Fast and Motion Robust Dynamic R_2^* Reconstruction for Functional MRI

by

Valur Thor Olafsson

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering: Systems)
in The University of Michigan
2009

Doctoral Committee:
- Professor Jeffrey A. Fessler, Co-Chair
- Professor Douglas C. Noll, Co-Chair
- Professor Thomas L. Chenevert
- Professor Alfred O. Hero III
- Professor Victor Solo, The University of New South Wales

Abstract

Fast and Motion Robust Dynamic R_2^* Reconstruction for Functional MRI

by

Valur Thor Olafsson

Co-Chairs: Jeffrey A. Fessler and Douglas C. Noll

Blood oxygen level dependent (BOLD) functional MRI (fMRI) imaging is the most common way of imaging neuronal activity in humans using MRI. The BOLD contrast is directly related to changes in vascular physiology associated with neuronal activity and can be directly linked to changes in cerebral blood volume, blood flow and metabolic rate of oxygen. Conventional BOLD imaging is done by reconstructing T_2^*-weighted images. T_2^*-weighted images are unitless and even though they measure the magnitude of the BOLD contrast they are still nonquantifiable in terms of the vascular physiology. An alternative approach is to reconstruct R_2^* maps which are quantifiable and can be directly linked to the vascular changes during activation. However, conventional R_2^* mapping involves long readouts and generally ignores relaxation and off-resonance during readout.

Since fMRI data is usually acquired over a course of several minutes, where the same image volume is collected multiple times, it is important for the time series of each pixel to only reflect changes due to neuronal activity. However, BOLD imaging suffers from temporal drift/fluctuations and subject motion which can confound the findings. Conventionally, a field map is collected at the start of the fMRI study to correct for off-resonance, ignoring any possible changes in it due to either drift or motion. Here we propose a new fast and motion robust R_2^* iterative reconstruction that jointly reconstructs initial magnetization and field maps along with the R_2^* changes, for all time frames in fMRI. To accelerate the algorithm we propose to linearize the MR signal model, enabling the use of fast regularized iterative reconstruction methods. The regularizer was designed to account for the different resolution properties of both R_2^* and field maps and provide uniform spatial resolution.

To My Parents

Acknowledgments

First and foremost I would like to thank my two advisers, Professor Jeffrey A. Fessler and Professor Douglas C. Noll for their guidance during my time as a graduate student at the University of Michigan. I was very fortunate to have the opportunity to work with both of them. Both Doug and Jeff are very accomplished and knowledgeable researchers but in their own way. Jeff's main research focus is signal processing and statistics while Doug's main focus is functional MRI and signal processing. Their wide knowledge of these topics allowed me to get excellent mentorship in the topic that I chose for my dissertation and for that I am very thankful. I also would like to thank the rest of my dissertation committee, Professor Alfred O. Hero III, Professor Thomas L. Chenevert and Professor Victor Solo, for their valuable input to the work presented here.

I would also like to thank my former and current labmates at the fMRI lab here at the University of Michigan. I especially want to thank Dr. Bradley Sutton, who is now at the University of Illinois, for his mentorship my first year at the fMRI lab. His help and the work that he did before I came laid the foundation on which this work stands. I also would like to thank Dr. Alberto Vazquez, Dr. Gregory Lee and Dr. Kiran Pandey for their great friendship and discussions during my time in the lab. My thanks to other students in the lab such as Dr. William Grissom, Yoon Chung Kim, Daehyun Yoon and Hesamoddin Jahanian among others. I also would like to thank the support staff at the fMRI lab, Dr. Luis Hernandez, Dr. Scott Peltier, Keith Newnham and Eve Gochis for all their help during my stay here.

Finally I would like to thank my parents and my two sister for all their love and support they have given me. I am very grateful to have had parents that supported me and my sisters academic choices and allowed us the full freedom of exploring what we wanted to become. Last, but not least, I would like to thank my girlfriend Wendy Yau for all the love and support she has given me during our time together. She has picked me up when I was down and cheered me on when needed. For that I will be forever in debt to her. Thank you my love.

Valur Thor Olafsson
Ann Arbor, MI
May 27, 2009

Table of Contents

List of Figures

Figure

List of Appendices

Appendix

Chapter 1

Introduction

1.1 Conventional MRI Based Neuroimaging and Reconstruction

Neuroimaging using magnetic resonance (MR) is generally not sensitive enough to be able to directly detect neuronal firing. Rather it is conventionally done using an indirect measure of neuronal activity, such as through increased supply of oxygenated blood. For a site with increased neuronal activity the situation can be described as follows. Increased neuronal activity leads to increased metabolic rate of oxygen. This rate change is facilitated by delivering more oxygenated blood by increasing blood flow and blood volume close to the site of activity. These hemodynamic changes perturb the local magnetization due to different magnetic susceptibility properties of oxygenated and deoxygenated hemoglobin. This perturbation is detectable in water based MR imaging since the local field changes affects neighboring magnetized water molecules by changing their R_2^* relaxation rate, thus changing the contrast of the image in that area. This contrast mechanism is called blood oxygen level dependent (BOLD) contrast and it is the most common way of imaging neuronal activity in humans using functional MRI (fMRI).

The conventionally reconstructed images for fMRI are so called T_2^*-weighted images. However, these images are unitless and thus unquantifiable in terms of the hemodynamic changes, such as changes in blood flow, blood volume and metabolic rate of oxygen. Since the concentration of deoxygenated hemoglobin affects R_2^* directly from changes in blood flow, blood volume and metabolic rate of oxygen associated with neuronal activity, R_2^* is a more quantifiable measure of neuronal activity. Thus, an alternative approach for quantitative fMRI is to reconstruct images of the R_2^* relaxation rate. However, conventional reconstruction methods involve long readouts and generally ignore relaxation while collecting the data for one image. If the reconstruction uses a model where this relaxation is included, the problem becomes nonlinear and calls for slow optimization algorithms to solve. Both these restrictions have made R_2^* imaging for fMRI not desirable. Also, most of these reconstructions do not try to build into the algorithm some form of robustness to noise, such as using regularization functions or penalties.

An fMRI experiment is designed to capture dynamic changes in the hemodynamics as a subject performs a particular task that is to be investigated. This involves acquiring multiple time frames of a brain volume that is involved in the task. The functional analysis is then performed on the resulting pixel time series. If this analysis is performed on quantitative data such as R_2^* it is imperative that any changes to the MR signal not directly from functional activity be controlled. Some contributors of non-functional changes are motion and physiological changes such as breathing and cardiac cycle. These processes affect other parameters in the MR signal such as the initial magnetization in the slice or sources of off-resonance (corrected by using field maps). If they are not accounted for in every time frame they could confound any quantitative R_2^* inferences.

