On joint reconstruction of spin density, R_2^* decay and off-resonance frequency maps through a single-shot or multi-shot acquisition in magnetic resonance imaging

by

Chenxi Hu

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
August 1, 2015

Keywords: quantitative MRI, model-based reconstruction, trajectory optimization, inverse problem

Approved by

Stanley J. Reeves, Chair, Professor of Electrical and Computer Engineering
Thomas S. Denney Jr., Professor of Electrical and Computer Engineering
Jitendra K. Tugnait, Professor of Electrical and Computer Engineering
Amnon J. Meir, Professor of Mathematics and Statistics
George T. Flowers, Professor and Dean of Graduate School

Abstract

Joint reconstruction of R_2^* and off-resonance frequency maps is very important in many MR applications. For example, R_2^* quantification can be applied to BOLD functional MRI, iron deposition measurement, and early detection of articular joint degeneration. Off-resonance quantification can be used to evaluate the severity of B_0 inhomogeneity, and it shows promise in MR thermometry. To reconstruct these parameter maps, a signal model must be specified, i.e., how the signal at each voxel evolves based on the spin density, R_2^* decay, and off-resonance frequency at this voxel. A common signal model for this problem is the mono-exponential model, where the signal is a complex exponential function in time with the amplitude from the spin density and the decay rate from the other two parameters.

A common approach to reconstruct the three parameter maps given the model is based on a multi-echo sampling—sampling k-space at a series of echo times. After the sampling, each k-space frame is inverse Fourier transformed to reconstruct a series of images. A curve fitting is applied to the series of images on a voxel-by-voxel basis along the time domain to reconstruct the three parameter maps based on the mono-exponential model. This method is very straightforward in its reconstruction; however, it generally takes a relatively long time to acquire the multi-echo data, ranging from tens of seconds to tens of minutes. The long acquisition time reduces the practicality of the method and makes it difficult to use in some clinical applications such as cardiac MRI. Reduction of the acquisition time in MRI in general is very important and has attracted numerous researchers. Typical methods include parallel imaging, partial Fourier sampling, sparse multi-echo, and single- or multi-shot acquisition.

Sparse multi-echo is generally a direct application of compressed sensing techniques to this problem. The simplest idea is to sparsely sample each k-space frame in the multi-echo sampling, and then reconstruct each image by using compressed sensing. A difference from

regular MR image reconstruction is the presence of a temporal dimension, and therefore a sparse representation of the signal in the temporal dimension is also very important. In fact, many authors have used sparse representations in both spatial and temporal domains so that a higher degree of undersampling is possible and a better reconstruction conditioning can be achieved.

Although the sparse multi-echo technique has been quite successful in its reconstruction quality, the time reduction achieved by the undersampling is typically only a factor of 4-6. Similar to the idea of undersampling, single- or multi-shot acquisition also undersamples the space that contains all k-space frames. However, a single- or multi-shot trajectory can achieve an even higher undersampling rate compared to sparse multi-echo. Typically, a single-shot trajectory only takes 40-80 milliseconds. Therefore, an important question is whether a quality reconstruction can be achieved with a single- or multi-shot k-space acquisition. In this dissertation, we propose two methods to do the underlying reconstruction and present analysis of the ill-conditioning of the problem under different single-shot trajectories. We present a new linear formulation of the mono-exponential model and demonstrate its power and potential applications in the reconstruction problem. In the following, we briefly introduce these contributions.

The first reconstruction method we propose in this dissertation is rooted in a classical nonlinear optimization strategy called trust-region methods. Specifically, we propose two trust region algorithms that use different local linearization techniques to address the same nonlinear optimization problem. A trust region is defined as a local area in the variable space where a local linear approximation is trustworthy. In each iteration, the method minimizes a local approximation within a trust region so that the step size can be kept in a suitable scale. A continuation scheme is applied to gradually reduce the regularization over the parameter maps and facilitate convergence from poor initializations. The two trust-region methods are compared to two other previously proposed methods—the nonlinear conjugate gradients algorithm and the gradual refinement algorithm. Experiments based on various synthetic

data and real phantom data show that the two trust-region methods have a clear advantage in both speed and stability.

The second method we propose in this work employs variable splitting, which is very popular in dealing with convex optimization problems such as image reconstruction with non-smooth regularizations. Most previous reconstruction algorithms are gradient-based iterative algorithms and the computational cost is high. We propose to reformulate the problem as a constrained optimization problem by employing auxiliary variables and use the well-known variable-splitting method to reduce the computational cost. We show that variable splitting for this problem is fundamentally different from variable splitting for many other applications, such as for regularized image reconstruction. As a result, the algorithm is very fast during the early stage of the iterations and much slower in the later stage. We propose a two-step method to address this issue. In the first step, we use the proposed variable-splitting method with regularization over both the auxiliary variable and the ordinary variables. We show the additional regularization is critical for the algorithm performance. In the second step, we employ our previously developed ordinary trust region algorithm to refine the estimate from the first step. We demonstrate that the hybrid method is faster than the ordinary trust region algorithm and the nonlinear conjugate gradient algorithm through simulation and *in vivo* data.

The third contribution of this work deals with a new formulation of the mono-exponential model. The formulation is in linear form which can be utilized by some optimization methods such as alternating directions. The new formulation results in a new estimator for the spin density, R_2^*, and off-resonance. The new estimator works on multi-echo data. We compare the new estimator with other traditional curve fitting methods and demonstrate the speed and accuracy of the new estimator. We show at the end of the chapter how this new formulation as well as the new estimator can be applied to the second iterative reconstruction algorithm we developed in addressing the joint reconstruction problem.

The last contribution of this work deals with analysis of the ill-conditioning of the underlying problem under a variety of single-shot trajectories. The aim of the analysis is to find the optimal existing trajectory in terms of the conditioning of the reconstruction problem. We use tools such as the condition number and the singular value curve to study the ill-conditioning. We verify the analysis result by comparing the reconstruction accuracy of a reconstruction algorithm for every trajectory. Our results show that some trajectories are better than other trajectories due to certain strategies used to sample the data space. The findings can be useful to improve the reconstruction accuracy and convergence speed for any algorithm developed for the joint reconstruction problem.

Acknowledgments

I would like to thank my advisor, Dr. Reeves. Dr. Reeves was very insightful about our research. The discussions with him always drove me to think more deeply about the various ideas we had. Dr. Reeves was also a devoted advisor. We met nearly every week over the past five years, which enabled discussion of every detail of my research. Finally, I want to thank Dr. Reeves for allowing me to pursue my applied mathematics degree and for helping me out many times even with my personal issues. I am so grateful to have had such a good advisor in my PhD education.

I would like to thank Dr. Denney, Dr. Beyers, Dr. Deshpande, and Dr. Twieg for providing so many great suggestions in my research. These suggestions were very helpful to my understanding of MRI physics, pulse sequence programming, and MRI applications. Without them, it would've been almost impossible for me to obtain a post-doctoral position in a medical school.

I would like to thank all my labmates, including Karthik, Meredith, Ming, Paul, Ranga, Xiaoxia, and Yuan, for their suggestions, help, and smiles during my research and life. I would like to thank all my friends; my life has become much more interesting because of them.

I would like to thank my parents and my grandma. They helped me to get started in this wonderful journey, and they have always taken care of me no matter where I was.

Finally, I would like to thank my wife, Ailu. She is the most beautiful lady for me. She is wise, cute, and helpful. Meeting her, marrying her, and graduating with her are among the most beautiful moments in my life.

Table of Contents

List of Figures

Chapter 1

Introduction

Magnetic Resonance Imaging (MRI) is an important medical imaging technology that has been used in numerous medical applications. In this first chapter, we first introduce some MRI terminology and familiarize readers with the basic knowledge of MRI that is essential to understanding the reconstruction problem. We then elaborate on the specific reconstruction problem we are dealing with. We finally review methods in dealing with related problems from various perspectives to unveil the context as well as the novelty of our research.

1.1 Introduction to MRI

1.1.1 MRI basics

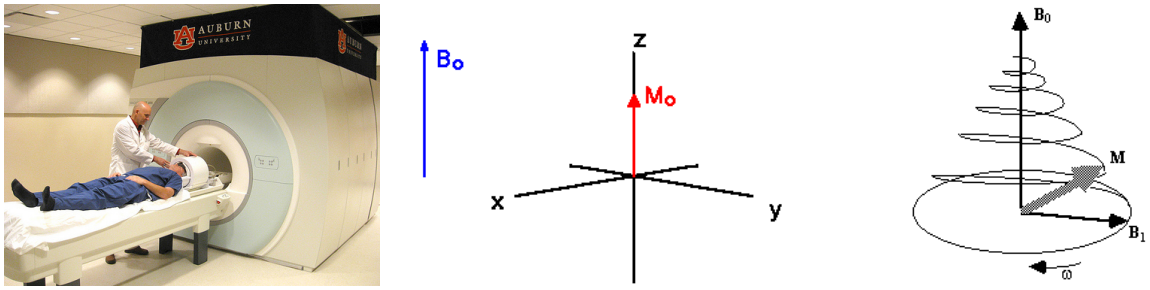
The physical basis of MRI is fairly complicated. A rigorous introduction to MRI must involve quantum physics and is beyond this work. Interested readers can refer to the monograph "Magnetic Resonance Imaging: Physical Principles and Sequence Design" by Robert W. Brown et al. for a rigorous introduction [8]. In this chapter, we briefly review the key concepts of MRI that closely relate to our reconstruction problem.

An MRI scanner has patients, subjects, or other objects placed within a strong magnetic field which is created by a large electromagnetic device (Figure 1.1(a)). When atomic nuclei are placed in a strong magnetic field, which in MRI is called the B_0 field, the nuclei are magnetized. As a result, the bulk magnetization of these nuclei points in the direction of the B_0 field (Figure 1.1(b)). It is conventional to denote the direction of the B_0 field as the z direction. The plane that is orthogonal to the z direction is the xy plane, where the x and y axes can be established by the right hand screw rule. The B_0 field is always on. When

imaging starts, a radiofrequency (RF) radiation field, the B_1 field, is superimposed onto the B_0 field. The B_1 field is tuned to a certain frequency called the Larmor frequency. The Larmor frequency is defined as

$$f = \gamma B_0 \tag{1.1}$$

where B_0 is the strength of the B_0 field and γ is the gyromagnetic ratio, which is different for different atoms. When the frequency of the B_1 field equals the Larmor frequency, the nuclei resonate, causing the bulk magnetization to spiral away from the direction of the B_0 field (Figure 1.1(c)). On one hand, the angle between the bulk magnetization and the B_0 field becomes larger and larger; on the other hand, the bulk magnetization is rotating about the B_0 field. When the B_1 field is stopped, the magnetization stops going away from the B_0 field but still rotates about it. The rotation of the magnetization introduces a dynamic magnetic field around the nuclei. The net field from the object being imaged is captured by an electromagnetic detecting device that is always placed around the object (Figure 1.1(a)). This detected signal forms the basis of MRI being able to "see" the internal structure of an object.



(a) (b) (c)

Figure 1.1: (a) shows a SIEMENS MRI scanner and a head receiver coil. (b) the equilibrium state of the bulk magnetization in B_0 field. (c) shows the motion of the magnetization after the B_1 field is applied. [1]

To form a useful image, the image must have contrast. The contrast in an MRI image is provided by the difference of the magnetic fields induced by the rotating magnetization

[1] Source: (a) AU MRI center webpage (b)Online tutorial "The basics of MRI", J.P. Hornak, https://www.cis.rit.edu/htbooks/mri/chap-3/images/t1-1.gif; (c)Thesis "Functional MRI : Methods and Applications", Stuart Clare, http://users.fmrib.ox.ac.uk/stuart/~thesis/chapter_2/image270.gif;

among different biological tissues. For example, water and fat have different concentrations of ^1H protons, and therefore their magnetic fields have different strength. The number of protons within a unit volume of material is called spin density. From the example, we see the difference in spin density among various biological tissues offers a type of image contrast. At the right side of Figure 1.2 is a proton-density-weighted image. Another main type of contrast is caused by relaxation. Relaxation refers to the fact that the bulk magnetization after the B_1 field is stopped does not remain constant; instead, the magnetization will gradually point back to the direction of the B_0 field. Different tissues often have different relaxation strength; that is, some restore their direction faster than others. Hence, those tissues having faster relaxation have a smaller magnetic field strength than others if one ignores their difference in spin density. There are several types of relaxations, such as T_1 relaxation, T_2 relaxation, and T_2^* relaxation. They refer to relaxation in different directions or of different types. Spin density as well as these relaxations are the main factors that influence the strength of the induced magnetic fields. The left and middle images of Figure 1.2 show T_1-weighted and T_2-weighted images, respectively. The interaction between various relaxations and the spin density is described by the famous Block equation in quantum physics. In this work, our main objective is to reconstruct the spin density and the T_2^* relaxation maps. Therefore, these concepts will be revisited in the next section.

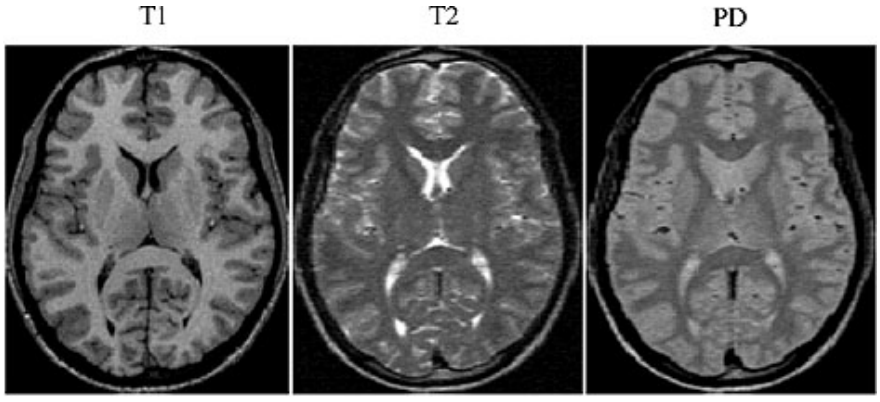


Figure 1.2: Different types of contrast for one human brain slice. [2]

[2] Source: Class notes, "Computed imaging system", Thomas Denney

The smallest visible unit of the object being imaged in MRI is called a voxel, corresponding to a pixel in a plane. The objective of MRI is to reconstruct the magnetic field strength induced by each voxel. If only the B_0 field is present, all voxels induce magnetic fields of the same frequency, causing the bulk magnetic fields to be a pure summation of all these induced magnetic fields. Each individual contribution would not be discernible. Modern MRI scanners solve this problem by superposing gradient fields onto the B_0 field. The result of superposition is a linearly varying magnetic field applied to the object. The strength of the composite external field at a particular voxel depends on its location in the image space. The varying external field fosters a Fourier encoding of the induced field strength of each voxel. The Fourier encoding is achieved by varying the gradient field strength many times, and each time corresponds to a line or a sample of the Fourier frequency domain. The image of the strength of induced fields from each voxel is then reconstructed by applying an inverse Fourier transform to the acquired spectrum. Nowadays, all MRI scanners are equipped with gradient coils that generate the gradient fields and use the Fourier encoding and decoding process to reconstruct images.

Other important concepts in modern MRI scanners, such as RF coils, coil arrays, and shimming coils, are less connected to our reconstruction problem and hence are not elaborated in this introduction section. As we proceed, we may come across these concepts in references or experiment parameter specifications. In this case, we may briefly describe the involved concepts to avoid confusion.

1.1.2 Excitation and relaxation

Before the B_1 field is applied, the magnetization of all nuclei point along the z direction. We thus can use a vector M to represent the magnetization. After the B_1 field is applied, the magnetization vector gradually spirals down to the xy plane (Figure 1.1(c)). The angular frequency of each magnetization vector is the Larmor frequency. In MRI, it is common to use the rotating frame instead of the laboratory frame to analyze the behavior of the

magnetization vector after resonance. The rotating frame means a coordinate system that rotates about the z axis at the Larmor frequency. In this frame, any vector that rotates at the Larmor frequency would be a static vector. We denote the x, y axes in the rotating frame by x' and y'. Since in the rotating frame, the movement of each magnetization vector in the $x'y'$ plane is zero, the magnetization vector, which begins perpendicular to the xy plane, rotates directly down to the $x'y'$ plane after the resonance without any rotation in the xy plane (Figure 1.3). The angle between the magnetization vector and the z axis at the end of the B_1 field is dependent on the strength and length of the B_1 field. A B_1 field that makes the magnetization vector completely tip down to the $x'y'$ plane is called a 90° pulse. When the pulse is stopped, the magnetization vector is completely in the $x'y'$ plane. In particular, if the RF pulse is produced along the x' direction, then the magnetization vector will tip down to the y' direction.

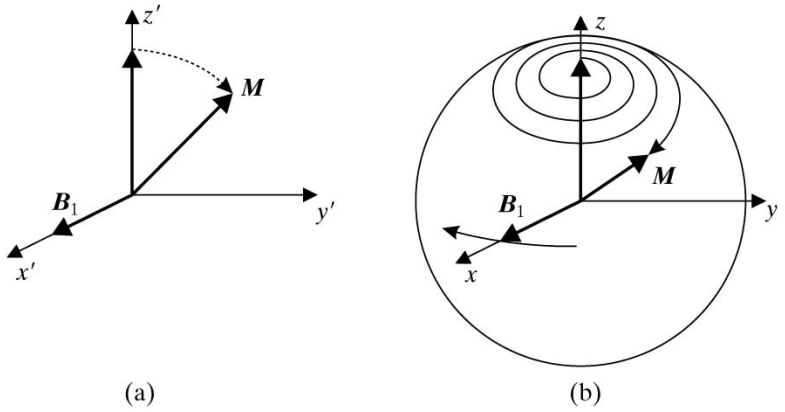


Figure 12.4

Figure 1.3: Motion of the magnetization vector in the rotating frame (left) and laboratory frame (right).

Before the B_1 field is applied, the magnetization is said to be in an equilibrium state (M_0). After the B_1 field is applied, several types of relaxation exist to change the magnetization vector back to its equilibrium state. Specifically, on one hand, relaxation causes the longitudinal magnetization (M_z), or the component of M in the z direction, to recover

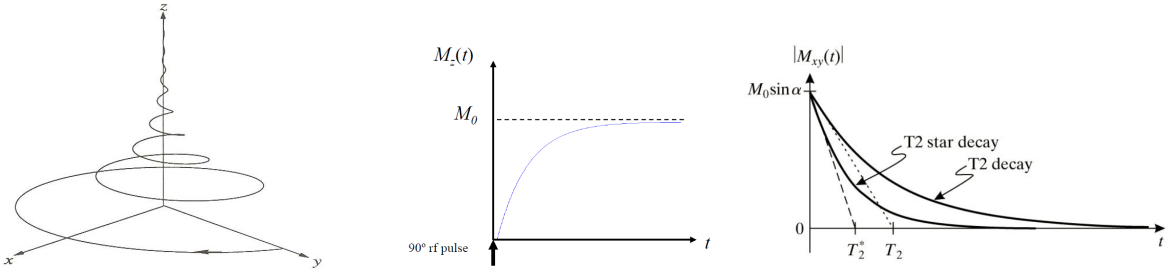
from 0 (assuming a 90° RF pulse) back to M_0. On the other hand, relaxation causes the transverse magnetization ($M_{x'y'}$), or the component of M in the $x'y'$ plane, to decay from M_0 down to 0. This process is illustrated in Figure 1.4(a). The z direction relaxation is called T_1 relaxation, or the spin-lattice relaxation in physics. The M_z dynamics under T_1 relaxation can be described as (Figure 1.4(b))

$$M_z(t) = M_0(1 - e^{-t/T_1}) \tag{1.2}$$

where T_1 is the time constant associated with the relaxation. Notice this representation assumes the used RF pulse is a 90° pulse. The relaxation in $x'y'$ plane is called T_2 relaxation, or the spin-spin relaxation. The $M_{x'y'}$ dynamics under the T_2 relaxation can be described by

$$M_{x'y'}(t) = M_{x'y'0}e^{-t/T_2} \tag{1.3}$$

where $M_{x'y'0}$ is the initial component of M in the $x'y'$ plane. When a 90° RF pulse is used, $M_{x'y'0} = M_0$. For an arbitrary α-pulse where α represents the angle between M and z, $M_{x'y'0} = M_0 \sin\alpha$.



(a) (b) (c)

Figure 1.4: (a) shows the composite function of longitudinal and transverse relaxation. (b) shows dynamics of M_z under T_1 relaxation. (c) shows dynamics of strength of $M_{x'y'}$ or M_{xy} under T_2 or T_2^* relaxation. [3]

The T_2 relaxation is caused by molecular interactions and therefore is not invertible. This is different from another relaxation T_2^*, which is invertible under certain operations.

[3] Source: (a) Class notes, "Computed imaging system", Thomas Denney

The T_2^* relaxation arises in any real MRI equipment, where the B_0 field cannot be kept absolutely constant in the xy plane. Even across one voxel, some segments of the voxel may have a slightly larger B_0 strength than other segments. The B_0 fluctuation, or B_0 field inhomogeneity, causes $M_{x'y'0}$ to decay faster than the T_2 process. In addition to B_0 inhomogeneity, some other factors, such as magnetic susceptibility difference between different tissues, can also cause a faster decay of $M_{x'y'0}$. The total $M_{x'y'0}$ decay in a realistic MRI system is called T_2^* relaxation. T_2^* relaxation is always shorter than T_2 relaxation (Figure 1.4(c)). The decay rate of T_2 and T_2^* relaxation refers to the reciprocal of the time constant T_2 and T_2^* and is denoted by R_2 and R_2^*, respectively. From previous analysis, R_2^* is always larger than R_2. Since the acceleration from T_2 to T_2^* is caused by an external field that is deterministic, the acceleration can be reversed. This is achieved by applying a spin-echo sequence, which we will briefly explain in the next section. The three relaxations have important implications for the contrast of the reconstructed image. They also have important implications in some applications of MRI. For example, R_2^* decay rate is an important biomarker for iron deposition in organs such as the liver and heart. Clinical protocols already exist to measure the R_2^* decay rate to identify the distribution and deposition of iron in these organs [4, 85, 114].

Another factor that changes M_{xy} throughout an imaging process is off-resonance. In the ideal situation, all bulk magnetizations rotate at the Larmor frequency. After data acquisition, demodulation is applied to remove the signal component at the Larmor frequency, resulting in a signal that is a Fourier transform of the spatial image. However, in reality, many factors destroy the homogeneity of the static field. The difference $f \triangleq \gamma(B - B_0)$, where B and B_0 denote the actual field and the main field provided by the scanner, is the off-resonance frequency. Notice the off-resonance frequency may slowly vary across the image space. The off-resonance frequency causes a phase term in the associated magnetization vector that linearly increases with time. The effect can be represented by

$$M_{x'y'}(t) = M_0 e^{ft} \tag{1.4}$$

The two main reasons for the presence of the off-resonance are (1) main field inhomogeneity and (2) susceptibility difference between different tissues. Other factors, such as eddy current induced by the time-varying field, can also influence the off-resonance but may be quite small compared to the previous two factors. In this work, one of our objectives is to reconstruct the off-resonance map, or the field map.

1.1.3 k-space

Excitation of the protons can generate signal from the object; however, excitation does not generate an image. Most modern MRI image reconstruction is based on the Fourier transform. The Fourier encoding is fulfilled by the dynamic gradient field, which is created after the excitation by the gradient coils. The encoded signal is then detected by the receiver coils and inverse Fourier transformed to reconstruct the spatial image. Since relaxation happens immediately after the excitation, the timing of the Fourier encoding influences the strength of the received signal. For example, in terms of T_2 relaxation or T_2^* relaxation, the later the encoding happens, the lower the received signal is. When there are tissues whose relaxation rates are different, contrast occurs. Figure 1.5 shows the contrast between two tissues of different T_1 or T_2 relaxation rates.

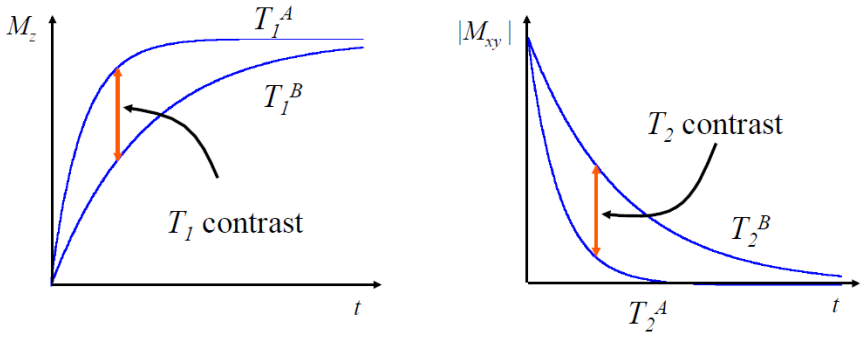


Figure 1.5: contrast between tissue A and B caused by different T_1 (left) or T_2 (right) relaxation rates [4]

[4] Source: Class notes, "Computed imaging system", Thomas Denney

The encoding process is closely related to the frequency domain of the image, called k-space in MRI. Suppose the image size is 256×256, then the properties of the discrete Fourier transform require 256×256 grid points in k-space to be able to take the inverse Fourier transform. Although any sampling trajectory should work as long as the trajectory passes through all grid points, the most common and well-established sampling trajectory or pattern is a line-by-line sampling. In 2D imaging, this usually means starting from a line at the k-space side, sampling each grid point on the line in one direction, and changing to the next line until all lines are sampled. This sampling pattern is illustrated in Figure 1.6.

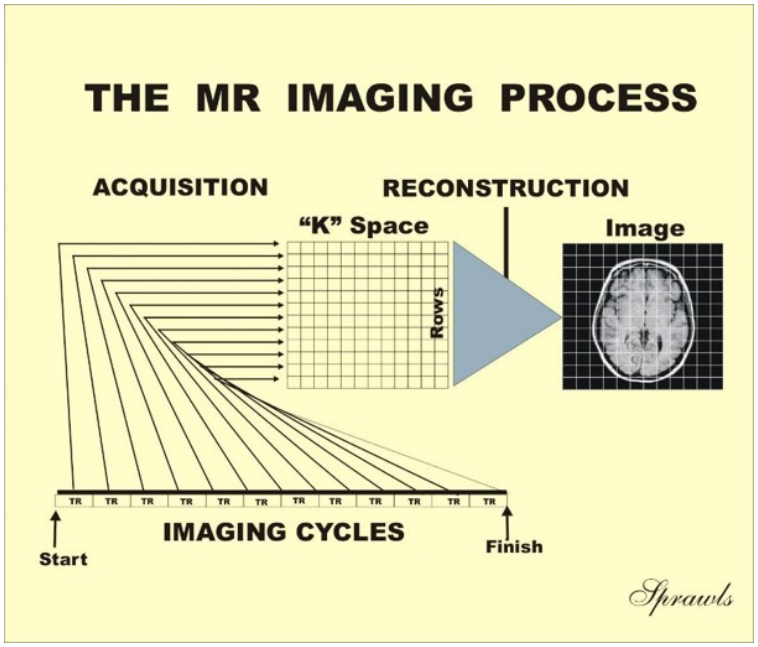


Figure 1.6: Line-by-line sampling of k-space. [5]

The gradient field is dispatched in three directions, namely the frequency encoding direction, phase encoding direction, and slice-selective direction, in 2D imaging. The slice-selective gradient does not participate in the Fourier encoding and therefore we do not elaborate on it in this section. For simplicity, assume x is the frequency encoding direction and y is the phase encoding direction. In reality, this may not be true, but the result from this case can be easily generalized. Let G_x and G_y represent the gradient slope of the frequency

[5] Source: "Magnetic Resonance Imaging Principles, Methods, and Techniques", Perry Sprawls

encoding gradient and phase encoding gradient, respectively. When $G_x > 0$, the positive side of the x axis has a positive magnetic field offset; while $G_x < 0$, the negative side of the x axis has a positive magnetic field offset. When $G_x = 0$, the entire x axis has a constant B_0 field. The same is true for G_y. The total field present at an arbitrary coordinate (x, y) can be represented by

$$B_0 + G_x x + G_y y \tag{1.5}$$

The magnetization vector of this voxel can be represented by

$$M(x, y, t) = m(x, y)e^{-2\pi\gamma(B_0+G_x x+G_y y)t} = m(x, y)e^{-2\pi\gamma[B_0 t+(G_x t)x+(G_y t)y]} \tag{1.6}$$

The total signal from the entire xy plane is an integral:

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} M(x, y, t)dxdy = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} m(x, y)e^{-2\pi\gamma B_0 t}e^{-2\pi\gamma[(G_x t)x+(G_y t)y]}dxdy \tag{1.7}$$

Usually, a demodulation step is applied to the measured signal to remove the high-frequency component caused by $e^{-2\pi\gamma B_0 t}$, causing the remaining signal to be

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} m(x, y)e^{-2\pi\gamma[(G_x t)x+(G_y t)y]}dxdy \tag{1.8}$$

Notice how close this is to a Fourier transform. Suppose $G_y = 0$ at the moment when the signal is collected. The sampled signal then is

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} m(x, y)e^{-2\pi\gamma(G_x t_l)x}dxdy \tag{1.9}$$

where $(t_l)_{l=1}^{L}$ represents the moments of sampling. Let $k_x(t_l) \triangleq G_x t_l$. Thus, expression (1.9) can be rewritten as

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} m(x, y)e^{-2\pi\gamma k_x(t_l)x}dxdy, \tag{1.10}$$

and $k_x(t_l)$ represents sampling of the central k-space line from the negative-to-positive direction.

In order to sample the entire k-space in a line-by-line fashion, the phase encoding gradient G_y and the frequency encoding gradient G_x are not present at the same time. Often, G_y is generated much earlier than G_x. Each time, G_y lasts T seconds to make $k_y \triangleq G_y T$ equal the position of a k-space line in the k_y direction. When jumping from one line to another, the gradient strength G_y is incremented or decreased by a small amount ΔG_y so that $\Delta k_y \triangleq \Delta G_y T$ equals to the difference of positions between the two k-space lines. This process is called phase encoding. After phase encoding, G_x is applied, and sampling continues until the entire k-space line can be sampled. This process is called frequency encoding. The time frame for the frequency encoding is called readout time, since the data is read out by the receiver coil. The frequency encoding gradient is also referred to as the readout gradient due to the same reason. If the readout starts from the center column of k-space, it cannot sample the entire k-space line. Therefore, a preparing gradient is usually applied before the readout time with an opposite polarity to the readout gradient to move the sampling backward by half of the k-space range. The entire process is illustrated in Figure 1.7.

1.1.4 MRI pulse sequences

So far, we know the MR imaging process requires running an RF field to excite the protons and several gradient fields to encode the image. In addition to these functions, an RF field and gradient field can be used to fulfill other purposes. The orderly combination of RF field and gradient fields is called a pulse sequence, since each magnetic field lasts a very short time—such as several milliseconds—resembling an electromagnetic pulse. From creation of the first MRI image, many different pulse sequences have been developed to realize different functions. A modern commercial MRI scanner is usually pre-programmed with tens of pulse sequences for radiologists and researchers to use for their own needs. The

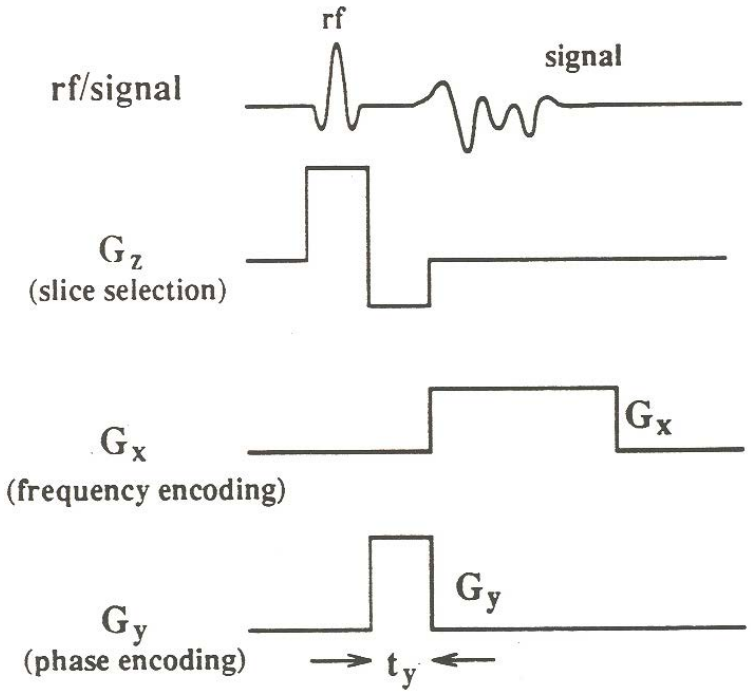


Figure 1.7: Function of frequency encoding gradient, phase encoding gradient, and slice-selective gradient in line-by-line sampling [6]

two most commonly used types of pulse sequence are spin echo sequences and gradient echo sequences. In this section, we briefly review the two types of pulse sequence.

Spin echo

As we mentioned, the magnetization vectors in a realistic magnetic field experience T_2^* relaxation. Part of the relaxation is from T_2 relaxation, which is irreversible. The rest of the T_2^* relaxation is caused by field inhomogeneity and can be reversed by running a refocusing RF pulse. The refocusing RF pulse is different from an excitation RF pulse in that the refocusing pulse causes a 180° tipping of the magnetization instead of 90° tipping (Figure 1.8). Also, the refocusing RF pulse is applied after the proton excitation is finished. Before the refocusing pulse, protons in a voxel experience so-called dephasing, since some of them precess faster than others. Protons gradually become out of phase, and their spins start to cancel each other. This leads to a very small bulk magnetization for the voxel. After the

[6] Source: Class notes, "Computed imaging system", Thomas Denney

refocusing pulse, all spins are flipped to the opposite side and continue the precession. The spin with the largest positive phase becomes the one with the largest negative phase. After an additional time frame equal to the time between the excitation and the refocusing, all spins precess back to the in-phase position. The signal at this point is very strong and is called an echo. A refocusing pulse can be applied multiple times to create multiple echoes. A pulse sequence that uses the refocusing RF pulse to recreate echoes is called a spin echo sequence (Figure 1.9).

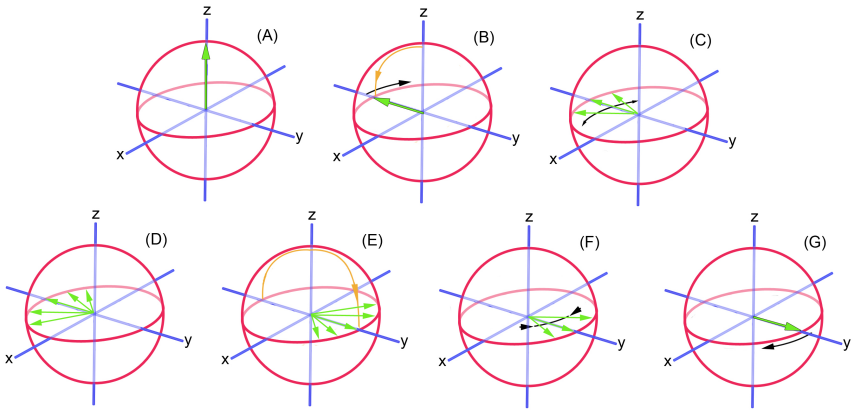


Figure 1.8: Refocusing of spins by 180° RF pulse. (A-D): spins are tipped down and start to dephase; (E): spins are tipped by an 180° pulse to completely opposite directions; (F-G): spins precess in the same direction and gradually become in-phase. Notice that the rotating frame is used for this graph. [7]

Figure 1.9 also shows the basic components of a pulse sequence: the RF pulse, slice-selective gradient pulse, phase encoding pulse, and frequency encoding pulse. It also shows two important timing concepts— time of repetition and time of echo. The time of repetition, often abbreviated as TR, means the repetition cycle between two consecutive excitation RF pulses. The time of echo, often shortened as TE, means the time between the center of the excitation pulse and the echo. In spin echo, both excitation and refocusing RF pulses must be used. The slice-selective pulse is used whenever an RF pulse is used, including both excitation and refocusing pulses. The phase encoding pulse is changing its amplitude each time when it is applied to sample each k-space line. It is therefore shown as a stack

[7] Source: Spin echo diagram, Aaron Filler, http://en.wikipedia.org/wiki/File:Spin_Echo_Diagram.jpg

of boxes to represent a collection of different amplitudes. The frequency encoding pulse, or the readout pulse, is applied to read out each k-space line. At the center of each readout is the point when all spins become in-phase. Since every echo is acquired when spins are in-phase, the reconstructed image is T_2-weighted. The outcome is illustrated in Figure 1.10. When multiple echoes are acquired, multiple T_2-weighted images are reconstructed and can be used to reconstruct an R_2 decay rate map.

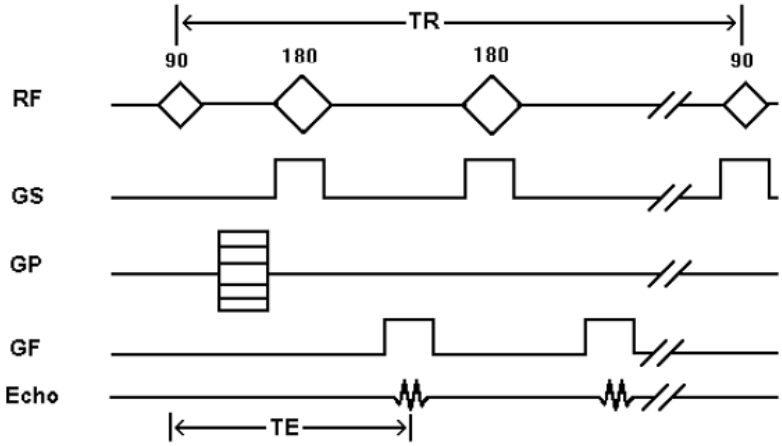


Figure 1.9: A double-echo spin echo sequence. [8]

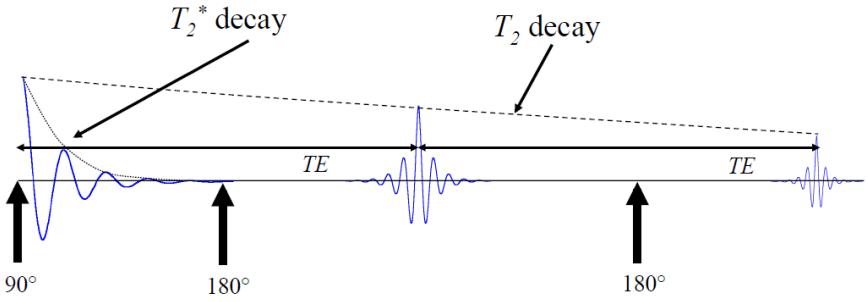


Figure 1.10: Relaxation of spins in a double spin echo sequence. [9]

Spin echo is the most commonly used pulse sequence clinically. This is mainly attributed to the high signal intensity and high SNR associated with the T_2-weighted image. Hardware imperfections such as field inhomogeneity, nonlinear gradients, and inevitable factors such as

[8] Source: Radiopaedia.org, http://radiopaedia.org/cases/spin-echo-sequence-diagrams-1

[9] Source: Class notes, "Computed imaging system", Thomas Denney

susceptibility differences and chemical shift are largely diminished. Most other sequences do not have this merit, including another commonly used sequence—the gradient echo sequence.

Gradient echo

The spin echo sequence uses refocusing RF pulse to create an echo. The gradient echo sequence does not use this method; instead, it uses the frequency-encode gradient to create an echo. In each TR, a gradient echo sequence applies the frequency-encode gradient twice. In the first time, a short frequency-encode lobe is applied together with a phase-encode lobe to move the sampling location from the center of k-space to one side of a k-space line. In the second time, a long frequency-encode lobe with opposite amplitude and double length should be applied to sample the k-space line. This lobe is the readout gradient lobe. At the end of this lobe, the sampling location should be at the other side of the k-space line. In each TR, this process is repeated except the amplitude of the phase-encode gradient is different to encode different k-space lines. Figure 1.11 illustrates the application of different gradient fields in one TR of the gradient echo sequence. Since refocusing is lacking, the signal experiences T_2^* relaxation. The reconstructed images from gradient echo are therefore T_2^*-weighted images.

Although gradient echo does not have the high signal intensity of spin echo and is more susceptible to field inhomogeneity, its main advantage is the high imaging speed. A spin echo sequence takes around a minute or even longer to acquire enough data for reconstructing a high-resolution image. A gradient echo sequence, however, can achieve this in several seconds. This is mainly attributed to the low flip angles a gradient echo sequence uses. Flip angle is the flipped angle between the rotated magnetization vector and the B_0 field, caused by the RF excitation. A spin echo sequence usually uses 90° pulse to gain the largest possible xy-plane magnetization component. In gradient echo, usually a very small flip angle such as 10°-20° is used. A large flip angle requires a long TR time to let the longitudinal relaxation finish. Otherwise, a phenomenon called saturation would happen which can reduce the

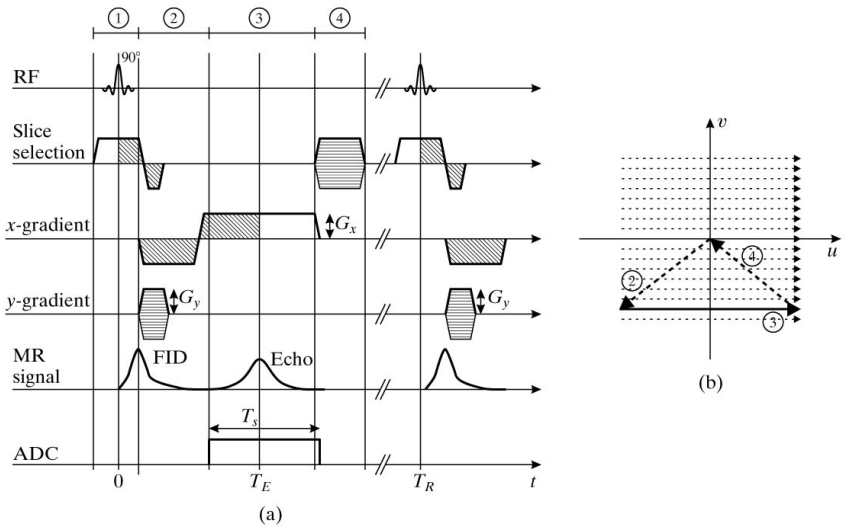


Figure 13.18

Figure 1.11: One TR of a gradient echo sequence and the corresponding movement in k-space.[10]

signal intensity of the image. Gradient echo uses very small flip angles to largely reduce this problem. The signal intensity is therefore generally low. However, this strategy facilitates a very short TR and makes the overall scan time much shorter, making gradient echo clinically valuable.

Another characteristic of gradient echo is allowing a very flexible sampling of k-space. In spin echo, all sampling must be concentrated around the echo time. In gradient echo, since there is no such restriction, k-space can be sampled in any fashion. Specifically, k-space can be sampled line-by-line, or along a trajectory. A trajectory is an arbitrarily shaped curve in k-space that tracks the sampling. In our work, we use many different trajectories, and they are important to the reconstruction result. We therefore elaborate on several common trajectories used in gradient echo sequences.

The first trajectory is echo planar imaging (EPI). It is very different from other k-trajectories since it still samples in a line-by-line fashion. For this reason, most literature

[10] Source: Class notes, "Computed imaging system", Thomas Denney

does not count EPI as a trajectory. However, since EPI is often finished in one TR in contrast to traditional line-by-line sampling, we place it in the trajectory category in this work. EPI was proposed very early; however, it experienced a slow development in the first few decades due to hardware restrictions [6, 20, 91]. EPI is very fast, since it can finish sampling the entire k-space in one TR. Such trajectories are called single-shot trajectories. The EPI pulse sequence is illustrated in Figure 1.12 and the trajectory of EPI in k-space is shown in Figure 1.14(a). After the excitation and the preparation pulses for shifting sampling to the k-space corner, multiple readout gradients are emitted with alternate amplitudes. Between two consecutive readout gradients, a phase-encoding blip is applied to move the sampling upward by one k-space line. The sampling therefore follows a zigzag trajectory to sample the entire k-space. The blip gradient greatly saves the time needed for phase encoding. The hardware challenge is to quickly change the frequency-encode gradient from negative amplitude to positive amplitude, which requires a large gradient slew rate.

