t_i}^{t} e^{-\mathbf{\Omega}(t_i)(\tau-t_i)}d\tau\right] M_0/T_1 \qquad (1.44)$$

Giving the solution of the inhomogeneous equation:

$$
\begin{aligned}
M(t) &= e^{\mathbf{\Omega}(t_i)(t-t_i)}M(t_i) + e^{\mathbf{\Omega}(t_i)(t-t_i)}\left[\int_{t_i}^{t} e^{-\mathbf{\Omega}(t_i)(\tau-t_i)}d\tau\right] M_0/T_1 \\
&= e^{\mathbf{\Omega}(t_i)(t-t_i)}M(t_i) + \left[e^{\mathbf{\Omega}(t_i)(t-t_i)} - \mathbf{I}\right]\mathbf{\Omega}^{-1}(t_i)M_0/T_1
\end{aligned}
\qquad (1.45)
$$

And for the grid points:

$$M(t_{i+1}) = e^{\mathbf{\Omega}(t_i)\Delta t}M(t_i) + \left[e^{\mathbf{\Omega}(t_i)\Delta t} - \mathbf{I}\right]\mathbf{\Omega}^{-1}(t_i)M_0/T_1 \qquad (1.46)$$

1.2.3 Approximate solution

To determine the response of the whole object, equation (1.46) has to be solved for each voxel and time point. In the following, an approximation of equation (1.46) is made with an accuracy of $\mathcal{O}(\Delta t^2)$, which does not require the inversion or exponent of the matrix $\mathbf{\Omega}(t_i)$.

Homogeneous part

Calculating the matrix exponential of a general matrix can be costly, but fortunately $\mathbf{\Omega}(t_i)$ has some structure. The matrix can be split into the relaxation effects $\mathbf{D}$ on the diagonal, and the magnetic field effects $\mathbf{B}_i$:

$$\mathbf{\Omega}(t_i) = \mathbf{D} - \gamma\mathbf{B}_i, \qquad (1.47)$$

with:

$$
\begin{aligned}
\mathbf{D} &:= \begin{pmatrix} -1/T_2 & 0 & 0 \\ 0 & -1/T_2 & 0 \\ 0 & 0 & -1/T_1 \end{pmatrix}, \\
\mathbf{B}_i &:= \begin{pmatrix} 0 & -B_z(t_i) & B_y(t_i) \\ B_z(t_i) & 0 & -B_x(t_i) \\ -B_y(t_i) & B_x(t_i) & 0 \end{pmatrix}.
\end{aligned}
\qquad (1.48)
$$

The exponents of the matrices in (1.48) are easily calculated. The exponent of a diagonal matrix is calculated element-wise:

$$e^{\mathbf{D}\Delta t} = \begin{pmatrix} e^{-\Delta t/T_2} & 0 & 0 \\ 0 & e^{-\Delta t/T_2} & 0 \\ 0 & 0 & e^{-\Delta t/T_1} \end{pmatrix}. \qquad (1.49)$$

21

For the skew-symmetric matrix $\mathbf{B}_i$, the exponent is given by Rodrigues' rotation formula in equation (1.11). A more efficient method for determining the rotation matrix $e^{-\gamma \mathbf{B}_i \Delta t}$, is by describing the rotation in the so-called *spin-domain*. Assuming there is no relaxation, the Bloch equation can be described in the spin-domain using the so-called *Cayley-Klein* parameters. A rotation around unit vector $\frac{B}{\|B\|_2}$ with angular velocity $\omega_j = -\gamma \|B\|_2 \Delta t$, is described by the complex parameters:

$$\alpha = \cos(\omega_j/2) - i\frac{B_z}{\|B\|_2}\sin(\omega_j/2), \quad \beta = -i\left(\frac{B_x}{\|B\|_2} - i\frac{B_y}{\|B\|_2}\right)\sin(\omega_j/2).$$

$$(1.50)$$

The rotation matrix $e^{-\gamma \mathbf{B}_i \Delta t}$ can be expressed in α and β, see [10] and [17]:

$$e^{-\gamma \mathbf{B}_i \Delta t} = \begin{pmatrix} \frac{1}{2}(\alpha^2 + \bar{\alpha}^2 - \beta^2 - \bar{\beta}^2) & -\frac{i}{2}(\alpha^2 - \bar{\alpha}^2 - \beta^2 + \bar{\beta}^2) & -(\alpha\beta + \bar{\alpha}\bar{\beta}) \\ \frac{i}{2}(\alpha^2 - \bar{\alpha}^2 + \beta^2 - \bar{\beta}^2) & \frac{1}{2}(\alpha^2 + \bar{\alpha}^2 + \beta^2 + \bar{\beta}^2) & -i(\bar{\alpha}\beta + \alpha\bar{\beta}) \\ -\alpha\beta - \bar{\alpha}\bar{\beta} & i(\alpha\beta - \bar{\alpha}\bar{\beta}) & \alpha\bar{\alpha} - \beta\bar{\beta} \end{pmatrix}.$$

$$(1.51)$$

Here $\bar{\alpha}$ and $\bar{\beta}$ denote the complex conjugates. The spin-domain version of the rotation matrix has an advantage over the Rodrigues' rotation formula in equation (1.11): when relaxation effects are ignored, the Bloch equations can expressed in α and β:

$$\frac{d}{dt}\begin{pmatrix} \beta \\ \bar{\alpha} \end{pmatrix} = \frac{i\gamma}{2}\begin{pmatrix} G(t) \cdot r & \bar{B}_1(r,t) \\ B_1(r,t) & -G(t) \cdot r \end{pmatrix}\begin{pmatrix} \beta \\ \bar{\alpha} \end{pmatrix}.$$

$$(1.52)$$

This spin-domain version of the Bloch equation will be useful for the design of radiofrequency pulses.

The exponent of $\mathbf{\Omega}(t_i)$ is not equal to the product of the exponent of $\mathbf{D}$ and $\mathbf{B}_i$, since these matrices do not commute. For square matrices $\mathbf{X}$ and $\mathbf{Y}$ the Zassenhaus formula [6] is used:

$$e^{(\mathbf{X}+\mathbf{Y})\Delta t} = e^{\mathbf{X}\Delta t}e^{\mathbf{Y}\Delta t}\prod_{k=2}^{\infty}e^{C_k(\mathbf{X},\mathbf{Y})\Delta t^k}.$$

$$(1.53)$$

Here C_k is a polynomial in nested commutators of order k of $\mathbf{X}$ and $\mathbf{Y}$. For $k = 2,3$, the polynomial C_k is calculated explicitly in [6] and equals:

$$C_2(\mathbf{X}, \mathbf{Y}) = -\frac{1}{2}[\mathbf{X}, \mathbf{Y}] = -\frac{1}{2}(\mathbf{XY} - \mathbf{YX}),$$

$$C_3(\mathbf{X}, \mathbf{Y}) = \frac{1}{3}[\mathbf{Y}, [\mathbf{X}, \mathbf{Y}]] + \frac{1}{6}[\mathbf{X}, [\mathbf{X}, \mathbf{Y}]].$$

$$(1.54)$$

See [6] for proof and how to calculate C_k for higher k. By the definition of the matrix exponential:

$$\prod_{k=2}^{\infty} e^{C_k(\mathbf{X},\mathbf{Y})\Delta t^k} = \prod_{k=2}^{\infty} \left[\mathbf{I} + C_k(\mathbf{X},\mathbf{Y})\Delta t^k + \mathcal{O}(\Delta t^{2k})\right]$$
$$= \mathbf{I} + C_2(\mathbf{X},\mathbf{Y})\Delta t^2 + C_3(\mathbf{X},\mathbf{Y})\Delta t^3 + \mathcal{O}(\Delta t^4). \tag{1.55}$$

Splitting the exponential $e^{\mathbf{\Omega}(t_i)\Delta t}$ in (1.46) using the Zassenhaus formula leads to a first-order approximation in Δt of the solution:

$$e^{\mathbf{\Omega}(t_i)\Delta t} = e^{(\mathbf{D} - \gamma \mathbf{B}_i)\Delta t} = e^{\mathbf{D}\Delta t}e^{-\gamma \mathbf{B}_i \Delta t} + \mathcal{O}(\Delta t^2). \tag{1.56}$$

In the following this first-order approximation will be used. Note that taking more terms in (1.55) leads to a higher-order approximation of the exact solution. Furthermore, when there is no RF power, $B_1 = 0$, the matrices $\mathbf{D}$ and $\mathbf{B}_i$ commute and the higher order terms in (1.58) are zero, in this case the response of the magnetization can be exactly determined for arbitrary large time steps.

Inhomogeneous part

Similar to the splitting of the exponent, the calculation of the inhomogeneous part can be simplified by making a first-order approximation in Δt:

$$\left[e^{\mathbf{\Omega}(t_i)\Delta t} - \mathbf{I}\right]\mathbf{\Omega}^{-1}(t_i)M_0/T_1 = \left[\sum_{k=1}^{\infty}(\mathbf{D} - \gamma \mathbf{B}_i)^{k-1}\frac{\Delta t^k}{k!}\right](-\mathbf{D})M_0$$
$$= \left[\mathbf{I} - e^{\mathbf{D}\Delta t}\right]M_0 + \mathcal{O}(\Delta t^2). \tag{1.57}$$

Note that $M_0/T_1 = -\mathbf{D}M_0$. Taking more terms in the sum in (1.57) leads to a higher-order approximation of the exact solution.

Complexity

The magnetization response to a radiofrequency pulse given in equation (1.46) is approximated by using equations (1.55) and (1.57) and ignoring the $\mathcal{O}(\Delta t^2)$ terms:

$$M(t_{i+1}) \approx e^{\mathbf{D}\Delta t}e^{-\gamma \mathbf{B}_i \Delta t}M(t_i) + \left[\mathbf{I} - e^{\mathbf{D}\Delta t}\right]M_0. \tag{1.58}$$

Most of the work is contained in calculating the rotation matrix. Using the Cayley-Klein parameters from equations (1.50) and (1.51), calculating the

rotation matrix and multiplying it with M takes 45 flops, while the relaxation effects can be done in 5 flops. Since equation (1.58) has to be solved for each voxel and time interval, the total complexity of the response calculation equals $50 N_t N_s$ flops.

1.3   Radiofrequency pulse design

In the previous section the magnetization response was calculated for given radiofrequency and gradient waveforms. In the following this line is reversed: find the radiofrequency pulse that transfers the magnetization from a given initial state to a desired state. The difference between the initial and desired state is typically described by the *flip angle θ* which equals the angle in radians between both states around the x-axis. Note that this does not mean that the rotation axis has to be fixed during the radiofrequency pulse. Since the radiofrequency and gradient waveforms are time dependent, the same is true for the rotation axis, see Figure 1.6. Furthermore, allowing the rotation axis to be free over time makes it possible to have spatially selective pulses: the same radiofrequency pulse has different effects for different voxels through the gradients and field inhomogeneities.

Radiofrequency pulses are grouped based on their flip angle and initial state. There are three types, see also Figure 1.7:

Excitation pulse Rotates the magnetization from the equilibrium state to a state with transverse magnetization, usually with $\theta \in [0, \frac{\pi}{2}]$ rad. Since MR signal is based on the rotating transversal magnetization, every MR method requires the use of an excitation pulse.

Inversion pulse Rotates the magnetization from the equilibrium state π rad such that the longitudinal magnetization is inverted and there is no transverse magnetization. These pulses are used to enhance the difference in T_1 relaxation rate between tissues.

Refocusing pulse Transfers the magnetization from an out-of-equilibrium state π rad around the x-axis. Used to cancel accumulated phase differences due to the T_2' effect.

The MRI process can in general be split into two parts: the *excitation* where radiofrequency pulses are used to create transverse magnetization, and the *acquisition* where the transverse magnetization is measured in order to generate an image.
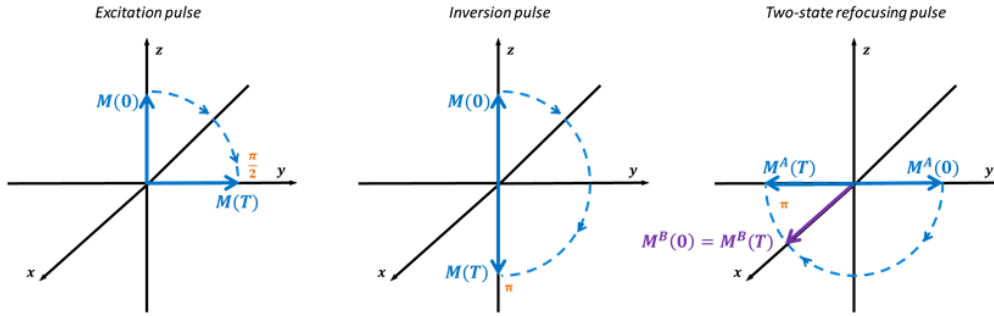
Figure 1.7: Examples of the three types of radiofrequency pulses.

1.3.1 Excitation

In discussing the magnetization excitation, the focus is changed from a single voxel and magnetization vector to a two-dimensional spatial domain or *slice* in the xy-plane. In practice, focusing on such a single slice is possible if the scanned object is small in this direction, or by using a so-called *slice-selecting gradient.*
The slice-selecting gradient gives a linearly variant rotation frequency along the z-axis, such that there is a certain slice approximately on-resonance with the RF pulse and hence excited. For more details see [11] or [4].
Within the slice, we focus on *spatially-selective* pulses which excites only a subset of the slice. This is useful since the MR signal will be based only on the magnetization in this region.

1.3.2 Acquisition

As mentioned before, the MR signal is generated by the rotating transverse magnetization. Proton density and T1/T2 relaxation rates determine the dynamics of the transverse magnetization. MR methods are designed to show the difference between voxels of these parameters. This allows the generation of contrast based on proton density or on different relaxation rates. Figure 1.8 shows examples of different types of contrast of the same slice of a human brain. Gradients are used during signal readout to determine signal strength at different positions.

1.3.3 Standard methods

Two often used methods for spatially-selective pulse design are small-tip-angle (STA) and the linear class large-tip-angle (LCLTA) method, which
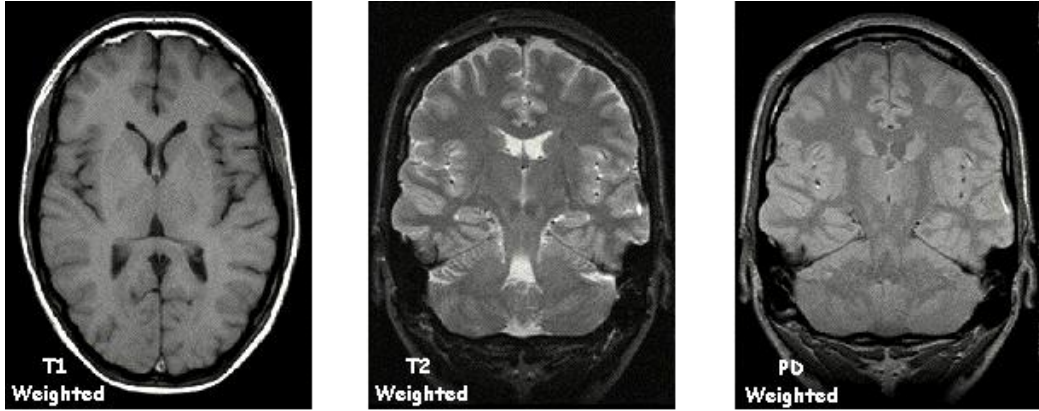
Figure 1.8: Different contrast MR images of a slice of a human brain.

were introduced by Pauly et al. in respectively [16] and [17]. Both methods approximate the magnetization response by a function linear in B_1, such that the desired RF pulse can found by solving a linear system. Note that the Bloch equation is not a linear but a bilinear system: it is linear in B_1 if M is fixed and vice versa.

Both methods make use of the following assumption:

- *The relaxation times are supposed to be much larger than the pulse duration: $T_1, T_2 \gg T$.*

In this case the relaxation effects are small and can be removed by setting the diagonal to zero. This is applicable in most cases, and by the discussion of the previous section makes the response to an RF pulse a rotation.

The two methods make assumptions in order to approximate the magnetization response as a linear function of B_1. For the STA method:

1. *The flip angle θ is small, such that $\sin(\theta) \approx \theta$,*

2. *The initial magnetization equals the equilibrium state: $M(0) = M_0$.*

Under these assumptions the longitudinal magnetization is approximately constant during the pulse, and hence bilinear terms with M_z can be approximated using $M_z = M_0$. In practice, this approach is sufficiently accurate for excitation pulses with a flip angle θ of up to $\frac{\pi}{2}$ rad. But for large-tip-angle pulses, such as the inversion pulse with $\theta = \pi$, the error is significant due to the non-linear effects. Furthermore, the method can not be applied to the refocusing pulse since the initial magnetization is not the equilibrium. This is why in a follow-up article Pauly et al. introduced the linear class

large-tip-angle (LCLTA) method [17] which generalizes the STA method by allowing starting magnetization out of equilibrium.

For the LCLTA method, the rotation of the magnetization is described in the spin-domain using the Cayley-Klein parameters α and β given in equations (1.50) and (1.51). The spin-domain version of the Bloch equation, shown in (1.52), is a coupled differential equation in the complex parameters α and β. It is solved by decoupling α using the following assumption:

The RF pulse can be decomposed into subpulses, such that:

1. *The effect of each subpulse on the excitation is small,*

2. *During each subpulse the gradient is inherently refocused: its integral equals the zero vector.*

The first condition leads to $B_1(r,t)\beta \approx 0$, and setting this term to zero in equation (1.52) decouples α. The second condition ensures that each subpulse rotates the magnetization around the same axis, see [22]. Under these assumptions, the spin-domain version of the Bloch equation (1.52) is simplified and the flip angle θ of the magnetization due to the effective B_1 radiofrequency pulse is given by:

$$\theta(r) = \gamma \int_0^T \bar{B}_1(t) e^{-i\gamma \Delta B_0(r)(t-T)} e^{ir \cdot k(t)} dt, \qquad (1.59)$$

where $\bar{B}_1$ denotes the complex conjugate of B_1 and:

$$k(t) = -\gamma \int_t^T G(s) ds. \qquad (1.60)$$

Here $k(t)$ defines the so-called *k-space trajectory* with unit cm^{-1}, and is used to invert equation (1.59) using a Fourier transform. Suppose the desired flip angle at each position is defined by the function θ_D for each position. Inverting equation (1.59) works if the Fourier transform of θ_D is sampled densely at positions with significant power by the trajectory of $k(t)$. Requiring the gradient to be inherently refocused is equivalent to demanding $k = 0$ at the end of each subpulse.

The LCLTA method was initially introduced for the case of one radiofrequency transmit coil, and Xu et al. extended the method to the multichannel case [22]. Setting the effective B_1 equal to the sum of the L radiofrequency

pulses of the different coils, $b_1^{(l)}$ with $1 \leq l \leq L$, weighted by their transmit sensitivities s_l as in equation (1.26), equation (1.59) becomes:

$$\theta(r) = \gamma \sum_{l=1}^{L} \bar{s}_l(r) \int_0^T \bar{b}_1^{(l)}(t) e^{-i\gamma \Delta B_0(r)(t-T)} e^{ir \cdot k(t)} dt. \tag{1.61}$$

The Fourier transform is no longer applicable to this case, and the equation is solved for the RF pulses, $b_1^{(l)}$, by discretizing the integral. In this case equation (1.61) becomes:

$$\begin{aligned} \theta(r) &= \gamma \Delta t \sum_{l=1}^{L} \bar{s}_l(r) \sum_{i=1}^{N_t} \bar{b}_1^{(l)}(t_i) e^{-i\gamma \Delta B_0(r)(t_i-T)} e^{ir \cdot k(t_i)} \\ &= \sum_{l=1}^{L} S^{(l)}(r) \cdot b^{(l)}, \end{aligned} \tag{1.62}$$

with $S^{(l)}(r), b^{(l)} \in \mathbb{C}^{N_t}$, defined by:

$$S^{(l)}(r) := \gamma \Delta t \begin{pmatrix} e^{-i\gamma \Delta B_0(r)(t_1-T)} e^{ir \cdot k(t_1)} \\ e^{-i\gamma \Delta B_0(r)(t_2-T)} e^{ir \cdot k(t_2)} \\ \vdots \\ e^{-i\gamma \Delta B_0(r)(t_{N_t}-T)} e^{ir \cdot k(t_{N_t})} \end{pmatrix}, \text{ and } b^{(l)} := \begin{pmatrix} \bar{b}_1^{(l)}(t_1) \\ \bar{b}_1^{(l)}(t_2) \\ \vdots \\ \bar{b}_1^{(l)}(t_{N_t}) \end{pmatrix}.$$

Furthermore, since the equation has to be correct for a number of spatial positions r_i, with $i \in 1, 2, \ldots, N_s$, we get a system of equations:

$$\begin{pmatrix} \theta(r_1) \\ \theta(r_2) \\ \vdots \\ \theta(r_{N_s}) \end{pmatrix} = \sum_{l=1}^{L} \begin{pmatrix} (S^{(l)}(r_1))^T \\ (S^{(l)}(r_2))^T \\ \vdots \\ (S^{(l)}(r_{N_s}))^T \end{pmatrix} b^{(l)}, \tag{1.63}$$

which is equivalent to:

$$\begin{pmatrix} \theta(r_1) \\ \theta(r_2) \\ \vdots \\ \theta(r_{N_s}) \end{pmatrix} = \begin{pmatrix} (S^{(1)}(r_1))^T & (S^{(2)}(r_1))^T & \cdots & (S^{(L)}(r_1))^T \\ (S^{(l)}(r_2))^T & (S^{(2)}(r_2))^T & \cdots & (S^{(L)}(r_2))^T \\ \vdots & \vdots & \ddots & \vdots \\ (S^{(l)}(r_{N_s}))^T & (S^{(2)}(r_{N_s}))^T & \cdots & (S^{(L)}(r_{N_s}))^T \end{pmatrix} \begin{pmatrix} b^{(1)} \\ b^{(2)} \\ \vdots \\ b^{(L)} \end{pmatrix}.$$

This can be written as:

$$\theta = \mathbf{Sb}, \tag{1.64}$$

where $\theta \in \mathbb{R}^{N_s}$, $\mathbf{S} \in \mathbb{C}^{N_s \times N_t}$, and $\mathbf{b} \in \mathbb{C}^{N_t}$. In practice, the system (1.64) is often overdetermined ($N_t < N_s$) and there is no exact solution. Instead, a radiofrequency pulse $\mathbf{b}$ is calculated by solving the system in the least squares sense, minimizing the residual:

$$\mathbf{b} = \arg\min_{\mathbf{x}} \left(\|\mathbf{Sx} - \theta\|_2^2 \right). \tag{1.65}$$

Since the matrix $\mathbf{S}$ is typically ill-conditioned and overdetermined, a regularization term λ is added to prevent large noise components in $\mathbf{b}$:

$$\mathbf{b} = \arg\min_{\mathbf{x}} \left(\|\mathbf{Sx} - \theta\|_2^2 + \lambda\|\mathbf{x}\|_2^2 \right). \tag{1.66}$$

This regularization also reduces the RF amplitude and therefore heating. In order to find an appropriate value for λ, equation (1.66) has to be solved for different values, where methods such as conjugate gradients for the least squares problem (CGLS) can be used. Plotting for different λ, with solution $\mathbf{b}_\lambda$, the RF amplitude, $\|\mathbf{b}_\lambda\|_2$, against the residual norm, $\|\mathbf{Sb}_\lambda - \theta\|_2$, typically leads to an L-curve as shown in Figure 1.9. The best λ value for the regularization parameter usually is in the corner. Compared with this value for λ, a small increase or decrease leads to a large increase in respectively the residual norm or the solution norm. A fast method for computing the solution $\mathbf{b}$ for different regularization parameters is by using the multi shift CGLS algorithm as described in [21].
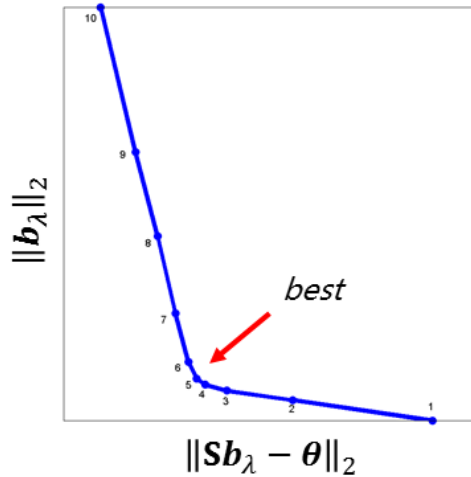
Figure 1.9: L-curve for the conjugate gradients least squares problem in equation (1.66).