In this dissertation we propose a new fast regularized iterative algorithm that jointly reconstructs the magnetization, R_2^* and field map for every time frame in the fMRI data. To accelerate the algorithm we make a linear approximation to the MR signal model, enabling the use of fast regularized iterative reconstruction algorithms. The regularizer was designed to account for the inherently different resolution properties of both R_2^* and field maps and provide uniform spatial resolution for the estimated images.

1.2 Summary of Contributions

Here follows a list of contributions that were made in this dissertation:

Fast Iterative Dynamic R_2^* and Field Map Reconstruction

We propose and implement a linear approximation to the original signal model that allows us to use fast iterative reconstruction algorithms to estimate R_2^* and the field maps for every time frame (see Chapter 3).

Separate R_2^* and Field Map Roughness Penalty

We perform the reconstruction using a roughness penalty function that penalizes R_2^* and field map separately. This allowed us to relax the dependencies on the data fit, and thus we are able to use more conventional k-spaced trajectories to acquire data during an fMRI experiment. This also allows us to closely maintain the temporal resolution and spatial coverage that conventional fMRI experiments use today (see Chapter 3 and Chapter 5).

Spatial Resolution Analysis of Regularized Reconstruction

We developed a spatially varying roughness penalty for the $\boldsymbol{f}$ and R_2^* estimates that approximately maintains spatially uniform resolution on the reconstructed images imposed by the penalty. This is critical in fMRI since the R_2^* images will be to be used to detect areas of activation. Any spatially varying resolution for the resulting images affect this analysis leading to a possible increase in false positives or decrease in true positives if not controlled (see Chapter 3).

Fast Algorithm to Evaluate Effects of Separate Roughness Penalty

We developed a fast algorithm to estimate the spatial resolution of the reconstructed images when using a roughness penalty. This information can then be used to choose regularization parameters that give a desired spatial resolution in a computationally fast manner (see Chapter 4).

Fast and Motion Robust Iterative R_2^* Reconstruction

We developed a fast joint reconstruction of initial magnetization, R_2^* and field map that uses a similar linear approximation used in the R_2^* and field map reconstruction. By adding the initial magnetization as one of the parameters to be estimated we can account better for MR signal effects from such things as motion and blood inflow (see Chapter 5).

Chapter 2

Background

2.1 The MR Signal Equation

The MR signal equation is the foundation underlying all reconstruction algorithms in MR imaging. The MR signal equation $s(t)$ is given by the following expression:

$$s(t) = \int f(\vec{r})\, \mathrm{e}^{-tz(\vec{r})}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}\right)}\, \mathrm{d}\vec{r}, \quad z(\vec{r}) \triangleq R_2^*(\vec{r}) + i\omega(\vec{r})\,, \tag{2.1}$$

where t is time with $t = 0$ directly after RF excitation, $\vec{r}$ is a 2D or 3D image space coordinate, $f(\vec{r})$ is the magnetization of the object at $t = 0$, $z(\vec{r})$ is a complex-valued rate map with $R_2^*(\vec{r})$ as the signal relaxation rate map and $\omega(\vec{r})$ as the magnetic field offset map, and $\vec{k}(t)$ is the k-space trajectory used to acquire the MR data. The goal of this section is to explain the basics of MRI and in the process derive the MR signal equation as shown above. The reader may skip this section if familiar with the basic concepts of MRI.

2.1.1 MR Signal Sources, Excitation and Measurement

Magnetic resonance imaging (MRI) is an imaging modality that uses a magnetic property of some atoms, *e.g.*, ^{1}H, ^{13}C, ^{17}O and ^{31}P. In particular, these are atoms that contain an odd number of protons and/or neutrons. All of them possess a *nuclear spin angular momentum* $\boldsymbol{S}$, which gives rise to a small *magnetic dipole moment* $\boldsymbol{\mu}$, and are thus sometimes collec-

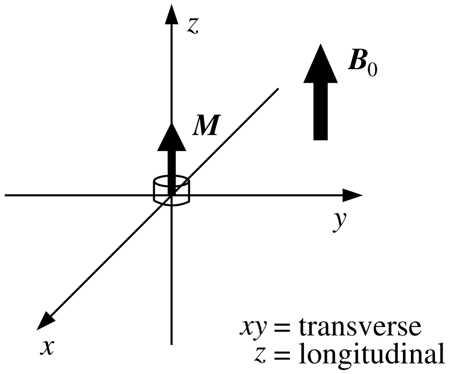


Figure 2.1: Applied magnetic field in longitudinal direction of strength B_0 forms a net magnetization moment $\boldsymbol{M}$ in the unit sample. At thermal equilibrium $\boldsymbol{M}$ is aligned in the direction of $\boldsymbol{B}_0$.

tively referred to as *spins*. Proton spins (^{1}H) are most often used in MR imaging of living organisms, since they are the most abundant (such as in water), the most sensitive (highest SNR) and the most studied.

If a collection of spins is exposed to an external static magnetic field $\boldsymbol{B}_0 \triangleq [0, 0, B_0]$, applied along the *longitudinal axis* (z-axis), two notable effects occur. First, a spins $\boldsymbol{\mu}$ tends to align in the direction of $\boldsymbol{B}_0$ to form a per unit volume *net magnetization moment* $\boldsymbol{M} = \sum \boldsymbol{\mu}$, as shown in Figure 2.1. Second, the spins exhibit resonance at a well-defined frequency ω_0, called the *Larmor frequency*. It is related to the applied magnetic field $\boldsymbol{B}_0$ by the following expression:

$$\omega_0 = \gamma \left| \boldsymbol{B}_0 \right|, \tag{2.2}$$

where γ is the *gyromagnetic ratio*, which is unique for each type of spin.