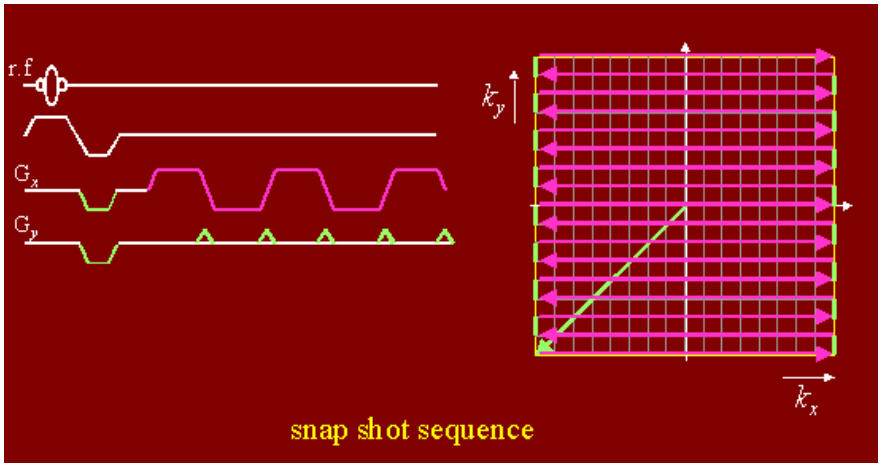
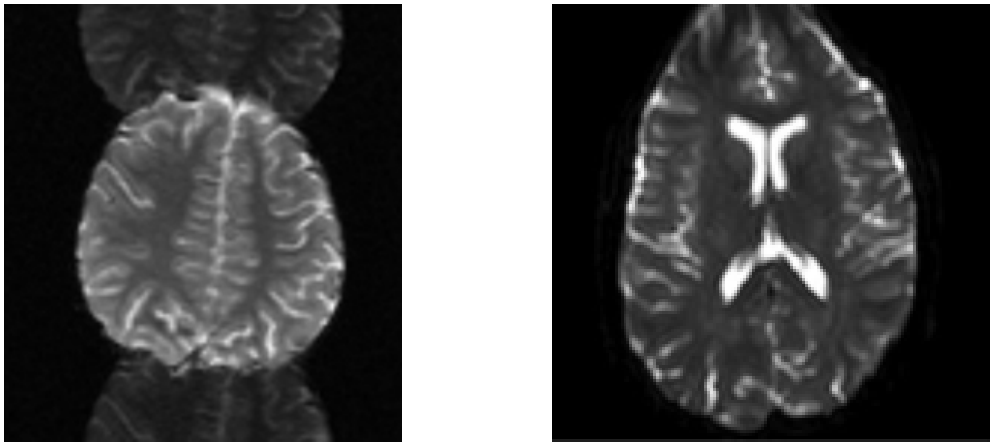


Figure 1.12: EPI pulse sequence and the corresponding sampling trajectory in k-space. [11]

Without any compensation, EPI can lead to many artifacts, including ghosting, blurring, geometric distortion, and signal drop-out [6, 91]. Figure 1.13 shows the ghosting and geometric distortion artifacts in two brain EPI images. These artifacts are caused by various factors, and solutions have been commonly used in commercial EPI protocols. A lot of these

[11] Source: Tutorial, "Basic physics of fMRI", Peter Jezzard, University of Oxford; http://users.fmrib.ox.ac.uk/~peterj/lectures/hbm_1/sld044.htm

artifacts can be dramatically reduced or even eliminated, such as ghosting; however, in some areas, such as the sinus-air interface, a large susceptibility difference is present and can still cause severe signal drop-out and distortion in EPI images. Off-resonance is one of the main causes of geometric distortion.



ghosting distortion

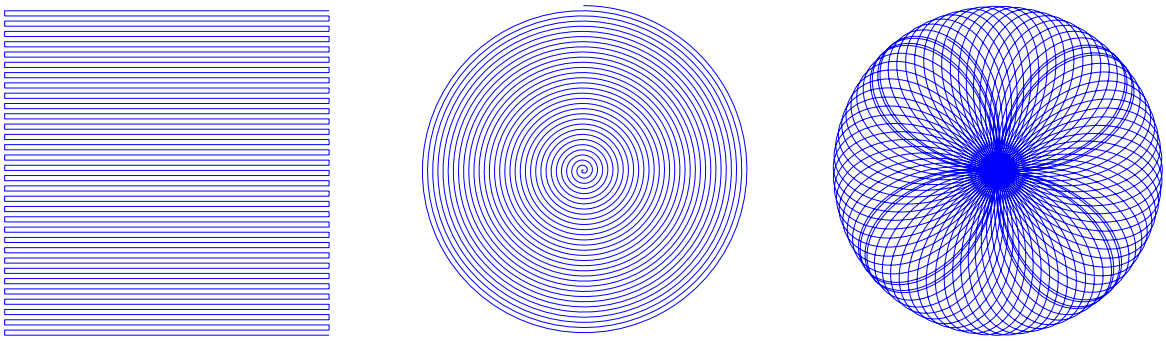
Figure 1.13: Ghosting and distortion artifacts of EPI sequence. [12]

The second trajectory is spiral (Figure 1.14(b)). The spiral trajectory has been used in many applications, such as fast imaging [38, 46], compressed sensing [68], and MR fingerprinting [70]. The main advantage of spiral is the fast covering of k-space. Typically, a single-shot spiral trajectory may take 15-25 milliseconds to cover k-space for a reasonable resolution. Two alternatives exist in the spiral trajectory family: spiral-in and spiral-out. Between them, spiral-out is more frequently used mainly because the echo time, i.e. the time when the trajectory passes the k-space origin, is at the beginning of the readout, resulting in a stronger signal intensity. Since spiral takes a long time to cover k-space, spiral has the same vulnerability to off-resonance and relaxation as EPI. However, since spiral has a lower gradient slew rate than EPI, artifacts associated with rosette are usually less severe. On the other hand, the reconstruction of spiral is much more complicated than EPI since

[12] Source: (a) MRIQuestions.com http://mri-q.com/nyquist-n2-ghosts.html; (b) Center for Imaging Science, Universitat zu Lubeck, http://www.cis.uni-luebeck.de/index.php?id=14;

spiral does not sample on each k-space grid. Such a trajectory is commonly referred to as a non-Cartesian trajectory.

The third trajectory is rosette. Rosette is also a non-Cartesian trajectory and was first introduced in [81]. In this work, Noll used this trajectory to do spectrally selective imaging, which keeps energy of the on-resonance portions of the image and destroys energy of the off-resonance portions. The trajectory is shown in Figure 1.14(c). The trajectory passes the k-space origin many times, causing a high sensitivity to image variation caused by relaxation, off-resonance frequency, and motion. Twieg later used this trajectory to map R_2^* relaxation maps and off-resonance maps [102]. The method, called "PARSE" in his paper, is the predecessor of our work. Details about the three trajectories are further discussed in the last chapter.



EPI Spiral Rosette

Figure 1.14: Three k-space trajectories

1.2 Formulation of the problem

So far, we have briefly explained the basics of MRI, as well as many critical concepts such as excitation, relaxation, k-space, gradient field (pulse), spin echo sequence, gradient echo sequence, and k-space trajectories. In this section, we formulate the problem addressed in this dissertation.

1.2.1 Mono-exponential modeling

Mono-exponential modeling represents the signal from a voxel as a mono-exponential function

$$m(\vec{r})e^{-z(\vec{r})t} \tag{1.11}$$

where $\vec{r}$ represents the spatial coordinate of the voxel in the image space and t represents time. $m(\vec{r}) \in \mathbb{C}$ represents spin density at the voxel. Although density is usually a real number, $m(\vec{r})$ also represents the angle of the initial magnetization vector in the image space. Therefore, in reality, it is a complex number. $z(\vec{r}) \in \mathbb{C}$ is the complex decay of the signal whose real and imaginary part are R_2^* decay and off-resonance frequency, respectively. Notice since the problem we deal with mainly focuses on gradient echo imaging, the prominent relaxation in the xy plane is the T_2^* relaxation. Let $d(\vec{r})$ and $f(\vec{r})$ represent the R_2^* decay rate and off-resonance at this voxel. Then $z(\vec{r}) \triangleq d(\vec{r}) + \iota f(\vec{r})$.

Mono-exponential modeling is commonly used for applications such as R_2^* relaxation mapping [4, 26, 40, 85]. However, some authors have used different models such as a bi-exponential model [9, 56], where the signal from each voxel is modeled as containing two exponential functions. It is argued in these papers that bi-exponential modeling can be superior to mono-exponential modeling in diagnosis of some musculo-skeletal disease since two types of water are usually present in connective tissues and each causes a different T_2^*. However, generally, mono-exponential fits the signal sufficiently well while keeping a low computational complexity, making it the most common modeling used for T_2^* relaxation mapping and other similar applications. For T_2 relaxation mapping, many different models have been employed, such as multi-exponential modeling [90, 113], continuous distribution modeling [59, 112], and pulse-sequence-dependent modeling [21, 93, 94]. Also, T_2 relaxation mapping uses spin echo sequences which are more complicated than gradient echo sequences. In this work, we only focus on reconstruction of spin density, R_2^* relaxation rate, and off-resonance frequency maps.

1.2.2 Signal acquisition

Our objective is to reconstruct the spin density, R_2^* relaxation rate, and off-resonance frequency maps. Let the size of each map (image) be $N \times N$, where N is typically a power of 2, such as 64, 128, or 256. The number of complex unknowns for this reconstruction problem is $2N^2$. To make the problem well-posed, one needs at least $2N^2$ complex data that are sampled from k-space. There are several ways to acquire the data. The clinically mostly used method is multi-echo sampling. This sampling approach basically samples multiple k-space frames each at a different echo time. The number of k-space frames needs to be at least 2 to satisfy the $2N^2$ requirement; however, the actual number of echoes used is usually larger than 2 to improve SNR in the reconstructed maps. Popular choices include 6, 8, 9, and 16 echoes; this number may vary with applications and tissues of interest [4,5,26,73,111,114]. Since multiple k-space frames are usually used, the time to finish all sampling is relatively long, ranging from tens of seconds to minutes for one image slice [4,23,26,73,114]. After acquiring all the k-space frames, they are inverse Fourier transformed to a sequence of decaying images. The three parameter maps can then be reconstructed by fitting the mono-exponential model (1.11) to the sequence of decaying images. Most applications so far only require reconstruction of the R_2^* decay map and the spin density map; however, the data processing is exactly the same as all three except that only the absolute value of each image is needed.

A method to shorten the data acquisition time is to undersample the multi-echo k-space frames [7, 22, 51, 58, 72, 107, 119]. Since each k-space frame is undersampled, the acquisition time for multiple k-space frames is reduced. The undersampling is achieved by schemes such as variable-density random phase encoding, EPI, or other non-Cartesian trajectories like spiral, radial, and rosette. Although these techniques considerably accelerate the data acquisition, the undersampling causes the reconstruction of each k-space frame to become underdetermined. Solutions to the problem include parallel imaging [39, 86] and compressed sensing [67]. The former approach is straightforward and is not in the scope of our research. The latter approach is usually a direct extension of ordinary compressed sensing MRI to

relaxation mapping. The simplest method is just to enforce a sparsity constraint on each image frame [119]. More advanced methods include the time dimension and exploit sparsity of the image sequence in the temporal domain [22, 107] or even simultaneously in both spatial and temporal domains [72]. Although these methods have been demonstrated to be quite successful, the undersampling rate of k-space is usually in a moderate range such as 4-6. A higher undersampling rate can usually cause visible noise-like artifacts in the reconstructed maps. Due to the rapid development of the compressed sensing theory, a number of approaches can be used to solve this problem and their implementations are also well known.

As a similar idea to undersampling, reconstruction based on single- or multi-shot trajectories has attracted increasing interest [83, 95, 102, 103]. These methods discard the idea of acquiring multiple k-space frames; instead, they use a long readout time in each readout cycle to expand their sampling to cover a long time frame. At each time point, only one or a few k-space samples are acquired, depending on the number of single-shot trajectories used. However, the sampling density in the time domain becomes much larger compared to multi-echo sampling. Such sampling can be done in tens of milliseconds, dramatically reducing the acquisition time. The challenge, however, lies in the exceedingly difficult image reconstruction problem. We have observed the problem to be ill-conditioned, nonlinear, and of large scale. This work thus focuses on algorithms developed to stably and rapidly solve the problem.

1.2.3 Problem formulation

Let $m(\vec{r})$, $d(\vec{r})$, $f(\vec{r})$ represent the spin density, the R_2^* decay and the off-resonance frequency of a voxel located at position $\vec{r}$ (relative to the center of the field of view). The time-varying signal at this voxel is modeled as

$$m(\vec{r})e^{-(d(\vec{r})+\iota f(\vec{r}))t_l} \tag{1.12}$$

where ι represents the imaginary unit. By defining $z(\vec{r}) \triangleq d(\vec{r}) + \iota f(\vec{r})$, we rewrite (1.12) as

$$m(\vec{r})e^{-z(\vec{r})t_l} \tag{1.13}$$

We call $z(\vec{r})$ the complex frequency at location $\vec{r}$. Let $\vec{k}_{lp}$ denote a multi-shot trajectory where $l = 1, 2, \ldots, L$ and $p = 1, 2, \ldots, P$ denote the lth time sample and the pth trajectory shot among L samples and P shots. When a single-shot trajectory is used, then $P = 1$. The acquired k-space data from a multi-shot trajectory can then be represented by

$$\begin{aligned} y_{lp} &= s_{lp}(m(\vec{r}), z(\vec{r})) + \varepsilon_{lp} \\ s_{lp}(m(\vec{r}), z(\vec{r})) &= \int_{\vec{r}} m(\vec{r})e^{-z(\vec{r})t_l} e^{-2\pi\iota(\vec{k}_{lp}\cdot\vec{r})} d\vec{r} \end{aligned} \tag{1.14}$$

for $l = 1, 2, \ldots, L$ and $p = 1, 2, \ldots, P$. y_{lp}, s_{lp} and ε_{lp} represent the acquired k-space data, the predicted k-space signal, and the noise at time t_l and shot p, respectively. The dot operator in the exponential term represents inner product. Given data y_{lp} for L discrete time points and P trajectory shots, we want to estimate $m(\vec{r})$ and $z(\vec{r})$ for every voxel in the FOV. To proceed, (1.14) requires a suitable discretization in the spatial domain since data is discrete while unknown maps are (piecewise) continuous. We therefore introduce a finite-dimensional representation for $m(\vec{r})e^{-z(\vec{r})t_l}$ so that

$$m(\vec{r})e^{-z(\vec{r})t_l} \approx \sum_n m_n e^{-z_n t_l} g(\vec{r} - \vec{r_n}) \tag{1.15}$$

where r_n is the nth voxel geometric center, and $m_n \triangleq m(\vec{r}_n)$, $z_n \triangleq z(\vec{r}_n)$. Function $g(\vec{r})$ is chosen as a linear interpolation basis function in our work. With the approximation (1.15), we can rewrite (1.14) as

$$s_{lp}(m_n, z_n) \approx \tilde{g}_{lp} \sum_{n=0}^{N-1} m_n e^{-z_n t_l} e^{-2\pi\iota(\vec{k}_{lp}\cdot\vec{r}_n)} \tag{1.16}$$

where $\tilde{g}_{lp} \triangleq \tilde{g}(\vec{k}_{lp})$ represents the sample of the Fourier transform of $g(\vec{r})$ at $\vec{k}_{lp}$. Because the noise in (1.14) is Gaussian, we estimate the discrete unknowns based on minimizing a least-squares objective function

$$\|\vec{y} - \vec{s}(\vec{m}, \vec{z})\|^2 \tag{1.17}$$

where

$$\begin{aligned}
\vec{y} &\triangleq [y_{11}, y_{21}, \ldots, y_{L1}, y_{12}, \ldots, y_{L2}, \ldots, y_{LP}]^T \\
\vec{s} &\triangleq [s_{11}, s_{21}, \ldots, s_{L1}, s_{12}, \ldots, s_{L2}, \ldots, s_{LP}]^T \\
\vec{m} &\triangleq [m_1, m_2, \ldots, m_N]^T \\
\vec{z} &\triangleq [z_1, z_2, \ldots, z_N]^T
\end{aligned}$$

Since the function (1.17) is usually ill-conditioned [47], we apply regularization with respect to the unknown maps, which changes (1.17) to:

$$\|\vec{y} - \vec{s}(\vec{m}, \vec{z})\|^2 + \lambda_1 \|D_1 \vec{m}\|^2 + \lambda_2 \|D_2 \vec{z}\|^2 \tag{1.18}$$

where the matrices D_1 and D_2 are the regularization matrices corresponding to $\vec{m}$ and $\vec{z}$. For simplicity, we use the first-order smoothness penalty for both D_1 and D_2 in this work. Separate regularizations on the real and imaginary parts of $\vec{z}$ have also been utilized by other authors [83]. In the next section, we focus on algorithms that can stably and rapidly minimize (1.18) with respect to $\vec{m}$ and $\vec{z}$.

1.2.4 Application to functional MRI

Functional magnetic resonance imaging (functional MRI or fMRI) is a functional neuroimaging technique that reflects the neural activities of human brains using MRI. fMRI has been developed over two decades, and currently this technique is widely used in neuroscience

research and even clinical medicine [53, 65]. Many image contrast mechanisms have been developed to show neural activities, but the most primary form of contrast is the blood-oxygen level dependent (BOLD) contrast [66]. This contrast mechanism takes advantages of the fact that neural activities cause variations of the local R_2^* decay rate. Since the R_2^* decay rate is an unknown variable in our problem, any algorithms that can solve the joint reconstruction problem can be applied to BOLD fMRI.

fMRI works by utilizing an important mechanism in the human brain called neurovascular coupling, which was discovered more than a century ago. Neurovascular coupling means that increased neural activities in a small part of the cerebrum cause an increased cerebral blood flow around that area. More specifically, the number of oxygenated hemoglobins in the cerebral blood flow around the area will increase following the increased amount of neural activity to provide more oxygen. Oxygenated hemoglobin is less paramagnetic than deoxygenated hemoglobin. In fact, oxygenated hemoglobin is about 20% less susceptible to an external magnetic field than deoxygenated hemoglobin. Therefore, when a subject is placed in an MRI scanner and his/her neural activity starts to increase in some part of his/her cerebrum, the increased oxygenated hemoglobin causes the local magnetic field to be more uniform and hence reduces the local R_2^* decay rate. This reduction of the R_2^* decay rate can be captured by MRI in T_2^*-weighted images, where the excited area becomes brighter. By acquiring multiple R_2^*-weighted images with a high temporal resolution, one can observe this variation of the image intensity and localize the excited area.

If one ignores the effect of off-resonance during the MR imaging process, the images acquired at a particular time t can be represented by

$$s_t(\vec{r}) = m(\vec{r})e^{-d(\vec{r})t} \tag{1.19}$$

where $m(\vec{r})$ and $d(\vec{r})$ represent the spin density map and the R_2^* decay map. The T_2^*-weighted imaging sets the time point t so that the resultant image s_t is contrasted by the T_2^*-weighted

spin density, where $T_2^* = 1/R_2^*$ is the time constant associated with the R_2^* decay. The time t cannot be chosen to be zero, since then the contrast would be a function only of the spin density. The time t cannot be too big either since then all voxels would have a small signal and the contrast would be poor. Depending on the R_2^* decay rate distribution of the imaged object, there is an optimal choice for the t so that the resultant image is most heavily influenced by the variation of the R_2^* decay. This imaging method is called the T_2^*-weighted imaging. Most BOLD contrast based fMRI utilizes this imaging method to reflect the temporal variation of the R_2^* decay rate.

BOLD functional MRI is an important application of joint reconstruction of spin density, R_2^* decay, and off-resonance maps mainly because R_2^* decay is the biomarker of the BOLD effect. In addition, the T_2^*-weighted imaging method is an indirect method in the sense that it only reconstructs the R_2^* decay weighted spin density instead of R_2^* decay itself. An important assumption in T_2^*-weighted imaging based fMRI is that the spin density must remain unchanged. However, if one estimates $\vec{m}$ and $\vec{z}$ in (1.14) simultaneously for fMRI, this assumption can be relaxed. This difference makes any practical method solving the problem in (1.14) valuable to fMRI due to the robustness of the R_2^* estimate to the variation of spin density, which can be caused by various factors such as inflow [34]. Inflow refers to the blood flow in a vessel that passes through the imaging slice. During the imaging process, some fresh blood will flow in and push the old flow out of the imaging slice, causing a change in the excited spin density. This variation for some scans is very strong compared to the actual fMRI signal and can therefore dramatically increase the error of fMRI data analysis.

1.3 Literature review

Overall, the published literature on joint reconstruction of spin density, R_2^* decay and off-resonance maps through a single- or multi-shot trajectory is quite sparse. This may be due to the various technical challenges of this problem, such as trajectory miscalibration and sequence imperfection, which are explained in later chapters. Although papers directly

addressing the reconstruction problem are sparse, a considerable number of papers address similar problems. Some of these papers offer important insights into our problem and help us in developing our reconstruction algorithms. The literature review begins with these papers.

All reconstructions that assume a prior model for the acquired signal can be categorized as model-based image reconstruction [28]. The model is usually proposed based on some simplified physical equations, such as the mono-exponential model we use throughout this work [96]. Most work that has used this model has attempted to reconstruct one or two parameters maps by assuming the remaining maps are known. An iterative algorithm is typically necessary due to the nonlinearity of the model.

1.3.1 Multi-echo sampling

Literature can be categorized based on the method used for acquisition. Most previously mentioned algorithms used continuous sampling in k-space, i.e. some k-trajectories such as EPI, spiral, radial, or rosette. In this and following sections, we review methods that have used full multi-echo sampling and undersampled multi-echo sampling. These methods are the mainstream methods for reconstruction of the R_2^* relaxation rate and are mainly applied to BOLD functional MRI or detection of iron overload in the body.

Most clinical R_2^* mapping protocols are based on the full multi-echo sampling strategy [4, 5, 26, 58, 73, 111, 114]. These methods usually follow a two-step reconstruction. In the first step, they reconstruct all images based on inverse Fourier transform. In the second step, they fit the mono-exponential model to the sequence of images. In [5], Barth et al. proposed a multi-echo spiral imaging sequence to sample k-space with eight single-shot spiral trajectories, each taking less than 25 milliseconds. The sequence was applied to a finger tapping fMRI experiment and was able to distinguish between BOLD signal and inflow effects. Wennerberg et al. compared R_2^* mapping to traditional fMRI image reconstruction and concluded that R_2^* mapping is more resistant to inflow effects and has a higher functional contrast; however, the activation volume of R_2^* mapping is smaller. Nevertheless, they only

used two-echo spiral imaging, with one echo at 20 ms and another at 90 ms. The low echo number and large interval between echoes can cause strong noise which can be much improved by having more echoes. Anderson et al. used a multi-echo T_2^* mapping to quantify iron overload in myocardium [4]. They used eight different echo times in reconstructing a T_2^* map, and the result provided an effective tool for quantifying the need for tricular dysfunction treatment. There are many other similar applications of multi-echo sampling in R_2^* or T_2^* parameter mapping. Most of them use 6-8 TEs and provide a gold standard for the undersampled multi-echo approach.

1.3.2 Undersampled multi-echo sampling

The drawback with full multi-echo sampling is its time consumption. Since usually 6-8 echoes are needed, a full multi-echo sampling takes at least tens of seconds. Using spirals or random EPI may reduce the acquisition time, but the induced undersampling can cause artifacts. To address this problem, many researchers discarded the two-step procedure used in full multi-echo sampling. Instead, they combined either the mono-exponential modeling or a sparsity constraint with the inverse Fourier transform [7, 22, 51, 72, 93, 107, 119]. In [7] and [93], Block et al. and Sumpf et al. proposed a radial and an undersampled Cartesian spin echo multi-echo sequence to reconstruct T_2 relaxation time, respectively. The undersampling rate used was 6-8, and l_2 regularization was used to enforce the estimate smoothness. Huang et al. proposed a multi-echo radial sequence whose undersampling rate is around 4. They also transformed the nonlinear curve fitting for the mono-exponential modeling to a linear inversion problem based on principle component analysis (PCA) of the exponential functions. This technique was also used in [22, 107] and is similar to the idea of MR fingerprinting [17,70]. Regularization in the temporal domain or simultaneously in the spatial and temporal domains is very common among papers that used undersampled multi-echo sampling [22, 51, 72, 107]. This inspired our work in the third chapter, and such regularization is very important in our algorithm. Compressed sensing (CS) has also been frequently used for this

type of method, where the regularization is based on sparsity of the signal in some transform domain [51, 72, 107, 119]. The use of CS in the multi-echo approach is somewhat similar to its use in dynamic MRI [105].

1.3.3 Reconstruction of a single parameter map

Literature can also be categorized based on the number of reconstructed parameter maps. Some work only addresses reconstruction of one parameter map. These algorithms often assume one parameter map is known and aim to reconstruct another map [15, 29, 31, 32, 97, 117]. For example, Sutton et al. presented a fast algorithm to reconstruct the spin density map with a known off-resonance map [97]. The fast computation is realized by time domain segmentation, which facilitates use of the nonuniform fast Fourier transform (NUFFT). Later, Fessler et al. proposed another fast algorithm dealing with the same problem [29]. The work generalizes the time-domain segmentation technique in [97] and combines it with Toeplitz-based matrix-vector multiplication which is much faster than regular matrix-vector multiplication. Both methods used conjugate gradients to solve a linear optimization problem, and the latter work showed that CG-Toeplitz is considerably faster than CG-NUFFT. Both papers used a single-shot spiral trajectory with 18.8 ms of readout time to acquire k-space data. Since these two techniques are very efficient, they have been used by many authors in reconstruction of more parameter maps [49, 83]. Similar work in reconstructing the spin density map with a known field map include [15, 117]. In [117], Zahneisen et al. generalized the CG-NUFFT algorithm to 3D image reconstruction by using a concentric shells 3D trajectory. In [15], Compton et al. proposed a hybrid regularization based on total variation and framelet norm for the same problem and presented a split Bregman algorithm to efficiently solve the resultant convex optimization problem. These techniques were mostly published from 2000 to 2010 and are therefore foundational for much later work on jointly reconstructing multiple parameter maps. Their insights in efficiency, regularization, and trajectory design have had an important impact on later work.

1.3.4 Joint reconstruction of two parameter maps

Some authors have attempted to reconstruct two parameter maps out of the three [10, 11, 74–76, 80, 82, 83, 98]. These problems are very similar to our problem, and some of them can be easily adapted to reconstructing three parameter maps. Sutton et al. proposed an algorithm to reconstruct the spin density map and the off-resonance map from a spiral-in/spiral-out trajectory [98]. The spiral-in/spiral-out trajectory is a succession of a spiral-in trajectory and a spiral-out trajectory. The authors claimed the trajectory to be superior due to its efficient sampling and the closely spaced TEs. They used a gradient descent algorithm and employed the acceleration technique in [97]. Other researchers proposed a similar signal model whose off-resonance map also includes the through-plane gradient [75, 76]. However, this algorithm reconstructs multiple slices together, which requires a large variable size and long computation. Olafsson et al. proposed a gradual refinement algorithm to jointly reconstruct the R_2^* decay map and the off-resonance map [83]. The spin density map must be provided by separate pulse sequences. The author demonstrated that the algorithm converges well when a good initialization is present. However, in the presence of poor initializations, we found the algorithm has a very low success rate in our experiments [49]. Another category of methods for reconstructing spin density and off-resonance maps is called automatic off-resonance correction. These methods are inspired by the conjugate phase algorithm [71] and assume the spin density map is completely real. Since this is usually not the case, a low-pass filtering is applied to preprocess the data. The off-resonance at each voxel can be reconstructed by minimizing the imaginary part of the reconstructed image. These algorithms are mostly noniterative and therefore very fast. However, they usually assume off-resonance is linearly varying in the image space which does not conform to the actual case and possibly causes errors. Also, these algorithms cannot be extended to include reconstruction of relaxation, since relaxation causes changes in the real part of the image. Overall, we found much more attention given to joint reconstruction of the spin density and field maps than joint reconstruction of the spin density and R_2^* relaxation.

From our experience, we believe reconstructing the R_2^* relaxation map is much more difficult than the off-resonance map.

1.3.5 Joint reconstruction of the three parameter maps

Although joint reconstruction of spin density, R_2^*, and off-resonance maps may be of the highest difficulty among all model-based approaches and is very sparse in the literature, this method was proposed in relatively early days. In 2003, before most of the previously mentioned model-based methods were published, Twieg proposed a conceptual approach called "PARSE", which is exactly the joint reconstruction of the three parameter maps [102]. He also pointed out that the rosette is a good k-trajectory candidate for this problem due to its dense sampling around the k-space origin. The reconstruction algorithm he used is a gradient descent algorithm which took hours to reach a satisfactory result. Properties of convergence for this problem were not well analyzed in the paper. Continuation of this work appeared in 2010, by Twieg and Reeves [103]. They proposed a modified conjugate gradient algorithm as a replacement of the gradient descent algorithm and analyzed many properties of the algorithm. They adopted a continuation framework on data length to avoid local minima and to increase convergence speed. They reported that the proposed conjugate gradient algorithm was able to yield a satisfactory estimate in a few minutes. However, the robustness of their method remains unclear, and the progressive data length technique is highly tailored specifically for this problem. Another early work on this topic is from Sutton et al. [95]. However, the work lacked thorough analysis and experiments and was not followed up by a more comprehensive journal paper. All these papers used gradient-based methods to address the underlying nonlinear optimization problem.

1.3.6 Algorithm usage

In previous sections, we analyzed the model-based approaches based on their sampling patterns and reconstruction objectives. In this section, we categorize the algorithms developed for all undersampling-based reconstruction problems. All algorithms described in the Section 1.3.2 to 1.3.5 used some kind of undersampling strategies. Most of these algorithms can be classified into two categories: gradient-based algorithms and alternate minimization algorithms. Gradient-based algorithms, such as those used in [7, 93, 119] in Section 1.3.2, [29, 97] in Section 1.3.3, [75, 76, 83, 98] in Section 1.3.4, and [95, 102, 103] in Section 1.3.5 used many classical unconstrained optimization algorithms, such as gradient descent and conjugate gradients. Their performance is well predicted by established theory in unconstrained optimization. Since these methods need to calculate a gradient in every iteration, they usually call for some fast computation techniques such as NUFFT and Toeplitz matrix-vector multiplication to reduce the computational burden. The algorithms are usually not scale independent; that is, their computation time increases linearly with the increase of dimensionality. If the image size grows from 64×64 to 128×128, the computation time quadruples.

Another type of algorithm used the alternate minimization method or variable splitting method, which is usually seen in convex optimization literature. This kind of algorithm was used in [58] in Section 1.3.1, and in [22, 51] in Section 1.3.2. Although they were used much less frequently than gradient-based methods, they were so different from the gradient-based methods and we believe they can be faster and more stable than gradient-based methods, especially for reconstruction of multiple parameter maps. Usually, these methods were adopted in multi-echo (full or sparse) sampling based approaches. However, we show that this method can also be applied to continuous sampling. More details about the alternate minimization method will be given in the third chapter.

1.3.7 Regularization usage

All approaches that undersample the k-space frames must use some sort of regularization. Most algorithms in this context used Tikhonov regularization [78]. The regularization is applied to the parameter maps of interest to enforce smoothness. Tikhonov regularization is especially a good fit for field map reconstruction because a field map is usually smooth except at some boundaries of two different tissues. When estimates of multiple parameter maps are needed, regularization can be applied uniformly or separately to each of the parameter maps. Since the off-resonance map is usually smoother than the other two maps, separate regularization over each parameter map may lead to a better reconstruction [83].

When CS is used in reconstruction of these parameter maps, other regularization methods may be used, such as total variation [15], l_1 norm in some transform domain [15,51,119], and regularization that promotes group sparsity [72,107]. The latter regularization promotes the correlation between different k-space frames since they are related by the parameter maps. Although a direct comparison between Tikhonov regularization and the sparsity-promoting regularization for the joint reconstruction problem is still lacking, it is reasonable to believe the latter regularization is superior to the former one since the spin density and R_2^* maps may not be very smooth.

No matter what regularization is used, regularization parameter choice is a problem. Since there are multiple regularization terms, multiple parameters are present and a fine tuning is not easy to achieve. However, so far most algorithms have focused on developing algorithms, and parameters have usually been selected by trial-and-error. Several authors have used a point spread function analysis to make sure the smoothness induced by the regularization is within a reasonable range [31, 32, 97]. L-curve method [44] has also been used in some papers [117]. Overall, a method for choosing an optimal set of parameters is still an open problem for the nonlinear optimization problem.

1.3.8 Related literature

Some problems are closely related to the joint reconstruction of multiple parameter maps. These problems are either similar to the latter problem, or compose part of the latter problem, or have a similar purpose with the latter problem. At the end of this chapter, we briefly review literature on these problems.

Coil sensitivity mapping

Coil sensitivity profiles are important to parallel imaging [39, 86]. The simplest method to obtain the sensitivity maps is to divide the images from the coil array by the reference image acquired from a single-channel body coil. Since a division induces a lot of noise while sensitivity maps are usually smooth, some kind of smoothing needs to be used [86]. The model-based approach can also be applied to reconstruction of sensitivity maps [28]. These methods usually involve the sensitivity map into modeling of the acquired k-space data and then solve for the map using iterative algorithms [2, 3, 57]. Joint reconstruction of coil sensitivity maps and the spin density map has also been investigated [100, 104, 116]. The model-based approach relates coil sensitivity mapping to our problem since their forward models are very similar. Some of the algorithms used in model-based coil sensitivity mapping can be adapted to our problem, such as [3].

T_1 & T_2 mapping

T_1 & T_2 mapping are important to our problem in many ways. First, most methods for T_1 & T_2 mapping are also model-based; some of them also used iterative algorithms which can inspire algorithm development for our problem [21, 93, 94]. Second, it is also interesting to extend algorithms for T_2^* mapping to T_1 or T_2 mapping to simultaneously reconstruct multiple relaxation rate maps [62]. However, in T_1 & T_2 mapping, the models can be much more complicated than mono-exponential due to the presence of a composite component (such as water) and pulse sequence design [59, 90, 112, 113].

Trajectory calibration

When a non-Cartesian trajectory is used in reconstruction, using trajectory calibration to find out the "true" trajectory can reduce the induced estimation error. This is especially the case when the forward model is ill-conditioned, such as in undersampling-based reconstruction, since the ill-conditioning would increase the error. One of the most commonly used trajectory calibration techniques is from [25]. Other methods include [19,63,64,99,118]. Compared to these methods, [25] is much easier to implement and does not require use of special phantoms. In our work, we also used the calibration method in [25].

Dynamic MRI

Dynamic MRI refers to real-time MR imaging of moving organs, such as the heart. Dynamic MRI has been investigated for a long time and has recently been clinically used in cardiac imaging [33, 55, 69, 101]. The aim of dynamic MRI is to reconstruct a sequence of images that correspond to a movie of an organ at different time points. Since usually only a small portion of the image is dynamically moving while the rest remains static, compressed sensing can be applied not only to the spatial domain but also the temporal domain. This characteristic is very similar to relaxation mapping in that a sequence of mutually correlated images is to be reconstructed. Therefore, the algorithms used in dynamic imaging can be good references for algorithm development in our problem [52, 72].

Chapter 2

Trust-region methods

In this chapter, we present two trust-region methods based on two different linearization strategies for the nonlinear signal model. A trust region is defined as a local area in the variable space where a local linear approximation is trustworthy. In each iteration, the method minimizes a local approximation within a trust region so that the step size can be kept in a suitable scale. A continuation scheme is applied to gradually reduce the regularization over the parameter maps and facilitate convergence from poor initializations. The two trust-region methods are compared to two other previously proposed methods—the nonlinear conjugate gradients and the gradual refinement algorithm. Experiments based on various synthetic data and real phantom data show that the two trust-region methods have a clear advantage in both speed and stability.

2.1 Formulation of the cost function

Recall from Section 1.2.3 that we want to minimize

$$\|\vec{y} - \vec{s}(\vec{m}, \vec{z})\|^2 + \lambda_1 \|D_1 \vec{m}\|^2 + \lambda_2 \|D_2 \vec{z}\|^2 \tag{2.1}$$

with respect to $\vec{m}$ and $\vec{z}$, where

$\vec{y} \triangleq [y_{11}, y_{21}, \ldots, y_{L1}, y_{12}, \ldots, y_{L2}, \ldots, y_{LP}]^T$ represents the acquired k-space data

$\vec{s} \triangleq [s_{11}, s_{21}, \ldots, s_{L1}, s_{12}, \ldots, s_{L2}, \ldots, s_{LP}]^T$ represents the predicted k-space data

$\vec{m} \triangleq [m_1, m_2, \ldots, m_N]^T$ represents the spin density map

$\vec{z} \triangleq [z_1, z_2, \ldots, z_N]^T$ represents the complex frequency map

and

$$s_{lp}(m_n, z_n) \approx \tilde{g}_{lp} \sum_{n=0}^{N-1} m_n e^{-z_n t_l} e^{-2\pi\iota(\vec{k}_{lp}\cdot\vec{r}_n)} \tag{2.2}$$

In (2.1), D_1 and D_2 are the regularization matrices corresponding to $\vec{m}$ and $\vec{z}$. In (2.2), $\tilde{g}_{lp} \triangleq \tilde{g}(\vec{k}_{lp})$ represents the sample of the Fourier transform of $g(\vec{r})$ at $\vec{k}_{lp}$, and $g(\vec{r})$ is the interpolation function of $m(\vec{r})e^{-z(\vec{r})t_l}$ for each l.

2.2 Proposed trust-region methods

In this section, we present two trust region (TR) methods [92], the ordinary trust-region method (OTR) and the change-of-variable trust-region method (CVTR) due to their different linearization strategies.

2.2.1 The ordinary trust-region method

Approximating the local cost function

Finding a quadratic approximation of the nonquadratic cost function is the first step in utilizing a trust-region method. Let m_{0n}, z_{0n} represent the parameters at a reference voxel and Δm_n, Δz_n the offsets. Taking the first-order Taylor expansion about m_{0n} and z_{0n} in the discrete version of (1.13) leads to the approximation

$$m_n e^{-z_n t_l} \approx m_{0n} e^{-z_{0n} t_l} + \Delta m_n e^{-z_{0n} t_l} - m_{0n} e^{-z_{0n} t_l} \Delta z_n t_l \tag{2.3}$$

where Δm_n and Δz_n are sufficiently small. With this approximation, one then minimizes the following function in each iteration:

$$\min_{\Delta\vec{m},\Delta\vec{z}} \ \|\vec{\zeta} - [\Theta_1 \ \Theta_2][\Delta\vec{m}^T \ \Delta\vec{z}^T]^T\|^2 \qquad \text{s.t. } \|\Delta\vec{m}\|_2 \le \eta_1; \|\Delta\vec{z}\|_2 \le \eta_2 \tag{2.4}$$

where $\eta_1, \eta_2 \in \mathbb{R}^+$ and

$$\begin{aligned}
[\vec{\zeta}]_{(l,p)} &\triangleq y_{lp} - \sum_n m_{0n} e^{-z_{0n} t_l} e^{-2\pi\iota(\vec{k}_{lp}\cdot\vec{r}_n)}; \\
[\Theta_1]_{(l,p),n} &\triangleq \tilde{g}_{lp} e^{-z_{n0} t_l} e^{-2\pi\iota(\vec{k}_{lp}\cdot\vec{r}_n)}; \\
[\Theta_2]_{(l,p),n} &\triangleq -\tilde{g}_{lp} t_l m_{0n} e^{-z_{n0} t_l} e^{-2\pi\iota(\vec{k}_{lp}\cdot\vec{r}_n)}
\end{aligned}$$

where $(l,p) \triangleq (p-1)L + l$. This is a constrained linear optimization problem.

Solving the sub-problem

Equation (2.4) with the regularization in (2.1) is equivalent to

$$\begin{aligned}
&\min_{\Delta\vec{m},\Delta\vec{z}} \ \|\vec{\zeta} - [\Theta_1 \ \Theta_2][\Delta\vec{m}^T \ \Delta\vec{z}^T]^T\|^2 \\
&+\lambda_1\|D_1(\vec{m}_0 + \Delta\vec{m})\|^2 + \lambda_2\|D_2(\vec{z}_0 + \Delta\vec{z})\|^2 \\
&+\sigma_1\|\Delta\vec{m}\|^2 + \sigma_2\|\Delta\vec{z}\|^2
\end{aligned} \tag{2.5}$$

where σ_1 and σ_2 are properly chosen positive numbers. To solve (2.5), we choose preconditioned linear conjugate gradients (PCG) [37]. We use diagonal preconditioners which are defined as the inverse of the diagonal of the Hessian in (2.5). The maximal number of iterations for the sub-problem is set to 40. However, convergence is usually reached in less than 40 iterations with either a small gradient or a low iterate variation.

Summarizing OTR

The following procedure defines one iteration of OTR. Throughout our work, we use $\mu_{1,2,3,4} = 0.60, 2, 0.99, 0.7$.

Procedure 1

- If gradient or iterate variation is sufficiently small, the algorithm is stopped.
- Minimize (2.5) with PCG. Let the solution be $\Delta\vec{m}$, $\Delta\vec{z}$.
- Calculate the ratio γ between the decrease in (2.4) and the decrease in (2.1) caused by the new iterate.
- If $\gamma > 0$ (descent), $\vec{m} \leftarrow \vec{m} + \Delta\vec{m}$ and $\vec{z} \leftarrow \vec{z} + \Delta\vec{z}$.
- If $\gamma < \mu_1$, then $\sigma_1 \leftarrow \mu_2\sigma_1$, $\sigma_2 \leftarrow \mu_2\sigma_2$. If $\gamma > \mu_3$, then $\sigma_1 \leftarrow \mu_4\sigma_1$, and $\sigma_2 \leftarrow \mu_4\sigma_2$.

2.2.2 The change-of-variable trust-region method with regularization

Introducing CVTR

The difference between OTR and CVTR lies in the local approximation. For CVTR, the nonlinear signal model is first approximated by

$$m_n e^{-z_n t_l} \approx m_n e^{-z_{0n} t_l}(1 - \Delta z_n t_l) = m_n e^{-z_{0n} t_l} - e^{-z_{0n} t_l} m_n \Delta z_n t_l \tag{2.6}$$

and then followed by combining $m_n \Delta z_n$ into a new variable c_n

$$m_n e^{-z_n t_l} \approx e^{-z_{0n} t_l} m_n - t_l e^{-z_{0n} t_l} c_n \tag{2.7}$$

Equation (2.7) also explains how the method is named. The approximation leads to a different data fidelity term from the one in (2.4):

$$\|\vec{y} - [\Theta_1 \ \Theta_3][\vec{m}^T \ \vec{c}^T]^T\|^2 \tag{2.8}$$

where

$$[\Theta_3]_{(l,p),n} = -\tilde{g}_{lp} t_l e^{-z_{n0} t_l} e^{-2\pi\iota(\vec{k}_{lp}\cdot\vec{r}_n)}$$

Reformulating the cost function for CVTR

A problem arises when constructing the approximated cost function for CVTR. The regularization of $\vec{z}$ would introduce a nonlinear term in the cost function since $z_n = c_n/m_n$ for the nth voxel. A straightforward solution is to regularize the point-wise product of $\vec{m}$ and $\vec{z}$. This solution leads to a new cost function

$$\|\vec{y} - \vec{s}(\vec{m}, \vec{z})\|^2 + \lambda_3 \|D_1 \vec{m}\|^2 + \lambda_4 \|D_2(\vec{m} \odot \vec{z})\|^2 \tag{2.9}$$

where $\lambda_{3,4} \in \mathbb{R}^+$ and $\odot$ represents the point-wise product between two vectors. The main difference compared to (2.1) lies in the regularization over $\vec{z}$. A new sub-problem also arises for CVTR

$$\begin{aligned} &\min_{\vec{m},\vec{c}} \ \|\vec{y} - [\Theta_1 \ \Theta_3][\vec{m}^T \ \vec{c}^T]^T\|^2 \\ &+\lambda_3 \|D_1 \vec{m}\|^2 + \lambda_4 \|D_2(\vec{m} \odot \vec{z}_0 + \vec{c})\|^2 \\ &\text{s.t. } \|\vec{m} - \vec{m}_0\|_2 \le \eta_3; \|\vec{c}\|_2 \le \eta_4 \end{aligned} \tag{2.10}$$

where $\eta_3, \eta_4 \in \mathbb{R}^+$. Similar to (2.5), we use PCG to solve the equivalent problem of (2.10) in each iteration:

$$\begin{aligned} &\min_{\vec{m},\vec{c}} \ \|y_l - [\Theta_1 \ \Theta_3][\vec{m}^T \ \vec{c}^T]^T\|^2 \\ &+\lambda_3 \|D_1 \vec{m}\|^2 + \lambda_4 \|D_2(\vec{m} \odot \vec{z}_0) + D_2 \vec{c}\|^2 \\ &+\sigma_3 \|\vec{m} - \vec{m}_0\|^2 + \sigma_4 \|\vec{c}\|^2 \end{aligned} \tag{2.11}$$

where σ_3 and σ_4 are properly chosen positive numbers.

Specifying the trust region for CVTR

The trust region of CVTR is less intuitive than OTR. Since $|\Delta b_n| = |c_n/m_n|$ must be bounded above, $|c_n|$ must be bounded above and $|m_n|$ must be bounded below. However, a lower bound for the modulus of a complex number is hard to implement. A more practical method is to set a bound for the variation of $\vec{m}$ like the bound for $\Delta\vec{m}$ in OTR. This bounding method gives the constraints in (2.10).

The two trust regions associated with the two linearization methods bear further discussion. Figure 2.1 illustrates their differences. The trust region of CVTR is nonconvex in the $(\vec{m}, \vec{z})$ domain due to the quotient c_n/m_n. This trust region makes CVTR either slower than OTR when the trust region is small or causes unacceptable updates when the trust region is large. An advantage of the CVTR trust region is that it can be much broader in the spin density direction. This advantage may help stabilize the convergence of the spin density since the variation of the spin density has a small influence on the convergence. This advantage may also speed up the convergence of the spin density when decay and frequency have good initial guesses.

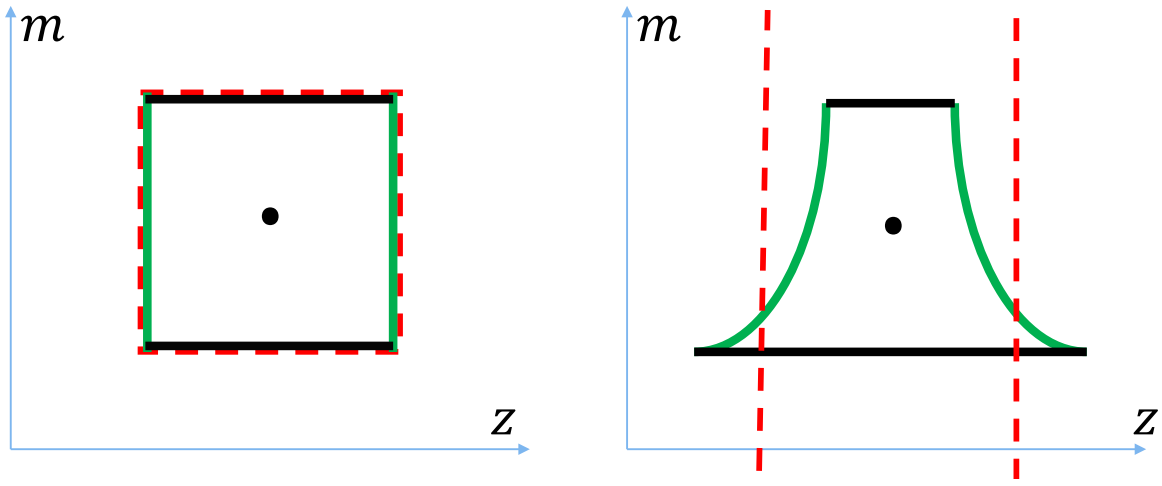


Figure 2.1: Comparison between OTR (left) and CVTR (right) trust regions in a simplified case where m and z are both real scalars. The black and green lines represent trust region constraints associated with m and z, respectively. The red dashed line represents the support for the local linearization.

Summarizing CVTR

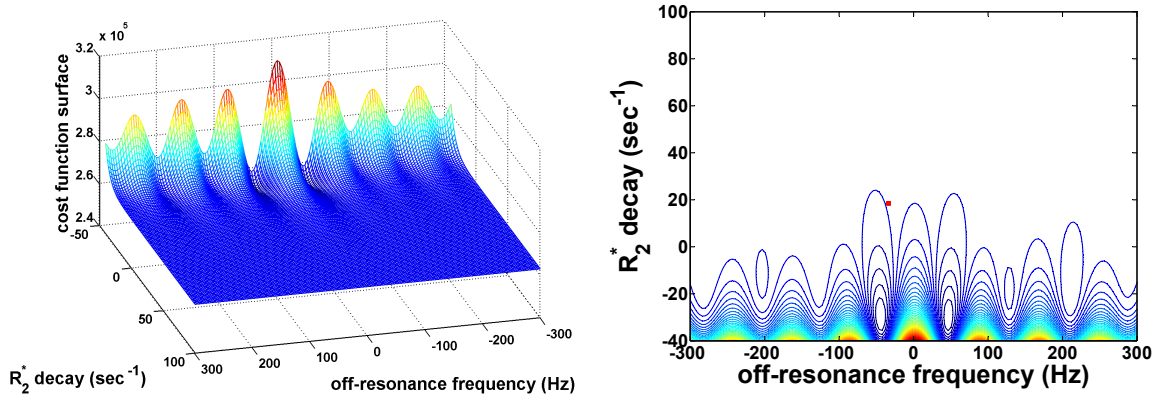
The following defines an iteration of CVTR. $\mu_{1,2,3,4} = 0.60, 2, 0.99, 0.7$ as for OTR.