1.4 Numerical tests

In the following, the LCLTA method is tested for spatially-selective excitation, inversion and refocusing pulses. The spatially dependent parameters are defined on a 128×128 grid. For the inhomogeneity maps the spherical phantom depicted in Figure 1.4 is used. The two B_1^+ maps are defined on a square with edges of 12 cm, giving a voxel size of just less then 1 mm^2.
Since we are only interested in the response within the phantom, a disk-shaped mask with a radius of 4.5 cm is placed over the relevant part of the phantom, and voxels outside of the phantom are ignored. The mask contains 7120 voxels for which the magnetization response will be considered.

The gradients follow a inherently refocused outward spiral as shown in Figure 1.10. Without proof we state that this trajectory satisfies the requirements of the LCLTA method given in the previous section. The associated gradient waveforms are bounded by $G_{max} = 30$ mT/m and the slew rates of the gradients by $S_{max} = 140$ mT/m/ms. Both are shown in Figure 1.11. The duration of the pulse is determined by the minimum time the k-space trajectory can be completed under these constraints.

The spatially-selective desired flip maps θ_D were determined using the following function:

$$\theta_D(r) = \mu \left(\frac{1}{e^{\frac{\rho - \alpha}{\beta}}} - \frac{1}{e^{\frac{\rho + \alpha}{\beta}}} \right). \tag{1.67}$$

with:

$$\rho = \sqrt{(r_x/a)^2 + (r_y/b)^2}$$

Here a and b determine the elliptic shape of the profile, α and β determine respectively the width and steepness of the transition region, and μ scales the flip angle. Equation (1.67) was used for the creation of two flip maps: one for a $\frac{\pi}{2}$ rad rotation used for the excitation pulse, and one for π rad rotation for both the inversion and refocusing pulses. Both maps are shown in Figure 1.12. The parameters for these maps were: $a, b = 3$, $\alpha = 1$, $\beta = 0.05$ and μ was equal to respectively $\frac{\pi}{2}$ and π.

For given initial magnetization profile $M(0)$ and flip map θ_D, the desired magnetization profile D is given by:

$$D(r) = \mathbf{R}_x(\theta_D(r))M(r,0). \tag{1.68}$$
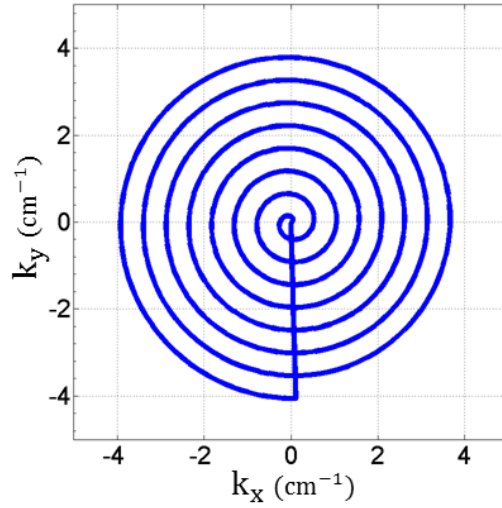
Figure 1.10: Inherently refocused outward spiral k-space trajectory with $k_{max} = 4$.
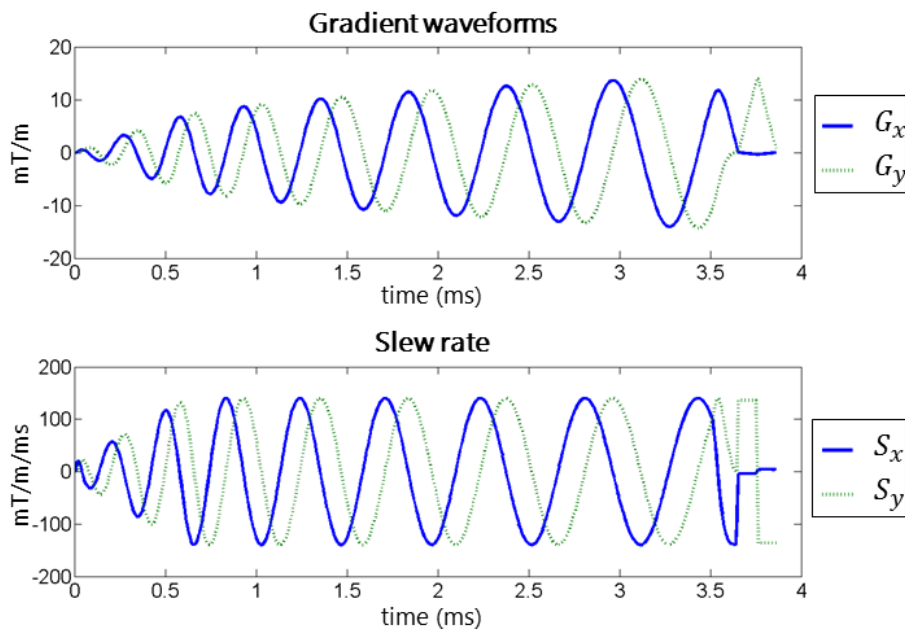
Figure 1.11: The gradient waveform and slew rate of the k-space trajectory shown in Figure 1.10.
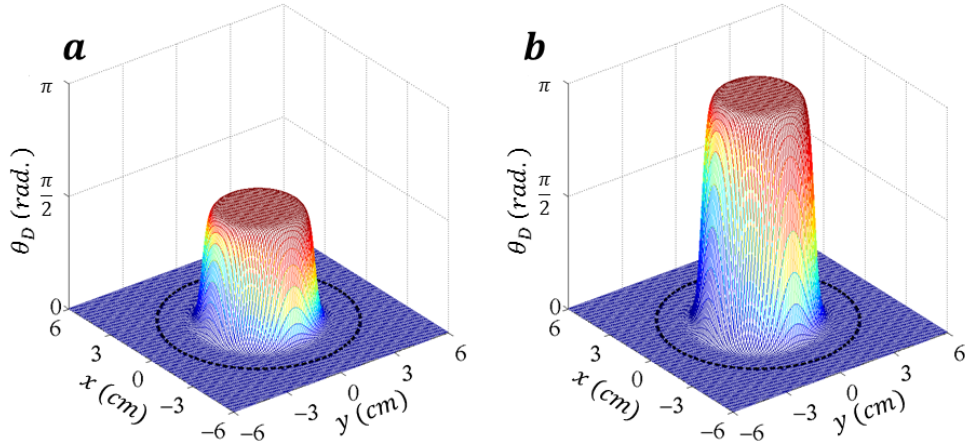
Figure 1.12: Flip maps for spatially selective $\frac{\pi}{2}$ (**a**) and π (**b**) rad rotation. Dashed black line indicates the border of the mask.

with:

$$\mathbf{R}_x(\theta_D(r)) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_D(r)) & \sin(\theta_D(r)) \\ 0 & -\sin(\theta_D(r)) & \cos(\theta_D(r)) \end{pmatrix} \tag{1.69}$$

Note that the LCLTA method does not take the initial magnetization state into account and will create the same radiofrequency pulse for both the inversion and refocusing case.

The results are evaluated based on the relative error of the magnetization response at time T compared with the desired magnetization profile D:

$$\varphi(M(T)) := \frac{\|M(T) - D\|_2^2}{\|D\|_2^2}. \tag{1.70}$$

Here all voxels are concatenated to express the whole magnetization profile:

$$\begin{aligned} M(t) &:= \left[M(r_1, t)^T, M(r_2, t)^T, \ldots, M(r_{N_s}, t)^T \right]^T, \\ D &:= \left[D(r_1)^T, D(r_2)^T, \ldots, D(r_{N_s})^T \right]^T. \end{aligned} \tag{1.71}$$

Furthermore, the radiofrequency amplitude of both channels are shown since high values lead to heating.

1.4.1 Excitation pulse

The desired magnetization profile D for the $\frac{\pi}{2}$ rad excitation pulse is created by applying flip map **a** from Figure 1.12 to M_0 using equation (1.68). The result is shown on the right in Figure 1.13. On the left is the magnetization at the end of the radiofrequency pulse designed using the LCLTA method. The figure seems to indicate that the LCLTA method performs well for this type of pulse. The relative error, defined in equation (1.70), equals 0.091. Figure 1.14 shows the RF amplitude of the two channels.

1.4.2 Inversion pulse

Figure 1.15 shows on the right the desired magnetization profile for the inversion pulse which was calculated by applying flip map **b** from Figure 1.12 to M_0. Note that in the flip map there is a smooth transition between π rad in the center and 0 rad at the edge of the mask. As a consequence, there are values $\theta(r) \in]0, \pi[$ in the transition region, leading to the ring of transversal magnetization as shown in Figure 1.15. The magnetization response of the inversion pulse calculated using the LCLTA method is shown on the left. There is still significant transverse magnetization in the center of the mask.

Figure 1.16 shows the RF amplitudes of the different channels. The calculated pulse is also used as the refocusing case in the following section, since the LCLTA method is based on the flip map (Figure 1.12) and not the initial magnetization. The RF amplitude profile looks similar to the profile of the excitation pulse in Figure 1.14, but with double the amplitude. The reason for this can be found in equation (1.66) which defines the LCLTA solution. The only difference between the excitation and the inversion/refocusing case is that the value of θ in the second case doubles. As a consequence of the linear system in equation (1.66), the calculated pulse amplitude also doubles.
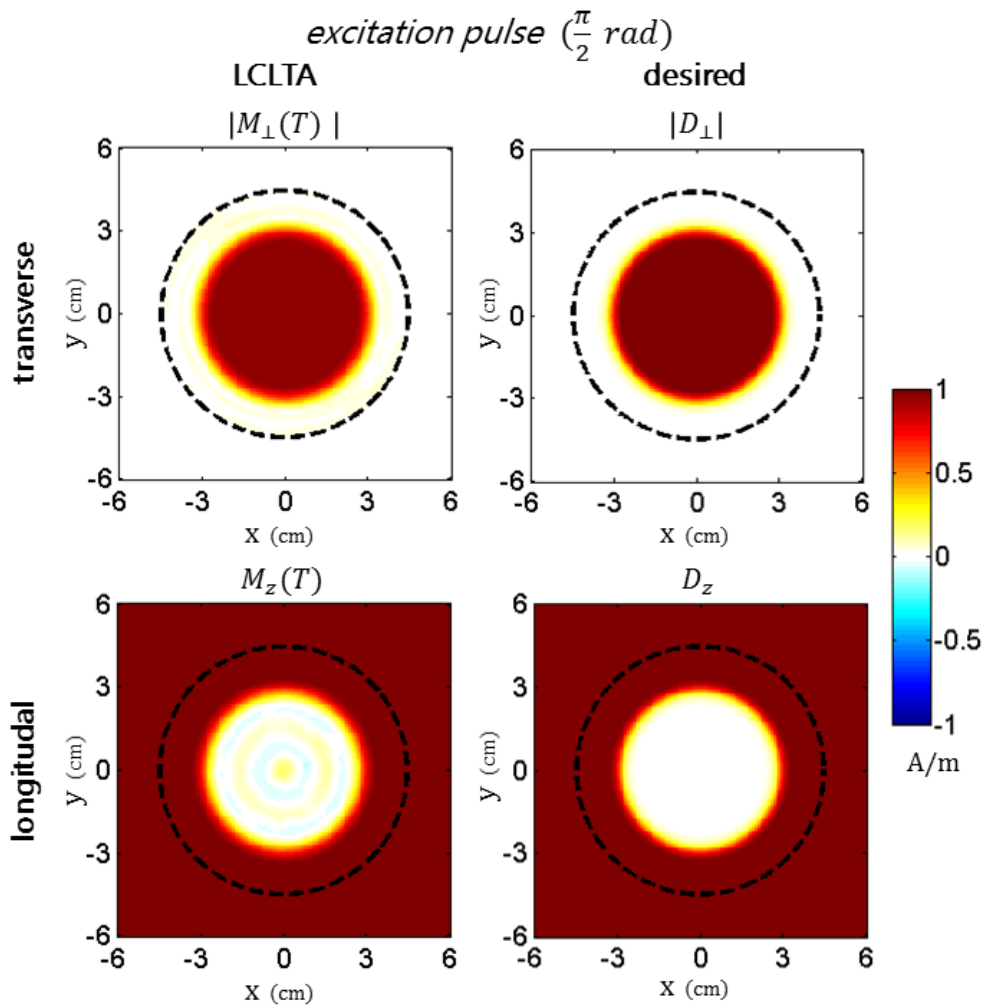
Figure 1.13: The transverse and longitudinal magnetization response at time T of the LCLTA excitation pulse compared with the desired magnetization.
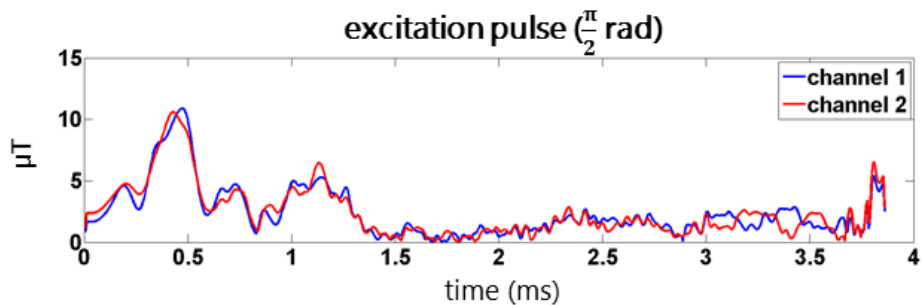
Figure 1.14: RF amplitude of both channels for the $\frac{\pi}{2}$ rad excitation pulse calculated using the LCLTA method.
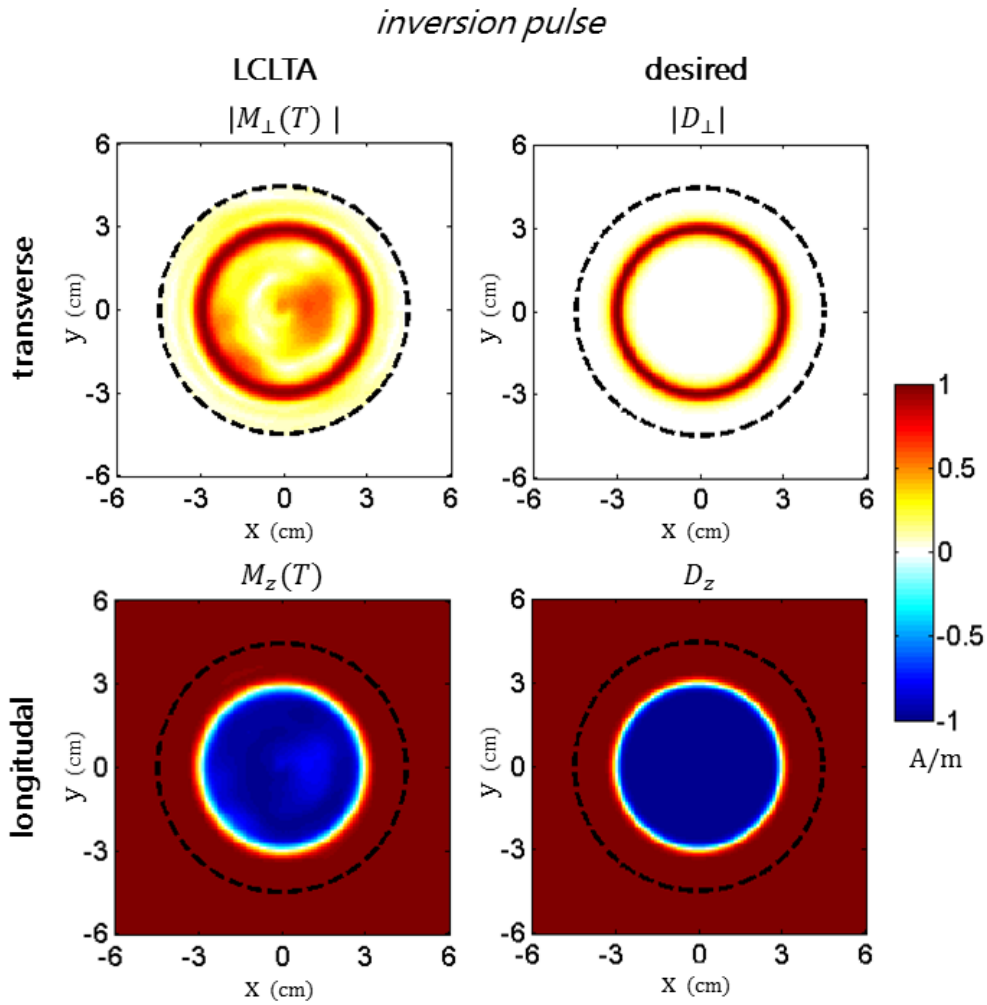
Figure 1.15: The transverse and longitudinal magnetization response at time T of the LCLTA inversion pulse compared with the desired magnetization.
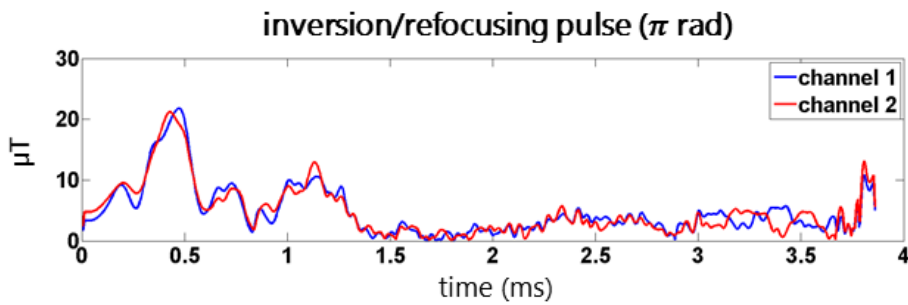
Figure 1.16: RF amplitude of both channels for the π rad inversion and refocusing pulse calculated using the LCLTA method.

1.4.3   Two-state refocusing pulse

As mentioned in the previous section, the refocusing pulse of the LCLTA method is the same as the inversion pulse and the RF amplitude is shown in Figure 1.16. The relative error of the magnetization in the refocusing case is tested in this section. Since the goal of the refocusing pulse is to rotate any vector in the transverse plane π rad around the x-axis, the pulse is tested for two different initial states for each voxel:

1. An initial state $M^A(0) := \hat{y}$, that at the end of the pulse should be rotated to $M^A(T) = -\hat{y}$.

2. An initial state $M^B(0) := \hat{x}$, that should be left fixed by the pulse: $M^B(T) = \hat{x}$.

Both effects are illustrated in Figure 1.7. Note that since the two initial states span the whole xy-plane, good refocusing of both states leads to good refocusing of any vector in the plane.
For both cases the magnetization vectors start and end within the transverse plane. To distinguish the different initial and end states the phase of the magnetization within the transverse plane will be evaluated apart from the transversal and longitudinal magnetization. The phase is relative to the $\hat{y}$ direction. In practice, for a refocusing pulse to be useful it is crucial that the phase is correct, such that all magnetization vectors were effectively rotated around the same axis.