For MR imaging the objective is to measure the spatial distribution of $\boldsymbol{M}$ for the object being imaged. While this distribution can be generated by exposing the object to $\boldsymbol{B}_0$ as described earlier it does not allow for $\boldsymbol{M}$ to be measured. This is due to $\boldsymbol{M}$ pointing in the same direction as $\boldsymbol{B}_0$ at thermal equilibrium, making it impossible to discriminate between the two. However, when $\boldsymbol{M}$ is made to point in a different direction than $\boldsymbol{B}_0$ it exhibits a rotational behavior due to a torque exerted on the now nonzero *transverse component*

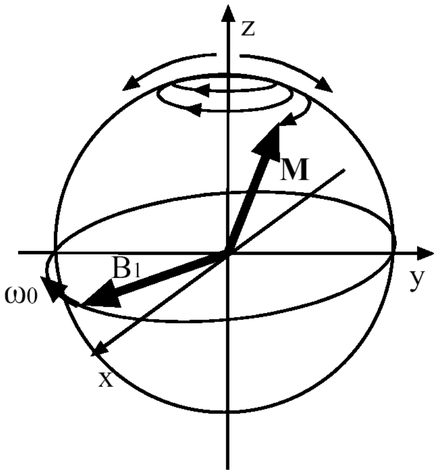


Figure 2.2: Applying an RF magnetic pulse $\boldsymbol{B}_1$ to excite $\boldsymbol{M}$ from equilibrium and tip it towards the transverse plane. In doing that it starts to precess around $\boldsymbol{B}_0$ at the Larmor frequency ω_0.

(xy-component) of $\boldsymbol{M}$. This torque makes $\boldsymbol{M}$ precess around $\boldsymbol{B}_0$ at its Larmor frequency ω_0, which forms a rotating magnetic field in the transverse plane. From Faraday's law the rotating field induces an electromotive force (EMF) in specifically oriented *receiver coils*. This allows the distribution of $\boldsymbol{M}$ to be measured.

To excite $\boldsymbol{M}$ from its equilibrium state towards the transverse plane the precessional property of the spins is exploited. By applying a rotating radiofrequency (RF) magnetic pulse $\boldsymbol{B}_1$ along the transverse plane and orthogonal to $\boldsymbol{M}$, $\boldsymbol{B}_1$ exerts a torque on $\boldsymbol{M}$ thus tipping it towards the transverse plane. However, as $\boldsymbol{M}$ is getting tipped it starts to precess around $\boldsymbol{B}_0$ at the Larmor frequency ω_0. To maintain the orthogonal orientation of $\boldsymbol{B}_1$ and $\boldsymbol{M}$ and hence also the torque, the RF pulse $\boldsymbol{B}_1$ has to be tuned to the spins Larmor frequency ω_0. This process is illustrated in Figure 2.2. The strength and duration of $\boldsymbol{B}_1$ dictates how far $\boldsymbol{M}$ is tipped towards the transverse plane and is quantified by a *flip angle* which usually ranges from 0°–180°.

A *slice-selective* RF excitation is usually used to restrict the spatial extent of excited spins in an object to be imaged. After the slice-selective excitation the selectively excited spins have a spatial distribution of $\boldsymbol{M}$ precessing at ω_0 irrespective of their spatial position.

To be able to form an image of the object some kind of spatial encoding of the precession frequency is needed to be able to discriminate between spins in different spatial positions in the object. In MRI this is done by using linear gradients to encode the spatially varying frequency across the object.

2.1.2 Gradient Based Spatial Encoding

After applying a slice-selective excitation to an object we can envision the precessing magnetization vectors in the transverse plane of the slice as being a spatial map of tiny ‘magnetic oscillators’. Each oscillator at spatial position $\vec{r}$ possesses a time varying magnitude $f(\vec{r}, t)$, equivalent to the transverse component of $\boldsymbol{M}$, and a phase term $\phi(\vec{r}, t)$. The MR signal from each oscillator at position $\vec{r}$ and time t can thus be expressed as $f(\vec{r}, t)\, \mathrm{e}^{-i\phi(\vec{r},t)}$. If the receiver coil is uniformly sensitive over the excited object, then the received MR signal $s(t)$ as measured by the coil can be written as follows:

$$s(t) = \int f(\vec{r}, t)\, \mathrm{e}^{-i\phi(\vec{r},t)}\, \mathrm{d}\vec{r}. \tag{2.3}$$

Since frequency is equal to the time rate of change in phase and by using the general form of (2.2), $\phi(\vec{r}, t)$ can be rewritten as follows:

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\phi(\vec{r}, t) &= \omega(\vec{r}, t) = \gamma B(\vec{r}, t)\,, \\ \Rightarrow\ \phi(\vec{r}, t) &= \int_0^t \omega(\vec{r}, \tau)\, \mathrm{d}\tau = \gamma \int_0^t B(\vec{r}, \tau)\, \mathrm{d}\tau, \end{aligned} \tag{2.4}$$

where the initial phase is assumed to be zero. To encode the spatial locations of all the oscillators in the excited slice we can use linear *field gradients* in addition to the main magnetic field $\boldsymbol{B}_0$. This forms a spatially dependent frequency distribution for the slice that makes $B(\vec{r}, t)$ in (2.4) have the following form:

$$B(\vec{r}, t) = B_0 + \vec{G}(t) \cdot \vec{r},$$

where $B_0 = |\boldsymbol{B}_0|$ and $\vec{G}(t)$ are 2D or 3D linear gradients depending on the data acquisition. Using this in (2.4) we get the following:

$$\begin{aligned} \phi(\vec{r}, t) &= \gamma \int_0^t B_0 \, \mathrm{d}\tau + \left(\gamma \int_0^t \vec{G}(\tau) \, \mathrm{d}\tau \right) \cdot \vec{r} \\ &= \omega_0 t + 2\pi \vec{k}(t) \cdot \vec{r}, \end{aligned} \tag{2.5}$$

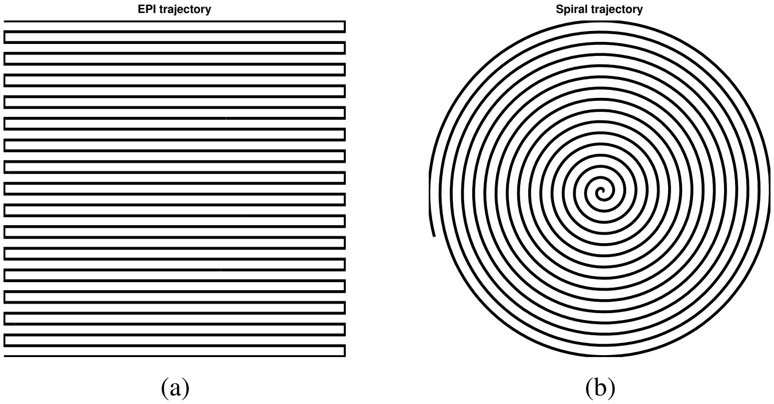
where $\vec{k}(t)$ is the *k-space trajectory* which has units of spatial frequency, *e.g.*, cycles/cm, and is defined by the following temporal integral of the gradients:

$$\vec{k}(t) \triangleq \frac{\gamma}{2\pi} \int_0^t \vec{G}(\tau) \, \mathrm{d}\tau \, .$$

The expression in (2.5) shows how $\vec{G}(t)$ (or $\vec{k}(t)$) enables control of the phase across the excited slice. Using this in (2.3) and demodulating by ω_0 yields the following signal model:

$$s(t) = \int f(\vec{r}, t) \, \mathrm{e}^{-i2\pi \left(\vec{k}(t) \cdot \vec{r} \right)} \, \mathrm{d}\vec{r}. \tag{2.6}$$

By convention the received MR signal $s(t)$ is sometimes called the *k-space* signal and represents a mapping from spatio-temporal space to k-space as determined by the $\vec{k}(t)$ trajectory. Figure 2.3 shows a couple of commonly used trajectories, the echo planar imaging (EPI) trajectory in Figure 2.3(a) that samples k-space on a Cartesian grid and the spiral trajectory in Figure 2.3(b).