Procedure 2

- If gradient or iterate variation is sufficiently small, then the algorithm is stopped.
- Minimize function (2.11) with PCG. Let the solution be $\vec{m}_{est}$, $\vec{c}_{est}$.
- Calculate the ratio γ between the decrease in (2.10) and the decrease in (2.9) caused by the new iterate.
- If $\gamma > 0$ (descent), $\vec{m} \leftarrow \vec{m}_{est}$ and $\vec{z} \leftarrow \vec{z} + \frac{\vec{c}_{est}}{\vec{m}_{est}}$.
- If $\gamma < \mu_1$, then $\sigma_1 \leftarrow \mu_2\sigma_1$, $\sigma_2 \leftarrow \mu_2\sigma_2$. If $\gamma > \mu_3$, then $\sigma_1 \leftarrow \mu_4\sigma_1$, and $\sigma_2 \leftarrow \mu_4\sigma_2$.

2.2.3 Ill-conditioning and nonlinearity

We observed the conditioning of jointly estimating the spin density, R_2^* decay and the off-resonance frequency maps out of a single-shot trajectory is very poor [47]. The severe ill-conditioning as well as the nonlinearity of the signal model (1.13) result in a great challenge for each iterative method. This situation is illustrated in Figure 2.2, where the cost function without regularization is plotted by varying R_2^* decay and off-resonance frequency at a reference voxel and holding all other values constant. On one hand, the ill-conditioning mainly exists in the R_2^* decay direction, since a large variation of the R_2^* decay introduces a small variation of the function value. On the other hand, the nonlinearity of the signal model is reflected by the multiple local minima in the axis of off-resonance frequency of this subspace. Such a function profile dramatically increases the challenge for an iterative method.



(a) (b)

Figure 2.2: Cost function without regularization in a neighborhood of $m_n = 0.5, z_n = 0$ for all n. (a) 3D plot of the cost function, and (b) level sets of the cost function in (a). The actual complex frequency z_{n_0} at this voxel is $z_{n_0} = -17.19 \text{ sec}^{-1} - \iota 24.04$ Hz, located at the red square in (b).

2.2.4 Alternative methods for comparison

We use a nonlinear conjugate gradient (NCG) method that we previously developed for the same problem [100]. In each iteration, the method uses an interpolating polynomial to make an inexact line search, followed by a check to see whether the residual decreases. If the residual does not decrease, an exact line search based on a bisection method is applied. This procedure improves the efficiency of the NCG method. For this particular problem, rescaling of the variables was reported to be necessary in order to hold m_n and z_n at the same scale [7]. However, the rescaling parameter is usually empirically selected, and any scale mismatch can cause a poor optimization [51]. In this work, we replace rescaling with a diagonal preconditioning which is defined for this nonlinear forward operator as $\mathbf{J}^H\mathbf{J}$, where $\mathbf{J}$ represents the Jacobian matrix at each iterate and H represents conjugate transpose. We observe that the method speed is significantly improved by the preconditioning. Moreover, preconditioning makes the comparison between NCG and trust-region methods fairer since both use the same preconditioner.

The gradual refinement (GR) method used in our context is inspired by the method developed in [83]. The reference method only reconstructs R_2^* decay and off-resonance frequency maps. In our work, we extend the reference method to simultaneously reconstruct the three parameter maps. In each iteration, the GR method minimizes the approximated cost function in (2.4) without the constraint. The method then accepts the minimizer of the approximated function without a line search (i.e. step length is always set to 1). [83] states that the method converges well when a good initialization is present. In our work, we observe that the GR method has a similar behavior. However, GR can sometimes fail the reconstruction when initialization is poor, which makes the algorithm behavior unpredictable. This characteristic is compared to the trust-region methods to demonstrate their advantage in stability for this problem.

2.2.5 Continuation methods

Solving an ill-conditioned nonlinear equation system from poor initializations usually requires a continuation method [79, 109]. The continuation method embeds the original problem into a one-parameter series of problems where the complexity of the function surface and therefore the estimation difficulty monotonically increase. The original problem is set to be the last problem of this series with the previous solution used as the initialization, so the sequence of solutions converges to the solution of the original problem. Examples of this method include the fixed-point continuation method (FPC) devised for sparse reconstruction [42] and the progressive data length method in [103].

In our work, the regularization parameters must be set high when the algorithm starts from a distant initial guess so that local minima can be avoided (Figure 2.2). The strong regularization causes severe oversmoothing artifacts in the estimates. These artifacts are then reduced by gradually reducing the regularization parameters. Each set of the parameters is called a continuation phase. The continuation method dramatically increases the probability of convergence from a distant initialization for all algorithms. For all our experiments,

the algorithms actually used are the trust region continuation methods, conjugate gradient continuation method (CGC), and the gradual refinement continuation method (GRC). The trust region continuation methods include the ordinary trust region continuation method (OTRC) and the change-of-variable continuation method (CVTRC).

Selecting an optimal continuation scheme is not straightforward. In this work, we use the same continuation scheme for all methods, and this continuation scheme is found in a trial-and-error manner. Specifically, we first determine the regularization parameters based on the object smoothness and the noise level (e.g., running OTR with a good initialization for varying regularization parameters). We then choose a suitable number of continuation phases based on the quality of the initialization. The closer the initialization is to the minimizer, the fewer continuation phases we need for convergence. We then find out the reduction factor for all regularization parameters by trial-and-error.

Procedure 3

- Set λ_1, λ_2 (λ_3, λ_4 for CVTRC)
- For $j = 1, \ldots, J$ (outer loop)
  - Set $\sigma_{1,2}$
  - For $i = 1, \ldots, I(j)$ (middle loop)
    - Run Procedure 1 or Procedure 2
  - End
  - $\lambda_1 \leftarrow \lambda_1/\xi_1$; $\lambda_2 \leftarrow \lambda_2/\xi_2$ or $\lambda_3 \leftarrow \lambda_3/\xi_1$; $\lambda_4 \leftarrow \lambda_4/\xi_2$
- End

2.2.6 Choices of parameters

CVTRC and OTRC are nested algorithms of three program loops. The outer and middle loop are shown in Procedure 3; the inner loop is the PCG loop in Procedures 1 & 2. In the outer loop, we need to initialize $\lambda_{1,2}$ or $\lambda_{3,4}$ (regularization parameters), assign ξ_1 and ξ_2 (regularization reductions in each continuation phase), and set J (number of continuation phases). We also need to initialize $\sigma_{1,2}$ before each middle loop. $\lambda_{1,2}$ or $\lambda_{3,4}$ are somewhat dependent on the object and noise level; however, the dependence is continuous and rather insensitive to many different objects. For example, a phantom and a human brain slice require about the same regularization parameters based on our experiments, since they are similarly smooth. $\xi_{1,2}$ and J can be predetermined; in fact, in all our experiments $\xi_1 = 10$, $\xi_2 = 6$ and $J = 4$. σ_1 and σ_2 are initialized to 10^4 and 10^2 respectively in the first continuation phase for all experiments. After that, they are automatically updated based on Procedures 1 & 2 and passed on to the next continuation phase when j increases.

The middle loop consists of $I(j)$ iterations of Procedure 1 (for OTRC) or Procedure 2 (for CVTRC). In our work, $I(1), I(2), I(3), I(4)$ are set to be (30, 10, 10, 5) for both OTRC and CVTRC. We set $I(1)$-$I(4)$ differently to avoid too many iterations for each continuation phase. Usually, the two algorithms activate the stopping rule before the iteration limits. Even when that does not happen, these numbers are large enough to obtain a significant improvement in the estimate. The main parameters in the inner loop are $\mu_{1,2,3,4}$. Choices of these parameters can be found in the standard literature on trust-region methods [13, 92], since they are a part of the algorithm. Since NCGC and GRC address the same cost function (2.1) as OTRC, we use identical regularization parameters and continuation parameters for these two comparison algorithms.

.

2.3 Real data reconstruction

The ultimate goal of the trust region algorithms is to reconstruct from real MRI data with a minimal number of trajectories. An important problem with real data is lack of ground truth. In this work, we used a multi-echo gradient echo sequence to estimate the three parameter maps for a cylindrical phantom and treated the estimates as the gold standard [4,5,26,73,111,114]. The parameters for the multi-echo gradient echo sequence are: TE = 5, 6, 7, 8, $\cdots$, 82 ms, TR = 200 ms, image size = 64×64, bandwidth = 390 Hz/Pixel, FOV = 120 mm $\times$ 120 mm, slice thickness = 2 mm, flip angle = 15°. We used a long echo train for phantoms to reduce noise. We employed a 32-channel head coil but only used one channel of data for the reconstruction. We chose the channel with the most uniform sensitivity. After acquiring all k-space frames, we applied an inverse Fourier transform to every frame and then a curve fitting on a voxel-by-voxel basis to reconstruct the three parameter maps.

Theoretically, the data from any single-shot trajectory should be approximately equal to $\vec{s}(\vec{m}, \vec{z})$ in (1.17) with $\vec{m}$, $\vec{z}$ given by the estimates from the multi-echo approach. However, the difference between the actual k-space data and model k-space data is very large in our findings. The large model mismatch causes the reconstruction for all algorithms to be suboptimal. The reason is still under investigation. Possible reasons include hidden inconsistency between the multi-echo gradient echo sequence and the single-shot rosette trajectory, trajectory miscalibration, and perhaps some model limitations such as imperfect modeling for intra-voxel gradients. We have especially paid attention to trajectory miscalibration because it is a crucial factor within the model (2.2). We used the calibration technique proposed in [118] to reduce the miscalibration, but the precision level may be insufficient for this reconstruction problem.

The trajectory we used in our experiment is a rosette trajectory. This trajectory has been verified by others to be superior in reconstruction quality to other single-shot trajectories such as spirals and echo planar imaging (EPI) trajectories for this problem [60, 102].

Its formulation is shown below

$$\vec{k}_l = \frac{1}{2} k_{max} \sin(w_{osc} t_l) e^{\iota w_{rot} t_l + \theta} \tag{2.12}$$

where k_{max} denotes the range of k-space, w_{osc} the oscillation frequency, w_{rot} the rotation frequency, and θ the initial angle of the trajectory in the complex domain. In our work, w_{osc} = 3196 rad/sec, w_{rot} = 1577 rad/sec, time span of the trajectory is 81.92 ms and a total of 8192 samples are acquired. When a single-shot rosette is used, θ=0°. However, since the model mismatch is large, we used multi-shot rosette trajectories for reconstruction to improve the conditioning and make the result less sensitive to modeling error. In particular, we used four interleaved single-shot rosettes with θ set to 0°, 22.5°, 45°, and 67.5°, respectively. We emphasize that the focus of the real data reconstruction is on comparing the four algorithms in terms of their convergence performance rather than their absolute accuracy relative to a gold standard.

2.4 Simulations

In this section, we show reconstructions based on a simulated phantom and a human brain slice. We used a single-shot rosette trajectory to synthesize k-space data, and the trajectory is specified in the previous section. The model of the k-space data is given by (2.2). The noise was white Gaussian and the signal-to-noise ratio (SNR) in this work is defined as

$$\text{SNR} = \frac{\|s\|_2}{\|s - s_0\|_2}$$

where s is the noisy data and s_0 is the noiseless data. Normalized mean square error (NMSE) was used as a metric for the accuracy of the reconstructions. The NMSE is defined as:

$$\text{NMSE} = \frac{\|f - f_0\|_2}{\|f_0\|_2}$$

where f and f_0 represent the estimate and the ground truth. All methods started at a trivial initialization: $m_n = 0.5$ and $z_n = 0$ for all n. For all experiments, we used a masking technique to mask out voxels with low spin density before applying any iterative algorithm. This technique has also commonly been used by some other authors [83, 97]. All algorithms were implemented using MATLAB and run with an Intel Core i7-4700MQ CPU.

2.4.1 Simulated phantom

We simulated a piecewise continuous cylinder phantom to mimic four small cylinders each filled with different materials and placed within a large cylinder container. Since the off-resonance map is often modeled as a smooth map [98], we have smoothed the off-resonance map by a circular averaging filter with a radius of 5 voxels. We use a triple to represent the maps of the spin density, R_2^* decay, and the off-resonance frequency for each material. The units for the R_2^* decay and off-resonance frequency are sec^{-1} and Hz. The parameters of the material in the large container are (1, 20, 100). The parameters of the four small containers in the middle are (0.2, 2, -20), (0.4, 10, 60), (0.6, 50, 140), and (0.8, 80, 200) in a left-right, top-bottom order. Figure 2.3 (a-c) shows the three parameter maps for the phantom.

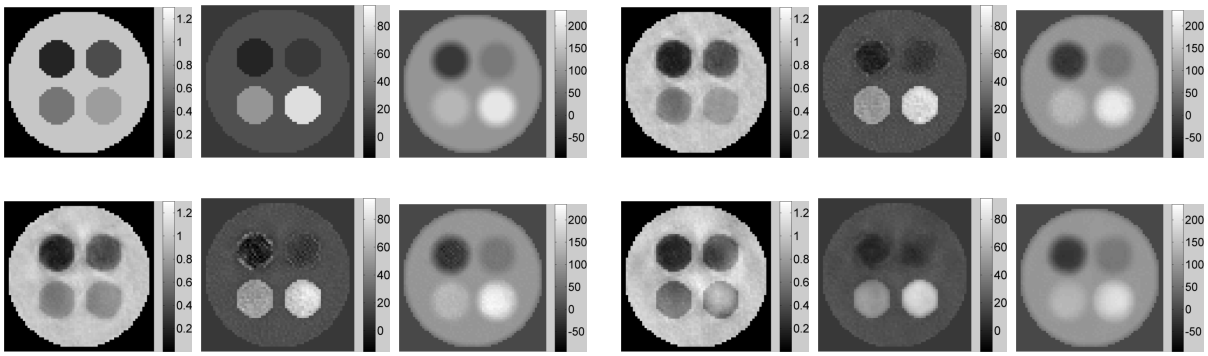


Figure 2.3: Simulation results: (a-c) ground truth for the spin density, R_2^* decay and the off-resonance frequency, (d-f) OTRC reconstruction results, (g-i) CVTRC reconstruction results, and (j-l) NCGC reconstruction results.

Figure 2.3 shows the estimation results obtained through OTRC, CVTRC, and NCGC with SNR = 100. All three algorithms obtained reasonable reconstructions. Single-shot

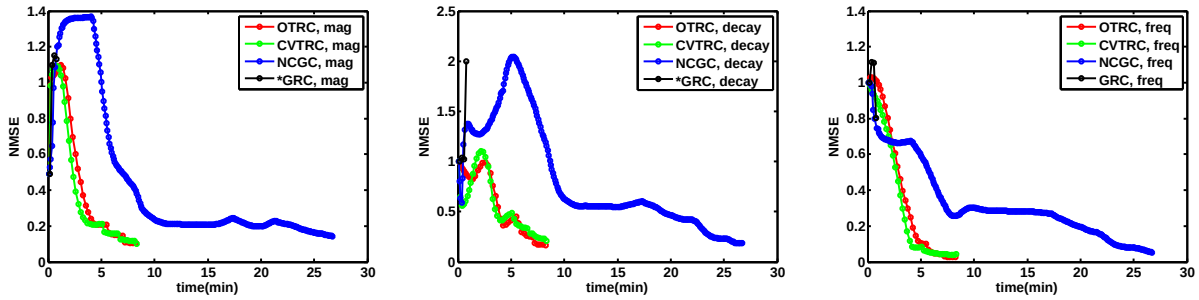


Figure 2.4: Convergence profiles of the spin density map (left), R_2^* decay map (middle) and the off-resonance frequency map (right). The red triangle line is the profile associated with the TRC algorithm, while the blue circle line is the profile associated with the CGC algorithm.

Table 1: Convergence accuracy and time of OTRC, CVTRC, NCGC, and GRC for different data sets

Data	**Criteria**	**OTRC**	**CVTRC**	**NCGC**	**GRC**
SNR = 100 (reference regularization)	Time(min)	**8.2**	8.3	26.7	--
	NMSE(SD, Decay, Freq.)	**(0.09, 0.14, 0.03)**	(0.09, 0.20, 0.04)	(0.14, 0.19, 0.05)	--
SNR = 20 (10x regularization)	Time(min)	**7.9**	8.2	27.2	--
	NMSE(SD, Decay, Freq.)	**(0.13, 0.26, 0.06)**	(0.14, 0.34, 0.08)	(0.21, 0.47, 0.23)	--
SNR = 10 (100x regularization)	Time(min)	7.8	9.7	25.7	**7.1**
	NMSE(SD, Decay, Freq.)	(0.18, 0.35, 0.10)	(0.20, 0.58, 0.10)	(0.26, 0.54, 0.33)	**(0.18, 0.34, 0.10)**
Double Resolution (SNR = 100, reference regularization)	Time(min)	10.9	**10.8**	34.2	--
	NMSE(SD, Decay, Freq.)	**(0.17, 0.29, 0.07)**	(0.16, 0.39, 0.10)	(0.22, 0.27, 0.07)	--

reconstruction commonly manifests edge artifacts. We conjecture the artifacts are caused by insufficient sampling of the k-space high-frequency band for a single-shot trajectory. These artifacts are present for every method and do not influence the comparison. Also, these artifacts can be reduced by increasing the overall sampling using a technique such as multi-shot acquisition.

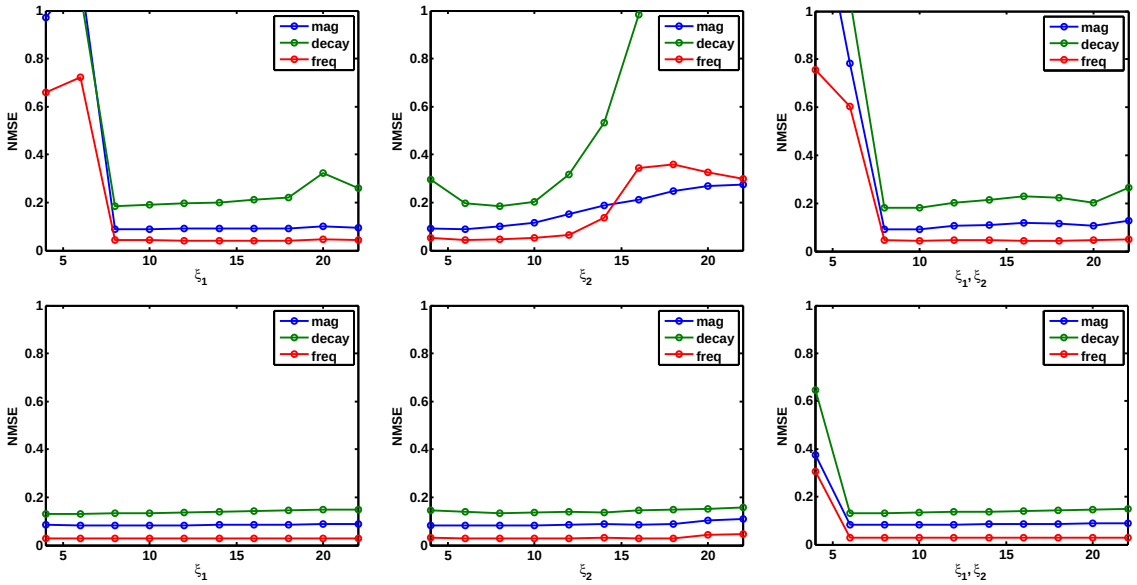


Figure 2.5: Robustness of CVTRC and OTRC to variation of $\xi_{1,2}$. The left, middle and right columns show NMSE of CVTRC (left) and OTRC (right) for varying ξ_1 only, varying ξ_2 only, and simultaneously varying ξ_1 and ξ_2, respectively.

Figure 2.4 shows the convergence profiles associated with OTRC, CVTRC, NCGC, and GRC for the same k-space data. OTRC has about the same profile with CVTRC. CVTRC appears to be slightly faster than OTRC and slightly less accurate than OTRC especially on the R_2^* decay reconstruction. However, both of them are much faster than NCGC. In addition, GRC does not converge. It stops because the residual and gradient become infinite in the next iterate. This numerical instability is caused by the fact that a line search is missing in GRC. Without a line search, the algorithm updates based on a local quadratic approximation, which may increase the actual cost function. Trust-region methods never have this issue because 1) minimization associated with the sub-problem is always applied within a trust region and 2) a check is carried out in each iteration to guarantee the residual reduction.

Robustness to different SNRs and discretization

Table 1 shows the accuracy and time required by different methods in processing data sets with different SNRs and different discretization resolutions. With different SNRs (100,

20, and 10), all methods except GRC are stable. However, OTRC has higher accuracy than both NCGC and CVTRC. Both OTRC and CVTRC are much faster than NCGC. GRC only converges for SNR = 10, and its accuracy is roughly the same as OTRC. Since GRC and OTRC have the same cost function, similar accuracy is not surprising. However, the stability of GRC is a major problem for the algorithm in the case of a poor initialization. Also, when SNR decreases, the regularization parameters for all methods increase in order to guarantee convergence. However, since all methods have the same regularization parameters for a given SNR, a direct comparison of performance can still be made.

As real k-space data arises physically from continuous parameter maps, a higher discretization resolution was used when synthesizing k-space data to test the robustness of the algorithms. As shown in Table 1, double resolution in the data synthesis leads to longer reconstruction time and moderately worse accuracy for all algorithms. OTRC and CVTRC still achieve reasonable accuracy and still outperform NCGC and GRC.

Robustness to ξ and λ parameter sets

Figure 2.5 shows the different sensitivity of OTRC and CVTRC to variation of ξ_1 only, of ξ_2 only, and of ξ_1 and ξ_2 simultaneously. When only ξ_1 or only ξ_2 changes, CVTRC displays a large variation in accuracy, while OTRC does not. When both ξ_1 and ξ_2 change by the same amount, CVTRC displays much better robustness than it does when a single parameter changes. This experiment demonstrates that the current implementation of CVTRC is susceptible to relative variation between ξ_1 and ξ_2. It also shows that OTRC is very robust for these parameters. Furthermore, since the two regularization parameters are reduced by ξ_1 and ξ_2 respectively in each continuation phase, the experiment implies that CVTRC relies more on a good match of the two regularization parameters than OTRC. Figure 2.6 shows the performance of the two algorithms when only λ_1 or only λ_2 varies at the final continuation phase. The two algorithms worked well for a wide range of variations of $\lambda_{1,2}$.

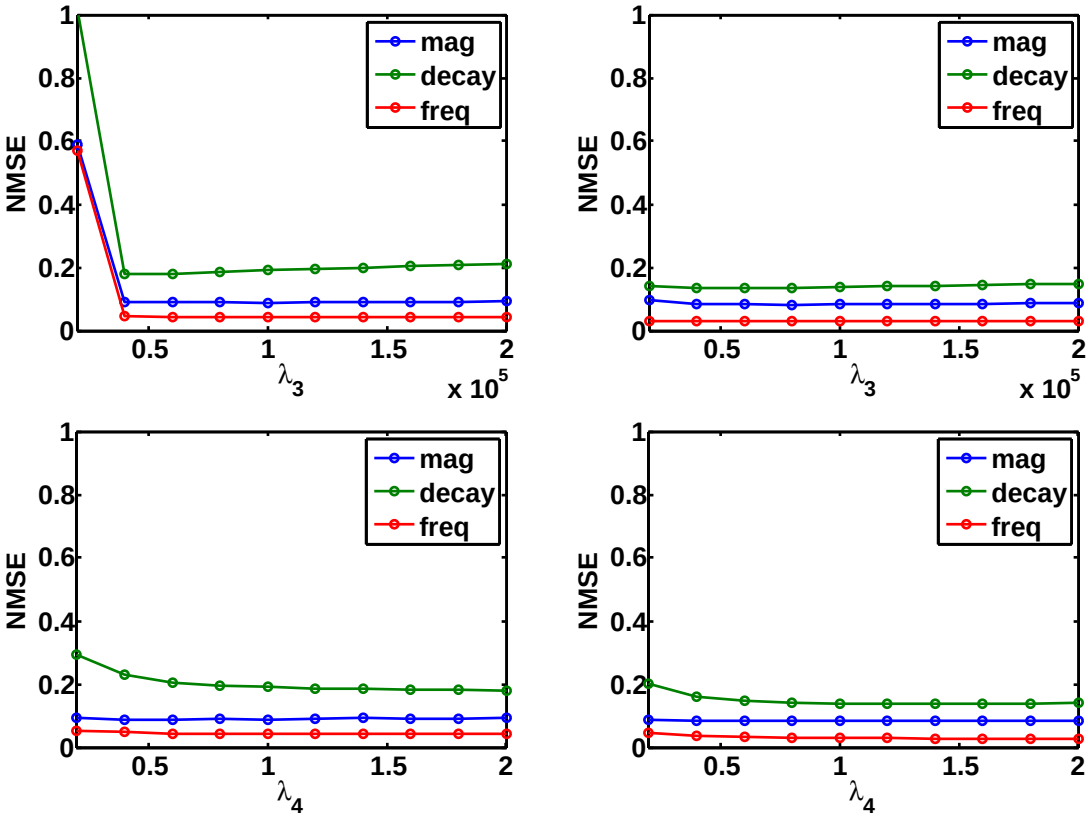


Figure 2.6: Robustness of CVTRC and OTRC to regularization parameter variation. The top and bottom rows show NMSE of CVTRC (left) and OTRC (right) for varying λ_3 only and varying λ_4 only, respectively.

Advantage of CVTRC

This experiment aims to show the speed advantage of CVTRC over OTRC when an educated $\vec{z}$ initialization exists. In this scenario, the fact that CVTRC has a larger approximation support in the spin density dimension should give it an advantage. We used the same simulated phantom and applied two initializations to the parameter maps. Both initializations are close to the ground truth of $\vec{z}$ and distant from that of $\vec{m}$. The first initialization adds a small amount of noise to the ground truth of the $\vec{z}$ map, and the second initialization randomly rotates the $\vec{z}$ map by a small angle in space. Since the two initializations are only statistically similar in initializing $\vec{z}$, this experiment can well verify that CVTRC converges faster than OTRC when the initialization of $\vec{z}$ is good. The results are shown in Figure 2.7.

Despite the two initialization differences, CVTRC has a more uniform and faster convergence profile than OTRC, demonstrating the advantage of CVTRC.

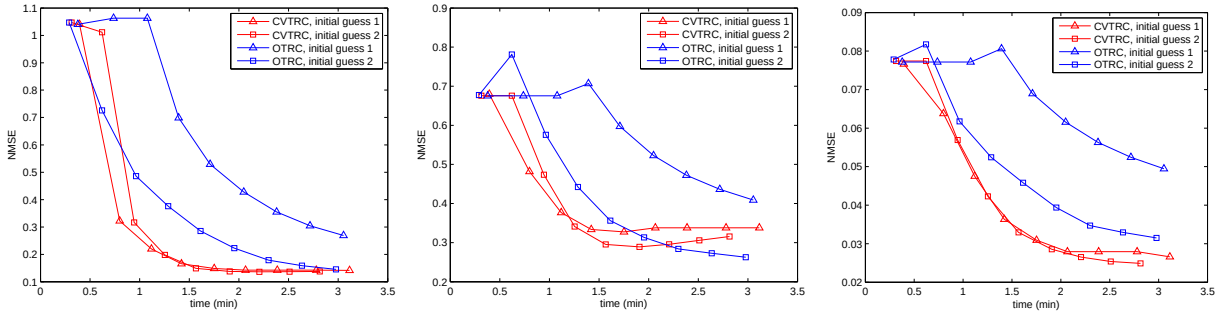


Figure 2.7: Verification of the faster convergence of CVTRC than OTRC when good initial guesses for R_2^* decay and off-resonance frequency are present.

2.4.2 Brain slice

Figure 2.8 shows the results of CVTRC, OTRC, NCGC, and GRC for a transversal slice of a human brain and synthetic k-space data. The slice crosses the orbitofrontal area of the brain cortex where a large off-resonance ranging from -414 Hz to 378 Hz is present. The three parameter maps were obtained through the aforementioned multi-echo approach. We used SNR = 100 in this simulation. All four methods converged by activating the stopping criterion, including GRC, which does not converge for the simulated phantom with the same SNR. In fact, as shown in Figure 2.9, GRC is even faster than OTRC by roughly 20%. The speed gain is due to the fact that GRC does not impose any trust region control and therefore the variables vary freely in every iteration. This can bring a faster convergence, as in this example, but it can also destroy the convergence, as shown in the last one. Thus, GRC is unreliable. In comparison, OTRC uses the same cost function but with a trust region control and maintains a good balance between stability and speed.

From Figure 2.8 and Figure 2.9, we can also see that the accuracy of reconstruction is generally worse than the simulated phantom. This is because the surrounding area of the brain slice has either a fast R_2^* decay or a low spin density. A low spin density makes

the estimation of the complex frequency $z(\vec{r})$ in these voxels challenging because the time signal is not sensitive to variations in $z(\vec{r})$ [32]. A large R_2^* decay makes estimation difficult for every parameter, since the time duration of useful data is limited. In this example, the intensity of many voxels in this area becomes insignificant after 10 ms, leading to a very small number of useful samples compared to an 80 ms trajectory. These factors together cause a high estimation bias in the surrounding area, which increases the MSE of the entire image. We calculated the NMSEs of algorithms for a rectangular area in the middle, and these NMSEs are consistent with findings in the previous experiment.

CVTRC generally has a larger estimation error than OTRC in R_2^* decay reconstruction in the presence of poor initializations. We conjecture that this is caused by the difference in regularization used in CVTRC. In order to accommodate the change of variables, we regularize $\vec{m} \odot \vec{z}$ in the CVTRC cost function rather than $\vec{z}$ alone. This causes a suboptimal regularization over $\vec{z}$, especially when regularization parameters are small or when spin density is low.

2.5 Real k-space data

As we state in section 2.3, we employed reconstructions from acquired multi-echo data to obtain a gold standard. A multi-echo rosette instead of a single-shot rosette was used to acquire more data and reduce the error caused by noise, model mismatch, and other possible unidentified factors. Figure 2.10 shows the reconstruction results of OTRC, CVTRC, and NCGC. GRC is not shown because it does not converge. The other three methods lead to a reasonable reconstruction, although considerable noise and boundary artifacts are present. Figure 2.11 shows the convergence profiles of the four methods. GRC failed to converge at the end, demonstrating its unreliability. OTRC and CVTRC converged much faster than NCGC, demonstrating their speed advantage. More severe boundary artifacts were present in the OTRC and CVTRC results than the GRC results. In fact, all our experiments indicate

that NCGC leads to a smoother result than other methods. This may be due to the well-known regularization property of the conjugate gradient method when it stops early [43]. However, the long time required to converge dramatically reduces this advantage of NCGC in the presence of a poor initialization.

2.6 Conclusions and Discussions

We have presented a new approach to joint reconstruction of the spin density, R_2^* decay and off-resonance frequency maps through single- and multi-shot trajectories. This approach adopts the trust region framework commonly used to solve nonlinear equation systems. Due to different local approximation strategies, two trust-region methods OTRC and CVTRC were proposed. Experiments show that the two methods outperform an NCG method in speed and the GR method in stability. The four methods were applied to synthetic k-space data based on simulated phantoms and an orbitofrontol brain slice as well as real phantom data. Between the two trust-region methods, OTRC usually has smaller boundary artifacts, but CVTRC can be faster in the presence of a good complex frequency initialization. To our knowledge, this is the first research focusing on algorithm stability and convergence speed in joint estimation of the three parameter maps.

The origins of the three methods NCG, TR, and GR can be traced to some of the classical methods for nonlinear optimization. NCG itself is a classical method for nonlinear optimization [41]. NCG is a variant of conjugate gradients and is therefore faster than steepest descent but slower than Newton-like methods. However, for a large system, even conjugate gradients can lead to an intolerable computation time, especially when a good preconditioning is absent (cf. [37] P.150). GR can be categorized as a truncated Gauss-Newton (TGN) method [77]. The method is rooted in the Gauss-Newton method, yet in each iteration it searches for the Gauss-Newton direction by iteratively solving an approximated linear system with methods such as conjugate gradients. This strategy makes TGN suitable for large-scale systems. Since TGN is a Newton-like method, it can possibly approach the

quadratic convergence rate that the Newton method possesses. In fact, TGN has been reported to be faster than NCG with a suitable preconditioner in the inner problem [77]. However, an important factor for a general unconstrained optimization problem is the line search in each iteration, which is lacking in the current GR method and its source method in [83]. The lack of line search causes the algorithm to be unstable since the residual at the next iterate may be far away from its predicted value. The trust-region method solves this issue by confining the variable space for calculating the Gauss-Newton direction to be within a local domain called a trust region. The algorithm examines the residual at the next iterate for every iteration to make sure that the residual is actually reduced. In addition, the trust region size can vary based on the behavior of the previous iteration. This flexibility greatly preserves the speed advantage of the Newton method. Overall, these strategies allow TR to be faster than NCG and more stable than the GR method. More discussion of the three methods can be found in [37].

Experiments show that the full advantage of CVTRC in utilizing a change-of-variable linearization is reduced by the restriction on variations in $\vec{m}$ and the indirect regularization of $\vec{z}$. In particular, the restriction of the variation in $\vec{m}$ in the trust region setup limits the convergence speed, and the regularization over $\vec{m} \odot \vec{z}$ instead of $\vec{z}$ allows room for boundary artifacts. It would be advantageous if these restrictions concerning the trust region setup and the regularization can be removed or reduced. However, this would be not trivial to accomplish, since simply removing these restrictions will cause the method to be unstable. With the current approach, CVTRC works better than OTRC only when initialization of $\vec{z}$ is good.

Although the two methods can handle a wide range of objects for joint reconstruction, there are still some limitations in their capabilities. First, there seems to be a "workable" range for each parameter map within which the convergence can be guaranteed [103]. Currently, this range is estimated empirically. For example, the largest R_2^* decay rate should be comparable to the readout time, and the spin density dynamic range should be limited.

Otherwise, artifacts would increase in regions of low spin density. A deeper understanding of the off-resonance frequency range is still needed. However, we emphasize that these ranges are present for all reconstruction methods and are dependent on the number of k-space samples. Second, parameter tuning including the regularization parameters and the continuation related parameters are demanding for the two methods. For most objects, these parameters do not need to be changed much. However, we have observed that a small variation of the parameters may help reduce the boundary artifacts. In addition, the continuation scheme must be changed when the initialization quality changes to exploit the acceleration advantage of a good initialization. This is challenging, since various initializations are possible in a given application. The auxiliary variable method introduced in the next chapter can very well solve this problem by incorporating a good initialization step [48]. The auxiliary variable method is therefore very promising as a way to replace the continuation method used in the trust-region method. Third, although the two trust-region methods are faster than the commonly used NCG method, they are still time-consuming under the metric for clinically applicable methods. A C++ implementation instead of the current MATLAB implementation might achieve a speed gain of 10-50. Some fast computational methods have also been proposed, such as using nonuniform FFTs and time segmentation [97] or using Toeplitz-based fast matrix multiplication [29]. These methods can be easily applied to the two trust-region methods, where the acceleration could be several fold. Overall, the two proposed trust-region methods take some vital steps toward improving clinical MRI methods for fast joint estimation of the spin density, R_2^* decay and off-resonance frequency maps.

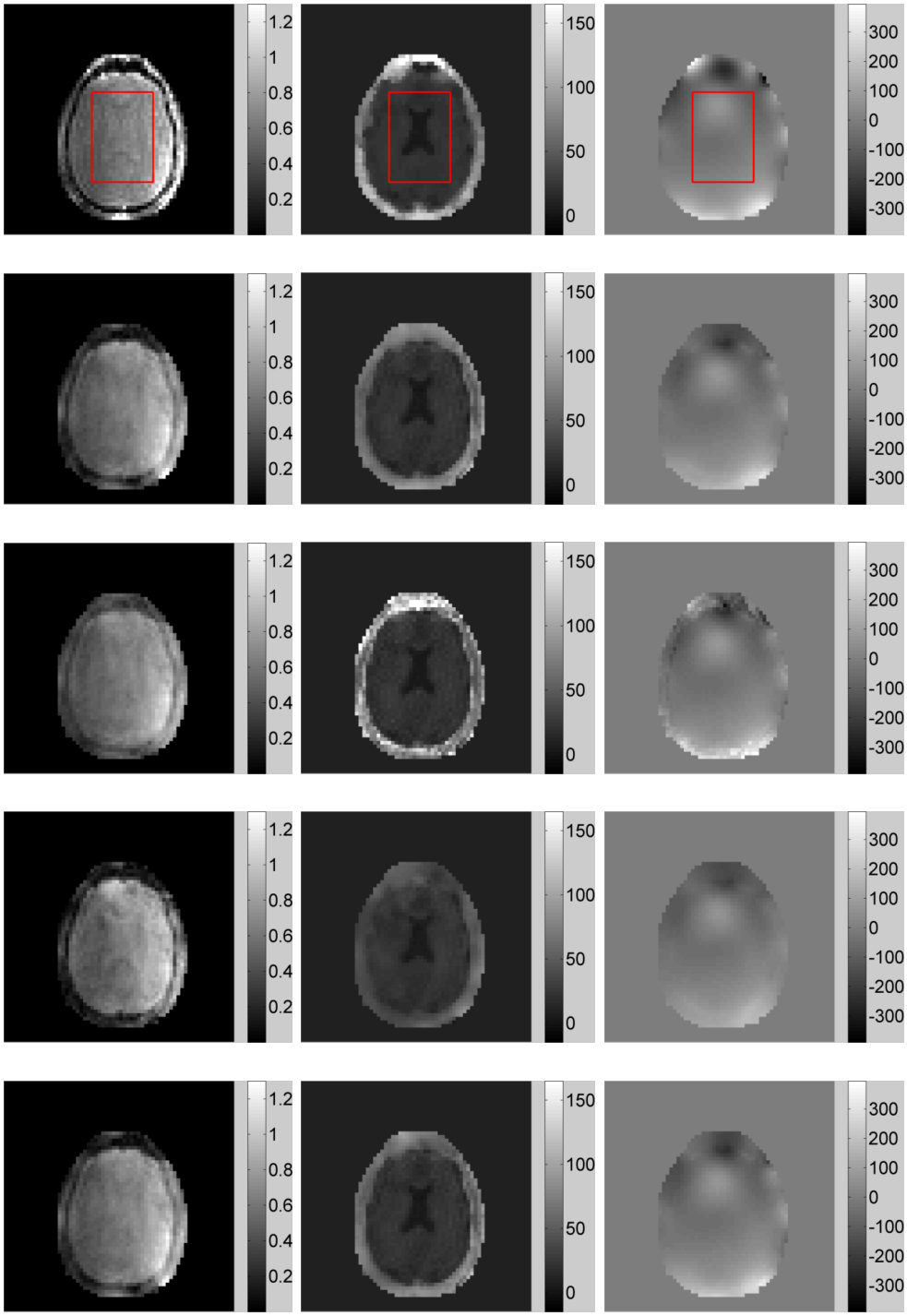


Figure 2.8: A human brain transversal slice with synthetic k-space data. White noise with SNR = 100 added to the data. Row 1 shows the ground truth of the spin density, R_2^* decay and the off-resonance frequency for the synthetic data. Row 2 shows OTRC reconstruction results. Row 3 shows CVTRC reconstruction results. Row 4 shows NCGC reconstruction results. Row 5 shows GRC reconstruction results. NMSE of the rectangle area for the four algorithms: (5.0%, 5.2%, 2.7%) for OTRC, (4.3%, 6.1%, 3.6%) for CVTRC, (13.7%, 7.9%, 13.5%) for NCGC, and (4.2%, 4.4%, 2.4%) for GRC.

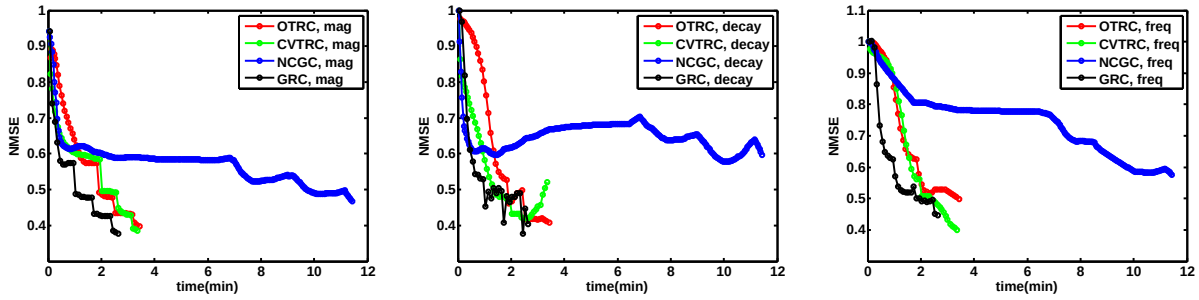


Figure 2.9: Convergence profiles of the spin density (left), R_2^* decay (middle) and the off-resonance frequency (right). Red, green, blue, and black lines represent OTRC, CVTRC, NCGC, and GRC, respectively.

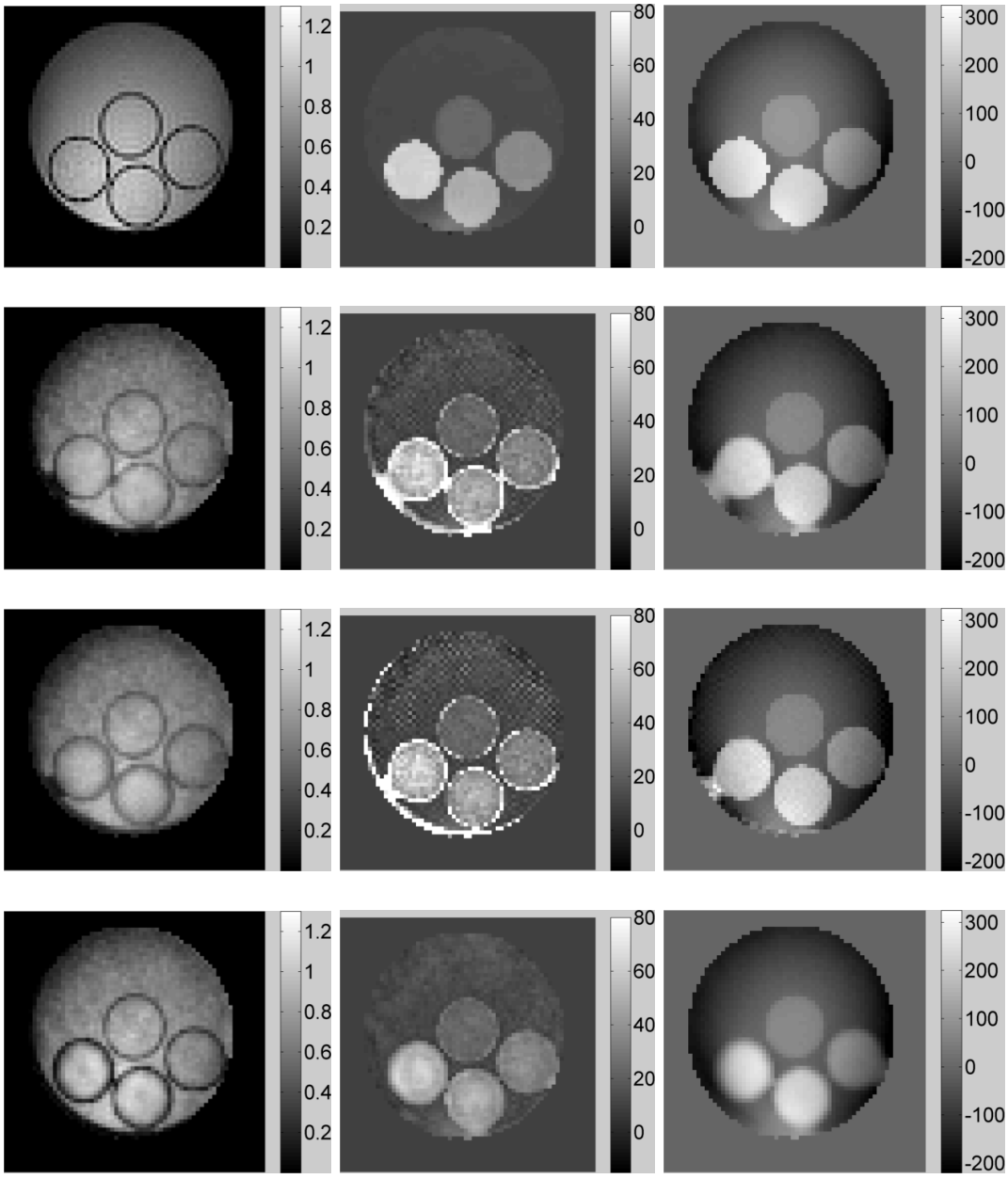


Figure 2.10: Reconstruction of a real four-cylinder phantom with acquired k-t space data. Row 1 shows the gold standard for spin density, R_2^* decay and off-resonance frequency. Row 2 shows OTRC reconstruction results. Row 3 shows CVTRC reconstruction results. Row 4 shows NCGC reconstruction results.

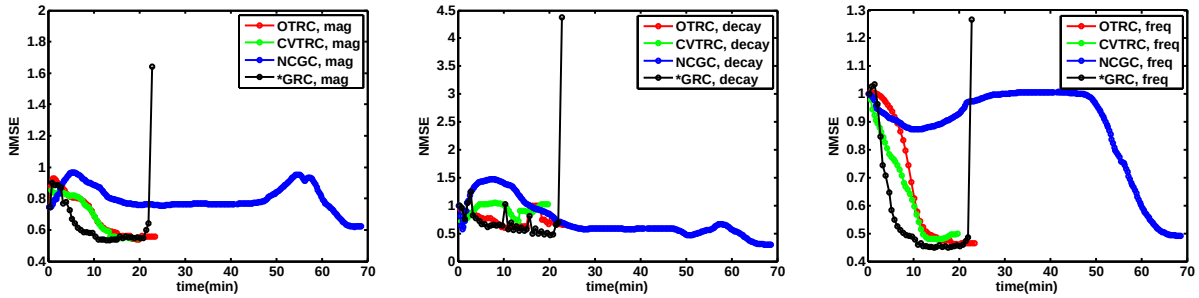


Figure 2.11: Convergence profiles of spin density (left), R_2^* decay (middle) and off-resonance frequency (right). Red, green, blue, and black lines represent OTRC, CVTRC, NCGC, and GRC, respectively.