The results for respectively the first and second case are shown in Figure 1.17, and Figure 1.18. Most noteworthy are the large errors in the phase of the magnetization in the transverse plane for both cases.
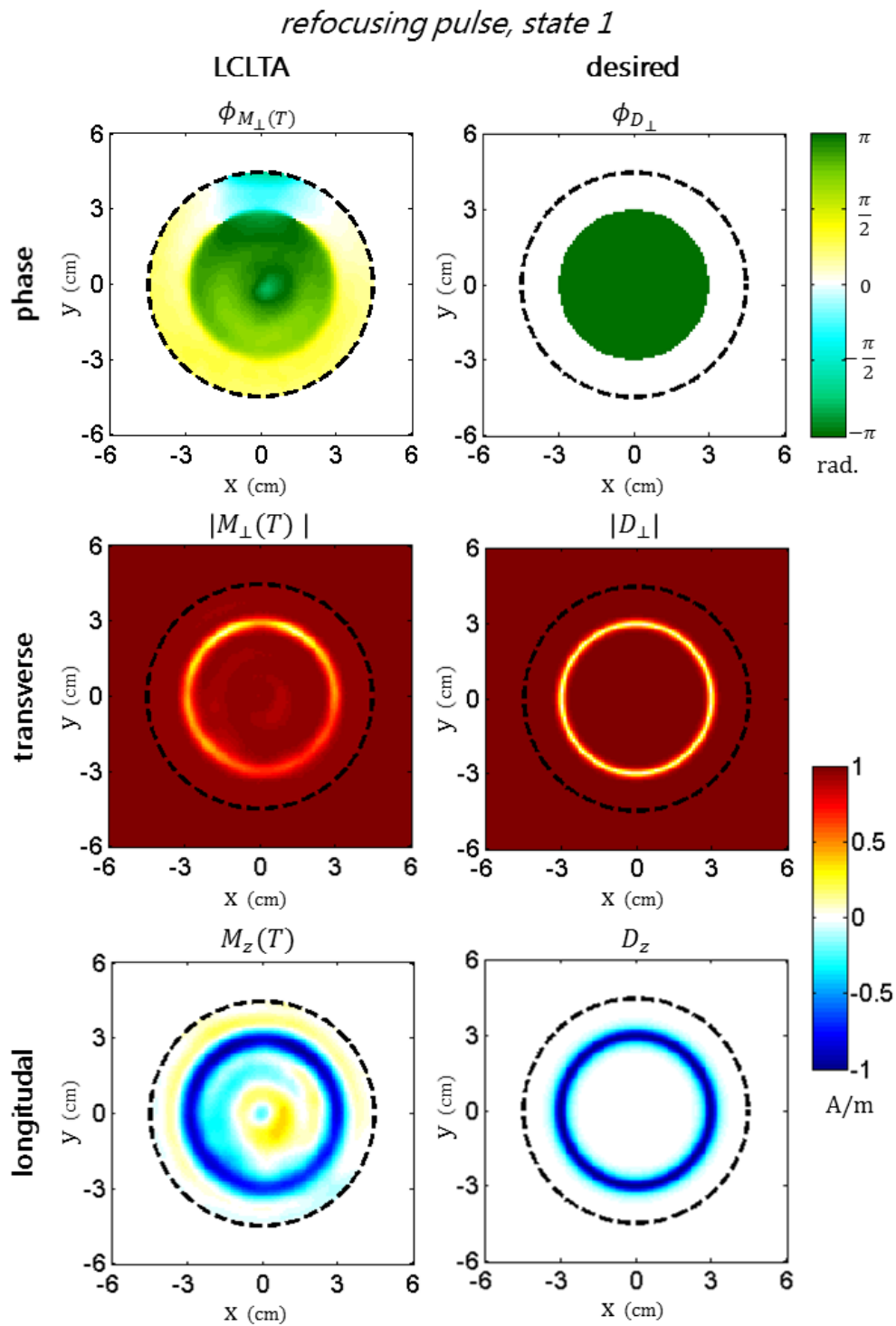
Figure 1.17: Comparison of the magnetization after the LCLTA refocusing pulse with the desired magnetization, for initial state $M^A(0) := \hat{y}$.
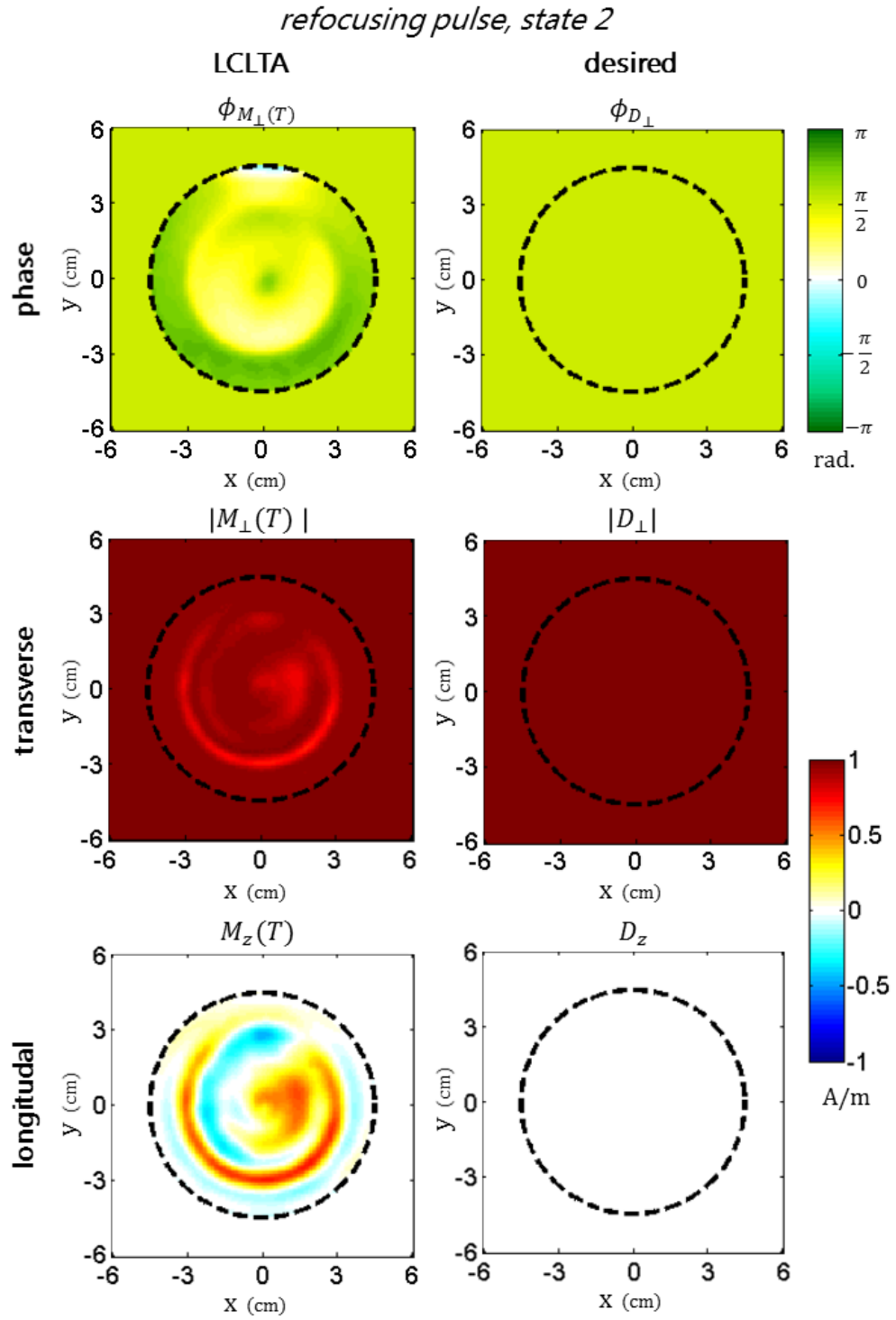
Figure 1.18: Comparison of the magnetization after the LCLTA refocusing pulse with the desired magnetization, for initial state $M^B(0) := \hat{x}$.

1.5   Summary

At the start of this chapter, the Bloch equation and its parameters were introduced and a numerical solution was given. In order to do fast simulations, this numerical solution was approximated by a function with an accuracy of order $\mathcal{O}(\Delta t^2)$, allowing for the response calculation to be done in $50N_tN_s$ flops.

Radiofrequency pulse design was introduced and applied using the linear class large-tip-angle (LCLTA) method. The method showed good results for the $\frac{\pi}{2}$ rad excitation pulse, but significant errors for the inversion and refocusing pulses. Finding a method for improving these pulses will be the subject of the next chapter.

Chapter 2

Optimal Control

In the previous chapter it was shown that radiofrequency pulses can be used to bring the magnetization to a desired state. The pulses were found by linearising and "inverting" the Bloch equation. But for larger flip angles, the error of the linear approximation increases. The *optimal control* method is able to iteratively improve the RF pulses, taking into account the non-linearity of the Bloch equation. In this chapter, the optimal control method will be introduced for radiofrequency pulse design and shown to improve the accuracy of the desired magnetization response, especially for refocusing pulses.

Most of the theory on optimal control in this chapter is based on the lecture notes by Chachuat [7]. Optimal control was initially introduced to pulse design for magnetic resonance by Conolly et al [8] and the extension to the multichannel case was described by Xu et al [23].

2.1 Basic framework

As the name suggests, *optimal control* is an optimization method, which attempts to minimize a cost function. In our case, the relative error between the desired magnetization profile D and the magnetization profile $M(T)$ at the end of the radiofrequency pulse. The relative error is quantified by the function φ in equation (1.70).

In the previous chapter, it was shown that the profile of $M(T)$ is the result of the application of radiofrequency and gradient waveforms, which can be defined by the user to fulfil certain constraints. In optimal control terminology the waveforms are the *control variables*, the magnetization is the *state variable* and the Bloch equation is the differential equation that relates the control to the state. The goal is to find the control function which brings

the magnetization from its initial state, $M(0)$, to the desired state D. Since this is not always exactly possible due to the relaxation of the magnetization over time, the optimal control functions are those that minimize the relative error φ.

In the following, our focus will be on RF pulse design and the gradient waveforms are left fixed. Although it is possible to include the gradients in the control, they are subject to additional constraints and it will be shown that only optimizing the RF pulse already gives good results.

In order to make the dependence of the magnetization dynamics on the RF pulses explicit, the Bloch equation (1.36) is rewritten for the L-channel case:

$$\dot{M}(r,t) = \left(\mathbf{A}(r,t) + \sum_{k=1}^{2L} \mathbf{N}_k(r) u_k(t) \right) M(r,t) + M_0(r)/T_1(r), \qquad (2.1)$$

where:

$$\mathbf{A}(r,t) := \begin{pmatrix} -1/T_2(r) & \gamma(G(t) \cdot r + \Delta B_0(r)) & 0 \\ -\gamma(G(t) \cdot r + \Delta B_0(r)) & -1/T_2(r) & 0 \\ 0 & 0 & -1/T_1(r) \end{pmatrix},$$

and for $l = 1, 2, \ldots, L$:

$$u_{2l-1}(t) := \mathrm{Re}(b_1^{(l)}(t)), \quad \mathbf{N}_{2l-1}(r) := \gamma \begin{pmatrix} 0 & 0 & -s_l^{(I)}(r) \\ 0 & 0 & s_l^{(R)}(r) \\ s_l^{(I)}(r) & -s_l^{(R)}(r) & 0 \end{pmatrix},$$

$$u_{2l}(t) := \mathrm{Im}(b_1^{(l)}(t)), \quad \mathbf{N}_{2l}(r) := \gamma \begin{pmatrix} 0 & 0 & -s_l^{(R)}(r) \\ 0 & 0 & -s_l^{(I)}(r) \\ s_l^{(R)}(r) & s_l^{(I)}(r) & 0 \end{pmatrix}.$$

Here $s_l^{(R)}(r) := \mathrm{Re}(s_l(r))$ and $s_l^{(I)}(r) := \mathrm{Im}(s_l(r))$. All terms in equation (2.1) are real-valued, and all are known except the RF pulses, u_l, of the different channels which will be described by the single control variable u:

$$u(t) = (u_1(t), u_2(t), \ldots, u_{2L}(t))^T \in \mathbb{R}^{2L}. \qquad (2.2)$$

Since the radiofrequency pulses should minimize the relative error of all voxels simultaneously, the Bloch equation is extended to describe to response of all voxels, by defining the block diagonal matrices:

$$\mathbf{A}(t) := \mathrm{diag}\left(\mathbf{A}(r_1,t), \mathbf{A}(r_2,t), \ldots, \mathbf{A}(r_{N_s},t)\right) \in \mathbb{R}^{3N_s \times 3N_s},$$
$$\mathbf{T}_1 := \mathrm{diag}\left([0,0,1/T_1(r_1)], \ldots, [0,0,1/T_1(r_{N_s})]\right) \in \mathbb{R}^{3N_s \times 3N_s}. \qquad (2.3)$$

and for $k = 1, \ldots, 2L$:

$$\mathbf{N}_k := \mathrm{diag}\left(\mathbf{N}_k(r_1), \mathbf{N}_k(r_2), \ldots, \mathbf{N}_k(r_{N_s})\right) \in \mathbb{R}^{3N_s \times 3N_s}. \qquad (2.4)$$

Then the Bloch equation becomes for $t \in [0, T]$:

$$
\begin{aligned}
\dot{M}(t) &= \left(\mathbf{A}(t) + \sum_{k=1}^{2L} \mathbf{N}_k u_k(t)\right) M(t) + \mathbf{T}_1 M_0 \\
&= \mathbf{\Omega}(t) M(t) + \mathbf{T}_1 M_0.
\end{aligned}
\qquad (2.5)
$$

with $M(t), M_0 \in \mathbb{R}^{3N_s}$ the concatenation of the magnetization vectors at all spatial positions, as defined in equation (1.71). The initial value $M(0) \in \mathbb{R}^{3N_s}$ is determined by the user and equals M_0 for the excitation and inversion pulses. Note that the magnetization response can still be calculated per voxel, but equation (2.5) will prove useful in describing the optimal control algorithm.

For use in the following chapter, a translated system is defined such that the equilibrium state, M_0, equals the zero state:

$$\hat{M}(t) := M(t) - M_0, \qquad\qquad \hat{M}_0 := 0. \qquad (2.6)$$

Noting that $\mathbf{A}(t)M_0 = -\mathbf{T}_1 M_0$ for all t, equation (2.5) becomes:

$$\dot{\hat{M}}(t) = \mathbf{A}(t)\hat{M}(t) + \sum_{k=1}^{2L} \mathbf{N}_k u_k(t)\hat{M}(t) + \mathbf{B}u(t), \qquad (2.7)$$

with:

$$\mathbf{B} := [\mathbf{N}_1 M_0, \mathbf{N}_2 M_0, \ldots, \mathbf{N}_{2L} M_0] \in \mathbb{R}^{3N_s \times 2L}. \qquad (2.8)$$

This translated system (2.7) emphasizes the bilinear response of the system.

Since the heating of tissue increases quadratically with the RF field strength, a regularization term on the pulse power is added to the cost function, leading to:

$$J(u) := \varphi(M(T)) + \frac{\alpha}{2}\int_0^T \|u(t)\|_2^2 \, dt, \quad \text{with } u \in \mathcal{C}^1([0, T], \mathbb{R}^{2L}). \qquad (2.9)$$

Here α is a fixed regularization parameter and $M(T)$ is determined by u through equation (2.5). Note that different costs can be added to the cost function, for instance to minimize the duration of the RF pulse. The problem of finding the optimal control results in a constrained optimization problem with cost functional (2.9) and constraint (2.5).

2.2 Optimization

The function u which minimizes J in equation (2.9) will be determined using an iterative method of the following form:

$$u^{(k+1)} = u^{(k)} + \mu_k d_k, \tag{2.10}$$

where in iteration k the intermediate solution $u^{(k)}$ is improved in the *search direction* d_k, both in $\mathcal{C}^1([0,T], \mathbb{R}^{2L})$, with *step size* $\mu_k \in \mathbb{R}$. For $u^{(0)}$, an initial guess is taken, such as the LCLTA solution described in the previous chapter. The procedure is stopped once the decrease of the cost function is smaller than a given tolerance value, such that it is suspected that the current solution is a local minimum of J.

Gradient-based methods will be used for selecting the search direction d_k. If the gradient direction $\nabla_u J(u)$ for given u can be calculated, then setting $d_k = -\nabla_u J(u^{(k)})$, leads to a decrease of J for small step sizes. This method of optimization is called *steepest descend*. More advanced methods increase convergence rate by taking into account the search directions of the previous iterations, such methods as: non-linear conjugate gradients (CG) method, the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method and limited-memory BFGS (LBFGS) method (see [18]).

For the step size, μ_k, a backtracking line search algorithm will be used, which starts by taking a large step and reducing it until a reduction of the cost function is found. The method is shown in algorithm 1, where the values of u_{max}, β_0 and β_1 are to be user defined with $0 < \beta_0 \leq \beta_1 < 1$.

Algorithm 1 Backtracking line search

$\mu = \mu_{max}$

while $J(u^{(k)} + \mu d_k) \geq J(u^{(k)})$ **do**

    $\mu = \beta_0 \mu$

end while

while $J(u^{(k)} + \beta_1 \mu d_k) \leq J(u^{(k)} + \mu d_k)$ **do**

    $\mu = \beta_1 \mu$

end while

$\mu_k = \mu$

At the end of the chapter, the different methods for selecting the search direction are tested in combination with the backtracking line search. There

is a large number of alternative methods available for both the search direction d_k and the step size μ_k. The tests therefore do not give a conclusive result on which method is most appropriate for RF pulse design through optimal control. Rather, an indication is given if there is a large difference in performance and result between methods.

2.2.1 Adjoint method

A standard approach to minimizing the cost functional (2.9) with constraint (2.5) is by minimizing a related functional $\bar{J}$, defined through the Lagrange formalism:

$$
\bar{J}(u, M, \lambda) := \varphi(M(T)) + \int_0^T \frac{\alpha}{2} \|u(t)\|_2^2
$$
$$
+ \lambda(t)^T \left[\left(\mathbf{A}(t) + \sum_{k=1}^{2L} \mathbf{N}_k u_k(t) \right) M(t) + \mathbf{T}_1 M_0 - \dot{M}(t) \right] dt. \quad (2.11)
$$

Here $\lambda(t) \in \mathbb{R}^{3N_s}$ is called the *costate* or *adjoint* variable, which is a yet to be determined continuously differentiable function on $[0, T]$. Equation (2.11) defines a functional on u, M, and λ. Candidate solutions of minimizing $\bar{J}$ unconstrained will be shown to be a candidate solution for the minimum of J with constraint (2.5).

In the following, notation is simplified by setting:

$$
H(t, u, M, \lambda) := \frac{\alpha}{2} \|u(t)\|_2^2 + \lambda(t)^T \left[\left(\mathbf{A}(t) + \sum_{k=1}^{2L} \mathbf{N}_k u_k(t) \right) M(t) + \mathbf{T}_1 M_0 \right].
$$
$$
(2.12)
$$

Substituting H in equation (2.11) gives:

$$
\bar{J}(u, M, \lambda) = \varphi(M(T)) + \int_0^T H(t, u, M, \lambda) - \lambda(t)^T \dot{M}(t) \, dt. \quad (2.13)
$$

A necessary condition for the minimum (u^*, M^*, λ^*) of $\bar{J}$, is that all directional derivatives are zero. For a functional such as $\bar{J}$, the directional derivatives equal the Gâteaux derivative, which is defined for functional F and functions x and ϕ as:

$$
dF(x, \phi) = \lim_{\tau \to 0} \frac{F(x + \tau\phi) - F(x)}{\tau} = \frac{d}{d\tau} F(x + \tau\phi)|_{\tau=0}. \quad (2.14)
$$

The variables x and ϕ are respectively the position and direction of the derivative and both are supposed to lie in the same linear space. Setting the directional derivatives of u, M, and λ at the minimum to zero, gives three equations a local minimum should fulfil.

First, the partial derivative of $\bar{J}$ with respect to λ is calculated at the minimum. Here it is supposed that $\lambda, \phi_\lambda \in \mathcal{C}^1([0,T], \mathbb{R}^{3N_s})$, and the following shorthand notation is used:

$$d\bar{J}(\lambda^*, \phi_\lambda) := d\bar{J}(u^*, M^*, \lambda^*, 0, 0, \phi_\lambda). \tag{2.15}$$

Similar notation will be used for the other partial derivatives. In this case:

$$
\begin{aligned}
d\bar{J}(\lambda^*, \phi_\lambda) &= \frac{d}{d\tau} \bar{J}(u^*, M^*, \lambda^* + \tau\phi_\lambda)\bigg|_{\tau=0} \\
&= \lim_{\tau \to 0} \int_0^T \frac{dH}{d\tau}(t, u^*, M^*, \lambda^* + \tau\phi_\lambda) \\
&\qquad - \frac{d}{d\tau}\left[\lambda^*(t) + \tau\phi_\lambda(t)\right]^T \dot{M}^*(t)\, dt \\
&= \int_0^T \phi_\lambda(t)^T \left[\nabla_\lambda H(t, u^*, M^*, \lambda^*) - \dot{M}^*(t)\right] dt,
\end{aligned}
\tag{2.16}
$$

with:

$$
\nabla_\lambda H(t, u^*, M^*, \lambda^*) := \\
\left[\frac{dH}{d\lambda(r_1)}(t, u^*, M^*, \lambda^*), \ldots, \frac{dH}{d\lambda(r_{3N_s})}(t, u^*, M^*, \lambda^*)\right]^T \in \mathbb{R}^{3N_s}. \quad (2.17)
$$

From equation (2.16) it follows that the derivative is zero for all directions ϕ_λ, if and only if for $t \in [0,T]$:

$$
\begin{aligned}
\dot{M}^*(t) &= \nabla_\lambda H(t, u^*, M^*, \lambda^*) \\
&= \left(\mathbf{A}(t) + \sum_{k=1}^{2L} \mathbf{N}_k u_k^*(t)\right) M^*(t) + \mathbf{T}_1 M_0.
\end{aligned}
\tag{2.18}
$$

This equals the Bloch equation (2.5), which leads to the following important observation: at the minimum of $\bar{J}$, the variables M^* and u^* are coupled by the Bloch equation and hence u^* satisfies the constraint for minimizing J.

Two notes are made before calculating the directional derivative with respect to the magnetization. First, the cost function $\bar{J}$ is rewritten using integration by parts to make the dependence on M explicit:

$$
\begin{aligned}
\bar{J}(u, M, \lambda) &= \varphi(M(T)) + \int_0^T H(t, u, M, \lambda) - \lambda(t)^T \dot{M}(t) \, dt \\
&= \varphi(M(T)) + \lambda(0)^T M(0) - \lambda(T)^T M(T) \qquad (2.19) \\
&\quad + \int_0^T H(t, u, M, \lambda) + \dot{\lambda}(t)^T M(t) \, dt.
\end{aligned}
$$

Second, since the initial magnetization $M(0)$ is given, it is a constant in the cost function and therefore M, ϕ_M are functions in $\mathcal{C}^1(]0, T], \mathbb{R}^{3N_s})$. Taking the derivative at the minimum using equation (2.19):

$$
\begin{aligned}
d\bar{J}(M^*, \phi_M) &= \frac{d}{d\tau} \bar{J}(M^* + \tau \phi_M) \bigg|_{\tau=0} \\
&= \left[\nabla_M \varphi(M^*(T))^T - \lambda^*(T)^T \right] \phi_M(T) \qquad (2.20) \\
&\quad + \int_0^T \left[\nabla_M H(t, u^*, M^*, \lambda^*)^T + \dot{\lambda}^*(t)^T \right] \phi_M(t) dt,
\end{aligned}
$$

which is equal to zero for all directions ϕ_M if:

$$
\begin{cases}
\lambda^*(T) &= \nabla_M \varphi(M^*(T)) = 2 \left(M^*(T) - D \right) / \|D\|_2^2, \\
\\
\dot{\lambda}^*(t) &= -\nabla_M H(t, u^*, M^*, \lambda^*) \\
&= \left(-\mathbf{A}(t) + \sum_{k=1}^{2L} \mathbf{N}_k u_k^*(t) \right) \lambda^*(t).
\end{cases} \qquad (2.21)
$$

It follows that at the minimum of $\bar{J}$, a differential equation closely similar to the Bloch equation (2.5) determines λ^*, with boundary value given a time T instead of time 0.

Finally, for the partial derivative of u in direction ϕ_u, both in $\mathcal{C}^1([0, T], \mathbb{R}^{2L})$:

$$
\begin{aligned}
d\bar{J}(u^*, \phi_u) &= \frac{d}{d\tau} \bar{J}(u^* + \tau \phi_u, M^*, \lambda^*) \bigg|_{\tau=0} \\
&= \int_{t=0}^T \nabla_u H(t, u^*, M^*, \lambda^*)^T \phi_u(t) dt.
\end{aligned} \qquad (2.22)
$$

Since this must equal zero for all directions ϕ_u, a necessary condition for the minimum is:

$$
0 = \frac{dH}{du_i}(t, u^*, M^*, \lambda^*) = \alpha u_i^*(t) + \lambda^*(t)^T \mathbf{N}_i M^*(t), \quad \text{for } 1 \leq i \leq 2L. \quad (2.23)
$$

Equations (2.18), (2.21), and (2.23) are collectively known as the *Euler-Lagrange equations*, and finding variables that satisfy all three constraints gives a candidate minimum of $\bar{J}$, what in extension will also be a candidate minimum of J.

To find variables that satisfy the Euler-Lagrange equations, the following remark is made: For given u and initial magnetization $M(0)$, the magnetization response is determined uniquely through the first necessary condition given in equation (2.18). Consequently, also λ is uniquely determined by second necessary condition in equation (2.21). Hence $\bar{J}$ with constraints (2.18) and (2.21) can be considered a functional on only u. This will be supposed from this point on, giving the following problem statement for finding the minimum (u^*, M^*, λ^*) of $\bar{J}$:

Find $u \in \mathcal{C}^1([0,T], \mathbb{R}^{2L})$, such that equation (2.23) is true for all $t \in [0,T]$, where M and λ as defined by equations (2.18) and (2.21) for given u and $M(0)$.

Note by definition of $\bar{J}$ in equation (2.11), defining the magnetization over time through the Bloch equation gives:

$$\bar{J}(u) = J(u), \text{ for all } u. \tag{2.24}$$

And furthermore:

$$
\begin{aligned}
dJ(u, \phi_u) &= \lim_{\tau \to 0} \frac{J(u + \tau\phi_u) - J(u)}{\tau} \\
&= \lim_{\tau \to 0} \frac{\bar{J}(u + \tau\phi_u) - \bar{J}(u)}{\tau} = d\bar{J}(u, \phi_u).
\end{aligned}
\tag{2.25}
$$

It follows that in this case $dJ(u, \phi_u) = 0$ if and only if $d\bar{J}(u, \phi_u) = 0$, leading to the important observation:

Equation (2.23) is a necessary condition for the minimum u^* of J, when M^* and λ^* are calculated through equations (2.18) and (2.21).