(a) (b)

Figure 2.3: Two examples of k-space trajectories. (a) A echo planer imaging (EPI) trajectory; (b) A spiral trajectory.

2.1.3 Signal Relaxation: Return to Equilibrium

After excitation and sampling of the MR signal, the spins are in an elevated energy state. Eventually all the spins return to their thermal equilibrium state along the longitudinal axis. In a perfectly homogeneous $|\boldsymbol{B}_0|$, this process is driven by interaction of the spins both with its surrounding spins and environment. The relaxation is characterized by an exponential *recovery* and *decay* of its longitudinal and transverse component respectively.

The longitudinal recovery is characterized by the time constant T_1 and the transverse decay by T_2. Both time constants are tissue dependent and T_1 is also field strength dependent [1]. For instance T_1 in the brain at 3.0T is $\sim$1330ms and $\sim$830ms in gray and white matter respectively [2], but $\sim$1200ms and $\sim$750ms at 1.5T [3]. The transverse decay constant T_2 is much shorter, generally around $\sim$95ms in gray matter and $\sim$70ms in white matter [4].

Both relaxation constants dictate the image contrast in MR imaging. For the simple case of a spatially uniform T_1 and T_2 and a 90° RF pulse, their relationship to $s(t)$ can be approximately expressed by the following:

$$s(\mathrm{TE}) \sim \mathrm{e}^{-\mathrm{TE}/T_2}\left(1-\mathrm{e}^{-\mathrm{TR}/T_1}\right),$$

where the *echo time* TE is the time instance when $\vec{k}(t) = \mathbf{0}$, *i.e.*, when the center of k-space gets sampled, and the *repetition time* TR is the time difference between consecutive RF excitations. From this relationship one can see how T_1 and T_2 weighting of the acquired data can be manipulated by changing when the center of k-space gets sampled and the time between consecutive excitations. For example, to get a T_2-weighted image one would choose $\mathrm{TR} \gg T_1$ and $\mathrm{TE} \approx T_2$ and for a T_1-weighted image one would choose $\mathrm{TE} \approx 0$ and $\mathrm{TR} \approx T_1$.

Since $T_2(\vec{r})$ is the spatially varying time constant for the decay of the transverse component of $\boldsymbol{M}$, $f(\vec{r}, t)$ can be written as follows [1]:

$$f(\vec{r}, t) = f(\vec{r})\,\mathrm{e}^{-t/T_2(\vec{r})} = f(\vec{r})\,\mathrm{e}^{-tR_2(\vec{r})}\,, \tag{2.7}$$

where $f(\vec{r})$ is the instantaneous transverse magnetization after excitation and $R_2(\vec{r}) \triangleq 1/T_2(\vec{r})$ is the R_2 relaxation rate map. This form of $f(\vec{r}, t)$ separates the temporally dependent relaxation of the magnetization from the initial magnetization generated in the object. By replacing $f(\vec{r}, t)$ in (2.6) with the formulation in (2.7), the MR signal equation can now be written as follows:

$$s(t) = \int f(\vec{r})\,\mathrm{e}^{-tR_2(\vec{r})}\,\mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}\right)}\,\mathrm{d}\vec{r}. \tag{2.8}$$

2.1.4 Macro- and Microscopic Off-Resonance Effects

Thus far we have assumed a perfectly homogeneous $|\boldsymbol{B}_0|$, so that after excitation the object only has spins resonating at the Larmor frequency ω_0. In practice that is not the case due to two primary sources of inhomogeneity. The first source is inhomogeneity in the main magnetic field and the second source is tissue dependent susceptibility variations. Susceptibility differences introduce a spatially varying magnetic field across the object. This causes spins in the object to resonate at different frequencies based on the local field strength $|\boldsymbol{B}|$ experienced by the spins as dictated by (2.2). This spatially dependent *off-resonance* leads to a

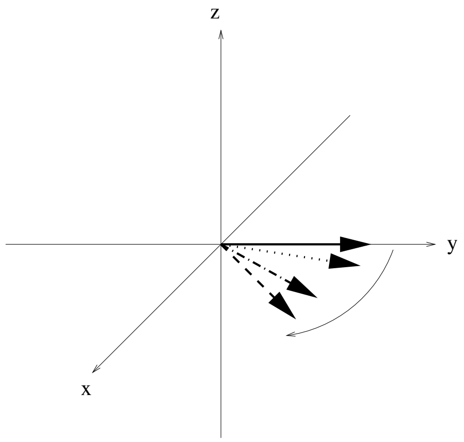


Figure 2.4: Spins dephasing leading to eventual loss in magnetization signal strength.

loss in magnetization strength due to destructive interference among the spins as shown in Figure 2.4, which causes signal voids or distortions in the MR images. The off-resonance can be categorized into *macroscopic* and *microscopic* effects that depend on the spatial scale of their field disturbances versus the spatial resolution of the acquired image [5].

The macroscopic off-resonance is due to large field distortions that act across typical voxel sizes [5]. The two main sources of these effects are $\boldsymbol{B}_0$ inhomogeneity and bulk magnetic susceptibility differences from air/tissue interfaces such as in areas close to the sinuses or lungs. This generally causes a bulk shift in spin resonance and can be modeled as a spatially dependent shift in phase in the MR signal equation. By including the effects of macroscopic off-resonance in $f(\vec{r}, t)$ we get the following:

$$f(\vec{r}, t) \triangleq f(\vec{r})\, \mathrm{e}^{-tR_2(\vec{r})}\, \mathrm{e}^{-it\omega(\vec{r})} = f(\vec{r})\, \mathrm{e}^{-t(R_2(\vec{r}) + i\omega(\vec{r}))},$$

where the *field map* $\omega(\vec{r})$ is the spatial map of the macroscopic off-resonance.