Chapter 3

An efficient auxiliary variable method

So far, almost all previous reconstruction algorithms for jointly reconstructing spin density, R_2^* decay, and off-resonance frequency maps are gradient-based iterative algorithms and the computational cost is high. In this chapter, we propose a novel auxiliary variable (AV) method for the joint reconstruction problem. We reformulate the problem to a constrained optimization problem by employing an auxiliary variable and then transform it to an unconstrained problem by the penalty method. The cost function is then minimized by the well-known alternating direction method, which reduces the computational cost by leveraging special structure of the cost function. We show empirically that the algorithm is very fast during the early stage of the iterations but much slower in the later stage. This may be due to the inevitable ill-conditioning associated with the penalty method [24, 37]. Another reason may be the nonconvexity of the new cost function due to the exponential function constraint. Regularization over both the auxiliary variable and the ordinary variables can effectively increase the speed but cannot completely solve the problem. To address this issue, we propose a two-step hybrid method. In the first step, we use the alternating minimization algorithm that alternates the minimization of the cost function with respect to different variables. In the second step, we employ the ordinary trust region algorithm we previously proposed to refine the estimate from the first step. The strategy of variable splitting has been used in multi-parameter reconstruction based on a multi-echo sequence [22, 58] and parallel imaging [88]. To our knowledge this is the first paper applying variable splitting to this reconstruction problem based on single- or multi-shot sampling of k-space. The advantage of the new algorithm mainly lies in the convergence speed and reduced computation. This advantage is demonstrated through both simulated and *in vivo* data .

The term "auxiliary variable" has been commonly used in the half-quadratic method. In that method, an auxiliary variable is introduced to facilitate the replacement of a non-quadratic function with a half-quadratic augmented function so that a splitting method can be used to efficiently minimize the equivalent cost function [35, 36]. Although a similar procedure is used for our problem, the auxiliary variable has a somewhat different application. A composite function that is quadratic on the outside is separated into a sum of a quadratic function and a half-quadratic function by introducing the auxiliary variable. That is, the original problem is to minimize

$$\|\vec{y} - (\gamma \circ \theta)(\vec{x})\|^2 \tag{3.1}$$

where the data $\vec{y}$ and the model $\vec{x}$ are two vectors satisfying $\dim(\vec{y}) > \dim(\vec{x})$, and $\gamma(\cdot)$ and $\theta(\cdot)$ are two functions whose composition forms the nonlinear observation operator. Often, the range space of the function $\theta(\cdot)$ has a much larger dimension than both $\vec{y}$ and $\vec{x}$. Instead of directly minimizing (3.1), one introduces an auxiliary variable $\vec{x}' \in \mathrm{dom}(\gamma)$ and solves a constrained optimization problem

$$\|\vec{y} - \gamma(\vec{x}')\|^2 \text{ s.t. } \vec{x}' = \theta(\vec{x}) \text{ for some } \vec{x} \tag{3.2}$$

As $\vec{x}'$ is the key to the new formulation, we call the method an auxiliary variable method. The theory concerning convergence and implementation of the auxiliary variable algorithm for a general $\gamma(\cdot)$ and $\theta(\cdot)$ is still somewhat immature, although algorithms for linear $\gamma(\cdot)$ and $\theta(\cdot)$ have been investigated [1, 88].Other algorithms, such as the trust region, have also been developed for the composite optimization problem [61], but they are usually computationally intense due to the gradient and residual evaluations.

The chapter is organized as follows. In Section I, we first define the reconstruction problem and introduce alternative algorithms. We then introduce the auxiliary variable algorithm, including its motivation, implementation, and limitations. Finally, we propose the two-step approach and summarize the reconstruction method. In Section II, we compare

the proposed method with alternative methods based on data from a simulated phantom, a real phantom, and brain *in vivo* data. In Section III, we discuss the context and limitations of the current implementation of the auxiliary variable method as well as the great potential of the method for this problem.

3.0.1 Formulation of the auxiliary variable algorithm

Basic framework

Let $\vec{s}_l(\cdot,\cdot) \triangleq [s_{l1}(\cdot,\cdot), s_{l2}(\cdot,\cdot), \ldots, s_{lP}(\cdot,\cdot)]^T$ for $l = 1, 2, \ldots, L$. Define $\rho_l(\vec{m},\vec{z}) \triangleq [m_1 e^{-z_1 t_l}, m_2 e^{-z_2 t_l}, \ldots, m_N e^{-z_N t_l}]^T$. Equation (2.2) can then be rewritten in a matrix form:

$$\begin{bmatrix} \vec{s}_1(\vec{m},\vec{z}) \\ \vec{s}_2(\vec{m},\vec{z}) \\ \vdots \\ \vec{s}_L(\vec{m},\vec{z}) \end{bmatrix} = \begin{bmatrix} D_1 & 0 & \cdots & 0 \\ 0 & D_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & D_L \end{bmatrix} \begin{bmatrix} F_0 & 0 & \cdots & 0 \\ 0 & F_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & F_0 \end{bmatrix} \begin{bmatrix} \rho_1(\vec{m},\vec{z}) \\ \rho_2(\vec{m},\vec{z}) \\ \vdots \\ \rho_L(\vec{m},\vec{z}) \end{bmatrix} \tag{3.3}$$

where F_0 represents a 2-D discrete Fourier transform, and D_l for $l = 1, 2, \ldots, L$ is a $P \times N^2$ matrix whose sparsity greatly depends on whether the trajectory is "on-grid" or "off-grid". "On-grid" means that all trajectory samples are located on the Cartesian grid, while "off-grid" means samples can be arbitrarily located. In the situation of "on-grid" sampling, each row of D_l is a standard basis vector of length N^2 which is zero everywhere except one element. This element has value 1 and its location represents the sampled grid point. In the situation of "off-grid" sampling, D_l is a full matrix and each row is a shifted ideal interpolator for a nonuniform FFT, which is actually a sampled periodic sinc function or Dirichlet function [30]. Let $\mathbf{D}$, $\mathbf{F}$, $\vec{\rho}(\vec{m},\vec{z})$ denote the left matrix, the middle matrix, and the right vector on the right side of the equation. We can then write (3.3) as

$$\vec{s}(\vec{m},\vec{z}) = \mathbf{D}\mathbf{F}\vec{\rho}(\vec{m},\vec{z}) \tag{3.4}$$

The objective is to estimate $\vec{m}$ and $\vec{z}$ given the acquired data $\vec{y} \triangleq [y_{11}, y_{12}, \ldots, y_{1P}, y_{21}, \ldots, y_{2P}, \ldots, y_{LP}]^T$. Since noise is usually white Gaussian, the estimation can be done by minimizing a least squares criterion:

$$\|\vec{y} - \mathbf{DF}\vec{\rho}(\vec{m}, \vec{z})\|^2 \tag{3.5}$$

with respect to $\vec{m}$ and $\vec{z}$. Notice that the relation between $\vec{\rho}$ and $\vec{y}$ is linear and temporally decoupled, while that between $[\vec{m}, \vec{z}]^T$ and $\vec{\rho}$ is spatially decoupled. This observation inspires us to introduce an auxiliary variable into the modeling so that (3.5) can be reformulated as

$$\|\vec{y} - \mathbf{DF}\vec{u}\|^2 \quad \text{s.t. } \vec{u} = \vec{\rho}(\vec{m}, \vec{z}) \text{ for some } \vec{m} \text{ and } \vec{z}. \tag{3.6}$$

Introducing the auxiliary variable separates the linear mapping **DF** from the nonlinear mapping $\vec{\rho}(\cdot, \cdot)$ and decouples the spatial Fourier transform from the temporal exponential function. In this way, computation associated with each variable can be reduced. Similar strategies have been applied to many image processing and computer vision applications such as image reconstruction or restoration [35, 36, 54] and shape modeling [12, 108], but most of these methods decouple the data term from the regularization. Splitting the data term was investigated in [88], but the observation operator in [88] is linear. We use the penalty method to transform the constrained problem to an unconstrained problem. The resulting cost function is

$$\|\vec{y} - \mathbf{DF}\vec{u}\|^2 + \alpha\|\vec{u} - \vec{\rho}(\vec{m}, \vec{z})\|^2 \tag{3.7}$$

where we estimate $\vec{u}$ as well as $\vec{m}$ and $\vec{z}$. The parameter $\alpha \in \mathbb{R}^+$ is the penalty coefficient. We minimize the cost function in an alternating direction fashion by minimizing w.r.t. $\vec{u}$ and $\vec{m}, \vec{z}$ alternately. This is the basic framework of this algorithm.

Regularization terms

Directly applying the alternating minimization to (3.7) would result in very slow convergence and low accuracy although it would be computationally very efficient. This is because solving (3.6) is equivalent to solving the original problem (2.2) which has been shown to be severely ill-conditioned with a single-shot trajectory [47, 49]. Notice that (3.7) may have even worse conditioning than (3.6) since the variable domain is augmented. Further analysis is given in later sections.

A remedy to the ill-conditioning is to add appropriate regularization terms. We have shown in previous work [50] that Tikhonov regularization of $\vec{m}$ and $\vec{z}$ maps works quite well with the ordinary cost function (2.1). However, our experience shows that regularizing these two variables leads to a limited improvement in convergence speed for the modified cost function (3.7). However, if we simultaneously regularize $\vec{u}$ in both the spatial and temporal dimension, the convergence is greatly accelerated. The importance of the extra regularization is analyzed in Section 3.0.3. Here, we first define this regularized cost function for the auxiliary variable algorithm:

$$\begin{aligned}\Phi(\vec{u},\vec{m},\vec{z}) \triangleq \|\vec{y}-\mathbf{D}\mathbf{F}\vec{u}\|^2+\alpha\|\vec{u}-\vec{\rho}(\vec{m},\vec{z})\|^2\\ +\beta_1\|\mathbf{R}_{L-1}\mathbf{E}\vec{u}\|^2+\beta_2\|\mathbf{E}\vec{u}\|^2+\lambda_1\|R\vec{m}\|^2+\lambda_2\|R\vec{z}\|^2\end{aligned} \tag{3.8}$$

where R represents a first-order finite difference operator in the spatial domain, and $\mathbf{R}_{L-1} \triangleq \mathrm{diag}(R, R, \ldots, R)$ is an $L-1$ block-diagonal matrix. $\mathbf{E}$ is the first-order finite difference operator in the temporal domain, which is formed by a Kronecker product of a finite difference operator in time and an $N^2 \times N^2$ identity matrix. $\mathbf{R}_{L-1}\mathbf{E}$ seems to result in slightly better performance than $\mathbf{R}_L$ in our experience. This is possibly because $\mathbf{R}_{L-1}\mathbf{E}$ also leads to a regularization in the temporal domain. Notice that $\mathbf{R}_{L-1}\mathbf{E}$ and $\mathbf{R}_L$ have a similar computational complexity, as we show later, making the former regularization more attractive.

3.0.2 The alternating direction method

We minimize the cost function (3.8) using the alternating direction method. In each iteration, the algorithm first minimizes (3.8) w.r.t. $\vec{u}$ given $\vec{m}$ and $\vec{z}$ estimates from the previous iteration:

$$\vec{u} \longleftarrow \min_{\vec{u}} \Phi(\vec{u}, \vec{m}, \vec{z}). \tag{3.9}$$

The algorithm then minimizes w.r.t. $\vec{m}$ and $\vec{z}$ given the $\vec{u}$ estimate from the previous step. Since the first term in (3.8) does not involve $\vec{m}$ and $\vec{z}$, the minimization is done by

$$\vec{m}, \vec{z} \longleftarrow \min_{\vec{m},\vec{z}} \|\vec{u} - \vec{\rho}(\vec{m}, \vec{z})\|^2 + \lambda_1 \|R\vec{m}\|^2 + \lambda_2 \|R\vec{z}\|^2 \tag{3.10}$$

The two steps are elaborated in the following.

Step 1: minimization w.r.t. $\vec{u}$

Notice that the function $\Phi(\cdot, \vec{m}, \vec{z})$ is quadratic when $\vec{m}$ and $\vec{z}$ are fixed. Therefore, an explicit solution actually exists:

$$\begin{aligned} \hat{\vec{u}} &= \Psi^{-1}(\mathbf{F}^H \mathbf{D}^H \vec{s} + \alpha \vec{\rho}(\vec{m}, \vec{z})) \\ \Psi &\triangleq \mathbf{F}^H \mathbf{D}^H \mathbf{D} \mathbf{F} + \beta_1 \mathbf{E}^H \mathbf{R}_{L-1}^H \mathbf{R}_{L-1} \mathbf{E} + \beta_2 \mathbf{E}^H \mathbf{E} + \alpha \mathbf{I}_L \end{aligned} \tag{3.11}$$

where Ψ is the normal matrix associated with (3.8) and $\mathbf{I}_L$ is an $LN^2 \times LN^2$ identity matrix. Notice that $\mathbf{R}_{L-1}\mathbf{E}$ and $\mathbf{E}$ are both block-tridiagonal and each block is R and I (the $N^2 \times N^2$ identity matrix), respectively. Therefore, we can then rewrite matrix Ψ as

$$\begin{aligned} \Psi &= \mathbf{F}^H \mathbf{D}^H \mathbf{D} \mathbf{F} + \beta_1 \mathbf{F}^H \mathbf{X}^H \mathbf{X} \mathbf{F} + \beta_2 \mathbf{F}^H \mathbf{E}^H \mathbf{E} \mathbf{F} + \alpha \mathbf{F}^H \mathbf{F} \\ &= \mathbf{F}^H (\mathbf{D}^H \mathbf{D} + \beta_1 \mathbf{X}^H \mathbf{X} + \beta_2 \mathbf{E}^H \mathbf{E} + \alpha \mathbf{I}_L)^{-1} \mathbf{F} \end{aligned} \tag{3.12}$$

where $\mathbf{X}$ is a block-diagonal matrix such that $\mathbf{R}_{L-1}\mathbf{E} = \mathbf{F}^H\mathbf{X}\mathbf{F}$. The estimate of $\vec{u}$ is then

$$\begin{aligned}\hat{\vec{u}} = \mathbf{F}^H(\mathbf{D}^H\mathbf{D} + \beta_1\mathbf{X}^H\mathbf{X} + \beta_2\mathbf{E}^H\mathbf{E} + \alpha\mathbf{I}_L)^{-1} \\ (\mathbf{D}^H\vec{s} + \alpha\mathbf{F}\vec{\rho}(\vec{m}, \vec{z}))\end{aligned} \tag{3.13}$$

An efficient implementation of (3.13) is very important to the efficiency of the auxiliary variable algorithm since the dimension of $\vec{u}$ is very large. Consider a non-Cartesian trajectory, which is used for most single-shot trajectories. In this case, $\mathbf{D}$ is a full matrix, and we cannot invert the matrix in close-form. Direct inversion based on LU-decomposition is possible and can be precomputed before the iteration. However, this approach may require a long computation time and a great deal of memory to store the decomposed matrices. In our approach, we choose to use preconditioned conjugate gradients (PCG) to iteratively estimate the inverse. The preconditioner is given by the inverse of $\mathbf{D}^H\mathbf{D} + \beta_1\mathbf{X}^H\mathbf{X} + \beta_2\text{diag}(\mathbf{E}^H\mathbf{E}) + \alpha\mathbf{I}_L$. Notice that all three matrices in the expression are block-diagonal. Therefore, the inversion can be done by separately inverting each block. Since X is diagonal, inversion of $D_l^H D_l + \beta_1 X^H X + \text{diag}(E^H E) + \alpha I$ for each l can be accomplished by applying the well-known Woodbury formula [115]. Notice that the inversion result can be precomputed to avoid inversion on the fly in each iteration. Since the preconditioner differs from direct inversion of the coefficient matrix only by the side band of $\beta_2\mathbf{E}^H\mathbf{E}$, the preconditioner works fairly well when β_2 is relatively small. When β_2 is large compared to the other terms, the performance of the preconditioner degrades. We set the maximum iterations to be 30 in most of our simulations to avoid excessive iterations. If a Cartesian trajectory is used, $\mathbf{D}$ is diagonal. Thus, the matrix to invert in (3.13) is block-tridiagonal, and each nontrivial block is diagonal. The well-known tridiagonal matrix inversion algorithm (Thomas algorithm) [16] can be used to invert the matrix, and the computational cost is on the same scale as the variable size. Surprisingly, even though EPI has more efficient computations, we observed that EPI—the only trajectory that is Cartesian—has a much slower convergence than a rosette trajectory has. A multi-shot EPI may help reduce the problem as reported by some authors [93];

however, accommodation for the general situation which includes any trajectories should be developed.

Step 2: minimization w.r.t. $\vec{m}$ and $\vec{z}$

The second step, as shown in (3.10), is very similar to the curve fitting approach used in multi-echo sampling [4,5,26,73,111,114]. The only difference is the regularization applied to $\vec{m}$ and $\vec{z}$. Without the regularization, the curve fitting is decoupled in the spatial domain and hence can be accomplished very rapidly. However, we observed that strong outliers exist in the fitting result if there is no regularization. Similar observations have been made by other authors when k-space is undersampled [94]. Regularization causes the reconstruction to be more complicated, and a rapid solution is therefore desired. In this work, we propose a new block-coordinate descent method to address this sub-problem, and the performance turns out to be satisfactory.

The block-coordinate descent method is used to decouple the variable space. We split the variable set $\vec{m}$ and $\vec{z}$ to $\vec{m}_1$, $\vec{m}_2$ and $\vec{z}_1$, $\vec{z}_2$ in the spatial domain according to a checkerboard pattern so that the black boxes correspond to positions of $\vec{m}_1$ and $\vec{z}_1$ and the white boxes correspond to positions of $\vec{m}_2$ and $\vec{z}_2$. Define two $N^2/2$-by-N^2 rectangular matrices P_1 and P_2 to represent the two downsampling matrices for odd and even coordinates, respectively. We then have $P_1\vec{m} = \vec{m}_1$ and $P_2\vec{m} = \vec{m}_2$. The cost function can be reformulated as

$$\begin{aligned} \|\vec{u} - \vec{\rho}(\vec{m}, \vec{z})\|^2 + \lambda_1 \|R_0 P_1^H (P_1\vec{m}) + R_0 P_2^H (P_2\vec{m})\|^2 \\ +\lambda_2 \|R_0 P_1^H (P_1\vec{z}) + R_0 P_2^H (P_2\vec{z})\|^2 \end{aligned} \tag{3.14}$$

since $P_1^H P_1 + P_2^H P_2 = I$. We then solve (3.14) through alternately minimizing w.r.t. $P_1\vec{m}$, $P_1\vec{z}$ and $P_2\vec{m}$, $P_2\vec{z}$. Let $\vec{m}_{1,2} \triangleq P_{1,2}\vec{m}$ and $\vec{z}_{1,2} \triangleq P_{1,2}\vec{z}$. Let $\vec{u}_1$ and $\vec{u}_2$ represent the spatial-domain partitions of $\vec{u}$ based on odd or even spatial coordinates. Define $R_{01} \triangleq R_0 P_1^H$ and $R_{02} \triangleq R_0 P_2^H$. Clearly, R_{01} and R_{02} are two column-downsampled matrices. Initialize vectors

$b_{m2} = R_0 P_2^H (P_2 \vec{m})$ and $b_{z2} = R_0 P_2^H (P_2 \vec{z})$. The splitting-based method executes steps (3.15-3.18) in each iteration

$$\begin{aligned} \vec{m}_1, \vec{z}_1 \longleftarrow & \min_{\vec{m}_1, \vec{z}_1} \|\vec{u}_1 - \vec{\rho}(\vec{m}_1, \vec{z}_1)\|^2 \\ & + \lambda_1 \|R_{01}\vec{m}_1 + b_{m2}\|^2 + \lambda_2 \|R_{01}\vec{z}_1 + b_{z2}\|^2 \end{aligned} \tag{3.15}$$

$$b_{m1} \longleftarrow R_0 P_1^H \vec{m}_1;\ b_{z1} \longleftarrow R_0 P_1^H \vec{z}_1 \tag{3.16}$$

$$\begin{aligned} \vec{m}_2, \vec{z}_2 \longleftarrow & \min_{\vec{m}_2, \vec{z}_2} \|\vec{u}_2 - \vec{\rho}(\vec{m}_2, \vec{z}_2)\|^2 \\ & + \lambda_1 \|R_{02}\vec{m}_2 + b_{m1}\|^2 + \lambda_2 \|R_{02}\vec{z}_2 + b_{z1}\|^2 \end{aligned} \tag{3.17}$$

$$b_{m2} \longleftarrow R_0 P_2^H \vec{m}_2;\ b_{z2} \longleftarrow R_0 P_2^H \vec{z}_2 \tag{3.18}$$

Both (3.15) and (3.17) are spatially decoupled, and therefore their Jacobian matrices are block diagonal. Since this structure makes the inverse of the Gauss-Newton matrix computationally efficient. we use the Gauss-Newton method. This method may have quadratic convergence rate especially when a good initialization is provided.

3.0.3 Performance analysis

The performance of the proposed auxiliary variable algorithm can be analyzed in two levels. For simplicity, we assume a Cartesian trajectory for performance analysis, since any trajectory can be rounded to a Cartesian trajectory with only a slight deviation. Without regularization of $\vec{u}$, the first step of the auxiliary variable algorithm leads to an estimate of

$\vec{u}$ expressed by

$$\begin{aligned}
\hat{\vec{u}} &= \mathbf{F}^H(\mathbf{D}^H\mathbf{D} + \alpha\mathbf{I})^{-1}\mathbf{F}(\mathbf{F}^H\mathbf{D}^H\vec{y} + \alpha\vec{\rho}_0) \\
&= \mathbf{F}^H(\frac{1}{\alpha}\mathbf{I} - \frac{1}{(\alpha+1)\alpha}\mathbf{D}^H\mathbf{D})(\mathbf{D}^H\vec{y} + \alpha\mathbf{F}\vec{\rho}_0) \\
&= \mathbf{F}^H(\frac{1}{\alpha+1}\mathbf{D}^H\vec{y} + \mathbf{F}\vec{\rho}_0 - \frac{1}{\alpha+1}\mathbf{D}^H\mathbf{D}\mathbf{F}\vec{\rho}_0) \\
&= \frac{1}{\alpha+1}\mathbf{F}^H\mathbf{D}^H\vec{y} + \vec{\rho}_0 - \frac{1}{\alpha+1}\mathbf{F}^H\mathbf{D}^H\mathbf{D}\mathbf{F}\vec{\rho}_0 \\
&= \vec{\rho}_0 + \frac{1}{\alpha+1}\mathbf{F}^H\mathbf{D}^H(\vec{y} - \mathbf{D}\mathbf{F}\vec{\rho}_0)
\end{aligned} \tag{3.19}$$

where $\vec{\rho}_0$ represents the current estimate of $\vec{\rho}$ in the iteration. The last expression demonstrates the potential efficiency in the absence of regularization on $\vec{u}$; however, it also indicates the importance of the regularization. First, performance of the algorithm is limited by the conditioning of $\mathbf{DF}\vec{\rho}(\cdot,\cdot)$, which represents the conditioning of the original reconstruction problem represented by (3.4). Since the conditioning is actually very poor [47,49], $\vec{y} - \mathbf{DF}\vec{\rho}_0$ can be very small even when $\vec{\rho}_0$ is far away from the true solution. Thus, $\hat{\vec{u}} \approx \vec{\rho}_0$ and the iterate would stop at that point. This observation also implies that the accuracy of the auxiliary variable algorithm cannot outperform the gradient-based methods, although the efficiency can be improved. Second, the vector $\frac{1}{\alpha+1}\mathbf{F}^H\mathbf{D}^H(\vec{y} - \mathbf{DF}\vec{\rho}_0)$ can be further reduced in its modulus by the projection operator in Step 2, which projects the vector into the set of exponential functions. This inference can be analytically verified when $\vec{\rho}_0$ is sufficiently close to the ground truth. Qualitatively, the statement can be reasoned from the fact that the mapping $\mathbf{F}^H\mathbf{D}^H$ tends to generate a vector that is rough in the temporal domain. Notice that $\mathbf{F}^H\mathbf{D}^H$ is nothing but an inverse Fourier transform of a randomly sampled spectrum with zero-filling. It is readily established in compressed sensing that random sampling leads to a "noise-like" signal in the Fourier domain [68]. The projection operator in Step 2 of the alternating direction method projects any signal to the nearest exponential function in terms of the Euclidean norm. The projection can smooth the "noisy" signal, which may result in $\vec{\rho}$ changing very little across two iterations. In (3.7), this phenomenon means there are many

noisy exponential functions which are close to an exponential function in l_2 norm but also greatly reduce the residual in the data space. This is reasonable since the amount of data is very limited for a single-shot trajectory. This phenomenon also implies that the penalty method may increase the degree of ill-conditioning, a characteristic of the method often seen in other literature [24, 37].

Regularization of $\vec{u}$ makes the estimate in (3.19) much smoother in both the spatial and temporal domains. The latter smoothing reduces the problem caused by $\mathbf{F}^H\mathbf{D}^H$. This observation explains why regularization of $\vec{u}$ is necessary. On the other hand, the ill-conditioning of $\mathbf{DF}\vec{\rho}(\cdot,\cdot)$can be improved by regularizing $\vec{m}$ and $\vec{z}$, as we have done in [49]. Thus, these regularization penalties are also necessary. Finally, the problem caused by $\mathbf{F}^H\mathbf{D}^H$ may be completely eliminated by avoiding the penalty method, which transforms (3.6) to (3.7). This is because (3.7) is only an approximation of (3.6) that relaxes the constraint—an indicator operator for the set of exponential functions—to a distance operator. If $\vec{u}$ is always confined to the set of all exponential functions, then Step 1 would no longer lead to an oscillatory estimate in $\vec{u}$. This is a subject for further investigation.

3.0.4 Memory usage

For 2-D imaging, the auxiliary variable $\vec{u}$ resides in a 3-D spatial-temporal space. The number of temporal samples in a single-shot trajectory is very large. For example, for a 64×64 image, neglecting the temporal variation, the number of k-space samples—and therefore temporal sampling points—needs to be at least 8192 to keep the reconstruction overdetermined. Hence, $\vec{u}$ is of size $64^2 \times 8192$ and the size is generally equal to $2N^4$ where N is the image size of one dimension. This size is very large and can be non-affordable for modern computers when N grows to 128. A practical solution is to use a coarser grid in the temporal domain, by assuming samples acquired within a small time interval can be modeled as acquired at the same time. For example, we partition the entire acquisition window of 81.92 msec into 128 segments. Since the effect of decay and off-resonance is very subtle

within such a small time window, the accuracy of such rounding in the temporal domain is very high. This approximation allows the size of $\vec{u}$ to drop to $64^2 \times 128$ for $N = 64$ In the example. Moreover, the strategy causes the size of $\vec{u}$ more dependent on the data readout time which is usually rather fixed, increasing the scalability of the memory usage.

3.0.5 The hybrid method

Our experience shows that the proposed auxiliary variable algorithm is much faster than the ordinary trust region algorithm during the early stage of the iteration. This advantage is at least partially due to the reduced computation in the alternating direction method. The constrained optimization formulation may also help accelerate convergence of $\vec{m}$ and $\vec{z}$ in the parameter space. The constrained optimization formulation creates a quadratic signal estimation problem in Step 1 where the signal is the vector $\vec{u}$. Thus, the solution to this problem in each iteration is a global minimizer. In (2.1), however, the cost function is non-quadratic and therefore the estimate of $\vec{m}$ and $\vec{z}$ in each iteration must be restricted in a local space to keep the linearization valid [49]. The variation of the parameter maps is therefore limited. However, as the number of iterations increases, the convergence is gradually slowed down until it is slower than the ordinary trust region algorithm, as discussed previously. To address this problem, we propose a hybrid method where the first phase uses the alternating direction algorithm.. As the algorithm becomes too slow, we switch to the OTR algorithm to refine the estimate. Notice that the OTR refinement is much simpler than the OTR continuation algorithm proposed in [49]. The OTR continuation algorithm requires that the regularization parameters be gradually reduced so that convergence from poor initializations can be reached. Typically, OTR refinement converges within 10 iterations after the AV algorithm initialization by activating the stopping rule of either sufficiently low gradient or iterate variation. The small number of iterations requires much less time compared to the original OTR continuation algorithm, which typically requires 40-60 iterations. Below is a summary of the proposed AV algorithm. Let $\xi_{1,2,3} \in \mathbb{R}^+$. Let $J_o, J_i \in \mathbb{Z}^+$.

- Set $\beta_{1,2}$, α, $\lambda_{1,2}$. $\xi_{1,2,3}$. Initialize $\vec{m}$ and $\vec{z}$.
- Calculate $\mathbf{D}^H\vec{s}$.
- Calculate the inverse of $\mathbf{D}^H\mathbf{D}+\beta_1\mathbf{X}^H\mathbf{X}+\beta_2\text{diag}(\mathbf{E}^H\mathbf{E})+\alpha\mathbf{I}_L$ (preconditioner for step 1) using the Woodbury formula.
- *For* $j_o = 1, \ldots, J_o$
  - *For* $j_i = 1, \ldots, J_i$
    - Calculate $\mathbf{D}^H\vec{s}$ (precomputed) $+\alpha\mathbf{F}\vec{\rho}(\vec{m},\vec{z})$.
    - Estimate $\vec{u}$ through (3.13) by using PCG.
    - Estimate $\vec{m}$ and $\vec{z}$ given update of $\vec{u}$.
  - End *For*
  - $\beta_1 = \beta_1/\xi_1$, $\beta_2 = \beta_2/\xi_2$, $\alpha = \alpha\cdot\xi_3$
- End *For*
- Run OTR refinement program.

3.0.6 Parameter selection

There are many parameters for the proposed hybrid method, namely $\beta_{1,2}$, α, $\lambda_{1,2}$, $\xi_{1,2,3}$, J_o, and $J_i(j_o)$ for the alternate direction algorithm, and $\bar{\lambda}_{1,2}$ for the OTR refinement algorithm. However, we have observed that most of these perform well with a constant value for a wide range of objects (images), SNRs, and algorithm initializations. α is the initial value of the penalty weight. α needs to be increased a few times in the algorithm to enforce the constraint in (3.6). Based on (3.19), we found a suitable range for the starting value of α to be $[0.1, 1]$. We chose $\alpha = 0.1$ for all our experiments. $\lambda_{1,2}$ and $\bar{\lambda}_{1,2}$ are regularization parameters for $\vec{m}$ and $\vec{z}$. They only depend on the smoothness of the two maps, which is

quite similar even for different objects. In this work, $\lambda_1 = \bar{\lambda}_1 = 1$ and $\lambda_2 = \bar{\lambda}_2 = 10^{-4}$ for all tested objects. Other parameters for OTR can be found in [49]. Similarly, the algorithm is quite robust to the choice of $\beta_{1,2}$. We used $\beta_1 = 10^4$ and $\beta_2 = 1$ in this work. $\xi_{1,2,3}$ are the variation factors of β_1, β_2, and α. Clearly, choices of $\xi_{1,2,3}$ depend on the choices of β_1, β_2, and α, respectively. In this paper, $\beta_1 = \beta_2 = \beta_3 = 10$ performs well. J_o and $J_i(j_o)$ are the maximal iterations for the penalty loop (outer loop) and the alternate direction method loop (inner loop). For the hybrid method, they were chosen to be $J_o = 2$, $J_i(1) = 12$, and $J_i(2) = 4$.

3.1 Experiments

We tested the AV algorithm with a range of simulations and MRI experiments. For both simulation and real data acquisition, we used a rosette trajectory (either single-shot or multi-shot), since it has been verified to be superior to other trajectories in terms of the reconstruction quality for this problem [60, 102]. A single-shot rosette trajectory was formulated as

$$\vec{k}_l = \frac{1}{2} k_{max} \sin(w_{osc} t_l) e^{\iota w_{rot} t_l + \theta} \tag{3.20}$$

where k_{max} represents the k-space range, w_{osc} the oscillation frequency, w_{rot} the rotation frequency, and θ the starting direction of the trajectory in k-space. In our work, $w_{osc} =$ 3196 rad/sec, $w_{rot} = 1577$ rad/sec, the time span of the trajectory is 81.92 ms and a total of 8192 samples were acquired. For a single-shot rosette, θ=0° in this work. For a multi-shot rosette, θ was uniformly distributed between 0° and 90° for each interleave. In particular, for a four-shot rosette we used θ = 0°, 22.5°, 45°, and 67.5°, respectively. Our experience shows such an arrangement of the angles leads to a uniform sampling of k-space.

In the simulation, we added white Gaussian noise to the model-predicted signal. The signal-to-noise ratio (SNR) in this work is defined as

$$\text{SNR} = \frac{\|s\|_2}{\|s - s_0\|_2}$$

where s is the noisy data and s_0 is the noiseless data. Normalized root mean square error (NRMSE) was used as a metric for the accuracy of the reconstructions. The NRMSE is defined as:

$$\text{NRMSE} = \frac{\|f - f_0\|_2}{\|f_0\|_2}$$

where f and f_0 represent the estimate and the ground truth. All algorithms were started at a trivial initialization: $m_n = 0.5$ and $z_n = 0$ for all n. Masking was used to mask out voxels with low spin density before applying any iterative algorithm. This technique has been commonly used in dealing with similar problems [83, 97]. All algorithms were implemented using MATLAB and run with an Intel Core i7-4700MQ CPU.

3.1.1 Simulated phantom

A nested cylinder phantom with four small cylinder containers contained in a large cylinder container was simulated. Each cylinder was defined with different materials of varying spin density, R_2^* decay, and off-resonance. Since the off-resonance map is often smooth in reality [98], we smoothed it from a sharp image like the spin density map by a circular averaging filter with a radius of 5 voxels. The units for the R_2^* decay and off-resonance frequency are sec^{-1} and Hz. Figure 3.1 (a), (f), (k) show the three parameter maps of the phantom respectively.

Figure 3.1 and 3.2 show the reconstruction performance of AV (with and without refinement), OTRC, and NCGC for the simulated phantom. AV without refinement has four continuation phases. The maximum number of iterations is 12-12-12-12, and the notation is defined in Section 3.0.6. AV with refinement has 2 continuation phases, and the maximum

number of iterations is 8-4. Other parameters of the two AV-based algorithms as well as OTRC and NCGC are given in Section 3.0.6. Figure 3.1 shows that AV with refinement, OTRC, and NCGC have about the same level of accuracy. Some artifacts are present, especially in boundary regions. These artifacts are probably caused by the heavy downsampling of the single-shot acquisition and are therefore unrelated to the chosen algorithm. Increasing k-space samples or using a better edge-preserving regularization can probably reduce these artifacts. However, these improvements are not the concern of this study. Figure 3.1 shows that AV without refinement results in a large estimation error. This kind of error is commonly seen in all our experiments.

Figure 3.2 shows the convergence profile of each algorithm for the same phantom. It is clearly seen that AV with refinement is much faster than OTRC and NCGC. Since OTR is used as the refinement algorithm, the speed gain mainly comes from the AV algorithm used to provide the initialization. Comparison between AV with refinement and AV without refinement demonstrates the critical contribution of the refinement. AV without refinement converges very fast in the early stage but then quickly slows down. In the discussion section, we present some explanations for this behavior. By replacing the later stage of AV with OTR, the overall speed gain of the hybrid algorithm is superior to any other algorithms presented in this work.

Figure 3.3 compares different regularization strategies on $\vec{u}$ to show that the $RE{+}E$ combination outperforms other strategies. The same simulated phantom was used and regularization parameters were adjusted separately for each regularization combination using a thorough search. The separate adjustment guarantees the optimality of the parameter selection for each regularization and a fair comparison between them. We found that the temporal regularization (E) of $\vec{u}$ is more important than the spatial regularization (R). $RE{+}E$, $R{+}E$, and E almost always perform better than R and no regularization. Although temporal regularization also makes the computation in Step 1 much more complicated, the regularization is still necessary in our current AV implementation.

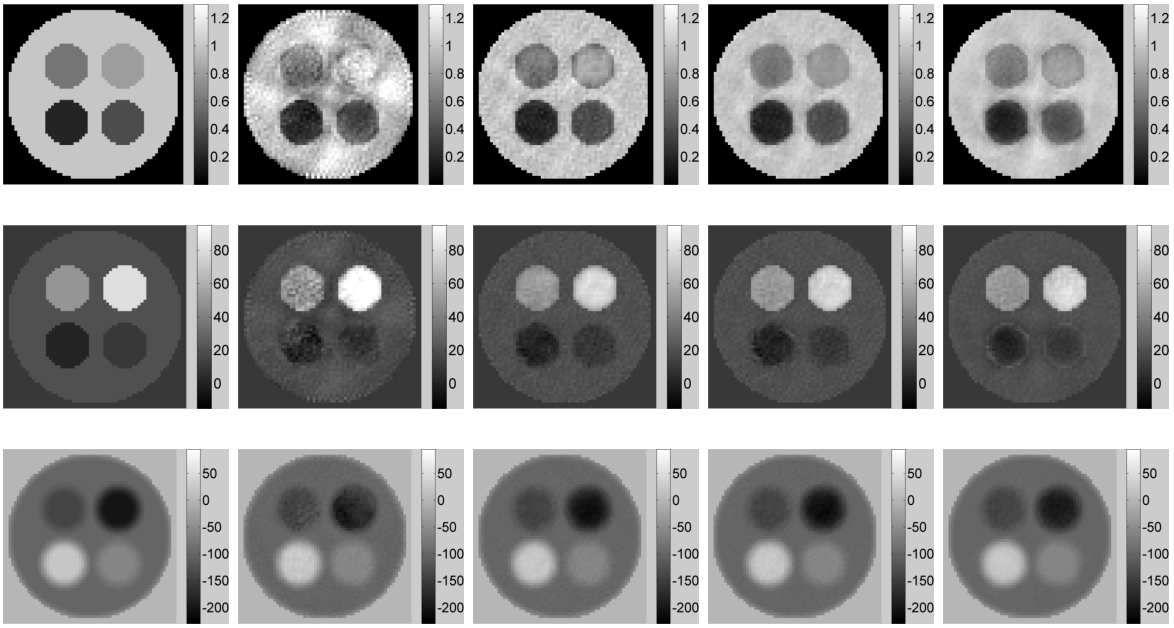
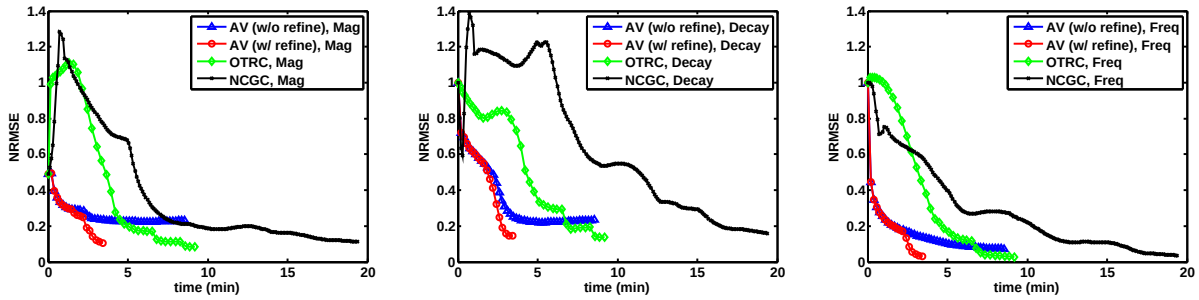


Figure 3.1: Reconstruction results of AV (w/o refinement), AV (w/ refinement), OTRC, and NCGC on a synthetic phantom and a rosette trajectory. The first, second, and third rows show the maps of phantom and these algorithms for spin density, R_2^* decay, and off-resonance frequency, respectively.

Figure 3.4 demonstrates the robustness of the AV algorithm for this problem. In this example, the AV algorithm was run without refinement to show the robustness of the algorithm alone. The robustness of OTR was shown in [49] and is not elaborated again in this paper. Our experiments show that variation of the two regularization parameters causes only a small accuracy variation. The algorithm is robust for a considerably large range for the two regularization parameters.

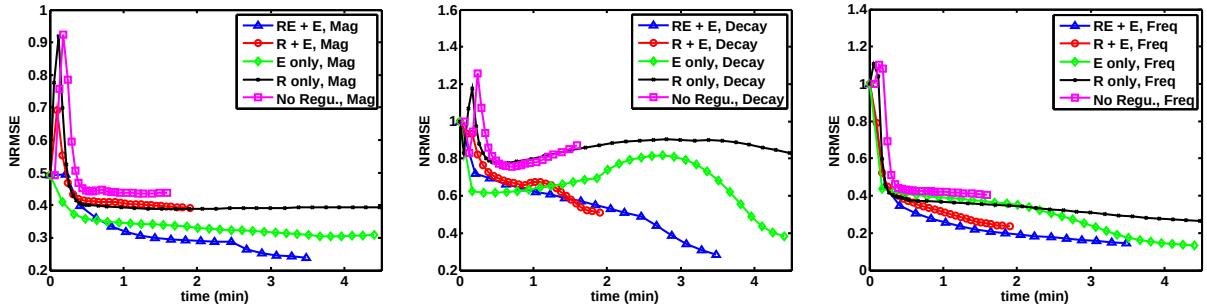
3.1.2 Real phantom and *in vivo* data

Application of the proposed algorithm to real data is our goal; however, high-quality reconstructions are much more difficult to achieve than with simulated data. Challenges of real-data reconstruction arise from various factors, such as pulse sequence programming, trajectory miscalibration, and even model mismatch caused by intra-voxel gradients [75, 80]. To enable analysis of the reconstruction results, we must establish a gold standard. A natural



(a) spin density (b) R_2^* decay (c) off-resonance frequency

Figure 3.2: Convergence profiles of AV (w/o refinement), AV (w/ refinement), OTRC, and NCGC for the simulated phantom

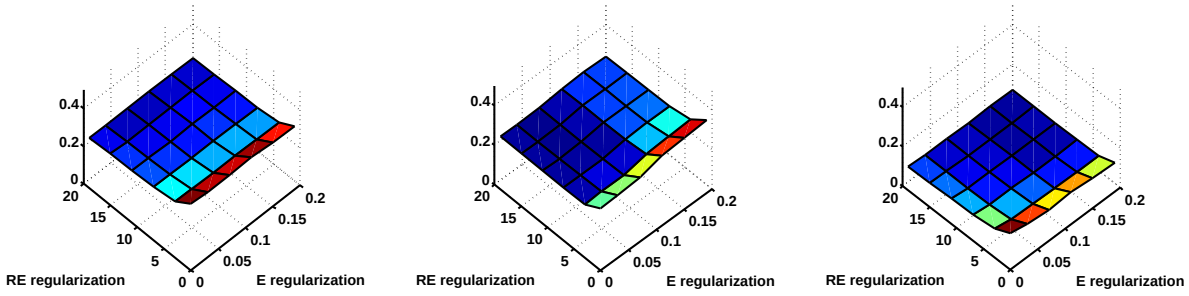


(a) spin density (b) R_2^* decay (c) off-resonance frequency

Figure 3.3: Convergence of the AV method with different regularizations on $\vec{u}$.

choice is to use a multi-echo FLASH sequence. In our work, we used a multi-echo FLASH sequence developed in-house. Various parameters of the sequence are: TE = 5, 6, 7, 8, $\cdots$ 82 ms, TR = 200 ms, FOV = 120 mm $\times$ 120 mm, slice thickness = 2 mm, bandwidth = 390 Hz/Pixel, image size = 64 $\times$ 64, flip angle = 15°. The long echo train is used to reduce estimation bias caused by noise. All experiments were done on a SIEMENS MAGNETOM Verio 3T machine with a 32-channel head coil.

We used a four-shot rosette trajectory whose parameters are described at the beginning of the section. After we obtained the three gold-standard parameter maps from the multi-echo sequence, we synthesized k-space data using the estimated parameters in the discrete model (2.2) to investigate the level of error associated with acquiring real data on a rosette trajectory. To synthesize k-space data, an accurate measurement of the rosette trajectory is needed to address the trajectory distortion caused by many factors such as eddy current.



(a) spin density (b) R_2^* decay (c) off-resonance frequency

Figure 3.4: Algorithm robustness to regularization parameters β_1 and β_2.

We used the calibration technique described in [118] and [25] to do the calibration. The two calibration techniques are very similar to each other. We found that the method of [25] is slightly better than that of [118]. However, with both calibration techniques, the difference between the synthetic data and the acquired data is still very large. With such a large difference, the reconstruction accuracy does not completely reflect the actual performance of these algorithms. However, their convergence speed can still be compared with each other. Moreover, we are still investigating the reason for the large mismatch.

Phantom reconstruction

We made a phantom from a large peanut butter jar containing four small test tubes, and each of them was filled with tap water mixed with different resolutions of copper sulfate. The copper sulfate was used to increase R_2 and therefore also R_2^* as well as to change magnetic susceptibility. Figure 3.5 compares AV without and with refinement, OTRC, and NCGC for reconstruction of this real phantom. The left column shows the gold-standard result obtained from the multi-echo sequence. All parameters related to the AV formulation (3.8) are the same as the simulation. The two regularization parameters used with the OTR refinement, OTRC, and NCGC are changed to 10^2 and 10^{-1} respectively . All reconstruction results are quite noisy, since we used small regularization parameters to reduce regularization-induced artifacts. AV with refinement leads to the best result. Figure 3.6 shows the convergence profiles of all four algorithms. The advantage of the auxiliary variable algorithm in the first

few iterations can be clearly seen. The AV algorithm with refinement achieves the best balance in keeping computation time short while reducing estimation error.