Finding an u that satisfies equation (2.23) then gives us a candidate solution for the minimum of J. For this the function H is used, which is defined in equation (2.12) and is a *Hamiltonian function* from control theory. As with $\bar{J}$, the function H can be considered only dependent on t and u, if M and λ are constrained by equations (2.18) and (2.21). Furthermore, equation (2.23) gives the gradient of H with respect to u, and shows that the minimum of J

is a critical point of H. Minimizing H then gives a candidate minimum for J. Setting in the iterative procedure (2.10):

$$d_k(t) = -\nabla_u H(t, u^{(k)}), \tag{2.26}$$

it follows from equation (2.22) that:

$$d\bar{J}(u^{(k)}, d_k) = \int_{t=0}^{T} -\nabla_u H(t, u^{(k)})^T \nabla_u H(t, u^{(k)}) dt \leq 0, \tag{2.27}$$

with $d\bar{J}(u^{(k)}, d_k) = 0$ if and only if equation (2.23) is fulfilled and $u^{(k)}$ is a candidate minimum. Updating the control in the direction which decreases H also decreases J, if the step size μ_k is chosen small enough, since:

$$\begin{aligned} J(u^{(k)} + \mu_k d_k) - J(u^{(k)}) &= \mu_k dJ(u^{(k)}, d_k) + \mathcal{O}(\mu_k^2) \\ &= \mu_k d\bar{J}(u^{(k)}, d_k) + \mathcal{O}(\mu_k^2) \\ &< 0, \text{ for small } \mu_k. \end{aligned} \tag{2.28}$$

The above equation motivates the choice of the backtracking linesearch (Alg. 1) for the step size μ_k.

2.2.2 Algorithm

In the previous section it was shown that a candidate minimum of J, defined in (2.9), can be found by minimizing H, defined in (2.12), where constraints (2.18) and (2.21) determine respectively M and λ. The calculation of the gradient $\nabla_u H$ in each iteration of (2.10) requires three steps:

1. For given initial state $M(0)$ and intermediate solution $u^{(k)}$, determine $M(t)$ for $t \in [0, T]$ through equation (2.18). This will be called the *forward step*.

2. Using the calculated $M(T)$ from step 1 and intermediate solution $u^{(k)}$, determine $\lambda(t)$ for $t \in [0, T]$ through equation (2.21). This will be called the *backward step*.

3. Calculate $\nabla_u H(t, u^{(k)}, M, \lambda)$ for $t \in [0, T]$, using equation (2.23).

In order to do these steps numerically, the variables u, M, λ and H are discretized in time, according to the discretization in equation (1.37). This gives the adjoint method shown in algorithm 2.

The computational complexity of this method is $\mathcal{O}(N_t N_s)$. In section 1.2.3 it

Algorithm 2 Adjoint method for gradient calculation

---

input   for $i := 1, 2, \ldots, N_t$: $t_i, u(t_i), G(t_i),$

            for $j := 1, 2, \ldots, N_s$: $r_j, M(r_j, t_0), D(r_j), T_1(r_j), T_2(r_j), \Delta B_0(r_j),$

                     and $\mathbf{N}_l(r_j)$ for $1 \le l \le 2L,$

output for $i := 1, 2, \ldots, N_t$: $\dfrac{dH}{du}(t_i).$

setting:

$\mathbf{D}_j := -\mathrm{diag}(1/T_2(r_j), 1/T_2(r_j), 1/T_1(r_j)),$

$\mathbf{B}_{i,j} := \gamma(G(t_i) \cdot r_j + \Delta B_0(r_j))(e_1 e_2^T - e_2 e_1^T) + \sum_{l=1}^{2L} \mathbf{N}_l(r_j) u_l(t_i).$

for $j := 1, 2, \ldots, N_s$, **do**

    for $i := 1, 2, \ldots, N_t$, **do**   ▷ 1: *forward*

        $M(r_j, t_i) = e^{\mathbf{D_j}\Delta t} e^{\mathbf{B}_{i,j}\Delta t} M(r_j, t_{i-1}) + \left[\mathbf{I} - e^{\mathbf{D}_j \Delta t}\right] M_0,$

    end for

    $\lambda(r_j, t_{N_t}) = 2\left(M(r_j, t_{N_t}) - D(r_j)\right)/\|D\|_2^2,$

    for $i := N_t, N_t - 1, \ldots, 1$, **do**   ▷ 2: *backward*

        $\lambda(r_j, t_{i-1}) = e^{-\mathbf{D}_j \Delta t} e^{\mathbf{B}_{i,j}\Delta t} \lambda(r_j, t_i),$

    end for

end for

for $l := 1, 2, \ldots, 2L$, **do**   ▷ 3: *gradient calculation*

    for $i := 1, 2, \ldots, N_t$, **do**

        $\dfrac{dH}{du_l}(t_i) = \sum_{j=1}^{N_s} \lambda(r_j, t_i)^T \mathbf{N}_l(r_j) M(r_j, t_i),$

    end for

end for

---

was shown that the complexity of calculating the response of M was of this order, and this is supposed approximately the same for λ since the differential equation (2.21) is solved in a similar manner. Furthermore, calculating the $\nabla_u H$ for each time point is also $\mathcal{O}(N_t N_s)$, according to equation (2.23).

A standard method for the numerical approximation of the gradient at time t is using finite difference methods, such as:

$$\frac{dH}{du_i(t_j)}\bigg|_{u'} = \frac{H(u') - H(u' + hu_i(t_j))}{h}. \tag{2.29}$$

For some small h. Calculating $H(u' + hu_i(t_j))$ for the $2L$ control variables and N_t time points requires $2LN_t$ response evaluations, giving a total complexity of $\mathcal{O}(N_t^2 N_s)$. Thus the adjoint method is expected to require much less time.

2.2.3   Improving search direction

Since the number of time and spatial points are typically large, calculating the response of a radiofrequency pulse and thereby the gradient through algorithm 2 is computationally expensive. The steepest descend method takes the opposite to the gradient of H as search direction, which locally equals the direction of fastest decrease of H. However, more advanced methods are able to increase the convergence speed by taking the search directions of the previous iterations into account. The methods that are implemented in the following section are: non-linear conjugate gradients (CG), Broyden-Fletcher-Goldfarb-Shanno (BFGS) and limited-memory BFGS (LBFGS) algorithm. The description is based on [18].

In this section, all control variables will be represented by a single vector u:

$$u := \left[u(t_1)^T, u(t_2)^T, \ldots, u(t_{N_t})^T\right] \in \mathbb{R}^{2LN_t}. \tag{2.30}$$

In this case H is a function on $\mathbb{R}^{2LN_t}$, where H is defined in (2.12) with M and λ defined by constraints (2.18) and (2.21). Supposing that in iteration k of the iterative method (2.10), H is twice differentiable at position $u^{(k)}$. Then by Taylor's theorem:

$$H(u^{(k)} + x) = f(x) + \mathcal{O}(|x|^3), \quad \text{for } x \in \mathbb{R}^{2LN_t}, \tag{2.31}$$

where:

$$f(x) = c - b^T x + \frac{1}{2}x^T \mathbf{A}x, \tag{2.32}$$

with $c := H(u^{(k)}) \in \mathbb{R}$, $b := -\nabla_u H(u^{(k)}) \in \mathbb{R}^{2LN_t}$, and $\mathbf{A} \in \mathbb{R}^{2LN_t \times 2LN_t}$ is the Hessian of H at $u^{(k)}$, which is symmetric and positive definite with $\mathbf{A}_{i,j} := \frac{d^2 H}{du_i du_j}(u^{(k)})$.

Choosing an update x^* that minimizes f approximately also minimizes H, if x^* is supposed small. The gradient of f is given by:

$$\nabla f(x) = \mathbf{A}x - b, \tag{2.33}$$

and at the minimum x^* of f, it is required that $\nabla f(x^*) = 0$ and thus $\mathbf{A}x^* = b$. The optimal step x^* in iteration k can thus be found by solving a linear system. The Hessian matrix of H is unknown to us, and the CG and the BFGS methods differ in how the matrix product $\mathbf{A}x$ is calculated. Either implicitly or through the construction of an approximate Hessian matrix.

Non-linear conjugate gradients

The conjugate gradients method is aimed at iteratively solving the linear system $\mathbf{A}x = b$. for $x \in \mathbb{R}^{2LN_t}$, where $\mathbf{A} \in \mathbb{R}^{2LN_t \times 2LN_t}$ is a symmetric and positive definite matrix. The step size μ_k and search direction d_k are given by the recurrence relation:

$$\mu_k = \frac{r_k^T r_k}{d_k^T \mathbf{A} d_k} \in \mathbb{R}, \tag{2.34}$$

$$d_{k+1} = r_{k+1} + \beta_k d_k \in \mathbb{R}^{2LN_t}, \tag{2.35}$$

with $r_{k+1} = r_k - \mu_k \mathbf{A} d_k \in \mathbb{R}^{2LN_t}$, and $\beta_k = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k} \in \mathbb{R}$, and initial values for given x_0:

$$d_0 = r_0 = b - \mathbf{A}x_0 \tag{2.36}$$

These recurrences lead to the following properties:

$$r_k^T r_j = 0, \quad d_j^T \mathbf{A} d_k = 0, \quad r_k^T d_j = 0, \quad \text{for } j < k. \tag{2.37}$$

Furthermore, r_k equals the residual of the approximate solution x_k, $r_k = b - \mathbf{A}x_k$, and since each new residual is orthogonal to all previous ones:

$$\lim_{k \to \infty} r_k = \lim_{k \to \infty} (b - \mathbf{A}x_k) = 0 \tag{2.38}$$

Applying conjugate gradients to find the minimum of H directly is not possible since the Hessian matrix $\mathbf{A}$ in (2.32) is unknown, and hence equation (2.34) is not directly solvable. But this step can be performed implicitly, not

requiring the explicit knowledge of $\mathbf{A}$ at the cost of a line search for determining μ_k.

Suppose the gradient of f is calculated at position x_k:

$$\nabla f(x_k) = \mathbf{A} x_k - b = -r_k. \tag{2.39}$$

Moving from x_k in an unspecified direction d_k with step size μ_k, defined by:

$$\mu_k := \arg\min_\mu \left\{ f(x_k + \mu d_k) \right\}. \tag{2.40}$$

At position $x_{k+1} := x_k + \mu d_k$, the gradient of f equals:

$$\nabla f(x_{k+1}) = \mathbf{A} x_{k+1} - b = \mathbf{A}(x_k + \mu_k d_k) - b = \nabla f(x_k) + \mu_k \mathbf{A} d_k \tag{2.41}$$

Since x_{k+1} lies at the minimum of f along line $x_k + \mu d_k$, at this position $d_k^T(\nabla f(x_{k+1})) = 0$ and thus by equation (2.41):

$$d_k^T \nabla f(x_k) = d_k^T \mu_k \mathbf{A} d_k \tag{2.42}$$

Noting $\nabla f(x_k) = -r_k$, it follows that the step size follows equation (2.34), and the recurrence relation for r_k can be replaced with equations (2.39) and (2.40).

Different choices for β_k lead to different methods and influence the convergence of the algorithm to the minimum. An overview of the different methods is given in [12]. In this thesis the method of Polak and Ribière is chosen, since it is often used in practice:

$$\beta_k = \frac{\nabla H(u^{(k+1)})^T (\nabla H(u^{(k+1)}) - \nabla H(u^{(k)}))}{\|\nabla H(u^{(k)})\|_2^2}. \tag{2.43}$$

Limited-memory BFGS

The gradient of f in equation (2.33) implies that the step d_k that minimizes f has $\mathbf{A} d_k = b$, and if $\mathbf{A}$ is invertible:

$$d_k = -\mathbf{A}^{-1} \nabla f(x_k). \tag{2.44}$$

Equation (2.44) defines Newton's method. The BFGS method is a *quasi-Newton* method since instead of using the inverse Hessian in equation (2.44), an approximate matrix $\mathbf{B}_k \approx \mathbf{A}^{-1}$ is used and updated after each iteration. The update is based on how the positions x_k and gradients $\nabla f(x_k)$ change between iterations. Define:

$$s_k := x_{k+1} - x_k, \quad y_k := \nabla f(x_{k+1}) - \nabla f(x_k), \quad \rho_k := (y_k^T s_k)^{-1}. \tag{2.45}$$

Note that for small s_k, it follows from equation (2.33) that:

$$\mathbf{A}s_k = \mathbf{A}x_{k+1} - \mathbf{A}x_k \approx \nabla f(x_{k+1}) - \nabla f(x_k) = y_k, \qquad (2.46)$$

and thus $s_k \approx \mathbf{A}^{-1}y_k$. The matrix $\mathbf{B}_k$ is constructed to approximate $\mathbf{A}^{-1}$, by requiring:

1. $s_k = \mathbf{B}_k y_k$,

2. the matrix $\mathbf{B}_k$ is symmetric and positive definite.

Furthermore, to do the calculations efficient, the matrix $\mathbf{B}_k$ is improved every iteration by a low rank update. A update formula that fits these requirements is the following *BFGS formula*:

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \rho_k \left[\rho_k y_k^T \mathbf{B}_k y_k + 1 \right] s_k s_k^T - \rho_k \left[s_k y_k^T \mathbf{B}_k + \mathbf{B}_k y_k s_k^T \right], \qquad (2.47)$$

where $\mathbf{B}_0$ is an initial guess for the inverse Hessian matrix which should be symmetric and positive definite. And after k iterations:

$$\mathbf{B}_k = \mathbf{B}_0 + \sum_{i=0}^{k-1} \rho_i \left[\rho_i y_i^T \mathbf{B}_i y_i + 1 \right] s_i s_i^T - \rho_i \left[s_i y_i^T \mathbf{B}_i + \mathbf{B}_i y_i s_i^T \right]. \qquad (2.48)$$

The limited-memory BFGS method (LBFGS) was developed in order to reduce the memory requirements by not storing the whole matrix $\mathbf{B}_k$ or all the vectors required for the low rank updates in equation (2.48), but only the vectors used in the last m updates. If only the last m updates are used, the lower limit in equation (2.48) is set to $i_0 := \min(0, k - m)$, and $\mathbf{B}_{k_0} = \mathbf{B}_0$. Updating the matrix $\mathbf{B}_k$ only based on the last m updates still requires the computation of $\mathbf{B}_k$ for $i \in i_0 + 1, \ldots, k - 1$ using formula (2.48). However the updates can also be written as:

$$\begin{aligned} \mathbf{B}_{k+1} &= (I - \rho_k y_k s_k^T)^T \mathbf{B}_k (I - \rho_k y_k s_k^T) + \rho_k s_k s_k^T \\ &= v_k^T \mathbf{B}_k v_k + \rho_k s_k s_k^T, \end{aligned} \qquad (2.49)$$

where $v_k := (I - \rho_k y_k s_k^T)$. In this case the formula for the approximate Hessian matrix $\mathbf{B}_k$ at step k is given by:

$$\mathbf{B}_k = \left[\prod_{k=0}^{k-1} v_k \right]^T \mathbf{B}_0 \left[\prod_{k=0}^{k-1} v_k \right] + \sum_{k=0}^{k-1} \left[\prod_{j=i+1}^{k-1} v_j \right]^T s_k \rho_k s_k^T \left[\prod_{k=j+1}^{k-1} v_j \right]. \qquad (2.50)$$

Determining $\mathbf{B}_k$ based only on the last m steps is equal to setting $v_k = I$ and $\rho_k = 0$ for $i < k - m$. This is the algorithm that will be used to calculate the approximate inverse Hessian matrix $\mathbf{B}_k$ and as a consequence the search direction $d_k = -\mathbf{B}_k \nabla_u H(u^{(k)})$. The method is shown in algorithm 3. Here, $\mathbf{B}_0 := I$ was used as initial guess for the inverse Hessian.

Algorithm 3 Determine search direction d_k through limited-memory BFGS with m vectors

input	for $i_0 := \max(0, k - m) \leq i \leq k - 1$:
	$s_i = u^{(i+1)} - u^{(i)},$
	$y_i = \nabla H(u^{(i+1)}) - \nabla H(u^{(i)}),$
	$\rho_i = 1/y_i^T s_i,$
	$\mathbf{B}_0 := I,$
	$q = \nabla H(u^{(k)}).$
output	$d_k.$

for $i = k - 1, k - 2, \ldots, i_0$ **do**

$\qquad \alpha_i = \rho_i s_i^T q$

$\qquad q = q - \alpha_i y_i$

end for

$d_k = \mathbf{B}_0 q$

for $i = i_0, i_0 + 1, \ldots, k - 1$ **do**

$\qquad \beta = \rho_i y_i^T d_k$

$\qquad d_k = d_k + s_i(\alpha_i - \beta_i)$

end for

2.3 Numerical tests

Two different tests with the optimal control method are performed in this section. First we attempt to improve the pulses that were calculated in section 1.4, where it was shown that the LCLTA method was well able to design the $\frac{\pi}{2}$ rad excitation pulse but was less successful with the inversion and refocusing pulses. Secondly the effect of varying the regularization parameter α in J (see eq. (2.9)) is investigated for refocusing pulses. Similar to the L-curve if the LCLTA method, a high value of α is expected to promote solution with a small l_2-norm at the cost of a larger relative error between the magnetization profile at the end of the pulse, $M(T)$, and the desired magnetization D.

2.3.1 Method

For the following tests, the same parameters as the numerical tests in the previous chapter are used. The optimal control method are tested using the following numerical methods for the search direction d_k in each iteration: steepest descent (SD), nonlinear conjugate gradients (CG), the limited-memory BFGS method with two (LBFGS2) or three (LBFGS3) updates, and the normal BFGS method.
For the backtracking line search (see algorithm 1), parameters $\mu_0 = 0.5$, $\beta_0 = 0.5$ and $\beta_1 = \sqrt{\beta_0}$ are used. The regularization parameter α is initially set to 10^{-2}.

The methods are compared by the number of function evaluations (Bloch equation solutions) required to find a local minimum. A local minimum is supposed to be found if the change in objective value was less than 10^{-3} in the last 10 iterations. The methods that use previous search directions for the calculation of a new direction are restarted before exiting, which means that all previous search directions are set to zero.

For a given control variable u, determining the cost $J(u)$, given in equation (2.9), requires $M(T)$ and therefore one response evaluation of the Bloch equation, while determining the gradient $\nabla_u H(u)$, given in (2.21), additionally involves determining $\lambda(t)$ for $t \in [0, T]$. Since the response computation requires the most computation time of the method, the number of response evaluation is a good indicator for the total simulation time.

2.3.2   Excitation pulse

Section 1.4.1 showed that the LCLTA method performed well for the $\frac{\pi}{2}$ rad excitation pulse, but according to the convergence graph in Figure 2.1 and the results in Table 2.1 the optimal control method was still able to slightly improve it. Comparing the convergence of the different optimization methods, the BFGS method is the fastest and SD the slowest. BFGS seems to have approximately converged to the minimum after 300 response evaluations, while SD requires 800 response evaluations.

Table 2.1 shows a summary of the results of the different optimization methods compared with the LCLTA method. All optimal control methods have the same relative error and approximately the same RF norm. Compared with the LCLTA method, the optimal control methods reduce the relative error at the cost of a higher radiofrequency pulse norm.

The RF channel amplitudes of the optimal control solutions of the different methods are compared with the LCLTA case in Figure 2.2. All optimal control methods find approximately the same minimum, not distinguishable from each other in the graph, but distinct from the LCLTA solution.

For the CG solution, the magnetization response profile at time T is compared with that of the LCLTA method in Figure 2.3. There is no observable difference in magnetization response between the two methods. Note that the desired magnetization profiles are shown in Figure 1.13. This result indicates that the optimal control method does not significantly improve the LCLTA solution and therefore is not required for the design of $\frac{\pi}{2}$ rad excitation pulses.

excitation pulse

method	LCLTA	optimal control				
		SD	CG	LBFGS2	LBFGS3	BFGS
$J(u)$	0.097	0.067	0.067	0.067	0.067	0.067
$\phi(M(T))$	0.091	0.059	0.059	0.059	0.059	0.059
$\|u\|_2$	11.1	12.2	12.2	12.3	12.1	12.4

Table 2.1: Results of the LCLTA and optimal control methods for the $\frac{\pi}{2}$ rad excitation pulse, with J the cost function (2.9) and $\phi(M(T))$ the relative error (1.70).
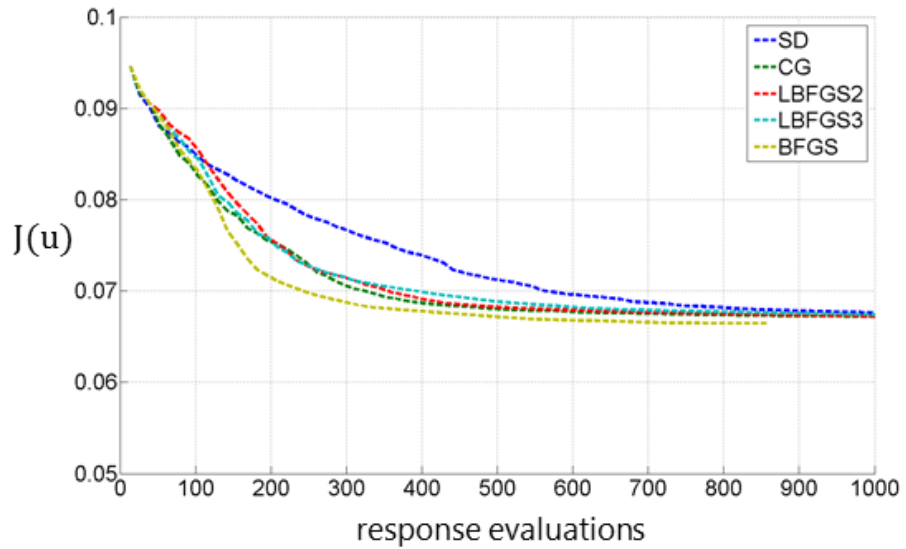
Figure 2.1: Convergence of the optimal control method for different optimization methods for the excitation pulse ($\frac{\pi}{2}$ rad). The graph is truncated at 1000 response evaluations.
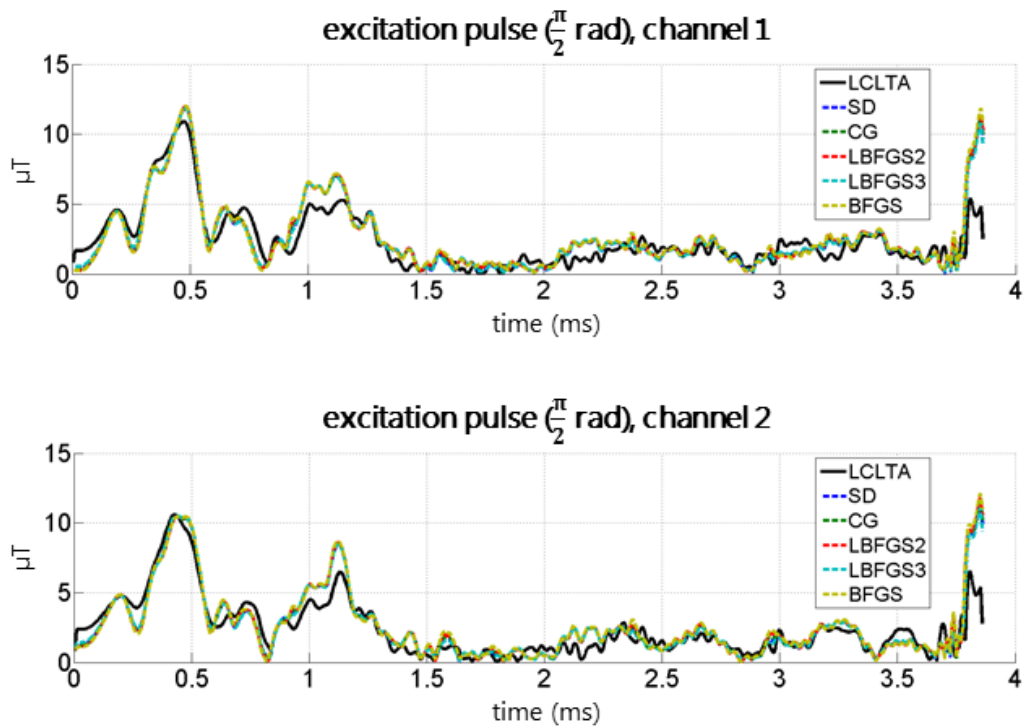
Figure 2.2: RF amplitudes of both channels for the $\frac{\pi}{2}$ rad excitation pulse, calculated using LCLTA and different optimal control methods.