The microscopic off-resonance is due to localized field distortions that act within typical voxel sizes [5]. These distortions are mostly driven by susceptibility differences on the molecular level that causes localized spin dephasing. This intravoxel *susceptibility weighting* increases the effective relaxation rate of the voxel, defined as $R_2^* \triangleq 1/T_2^*$ to differentiate it from R_2 relaxation. The T_2^*-weighting is conventionally modeled as an additive relaxation R_2' to R_2 as follows:

$$R_2^* \triangleq R_2 + R_2'.$$

Even though this model is somewhat simplistic it does work well in areas with low macroscopic off-resonance. For areas with high macroscopic off-resonance there exist more complete models as will be discussed in Section 2.4.1.

By including both the macro- and microscopic effects, the MR signal equation in (2.8) can be written as follows:

$$\begin{aligned} s(t; z(\vec{r}), f(\vec{r})) &\triangleq \int f(\vec{r})\, \mathrm{e}^{-t\left(R_2^*(\vec{r})+i\omega(\vec{r})\right)}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}\right)}\, \mathrm{d}\vec{r}, \\ &= \int f(\vec{r})\, \mathrm{e}^{-tz(\vec{r})}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}\right)}\, \mathrm{d}\vec{r}, \end{aligned} \tag{2.9}$$

where $f(\vec{r})\, \mathrm{e}^{-tR_2^*(\vec{r})}$ is the T_2^*-weighted image. This form of the MR signal equation is generally used when including bulk off-resonance and susceptibility weighted signal relaxation. In some cases the relaxation is not included in the model but is instead included in the magnetization term $f(\vec{r})$ which then becomes a *weighted magnetization*, denoted by $f_{\mathrm{TE}}(\vec{r})$ due to its strong link to the echo time TE. This version of the signal equation can then be written as follows:

$$s(t; z_{\mathrm{I}}(\vec{r}), f_{\mathrm{TE}}(\vec{r})) \triangleq \int f_{\mathrm{TE}}(\vec{r})\, \mathrm{e}^{-itz_{\mathrm{I}}(\vec{r})}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}\right)}\, \mathrm{d}\vec{r}, \tag{2.10}$$

where $z_{\mathrm{I}}(\vec{r}) = \omega(\vec{r})$. This form of the signal equation is more commonly used (often implicitly) in the conventional reconstruction methods as shown in the next section.

2.2 MR Image Reconstruction

In MRI the discrete, complex-valued and T_2^*-weighted acquired MR data $\boldsymbol{y} = [y_1, \ldots, y_M]$ can be modeled using the MR signal equation in (2.9) or (2.10), as follows:

$$y_m = s(t_m; \cdot) + \epsilon_m, \quad m = 1, \ldots, M, \tag{2.11}$$

where $s(t_m; \cdot)$ is a temporal sample of the MR signal equation and ϵ_m is complex-valued independent and identically distributed (iid) Gaussian noise [5, 6]. This equation is sometimes written in a vector format as follows:

$$\boldsymbol{y} = \boldsymbol{s}(\cdot) + \boldsymbol{\epsilon}, \tag{2.12}$$

where all the vectors are column vectors of length M with elements given by (2.11).

The objective of MRI image reconstruction is then to use (2.12) to either reconstruct the magnetization image $f(\vec{r})$ or the weighted image $f_{\mathrm{TE}}(\vec{r})$. For simplicity the following discussion will assume $f_{\mathrm{TE}}(\vec{r})$ reconstruction since it is more commonly used and more studied of the two reconstruction problems. However, the discussion can easily be applied to reconstructing $f(\vec{r})$, with minor differences that are discussed at the end of this section.

It should be noted that reconstructing $f_{\mathrm{TE}}(\vec{r})$ requires a prior estimate of the field map (or to assume $\omega(\vec{r}) = 0$). The field map is conventionally estimated from the phase difference of two scans, where one is acquired with a slight delay relative to the first one [7, 8]. For a 3T MRI scanner this delay is usually 2-3ms to minimize the chance of phase wrapping.

Next, we will review two different methods of reconstructing $f_{\mathrm{TE}}(\vec{r})$. The first one reconstructs $f_{\mathrm{TE}}(\vec{r})$ by linearly mapping the acquired k-space data back to image space, while the second one is an iterative method that involves minimizing a squared error term of $\boldsymbol{y}$ and $\boldsymbol{s}(z_{\mathrm{I}}(\vec{r})\,, f_{\mathrm{TE}}(\vec{r}))$. Both methods offer some form of noise suppression and fast implementations with and without field map correction. The non-iterative method is slightly faster than the iterative method but its solution becomes more involved for k-space data acquired on a non-Cartesian grid, which is less of an issue for the iterative method.

2.2.1 Non-iterative MRI Reconstructions

Ignoring off-resonance correction, the most commonly used class of non-iterative discrete reconstruction methods is the *conjugate phase* (CP) reconstruction, described by the following weighted correlation method [9]:

$$\hat{f}_{\mathrm{TE}}(\vec{r}) \approx \sum_{m=1}^{M} y_m \mathrm{D}(t_m)\, \mathrm{e}^{i2\pi\left(\vec{k}(t_m)\cdot\vec{r}\right)}\,, \tag{2.13}$$

where $\hat{f}_{\mathrm{TE}}(\vec{r})$ is the reconstructed image, y_m is the acquired k-space data (sometimes it is low-pass filtered to reduce noise) and $\mathrm{D}(t_m)$ is the weight function. The weight function is called the sample *density compensation function* (DCF) and as the name implies is supposed to compensate for nonuniform sampling in k-space when using non-Cartesian k-space trajectories to acquire the data. This is a critical portion of the reconstruction and it is important it be calculated efficiently and accurately.

Finding the DCF for the k-space trajectory that is used to acquire the data is nontrivial and there have been many iterative and non-iterative methods developed for that purpose. These include using Voronoi areas [10], the sample Jacobian determinant [11, 12] or least squares optimization, which is either solved using SVD [13] or analytically as shown in equation (34) in [14].

Calculating (2.13) for Cartesian k-space trajectories involves a 2D FFT operation which is fast. For non-Cartesian k-space data the conventional fast reconstruction is based on *gridding*, where the k-space data is interpolated to an oversampled Cartesian grid followed by a 2D FFT. The gridding process usually uses optimized Kaiser-Bessel interpolation kernels [15] and an apodization/de-apodization to correct for aliasing. The gridding process can be optimized further for speed such as by choosing a “wise” oversampling factor or Kaiser-Bessel parameters [16].