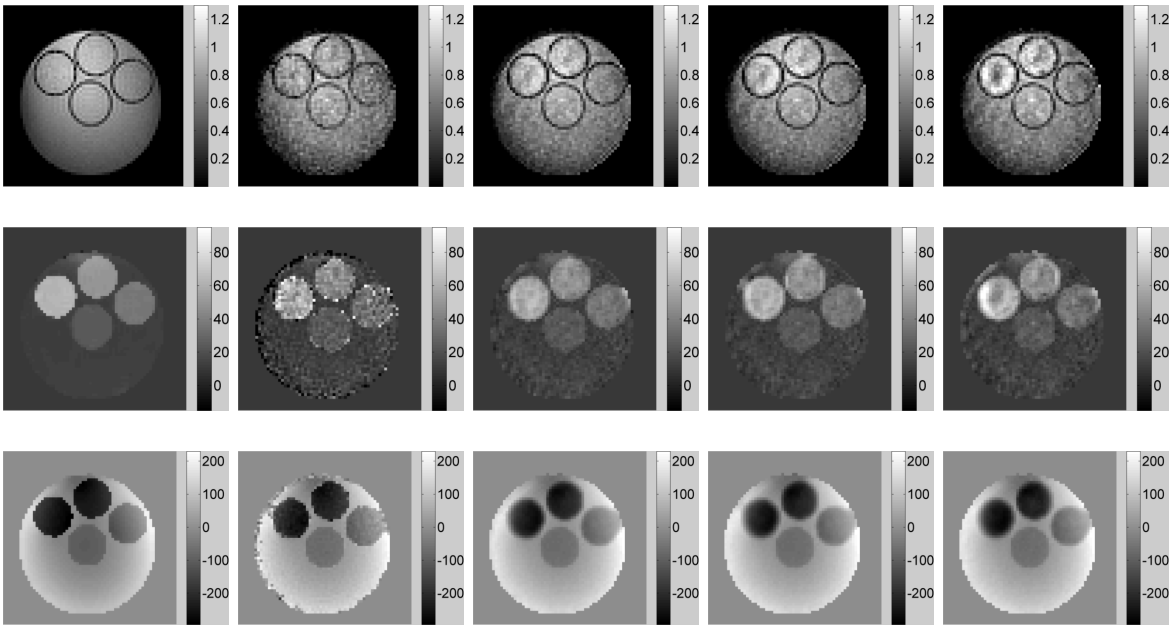
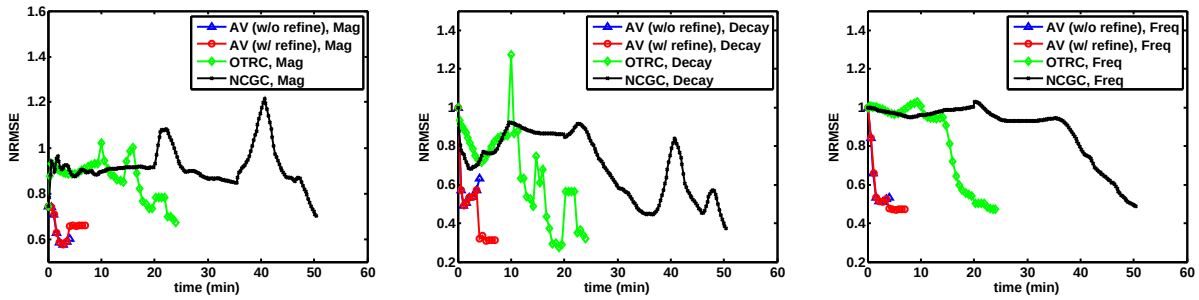


Figure 3.5: Comparison of various algorithms for a real phantom and a rosette trajectory. Column 1-5 show the gold standard, the reconstruction of AV (w/o refinement), AV (w/ refinement), OTRC, and NCGC, respectively. The first, second, and third rows show the maps of phantom and these algorithms for spin density, R_2^* decay, and off-resonance frequency, respectively.

In vivo reconstruction

An axial slice of a human head was scanned with the multi-echo sequence and the four-shot rosette trajectory. The gold standard based on the data from the multi-echo sequence is shown in the left column of Figure 3.7. Reconstructions from the four algorithms are shown in the remaining part. AV without refinement led to a result with abnormally-valued, discontinuously-shaped artifacts. The artifacts were removed after the OTR refinement. Similarly good results were present for OTRC and NCGC. Some geometric distortions are present, such as the wider appearance of the spin density map and the expanded ventricle in the R_2^* decay map. We suspect that this is caused by miscalibration of the trajectory that may lead to unmodeled phase accumulation. It is known that phase accumulation can cause



(a) spin density (b) R_2^* decay (c) off-resonance frequency

Figure 3.6: Convergence profiles of AV (w/o refinement), AV (w/ refinement), OTRC, and NCGC for the real phantom

geometric distortion in EPI reconstructions. Figure 3.8 shows the convergence profiles of the four algorithms. The time advantage of the AV-based methods is still quite apparent.

3.2 Discussion and Conclusions

We have presented an auxiliary variable (AV) algorithm to jointly reconstruct spin density, R_2^* decay, and off-resonance frequency maps in MRI. A strong body of literature has developed powerful iterative algorithms to reconstruct the three maps with sparsely sampled multi-echo acquisition (SSME), and single- or multi-shot acquisition (SS/MS). Gradient-based algorithms have been successfully applied in both acquisition cases (e.g. [7, 51, 119] for SSME and [83, 102, 103] for SS/MS). These algorithms are usually straightforward and generally converge and achieve good accuracy when initialization conditions are satisfied. Splitting-based algorithms have been studied by several authors for SSME acquisition [22, 72, 107]. These algorithms often aim to reconstruct an image sequence ($\vec{u}$ in our paper) with either a structure constraint that the sequence satisfies a model [22] and/or sparsity constraints [22, 72, 107]. They usually work quite well with SSME acquisition when the sampling reduction factor is less than 8. However, for the SS/MS acquisition, we have not seen any work using a splitting-based method. Several challenges arise in applying a splitting-based method for SS/MS acquisition. First, since k-space data is continuously acquired in time, it is not intuitive to view the problem as a constrained optimization problem

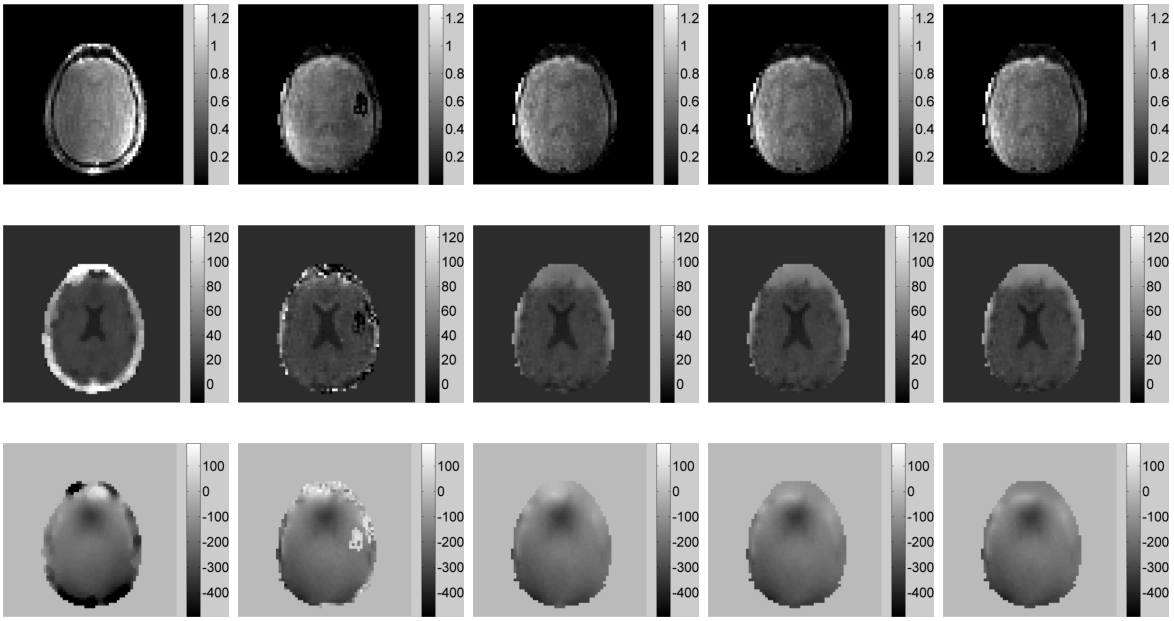
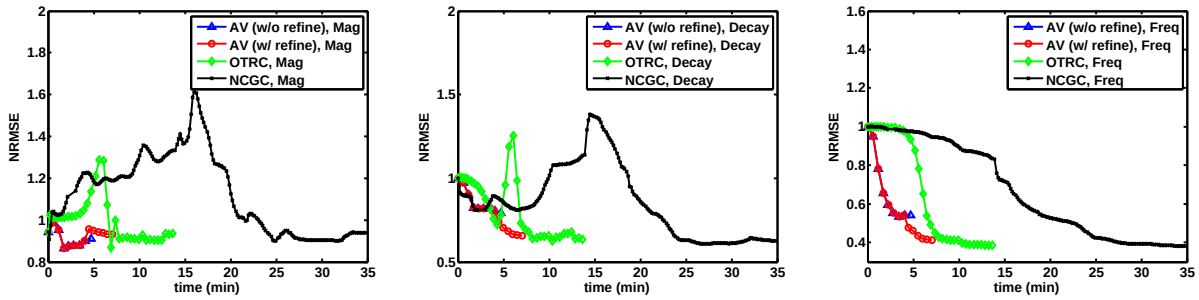


Figure 3.7: Comparison of various algorithms for *in vivo* data and a rosette trajectory. The column 1-5 show the gold standard, the reconstruction result of AV (w/o refinement), AV (w/ refinement), OTRC, and NCGC, respectively. The first, second, and third rows show the maps of phantom and these algorithms for spin density, R_2^* decay, and off-resonance frequency, respectively.

as is naturally formulated for any multi-echo acquisition. An auxiliary variable must be introduced to model the "multi-echo" image sequence, which does not actually exist with SS/MS acquisition. Second, since the number of k-space samples in SS/MS for each "echois considerably less than in SSME acquisition, whether splitting-based methods can converge or achieve good accuracy for SS/MS remains unclear. For example, we found [22] actually uses a similar cost function and exactly the same alternating direction algorithm to reconstruct T_1 or T_2 through an SSME acquisition. The paper uses the same structure constraint and a sparsity constraint over $\vec{u}$ and achieves a good T_1 or T_2 reconstruction (without off-resonance frequency). However, when almost the same algorithm is applied to reconstructing the three parameter maps in SS/MS acquisition, we observed a much slower convergence (later stage) as well as worse estimate accuracy. Therefore, the impact of the acquisition strategy on algorithm performance remains an open question. By continuing to work on this question,



(a) spin density (b) R_2^* decay (c) off-resonance frequency

Figure 3.8: Convergence profiles of AV (w/o refinement), AV (w/ refinement), OTRC, and NCGC for the *in vivo* data

we hope to compare different acquisition strategies and form insights on how to select the sampling scheme and its specific parameters (e.g. number of echoes, number of shots, etc.).

The penalty method may not be the optimal algorithm for solving the constrained problem (3.6). It is well known that the penalty method can be inferior and numerically unstable due to increased ill-conditioning with the increasing penalty weight [24, 37]. To address this issue, we proposed a hybrid algorithm that uses the OTR algorithm in the second phase, since OTRhas been shown to have good local convergence performance [49]. A hybrid strategy of combining the penalty method and other methods has been employed by many other researchers to address constrained optimization problems [89, 106]. However, other algorithms may hold more promise for this problem than the penalty method based algorithm. In particular, methods such as forward-backward splitting or the alternating direction method of multipliers (ADMM) may be able to achieve local convergence performance similar to the OTR algorithm while maintaining the speed advantage. [14, 27, 88].

Two other shortcomings of the hybrid algorithm may be overcome in future work. First, the Tikhonov regularization terms over $\vec{m}$ and $\vec{z}$ can be replaced by edge-preserving regularization strategies that have been commonly used in the SSME acquisition. The latter regularization may not only improve performance of the algorithm but also may facilitate a comparison among different acquisition strategies. Although adding a nondifferentiable regularization term introduces more computational complexity for the gradient-based algorithms,

this strategy is much easier for splitting-based algorithms since the regularization terms only appear in one step of the splitting-based algorithm. Second, the computational cost in Step 1 when a non-Cartesian trajectory is used is relatively high due to the PCG algorithm. It would be highly desired in this step if non-Cartesian trajectories could be addressed with a computational complexity similar to Cartesian trajectories. If this can be achieved, extension of the approach to high-resolution reconstruction of these parameter maps can be made much more practical. We are currently working to solve these two issues. The nonconvexity of the exponential functions may cause stability issue for the splitting-based algorithm, since most of these algorithms assume a convex cost function. However, our experience with the alternating direction algorithm does not show any sign of instability. Moreover, several authors have used principle component analysis (PCA)-based approximation to linearize the set of decaying exponential functions for T2 mapping [51, 84, 107]. Similar linearization methods have also been used in MR fingerprinting [70]. These techniques can be used in our problem to linearize the constraint in (3.6), making the cost function convex.

In conclusion, we have proposed a hybrid algorithm for jointly reconstructing spin density, R_2^* decay, and off-resonance frequency maps. Overall, the proposed hybrid algorithm achieves the best balance between efficiency and accuracy, demonstrated by both simulated and *in vivo* data. The success of these results suggests that more advanced splitting-based algorithms might be developed to replace use of the refinement algorithm and further improve the time efficiency. .

Chapter 4

A novel non-fit estimator of decaying exponential functions

A difficulty of relaxation mapping problems in MRI is the fact that the mathematical relation between the spatial-temporal signal (the sequence of decaying images) and the relaxation maps is nonlinear. No matter which relaxation parameter is of interest among T_1, T_2, or T_2^*, the spatial-temporal signal must satisfy an exponential decay (T_2 or T_2^*) or rise (T_1) based on the relaxation rate. This nonlinear structure constraint of the spatial-temporal signal not only increases numerical complexity but also increases the difficulty in analytical investigation. It is therefore interesting if we can find a new formulation which is linear or can be linearized easily. This is possible if we consider the frequency-temporal domain instead of the spatial-temporal domain. Since the signal in the spatial-temporal domain must satisfy the mono-exponential modeling, the signal in the spatial-temporal domain must satisfy a convolution relationship which is linear.

In this chapter, we propose a novel non-fit estimator of the spin density map $\vec{m}$ and the complex frequency map $\vec{z}$. Traditionally, when multi-echo data is acquired, the data is inverse Fourier transformed to spatial-temporal domain and then a curve fitting is applied to find out $\vec{m}$ and $\vec{z}$. The proposed non-fit estimator can replace the curve fitting to estimate the two parameter sets. We first derive the new estimator, and then show its comparison with two commonly used curve fitting methods through simulated data. The new estimator has some inherent drawbacks and we propose two modifications of the estimator to reduce the drawbacks. The result shows that the new modified estimator is much faster than the other two estimators while keeping a similar estimation accuracy when number of echoes is small. Moreover, the new modified estimator can be potentially useful in the auxiliary variable method to improve the computational efficiency.

4.1 A new formulation of the mono-exponential model

In this section, we derive a new formulation of the mono-exponential model. The new formulation is in a linear form, which may provide convenience in related analysis and numerical computations. The derivation is based on the fact that a k-space frame is a convolution of a previous k-space frame with a function that only depends on the complex frequency map $\vec{z}$ and the time between the two frames.

Like the memory-saving technique we used in the auxiliary variable algorithm, we divide the temporal domain into multiple short segments, each taking around 1 millisecond. This short length guarantees that the variation of the signal within the segment is small, and the error of analysis is therefore also small. Let's abbreviate the temporal-frequency space as k-t space. Notice that the k-t space used in our application is slightly different from the k-t space used in dynamic MRI (see Section 1.3.8) in that our application assumes a deterministic model in the temporal dimension. Let $\vec{v}$ represent the signal in the t-k space. Assume there are Q k-frames and N^2 voxels in the spatial domain. Then the vector $\vec{v}$ has dimension QN^2. Assume $Q \geq 2$. Let $(\vec{v}_q)_{q=1}^{Q}$ represent the Q k-space frames. Let $\check{v}$ and $\hat{v}$ represent the first and the last $(Q-1)N^2$ elements of $\vec{v}$, respectively. That is,

$$\check{v} \triangleq [\vec{v}_1^T, \vec{v}_2^T, \ldots, \vec{v}_{Q-1}^T]^T \tag{4.1}$$

$$\hat{v} \triangleq [\vec{v}_2^T, \vec{v}_3^T, \ldots, \vec{v}_Q^T]^T \tag{4.2}$$

Let $\vec{\mathbb{z}} \triangleq e^{-\vec{z}\Delta t}$ where Δt is the time interval between two k-space frames. Notice that the k-space frames must be spaced equally in our analysis. Let $\mathbf{C}_{\check{v}}$ represent a tall matrix that is stacked up by $C_{\vec{v}_q}$ for $q = 1, 2, \ldots, Q-1$. $C_{\vec{x}}$ for a vector $\vec{x}$ represents a convolution matrix which is a circulant matrix whose first column is $\vec{x}$. Mathematically,

$$\mathbf{C}_{\check{v}} \triangleq [C_{\vec{v}_1}^T, C_{\vec{v}_2}^T, \ldots, C_{\vec{v}_{M-1}}^T]^T \tag{4.3}$$

We then can represent the mono-exponential structure constraint of $\vec{u}$ in the spatial-temporal domain as

$$\hat{v} = \mathbf{C}_{\check{v}}\vec{\mathbb{z}} \tag{4.4}$$

which is a linear equation in k-t space. This condition is equivalent to the mono-exponential structure constraint in the spatial-temporal domain. Since $\mathbf{C}_{\check{v}}$ is a "tall" matrix, the minimum-norm solution $\hat{\vec{\mathbb{z}}}$ of (4.4) is

$$\hat{\vec{\mathbb{z}}} = (\mathbf{C}_{\check{v}}^H\mathbf{C}_{\check{v}})^{-1}\mathbf{C}_{\check{v}}^H\hat{v} \tag{4.5}$$

Thus,

$$\|\hat{v} - \mathbf{C}_{\check{v}}(\mathbf{C}_{\check{v}}^H\mathbf{C}_{\check{v}})^{-1}\mathbf{C}_{\check{v}}^H\hat{v}\|^2 = 0 \tag{4.6}$$

for any $\vec{v}$ that satisfies (4.4). If for a $\vec{v}$, (4.6) is not satisfied, (4.4) is not satisfied either since the minimum-norm solution is unique when $Q \geq 2$. Therefore (4.6) is an equivalent condition to the mono-exponential structure constraint.

For each $q = 1, 2, \ldots, Q$,

$$C_{\vec{v}_q} = F_0\vec{\mathbb{u}}_q F_0^H \tag{4.7}$$

where $\vec{\mathbb{u}}_q \triangleq \text{diag}(\vec{u}_q)$, F_0 represents the 2D Fourier transform matrix, and $\vec{u}_q$ represents the qth image in the image sequence (spatial temporal domain). Due to (4.3), $\mathbf{C}_{\check{v}}$ is a stacked matrix of $F_0\vec{\mathbb{u}}_q F_0^H$ for $q = 1, 2, \ldots, Q-1$. We then have

$$\mathbf{C}_{\check{v}}^H\mathbf{C}_{\check{v}} = \sum_{q=0}^{Q-1} F_0\vec{\mathbb{u}}_q^H F_0^H F_0\vec{\mathbb{u}}_q F_0^H = F_0\sum_{q=1}^{Q-1}(\vec{\mathbb{u}}_q^H\vec{\mathbb{u}}_q)F_0^H = F_0\sum_{q=1}^{Q-1}(|\vec{\mathbb{u}}_q|^2)F_0^H \tag{4.8}$$

where $|\vec{\mathbb{u}}_q|^2$ represents the point-wise square of the absolute value of the diagonal matrix $\vec{\mathbb{u}}_q$. Hence,

$$(\mathbf{C}_{\check{v}}^H\mathbf{C}_{\check{v}})^{-1} = F_0(\sum_{q=1}^{Q-1}(|\vec{\mathbb{u}}_q|^2))^{-1}F_0^H, \tag{4.9}$$

$$\mathbf{C}_{\check{v}}(\mathbf{C}_{\check{v}}^H\mathbf{C}_{\check{v}})^{-1}\mathbf{C}_{\check{v}}^H = \begin{bmatrix} F_0\vec{\mathfrak{u}}_1 \\ F_0\vec{\mathfrak{u}}_2 \\ \vdots \\ F_0\vec{\mathfrak{u}}_{Q-1} \end{bmatrix} (\sum_{q=1}^{Q-1}(|\vec{\mathfrak{u}}_q|^2))^{-1} \begin{bmatrix} \vec{\mathfrak{u}}_1^H F_0^H & \vec{\mathfrak{u}}_2^H F_0^H & \dots & \vec{\mathfrak{u}}_{Q-1}^H F_0^H \end{bmatrix}, \quad (4.10)$$

and the left-side expression

$$\|\hat{v} - \mathbf{C}_{\check{v}}(\mathbf{C}_{\check{v}}^H\mathbf{C}_{\check{v}})^{-1}\mathbf{C}_{\check{v}}^H\hat{v}\|^2 \quad (4.11)$$

$$= \| \begin{bmatrix} F_0\vec{u}_2 \\ F_0\vec{u}_3 \\ \vdots \\ F_0\vec{u}_Q \end{bmatrix} - \begin{bmatrix} F_0\vec{\mathfrak{u}}_1 \\ F_0\vec{\mathfrak{u}}_2 \\ \vdots \\ F_0\vec{\mathfrak{u}}_{Q-1} \end{bmatrix} (\sum_{q=1}^{Q-1}(|\vec{\mathfrak{u}}_q|^2))^{-1} \begin{bmatrix} \vec{\mathfrak{u}}_1^H F_0^H & \vec{\mathfrak{u}}_2^H F_0^H & \dots & \vec{\mathfrak{u}}_{Q-1}^H F_0^H \end{bmatrix} \hat{v}\|^2 \quad (4.12)$$

$$= \| \begin{bmatrix} F_0\vec{u}_2 \\ F_0\vec{u}_3 \\ \vdots \\ F_0\vec{u}_Q \end{bmatrix} - \begin{bmatrix} F_0\vec{\mathfrak{u}}_1 \\ F_0\vec{\mathfrak{u}}_2 \\ \vdots \\ F_0\vec{\mathfrak{u}}_{Q-1} \end{bmatrix} (\sum_{q=1}^{Q-1}(|\vec{\mathfrak{u}}_q|^2))^{-1} \sum_{q=1}^{Q-1}(\vec{u}_q^*\vec{u}_{q+1})\|^2 \quad (4.13)$$

Let $\mathbf{F}_{Q-1} \triangleq \text{diag}(\underbrace{F_0, F_0, \dots, F_0}_{Q-1})$. Let $\vec{\lambda} \triangleq (\sum_{q=1}^{Q-1}(|\vec{\mathfrak{u}}_q|^2))^{-1}\sum_{q=1}^{Q-1}(\vec{u}_q^*\vec{u}_{q+1})$. Notice $\vec{\lambda}$ is an N^2-long vector whose nth element λ_n satisfies

$$\lambda_n = \sum_{q=1}^{Q-1}(u_{n,q}^* u_{n,q+1}) / \sum_{q=1}^{Q-1}(|u_{n,q}|^2). \quad (4.14)$$

Let $\Lambda \triangleq \text{diag}([\underbrace{\vec{\lambda}^T, \vec{\lambda}^T, \dots, \vec{\lambda}^T}_{Q-1}]^T)$. Then (4.13) is equal to

$$\|\mathbf{F}_{Q-1}(\hat{u} - \Lambda\check{u})\|^2 = \|\hat{u} - \Lambda\check{u}\|^2 \quad (4.15)$$

where $\hat{u}$ and $\check{u}$ are defined in a similar manner with $\hat{v}$ and $\check{v}$.

$$\check{u} \triangleq [\vec{u}_1^T, \vec{u}_2^T, \ldots, \vec{u}_{Q-1}^T]^T \tag{4.16}$$

$$\hat{u} \triangleq [\vec{u}_2^T, \vec{u}_3^T, \ldots, \vec{u}_Q^T]^T \tag{4.17}$$

Just like in (4.6), the equivalent condition of the mono-exponential structure constraint can be represented by

$$\|\hat{u} - \Lambda\check{u}\|^2 = 0 \tag{4.18}$$

That is, when $\vec{u}$ satisfies (4.18), $\vec{u}$ is an exponential function. Although (4.15) seems like a linear function, it is not a purely linear function since the matrix Λ is also a function of $\vec{u}$. However, this formulation may provide convenience in related analysis and numerical computations. In particular, we have developed a new approach in estimating $\vec{z}$ based on the definition of λ . In the following, we focus on this new approach and compare it to the other two commonly used approaches. At the end, we mention the application of the new formulation to the auxiliary variable method.

4.2 A new estimator of $\vec{z}$

The decay and off-resonance frequency maps can be derived from the coefficient $\vec{\lambda}$. Based on the definition of $\vec{\lambda}$ in (4.14), the nth element is given by

$$\lambda_n = \sum_{q=1}^{Q-1}(u_{n,q}^* u_{n,q+1}) / \sum_{q=1}^{Q-1}(|u_{n,q}|^2) = \sum_{q=1}^{Q-1}(m_n^* e^{-z_n^* t_q}) m_n e^{-z_n t_{q+1}} / \sum_{q=1}^{Q-1} m_n^2 e^{-Re\{z_n\}2t_q}) = e^{-z_n \Delta t} \tag{4.19}$$

where $\Delta t \triangleq t_{q+1} - t_q$ for any $q = 1, 2, \ldots, Q-1$. From (4.19), we see that $e^{-\vec{z}\Delta t}$ is equal to $\vec{\lambda}$ when $\vec{u}$ is an array of exponential functions in the time domain. The $\vec{z}$ map can then be derived by $(-\log\vec{\lambda})/\Delta t$. If noise is present in $\vec{u}$, $\vec{\lambda}$ would be biased from $e^{-\vec{z}\Delta t}$ due to the

noise. However, since $\vec{\lambda}$ minimizes

$$\|\hat{u} - \Lambda\breve{u}\|^2 = \sum_{q=1}^{Q-1} \|\vec{u}_{q+1} - \vec{\lambda} \odot \vec{u}_q\|^2, \tag{4.20}$$

$\vec{\lambda}$ should be a reasonable estimate of $e^{-z_n \Delta t}$ under mild noise. In this work, we call this estimator the geometric progression estimator since it is rooted in the fact that an exponential function in its discrete form is a geometric progression.

4.3 Three estimators

Two estimators have been used for measuring the $\vec{z}$ map given reconstructed image sequence from multi-echo data. The most commonly used estimator is the least squares fit estimator. The estimator minimizes

$$\|\vec{u} - \vec{\rho}(\vec{m}, \vec{z})\|^2 \tag{4.21}$$

with respect to $\vec{m}$ and $\vec{z}$, where $\vec{\rho}(\cdot)$ defined in (3.3) represents the mono-exponential model of the signal. We call this estimator the least squares fit (LSF). This estimator is also the maximum likelihood estimator, since noise in the k-space is usually white Gaussian. Despite this advantage, the estimator is essentially solving a nonlinear optimization problem and therefore calls for nonlinear solvers such as Gauss-Newton and Levenberg-Marquart algorithms. These algorithms can be hard to implement and can stick in a local minimum [18]. The performance of this estimator thus depends on SNR and data length, making the algorithm performance a bit unpredictable. Masking is usually required to mask out background voxels since the low spin density associated with them causes an ill-conditioned system for decay and off-resonance frequency. A mask is nontrivial to build and may require other pulse sequences to find the foreground area. If masking is absent, the algorithm may take a long time to stop since the iterate for background voxels would keep searching for the

"solution" although such searching is meaningless. This problem can be relieved by some adaptive stopping rules, such as stopping searching for low spin density voxels. However, these rules can cause wrong identification of background voxels or too-early stopping of iterations for some voxels.

Another well known estimator for mono-exponential modeling is to first take a logarithm over the reconstructed image sequence. Since the image sequences are close to a mono-exponential function array, the transformed data should be close to a line in the logarithm domain. Linear curve fitting is then applied to the transformed data to estimate the spin density, the R_2^* relaxation, and the off-resonance frequency maps. We call the estimator the log estimator (LOG). The criterion to minimize the LOG is

$$\sum_q \| \log \vec{u}_q - \log \vec{m} + \vec{z} t_q \|^2 \tag{4.22}$$

where t_q represents the time for the qth k-space frame. Since the relationship between $\log \vec{u}_q$ and $\vec{m}$, $\vec{z}$ is linear, the function (4.22) can be written in matrix form

$$\| \vec{u}_{log} - [\mathbb{1} \ \mathbb{T}][\vec{m}_{log} \ \vec{z}]^T \|^2 \tag{4.23}$$

where $\mathbb{1} \triangleq [\underbrace{1, 1, \ldots 1}_{Q}]^T$, $\mathbb{T} \triangleq [-t_1, -t_2, \ldots, -t_Q]^T$, $\vec{u}_{log}$ and $\vec{m}_{log}$ are counterparts of $\vec{u}$ and $\vec{m}$ in the log domain, respectively. Since the coefficient matrix in (4.23) is $Q \times 2$, its associated normal matrix is 2×2 and therefore can be inverted explicitly. Therefore, a closed-form solution exists for minimizing (4.23). Although the estimator is computationally inexpensive, it has several shortcomings as well. First, it is a biased estimator, and the bias is larger in decay and off-resonance frequency estimation [110]. The bias is dependent on the value of the spin density, R_2^* decay and off-resonance, and an explicit relationship is not clear yet. The bias makes the estimates somewhat unreliable. Although [110] proposed a compensation method by minimizing a data-weighted least squares criterion, the method introduces data

into the numerator of the estimator and causes very large variance in our experiments. The second shortcoming of the estimator is that an extra phase unwrapping must be used after taking the log over the data. The unwrapping may introduce an extra error source into the estimation. Third, our experiments show that the LOG estimator has a larger variance than the LSF estimator. The advantage of LOG, compared to LSF, is its speed advantage since the solution is in closed form.

The third estimator for the $\vec{z}$ map is the geometric progression (GEO) estimator. GEO has many advantages compared to the other two estimators. The main advantage of GEO is its straightforward computation and efficient implementation. The GEO criterion is defined by minimizing the function (4.20) with respect to $\vec{\lambda}$. If only the relaxation and off-resonance maps are needed, the estimate can be obtained by evaluating the autocorrelation function of each exponential function at 0 and 1 and then dividing the two numbers to estimate $e^{z_n \Delta t}$, as defined in (4.19). If the spin density is also needed, the map can be calculated by minimizing the least squares function (4.21). However, since $\vec{z}$ has already been estimated, (4.21) is a quadratic function with respect to $\vec{m}$ and explicit solution exists. For the nth voxel, the estimate $\hat{m}_n$ is

$$\hat{m}_n = \frac{\sum_{q=1}^{Q} e^{-\hat{z}_n^* \Delta t q} u_{n,q}}{\sum_{q=1}^{Q} e^{-2Re\{\hat{z}_n\} \Delta t q}} \tag{4.24}$$

where $\hat{z}_n$ represents the estimate of $\vec{z}$ in the nth voxel. Notice $e^{-\hat{z}_n \Delta t}$ can be replaced by λ_n, and the former expression is not explicitly calculated in the estimation. This estimation process is dramatically faster than that of LSF. Since LOG involves phase unwrapping, GEO may also be slightly faster than LOG.

4.4 Modified geometric progression estimators

When one solves the least squares fit in the LSF and LOG estimation processes, one needs to invert a normal matrix since the system is usually overdetermined. The inverse can be implemented as a division, and the denominators of these two estimators do not

include data. This is rather important to the stability of an estimator. The noise contained in the data can induce problems such as bias and large variance. For GEO, the formula is shown in (4.14). The denominator is a summation of squared data along the time axis. We demonstrate that this is a biased estimator. The demonstration starts from calculating the expected values of the denominator and numerator in (4.14). Let $\vec{\varepsilon}$ denote the white Gaussian noise. For the numerator,

$$\begin{aligned}
&\mathbf{E}\{\sum_{q=1}^{Q-1}((u_{n,q}+\varepsilon_{n,q})^*(u_{n,q+1}+\varepsilon_{n,q+1}))\} \\
=&\sum_{q=1}^{Q-1}\mathbf{E}\{u_{n,q}^*u_{n,q+1}+u_{n,q}^*\varepsilon_{n,q+1}+\varepsilon_{n,q}^*u_{n,q+1}+\varepsilon_{n,q}^*\varepsilon_{n,q+1}\} \\
=&\sum_{q=1}^{Q-1}u_{n,q}^*u_{n,q+1},
\end{aligned} \tag{4.25}$$

so the numerator has an unbiased expected value. However, for the denominator,

$$\begin{aligned}
&\mathbf{E}\{\sum_{q=1}^{Q-1}(|u_{n,q}+\varepsilon_{n,q}|^2)\} \\
=&\sum_{q=1}^{Q-1}\mathbf{E}\{|u_{n,q}|^2+u_{n,q}^*\varepsilon_{n,q}+\varepsilon_{n,q}^*u_{n,q}+|\varepsilon_{n,q}|^2\} \\
=&\sum_{q=1}^{Q-1}(|u_{n,q}|^2+\sigma^2) \\
=&\sum_{q=1}^{Q-1}|u_{n,q}|^2+(Q-1)\sigma^2
\end{aligned} \tag{4.26}$$

where σ is the noise standard deviation. Therefore, the expected value of the denominator is biased by adding a positive term $(Q-1)\sigma^2$. The expected value of the estimation is therefore biased toward a smaller value, causing the estimated decay map to be larger than the actual value. This behavior is confirmed by our numerical evaluation. Although there is a bias in decay estimation, the off-resonance frequency is not influenced by the bias. This is because the denominator is a real number, thereby only influencing the modulus of the estimator.

The phase of the estimate completely depends on the numerator and is unbiased. Notice that a completely correct calculation of the expected value of $\vec{\lambda}$ should have the expectation outside the fraction; however, this expected value is very hard to derive. The method we use in this section is only an approximation, and the quality of the approximation is actually not guaranteed. However, this approximation can still give a broad idea about the estimation bias. More importantly, the approximated calculation matches the numerical evaluation very well.

Since the denominator is biased by a positive term, an immediate solution is to subtract the same positive term from the denominator. We therefore want to calculate a new vector $\vec{\lambda}'$ in whch each element λ'_n is defined as

$$\lambda'_n = \sum_{q=1}^{Q-1}(u^*_{n,q}u_{n,q+1}) / \sum_{q=1}^{Q-1}(|u_{n,q}|^2 - (Q-1)\sigma^2). \tag{4.27}$$

We call this new estimator the first modified geometric progression estimator (MODI). Although having a simple form, this estimator requires knowing the standard deviation of the noise. One of simplest way to measure the noise standard deviation may be imaging the object twice and then calculating the squared error between the two images. However, this approach requires additional scans. Other methods for measuring the noise standard deviation remain unclear. However, any method may probably need additional actions to find the standard deviation. Moreover, experiments show that this modified estimator still has a relatively large estimation variance compared to LSF and even LOG.

Another modified estimator, called the second modified geometric progression estimator (MODII), aims to reduce the bias in a different way. Instead of adding a negative term in the denominator, it adds the same positive term in the numerator to cancel the bias caused by the noise. MODII attempts to calculate $\vec{\lambda}''$ which minimizes the following criterion:

$$\|\Lambda''^{-1}\hat{u} - \Lambda''\check{u}\|^2 = \sum_{q=1}^{Q-1}\|\vec{\lambda}''^{-1} \odot \vec{u}_{q+1} - \vec{\lambda}'' \odot \vec{u}_q\|^2 \tag{4.28}$$

This leads to the solution

$$|\lambda''_n| = \sum_{q=2}^{Q} |u_{n,q}|^2 / \sum_{q=1}^{Q-1} |u_{n,q}|^2 \tag{4.29}$$

and

$$\angle\lambda''_n = \frac{1}{2} \sum_{q=1}^{Q-1} \angle u^*_{n,q} u_{n,q+1} \tag{4.30}$$

for its nth element. The proof that shows minimizing (4.28) leads to the MODII decay and off-resonance estimates is quite straightforward. This criterion basically splits the original Λ matrix to a multiplication of two identical matrices Λ'' and times $\hat{u} - \Lambda\check{u}$ with Λ''^{-1}. For noiseless $\vec{u}$, minimizing (4.28) and (4.20) lead to the same soluton. For noisy $\vec{u}$, minimizing (4.28) leads to smaller bias in R_2^* decay estimate than (4.20) because the numerator in (4.29) also has a positive bias term $(Q-1)\sigma^2$, which keeps the quotient very close to the unbiased estimate. However, (4.29) can only be used to estimate the decay, since both denominator and numerator are real. The estimation of off-resonance is as the same as the ordinary GEO estimator. Notice that the vector $\vec{\lambda}''$ is an estimator for $e^{-\frac{1}{2}\vec{z}\Delta t}$ instead of $e^{-\vec{z}\Delta t}$ in the original GEO estimator.

4.5 Performance evaluation

In this section, we compare the five estimators, LSF, LOG, GEO, MODI, and MODII. We first compare their dynamic and static performance on estimating parameters from a single exponential function. We then extend the experiment to a simulated phantom.

4.5.1 Single exponential function

There are many criteria in evaluating the performance of an estimator. In this section, we mainly focus on the response to different SNRs, response to different data lengths, distribution of the estimated parameters, and the CPU time. A single exponential signal is simulated with spin density $m = 1 + \iota$, and complex frequency $z = 30 - \iota 100$. Total acquisition time is 30 milliseconds for all experiments. White Gaussian noise is added to the signal.

SNR is defined as the ratio of $|m|^2$ to noise variance in decibel (dB) scale. Mathematically, the SNR is defined

$$\text{SNR} = 10 \log_{10} \frac{|m|^2}{\sigma^2} \tag{4.31}$$

All experiments were implemented using MATLAB and run with an Intel Core i7-4700MQ CPU.

Figures (4.1) and (4.2) show the dynamics of each estimator with decreasing SNRs. SNR decreases from 40 dB to 10 dB in both figures. In Figure (4.1), 8 total samples of the exponential signal are acquired and estimations are made based on the 8 samples. From the top row, we can see that the GEO estimates of decay are biased toward larger values when SNR is low. The variances of the other four estimators are very close; however, MODI has slightly larger variance than the other three estimators. The other three estimators are also shown below to make their samples clearer. When number of signal samples is low, the three estimators have similar performance. Figure (4.2) shows the same comparison when total number of signal samples becomes 64. From the first row, the bias of GEO and larger variance of MODI in decay estimates are much clearer. The differences in the other three estimators are shown in the second row. It can be seen that MODII > LOG > LSF in their variances, and LOG and LSF have about the same performance. However, 64 is usually too large for number of echoes in a multi-echo sequence. Usually, the number of echoes is around 8, when MODII, LOG, and LSF have about the same performance.

Figure 4.3 shows the distribution of estimates for the five estimators when the number of signal samples increases from 4 to 64. While the top two plots show the performance of the five estimators, the bottom plots only show MODII, LSF, and LOG for a clearer comparison. From the top left plot, we see GEO's bias in decay estimation increases about linearly with number of samples. The increasing tendency is well predicted by the analysis in (4.25) and (4.26). The plot also shows that MODI has a larger variance than MODII, LSF, and LOG in decay estimation. The top right plot shows that when the number of signal samples is less than 10, all estimators have similar performance. With an increasing number of samples,

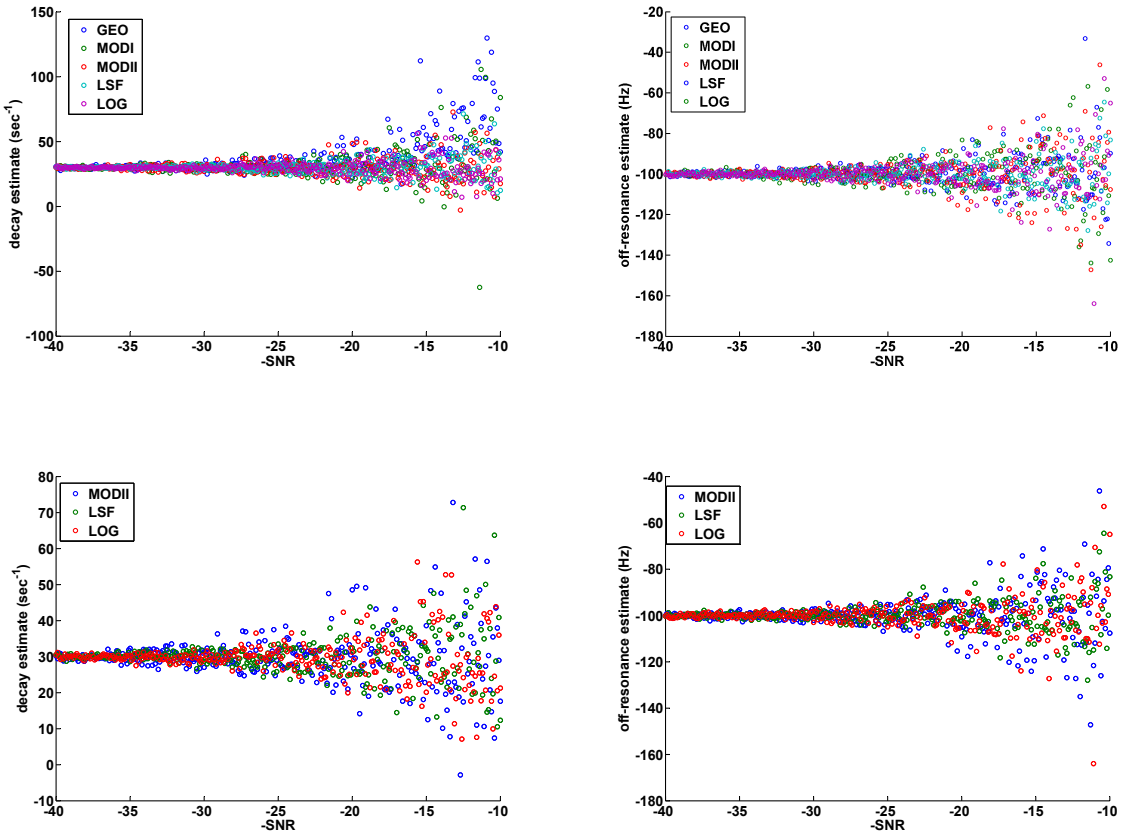


Figure 4.1: The left and right columns show the estimates of decay and off-resonance, respectively. The top two plots show the distribution of estimates from the five estimators, while the bottom two plots show the three estimators to make a clearer comparison among them. The number of data used in this figure is 8. Time used by the five estimators is 21.67, 21.14, 29.12, 2839, and 108.7 microseconds for one estimation.

the variance of LSF and LOG gradually decreases, while that of all geometric progression estimators does not change much. The comparison between MODII, LSF, and LOG in the bottom row shows this difference more clearly. It is easy to understand the behavior of LSF and LOG since they both use a least squares criterion, which basically averages out the noisy signal and reduces the variance. All geometric progression estimators do not use least squares, so the noise is not averaged out, causing the estimation variance to be unchanged with increasing data. This is clearly a shortcoming of all geometric progression estimators.

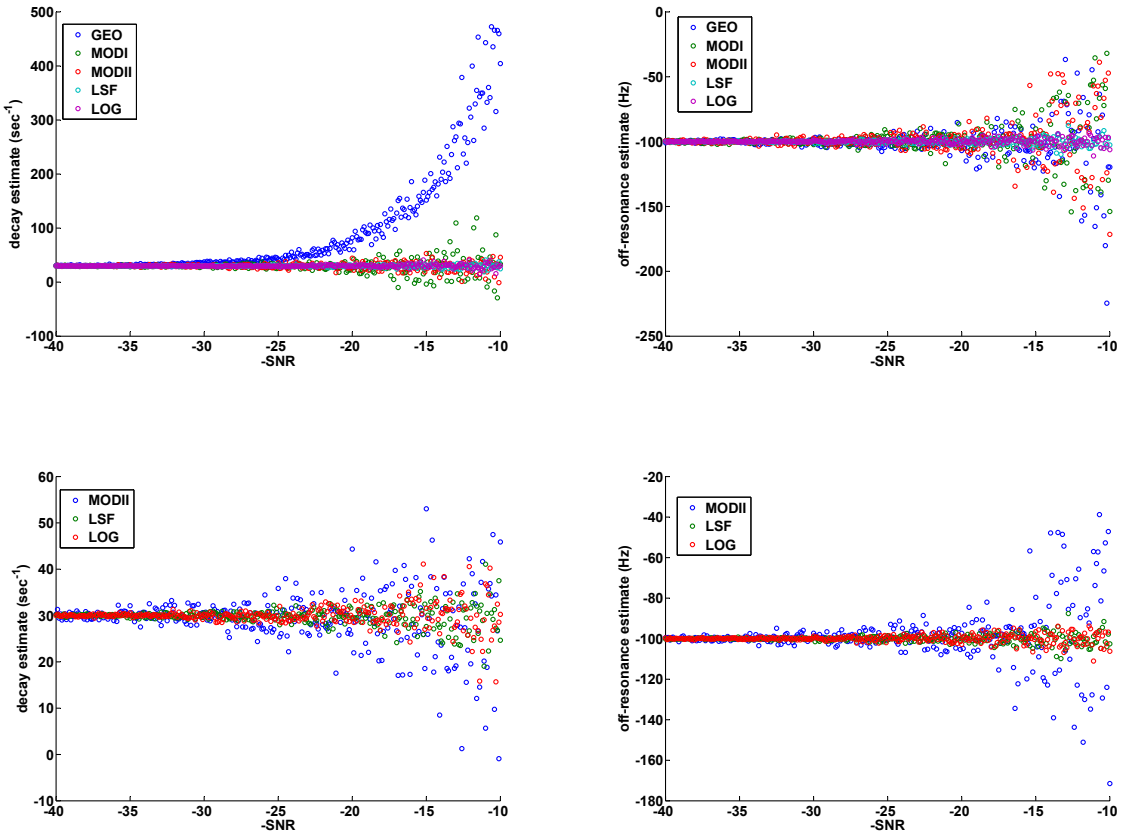


Figure 4.2: The left and right columns show the estimates of decay and off-resonance, respectively. The top two plots show the distribution of estimates from the five estimators, while the bottom two plots show the three estimators to make a clearer comparison among them. The number of data used in this figure is 64. Time used by the five estimators is 30.75, 30.60, 39.48, 4165, and 118.2 microseconds for one estimation.

However, since most multi-echo sequences sample up to 16 echoes, the difference between MODII, LSF, and LOG is small in practical images.

In the captions of the three figures4.1, 4.2, and 4.3, times used by the five estimators are reported. Geometric progression estimators are about 3-4 times faster than LOG and 80-100 times faster than LSF.