58

Figure 2.3: The transverse and longitudinal magnetization response at time T of the excitation pulse ($\frac{\pi}{2}$ rad) designed by LCLTA and optimal control method using conjugate gradients.

2.3.3   Inversion pulse

The LCLTA inversion pulse designed in Section 1.4.2 showed large errors in the transverse magnetization at the end of the pulse. For the different optimal control methods the convergence graph is given in Figure 2.4, and shows a large reduction of the cost function for all cases. Most methods have approximatedly converged after 200 response evaluation. Much faster than in the excitation pulse case. The BFGS method converges the slowest with a sudden reduction after 900 response evaluation. This reduction occured after a restart of the BFGS method ($\mathbf{B}_k = I$, see Section 2.2.3).

The results of the different optimal control methods are compared with the LCLTA method in Table 2.2. In contrast to the excitation pulse case, the optimal control methods lead to different solutions. However, all the optimal control pulses greatly reduce the error in the magnetization at the cost of a small increase of the RF norm. The BFGS method is the most extreme case, with the lowest relative error at the cost of a highest RF norm. Determining which result is best depends on the practical limit of the RF amplitude.

The difference between the optimal control solutions is also illustrated in Figure 2.5, where the RF amplitudes of the different methods are shown. The optimal control pulses are more distinguishable than in the excitation pulse case. The BFGS pulse clearly has the largest RF amplitude with a peak at the end of the pulse.

inversion pulse

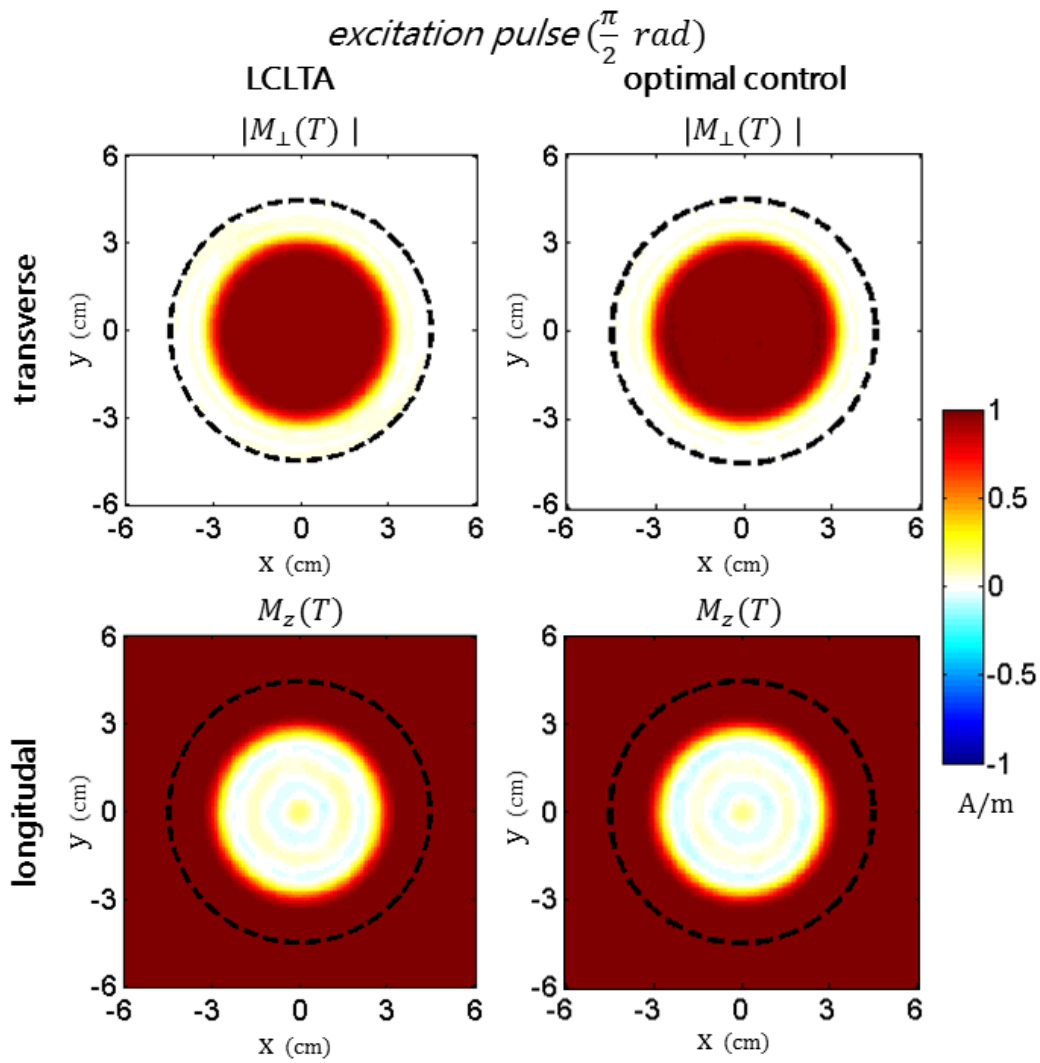| method	LCLTA	optimal control				
		SD	CG	LBFGS2	LBFGS3	BFGS
$J(u)$	0.290	0.166	0.166	0.167	0.167	0.160
$\phi(M(T))$	0.266	0.140	0.140	0.140	0.138	0.112
$\|u\|_2$	22.1	22.5	22.6	23.0	24.3	31.0

Table 2.2: Results of the LCLTA and optimal control methods for the inversion pulse, with J the cost function (2.9) and $\phi(M(T))$ the relative error (1.70).
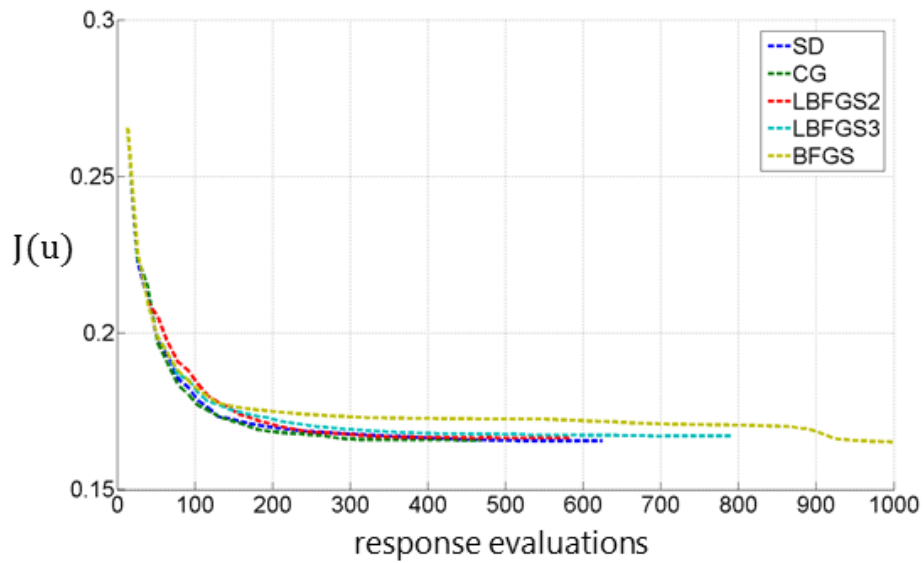
Figure 2.4: Convergence of the optimal control method for different optimization methods for the inversion pulse. The graph is truncated at 1000 response evaluations.
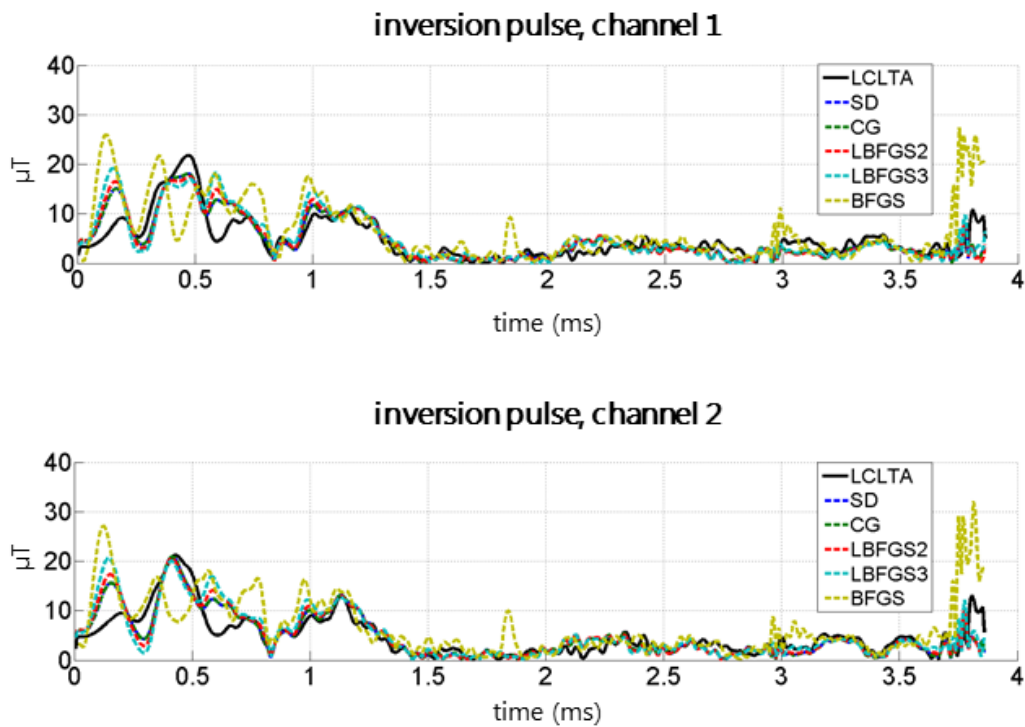
Figure 2.5: RF amplitudes of both channels for the inversion pulse, calculated using LCLTA and different optimal control methods.

61

Figure 2.6 shows the magnetization response of the CG pulse compared with the LCLTA case. The optimal control method leads to a much more homogeneous transverse magnetization through the center of the disk. Note that the BFGS method has the lowest relative error, but the result of CG is shown since in this case the RF amplitude is approximately the same as in the LCLTA case. The results of this section shows that a large reduction of the magnetization error is possible without requiring a large increase in RF norm.
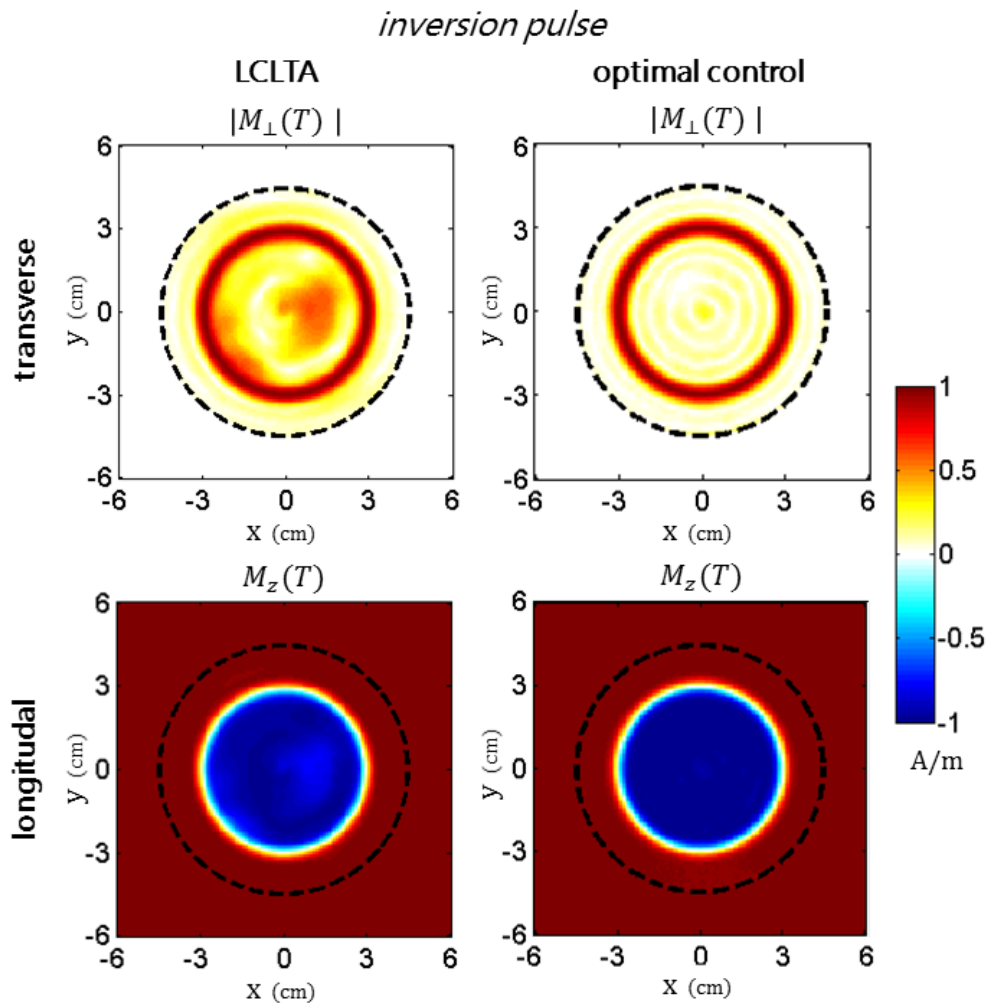
Figure 2.6: The transverse and longitudinal magnetization response at time T of the inversion pulse designed by LCLTA and optimal control method using conjugate gradients.

2.3.4 Two-state refocusing pulse

Designing the refocusing pulse through LCLTA method resulted in large errors in the magnetization profile after the pulse, most observable in the phase of the transverse magnetization, see Section 1.4.3. In order to apply the optimal control method to design refocusing pulses with two initial states per voxel, the two magnetization profiles are concatenated:

$$M(t) := \begin{pmatrix} M^A(t) \\ M^B(t) \end{pmatrix} \text{ for } t \in [0, T], \quad \text{and } D := \begin{pmatrix} D^A \\ D^B \end{pmatrix}, \tag{2.51}$$

where $M^A(r, 0) = \hat{y}$, $D^A(r) = -\hat{y}$, $M^B(r, 0) = D^B(r) = \hat{x}$, for each position r as illustrated in Figure 1.7. Note that compared to the inversion and excitation pulses the length of the magnetization vector doubles, and therefore also the simulation time of the Bloch equation.

The convergence of the different optimal control methods is shown in Figure 2.7. The SD method converges the slowest, requiring approximately 600 response evaluations. The other method converge in half the time.
The results of the optimization methods are shown in Table 2.3. Compared to the LCLTA pulse, the optimal control methods greatly improve the relative error of the magnetization at the cost of higher RF amplitude. Balancing the RF amplitude while obtaining a desired magnetization pattern through regularization in the cost function J will be the subject of the next section.

The RF amplitudes of the optimal control methods are compared with the LCLTA solution in Figure 2.8. It shows that indeed the RF amplitude is high compared to the LCLTA case. The difference betweeen the optimal control solutions is small.

two-state refocusing pulse

method	LCLTA	optimal control				
		SD	CG	LBFGS2	LBFGS3	BFGS
$J(u)$	0.650	0.260	0.260	0.260	0.259	0.262
$\phi(M(T))$	0.626	0.207	0.207	0.199	0.199	0.198
$\|u\|_2$	22.1	32.5	32.7	34.9	34.6	35.6

Table 2.3: Results of the LCLTA and optimal control methods for the refocusing pulse, with J the cost function (2.9) and $\phi(M(T))$ the relative error (1.70).
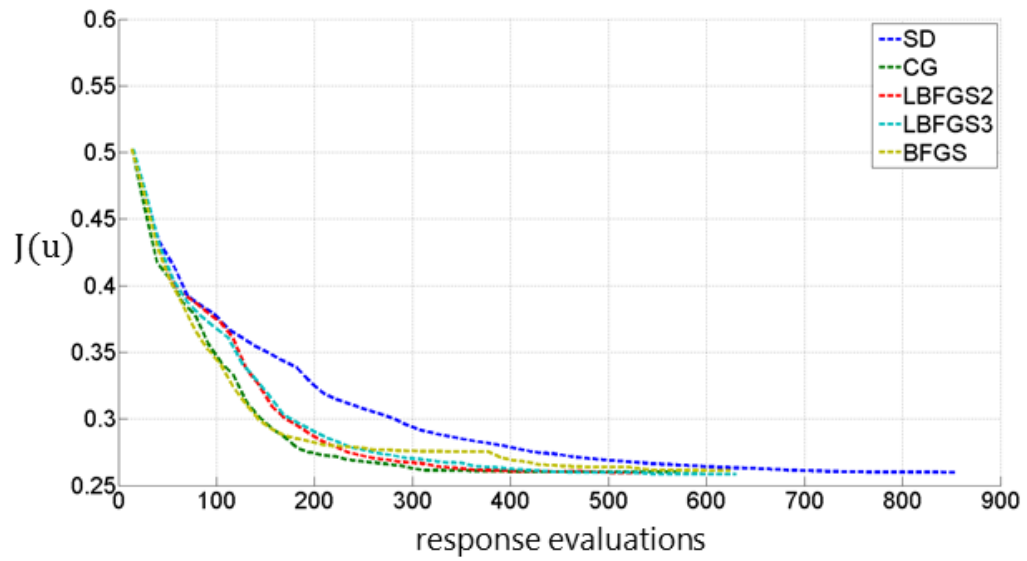
Figure 2.7: Convergence of the optimal control method for the refocusing pulse using different optimization methods.
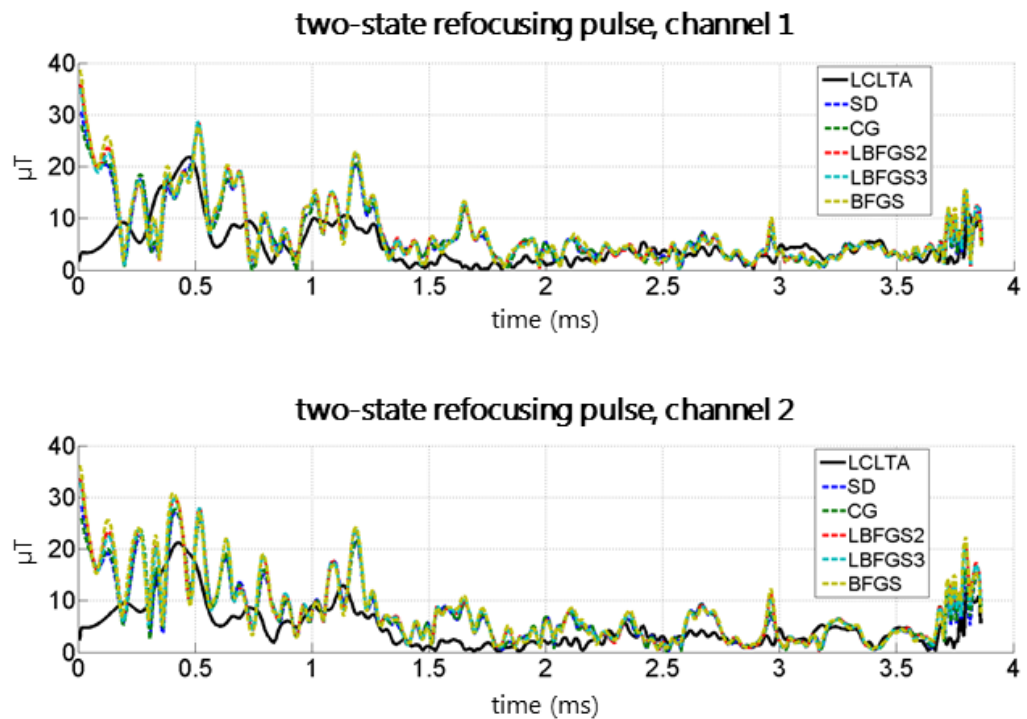
Figure 2.8: RF amplitudes of both channels for the refocusing pulse, calculated using LCLTA and different optimal control methods.

64

The effect of the optimal control pulse (CG) on the two different initial states are shown in respectively Figures 2.9 and 2.10. These results show that the optimal control method greatly improves the magnetization profile compared to the LCLTA case. Especially the phase profile is improved compared with the LCLTA method.