When including off-resonance correction, the CP reconstruction in (2.13) is modified as follows:

$$\hat{f}_{\mathrm{TE}}(\vec{r}) \approx \sum_{m=1}^{M} y_m \mathrm{D}(t_m)\, \mathrm{e}^{it_m z_{\mathrm{I}}(\vec{r})}\, \mathrm{e}^{i2\pi\left(\vec{k}(t_m)\cdot\vec{r}\right)}\,. \tag{2.14}$$

Due to the spatio-temporal dependence of $\mathrm{e}^{it_m z_{\mathrm{I}}(\vec{r})}$, (2.14) cannot be calculated using a 2D FFT. To speed up the computation, a fast implementation of this reconstruction can use either temporal or frequency segmentation [17, 18]. These methods break up the off-resonance correction term into L time or frequency segments and its associated interpolation coefficients as follows:

$$\mathrm{e}^{it_m z_{\mathrm{I}}(\vec{r}_n)} \approx \sum_{l=1}^{L} a_{ml} b_{ln}, \tag{2.15}$$

where a_{ml} and b_{ln} are basis functions or interpolation coefficients based on the choice of segmentation. By including this approximation into (2.14) the reconstruction becomes a sum of L CP reconstructions. This idea has also been used in iterative reconstruction algorithms using min-max interpolators [19, 20].

2.2.2 Iterative MRI Reconstruction

The reconstruction problem can be restated in the form of finding the maximum likelihood estimator (MLE) of $f_{\mathrm{TE}}(\vec{r})$ using an iterative optimization algorithm. From (2.12) $\boldsymbol{y}$ is a complex-valued iid Gaussian random variable with mean $\boldsymbol{s}(z_{\mathrm{I}}(\vec{r})\,, f_{\mathrm{TE}}(\vec{r}))$, thus the MLE is equal to the least squares estimator (LSE). However, finding the MLE of the continuous-space map $f_{\mathrm{TE}}(\vec{r})$ from the discrete MR data $\boldsymbol{y}$ using (2.12) is an ill-posed inverse problem. The ill-posedness is due to there being an infinite amount of solutions in the continuous solution space that fit the finite dimensional measurement vector $\boldsymbol{y}$.

To make the problem more tractable we parametrize $f_{\mathrm{TE}}(\vec{r})$ and $z_{\mathrm{I}}(\vec{r})$ using a voxel basis function $\mathrm{b}(\cdot)$ as follows:

$$\begin{aligned} f_{\mathrm{TE}}(\vec{r}) &\triangleq \sum_{n=1}^{N} f_{\mathrm{TE}}(\vec{r}_n)\,\mathrm{b}(\vec{r}-\vec{r}_n)\,, \\ z_{\mathrm{I}}(\vec{r}) &\triangleq \sum_{n=1}^{N} z_{\mathrm{I}}(\vec{r}_n)\,\mathrm{b}(\vec{r}-\vec{r}_n)\,, \end{aligned} \tag{2.16}$$

where $\vec{r}_n$ is the spatial coordinate in physical units, such as mm or cm, of pixel n in lexicographical order. There are multiple ways of choosing $\mathrm{b}(\cdot)$, but due to convenience and the eventual displaying of the image on a screen the $\mathrm{rect}(\cdot)$ function is commonly used [19, 20].

Using (2.16) with $\mathrm{b}(\cdot) \triangleq \mathrm{rect}(\cdot)$ in (2.10) we get the following:

$$s(t; \boldsymbol{z}_{\mathrm{I}}, \boldsymbol{f}_{\mathrm{TE}}) \triangleq \Phi(\vec{k}(t)) \sum_{n=1}^{N} f_{\mathrm{TE}}(\vec{r}_n)\, \mathrm{e}^{-it z_{\mathrm{I}}(\vec{r}_n)}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)}\,, \tag{2.17}$$

$$\boldsymbol{f}_{\mathrm{TE}} \triangleq [f_{\mathrm{TE}}(\vec{r}_1)\,, \ldots, f_{\mathrm{TE}}(\vec{r}_N)]^{\mathrm{T}}\,,\ \boldsymbol{z}_{\mathrm{I}} \triangleq [z_{\mathrm{I}}(\vec{r}_1)\,, \ldots, z_{\mathrm{I}}(\vec{r}_N)]^{\mathrm{T}}\,,$$

where $s(t; \boldsymbol{z}_{\mathrm{I}}, \boldsymbol{f}_{\mathrm{TE}})$ is the discrete-space MR signal equation and $\Phi(\vec{k}(t))$ is the Fourier transform of $\mathrm{rect}(\cdot)$. This form of the signal equation can now be used with (2.12) to find the MLE of the discrete image $\boldsymbol{f}_{\mathrm{TE}}$.

Note that (2.17) can be written in terms of $\boldsymbol{f}_{\mathrm{TE}}$ as a matrix-vector multiplication as follows:

$$\boldsymbol{s}(\boldsymbol{z}_{\mathrm{I}}, \boldsymbol{f}_{\mathrm{TE}}) = \boldsymbol{A}\boldsymbol{f}_{\mathrm{TE}}, \tag{2.18}$$

where the elements of the *system matrix* $\boldsymbol{A}$ are given as follows:

$$[\boldsymbol{A}]_{mn} = \Phi(\vec{k}(t_m))\ \mathrm{e}^{-it_m z_{\mathrm{I}}(\vec{r}_n)}\ \mathrm{e}^{-i2\pi\left(\vec{k}(t_m)\cdot\vec{r}_n\right)}\,.$$

Using (2.18), $\boldsymbol{f}_{\mathrm{TE}}$ can be reconstructed by minimizing a regularized log-likelihood function as follows:

$$
\begin{aligned}
\Psi(\boldsymbol{f}_{\mathrm{TE}}) &= \frac{1}{2}\left\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{f}_{\mathrm{TE}}\right\|^2 + \mathrm{R}(\boldsymbol{f}_{\mathrm{TE}})\,, \\
\hat{\boldsymbol{f}}_{\mathrm{TE}} &= \underset{\boldsymbol{f}_{\mathrm{TE}}}{\arg\min}\, \Psi(\boldsymbol{f}_{\mathrm{TE}})\,,
\end{aligned}
\tag{2.19}
$$

where the regularization function $\mathrm{R}(\cdot)$ is used to control the tradeoff of bias and noise in the reconstructed image.