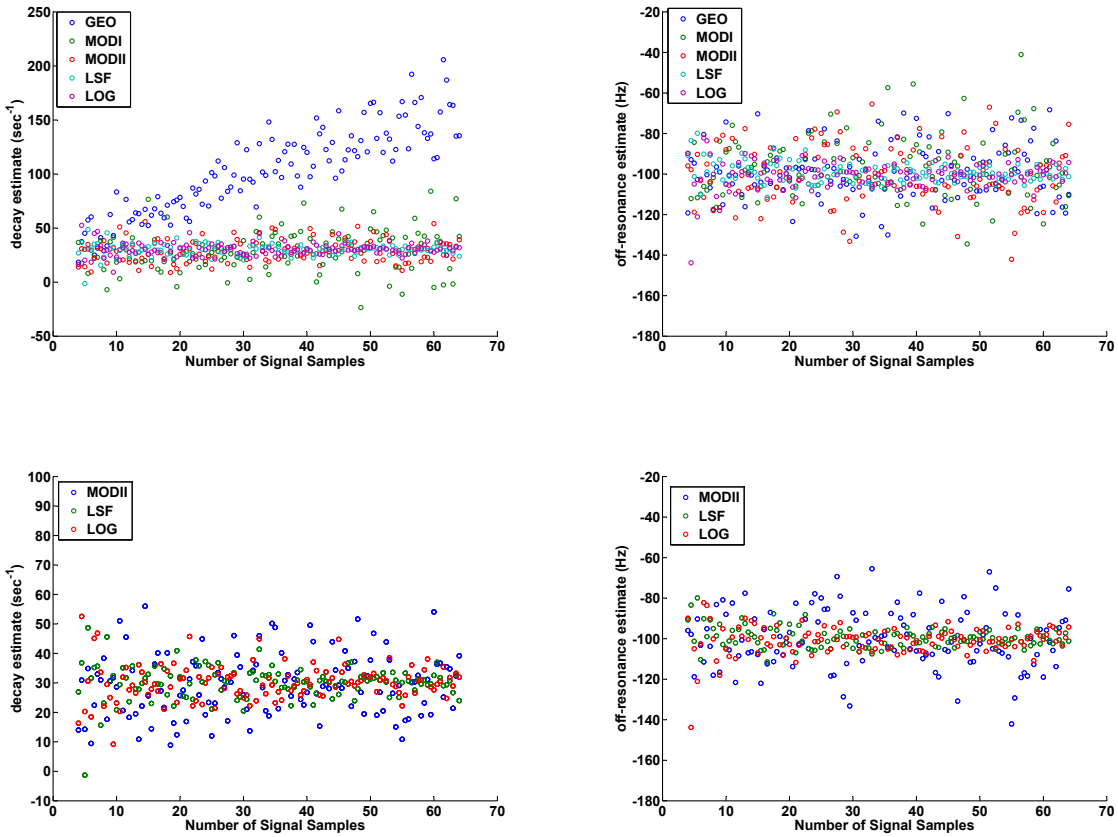


Figure 4.3: The left and right columns show the estimates of decay and off-resonance, respectively. The top two plots show the distribution of estimates from the five estimators, while the bottom two plots show the three estimators to make a clearer comparison among them. SNR = 15 for all estimators. Average time used by the five estimators is 46.78, 32.95, 41.49, 3756, and 167.3 microseconds for one estimation.

4.5.2 Simulated phantom

A phantom is simulated with 9 small cylinders, each with different parameter combinations, seated in a larger cylinder. The image size is 256×256. The spin density and R_2^* decay are designed to vary in orthogonal directions (cf. Figure 4.4). There are three spin density values, namely 0.8, 0.5, and 0.2, decreasing from top to bottom in the phantom. There are three R_2^* decay values, namely 10, 30, and 60 (sec^{-1}), increasing from left to right in the phantom. The off-resonance varies from -200 to 40 (Hz) in a right-left, top-bottom fashion. The phantom is designed to test how the reconstruction performance varies for different

combinations of parameters. For example, the small cylinder at the right bottom corner has the lowest spin density and highest R_2^* decay. The signal from voxels in this cylinder is very small and quickly vanishes, creating a much larger challenge for estimators.

Figure 4.4 compares the three geometric progression estimators, GEO, MODI, and MODII. GEO clearly has bias in its spin density and R_2^* decay estimates. MODI is better than GEO but has larger variance than MODII especially in the decay estimation. The normalized root mean square error (NRMSE) was also investigated. NRMSE is defined as

$$\text{NRMSE} = \frac{\|f - f_0\|_2}{\|f_0\|_2}$$

where f and f_0 represent the estimate and the ground truth. If we use a 3-tuple to represent the NRMSE of spin density, R_2^* decay, and off-resonance frequency maps, the NRMSE of GEO, MODI, and MODII are (0.20, 1.55, 0.20), (0.19, 1.07, 0.21), and (0.18, 0.42, 0.20), respectively. MODII is the best one among the three geometric progression estimators. GEO, MODI, and MODII took 53, 84, and 64 milliseconds to accomplish the estimation.

Figure 4.5 compares LSF, LOG, and MODII. Visually, LSF leads to the best estimate and LOG has about the same performance with MODII. This is especially apparent from the small cylinder at the right bottom corner. Since the spin density is very low and the R_2^* decay is very high in this cylinder, quality reconstruction is hard to obtain. The visual difference between LSF and the ground truth is smaller than the other estimators. The NRMSE of LSF and LOG is (0.12, 0.44, 0.11) and (0.10, 0.37, 0.20). The time they spent in estimation is 1.84 second and 0.88 second, respectively.

4.6 Other applications

The new formulation of the mono-exponential model in (4.18) can be used in more advanced applications such as undersampling-based reconstruction. Undersampling-based reconstruction refers to reconstruction of relaxation maps or the off-resonance map when

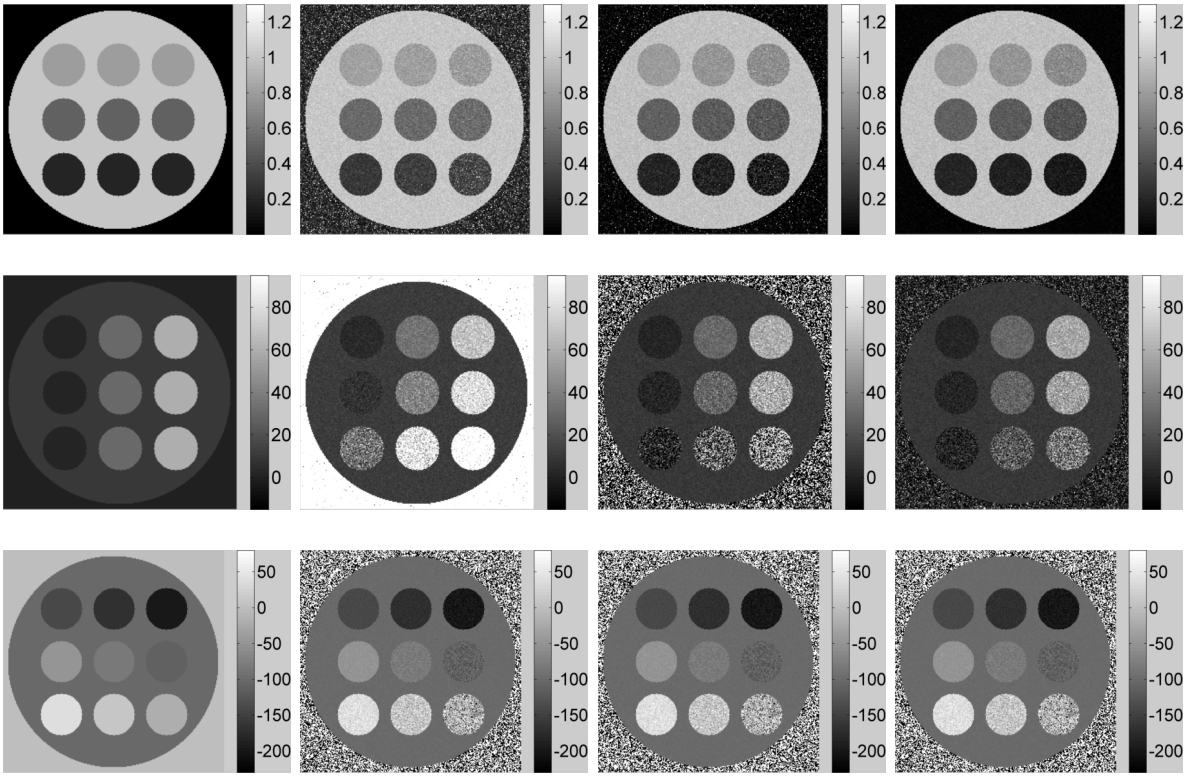


Figure 4.4: Reconstruction results of GEO, MODI, and MODII for the simulated phantom. The four columns from left to right represent ground truth, results from GEO, MODI, and MODII, respectively.

only a part of k-t space is sampled. Specifically, this includes reconstruction based on sparsely sampled multi-echo data or the single-shot/multi-shot sampling. If the k-t space is strongly undersampled, one cannot reconstruct the spatial-temporal signal without use of the mono-exponential structure constraint. In this situation, $\hat{u} = \Lambda\check{u}$ can be used as the structure constraint to replace the traditional nonlinear constraint. The linear form of this expression creates efficiency when techniques such as variable splitting are used. For example, if we want to estimate $\vec{u}$ through minimizing

$$\|\mathbf{DF}\vec{u} - \vec{s}\|^2 + \alpha\|\hat{\mathbb{D}}\vec{u} - \Lambda\check{\mathbb{D}}\vec{u}\|^2, \tag{4.32}$$

we can minimize it through a two-step procedure. Here, $\hat{\mathbb{D}}$ and $\check{\mathbb{D}}$ are defined by $\hat{u} = \hat{\mathbb{D}}\vec{u}$, and $\check{u} = \check{\mathbb{D}}\vec{u}$. Let $\mathbb{B} = \hat{\mathbb{D}} - \Lambda\check{\mathbb{D}}$. Notice $\mathbb{B}$ is a bidiagonal matrix. In step 1, we assume $\mathbb{B}$ is

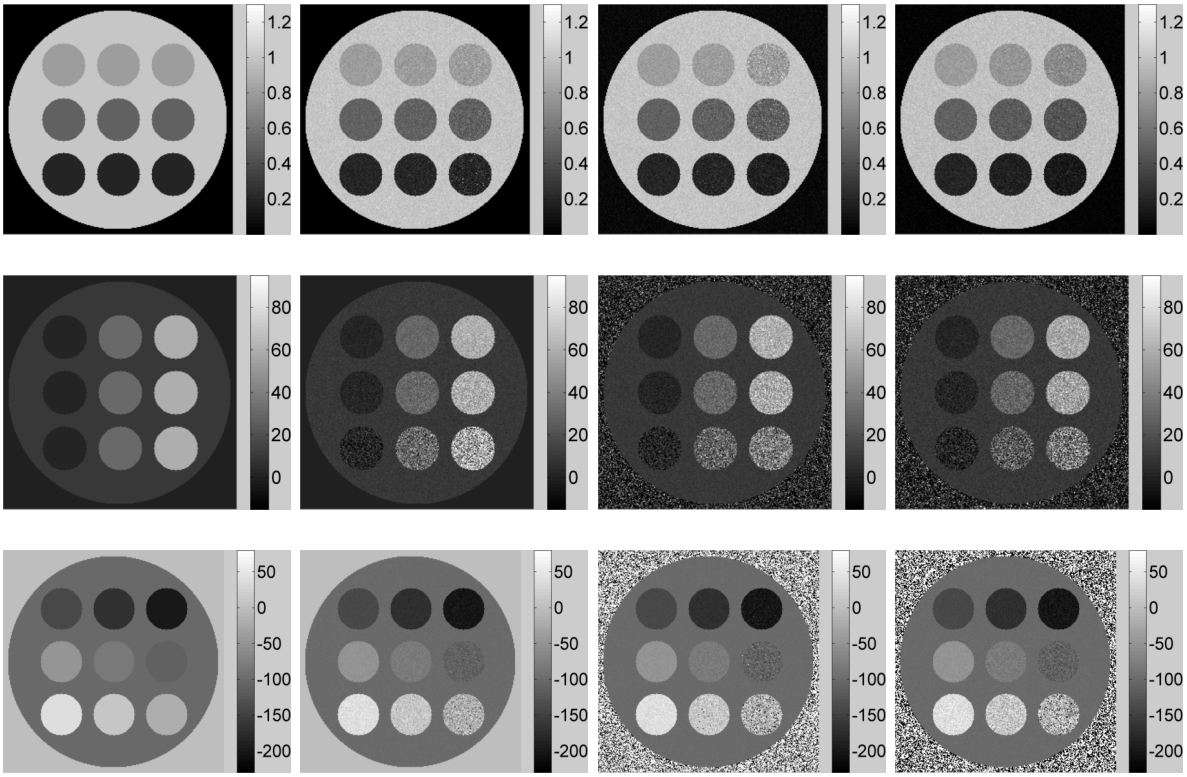


Figure 4.5: Reconstruction results of LSF, LOG, and MODII for the simulated phantom. The four columns from left to right represent ground truth, results from LSF, LOG, and MODII, respectively.

fixed, then

$$\vec{u} \longleftarrow (\mathbf{F}^H\mathbf{D}^H\mathbf{D}\mathbf{F} + \alpha\mathbb{B}^H\mathbb{B})^{-1}(\mathbf{F}^H\mathbf{D}^H\vec{s}) \tag{4.33}$$

In Step 2, we update $\vec{\lambda}$ based on $\vec{u}$ and then update $\mathbb{B}$. In the auxiliary variable method, step 2 is a nonlinear optimization problem and requires an iterative algorithm. Through use of the new formulation, solution of step 2 is based on the definition of $\vec{\lambda}$ and therefore can be calculated very rapidly. Notice that the second modified geometric progression estimator (MODII) can be introduced into the cost function (4.32) as well. The computation complexity of MODII is nearly the same as the ordinary geometric progression estimator. Thus, application of the new estimator to methods such as the auxiliary variable is very promising. This part will be studied in future.

4.7 Conclusions

In this chapter, we have developed a new formulation for the mono-exponential model of the spatial-temporal signal. The new formulation has a linear form and its solution can be calculated in explicit form, making the new formulation interesting to many relaxation and B_0 mapping applications. Based on the new formulation, we have presented a novel non-fit estimator for $\vec{z}$ and $\vec{m}$. The new estimator has a reasonable estimation accuracy but is much faster than traditional estimators. The new formulation as well as the new estimator can be applied to a wide variety of applications and may create a new avenue in approaching a fast, accurate estimation method for all parameter maps based on single-shot/multi-shot sampling.

Chapter 5

Assessment of single-shot trajectories

The joint reconstruction of spin density, R_2^* decay, and off-resonance frequency from a single-shot trajectory sampling is an ill-conditioned reconstruction problem. The cause of the ill-conditioning remains unclear. Since the reconstruction problem is nonlinear, the conditioning also depends on the value of the independent variables. If one let the algorithm used to do the reconstruction start from a good initial guess, then whether an accurate estimation can be achieved largely depends on the local condition number evaluated at the ground truth. Thus, a phantom and a human brain slice may lead to different local condition numbers. This dynamic condition number greatly increases the challenge in reducing ill-conditioning.

Our experience shows that the trajectory seems to play an important role in changing the conditioning globally. In other words, it seems that some trajectories may lead to a better conditioning regardless of the ground truth. The advantage in conditioning may vary in degree with the ground truth, but the advantage always exist. However, this instinct remains unproved. Moreover, a cohort study with a large amount of trajectories and objects (images) is still lacking. This chapter aims to make a contribution to this poorly explored area and provide some fundamental insight into the ill-conditioning of the problem. While major breakthroughs remain to be discovered, this chapter provides a cornerstone to start more affirmative and proof-based study in this topic.

The chapter adopts three tools to pry into the world of ill-conditioning, the condition number calculation, the singular value curve, and the reconstruction algorithm. The condition number can be calculated around the ground truth to see how ill-conditioned the local area around the ground truth can be. Varying the ground truth can lead to discoveries about

the dynamic behavior of the condition number and its relation to the underlying trajectory. The singular value curve consists of singular values sorted in a descending order. The shape of each singular value curve of each trajectory can be used to interpret the direct causes of ill-conditioning. Running reconstruction algorithm from a good initialization for every trajectory and every object (image) can finally confirm the discoveries made from the previous tools and can be used as direct evidence for advantages of some trajectories.

The chapter is organized as follows. In the first section, concepts such as condition number, singular value curve, and their relation to the reconstruction algorithm are briefly reviewed. Various objects and trajectories are described in detail in the second section. The third section shows all results under each object and each trajectory. Analysis about the results is also given. Finally, the last section discusses all findings, limitations, and future directions.

5.1 Background

Recall that the objective of the reconstruction is to estimate the spin density map $\vec{m}$ and the complex frequency map $\vec{z}$ based on the function

$$\begin{aligned} y_{lp} &= s_{lp}(m(\vec{r}), z(\vec{r})) + \varepsilon_{lp} \\ s_{lp}(m(\vec{r}), z(\vec{r})) &= \int_{\vec{r}} m(\vec{r}) e^{-z(\vec{r})t_l} e^{-2\pi\iota(\vec{k}_{lp}\cdot\vec{r})} d\vec{r} \end{aligned} \tag{5.1}$$

where y_{lp} is the acquired k-t space data and is known, s_{lp} is the predicted data, ε_{lp} is the noise, $\vec{k}_{lp}$ is the multi-shot trajectory, and $l = 1, 2, \ldots, L$, $p = 1, 2, \ldots, P$ are indices of time and shot. When a single-shot trajectory is used, then $P = 1$. Our objective is to study the conditioning of the mapping $s_{lp}(\cdot, \cdot)$ and its relation to the underlying trajectory $\vec{k}_{lp}$. Notice that the mapping is nonlinear; therefore, the conditioning varies with the independent variables $\vec{m}$ and $\vec{z}$. The dynamic conditioning greatly increases the research difficulty. There may be no way to find an analytic expression of the condition number or to calculate condition

number for every value of the independent variables. We therefore limit our attention to the ground truth for these variables, which are the objects being imaged. If we allow the reconstruction algorithm to start from a good initialization, which, for example, can be done by using the auxiliary variable method, then the ability of the algorithm to lead to an accurate solution largely depends on the local conditioning. Therefore, we can simply calculate the condition number at the ground truth for various objects—phantoms and *in vivo* images—which compose a subset of the vector space of the independent variables. The study can help us gain insight into the local convergence property of any algorithm, and can affirm us whether trajectory advantages actually exist.

5.1.1 Inverse problem and condition number

An inverse problem is defined by data, model, and an observation operator, or a forward operator. The operator maps the model to the data. An inverse problem is to estimate the model given the data. Mathematically, let d represent the data, m represent the model, and both of them are contained in some normed vector spaces. An inverse problem is to estimate m from d such that

$$d = f(m) \tag{5.2}$$

where f represents the forward operator. If we think of the estimation as a system whose inputs are the data and the forward operator and the output is the model, then the condition number is a criterion on how much error would be produced at the output given certain error in the inputs. For example, if we assume certain noise is present in the data, and the computing is perfect and void of any round-off error, then the relative condition number of f is defined by

$$\lim_{\epsilon \to 0^+} \sup_{\|\delta d\| < \epsilon} \frac{\|f^{-1}(d + \delta d) - f^{-1}(d)\|}{\|f^{-1}(d)\|} \Big/ \frac{\|\delta d\|}{\|d\|} \tag{5.3}$$

where f^{-1} represents the inverse function, d the unperturbed data, and δd the data perturbation or the error. Notice that f must be assumed to be well-posed in this case; otherwise,

the condition number may be not well defined. Also, notice that we assume the two normed vector spaces of the data and model have the same norm, which is a quite reasonable assumption in engineering applications. The definition basically measures the sensitivity of the estimate to a small perturbation of the data. When the condition number is large, a small data perturbation can cause a large variation in the estimate. Such a system is called ill-conditioned because it is hard to find an accurate estimate since error is always present. Ill-conditioning belongs to a problem (system) instead of an algorithm. This means changing an algorithm cannot change the condition number. For interested readers, further explanation about condition number can be found in [87].

In engineering, one often assumes d and m are contained in some Euclidean spaces such as $\mathbb{R}^n$ or $\mathbb{C}^n$, where n represents the dimension. Thus, when f is a linear operator, it can be represented as a matrix. Let the matrix be F. Equation (5.2) can be then be written as

$$\vec{d} = F\vec{m} \tag{5.4}$$

The condition number κ associated with a linear mapping can be explicitly formulated as

$$\kappa = \|F\|\|F^{-1}\| \tag{5.5}$$

where F^{-1} is the inverse of the matrix F. One can easily verify that $\kappa >= 1$. When κ is close to 1, we say the matrix is well-conditioned. When κ is much greater than 1, we say the matrix is ill-conditioned. A well-conditioned matrix usually means the algorithm used to solve (5.4) can be more accurate and faster than that for an ill-conditioned matrix, although the convergence speed also depends on some other factors such as the algorithm design. This impact makes condition number an important criterion and tool in numerical analysis and methods.

When the norm used in the definition (5.5) is the 2-norm, the definition is equivalent to another definition of the condition number κ

$$\kappa = \sigma_{max}(F)/\sigma_{min}(F) \tag{5.6}$$

where σ_{max} and σ_{min} represent the maximal and minimal singular values of the matrix F. Therefore, the condition number is a ratio of the maximal singular value to the minimal singular value. This is also the definition one usually uses to calculate the condition number, since the condition number of rectangular matrices can also be computed.

Calculating condition number for a large-size matrix (thousands of elements in each dimension) can be computationally very expensive. Our work usually deals with reconstruction of 64×64 sized images. This is a quite small resolution; however, the linearized forward operator has dimensions 8192×8192, which requires around 10 minutes to calculate the condition number. We therefore often use a low-resolution counterpart of the image to form a forward operator with much smaller size. In this work, we first downsample the two images $\vec{m}$ and $\vec{z}$ to 32×32 and then calculate the condition number of the associated forward operator. We found that the condition number based on the lower-resolution images is sufficiently accurate in quantifying the conditioning at the original resolution.

5.1.2 Singular value curve

The singular value decomposition (SVD) is a factorization of an arbitrarily shaped matrix. The matrix can be square or rectangular and the shape of the matrix does not influence calculation of the SVD. The SVD for a matrix A of size $m \times n$ is

$$A = U\Sigma V^H \tag{5.7}$$

where U and V are two unitary matrices of size $m \times m$ and $n \times n$, respectively, and the matrix Σ is an $m \times n$ diagonal matrix whose diagonal is always non-negative and sorted in a

descending order. The elements of the diagonal are called singular values. The curve formed by plotting the diagonal is called the singular value curve (SVC) in this work.

Recall that the division of the greatest singular value by the smallest singular value is the condition number. However, the value of the SVD goes beyond calculating a number. The SVC actually shows where the ill-conditioning originates. There are two primary sources of ill-conditioning, a large gap between adjacent singular values and a continuous reduction of singular values. If none of these two sources in the SVC is present, then the condition number must be very low and the conditioning of the matrix is good. Details about the two ill-conditioning sources can be found in the book [45]. The matrix with a clear gap between adjacent singular values is a rank-deficient matrix, which means that some of its columns or rows are very close to linear combinations formed by the other columns or rows. A simple example occurs when some columns of rows of the matrix are out of scale (small values compared to other columns and rows), which may actually be the case in our problem. The other source of ill-conditioning is a continuously decreasing series of singular values, a case called a discrete ill-posed problem by the author. This case happens when the matrix is a discrete version of an ill-posed problem.

Similar to calculating condition number, calculating the SVD is also computational expensive. The computation complexity is usually in the scale of $\mathcal{O}(mn^2)$ floating-point operations for an $m \times n$ matrix. We therefore use the same strategy the condition number calculation uses—reducing the resolution of the images and then calculating the SVD.

The performance of a reconstruction algorithm given a good initialization is highly dependent on the local conditioning of the forward operator. The performance includes accuracy and convergence speed; however, we mainly focus on accuracy in this chapter because we are concerned that the final maps be reasonably accurate regardless of the computational cost. Both accuracy and convergence speed not only depend on the conditioning but also the algorithm itself. Some algorithms may inherently lead to better accuracy and speed than

others. Therefore, the conditioning provides more of an upper bound than an exact prediction so that the accuracy and the speed cannot be better than the upper bound paired with a certain conditioning. Hence, investigating conditioning is important to an inverse problem since conditioning establishes a rough estimate for the worst performance of an algorithm.

In this work, we need to calculate condition number and run a reconstruction algorithm for some objects and some trajectories. Since the algorithm is fixed, the performance of the algorithm between different objects and trajectories almost completely depends on the conditioning, thereby relating the objects and trajectories to the conditioning. Running the algorithm also provides direct evidence for the influence of trajectories on the reconstruction of different objects.

5.2 Objects and trajectories

5.2.1 Objects

Here, objects refer to the objects scanned by the MRI scanner, including mainly phantoms and *in vivo* images. Since the condition number of the nonlinear forward operator in (5.1) varies with the objects, we need to investigate the conditioning for a number of objects to make any convincing conclusions. However, we have not acquired enough *in vivo* data and images, which greatly limits the robustness of our current conclusions. Therefore, this work is intended to be a preliminary study that formalizes the problem and research tools and applies the tools within a small group of objects.

Specifically, we have investigated the conditioning of the forward operator for a simulated phantom and a real brain slice. The simulated phantom is based on the four-cylinder phantom we have used in previous chapters. The four-cylinder phantom mimics four small cylinder containers placed in a large cylinder container. Each cylinder container contains material of independent spin density, R_2^*, and off-resonance values. The phantom is shown in Figure 2.3. To obtain a statistically more precise estimate of the condition number, we randomly vary the spin density, the R_2^* decay, and the off-resonance for the five cylinder

phantoms based on a uniform distribution. The range for the uniform distribution is set to be $[0, 1]$ for spin density, $[0, 80]$ (sec^{-1} for the R_2^* decay, and $[-200, 200]$ Hz for the off-resonance frequency. 10 realizations of the four-cylinder phantom were used to calculate the condition number. The mean of the 10 calculated condition numbers for each trajectory was used in the following analysis. The brain slice image was also used in Chapter 2 (Figure 2.8). Notice that all condition numbers and SVDs were obtained based on the lower resolution (32×32) version of these parameter maps.

5.2.2 k-space trajectories

Since single-shot trajectories are most time-efficient in the data acquisition, we mainly focus on single-shot trajectories, such as echo planar imaging (EPI), spiral, rosette, and radial. In particular, we have calculated condition numbers, SVDs, and run the reconstruction algorithm (for nonsingular matrices) for all these single-shot trajectories. We have also extended the condition number calculation to the multi-shot and multi-echo trajectories. The extension was quite easy to accomplish and can provide help in interpreting the results of the single-shot trajectories. We have not run any reconstruction algorithm for the multi-shot and multi-echo trajectories due to the technical complexity; however, this part of the extension can be considered in our future direction.

Let N represent the number of voxels in one dimension of the image. To reconstruct an image from its Fourier encoding, N^2 samples need to be acquired in k-space. However, if two images need to be reconstructed, the number of samples in k-space needs to be doubled as well. In joint reconstruction of spin density, R_2^* decay and off-resonance maps with the mono-exponential modeling, two complex images are to be reconstructed—the spin density map and the complex frequency map, whose real and imaginary part are the R_2^* decay and off-resonance maps. Hence, $2N^2$ samples are needed in k-t space. There are usually multiple ways to change a single-shot trajectory so that it can sample $2N^2$ points in the k-t space. All trajectories, including multi-shot and multi-echo trajectories, starts at 5 milliseconds and

finish at 25 milliseconds after the RF excitation. Exceptions are explicitly pointed out in the text.

Single-shot trajectories

A single-shot trajectory is the fastest k-t space acquisition fashion currently in use for joint reconstruction of spin density, R_2^* decay, and off-resonance frequency maps. However, due to the very limited data from a single-shot trajectory, the conditioning of the resulting reconstruction problem is usually very poor. Studying these trajectories is therefore very important.

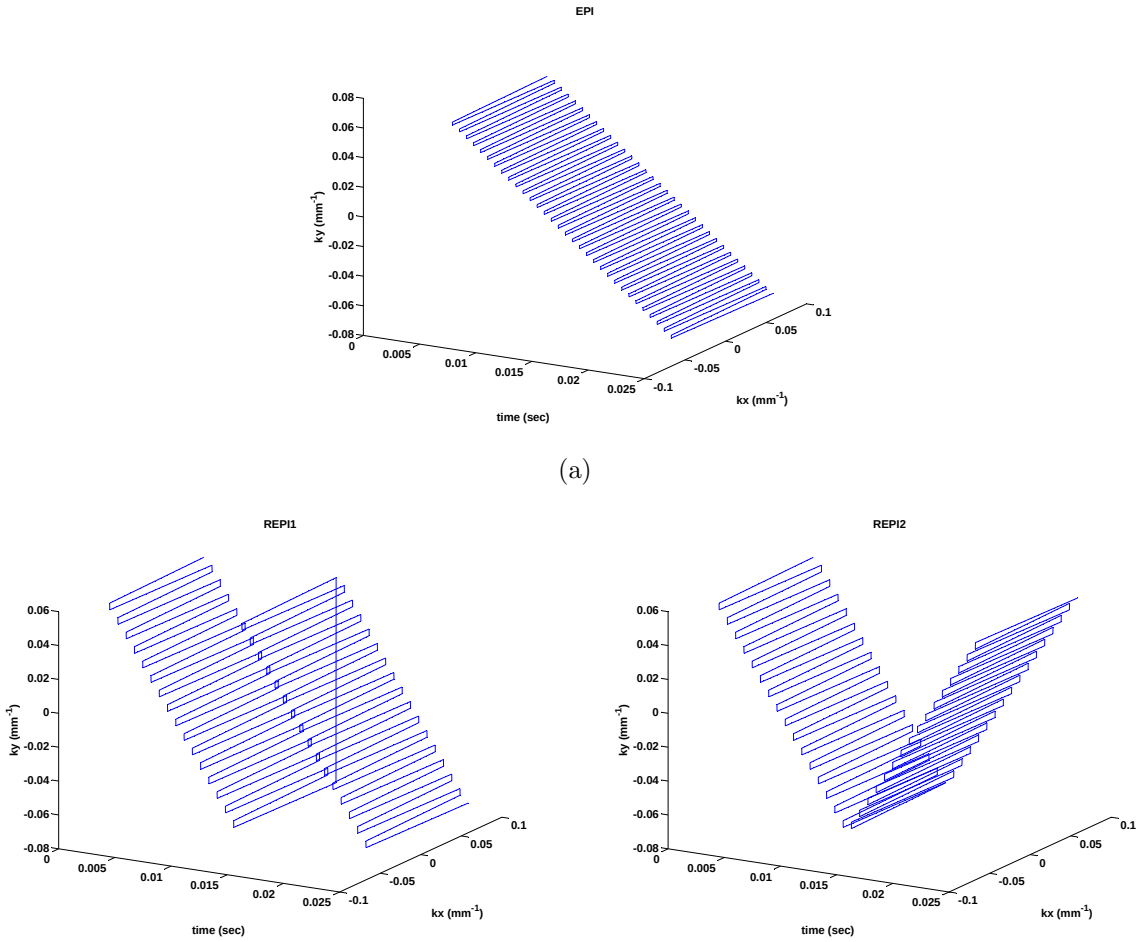
Echo planar imaging (EPI) The EPI trajectory is a zigzag rectilinear sampling trajectory in k-space. In regular imaging, the EPI trajectory only needs to sample the N^2 grids in k-space. Such an EPI trajectory is shown in Figure 1.14(a). In our problem, since $2N^2$ samples are needed, one needs to adjust the EPI sampling pattern to have double samples. Three variants of EPI were used in our experiments.

- A simple way is to sample one dimension of k-space in a double density, such as in the k_y direction. This trajectory does not repeatedly sample any grid in k-space. The abbreviation of the trajectory is simply EPI. Figure 5.1(a) shows the EPI trajectory.

- Another way is to repeatedly sample the same k-space. That is, after the first regular EPI is done, a second covering of k-space through exactly the same trajectory is applied. This repeated EPI trajectory is shortened as REPI1. Figure 5.1(b) shows the REPI1 trajectory.

- The third way also repeats a single EPI trajectory; however, the repetition is accomplished in a reverse direction. This trajectory is abbreviated as REPI2. Figure 5.1(c) shows the REPI2 trajectory.

Spiral The spiral trajectory is probably the most popular non-Cartesian trajectory used in MRI. The spiral trajectory has many forms. It can spiral in or spiral out, generating



(a)

(b)

(c)

Figure 5.1: The three EPI trajectories: (a)EPI, (b)REPI1, and (c)REPI2

the spiral-in trajectory and the spiral-out trajectory. A single-echo spiral trajectory needs a function to specify each sampling location in k-space. The function we used is a constant-speed spiral. In the presence of a square field of view (FOV), each sample of a spiral-out trajectory is represented by

$$\vec{k}(t) = [\frac{k_{max}}{2} w_{rad} t \cos(w_{rot} t) \quad \frac{k_{max}}{2} w_{rad} t \sin(w_{rot} t)] \tag{5.8}$$

where t represent sample time, k_{max} the k-space range, w_{rad} the radial speed, and W_{rot} the rotation speed. When only N^2 samples are needed, $w_{rad} = 1/T_{acq}$ where T_{acq} is the readout

time. When representing k-space as a complex domain, the trajectory can be represented by

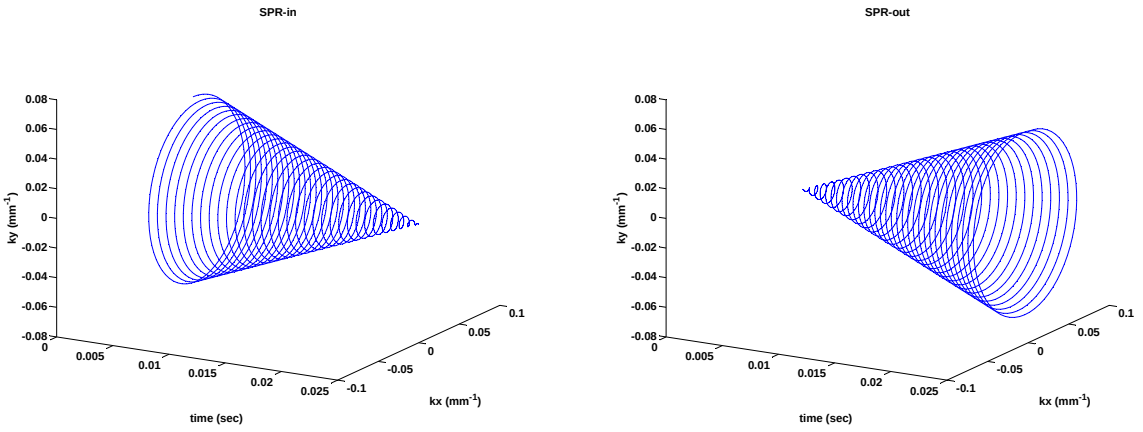
$$k(t) = \frac{k_{max}}{2} w_{rad} t e^{2\imath\pi w_{rot} t} \tag{5.9}$$

The rotation speed is chosen to finish $N/2$ circles of spirals so that the number of curves passing the k_y axis is N. The trajectory is shown in Figure 1.14(b). Notice that a spiral-in trajectory can use the same generating function with the argument t changed to $T_{acq} - t$. In this work, we used 6 variants of the spiral trajectory to meet the $2N^2$ requirement.

- Both the spiral-out and spiral-in trajectories must use an w_{rot} that is twice as high as that used for regular imaging. The corresponding spiral-out and spiral-in trajectories are abbreviated as SPR-out and SPR-in. Figures 5.2(a-b) show these two trajectories.

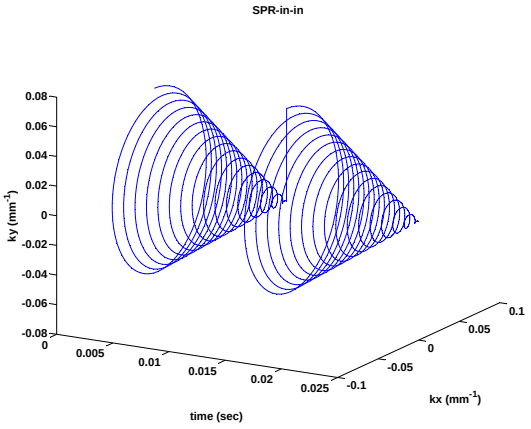
- Another method to double the number of samples is to repeat the same sampling twice, similar to EPI. However, in spiral, the repetition can be accomplished by using exactly the same trajectory or by concatenating spiral-out and spiral-in. These strategies lead to four different trajectories, the spiral-in-in (SPR-in-in), the spiral-in-out (SPR-in-out), the spiral-out-out (SPR-out-out), and the spiral-out-in (SOR-out-in). Figures 5.2c-f show these four trajectories.

Another trajectory similar to a spiral is the rectangular spiral (RectSPR). The rectangular spiral is identical to the spiral trajectory except that it always moves in a straight line. The trajectory is shown in Figure 5.3. In this work, we study the repeated rectangular spiral-out as a comparison to SPR-out-out to demonstrate the importance of sampling every k-space grid. In reality, this trajectory is very hard to implement since it calls for a very large gradient slew rate to accomplish the sharp orientation changes. The trajectory is therefore of more importance in theory than in practice.

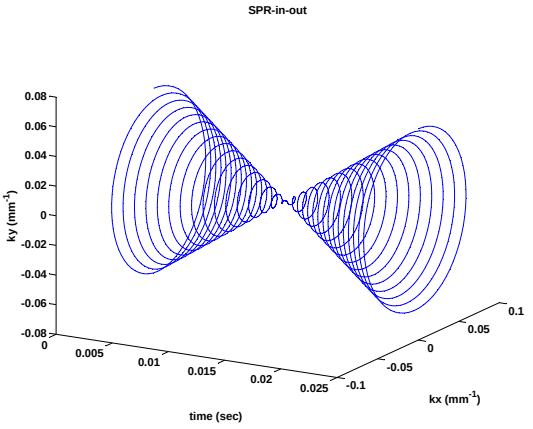


(a)

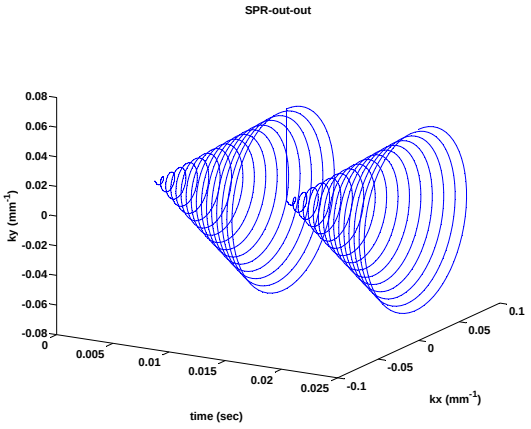
(b)



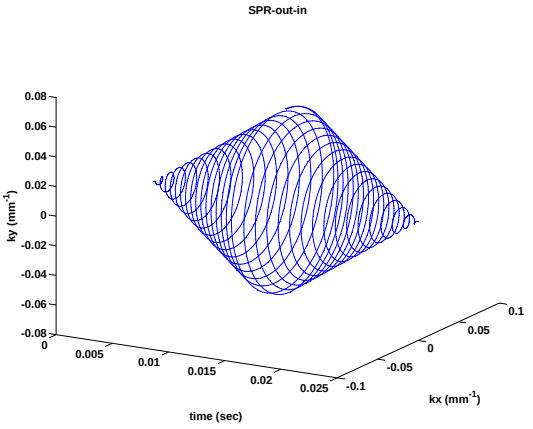
(c)



(d)



(e)



(f)

Figure 5.2: The six spiral trajectories: (a)SPR-in, (b)SPR-out, (c)SPR-in-in, (d)SPR-in-out, (e)SPR-out-out, (f)SPR-out-in

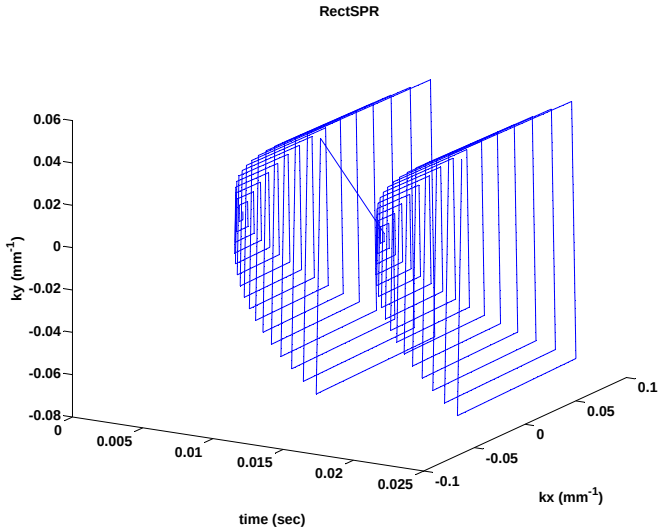


Figure 5.3: The repeated rectangular spiral-out trajectory

Rosette The rosette trajectory is frequently used by our research group for this problem. One of its main characteristics is that it crosses the k-space origin many times. Twieg proposed to use the rosette trajectory for the joint reconstruction problem [102]. Twieg and Reeves compared rosette to a repeated EPI in [103] and concluded rosette leads to smaller estimation variance than EPI. In this chapter, we compare rosette to a large group of single-shot trajectories. The generating function for the rosette is given below:

$$k(t) = \frac{k_{max}}{2} \sin(w_{osc} t) e^{\iota w_{rot} t} \tag{5.10}$$

where w_{osc} represents the oscillation frequency of the trajectory. We usually set $w_{osc} = 3196$ rad/sec, $w_{rot} = 1577$ rad/sec in our work since this combination results in a very uniform sampling of k-space. This rosette is shown in Figure 1.14(c) in 2D k-space. Two kinds of rosettes are used in our analysis.

- Since the trajectory periodically curves back to the low frequency area in k-space, there is no need to increase the oscillation or rotation frequency of the trajectory to increase the number of k-space samples. Instead, one can just sample more points using the

same generating function (5.10). The rosette trajectory is therefore much easier to implement and more friendly to the gradient coil. Such a rosette for sampling $2N^2$ samples is simply called ROS. Figure 5.4(a) shows the trajectory.

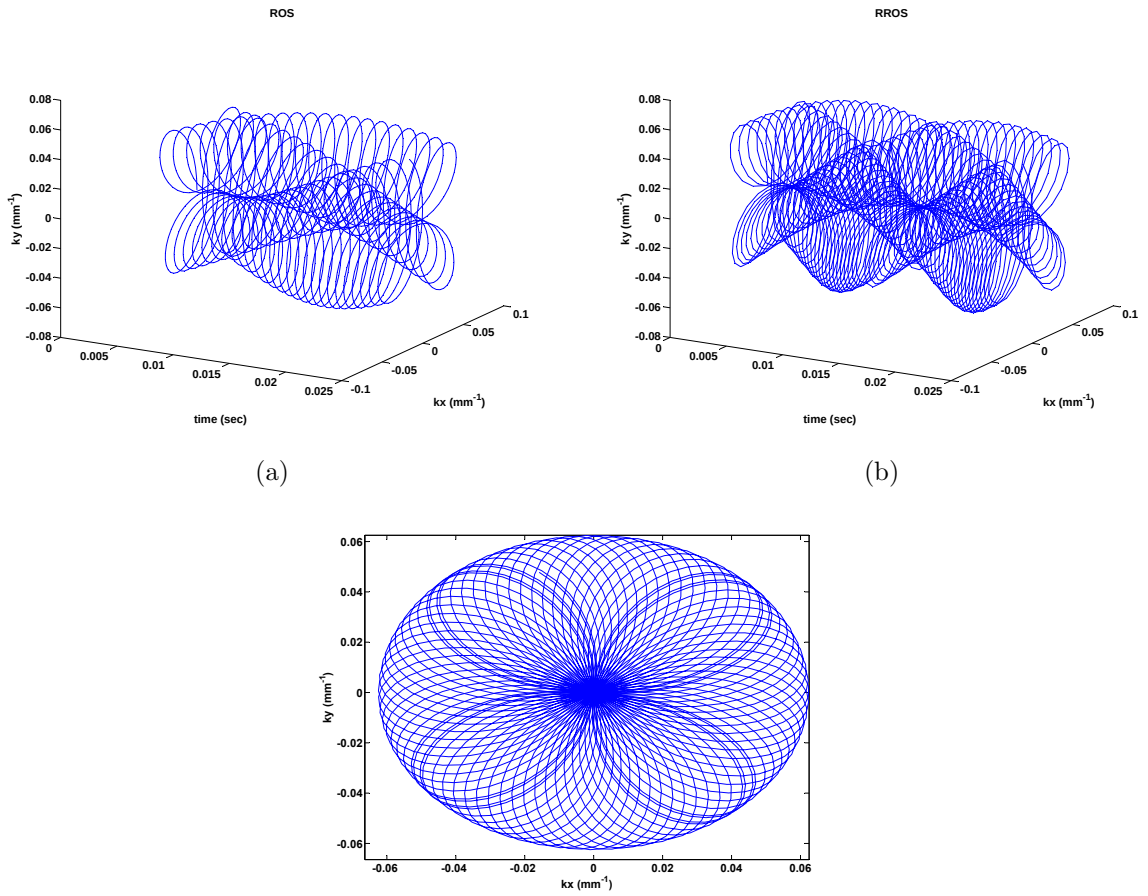
- Since all other trajectories use repetition as a strategy to meet the $2N^2$ sampling requirement, we also repeat an N^2-long segment of the rosette trajectory to facilitate a direct comparison between rosettes and other trajectories. Such a repeated rosette trajectory is abbreviated as RROS and is shown in Figure 5.4) (b). Since each N^2-long segment must uniformly sample the entire k-space, we doubled the w_{rot} and w_{osc}, reducing the practical value of RROS. The trajectory is therefore more important in the theoretical sense.

Radial The radial trajectory is also an important non-Cartesian trajectory. Although it samples off-grid points in k-space, the trajectory moves in a straight line, making the trajectory easy to implement and more resistant to eddy currents. To use a radial trajectory, one must specify the number of strokes and the number of samples in each stroke. The number of strokes is analogous to the number of lines in the line-by-line sampling, and the number of samples in each stroke is analogous to the number of samples in each line. We therefore set the number of strokes as the base resolution in the k_y direction and the number of samples in each stroke as the base resolution in the k_x direction. The resultant radial trajectory is shown in Figure 5.5. In this study, we used three variants of the radial trajectory to meet the $2N^2$ sampling requirement.

- Doubling the number of strokes doubles the number of samples. We call this radial trajectory RAD1. It is shown in Figure 5.5(a).

- Doubling the number of samples in each stroke doubles the number of samples. We call this radial trajectory RAD2. It is shown in Figure 5.5(b).

- Alternately, one can repeat the radial trajectory, abbreviated by RRAD and shown in Figure 5.5(c).



(a) (b)

(c)

Figure 5.4: The two rosette trajectories: (a)ROS and (b)RROS. (c) is a rosette trajectory in 2D k-space.