However, Table 2.3 showed that all optimal control solutions have a much higher RF norm than the LCLTA pulse. Limiting the RF norm is required for using the optimal control refocusing pulse in practice.
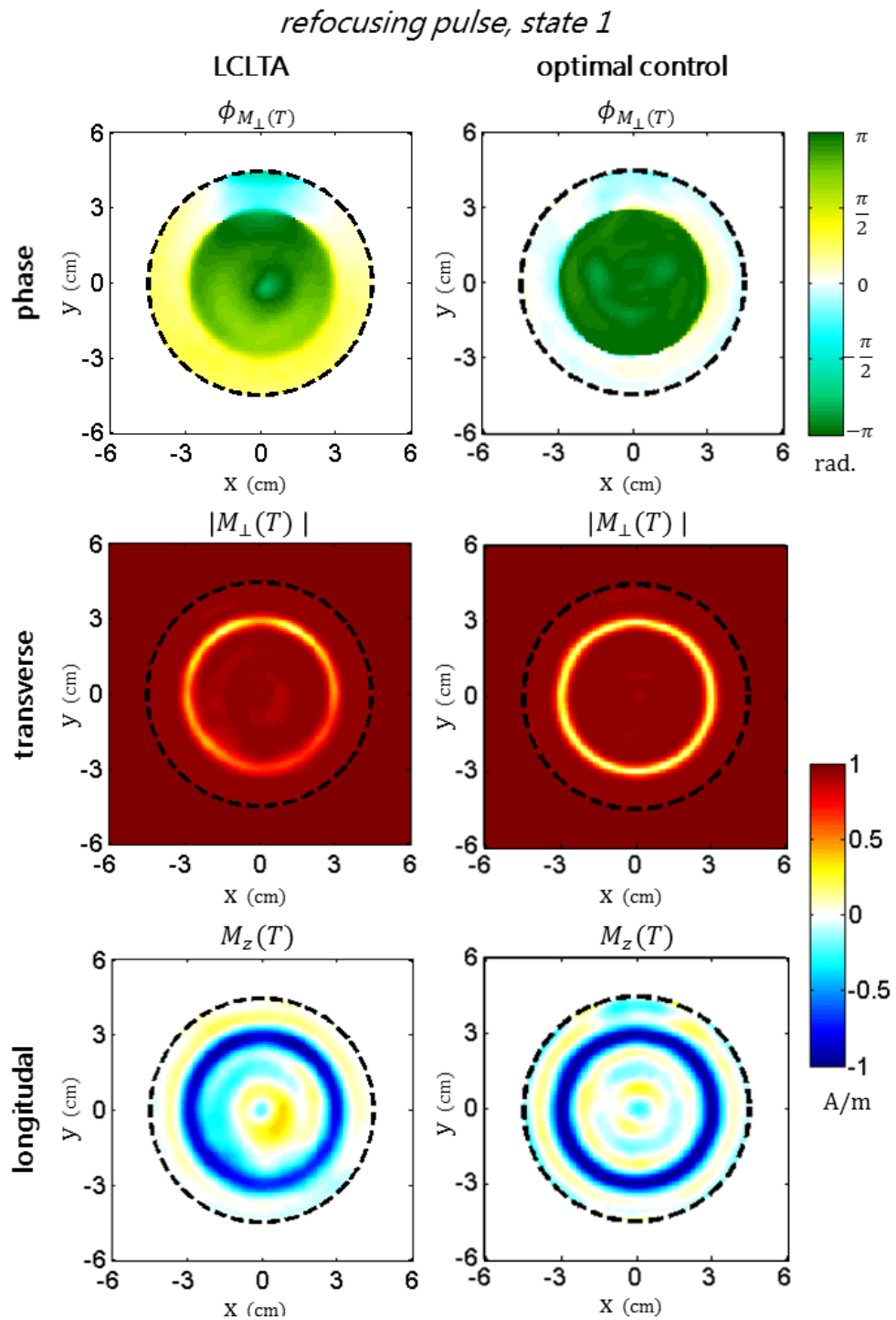
Figure 2.9: Comparison of the magnetization after the LCLTA refocusing pulse with that of the optimal control method (CG), for initial state $M^A(0) := \hat{y}$. The desired response is shown in Figure 1.17.
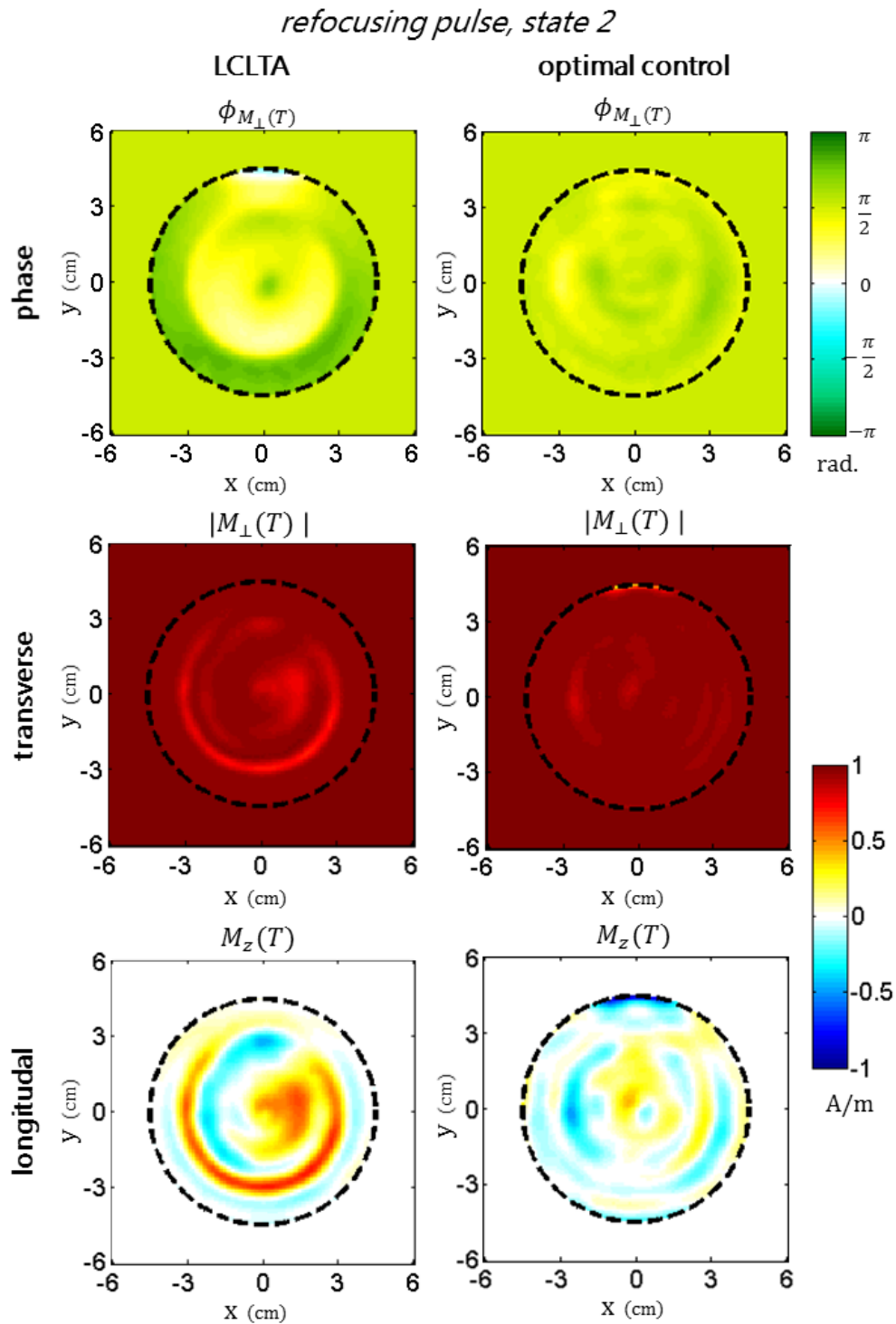
Figure 2.10: Comparison of the magnetization after the LCLTA refocusing pulse with that of the optimal control method (CG), for initial state $M^B(0) := \hat{x}$. The desired response is shown in Figure 1.18.

2.3.5 Effect of the regularization parameter α

In the previous experiments, the cost function J (eq. (2.9)) was minimized using $\alpha = 10^{-2}$. Compared to the LCLTA method, this led for each pulse type to a reduction in relative error of the magnetization at the cost of an increase in the RF amplitude. In practice, a low RF amplitude is required to prevent heating of the object. In this section, the effect of varying the regularization parameter α on the RF amplitude and magnetization response error is investigated.

For given value of α, the solution of the optimal control method is denoted u_α and its associated magnetization response, M_α. All other parameters are the same as for the two-state refocusing pulse in Section 2.3.4. It is expected that the RF norm and magnetization error for different α follow an L-curve as in Figure 1.9, with a high or low value of α leading to respectively a small RF amplitude or small magnetization error. The optimal control method was applied using conjugate gradients method since it showed relatively good results in the previous sections.

Table 2.4 shows the results of the optimal control method for different values of α. Figure 2.11 places the RF amplitude against the relative error of the magnetizatio, giving indeed an L-curve. An increase in the value of α leads to a decrease of the RF norm and an increase of the relative error in the magnetization. Furthermore, Table 2.4 indicates that the number of iterations to find the minimum seems to have an inverse relationship with α. For small values of α the convergence is slower. This could be due to the fact that for small values of α the RF amplitude becomes less important in the cost and hence the solution space grows.

The best value of α depends on the maximum allowed pulse amplitude. Compared with the LCLTA solution, taking $\alpha = 0.1$ leads to a large decrease in the relative error at the cost of a small increase in the RF amplitude while $\alpha = 0.5$ both decreases the relative error and the amplitude.

two-state refocusing pulse

α	0.001	0.005	0.010	0.050	0.100	0.500	1.000
$\varphi(M_\alpha(T))$	0.200	0.208	0.207	0.260	0.315	0.442	0.723
$\|u_\alpha\|_2$	35.5	33.4	32.7	26.7	23.4	18.5	13.7
fun. eval.	588	476	616	308	196	238	112

LCLTA solution: relative error $= 0.622$, $\|u\|_2 = 22.1$

Table 2.4: Effect of the regularization parameter α on the relative error, radiofrequency amplitude and number of response evaluations of the optimal control solution (CG) for the two-state refocusing pulse.
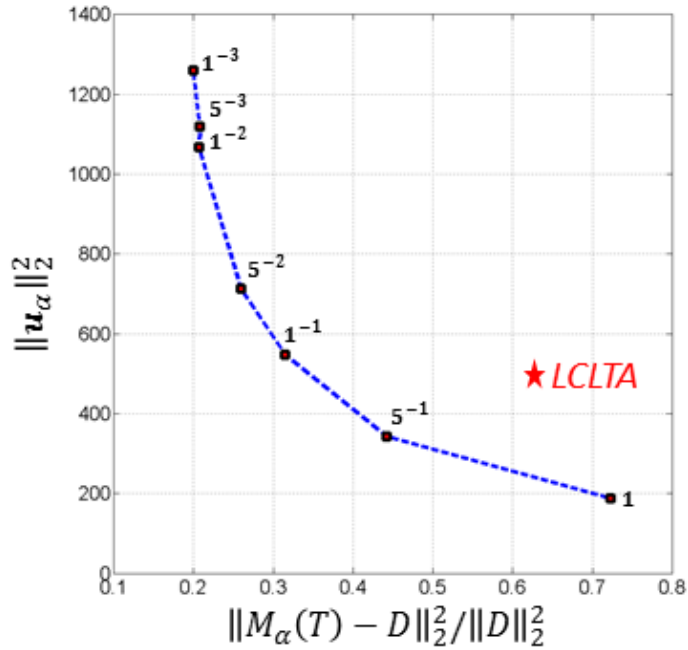
Figure 2.11: L-curve of the squared amplitude of the optimal control solution, u_α, against the squared relative error of its magnetization response M_α, for different values of α as indicated in the graph.

2.4 Discussion

The subject of this chapter was to apply the optimal control method for RF pulse design. The motivation of the method was the inability of the LCLTA method to design accurate inversion and refocusing pulses as presented in the previous chapter. Furthermore, the optimal control method allows the inclusion of relaxation effects, which can be important when attempting to simulate larger sequences.

The results in sections 2.3.3, and 2.3.4 show that the optimal control method was able to greatly improve the magnetization response of the inversion and refocusing pulses compared with the LCLTA method. The optimal control method did increase RF amplitude and thus heating compared with the LCLTA method. However, results from Section 2.3.5 showed that balancing RF amplitude and magnetization response error can be effectively done using the regularization parameter α. An interesting method for future research, is performing regularization on the iteration number as described in [13]. The idea is that in each iteration the magnetization error decreases at the cost of an increase of the RF norm. If a line graph is made of the magnetization error against the RF norm of each iteration, it might follow an L-curve which allows for selection of the optimal solution.

A large part of this chapter was focused on different numerical methods for determining the search directions of the iterative process. Calculating the gradient and the value of the cost function for different RF pulses is computationally expensive, and decreasing the number of search directions required to find the minimum can greatly increase convergence speed. The tests in this chapter did not indicate a clear superior between methods, but more numerical experiments are required for a conclusive result. Steepest descend had slow convergence in each numerical test and thus looking into more advanced numerical methods is worthwhile. Furthermore, a good idea for further research would be to investigate when to restart the BFGS method, since this method seems to lose convergence speed when it has not been restarted in a number of iterations.

Even when using more advanced numerical methods, it takes hundreds of computationally costly repetitions of solving the Bloch equation in order to find the optimal RF pulse. This is the motivation for attempting to reduce the simulation time of the Bloch equation through model order reduction in the next chapter.

Chapter 3

Model order reduction

In the previous chapter, it was shown that the optimal control method can greatly improve the inversion and refocusing pulses. However, the method is computationally costly since it requires the repeated evaluation of the magnetization response through the solution of the Bloch equation. Simulation time scales with the number of spatial and temporal points. Controlling the magnetization in a large region or with high resolution quickly leads to a large number of spatial positions, while the radiofrequency pulses are defined on the microsecond scale, giving thousands of time points for pulses of a few milliseconds.

The focus of this chapter is determining the response of the magnetization using fewer spatial or temporal variables, thus reducing simulation time.

In Section 3.1, projection techniques for the spatial domain are discussed and attempted. No previous work on spatial reduction of the Bloch equations has been found and such a method would be easily included in the optimal control framework.

In Section 3.2, temporal domain reduction is discussed. Two ideas from recent articles are presented, and in Section 3.2.2 an original method for rapid MR sequence simulations is shown.

3.1   Spatial domain

In this section, we attempt to find a method for projecting the Bloch equation (2.1) for all spatial positions, or full order model (FOM), to a reduced order model (ROM). The motivation of creating such a ROM is discussed, and two different projection methods are shown. First, the *principle orthogonal directions* (POD) method based on the singular value decomposition of the data. And second, *moment matching* methods based on the Krylov

71

space of the Bloch equation matrix. The discussion is largely based on the book by Antoulas [2] and the lecture notes by Amsallem [1].

The magnetization dynamic will be described using the translated Bloch equations (2.7) such that the equilibrium state equals the zero vector. For spatial domain reduction, it is assumed that the dynamics of magnetization response of all voxels is contained in a k-dimensional subspace $\mathcal{S}$ with $k \ll 3N_s$. Then there is a basis $\mathbf{V}_{\mathcal{S}} = [V_1, V_2, \ldots, V_k] \in \mathbb{R}^{3N_s \times k}$ and a time-dependent coordinate vector $C(t) \in \mathbb{R}^{k \times 1}$, such that:

$$\hat{M}(t) = \mathbf{V}_{\mathcal{S}} C(t), \text{ for } t \in [0, T]. \tag{3.1}$$

The vector $C(t)$ is called the reduced state vector and can be substituted in the translated Bloch equations (2.7) to get:

$$\mathbf{V}_{\mathcal{S}} \dot{C}(t) = \left(\mathbf{A}(t) + \sum_{k=1}^{2L} \mathbf{N}_k u_k(t) \right) \mathbf{V}_{\mathcal{S}} C(t) + \mathbf{B} u(t). \tag{3.2}$$

In practice, the matrix $\mathbf{V}_{\mathcal{S}}$ might not be known or the magnetization trajectories are only approximately contained within $\mathcal{S}$. Suppose we have the matrix $\mathbf{V}$ with $\text{span}(\mathbf{V}) = \mathcal{V}$, and the same substitution is done, then since the Bloch equation might move the state out of $\mathcal{V}$, there is a residual R:

$$R(t) = \left(\mathbf{A}(t) + \sum_{k=1}^{2L} \mathbf{N}_k u_k(t) \right) \mathbf{V} C(t) + \mathbf{B} u(t) - \mathbf{V} \dot{C}(t). \tag{3.3}$$

If a matrix $\mathbf{W} \in \mathbb{R}^{3N_s \times k}$ is found such that its columns span a subspace orthogonal to the residual:

$$\mathbf{W}^T R(t) = 0 \text{ for } t \in [0, T]. \tag{3.4}$$

Then the following reduced system can be defined:

$$\mathbf{W}^T \mathbf{V} \dot{C}(t) = \mathbf{W}^T \left(\mathbf{A}(t) + \sum_{k=1}^{2L} \mathbf{N}_k u_k(t) \right) \mathbf{V} C(t) + \mathbf{W}^T \mathbf{B} u(t). \tag{3.5}$$

The columns of $\mathbf{V}$ and $\mathbf{W}$ are called respectively the trial and test basis. If $\mathbf{W}^T \mathbf{V} \in \mathbb{R}^{k \times k}$ is non-singular, both sides in equation can be multiplied with its inverse to find the reduced-order model:

$$\dot{C}(t) = \left(\mathbf{W}^T \mathbf{V} \right)^{-1} \mathbf{W}^T \left(\mathbf{A}(t) + \sum_{k=1}^{2L} \mathbf{N}_k u_k(t) \right) \mathbf{V} C(t) + \left(\mathbf{W}^T \mathbf{V} \right)^{-1} \mathbf{W}^T \mathbf{B} u(t).$$

$$= \left(\tilde{\mathbf{A}}(t) + \sum_{k=1}^{2L} \tilde{\mathbf{N}}_k u_k(t) \right) C(t) + \tilde{\mathbf{B}} u(t).$$

$$\tag{3.6}$$

with:

$$\tilde{\mathbf{A}}(t) = \left(\mathbf{W}^T\mathbf{V}\right)^{-1}\mathbf{W}^T\mathbf{A}(t)\mathbf{V} \in \mathbb{R}^{k \times k},$$
$$\tilde{\mathbf{N}}_k = \left(\mathbf{W}^T\mathbf{V}\right)^{-1}\mathbf{W}^T\mathbf{N}_k\mathbf{V} \in \mathbb{R}^{k \times k}, \qquad (3.7)$$
$$\tilde{\mathbf{B}} = \left(\mathbf{W}^T\mathbf{V}\right)^{-1}\mathbf{W}^T\mathbf{B} \in \mathbb{R}^{k \times 2L}.$$

The matrices $\mathbf{V}$ and $\mathbf{W}$ uniquely define a projection $\mathbf{\Pi}$ with $\mathcal{R}(\mathbf{\Pi}) = \mathcal{V}$ and $\mathcal{N}(\mathbf{\Pi}) = \mathcal{W}^{\perp}$:

$$\mathbf{\Pi} = \mathbf{V}\left(\mathbf{W}^T\mathbf{V}\right)^{-1}\mathbf{W}^T. \qquad (3.8)$$

If all trajectories of $\hat{M}$ in the full order model are invariant under the projection of $\mathbf{\Pi}$:

$$\mathbf{\Pi}M(t) = M(t), \text{ for } t \in [0, T], \qquad (3.9)$$

then determining the response of $C = \mathbf{V}^T\hat{M}$ through equation (3.6) is equivalent to determining the response of $\hat{M}$ through the full-order system.

If $\mathbf{V} = \mathbf{W}$, the projection in equation (3.5) simplifies to $\mathbf{VV}^T$ which is an orthogonal projection called the *Galerkin projection*. If $\mathbf{V} \neq \mathbf{W}$, the projection is oblique and called the *Petrov-Galerkin projection*.

3.1.1 Projection methods

Determining the relevant subspaces $\mathcal{V}$ and $\mathcal{W}$ for a given dynamical system is not trivial. The model reduction method should fulfil a number of different requirements:

- The ROM should approximate the FOM for a given error criterion.

- The ROM should preserve most system properties from the FOM, such as stability.

- The ROM should be computationally affordable.

- The method should be applicable for a wide range of parameters of the full order model.

In general, two types of methods are distinguished. The singular value decomposition (SVD) methods are based on the observed data for some trial parameters. Best known is the *proper orthogonal decomposition* (POD) method, explained in the following section. Krylov subspace methods attempt to find a ROM which accurately describe the impulse response of the original system by doing so-called *moment-matching*.

POD method

Here the matrix $\mathbf{V}$ is constructed such that its columns approximate some relevant data set, whose elements are called *snapshots*. In this case, the snapshots are the magnetization response, $M(t)$, at different time point and for a test set of p different radiofrequency pulses $\left(u^{(i)}\right)_{1 \leq i \leq p}$. It is assumed that a matrix $\mathbf{V}$ constructed in this manner will also contain the magnetization response of a much larger set of RF pulses. Defining the magnetization response of an RF pulse $u^{(i)}$ as $M^{u^{(i)}}(t)$, and concatenating all time points:

$$\mathbf{M}^{u^{(i)}} := \left[M^{u^{(i)}}(t_1), M^{u^{(i)}}(t_2), \ldots, M^{u^{(i)}}(t_{N_t})\right] \in \mathbb{R}^{3N_s \times N_t}, \qquad (3.10)$$

where the time points t_i follow the discretization of the RF pulses given in equation (1.37). The *snapshot matrix* $\mathbf{M}$ is created by adding the responses of the different pulses:

$$\mathbf{M} := \left[\mathbf{M}^{u^{(1)}}, \mathbf{M}^{u^{(2)}}, \ldots, \mathbf{M}^{u^{(p)}}\right] \in \mathbb{R}^{3N_s \times pN_t}. \qquad (3.11)$$

The POD method attempts to find a basis $\mathbf{V} \in \mathbb{R}^{3N_s \times k}$, for preset k, which minimizes the 2-norm error between the trajectories and their Galerkin projection $\mathbf{\Pi} = \mathbf{V}^T \mathbf{V}$:

$$\mathbf{V} = \underset{\mathbf{X} \in \mathbb{R}^{3N_s \times k}}{\arg\min} \sum_{j=1}^{p} \sum_{i=1}^{N_t} \|M^{u^{(j)}}(t_i) - \sum_{l=1}^{k} X_l X_l^T M^{u^{(j)}}(t_i)\|_2^2, \qquad (3.12)$$

with $\mathbf{X} = [X_1, X_2, \ldots, X_k] \in \mathbb{R}^{3N_s \times k}$. The solution $\mathbf{V}$ is found using the SVD of the matrix $\mathbf{M}$:

$$\mathbf{M} = \mathbf{U}\mathbf{\Sigma}\mathbf{W}^T, \qquad (3.13)$$

where $\mathbf{U} \in \mathbb{R}^{3N_s \times 3N_s}$ is an orthogonal matrix, $\mathbf{W} \in \mathbb{R}^{pN_t \times pN_t}$ is also orthogonal and $\mathbf{\Sigma} \in \mathbb{R}^{3N_s \times pN_t}$ is a diagonal matrix. The singular values $(\sigma_j)_{1 \leq j \leq \min(3N_s, pN_t)}$ of $\mathbf{M}$ are sorted in descending order on the diagonal of $\mathbf{\Sigma}$. It is proven in [2] that the first k columns of $\mathbf{U}$ solve equation (3.12), and furthermore, in this case:

$$\sum_{j=1}^{p} \sum_{i=1}^{N_t} \|M^{u^{(j)}}(t_i) - \sum_{l=1}^{k} V_l V_l^T M^{u^{(j)}}(t_i)\|_2^2 = \sum_{j=k+1}^{\min(3N_s, pN_t)} \sigma_j^2. \qquad (3.14)$$

Equation (3.14) is important since for given k it directly quantifies the minimal error in approximating the snapshot matrix $\mathbf{M}$. Although this does not necessarily indicates if the columns of $\mathbf{V}$ accurately approximate the response of any RF pulse, it does give an indication when this is definitely not the case.

Krylov method

The following method attempts to accurately describe the impulse response of the system by so-called moment-matching. It takes an intermediate step to explain what the moments of the Bloch equation are. For this intermediate step, it is supposed that there is a single real-valued radiofrequency pulse and constant gradient, such that the shifted system from equation (2.7) becomes:

$$\dot{\hat{M}}(t) = \mathbf{A}\hat{M}(t) + u(t)\mathbf{N}\hat{M}(t) + Bu(t), \qquad (3.15)$$

With $u(t) \in \mathbb{R}$ and $B \in \mathbb{R}^{3N_s}$. In this case the response of the magnetization can be described using the *Volterra series* representation (see [5], [19]):

$$\hat{M}(t) = \sum_{k=1}^{\infty} \int_0^t \int_0^{t_1} \dots \int_0^{t_{k-1}} h(t_1, \dots, t_k) u(t - t_1 - \dots - t_k) \dots$$

$$u(t - t_k)\mathrm{d}t_k \dots \mathrm{d}t_1, \quad (3.16)$$

where h is the degree-k kernel that characterizes the response:

$$h(t_1, \dots, t_k) := e^{\mathbf{A}t_k}\mathbf{N} \dots e^{\mathbf{A}t_2}\mathbf{N}e^{\mathbf{A}t_1}B \in \mathbb{R}^{3N_s}. \qquad (3.17)$$

Equation (3.16) generalizes the magnetization response given in equation (1.45) for a time-varying RF pulse. The kernel can be represented by a degree-k transfer function H by applying the multi-variable Laplace transform:

$$H(s_1, \dots, s_k) := (s_k\mathbf{I} - \mathbf{A})^{-1}\mathbf{N} \dots (s_2\mathbf{I} - \mathbf{A})^{-1}\mathbf{N}(s_1\mathbf{I} - \mathbf{A})^{-1}B. \qquad (3.18)$$

Which can be written using expansion point $\eta_i \in \mathbb{R}$:

$$H(s_1, \dots, s_k) = \sum_{l_k=1}^{\infty} \dots \sum_{l_1=1}^{\infty} m(l_1, \dots, l_k) \cdot (s_1 - \eta_1)^{l_1-1} \dots (s_k - \eta_k)^{l_k-1}. \quad (3.19)$$

with:

$$m(l_1, \dots, l_k) = (-1)^k(\mathbf{A} - \eta_k\mathbf{I})^{-l_k}\mathbf{N} \dots (\mathbf{A} - \eta_2\mathbf{I})^{-l_2}\mathbf{N}(\mathbf{A} - \eta_1\mathbf{I})^{-l_1}B. \quad (3.20)$$

For the following, it is assumed that the kth expansion point η_k in (3.19) is the same for all transfer functions H of degree k and higher.