Excluding regularization in (2.19) the reconstruction is a quadratic least squares problem that can be solved efficiently using an iterative algorithm such as conjugate gradient (CG) [21]. However, this exclusion can make the solution sensitive to errors from noise in $\boldsymbol{y}$ if $\boldsymbol{A}$ is ill-conditioned. However, by choosing a quadratic regularization function, such as a *roughness penalty*, allows for CG to be still used to solve (2.19). This is at a cost in reduced spatial resolution of the reconstructed image, which is minor relative to the overall increase in stability of the reconstruction algorithm.

The degree of regularization is controlled through a *regularization parameter* that dictates how much bias is imposed on the final solution. There are many methods available that find the regularization parameter in a sensible manner [22], such as cross validation, but those can be computationally intensive. For roughness penalties, local point spread functions [23] is a method that has been used to investigate the local resolution properties of the reconstructed images. This information can then be used to design regularizers for which the reconstructed images have a predetermined resolution that is spatially uniform [24]. These methods can also be computed in an efficient manner using FFT.

To reduce memory usage $\boldsymbol{A}$ is never explicitly formed, but rather a software object is used to represent it[1]. The CG algorithm involves both a forward projection $\boldsymbol{A}\boldsymbol{f}_{\mathrm{TE}}$ and a back projection $\boldsymbol{A}'\boldsymbol{y}$ that, when excluding the field map correction, can be implemented efficiently using FFT for Cartesian k-space data or NUFFT for noncartesian k-space data [25]. Including field map correction in either projection cannot be implemented in this fast

---

[1] Software available at http://www.eecs.umich.edu/~fessler/

manner. However, the off-resonance term can be approximated by temporal segmentation as mentioned in Section 2.2.1 (see equation (2.15)) [19, 20]. This renders the forward and back projections to a weighted sum of L FFT or NUFFT operations, which greatly decreases computation time.

2.2.3 Reconstructing $f(\vec{r})$

As was noted earlier in the section, reconstructing $f(\vec{r})$ is very similar to reconstructing $f_{\mathrm{TE}}(\vec{r})$. There are some minor additional things to have in mind such as the inclusion of an R_2^* relaxation map in the MR signal equation and consequently in the system matrix. This implies an estimate of the R_2^* map is needed. A simple way of estimating these maps is to reconstruct multiple $\boldsymbol{f}_{\mathrm{TE}}$ images with a range of different TE values followed by fitting a mono-exponential to each voxel magnitude value using all the acquired $\boldsymbol{f}_{\mathrm{TE}}$ as is discussed in Section 2.4.1.

Including a spatially varying R_2^* map in the CP reconstructions is problematic and is usually performed using a post processing method, such as by de-blurring the reconstructed $\boldsymbol{f}_{\mathrm{TE}}$ [26, 27]. However, for the iterative reconstruction this is not a problem, since we can still use the temporal/frequency segmentation methods. The only difference is that the exponent is now a complex valued rate map, which can be dealt with in a similar way as if the rate map were real valued [19].

2.3 Functional MRI

Functional magnetic resonance imaging (fMRI) is an MR imaging method to study neuronal activity in the brain or spinal cord of humans or other animals. In an fMRI study, a subject lies in the scanner and is asked to perform a prespecified task that will activate neurons associated with that task. At the same time, multiple MRI images are collected of the neuronal population of interest either in the brain or spinal cord. Since MRI images are

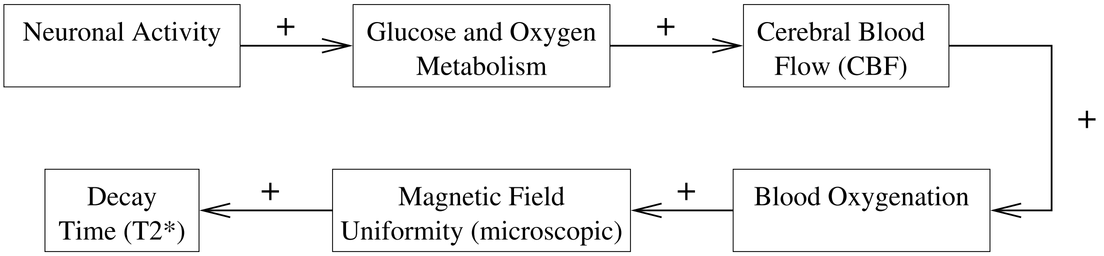


Figure 2.5: The BOLD contrast, where '+' indicates a causal increase from one block to the next.

not sensitive to the local field potential from neurons other indirect measures are used such as the associated changes in local hemodynamics. Of those changes the blood oxygen level dependent (BOLD) contrast mechanism is the most popular for detecting brain activation.

2.3.1 BOLD: T_2^*-Weighted fMRI

BOLD contrast comes from localized magnetic field distortions due to changes in cerebral hemodynamics, such as blood flow, blood volume and blood oxygenation. These local microscopic field distortions change the intra-voxel dephasing of spins that make up the voxel magnetization. This affects the R_2^* relaxation of the voxel [28, 29] and hence the T_2^*-weighted magnitude images $f_{\mathrm{TE}}(\vec{r})$. The temporal changes of the voxel magnitude values in brain areas involved in the task have been shown to correlate well with the temporal fluctuations of the task [30–32].

Figure 2.5 shows the causal time line of the BOLD contrast [33]. During an increase in local neuronal activity, cerebral blood flow (CBF) increases also around that area thus causing a drop in the concentration of deoxygenated hemoglobin. This increases uniformity of the local magnetic field due to less concentration of diamagnetic deoxygenated hemoglobin. That decreases locally the intravoxel dephasing that is modeled by the R_2^* relaxation rate. The temporal fluctuations associated with the neuronal activity show up in the voxel time series of either the T_2^*-weighted magnitude images or the R_2^* spatial maps.

Figure 2.6 shows the temporal characteristics of the hemodynamic response associated

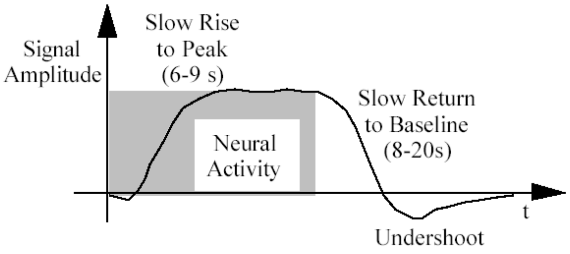


Figure 2.6: The temporal characteristics of the hemodynamic response during an fMRI study associated with the BOLD contrast. The response is shown for a sustained period of task-related neuronal activity (gray box) along with the rise-time and time-to-baseline for the hemodynamics.

with the BOLD contrast. The hemodynamics respond slowly to the sustained neuronal activity. This allows the MR image frame rate to be on the order of 1-2 seconds without having to worry about aliasing of the hemodynamic response. For example a typical whole brain acquisition on a 3T scanner with in-plane resolution (xy-plane) of 3mm and slice resolution of 3-5mm (z-plane) has a total acquisition time of 1.5-2 seconds.