Trajectory feasibility

A feasible trajectory means the trajectory can be practically realized by the gradient coils equipping the MRI system. Typically, two factors restrict a theoretical trajectory, such as the above trajectories, from being implemented in an MRI system. First, the amplitude of the gradient must be lower than the maximal gradient magnitude. Since a k-space trajectory is an integral of the gradient over time, this restriction on gradient magnitude limits how fast the trajectory can move. Hence, for some trajectories, we may not be able to restrict the readout time to be less than 20 milliseconds due to the gradient magnitude limits. Second, the slew rate of the gradient must be lower than the maximal gradient slew rate. Many

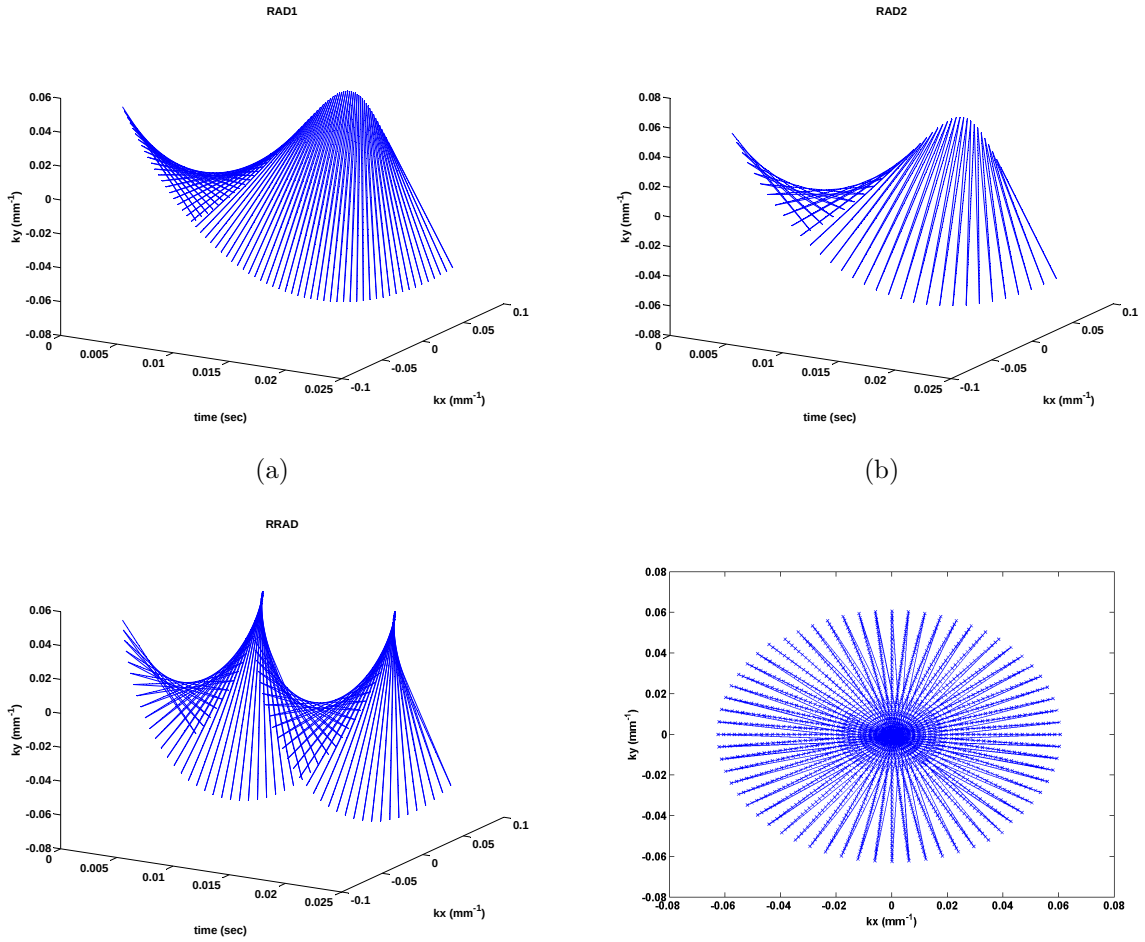


Figure 5.5: The three radial trajectories: (a)RAD1, (b)RAD2, and (c)RRAD. (d) is a radial trajectory in 2D k-space.

trajectories require a high slew rate, such as EPI and rosettes. The slew rate limits how fast a trajectory can turn. For EPI, this slew rate restriction causes the "dead time" for EPI when the trajectory reaches the end of each k-space line. For rosettes, the slew rate constraint, together with the gradient magnitude constraint, poses a limit on the shortest readout time the rosette can achieve. The trajectories introduced in this section have not been thoroughly investigated for feasibility in a real MRI system. They are mainly of theoretical importance in establishing a criterion for the quality of the signal encoding from each trajectory.

5.3 Experiments

We chose the ordinary trust region (OTR) algorithm we developed in the second chapter. We did not use the Auxiliary variable method since it demands OTR in the refinement process. Such a two-step procedure is good for poor initializations. In this work, however, our focus is on reconstruction accuracy. Therefore, starting from poor initializations is unnecessary. Instead, we choose a good initialization by adding a moderate amount of Gaussian noise to the ground truth of $\vec{m}$ and $\vec{z}$. The investigation mainly concerns whether convergence can be achieved in 200 iterations by having a sufficiently small gradient or residual, and if achieving convergence, how good the estimation bias is. Our experience shows that 200 iterations are sufficient for this algorithm. Usually, the algorithm stops before 200 iterations.

The estimation bias is measured by normalized root mean square error (NRMSE), which is defined by

$$\text{NRMSE} = \frac{\|x - x_0\|_2}{\|x_0\|_2}$$

where x and x_0 represent the estimate and the ground truth.

Masking is a common technique used in model-based reconstruction to rule out voxels of low spin density from the reconstruction target voxels [49, 83]. However, we point out that masking is actually a type of regularization. Since the field of view (FOV) is made smaller with masking, the spectrum is enforced to be somewhat smooth. Such enforcement is especially helpful for rosette, spiral and radial trajectories. Since they do not sample the most peripheral area of k-space, they should lose the information in this area. However, due to the masking, the spectrum including the peripheral area must be somewhat smooth, forcing the value at an unsampled k-space point to be close to its nearest sampled point. Therefore, masking may improve the conditioning associated with these trajectories, making the original ill-conditioning indiscernible. We therefore do not use any masking in this work.

5.3.1 Condition number

Table 5.1 shows the condition number evaluated at the phantom and brain slice under different trajectories. The evaluation at the phantom is quite consistent with the evaluation at the brain slice. Several observations are made in both condition number evaluations:

1. EPI, REPI1, REPI2, RectSPR all lead to better condition numbers than the other trajectories. Notice that there are two differences between the first four trajectories and the other trajectories. First, the four trajectories are all Cartesian trajectories. Second, the four trajectories sample every grid point of k-space. On the other hand, all the other trajectories, including rosette trajectories, are non-Cartesian trajectories and only sample within an inscribed circle of k-space. Therefore, the cause of the large ill-conditioning among the rest trajectories is likely to be related to at least one of the two factors, Cartesian sampling and full sampling. We think the latter factor is actually the reason for the large ill-conditioning.

2. Among all non-Cartesian trajectories, rosettes have the best conditioning. Specifically, the condition number of both ROS and RROS is lower than that of all non-Cartesian spiral trajectories and radial trajectories. This implies rosette trajectories are indeed advantageous over many other non-Cartesian trajectories.

3. Repetition of a trajectory usually leads to better conditioning than the equivalent non-repeating trajectory. Specifically, Table 5.1 shows that for every trajectory, including EPI, spiral, rosette, and radial, the repeated version always has a slightly smaller condition number than the non-repeated one. This fact is not very obvious from comparing condition numbers but is much more obvious from comparing SVCs and reconstruction accuracy.

The first finding above is quite surprising. Most researchers who have worked on this problem, including Twieg [102], have thought that rosettes would work better than EPI. However, this study suggests that EPI has the best conditioning. If EPI does have a better

conditioning, there seems to be no reason to use rosette instead of EPI. EPI is easier to program and generate and has a lower requirement on the hardware. EPI is also much more commonly used in reality than rosette. Therefore, it is quite surprising to us that EPI can be much less ill-conditioned than rosettes. The reason for the conditioning advantage in EPI is probably due to fact that EPI samples every grid point in k-space while rosettes only sample a part of k-space. In the Discussion sections, we give further analysis about why the loss of sampling can cause a serious conditioning problem. The second finding shows that rosettes are indeed better than other non-Cartesian trajectories, including spiral and radial, which have been used by other researchers in related problems such as T_2 mapping [7, 98]. The third finding demonstrates the advantage of repeating the trajectory design for this problem. Repeating is similar to multi-echo sampling. However, in the latter sampling, every sample of a single k-space frame is treated as if it is acquired at the same time point. For single-shot trajectories, the instant sampling of k-space is impossible; however, the experiment shows that even repeating the samples regardless of time still improves the conditioning of the reconstruction.

5.3.2 Singular value curve

Figure 5.6-5.7 show the singular value curves (SVCs) related to various trajectories and the two objects. Although the range of the singular values becomes larger for nearly every trajectory from the phantom to the brain slice, the shape of the SVC for each trajectory stays nearly the same. Two features are commonly present in all trajectories:

1. There is a gap at the middle of each singular value curve which reduces the singular value by about 10^3. This gap is especially sharp in EPI trajectories or RectSPR, but is smoother in non-Cartesian trajectories. What causes the gap is not completely clear. However, we found that the multi-echo version (see Appendix) of the four single-shot trajectories also have this gap (Figure 5.8). This means the gap probably originates

Table 5.1: Condition number of various trajectories

Phantom					
Trajectory	**Condition number**	**Trajectory**	**Condition number**	**Trajectory**	**Condition number**
EPI	4.74E07	REPI1	1.26E04	REPI2	6.51E05
RectSPR	**3.25E03**	SPR-in	5.49E21	SPR-out	5.05E20
SPR-in-in	2.95E20	SPR-in-out	7.38E20	SPR-out-out	7.74E20
SPR-out-in	9.17E20	ROS	3.48E13	RROS	1.38E13
RAD1	2.30E21	RAD2	1.64E21	RRAD	3.84E20
Brain slice					
Trajectory	**Condition number**	**Trajectory**	**Condition number**	**Trajectory**	**Condition number**
EPI	6.81E010	**REPI1**	**1.54E08**	REPI2	2.33E09
RectSPR	2.03E08	SPR-in	2.99E24	SPR-out	1.14E24
SPR-in-in	8.35E22	SPR-in-out	5.88E22	SPR-out-out	7.92E22
SPR-out-in	9.01E23	ROS	2.58E16	RROS	1.08E16
RAD1	1.49E24	RAD2	1.76E24	RRAD	1.73E22

from ill-conditioning out of the mono-exponential curve fitting which is an unavoidable problem for every approach in MRI parameter mapping.

2. At the end of every SVC, one can see the SVC usually continuously descends to very small singular values. This is another source of ill-conditioning for the forward operator in (5.1). For some trajectories, such as EPI trajectories, RectSPR, and rosettes, the knee point of the curve (the position to start descend) is late, i.e. close to the last singular value entry. The knee points of the other trajectories appear to be earlier.

This is very important to regularization. The former trajectories can be improved in their conditioning with much smaller regularization than the latter trajectories. Details for the theory about SVDs and regularization can be found in the book [45].

The second finding from studying the SVC is very important and may lead to further research on regularization. Our experience shows that regularization is probably necessary for any reconstruction algorithm applied to this problem. However, theory about what regularization to choose and what parameter to choose for the regularization is somewhat immature. An inappropriate regularization or an overly weighted regularization can cause serious estimation bias. With the help of the SVD and the SVC, we may be able to find the optimal regularization for each trajectory and find the optimal weight of the regularization to balance the noise amplification and the regularization-induced bias.

5.3.3 Reconstruction accuracy

Table 5.2 shows the estimation accuracy of the phantom and the brain slice for all single-shot trajectories. No noise was added to the k-t space data, and therefore the only error present in the input may be the round-off error. Since we wanted to examine the conditioning of forward operator in (5.1), we did not use any regularization. Adding regularization changes the conditioning, and then conditioning associated with the original problem would be indistinguishable. OTR was kept running until activating the stopping rule that the gradient becomes sufficiently small or the number of iterations exceeds 200. Our experience shows that most algorithms can stop within 200 iterations for the phantom. For the human brain slice, although some trajectories did not stop in 200 iterations, 200 has been large enough to demonstrate their conditioning property.

The accuracy distributions with respect to the trajectories are quite consistent between the phantom and the brain slice. REPI2 achieves the highest accuracy for the phantom, and RectSPR achieves the highest one for the brain slice. However, both of then, as well as REPI1, achieved very high accuracy for both objects. This result is consistent with

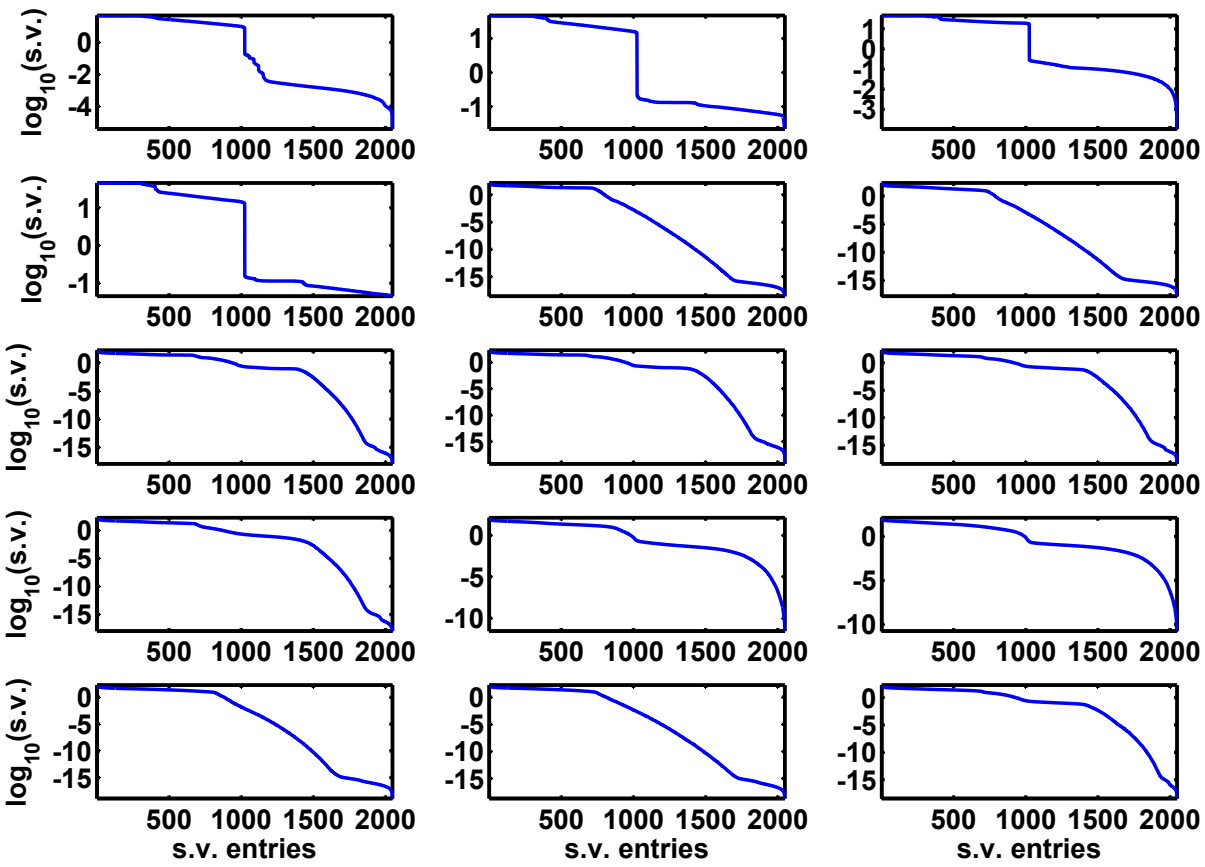


Figure 5.6: Singular value curves for all single-shot trajectories for the phantom. The trajectories from left to right are: (1st row) EPI, REPI1, REPI2; (2nd row) RectSPR, SPR-in, SPR-out; (3rd row) SPR-in-in, SPR-in-out, SPR-out-out; (4th row) SPR-out-in, ROS, RROS; and (5th row) RAD1, RAD2, RRAD.

the findings in condition number and SVC analysis. RROS (underlined) performs much better than ROS, but both of them perform better than the other non-Cartesian trajectories. All repeating involved trajectories, including REPI1, REPI2, SPR-in-in, SPR-in-out, SPR-out-out, SPR-out-in, RROS, and RRAD achieved higher accuracy than their non-repeating counterparts. Therefore, all findings obtained in the condition number analysis have been verified by running the reconstruction algorithm.

Figure 5.9-5.11 show the reconstructed maps from the 15 trajectories for the phantom, and Figure 5.12-5.14 show the reconstructed maps for the human brain slice. The advantage of REPI1, REPI2, and RectSPR in accuracy is clear by visual inspection. It is also apparent

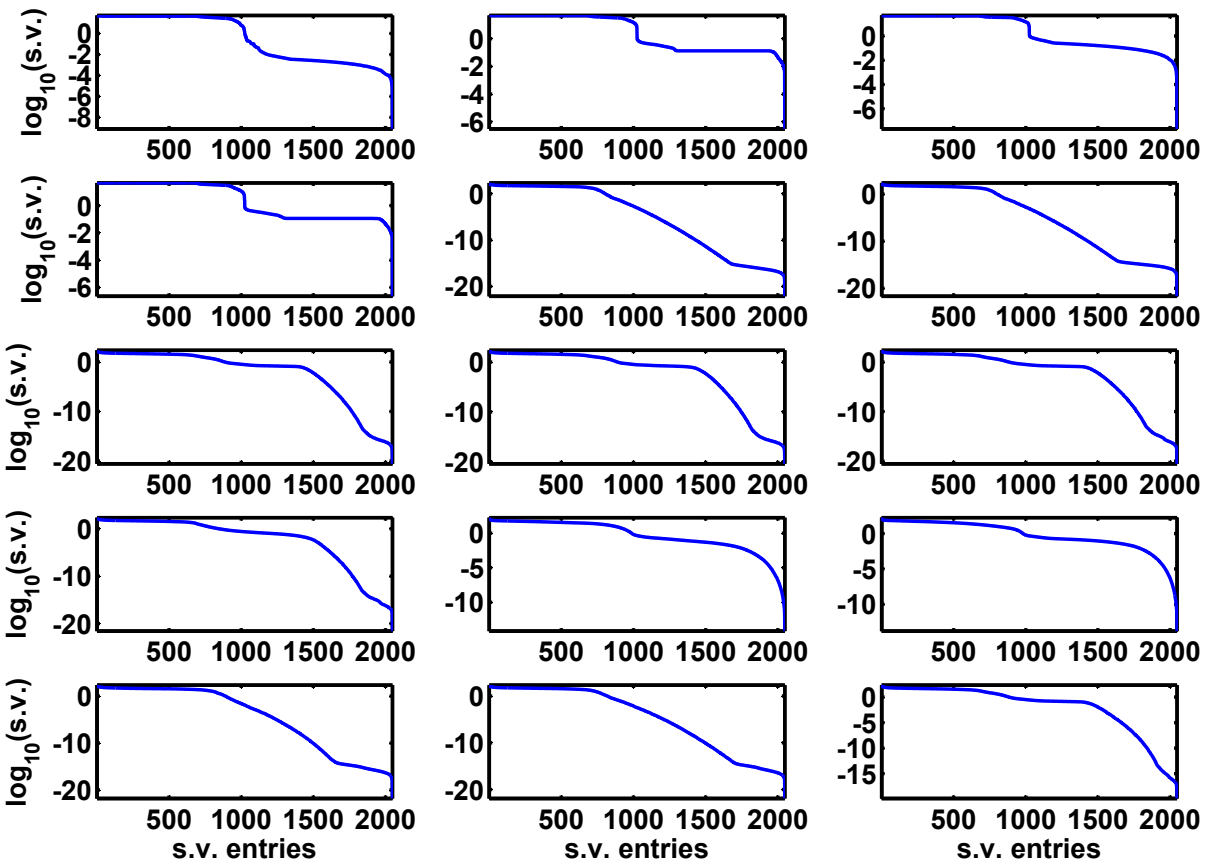


Figure 5.7: Singular value curves for all single-shot trajectories for the brain slice. The trajectories from left to right are: (1st row) EPI, REPI1, REPI2; (2nd row) RectSPR, SPR-in, SPR-out; (3rd row) SPR-in-in, SPR-in-out, SPR-out-out; (4th row) SPR-out-in, ROS, RROS; and (5th row) RAD1, RAD2, RRAD.

that RROS is better than other non-Cartesian trajectories in suppressing the noise. The advantage of repeating is also very obvious.

5.4 Discussion and conclusions

The results have manifested a high consistency between the condition number, the singular value curve, and the reconstruction accuracy from the OTR algorithm. The condition number provides a quantity to approximate the conditioning of the forward operator in (5.1). The singular value curve provides information about what causes the ill-conditioning and

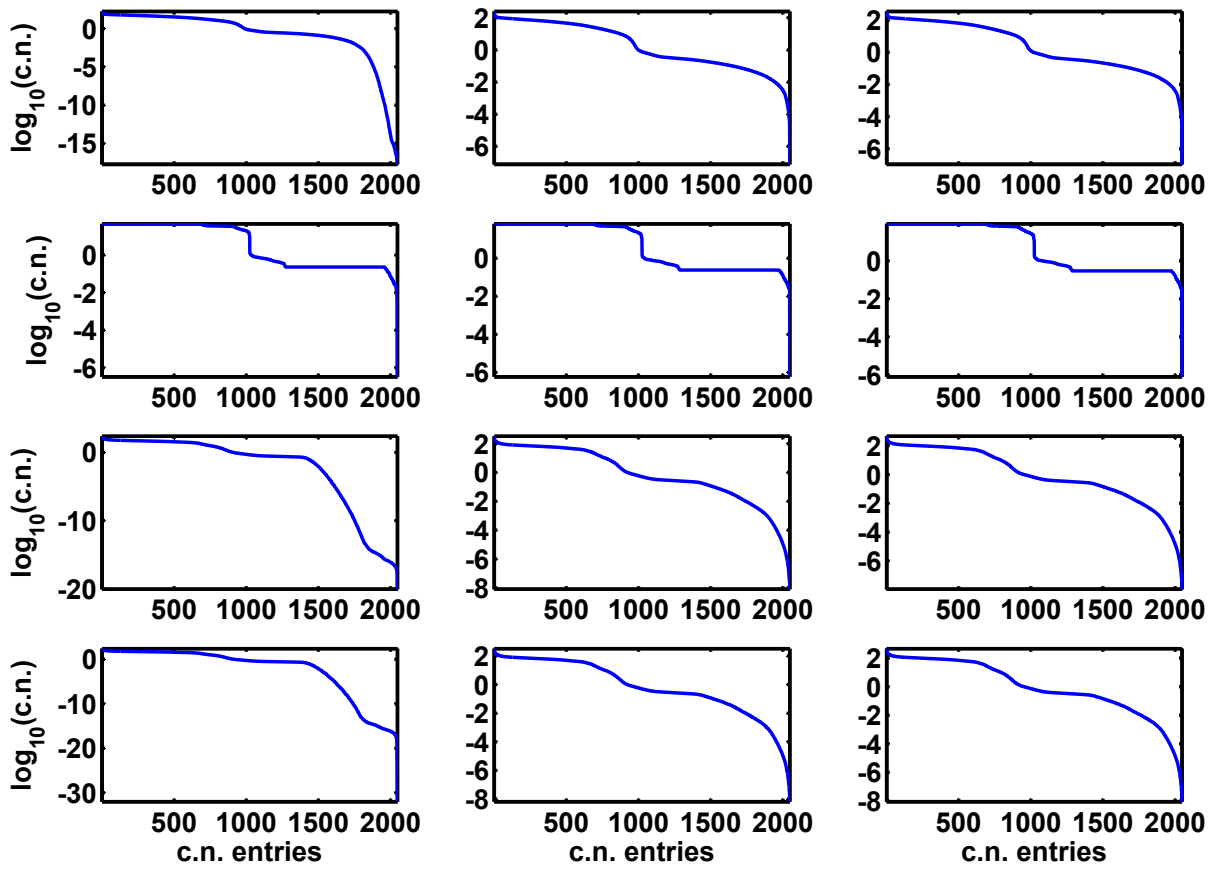


Figure 5.8: Singular value curves for the multi-echo version of all single-shot trajectories for the brain slice. The trajectories from left to right are: (1st row) 2-echo, 4-echo, and 8-echo rosette; (2nd row) 2-echo, 4-echo, and 8-echo EPI; (3rd row) 2-echo, 4-echo, and 8-echo spiral-out; and (4th row) 2-echo, 4-echo, and 8-echo radial. The explanations of the 16 multi-echo trajectories are shown in Appendix.

inspires usage of regularization and selection of the regularization weight. The reconstruction accuracy based on OTR verifies the previous analysis results and directly shows the influence of conditioning on the reconstruction quality.

The results show that EPI or rectangular spiral has a much better conditioning than other trajectories, such as rosette, spiral, and radial. We point out in the previous section that this may be caused by the fact that only the EPI and rectangular spiral trajectories sample every grid point in k-space. The other trajectories, however, only sample a circular area inscribed within the modeled k-space. We explain the reason behind the inference by

a simple example. Assume the complex frequency map $\vec{z}$ is a constant decay z over the entire image space and assume only two k-space frames are incompletely sampled. Since $\vec{z}$ is spatially constant, the second k-space frame equals $e^{z\Delta t}$ multiplied by the first frame, where Δt is the time interval between the two frames. Assume the sampling of the two frames skips one k-space grid point in both frames, similar to the way non-Cartesian trajectories skip sampling of the peripheral region of k-space. Since all other grid points are sampled, finding z is immediate. However, we can never obtain the spin density $\vec{m}$ map correctly because the two skipped samples of the grid point can have any value as long as their division result is $e^{z\Delta t}$. Therefore, the forward operator in this case is singular, which implies an infinite condition number. This simple example is shown by a two-echo EPI trajectory with one sample replaced by another sample, which is shown in Figure 5.15(a) and (b). The singular value curve is shown in Figure 5.15 (c). Twieg and Reeves also pointed out similar observations in [103].

There are several limitations regarding the analysis. First, the analysis is restricted to single-shot trajectories. However, due to practical factors such as eddy current, multi-shot trajectories may be more practical [49]. The analysis for multi-shot trajectories is therefore also desired. Second, regularization is not involved in the analysis. However, since all of our algorithms used regularization, involving regularization in the analysis is also important. Moreover, although EPI appears much less ill-conditioned than a rosette, whether the advantage can hold with certain regularization terms remains unclear. Third, a larger group of objects are highly desired since many of the analysis conclusions, such as EPI outperforming rosette, are very new and need thorough verification. Fourth, the analysis controls the readout time of all trajectories to be equal, which is not very possible in reality. In [103], the authors concluded rosette outperforms a repeated EPI trajectory. The readout time for the two trajectories are about 50 and 160 milliseconds, respectively. Hence, using different readout time for each trajectory may results in different conclusions.

This analysis may lead to finding an appropriate regularization for each trajectory such as REPI1, REPI2, and RROS. The regularization should very effectively increase the small singular values at the end of the singular value curve. However, the causes of the ill-conditioning remain unclear. Therefore, proposing a suitable regularization is nontrivial. Existing regularization methods, such as a finite difference penalty, also need further analysis concerning their regularizing effect and efficiency.

In conclusion, we have presented a preliminary study on the influence of different trajectories on the conditioning of the problem of jointly reconstructing spin density, R_2^* decay, and off-resonance frequency maps. The study uses condition number, singular values, and algorithm accuracy to examine the conditioning of the problem in the presence of different objects and trajectories. The analysis has found that undersampling k-space can cause severe ill-conditioning in this problem, and EPI therefore has a much better conditioning than rosettes. This conclusion is very different from our intuition that a rosette should outperform EPI since it samples the k-space origin multiple times, and therefore further verification is needed. The analysis has also found that a rosette has better conditioning than other non-Cartesian trajectories and repeated sampling of the same k-space point is important in improving the conditioning. Future application of the analysis includes proposing an optimal regularization for the reconstruction of the three parameter maps based on a single-shot acquisition.

Table 5.2: Estimation error of various trajectories.

Trajectories	**Phantom**			**Brain Slice**		
	Spin Density	**R2* Decay**	**Off-resonance**	**Spin Density**	**R2* Decay**	**Off-resonance**
EPI	0.9657	0.3691	0.1893	0.5740	0.1815	0.2035
REPI1	0.0823	0.0813	0.0265	0.0747	0.1040	0.0696
REPI2	**0.0152**	**0.0143**	**0.0045**	0.1141	0.1228	0.0772
RectSPR	0.3649	0.1487	0.0538	**0.0507**	**0.0428**	**0.0331**
SPR-in	0.8362	0.5151	0.2597	0.4847	0.3047	0.3070
SPR-out	0.9566	0.435	0.27	0.5599	0.2735	0.3484
SPR-in-in	0.8807	0.4276	0.1991	0.4007	0.2152	0.1868
SPR-in-out	0.9334	0.4265	0.2155	0.4232	0.2388	0.2032
SPR-out-out	1.0038	0.4921	0.2014	0.4629	0.2130	0.1856
SPR-out-in	0.9734	0.5216	0.2003	0.4598	0.2927	0.1978
ROS	0.9128	0.4805	0.179	0.3816	0.1938	0.1474
RROS	0.6727	0.3888	0.1285	0.2893	0.1208	0.0615
RAD1	1.0071	0.4847	0.2378	0.5507	0.2705	0.3002
RAD2	0.9651	0.4986	0.2392	0.5257	0.2763	0.3001
RRAD	0.929	0.4502	0.1887	0.4320	0.2261	0.1741

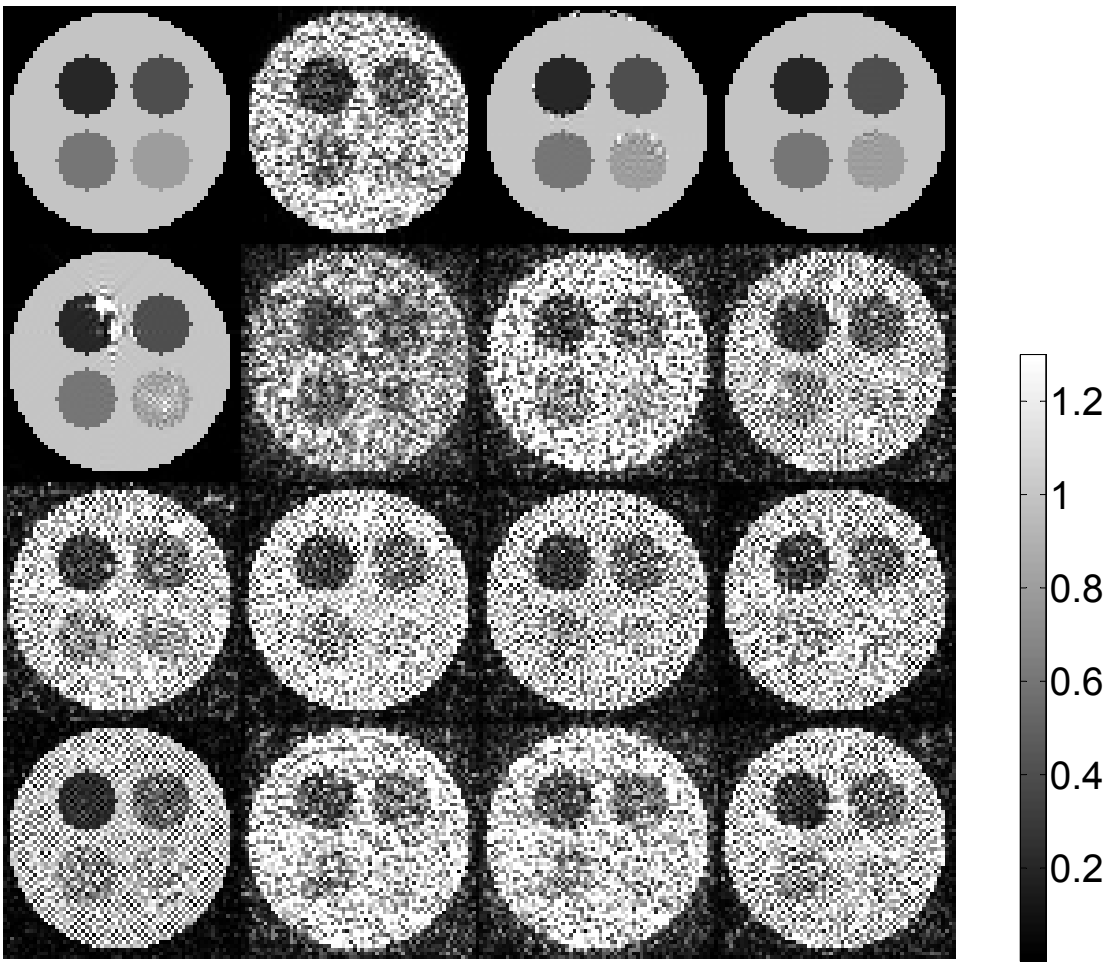


Figure 5.9: The true spin density map and its estimation from the 15 single-shot trajectories: (1st row) ground truth, EPI, REPI1, REPI2; (2nd row) RectSPR, SPR-in, SPR-out, SPR-in-in; (3rd row) SPR-in-out, SPR-out-out, SPR-out-in, ROS; (4th row) RROS, RAD1, RAD2, RRAD.

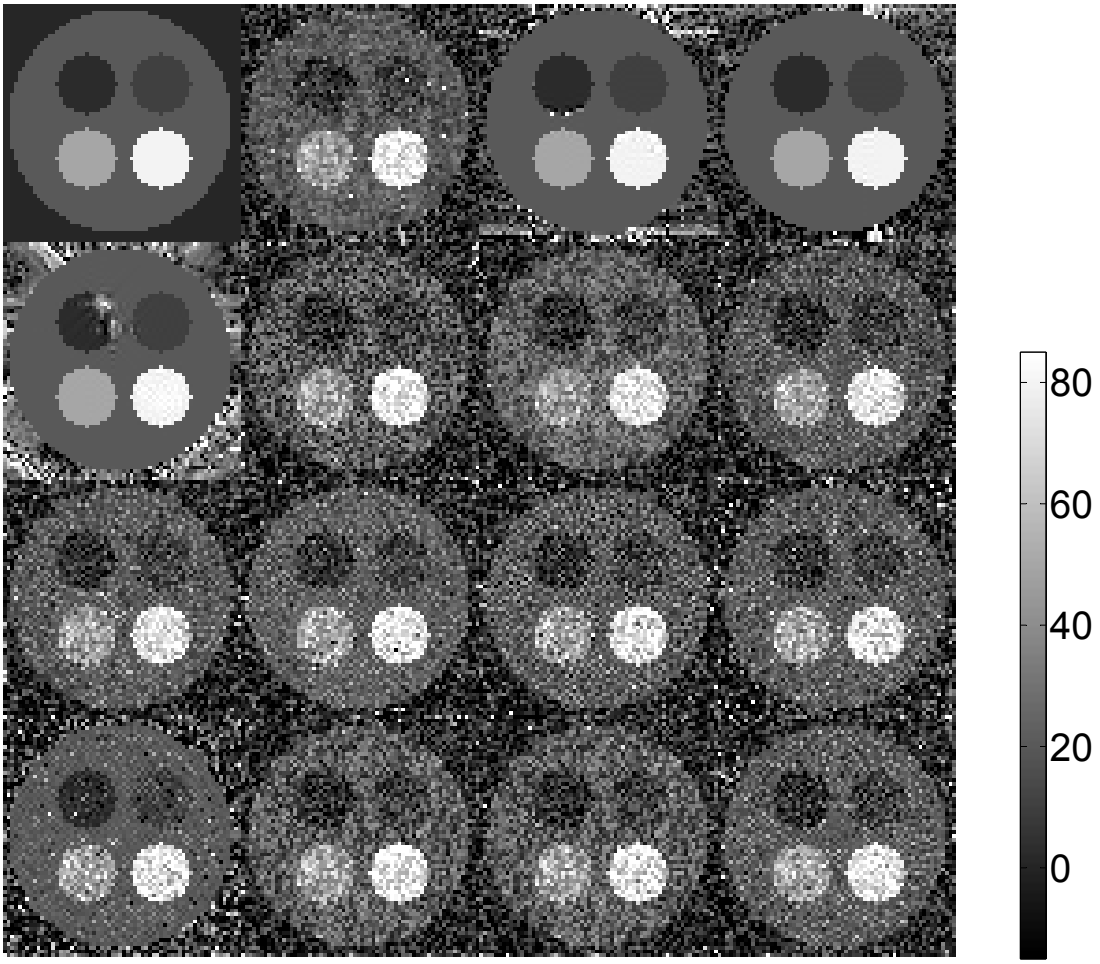


Figure 5.10: The true R_2^* decay map and its estimation from the 15 single-shot trajectories. The images are arranged in the same order Figure 5.9 has.

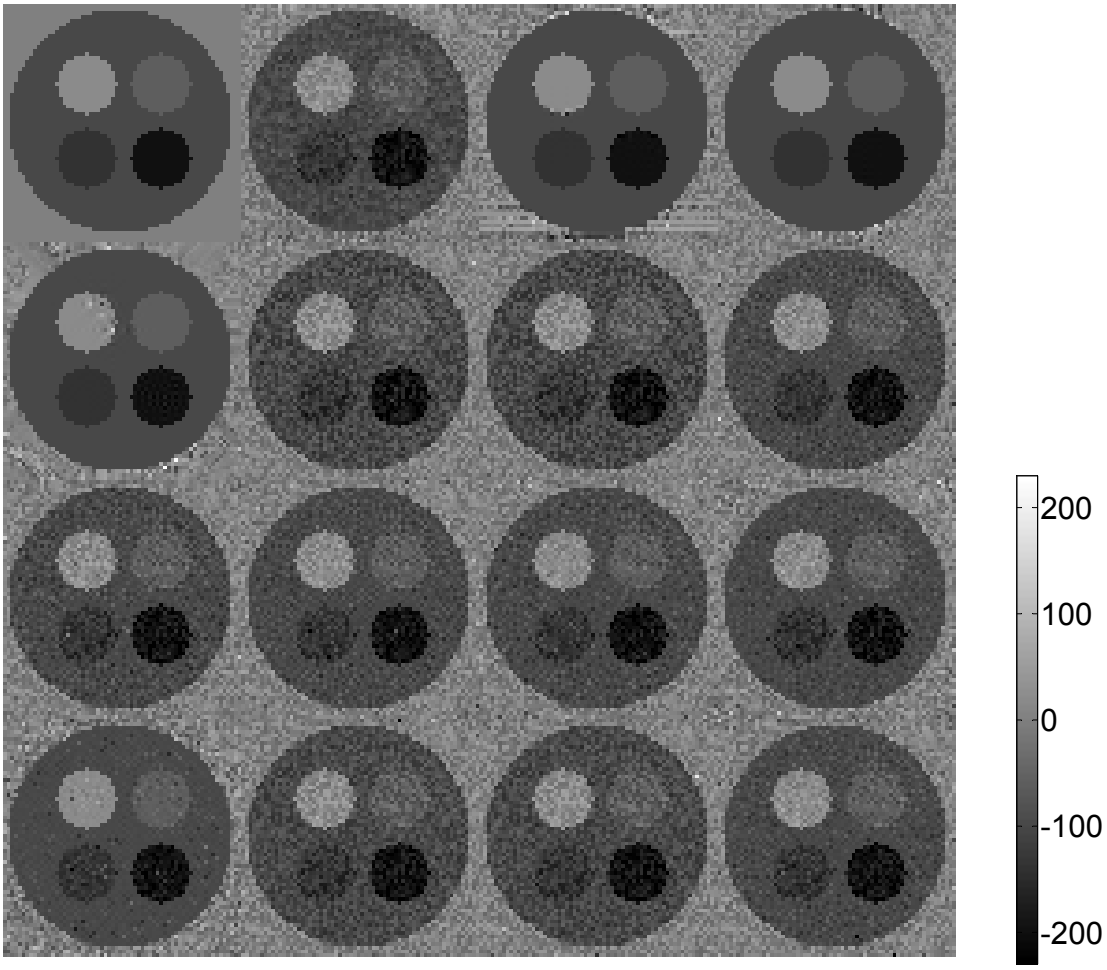


Figure 5.11: The true off-resonance frequency map and its estimation from the 15 single-shot trajectories. The images are arranged in the same order Figure 5.9 has.

Figure 5.12: The true spin density map and its estimation from the 15 single-shot trajectories: (1st row) ground truth, EPI, REPI1, REPI2; (2nd row) RectSPR, SPR-in, SPR-out, SPR-in-in; (3rd row) SPR-in-out, SPR-out-out, SPR-out-in, ROS; (4th row) RROS, RAD1, RAD2, RRAD.

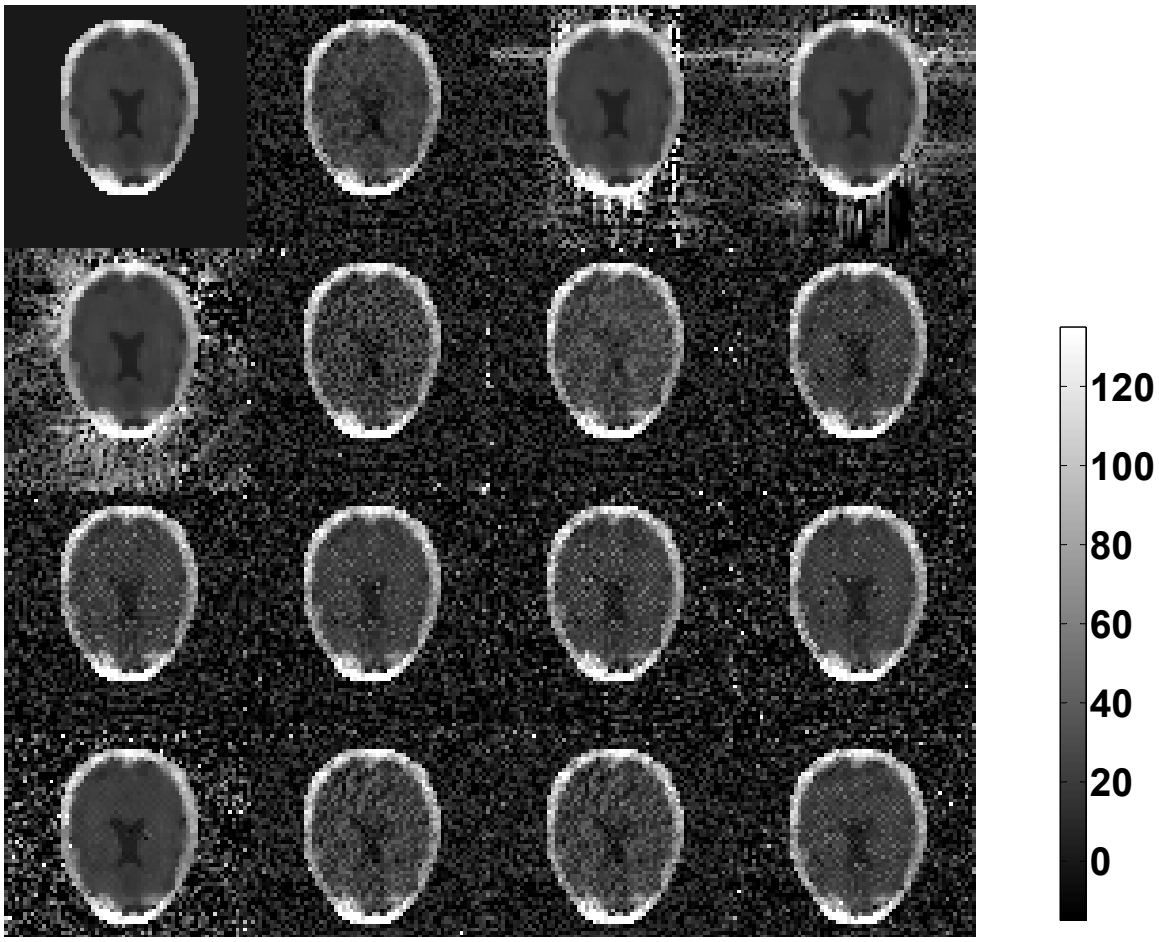


Figure 5.13: The true R_2^* decay map and its estimation from the 15 single-shot trajectories. The images are arranged in the same order Figure 5.12 has.