The multimoment-matching method attempts to find a reduced system as in equation (3.6), with moments $\hat{m}$ that match the moments of the original system:

$$m(l_1, \dots, l_k) = \mathbf{V}\hat{m}(l_1, \dots, l_k). \qquad (3.21)$$

The matrix $\mathbf{V}$ is designed such that for given values of $r, q \in \mathbb{N}$, equation (3.21) holds for $1 \leq k \leq r$, and $1 \leq l_i \leq q$.

For linear systems, the book of Antoulas [2] explains how to create the matrix $\mathbf{V}$ using Krylov techniques such that equation (3.21) holds. For non-linear systems, creating such a basis is more difficult but fortunately, in recent years a number of different articles on Krylov methods for the bilinear case have been published. For the single-input/single-output (SISO) case a good explanation is found in [5]. An extension to multiple-input/multiple-output (MIMO) is based on the same ideas but greatly increases bookkeeping and notation [14]. Here, both methods are mixed and a single-input/multiple-output (SIMO) system is discussed. Note that the multichannel RF case and time dependent gradients belong to the MIMO type.

In [5] it is proven that equation (3.21) holds for $1 \leq k \leq r$, and $1 \leq l_i \leq q$, if $\mathbf{V}$ equals:

$$\text{span}\{\mathbf{V}\} = \text{span}\left\{\bigcup_{k=1}^{r}\text{span}\{\mathbf{V}^{(k)}\}\right\}, \tag{3.22}$$

where the matrices $\mathbf{V}^{(k)}$ are constructed such that:

$$\text{span}\{\mathbf{V}^{(1)}\} = \mathcal{K}_q((\mathbf{A} - \eta_1 I)^{-1}, (\mathbf{A} - \eta_1 I)^{-1}B), \tag{3.23}$$

and for $k = 2, \ldots, r$:

$$\text{span}\{\mathbf{V}^{(k)}\} = \mathcal{K}_q((\mathbf{A} - \eta_k I)^{-1}, (\mathbf{A} - \eta_k I)^{-1}\mathbf{N}\mathbf{V}^{(k-1)}). \tag{3.24}$$

Note that $(\mathbf{A} - \eta_k I)^{-1}\mathbf{N}\mathbf{V}^{(k-1)}$ is a matrix and thus equation (3.24) defines a block Krylov subspace, for which a basis can be constructed using Arnoldi iteration. The matrix $\mathbf{V}$ can have up to $\sum_{k=1}^{r} q^k$ vectors if all vectors in the Krylov spaces are linearly independent. Furthermore, in [5] different choices are presented for the matrix $\mathbf{W}$ in (3.9) that either guarantee preservation of stability or matching more modes.

To construct $\mathbf{V}$ in the multichannel case and with time-dependent gradients, the gradient waveforms in each direction are considered as an extra control u_k. Since in this case the magnetization is dependent on multiple time-dependent control variables, the formula for the response in equation (3.16) becomes more complicated. The matrix $\mathbf{V}$ is then calculated by replacing B in equation (3.23) by $\mathbf{B} \in \mathbb{R}^{3N_s \times 2L}$ defined in equation (2.7), and

by replacing equation (3.24) with:

$$\text{span}\{\mathbf{V}^{(k)}\} = \bigcup_{i=1}^{2L+3} \mathcal{K}_q((\mathbf{A} - \eta_k I)^{-1}, (\mathbf{A} - \eta_k I)^{-1}\mathbf{N}_i\mathbf{V}^{(k-1)}). \qquad (3.25)$$

The matrix $\mathbf{V}$ constructed in this manner has a maximum of $\sum_{k=1}^{r} ((2L+3)q)^k$ columns.

3.1.2 Application to the Bloch equation

Both a numerical implementation of the POD as the moment-matching method for spatial reduction of the Bloch equations were attempted, but the results obtained were not encouraging in terms of computational efficiency. A working reduction technique might be possible, and in the following some remarks are made on the difficulties and possible solution strategies.

Projection

The matrices of the original system as given in equation (2.1) are tridiagonal and sparse. When projecting the matrices in equation (3.5), the sparsity and structure is lost, and the response can no longer be determined by rotating and decaying individual voxels. In order to make response calculation of the reduced system efficient compared with the full-order system, the Bloch equations dynamic should be contained in a very low-dimensional subspace. It is not clear whether this is the case.
Using a Galerkin projection, the ROM still retains some structure since projecting a skew-symmetric matrix gives a matrix which is still skew-symmetric and its exponential equals a high-dimensional rotation. However, the rotation formulas used in Chapter 1 do not apply to high-dimensional rotations. A possible strategy for this case can be found in [9].

Singular values of snapshot matrix

To apply the POD method to the Bloch equation, a snapshot matrix (3.11) of the magnetization response has to be created. Since calculating the snapshots requires the response simulation of the Bloch equation for each position, computational efficiency can only be gained if the Bloch equation has to be solved repeatedly. Two cases for the snapshot matrix were compared: the response of a two-state refocusing pulse as designed by the LCLTA method and that of the optimal control solution. The same parameters as in the Sections 2.3.4 are used ($N_t = 603$, $N_s = 7120$).
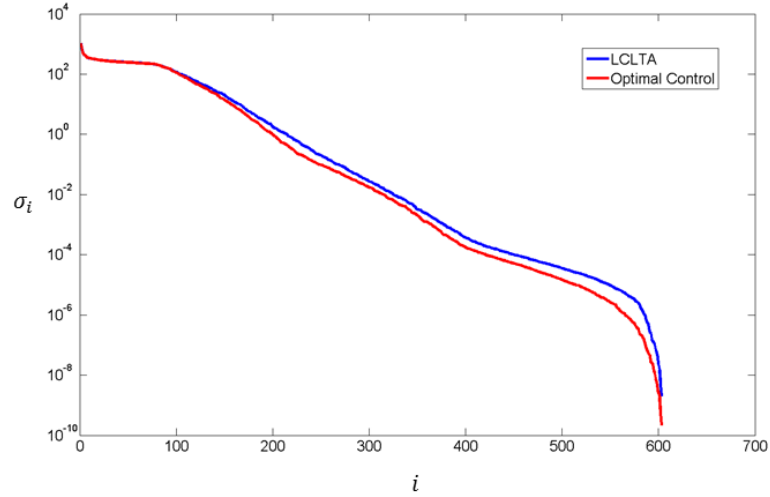
Figure 3.1: Decay of the singular values for the magnetization response of the LCLTA refocusing pulse.

The decay of the singular values for both cases are shown on a logarithmic scale in Figure 3.1. The singular values decrease exponentially. Based on equation (3.14) it is expected that the reduced bases will require in the order of hundreds of vectors to accurate approximate the range of different magnetization responses. In practice it is possible that a smaller set leads to a good result and additional research is suggested.

Eigenvalues

Constructing the matrix $\mathbf{V}$ requires knowledge of which magnetization states can be considered to be "important" for the representation of all magnetization states. The eigenvalues of the matrix $\mathbf{\Omega}(t)$ for $t \in [0, T]$, as given in equation (2.5), can give some insight. For instance, magnetization states are damped over time if they are associated with eigenvalues having a relatively small real part. These states are then less important to include in $\mathbf{V}$.

The matrix $\mathbf{\Omega}(t)$ is time-dependent, and the eigenvalues change over time. Note that the matrix $\mathbf{\Omega}(t)$ consists of blocks, $\mathbf{\Omega}(r_j, t) \in \mathbb{R}^{3\times 3}$, of the Bloch equation matrix for different voxels r_j, as defined in (1.36). The spectrum of $\mathbf{\Omega}(t)$ thus equals the union of the spectra of $\mathbf{\Omega}(r_j, t)$ of all positions.

The eigenvalue distribution for different time points, and thus values of $B_1(t)$ and $G(t)$, look similar and an example is shown in Figure 3.2. Each matrix
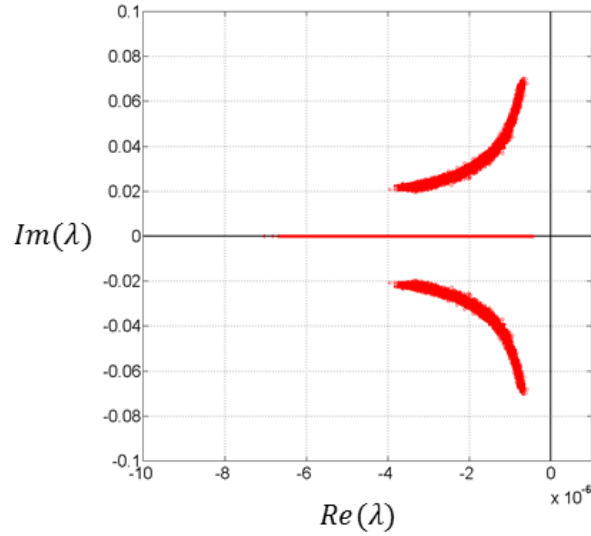
Figure 3.2: Example eigenvalues of the Bloch matrix $\mathbf{\Omega}(t)$, for given $t \in [0, T]$.

$\mathbf{\Omega}(r_j, t)$ has one real eigenvalue and a pair of complex conjugate eigenvalues, and this distribution is clearly shown in the figure. Note that there is a large scale difference between the x- and y-axis, and all eigenvalues are close to being purely imaginary. Pure imaginary eigenvalues are associated with rotations, as shown in Section 1.1.1 Here it was also stated that the rotation was the dominant effect in the magnetization response. From Figure 3.2, it is not directly clear which eigenvalues and associated magnetization states are essential for the matrix $\mathbf{V}$. However, an important observation is that there are no dominant eigenvalues that either have relatively large or small real part. Thus selecting states on this criterion is not directly applicable.

3.2 Temporal domain

In contrast to the spatial domain, using temporal domain reduction for MR pulse design has been actively studied in recent years. In this section, two recent methods from literature ([3], [10]) will be discussed that allow for fast response calculation in different cases. Furthermore, a new idea for reducing the dimension of the problem in the time domain will be illustrated and tested.

3.2.1 Previous literature

Two existing methods are discussed in this section. In Section 3.2.1, the method of Balac and Chupin [3] is shown, which allows for fast magnetization response calculation when the RF field has a perturbation term. Section 3.2.1 discusses the method of Grissom et al [10], which extents the LCLTA method from Section 1.3.3 and is applied to pulse design through optimal control.

Truncated series expansion

In Section 1.2.1, it was shown that the RF field, B_1, rotates at the Larmor frequency ω_0 in the laboratory frame. In the following, the RF field in the laboratory frame is described by the formula:

$$\begin{pmatrix} B_x(r,t) \\ B_y(r,t) \end{pmatrix} = b_1 \begin{pmatrix} \cos(\omega_0 t) \\ -\sin(\omega_0 t) \end{pmatrix} + \begin{pmatrix} v_1(r) & w_1(r) \\ v_2(r) & w_2(r) \end{pmatrix} \begin{pmatrix} \cos(\omega_0 t) \\ \sin(\omega_0 t) \end{pmatrix}. \quad (3.26)$$

The first term on the right equals the RF field from Section 1.2.1 with constant amplitude, b_1, and aligned with the x-axis at $t = 0$. The second term is the so-called *perturbed RF field*. The perturbed RF field has the same frequency as the RF field since the latter typically induces the former by so-called *Eddy currents*. All terms in equation (3.26) are real-valued.

In the following, the gradient and field inhomogeneity effects are ignored: $G(t) = \Delta B_0 = 0$ for $t \in [0, T]$. Furthermore, the response is calculated for a single position and the spatial dependence is dropped from the equations. The matrix of the Bloch equation in the rotating frame, defined in (1.35), becomes:

$$\mathbf{\Omega}'(t) = \begin{pmatrix} -1/T_2 & 0 & -\omega_a(t) \\ 0 & -1/T_2 & \omega_b(t) \\ \omega_a(t) & -\omega_b(t) & -1/T_1 \end{pmatrix}. \quad (3.27)$$

with:

$$\omega_a(t) := \gamma B_y(t) \cos(\omega_0 t) + \gamma B_x(t) \sin(\omega_0 t),$$
$$\omega_b(t) := \gamma B_x(t) \cos(\omega_0 t) - \gamma B_y(t) \sin(\omega_0 t). \tag{3.28}$$

Note that by definition in (3.26), the functions $B_x(t)$ and $B_y(t)$ a periodic with frequency ω_0. It follows that the matrix $\mathbf{\Omega}'$ has continuous and periodic coefficients, and can be written as:

$$\mathbf{\Omega}'(t) = \mathbf{\Omega}_{-2} e^{-2i\omega_0 t} + \mathbf{\Omega}_0 + \mathbf{\Omega}_2 e^{2i\omega_0 t}. \tag{3.29}$$

where:

$$\mathbf{\Omega}_0 := \begin{pmatrix} -1/T_2 & 0 & -\omega_a^{(0)} \\ 0 & -1/T_2 & \omega_b^{(0)} \\ \omega_a^{(0)} & -\omega_b^{(0)} & -1/T_1 \end{pmatrix}, \tag{3.30}$$

with $\omega_a^{(0)} := \frac{1}{2}\gamma(v_2 + w_1)$, $\omega_b^{(0)} := \frac{1}{2}\gamma(2b_1 + v_1 - w_2)$, and:

$$\mathbf{\Omega}_2 = \bar{\mathbf{\Omega}}_{-2} := \begin{pmatrix} 0 & 0 & -\omega_a^{(2)} \\ 0 & 0 & \omega_b^{(2)} \\ \omega_a^{(2)} & -\omega_b^{(2)} & 0 \end{pmatrix}, \tag{3.31}$$

with $\omega_a^{(2)} := -\frac{1}{4}\gamma(w_1 - v_2 + i(v_1 + w_2))$, $\omega_b^{(2)} := \frac{1}{4}\gamma(v_1 + w_2 + i(v_2 - w_1))$.

Decomposing the magnetization response as:

$$M(t) = \sum_{k \in \mathbb{Z}} m_k(t) e^{2ik\omega_0 t}. \tag{3.32}$$

and substituting the infinite sequence into the Bloch equation (1.36), taking note of equation (3.29), the following infinite sequence of differential equations is obtained:

$$\forall k \in \mathbb{Z} \quad \begin{cases} \frac{d}{dt} m_k(t) = \sum_{j=-1}^{1} \mathbf{\Omega}_{2j} m_{k-j}(t) - 2ik\omega_0 m_k(t) + \delta_k M_0/T_1, \\ m_k(0) = \delta_k M_0, \end{cases} \tag{3.33}$$

with $\delta_k = 1$ for $k = 0$, and zero elsewhere. The sequence of differential equations (3.33) has an unique solution and for each time point t the sequence $(m_k(t))_{k \in \mathbb{Z}}$ converges quickly to zero if k goes to $\pm\infty$ (see [3]).

The algorithm is based on concatenating the coefficients in equation (3.32) and (3.33) for the first N terms:

$$m^{[N]}(t) := \begin{pmatrix} m_N(t) \\ m_{N-1}(t) \\ m_{N-2}(t) \\ \vdots \\ m_{-N}(t) \end{pmatrix} \in \mathbb{R}^{3(2N+1)}, \tag{3.34}$$

and:

$$\boldsymbol{\Omega}^{[N]} := \begin{pmatrix} \boldsymbol{\Omega}_0(N) & \boldsymbol{\Omega}_{-2} & & & \\ \boldsymbol{\Omega}_2 & \boldsymbol{\Omega}_0(N-1) & \boldsymbol{\Omega}_{-2} & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \boldsymbol{\Omega}_{-2} \\ & & & \boldsymbol{\Omega}_2 & \boldsymbol{\Omega}_0(-N) \end{pmatrix} \in \mathbb{R}^{3(2N+1)\times 3(2N+1)}, \tag{3.35}$$

with $\boldsymbol{\Omega}_0(k) = \boldsymbol{\Omega}_0 + 2ik\omega_0 I$. The coefficients of the solution at time $t = T$, are given by:

$$m^{[N]}(T) = e^{T\boldsymbol{\Omega}^{[N]}} m^{[N]}(0) + \left(\int_0^T e^{(T-s)\boldsymbol{\Omega}^{[N]}} ds \right) M_0^{[N]}. \tag{3.36}$$

Here $M_0^{[N]}$ is defined using the sequence (3.34) with $m_k(t) = \delta_k M_0$. The algorithm calculates the solution by diagonalizing $\boldsymbol{\Omega}^{[N]} = \mathbf{P}\mathbf{D}\mathbf{P}^{-1}$, and moving the matrices $\mathbf{P}$ and $\mathbf{P}^{-1}$ out of the exponential such that the we get the exponential of a diagonal matrix. The matrices $\mathbf{P}$ and $\mathbf{D}$ can be found by calculating the eigenvalues and vectors of $\boldsymbol{\Omega}^{[N]}$, since:

$$\boldsymbol{\Omega}^{[N]}\mathbf{P} = \mathbf{P}\mathbf{D}. \tag{3.37}$$

Hence the ith diagonal element of $\mathbf{D}$ is the eigenvalue of $\boldsymbol{\Omega}^{[N]}$ with eigenvector equal to the ith column of $\mathbf{P}$. By the block tridiagonal structure of $\boldsymbol{\Omega}^{[N]}$, there are fast algorithms for the computation of the eigenvalues and vectors, allowing fast evaluation of (3.36).

The resulting algorithm to compute $m^{[N]}(T)$ was compared in [3] with the ODE45 solver from Matlab applied to equation (1.35). This was done in the case without RF perturbation where the exact solution is known. Even for $N = 0$, the method based on equation (3.36) is an order of magnitude faster and slightly more accurate than the ODE45 solver.

There are a number of advantages of this method. First, it allows for taking arbitrary large time steps when determining the magnetization response. In practice this means moving from the $\mathcal{O}(10^{-6}s)$ on which de RF and gradient waveforms are defined, such as in the Bloch simulator used in the previous chapters, to $\mathcal{O}(10^{-3}s)$ in which the RF pulse duration is contained. Second, there is a lot of freedom in determining the field inhomogeneities in equation (3.26). Third, the method includes relaxation rates. Finally the matrices $\mathbf{P}$ and $\mathbf{D}$ have to be determined only once for each value of b_1, supposing the field inhomogeneities stay the same.

However, the restriction to constant RF and gradient waveforms makes the method not directly applicable to a general setting. If the method can be extended by including time-dependent RF and gradient waveforms, it might be well suited for increasing the convergence speed of the optimal control method.

Small-perturbation approximation

In the method by Grissom [10], the relaxation effects are ignored and the magnetization response is considered in the spin-domain using equation (1.52), as in the LCLTA method. The small-perturbation method assumes the RF pulses $b_1^{(l)}(t)$ are perturbed with small pulses $b_1'^{(l)}(t)$, leading to a perturbation $B_1'(r,t)$ of the effective RF field. In consequence, it follows from equation (1.52) that the parameters α and β, defining the magnetization response in the spin-domain, are perturbed with $\alpha', \beta' \in \mathbb{C}$:

$$\frac{d}{dt} \begin{pmatrix} \beta + \beta' \\ \alpha + \bar{\alpha}' \end{pmatrix} = \frac{i\gamma}{2} \begin{pmatrix} G(t) \cdot r & \bar{B}_1(r,t) + \bar{B}_1'(r,t) \\ B_1(r,t) + B_1'(r,t) & -G(t) \cdot r \end{pmatrix} \begin{pmatrix} \beta + \beta' \\ \bar{\alpha} + \bar{\alpha}' \end{pmatrix}. \tag{3.38}$$

Substituting equation (1.52) into the above equation gives:

$$\frac{d}{dt} \begin{pmatrix} \beta' \\ \bar{\alpha}' \end{pmatrix} = \frac{i\gamma}{2} \begin{pmatrix} G(t) \cdot r & \bar{B}_1(r,t) + \bar{B}_1'(r,t) \\ B_1(r,t) + B_1'(r,t) & -G(t) \cdot r \end{pmatrix} \begin{pmatrix} \beta + \beta' \\ \bar{\alpha} + \bar{\alpha}' \end{pmatrix}$$
$$- \frac{i\gamma}{2} \begin{pmatrix} G(t) \cdot r & \bar{B}_1(r,t) \\ B_1(r,t) & -G(t) \cdot r \end{pmatrix} \begin{pmatrix} \beta \\ \bar{\alpha} \end{pmatrix}. \tag{3.39}$$

This equals the following coupled differential equations:

$$\frac{d}{dt} \beta' = \frac{i\gamma}{2} \left((G(t) \cdot r)\beta' + \bar{B}_1(r,t)\bar{\alpha}' + \bar{B}_1'(r,t)\bar{\alpha} + \bar{B}_1'(r,t)\bar{\alpha}' \right) \tag{3.40}$$

$$\frac{d}{dt} \bar{\alpha}' = \frac{i\gamma}{2} \left(-(G(t) \cdot r)\bar{\alpha}' + \bar{B}_1'(r,t)\beta + \bar{B}_1(r,t)\beta' + \bar{B}_1'(r,t)\beta' \right). \tag{3.41}$$

Equations (3.40) and (3.41) are decoupled by supposing that some of the terms are relatively small (see [10]), such the equations are approximated by:

$$\frac{d}{dt}\beta' = \frac{i\gamma}{2}\left((G(t) \cdot r)\beta' + \bar{B}_1'(r,t)\bar{\alpha}\right) \tag{3.42}$$

$$\frac{d}{dt}\bar{\alpha}' = \frac{i\gamma}{2}\left(-(G(t) \cdot r)\bar{\alpha}' + \bar{B}_1'(r,t)\beta\right). \tag{3.43}$$

With solutions:

$$\beta'(r,T) = \frac{i\gamma}{2}\sum_{l=1}^{L}\bar{s}_l(r)\int_0^T \bar{b}_1'^{(l)}\bar{\alpha}(r,t)e^{-\frac{i}{2}r\cdot k(t)}dt, \tag{3.44}$$

$$\alpha'(r,T) = \frac{i\gamma}{2}\sum_{l=1}^{L}s_l(r)\int_0^T b_1'^{(l)}\beta(r,t)e^{-\frac{i}{2}r\cdot k(t)}dt. \tag{3.45}$$

Similar as to the LCLTA method, discretizing the integral and solving for different positions at the same time gives a linear system, ignoring the discretization error:

$$\beta' = \sum_{l=1}^{L}\text{diag}(\bar{s}_l)\mathbf{B}\bar{u}_l', \qquad \text{with: } \mathbf{B}_{ij} = \frac{i\gamma}{2}\Delta t\bar{\alpha}_{ij}e^{-\frac{i}{2}r_i\cdot k(t_j)}, \tag{3.46}$$

$$\alpha' = \sum_{l=1}^{L}\text{diag}(s_l)\mathbf{A}\bar{u}_l, \qquad \text{with: } \mathbf{A}_{ij} = \frac{i\gamma}{2}\Delta t\beta_{ij}e^{\frac{i}{2}r_i\cdot k(t_j)}. \tag{3.47}$$

The matrices $\mathbf{A}$ and $\mathbf{B}$ describe the effect of perturbations in b_1 on α and β. The time domain reduction is found in the fast determination of the effect of small changes in the RF pulse on the magnetization response.

In order to quickly calculate the perturbation effect, a domain reduction is performed on the rows of the matrices $\mathbf{A}$ and $\mathbf{B}$. As shown on the right of equations (3.46) and (3.47) these equal the trajectory of α or β for a given position weighted by the gradient effect.
If a low-rank basis can be found for the rows, the RF perturbation effect only has to be calculated for this basis, reducing computation time. However the singular values of the matrices rows of $\mathbf{A}$ and $\mathbf{B}$ decay slowly, which would require a large number vectors to accurately describe all trajectories.