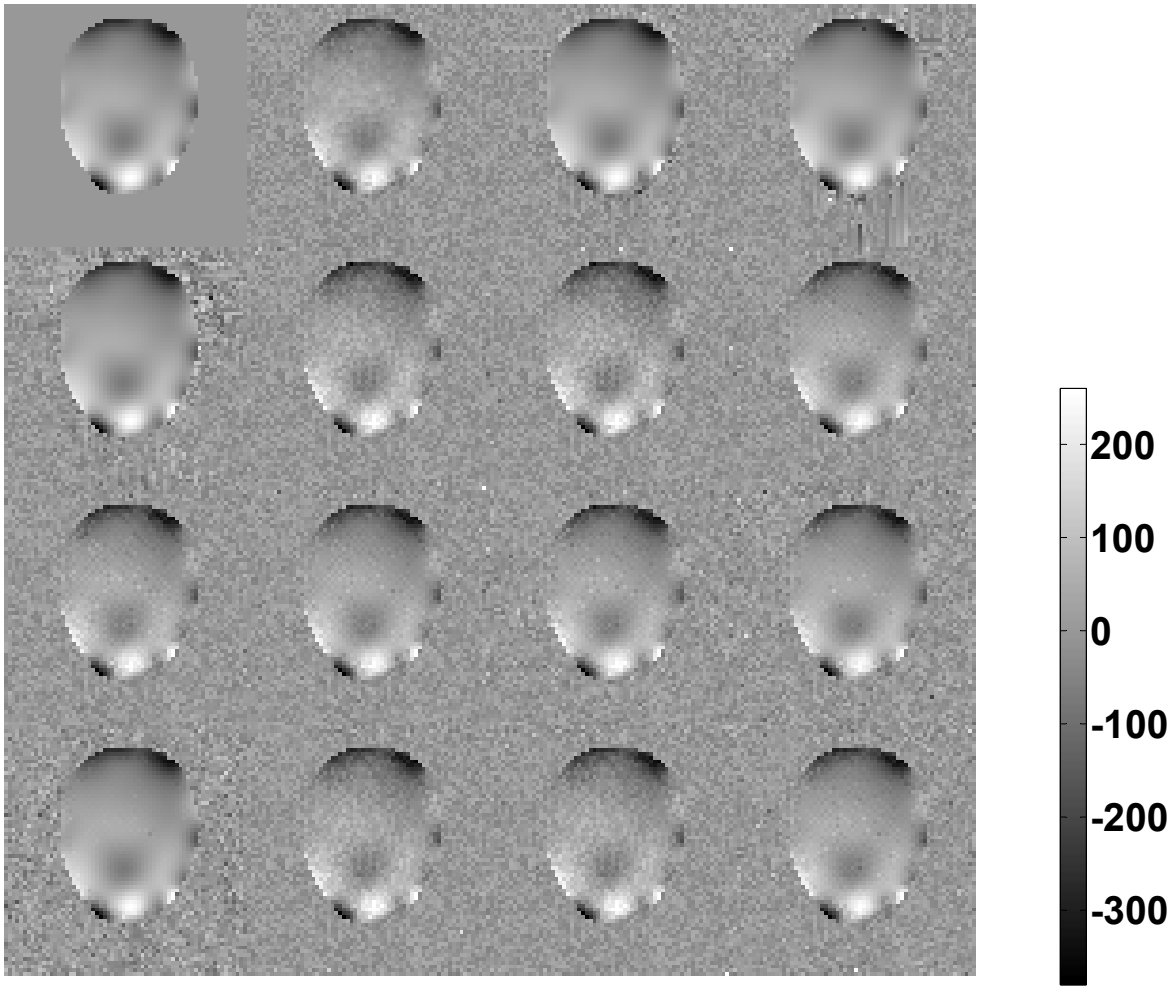
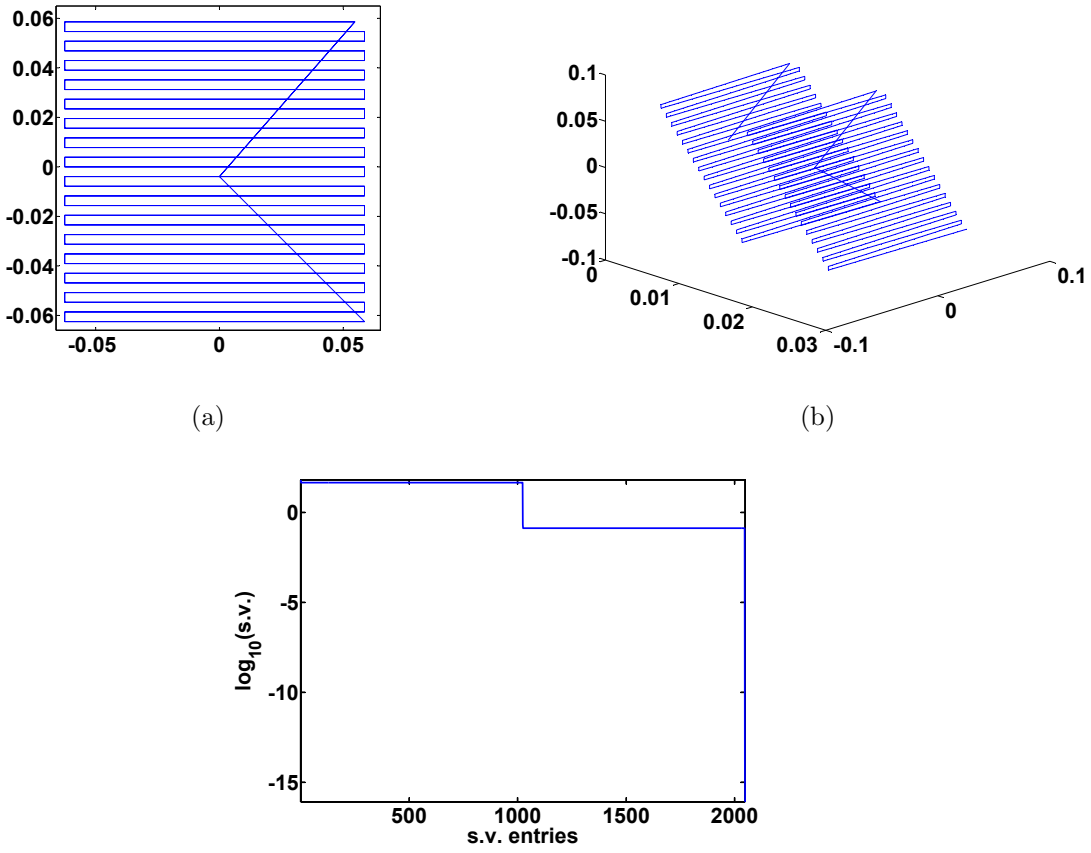


Figure 5.14: The true off-resonance frequency map and its estimation from the 15 single-shot trajectories. The images are arranged in the same order Figure 5.12 has.



(a) (b)

(c)

Figure 5.15: The modified REPI1 trajectory replaces the sampling of the grid point on the right upper corner of k-space (a) with a sampling of k-space origin. The condition number of the modified REPI1 trajectory evaluated at the phantom is 7.93E17, which is dramatically larger than the original REPI1.

Chapter 6

Conclusions

6.1 Summary of the dissertation

We have presented several studies on methods and properties of the problem of jointly reconstructing spin density, R_2^* decay, and off-resonance images in MRI with a single-shot or multi-shot acquisition. Specifically, the dissertation includes the following contributions to this problem:

- We have developed two trust region methods to address the joint reconstruction problem. The two different linearization strategies cause the two methods to have their own advantages. Overall, the ordinary trust region method is more stable in the face of variation of parameters and therefore is more reliable. The trust region framework helps the two algorithms maintain stability and converge fast in solving the nonlinear optimization problem. We have demonstrated through both simulated phantom, real phantom, and *in vivo* data that the proposed trust region algorithms outperform other traditional methods.

- We have also developed a novel auxiliary variable method to solve the same problem in a more efficient manner. We reformulated the problem as a constrained optimization problem by employing an auxiliary variable and then transformed it to an unconstrained problem by the penalty method. An alternating direction method was used to dramatically reduce the computational burden of evaluating the nonlinear cost function and gradient in traditional gradient-based iterative methods. We identified a shortcoming of the generic formulation of the auxiliary variable method and sped up the algorithm by using a combination of spatial and temporal regularization. Since the

speed gradually decreases as the iteration proceeds, we proposed a hybrid minimization scheme that uses the ordinary trust region method to refine the result from the auxiliary variable method. The hybrid algorithm is robust to variation of initialization and parameters. We showed through simulated and real *in vivo* data that the hybrid algorithm is faster than the ordinary trust region method and nonlinear conjugate gradients.

- We have proposed a novel non-fit estimator of spin density, R_2^* decay, and off-resonance maps in the presence of multi-echo data. The new estimator is much faster than traditional log-scale curve fitting and the ordinary nonlinear curve fitting of the decaying exponential model. With certain modifications, the estimator can also achieve similar accuracy and estimation variance compared to the curve fitting techniques in the presence of a small number of echoes. The criterion that defines the estimator may provide a new way of representing the mono-exponential model. Since the criterion is of linear form, the criterion as well as the new estimator can be used in the auxiliary variable method to accelerate the computation.

- We have presented a study of ill-conditioning for a variety of single-shot trajectories. We used tools such as the condition number, singular value curve, and reconstruction accuracy from the ordinary trust region method to investigate the conditioning associated with these trajectories. Our result shows that for a fixed readout time, repeated EPI has a better conditioning than repeated rosette. This advantage is mainly due to the shortcoming of the rosette that only an inscribed circular area of k-space is sampled. This shortcoming is also present for spiral and radial trajectories; however, we found that rosette trajectories are better than these trajectories in terms of conditioning. We have also found that repeating samples can effectively improve the conditioning. These findings can be useful for any algorithms developed for the joint

reconstruction problem. These findings can also be very helpful for researchers to find an optimal regularization and regularization weight for a particular trajectory.

6.2 Future work

The following future work is suggested by our previous studies on this problem:

- The auxiliary variable method still manifests slow convergence near the end of the iteration. This problem calls for an early stopping of the auxiliary variable method followed by an additional refinement algorithm. We suspect the problem is due to use of the penalty method. Other algorithms that can change a constrained problem to an unconstrained problem also exist, such as the alternating direction method of multipliers. We would like to investigate whether this method can be applied to our problem and to compare its performance with the current auxiliary variable method.

- The first step of the auxiliary variable method uses a preconditioned conjugate gradient algorithm, which slows down the algorithm. Reducing the computation in the first step can dramatically increase the speed. Therefore, we desire to find a new formulation of the first step so that a direct solution is present.

- The novel linear formulation proposed in Chapter 4 can potentially be used to develop a new auxiliary variable method. The new formulation can result in faster computation and may also produce other good properties. Research on the new formulation as well the geometric progression estimator is still immature. Continuing study on this topic may bring a new avenue for solving the underlying joint reconstruction problem.

- Regularization is still an open question for this problem. The study of singular values in Chapter 5 suggests we use SVD and generalized SVD [45] to investigate the creation of an optimal regularization for each individual trajectory. A good regularization may dramatically improve the conditioning. The selection of regularization weight can also be studied with a similar approach.

- Sparsity-promoting regularization has been used to reconstruct the parameter maps given sparsely sampled multi-echo data. Despite its success in this approach, such regularization has never been applied to the joint reconstruction problem based on single-shot or multi-shot acquisition. The main difficulty lies in increased computational complexity; however, the auxiliary variable formulation may very well accommodate the sparsity-promoting regularization while keeping the computation relatively easy.

- The reconstruction based on real data, especially *in vivo* data, still calls for improvement. We have used a state-of-the-art trajectory calibration technique in our research; however, the model mismatch is still very strong. This suggests perhaps we have some remaining obstacles in trajectory design or in other real-world factors. A step-by-step diagnosis process may need to be designed to find out the problem and enable reconstruction with real data to achieve clinically acceptable accuracy.

Appendix

Multi-echo trajectories

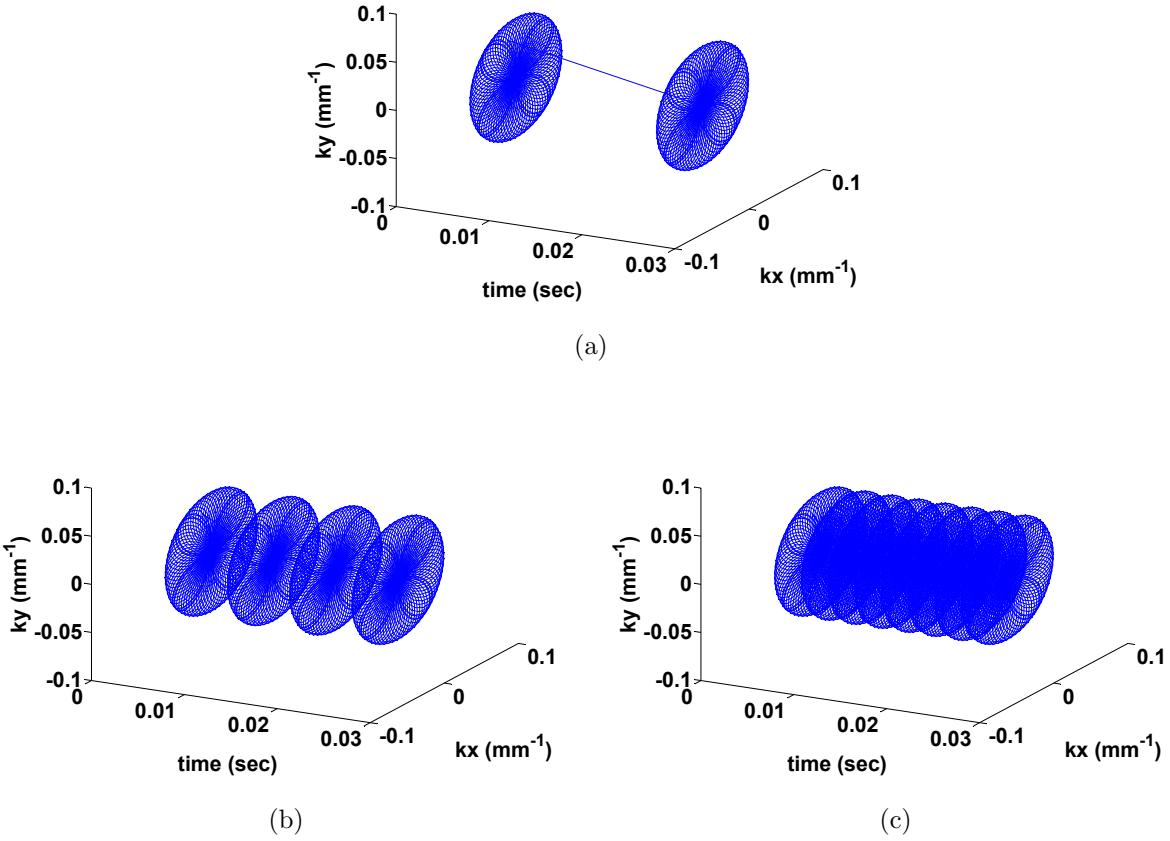
We extended the four types of trajectory, namely the rosette, EPI, spiral, and radial, to multi-echo patterns and tested their conditioning. Notice that a multi-echo sampling means that data of each echo is acquired at a single time point (the echo time) instead over a time frame. Although this is impossible to realize, the error caused by violation of the assumption is small as long as the readout time is no greater than 1 or 2 milliseconds, such as in a line-by-line sampling scheme. However, when a trajectory is used for acquiring an echo, finishing a single-shot trajectory in 1 or 2 milliseconds is generally impossible. Interleaving is therefore required. In this chapter, since we work only on calculating condition numbers of multi-echo trajectories, interleaving is ignored and we just assume each trajectory is finished instantly. However, a real multi-echo acquisition may take much longer time to acquire all data.

- For rosette, 2-echo, 4-echo, and 8-echo rosette trajectories were simulated. In each echo, we used a regular N^2-long rosette trajectory. Notice that the samples of each rosette trajectory must be collected at a single time point for a multi-echo acquisition. All echoes were uniformly placed between 5 milliseconds and 25 milliseconds so that the first and last echo were placed at 5 and 25 milliseconds respectively and the rest echoes were equally placed within the time frame. The three trajectories are shown in Figure 1. Their abbreviations are MEROS2, MEROS4, and MEROS8.

- Similarly, three multi-echo variants of the EPI trajectory were studied. They are 2-echo, 4-echo, and 8-echo EPI trajectories. Their abbreviations are MEEPI2, MEEPI4, and MEEPI8, and they are shown in Figure 2.

- Three multi-echo spiral-out trajectories were studied, namely the 2-echo (MESPR2), 4-echo (MESPR4), 8-echo (MESPR8) spiral-out trajectories. They are shown in Figure 3.

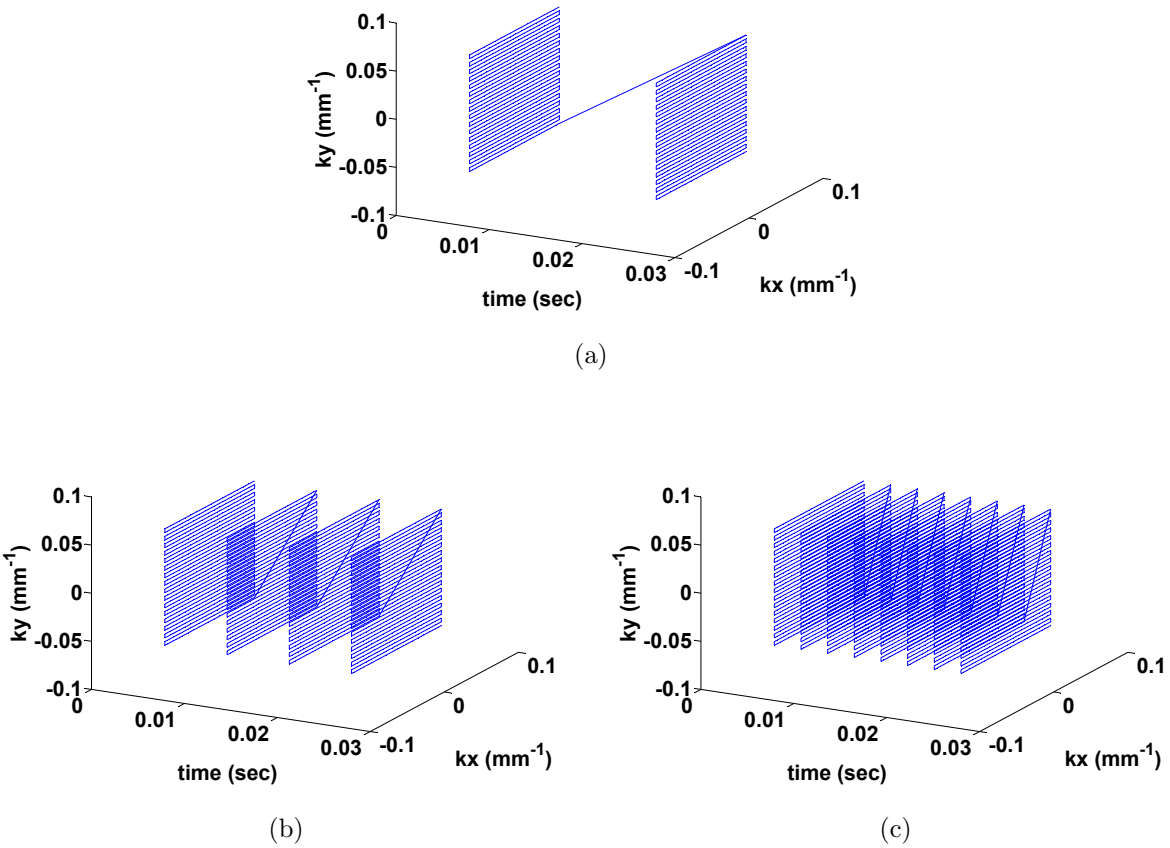
- Three multi-echo radial trajectories were studied, namely the 2-echo (MERAD2), 4-echo (MERAD4), 8-echo (MERAD8) radial trajectories. They are shown in Figure 3.



(a)

(b) (c)

Figure 1: The three multi-echo rosette trajectories: (a)MEROS2, (b)MEROS4, and (c)MEROS8

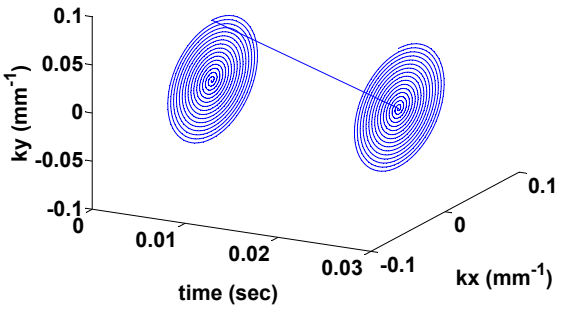


(a)

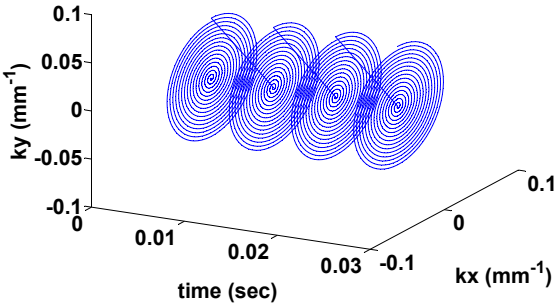
(b)

(c)

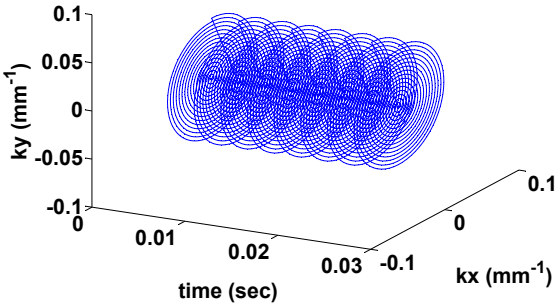
Figure 2: The three multi-echo EPI trajectories: (a)MEEPI2, (b)MEEPI4, and (c)MEEPI8



(a)

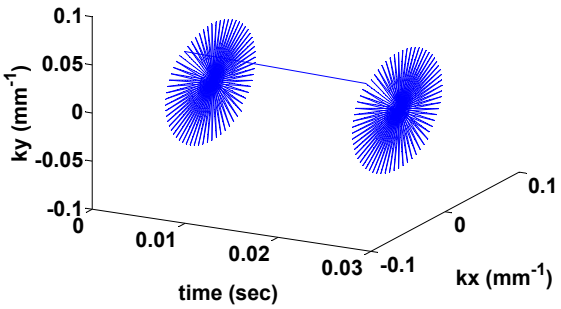


(b)

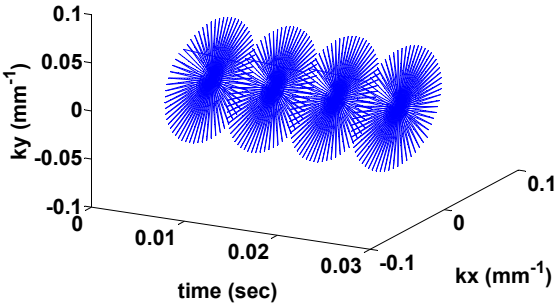


(c)

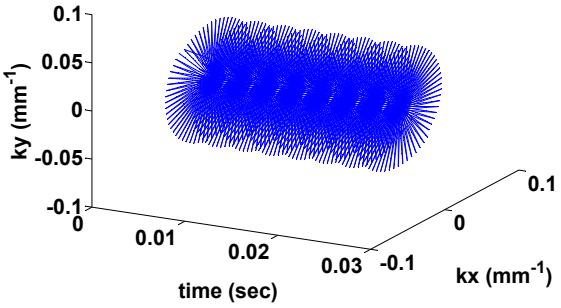
Figure 3: The three multi-echo spiral-out trajectories: (a)MESPR2, (b)MESPR4, and (c)MESPR8



(a)



(b)



(c)

Figure 4: The three multi-echo radial trajectories: (a)MERAD2, (b)MERAD4, and (c)MERAD8

Bibliography

[1] M.V. Afonso, J.M. Bioucas-Dias, and M.A.T. Figueiredo. An augmented lagrangian approach to the constrained optimization formulation of imaging inverse problems. *Image Processing, IEEE Transactions on*, 20(3):681–695, March 2011.

[2] Michael J Allison, Sathish Ramani, and Jeffrey A Fessler. Regularized mr coil sensitivity estimation using augmented lagrangian methods. In *Biomedical Imaging (ISBI), 2012 9th IEEE International Symposium on*, pages 394–397. IEEE, 2012.

[3] M.J. Allison, S. Ramani, and J.A. Fessler. Accelerated regularized estimation of mr coil sensitivities using augmented lagrangian methods. *Medical Imaging, IEEE Transactions on*, 32(3):556–564, March 2013.

[4] LJ Anderson, S Holden, B Davis, E Prescott, CC Charrier, NH Bunce, DN Firmin, B Wonke, J Porter, JM Walker, et al. Cardiovascular T2-star (T2*) magnetic resonance for the early diagnosis of myocardial iron overload. *European Heart Journal*, 22(23):2171–2179, 2001.

[5] Markus Barth, Alexander Metzler, Markus Klarhöfer, Stefan Röll, Ewald Moser, and Dieter Leibfritz. Functional MRI of the human motor cortex using single-shot, multiple gradient-echo spiral imaging. *Magnetic resonance imaging*, 17(9):1239–1243, 1999.

[6] Matt A Bernstein, Kevin F King, and Xiaohong Joe Zhou. *Handbook of MRI pulse sequences*. Elsevier, 2004.

[7] K.T. Block, M. Uecker, and J. Frahm. Model-based iterative reconstruction for radial fast spin-echo MRI. *Medical Imaging, IEEE Transactions on*, 28(11):1759–1769, Nov 2009.

[8] Robert W Brown, Y-C Norman Cheng, E Mark Haacke, Michael R Thompson, and Ramesh Venkatesan. *Magnetic resonance imaging: physical principles and sequence design*. John Wiley & Sons, 2014.

[9] Eric Y Chang, Jiang Du, Kenyu Iwasaki, Reni Biswas, Sheronda Statum, Qun He, Won C Bae, and Christine B Chung. Single-and Bi-component T2* analysis of tendon before and during tensile loading, using UTE sequences. *Journal of Magnetic Resonance Imaging*, 2014.

[10] Weitian Chen and Craig H. Meyer. Fast automatic linear off-resonance correction method for spiral imaging. *Magnetic Resonance in Medicine*, 56(2):457–462, 2006.

[11] Weitian Chen and Craig H. Meyer. Semiautomatic off-resonance correction in spiral imaging. *Magnetic Resonance in Medicine*, 59(5):1212–1219, 2008.

[12] Laurent D Cohen. Auxiliary variables and two-step iterative algorithms in computer vision problems. *Journal of Mathematical Imaging and Vision*, 6(1):59–83, 1996.

[13] T. Coleman and Y. Li. An interior trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on Optimization*, 6(2):418–445, 1996.

[14] Patrick L Combettes and Jean-Christophe Pesquet. Proximal splitting methods in signal processing. In *Fixed-point algorithms for inverse problems in science and engineering*, pages 185–212. Springer, 2011.

[15] Ryan Compton, Stanley Osher, and Louis Bouchard. Hybrid regularization for MRI reconstruction with static field inhomogeneity correction. In *Biomedical Imaging (ISBI), 2012 9th IEEE International Symposium on*, pages 650–655. IEEE, 2012.

[16] Biswa Nath Datta. *Numerical linear algebra and applications*. Siam, 2010.

[17] Mike Davies, Gilles Puy, Pierre Vandergheynst, and Yves Wiaux. A compressed sensing framework for magnetic resonance fingerprinting. *SIAM Journal on Imaging Sciences*, 7(4):2623–2656, 2014.

[18] H. de Leeuw and C.J.G. Bakker. Correction of gradient echo images for first and second order macroscopic signal dephasing using phase derivative mapping. *NeuroImage*, 60(1):818 – 829, 2012.

[19] Nicola De Zanche, Christoph Barmet, Jurek A Nordmeyer-Massner, and Klaas P Pruessmann. Nmr probes for measuring magnetic fields and field dynamics in mr systems. *Magnetic Resonance in Medicine*, 60(1):176–186, 2008.

[20] Robert L DeLaPaz. Echo-planar imaging. *Radiographics*, 14(5):1045–1058, 1994.

[21] Sean CL Deoni, Brian K Rutt, and Terry M Peters. Rapid combined T1 and T2 mapping using gradient recalled acquisition in the steady state. *Magnetic Resonance in Medicine*, 49(3):515–526, 2003.

[22] Mariya Doneva, Peter Börnert, Holger Eggers, Christian Stehning, Julien Sénégas, and Alfred Mertins. Compressed sensing reconstruction for magnetic resonance parameter mapping. *Magnetic Resonance in Medicine*, 64(4):1114–1120, 2010.

[23] Yiping P Du, Zhaoyang Jin, Yuzheng Hu, and Jody Tanabe. Multi-echo acquisition of MR angiography and venography of the brain at 3 Tesla. *Journal of Magnetic Resonance Imaging*, 30(2):449–454, 2009.

[24] Jean-Pierre Dussault. Numerical stability and efficiency of penalty algorithms. *SIAM Journal on Numerical Analysis*, 32(1):296–317, 1995.

[25] Jeff H Duyn, Yihong Yang, Joseph A Frank, and Jan Willem van der Veen. Simple correction method fork-space trajectory deviations in MRI. *Journal of Magnetic Resonance*, 132(1):150–153, 1998.

[26] Arin M Ellingson, Hitesh Mehta, David W Polly, Jutta Ellermann, and David J Nuckley. Disc degeneration assessed by quantitative T2*(T2 star) correlated with functional lumbar mechanics. *Spine*, 38(24):E1533–40, 2013.

[27] John Ernest Esser. *Primal dual algorithms for convex models and applications to image restoration, registration and nonlocal inpainting*. PhD thesis, University of California Los Angeles, 2010.

[28] Jeffrey Fessler. Model-based image reconstruction for MRI. *Signal Processing Magazine, IEEE*, 27(4):81–89, 2010.

[29] Jeffrey A Fessler, Sangwoo Lee, Valur T Olafsson, Hugo R Shi, and Douglas C Noll. Toeplitz-based iterative image reconstruction for mri with correction for magnetic field inhomogeneity. *Signal Processing, IEEE Transactions on*, 53(9):3393–3402, 2005.

[30] Jeffrey A Fessler and Bradley P Sutton. Nonuniform fast fourier transforms using min-max interpolation. *Signal Processing, IEEE Transactions on*, 51(2):560–574, 2003.

[31] Jeffrey A Fessler, Desmond Yeo, and Douglas C Noll. Regularized fieldmap estimation in mri. In *Biomedical Imaging: Nano to Macro, 2006. 3rd IEEE International Symposium on*, pages 706–709. IEEE, 2006.

[32] Amanda K Funai, Jeffrey A Fessler, Desmond Yeo, Valur T Olafsson, and Douglas C Noll. Regularized field map estimation in MRI. *Medical Imaging, IEEE Transactions on*, 27(10):1484–1494, 2008.

[33] Urs Gamper, Peter Boesiger, and Sebastian Kozerke. Compressed sensing in dynamic mri. *Magnetic Resonance in Medicine*, 59(2):365–373, 2008.

[34] Jia-Hong Gao, Ira Miller, Song Lai, Jinhu Xiong, and Peter T Fox. Quantitative assessment of blood inflow effects in functional mri signals. *Magnetic resonance in medicine*, 36(2):314–319, 1996.

[35] Donald Geman and George Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Transactions on pattern analysis and machine intelligence*, 14(3):367–383, 1992.

[36] Donald Geman and Chengda Yang. Nonlinear image recovery with half-quadratic regularization. *Image Processing, IEEE Transactions on*, 4(7):932–946, 1995.

[37] Philip E Gill, Walter Murray, and Margaret H Wright. *Practical optimization*, volume 5. Academic press London, 1981.

[38] Gary H Glover et al. Simple analytic spiral k-space algorithm. *Magnetic Resonance in medicine*, 42(2):412–415, 1999.

[39] Mark A Griswold, Peter M Jakob, Robin M Heidemann, Mathias Nittka, Vladimir Jellus, Jianmin Wang, Berthold Kiefer, and Axel Haase. Generalized autocalibrating partially parallel acquisitions (GRAPPA). *Magnetic Resonance in Medicine*, 47(6):1202–1210, 2002.

[40] Gisela E Hagberg, Marta Bianciardi, Fabiana Patria, and Iole Indovina. Quantitative NumART_2^* mapping in functional MRI studies at 1.5 T. *Magnetic resonance imaging*, 21(10):1241–1249, 2003.

[41] William W Hager and Hongchao Zhang. A survey of nonlinear conjugate gradient methods. *Pacific journal of Optimization*, 2(1):35–58, 2006.

[42] Elaine T Hale, Wotao Yin, and Yin Zhang. A fixed-point continuation method for l1-regularized minimization with applications to compressed sensing. *CAAM TR07-07, Rice University*, 2007.

[43] Martin Hanke. *Conjugate gradient type methods for ill-posed problems*, volume 327. CRC Press, 1995.

[44] Per Christian Hansen. Analysis of discrete ill-posed problems by means of the l-curve. *SIAM review*, 34(4):561–580, 1992.

[45] Per Christian Hansen. *Rank-deficient and discrete ill-posed problems: numerical aspects of linear inversion*, volume 4. Siam, 1998.

[46] Richard D Hoge, Remi KS Kwan, and G Bruce Pike. Density compensation functions for spiral MRI. *Magnetic Resonance in Medicine*, 38(1):117–128, 1997.

[47] Chenxi Hu and Stanley Reeves. On the characterization and reduction of ill-conditioning in jointly estimating proton density, T_2^* decay and the field map. In *International Symposium on Biomedical Imaging*. http://hdl.handle.net/11200/44672, 2014.

[48] Chenxi Hu and Stanley Reeves. An efficient auxiliary variable method for quantification of spin density, R2* decay and field inhomogeneity maps in magnetic resonance imaging. IEEE symposium on biomedical imaging (ISBI), 2015.

[49] Chenxi Hu and Stanley Reeves. Trust region methods for estimation of a complex exponential decay model in mri with a single-shot or multi-shot trajectory (in press). 2015.

[50] Chenxi Hu and Stanley J Reeves. A regularized trust region method for joint reconstruction of spin magnitude, t2* decay, and off-resonance field map. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 1882–1886. IEEE, 2014.

[51] Chuan Huang, Christian G Graff, Eric W Clarkson, Ali Bilgin, and Maria I Altbach. T2 mapping from highly undersampled data by reconstruction of principal component coefficient maps using compressed sensing. *Magnetic Resonance in Medicine*, 67(5):1355–1366, 2012.

[52] Junzhou Huang, Chen Chen, and Leon Axel. Fast multi-contrast MRI reconstruction. *Magnetic resonance imaging*, 32(10):1344–1352, 2014.

[53] Scott A Huettel, Allen W Song, and Gregory McCarthy. *Functional magnetic resonance imaging*, volume 1. Sinauer Associates Sunderland, 2004.

[54] Jérôme Idier. Convex half-quadratic criteria and interacting auxiliary variables for image restoration. *Image Processing, IEEE Transactions on*, 10(7):1001–1009, 2001.

[55] Hong Jung, Kyunghyun Sung, Krishna S Nayak, Eung Yeop Kim, and Jong Chul Ye. k-t focuss: A general compressed sensing framework for high resolution dynamic mri. *Magnetic Resonance in Medicine*, 61(1):103–116, 2009.

[56] Vladimir Juras, Sebastian Apprich, Pavol Szomolanyi, Oliver Bieri, Xeni Deligianni, and Siegfried Trattnig. Bi-exponential T2* analysis of healthy and diseased Achilles tendons: an in vivo preliminary magnetic resonance study and correlation with clinical score. *European radiology*, 23(10):2814–2822, 2013.

[57] Stephen L Keeling and Roland Bammer. A variational approach to magnetic resonance coil sensitivity estimation. *Applied Mathematics and Computation*, 158(2):359–388, 2004.

[58] Tobias Knopp, Holger Eggers, Hannes Dahnke, Jürgen Prestin, and Julien Senegas. Iterative off-resonance and signal decay estimation and correction for multi-echo MRI. *Medical Imaging, IEEE Transactions on*, 28(3):394–404, 2009.

[59] Randall M Kroeker and R Mark Henkelman. Analysis of biological nmr relaxation data with continuous distributions of relaxation times. *Journal of Magnetic Resonance (1969)*, 69(2):218–235, 1986.

[60] S Lee, D Noll, and JA Fessler. EXTended Rosette ACquisition technique (EXTRACT): A dynamic R2* mapping method using extended rosette trajectory. In *ISMRM Tenth Scientific Meeting*, page 2128, 2004.

[61] Adrian S Lewis and Stephen J Wright. A proximal method for composite minimization. *arXiv preprint arXiv:0812.0423*, 2008.

[62] Ningzhi Li, Mark Bolding, and Donald B Twieg. Spin-echo SS-PARSE: a PARSE MRI method to estimate frequency, R2 and R2* in a single shot. *Magnetic resonance imaging*, 28(9):1270–1282, 2010.

[63] Feng Liu and Stuart Crozier. An FDTD model for calculation of gradient-induced eddy currents in MRI system. *Applied Superconductivity, IEEE Transactions on*, 14(3):1983–1989, 2004.

[64] Hui Liu and Gerald B Matson. Accurate measurement of magnetic resonance imaging gradient characteristics. *Materials*, 7(1):1–15, 2013.

[65] Nikos K Logothetis. What we can do and what we cannot do with fmri. *Nature*, 453(7197):869–878, 2008.

[66] Nikos K Logothetis and Josef Pfeuffer. On the nature of the bold fmri contrast mechanism. *Magnetic resonance imaging*, 22(10):1517–1531, 2004.

[67] Michael Lustig, David Donoho, and John M Pauly. Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magnetic resonance in medicine*, 58(6):1182–1195, 2007.

[68] Michael Lustig, David L Donoho, Juan M Santos, and John M Pauly. Compressed sensing MRI. *Signal Processing Magazine, IEEE*, 25(2):72–82, 2008.

[69] Michael Lustig, Juan M Santos, David L Donoho, and John M Pauly. kt SPARSE: High frame rate dynamic MRI exploiting spatio-temporal sparsity. In *Proceedings of the 13th Annual Meeting of ISMRM, Seattle*, volume 2420, 2006.

[70] Dan Ma, Vikas Gulani, Nicole Seiberlich, Kecheng Liu, Jeffrey L Sunshine, Jeffrey L Duerk, and Mark A Griswold. Magnetic resonance fingerprinting. *Nature*, 495(7440):187–192, 2013.

[71] A Maeda, K Sano, and T Yokoyama. Reconstruction by weighted correlation for MRI with time-varying gradients. *Medical Imaging, IEEE Transactions on*, 7(1):26–31, 1988.

[72] Angshul Majumdar and Rabab K Ward. Joint reconstruction of multiecho MR images using correlated sparsity. *Magnetic resonance imaging*, 29(7):899–906, 2011.

[73] Tallal Charles Mamisch, Timothy Hughes, Timothy J Mosher, Christoph Mueller, Siegfried Trattnig, Chris Boesch, and Goetz Hannes Welsch. T2 star relaxation times for assessment of articular cartilage at 3 T: a feasibility study. *Skeletal radiology*, 41(3):287–292, 2012.

[74] Lai-Chee Man, John M Pauly, and Albert Macovski. Improved automatic off-resonance correction without a field map in spiral imaging. *Magnetic resonance in medicine*, 37(6):906–913, 1997.

[75] Antonios Matakos and Jeffrey A Fessler. Dynamic mr image and fieldmap joint reconstruction accounting for through-plane fieldmap gradients. In *Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on*, pages 393–396. IEEE, 2011.

[76] Antonis Matakos. *Dynamic Image and Fieldmap Joint Estimation Methods for MRI Using Single-Shot Trajectories*. PhD thesis, Citeseer, 2013.

[77] Stephen Gregory Nash. Truncated-newton methods. 1982.

[78] Arnold Neumaier. Solving ill-conditioned and singular linear systems: A tutorial on regularization. *SIAM review*, 40(3):636–666, 1998.

[79] Kien-Ming Ng. *A continuation approach for solving nonlinear optimization problems with discrete variables*. PhD thesis, stanford university, 2002.

[80] Giang Chau Ngo and Bradley P Sutton. R2* mapping for robust brain function detection in the presence of magnetic field inhomogeneity. In *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE*, pages 1537–1540. IEEE, 2014.

[81] Douglas C Noll. Multishot rosette trajectories for spectrally selective MR imaging. *Medical Imaging, IEEE Transactions on*, 16(4):372–377, 1997.

[82] Douglas C Noll, John M Pauly, Craig H Meyer, Dwight G Nishimura, and Albert Macovskj. Deblurring for non-2d fourier transform magnetic resonance imaging. *Magnetic Resonance in Medicine*, 25(2):319–333, 1992.

[83] Valur T. Olafsson, Douglas C. Noll, and Jeffrey A. Fessler. Fast joint reconstruction of dynamic R_2^* and field maps in functional MRI. *Medical Imaging, IEEE Transactions on*, 27(9):1177–1188, Sept 2008.

[84] Frederike H Petzschner, Irene P Ponce, Martin Blaimer, Peter M Jakob, and Felix A Breuer. Fast mr parameter mapping using k-t principal component analysis. *Magnetic Resonance in Medicine*, 66(3):706–716, 2011.

[85] Vincenzo Positano, Benedetta Salani, Alessia Pepe, Maria Filomena Santarelli, Daniele De Marchi, Anna Ramazzotti, Brunella Favilli, Eliana Cracolici, Massimo Midiri, Paolo Cianciulli, et al. Improved T2* assessment in liver iron overload by magnetic resonance imaging. *Magnetic resonance imaging*, 27(2):188–197, 2009.

[86] Klaas P Pruessmann, Markus Weiger, Markus B Scheidegger, Peter Boesiger, et al. SENSE: sensitivity encoding for fast MRI. *Magnetic resonance in medicine*, 42(5):952–962, 1999.

[87] Alfio Quarteroni, Riccardo Sacco, and Fausto Saleri. *Numerical mathematics*, volume 37. Springer Science & Business Media, 2010.

[88] S. Ramani and J.A. Fessler. Parallel mr image reconstruction using augmented lagrangian methods. *Medical Imaging, IEEE Transactions on*, 30(3):694–706, March 2011.

[89] JB Rosen. Two-phase algorithm for nonlinear constraint problems. *Nonlinear programming*, 3(0):97–124, 1977.

[90] Greg J Stanisz, Ewa E Odrobina, Joseph Pun, Michael Escaravage, Simon J Graham, Michael J Bronskill, and R Mark Henkelman. T1, T2 relaxation and magnetization transfer in tissue at 3T. *Magnetic Resonance in Medicine*, 54(3):507–512, 2005.

[91] Michael K Stehling, Robert Turner, and Peter Mansfield. Echo-planar imaging: magnetic resonance imaging in a fraction of a second. *Science*, 254(5028):43–50, 1991.

[92] T. Steihaug. The conjugate gradient method and trust regions in large scale optimization. *SIAM Journal on Numerical Analysis*, 20(3):626–637, 1983.

[93] Tilman J Sumpf, Martin Uecker, Susann Boretius, and Jens Frahm. Model-based nonlinear inverse reconstruction for T2 mapping using highly undersampled spin-echo MRI. *Journal of Magnetic Resonance Imaging*, 34(2):420–428, 2011.

[94] T.J. Sumpf, A. Petrovic, M. Uecker, F. Knoll, and J. Frahm. Fast T2 Mapping With Improved Accuracy Using Undersampled Spin-Echo MRI and Model-Based Reconstructions With a Generating Function. *Medical Imaging, IEEE Transactions on*, 33(12):2213–2222, Dec 2014.

[95] B. P. Sutton, S. J. Peltier, J. A. Fessler, and D. C. Noll. Simultaneous estimation of I_0, R_2^*, and field map using a multi-echo spiral acquisition. page 1323. Proc. Intl. Soc. Mag. Reson. Med., Oct 2002.

[96] Bradley P Sutton. *Physics Based Iterative Reconstruction for MRI: Compensating and Estimating Field Inhomogeneity and T2 Relaxation*. PhD thesis, The University of Michigan, 2003.

[97] Bradley P Sutton, Douglas C Noll, and Jeffrey A Fessler. Fast, iterative image reconstruction for MRI in the presence of field inhomogeneities. *Medical Imaging, IEEE Transactions on*, 22(2):178–188, 2003.

[98] Bradley P Sutton, Douglas C Noll, and Jeffrey A Fessler. Dynamic field map estimation using a spiral-in/spiral-out acquisition. *Magnetic Resonance in Medicine*, 51(6):1194–1204, 2004.

[99] Hao Tan and Craig H Meyer. Estimation of k-space trajectories in spiral mri. *Magnetic Resonance in Medicine*, 61(6):1396–1404, 2009.

[100] Weidong Tang. *Reconstruction of Parametric Image Maps in Single- and Multiple-Coil Functional Magnetic Resonance Imaging*. PhD thesis, Auburn University, 2009.

[101] Jeffrey Tsao, Peter Boesiger, and Klaas P Pruessmann. k-t BLAST and k-t SENSE: Dynamic MRI with high frame rate exploiting spatiotemporal correlations. *Magnetic Resonance in Medicine*, 50(5):1031–1042, 2003.

[102] Donald B Twieg. Parsing local signal evolution directly from a single-shot MRI signal: A new approach for fMRI. *Magnetic Resonance in Medicine*, 50(5):1043–1052, 2003.

[103] Donald B Twieg and Stanley J Reeves. Basic properties of SS-PARSE parameter estimates. *Medical Imaging, IEEE Transactions on*, 29(5):1156–1172, 2010.

[104] Martin Uecker, Thorsten Hohage, Kai Tobias Block, and Jens Frahm. Image reconstruction by regularized nonlinear inversion–joint estimation of coil sensitivities and image content. *Magnetic Resonance in Medicine*, 60(3):674–682, 2008.

[105] Martin Uecker, Shuo Zhang, Dirk Voit, Alexander Karaus, Klaus-Dietmar Merboldt, and Jens Frahm. Real-time MRI at a resolution of 20 ms. *NMR in Biomedicine*, 23(8):986–994, 2010.

[106] G Van der Hoek. Asymptotic properties of reduction methods applying linearly equality constrained reduced problems. In *Algorithms for Constrained Minimization of Smooth Nonlinear Functions*, pages 162–189. Springer, 1982.

[107] Julia V Velikina, Andrew L Alexander, and Alexey Samsonov. Accelerating MR parameter mapping using sparsity-promoting regularization in parametric dimension. *Magnetic Resonance in Medicine*, 70(5):1263–1273, 2013.

[108] Baba C Vemuri and Yanlin Guo. Snake pedals: compact and versatile geometric models with physics-based control. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(5):445–459, 2000.

[109] Eliahu Wasserstrom. Numerical solutions by the continuation method. *SIAM Review*, 15(1):89–119, 1973.

[110] Eric W Weisstein. Least squares fitting–exponential. *MathWorld-A Wolfram Web Resource. http://mathworld. wolfram. com/LeastSquaresFittingExponential. html*, 2011.

[111] Anders B. A. Wennerberg, Tomas Jonsson, Hans Forssberg, and Tie-Qiang Li. A comparative fMRI study: T2*-weighted imaging versus R2* mapping. *NMR in Biomedicine*, 14(1):41–47, 2001.

[112] Kenneth P Whittall, Alex L Mackay, Douglas A Graeb, Robert A Nugent, David KB Li, and Donald W Paty. In vivo measurement of T2 distributions and water contents in normal human brain. *Magnetic Resonance in Medicine*, 37(1):34–43, 1997.

[113] Kenneth P Whittall, Alex L MacKay, and David KB Li. Are mono-exponential fits to a few echoes sufficient to determine T2 relaxation for in vivo human brain? *Magnetic resonance in medicine*, 41(6):1255–1257, 1999.

[114] John C Wood, Cathleen Enriquez, Nilesh Ghugre, J Michael Tyzka, Susan Carson, Marvin D Nelson, and Thomas D Coates. MRI R2 and R2* mapping accurately estimates hepatic iron concentration in transfusion-dependent thalassemia and sickle cell disease patients. *Blood*, 106(4):1460–1465, 2005.

[115] Max A Woodbury. Inverting modified matrices. *Memorandum report*, 42:106, 1950.

[116] Leslie Ying and Jinhua Sheng. Joint image reconstruction and sensitivity estimation in sense (jsense). *Magnetic Resonance in Medicine*, 57(6):1196–1202, 2007.

[117] Benjamin Zahneisen, Thimo Hugger, Kuan J Lee, Pierre LeVan, Marco Reisert, Hsu-Lei Lee, Jakob Assländer, Maxim Zaitsev, and Jürgen Hennig. Single shot concentric shells trajectories for ultra fast fmri. *Magnetic Resonance in Medicine*, 68(2):484–494, 2012.

[118] Yantian Zhang, Hoby P. Hetherington, Ernest M. Stokely, Graeme F. Mason, and Donald B. Twieg. A novel k-space trajectory measurement technique. *Magnetic Resonance in Medicine*, 39(6):999–1004, 1998.

[119] Bo Zhao, Fan Lam, Wenmiao Lu, and Zhi-Pei Liang. Model-based mr parameter mapping with sparsity constraint. In *Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium on*, pages 1–4. IEEE, 2013.