A solution is found by looking at the elements of the matrices in the so-called *gradient frame*, which defines matrices $\hat{\mathbf{A}}$ and $\hat{\mathbf{B}}$ by:

$$\hat{\mathbf{A}}_{ij} = e^{\frac{i\gamma}{2}r_i\cdot\int_0^{t_j}G(\tau)d\tau}\mathbf{A}_{ij}, \qquad \hat{\mathbf{B}}_{ij} = e^{\frac{i\gamma}{2}r_i\cdot\int_0^{t_j}G(\tau)d\tau}\mathbf{B}_{ij}. \tag{3.48}$$

The singular values of $\hat{\mathbf{A}}$ and $\hat{\mathbf{B}}$ decay fast and we can use the SVD to create a low rank approximation. Converting back to the rotating reference frame by inverting equations (3.48). Note that in this case the reduced basis describes the different temporal trajectories while in the spatial reduction section we attempted to create a basis of the spatial domain.

The algorithm now alternates between simulating the response of the baseline pulses $b_1^{(l)}$, and minimizing the cost function over the perturbations, taking the minimum as new baseline. The method showed that compared to the conventional optimal control method as shown in the previous chapter, the design time was reduced almost 30 fold.

3.2.2 Piecewise constant representation of RF waveforms

The RF pulses used in the previous chapters were so-called *fully-sampled*, with a time interval size in the order of microseconds. For a typical RF pulse of a few milliseconds, this gives an order of 10^3 intervals. As shown in Chapter 1, the simulation time of the response is linear in the number of time intervals. To decrease the simulation time, the goal of this section is to design RF pulses based on the fully-sampled pulse, such that:

1. The RF pulse is piecewise constant on much fewer intervals than the fully sampled pulse,

2. The response of this RF pulse approximates that of the fully-sampled pulse for a large number of parameters: T_1, T_2, B_1 and ΔB_0.

Such an RF pulse will be called a *reduced pulse*. Substituting fully sampled pulses with reduced pulses greatly decreases simulation time, especially for sequences of a repeated RF pulse. Examples of methods that can benefit from the reduced pulses are for instance the so-called magnetic resonance fingerprinting (MRF) [15], and the new MR-STAT method [20].

Method

In the following, the focus is for simplicity on the magnetization response to a single real-valued RF pulse. The method can easily be extended to the multichannel and complex RF pulses. Two types of RF pulse shapes are compared: one defined by a Gaussian function and one by a sinc function:

$$\text{gauss}(x) := e^{-x^2}, \qquad \qquad \text{sinc}(x) := \frac{\sin(\pi x)}{\pi x}. \qquad (3.49)$$

The motivation for these pulse shapes is that for small flip angles the magnetization response of an object approximates the Fourier transform of the RF pulse. Since the Fourier transform of a Gaussian is again Gaussian, and that of sinc function is a rectangle, both are used for slice selective excitation. These functions have unlimited support and need to be truncated. The Gaussian function can be truncated once the amplitude is lower than a given tolerance level. For the sinc function, two cases were tested: truncation after three zero crossings, giving a total of four side lobes (*sinc(4)*), and after five zero crossings, giving eight side lobes (*sinc(8)*). The three pulses were scaled to have a length of 1 ms and a flip angle of $\pi/2$, and sampled in 2500 intervals as shown in Figure 3.3.
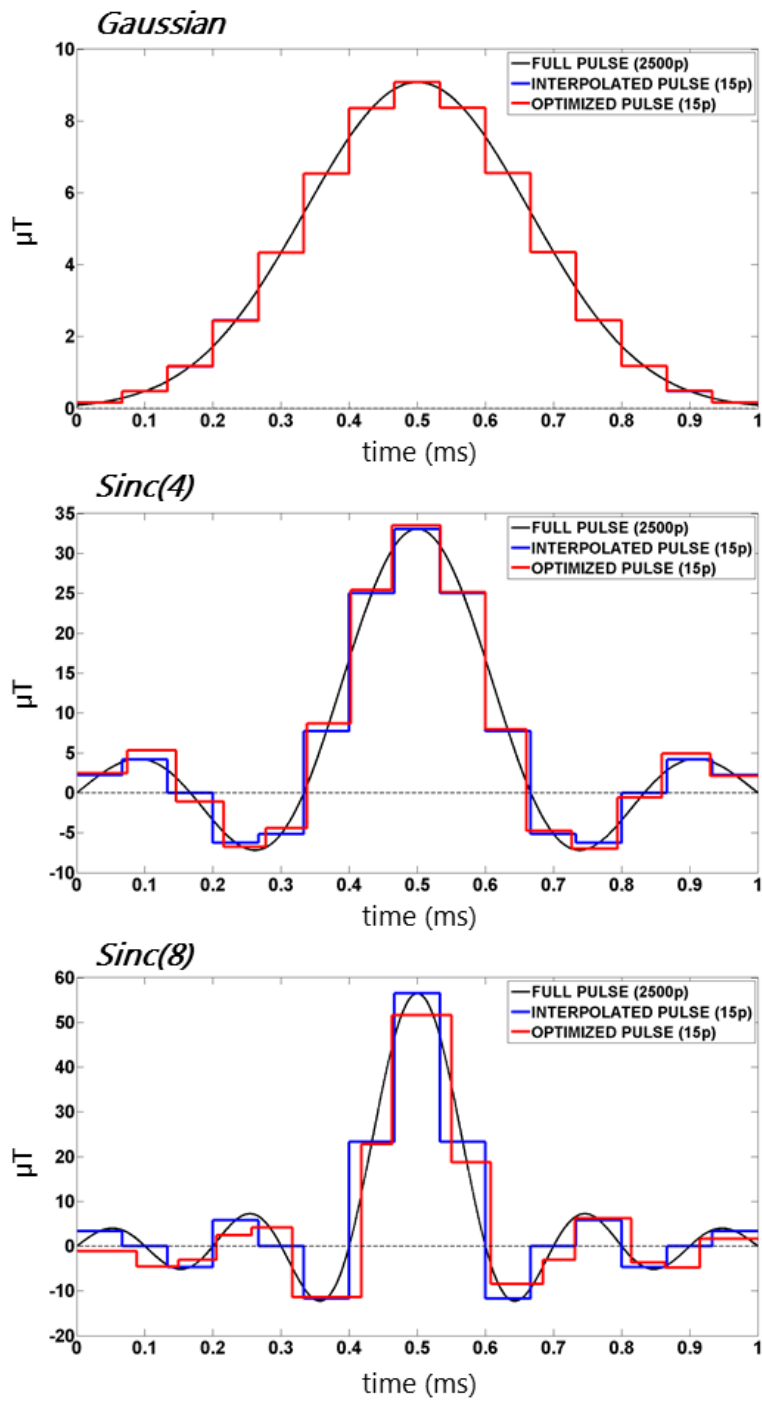
Figure 3.3: The fully-sampled, interpolated and optimized RF pulses for three pulse shapes: Gaussian, sinc with four side lobes, and sinc with eight side lobes.

87

A test set of 2000 points is generated with random parameters taken from uniform distributions with ranges shown in Table 3.1. For the test set, the response of the fully sampled pulse is calculated and the magnetization profile at the end of the pulse time T is chosen as the desired response D. The designed reduced pulse should approximate D as measured by the relative error of the magnetization at the end of the pulse:

$$\varphi(M(T)) = \frac{\|M(T) - D\|_2^2}{\|D\|_2^2}. \tag{3.50}$$

Suppose the designed pulse is piecewise constant on $n_t = 15$ time intervals. If the pulse length T is distributed in n_t intervals of equals length, the grid points equal:

$$t_i := \frac{T}{n_t}i = i\Delta t, \text{ for } i \in \{0, 1, \ldots, n_t\}. \tag{3.51}$$

A natural method to define a pulse on these intervals, approximating the fully-sampled pulse in shape, is interpolation. The reduced pulse $\tilde{b}_1$ is defined on each interval $[t_i, t_{i+1})$, by setting the pulse equal to the value of the fully sampled pulse in the midpoint:

$$\tilde{b}_1(t) := b_1(t_i + \Delta t/2), \quad \text{for } i \in \{0, 1, \ldots, n_t - 1\} \text{ and } t \in [t_i, t_{i+1}). \tag{3.52}$$

For smooth pulse shapes, the interpolated pulse closely approximates the shape of the fully sampled pulse and we expect that also the magnetization response of both cases is similar.

A better method for creating a reduced pulse would be improving the interpolated pulse. Here the optimization toolbox of MATLAB was used which varied the length of each interval and its amplitude to find the reduced pulse that minimizes (3.50). The total pulse time was constrained to be equal to T. Note that the optimized pulse has to be calculated only once for a given fully-sampled pulse and can then be used for different applications requiring simulations.

T_1	600–1200 ms
T_2	50–130 ms
ΔB_0	-500–500 Hz
$B_1^+, ampl.$	0.1–1
$B_1^+, phase$	0–2π rad

Table 3.1: Distribution of spatial parameters.

The reduced pulses are validated by application in two pulse sequences: a pseudorandom sequence and a spin-echo sequence. The pseudorandom sequence consists of 160 repetitions of a random flip angle in the range $[0, \frac{\pi}{2}]$ rad followed by random waiting time in $[5, 15]$ ms. Different flip angles are achieved by scaling the RF amplitudes linearly with the flip angles. The spin-echo sequence consists of 160 repetitions of first a $\frac{\pi}{2}$ pulse followed 14 ms later by a π pulse followed by 6 ms waiting time. For each reduced pulse, at the end of each repetition the relative error of the response was compared with the fully-sampled pulse case.

Numerical results

For each pulse shape, the fully-sampled, interpolated, and optimized versions are shown in Figure 3.3. The calculation of the optimized pulse never took more than a few minutes. The difference between the interpolated and optimized pulse seems to increase for more complex pulse shapes. For the Gaussian case the difference between the interpolated and optimized pulse is too small to be observed in the figure.
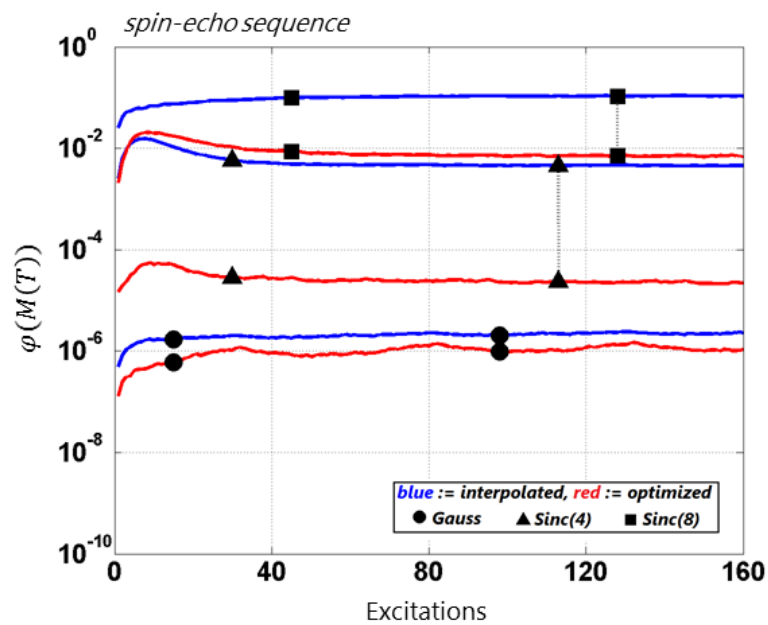
Figure 3.4 shows the approximation error in the pseudorandom sequence for the different reduced pulse shapes. There is a large difference between the approximation errors of the reduced pulses. However, a relative error value of $\mathcal{O}(10^{-3})$ is expected to be below experimental noise level and thus sufficient for most sequence simulations. Only the interpolated sinc(8) reduced pulse has a higher error. As expected, the method gives the best results for simple pulse shapes such as the Gaussian. The optimized pulse improved for all pulse shapes the accuracy measured by the relative error by at least a factor of 10 compared with the interpolated pulse.

In Figure 3.5, the approximation error for the spin-echo sequence is shown. The results are similar to those of the pseudorandom sequence. The lines of the graph are smoother compared with the previous sequence since each excitation has the same flip angles. As with the pseudorandom sequence, the optimized sinc pulses lead to a clear improvement compared to the interpolated case.

The minimal timings for the simulation of both sequences using either a fully-sampled or a reduced pulse are shown in Table 3.2. It shows that the use of reduced pulses greatly decreases simulation time compared with the fully-sampled case.
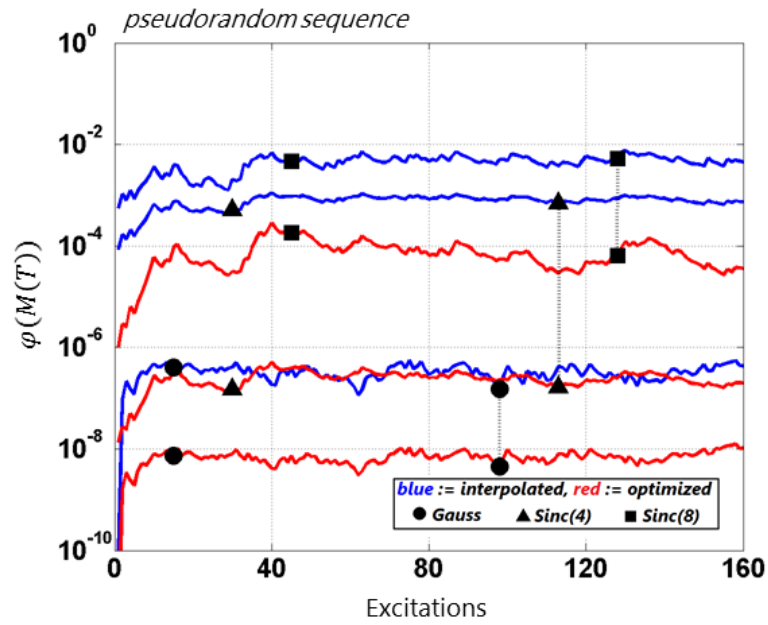
Figure 3.4: Approximation error in the pseudorandom sequence for the response of different reduced pulses compared with the fully-sampled case.

Figure 3.5: Approximation error in the spin-echo sequence for the response of different reduced pulses compared with the fully-sampled case.

	fully-sampled	reduced	acceleration
pseudorandom	682	19	35.9
spin-echo	1352	23	58.8

Table 3.2: Minimal timings in seconds for whole sequence simulation for all pulse shapes.

3.3 Discussion

The goal of this chapter was reducing the simulation time of the Bloch equations by either spatial or temporal domain reduction techniques. The initial focus laid on the spatial domain reduction, since a working projection method would be straightforward to implement in the optimal control RF pulse design method. No previous work was found on such a method. The projection framework is simple, but selecting the relevant subspaces for the projection and keeping simulations efficient is challenging. The projected matrices of the Bloch equations are no longer sparse and either a small basis or an efficient calculation strategy has to be found to reduce the simulation time compared to the full order method. A possible solution is undersampling the spatial domain. For such a method a subset of voxels has to be selected for which the magnetization response is representative of all positions.

Two methods for creating a basis for the magnetization response were investigated. The POD method might be applicable, but difficulties lie in the large number of significant singular values of the snapshot matrix, requiring a large basis to represent all states. Furthermore, it takes time to compute the snapshot matrix. However, methods such as the gradient reference frame by Grissom might be able to reduce the number of significant singular values.
The Krylov method is promising since it is based on the Bloch equations themselves instead of the snapshot based POD method. Here the challenges are which expansion points should be chosen and how to keep the Krylov basis small.
Overall, there are both still a number of ideas and challenges for application of spatial reduction to the Bloch equations, and further research is recommended.

Two previous methods for time domain reduction of the Bloch equations were found in literature. The method of Balac and Chupin gives a framework for quickly calculating the magnetization response in the case of perturbations in the RF field. However, the assumption is that both RF and gradient waveforms are constant and it is not clear if this method is also applicable if these assumptions are not met, such as in the optimal control method.
The article by Grissom has two interesting ideas: the small-perturbation method and the gradient frame that allows for SVD calculation. Combining these two techniques leads to a fast RF pulse design method using optimal control.

Finally, our own method for time domain reduction showed that it was pos-

sible to greatly reduce the simulation time of pulse sequences while still accurately describing the response of the fully-sampled pulses. The effect of a fully sampled pulse on the magnetization can be seen as a long sequence of rotations and decay. The reduced pulses show that approximately the same response can be found using only 15 rotations and decay operations. An interesting idea is if for a given fully sampled RF pulse, a single rotation and decay operator can be found which maps the initial magnetization to the desired state.

Conclusion and outlook

The main subject of this thesis was the optimal control method for RF pulse design as discussed in Chapter 2. The method was shown to greatly improve the magnetization response of inversion and refocusing pulses compared with the LCLTA method presented in Chapter 1.

In order to apply the optimal control pulses in an experimental setting, some additional research is required. First, the RF pulses should be validated in the MR scanner to confirm that the desired magnetization response is indeed obtained. This requires the RF coils to accurately produce the densely sampled and non-smooth RF pulses. Furthermore, effects not modelled by the Bloch equation might influence the magnetization.

Second, the RF pulse power (l_2-norm) should be limited to prevent heating of the scanned object. Section 2.3.5 discussed the regularization parameter α for balancing the RF amplitude with the magnetization error. Determining the value of α that leads to acceptable RF pulse power and thus prevents heating is required.

Third, the spatial domain has to be examined. In Section 2.3, the magnetization was controlled on a square with edges of 12 cm discretized into 128 steps in each direction. For clinical use, the domain has to be extended to a three-dimensional volume. Determining a good resolution for the optimal control method is also useful. A higher resolution gives a smaller discretization error at the cost of more voxels and thus more calculation time for solving the Bloch equation.

The optimal control method can also be extended from a mathematical point of view. First, the inclusion of the gradient waveforms as control variables might further improve the magnetization response. Note that the gradient waveforms require additional constraints on the amplitude and slew rate.

Second, different numerical methods were applied for finding the optimal search direction in each iterations. A comparison showed that there is no clear superior method, but the steepest descend method converged much

slower than the other methods. This means the choice of numerical method can have a large effect on the performance of the optimal control method. Additional research is required to determine if there is a method which is best suited for RF pulse design through optimal control. The line search algorithm might also be improved with more advanced algorithms.

Finally, the required calculation time of the optimal control method can be decreased if the magnetization response can be determined more quickly. This leads us to the subject of model order reduction.

In Chapter 3, the initial focus was on applying spatial domain reduction to the Bloch equation. Although no efficient working method has been found, some remarks on solution strategies are made. First, an efficient method for determining the magnetization response in the projected system should be found. In the projected system, the magnetization response can no longer be determined on a voxel level and the strategies of Chapter 1 no longer apply. Second, a basis should be found that accurately represents the different magnetization responses. Both POD as Krylov methods were described in Section 3.1, but are not simple to implement due to the respectively slow decay of singular values of the snapshot matrix and the almost purely imaginary eigenvalues.

For domain reduction of the temporal domain, two existing methods were discussed. Especially the method by Grissom et al. [10] seems well suited for RF pulse design through optimal control.

In Section 3.2.2, an original method was introduced for doing fast magnetization response simulations by applying piecewise constant pulses. These pulses were defined on much fewer time intervals than fully sampled pulses, but were still able to accurately approximate the magnetization response for a large number of test cases. Suggested is some research into finding substitute pulses for additional pulse shapes and small pulse sequences.

Bibliography

[1] D. Amsallem and C. Farhat. Projection-based model reduction (sheets). URL: web.stanford.edu/~amsallem/CA-CME345-Ch3.pdf (visited on 03/28/2015).

[2] A. C. Antoulas. *Approximation of large-scale dynamical systems*. Vol. 6. Siam, 2005.

[3] S. Balac and L. Chupin. Fast approximate solution of Bloch equation for simulation of RF artifacts in Magnetic Resonance Imaging. *Mathematical and computer modelling*, 48(11):1901–1913, 2008.

[4] M. A. Bernstein, K. F. King, and X. J. Zhou. *Handbook of MRI pulse sequences*. Elsevier, 2004.

[5] T. Breiten and T. Damm. Krylov subspace methods for model order reduction of bilinear control systems. *Systems & control letters*, 59(8):443–450, 2010.

[6] F. Casas, A. Murua, and M. Nadinic. Efficient computation of the Zassenhaus formula. *Computer physics communications*, 183(11):2386–2391, 2012.

[7] B. Chachuat. Nonlinear and dynamic optimization: from theory to practice. École Polytechnique Fédérale de Lausanne, 2007.

[8] S. Conolly, D. Nishimura, and A. Macovski. Optimal control solutions to the magnetic resonance selective excitation problem. *Medical imaging, IEEE transactions on*, 5(2):106–115, 1986.

[9] N. Del Buono, L. Lopez, and R. Peluso. Computation of the exponential of large sparse skew-symmetric matrices. *Siam journal on scientific computing*, 27(1):278–293, 2005.

[10] W. A. Grissom, D. Xu, A. B. Kerr, J. A. Fessler, and D. C. Noll. Fast large-tip-angle multidimensional and parallel RF pulse design in MRI. *Medical Imaging, IEEE transactions on*, 28(10):1548–1559, 2009.

[11] E. Haacke et al. *Magnetic resonance imaging: physical principles and sequence design*. John Wiley & Sons, 1999.

[12] W. W. Hager and H. Zhang. A survey of nonlinear conjugate gradient methods. *Pacific journal of optimization*, 2(1):35–58, 2006.

[13] P. C. Hansen. Deconvolution and regularization with Toeplitz matrices. *Numerical algorithms*, 29(4):323–378, 2002.

[14] Y. Lin, L. Bao, and Y. Wei. Order reduction of bilinear MIMO dynamical systems using new block Krylov subspaces. *Computers & mathematics with applications*, 58(6):1093–1102, 2009.

[15] D. Ma et al. Magnetic resonance fingerprinting. *Nature*, 495(7440):187–192, 2013.

[16] J. Pauly, D. Nishimura, and A. Macovski. A k-space analysis of small-tip-angle excitation. *Journal of magnetic resonance*, 81(1):43–56, 1989.

[17] J. Pauly, D. Nishimura, and A. Macovski. A linear class of large-tip-angle selective excitation pulses. *Journal of magnetic resonance*, 82(3):571–587, 1989.

[18] W. H. Press. *Numerical recipes 3rd edition: the art of scientific computing*. Cambridge university press, 2007.

[19] W. J. Rugh. *Nonlinear system theory*. Johns Hopkins University Press Baltimore, 1981.

[20] A. Sbrizzi, A. van der Toorn, H. Hoogduin, P. Luijten, and C. A. van den Berg. Spin TomogrAphy in Time domain: the MR-STAT project. In *ISMRM*, 2015, p. 3712.

[21] A. Sbrizzi et al. Time efficient design of multi dimensional rf pulses: application of a multi shift CGLS algorithm. *Magnetic resonance in medicine*, 66(3):879–885, 2011.

[22] D. Xu, K. F. King, Y. Zhu, G. C. McKinnon, and Z.-P. Liang. A non-iterative method to design large-tip-angle multidimensional spatially-selective radio frequency pulses for parallel transmission. *Magnetic resonance in medicine*, 58(2):326–334, 2007.

[23] D. Xu, K. F. King, Y. Zhu, G. C. McKinnon, and Z.-P. Liang. Designing multichannel, multidimensional, arbitrary flip angle RF pulses using an optimal control approach. *Magnetic resonance in medicine*, 59(3):547–560, 2008.