To get the needed BOLD contrast in T_2^*-weighted magnitude images, TE has to be chosen large enough to allow for the susceptibility weighting to build up. This can be seen from the relationship of $s(t)$ to R_2^* relaxation which, if we assume a spatially nonvarying R_2^* relaxation, is the same as for R_2 shown in Section 2.1.3:

$$s(t) \sim \mathrm{e}^{-\mathrm{TE} \cdot R_2^*} .$$

This relationship indicates a strong connection between the amount of susceptibility weighting and the echo time (TE). In fact, the largest difference in T_2^*-weighting for two time frames in an fMRI experiment that are reconstructed using (2.10) is when $\mathrm{TE} = T_2^*$ [5]. However, different tissue types such as gray matter and white matter in the brain have different T_2^* values, so choosing a single TE is always going to be a compromise. There are also limitations in how large TE can be chosen due to bulk magnetic susceptibility related artifacts and a decrease in SNR. These issues dictate the chosen TE value used for any given fMRI study. With T_2^* values being around 40-60ms for gray and white matter a reasonable choice for TE is around 30-35ms.

The conventional method of detecting brain activation for fMRI is to collect multiple T_2^*-weighted k-space data and reconstruct all T_2^*-weighted time frames. This is then followed by a statistical detection of task-related signal changes in the time series of each voxel of the magnitude images. The drawback with using the T_2^*-weighted magnitude images is that they give only a qualitative measure of the BOLD contrast but are not quantitative. BOLD contrast is more interpretable using R_2^* relaxation maps [34, 35] since they are quantifiable, *i.e.*, have units of 1/sec, and can be directly related to hemodynamic quantities such as blood flow and blood volume. They also exhibit a larger functional contrast, although with some loss of spatial specificity as observed using standard reconstruction methods [36, 37] to be reviewed in Section 2.4.1.

2.3.2 Non-Idealities in BOLD fMRI Imaging

BOLD fMRI imaging usually involves scans where MRI images are collected continuously for tens of minutes. During this time there are some temporal low frequency fluctuations in the off-resonance, both from the hardware and also related to physiological processes. Additionally, since the images are susceptibility weighted, there are also problems from bulk susceptibility related signal loss. These problems will be discussed further in this subsection.

Field Map Temporal Instability

While acquiring data for an fMRI study the main magnetic field $\boldsymbol{B}_0$ and hence the spatially global spin resonance can drift, effectively causing drift in the field map. This drift is usually ignored in the conventional "static" field map corrected image reconstruction [9, 17, 19, 20, 38] where the field map is collected at the start of the study and then used to reconstruct all subsequent image time frames. The main cause of the drift in $\boldsymbol{B}_0$ is heating of the passive shim when the scanner runs for a long time [39]. This heating changes the magnetic properties of the shim elements which causes a near spatially uniform drift in $\boldsymbol{B}_0$. If this drift in the field map is ignored during the reconstruction, it can cause a global drift in the voxel time series of the T_2^*-weighted magnitude images. This is a confound which has to be dealt with during the post-processing statistical detection stage of the functional data.

There have been several methods proposed to correct for this drift both, online and offline. The simplest method is to use a 1D *navigator echo* that measures phase changes at the center of k-space while the fMRI data is collected [40]. This method measures the spatially averaged phase offset in image space, which can be corrected online while acquiring the k-space data or offline during post-processing. However, this method does not take into account any possible spatial variation in the drift.

A more appropriate method would be correct for this effect during the image reconstruction via a nonlinear joint reconstruction of the field map and the T_2^*-weighted image [41–45]. Existing methods of this type can correct for spatially varying effects in the drift but they all rely on longer readouts than conventional fMRI acquisition schemes demand. That impacts the temporal or spatial resolution of the fMRI data. Also, nonlinear reconstruction is more time consuming than conventional T_2^*-weighted reconstruction algorithms.

It is more more common to ignore this effect during acquisition or reconstruction and to deal with this artifact during post-processing of the time series. The drift is most often modeled as a 2nd or 3rd degree polynomial regressor during the statistical detection of the neuronal activation. However, this regressor could mask any true physiologically based baseline signal drift present in the time series. This includes gradual changes in baseline blood flow, blood volume and oxygen consumption that need to be taken into consideration to get quantifiable activation changes.

Bulk Susceptibility Artifacts

Even for an ideal main magnetic field there can still be off-resonance differences in the object to be imaged due to bulk magnetic susceptibility differences in the object. For brain imaging the air-tissue interfaces are the primary source of such spatially varying susceptibility differences. Since BOLD fMRI is susceptibility weighted it is very sensitive to these bulk magnetic susceptibility related artifacts.

Most currently used reconstruction methods account for off-resonance by using field maps. The field map is a discrete representation of the true off-resonance in continuous space and typically has a $\mathrm{rect}(\cdot)$ bases function for the continuous to discrete mapping. Thus, the field map represents the *average* off-resonance within a voxel, induced by the spatially varying and continuous gradients from the bulk susceptibility differences. There has been some work in including first order linear gradient correction in both iterative [46, 47] and noniterative [12] reconstruction methods, which should show some benefits over using only the average off-resonance correction.

Physiological Related Artifacts

In addition to noise introduced when acquiring k-space there is also additional noise from physiological processes that introduce temporally correlated artifacts in the voxel time series. The effects of these processes manifest themselves differently, based on the how the k-space data is acquired. For acquisitions that sample the fMRI data over multiple readouts, or *multi-shot* acquisitions, the effect induces ghosting and intensity and phase fluctuations due to shot-to-shot discrepancies. For acquisitions that acquire the data in one readout, or *single-shot* acquisitions, the effect causes a more localized intensity variations and global phase fluctuations in the time frames. These diminished effects relative to the multi-shot acquisitions is one of the primary reasons why single-shot is more popular for fMRI data. The main sources of physiological noise are respiration and the cardiac cycle.

Of the two physiological artifact sources, the respiratory effect is a more spatially global source in an fMRI study than the more local cardiac related effects. Respiration generates a small magnetic field shift due to the gross magnetic susceptibility changes that happen when the lungs expand and contract during breathing. This causes temporally dependent fluctuations in the magnetic field across the brain, where the fluctuations are higher at the inferior part of the brain than the superior. The cardiac related effects are more localized than the respiratory effects. They mainly introduce local image magnitude changes when fully relaxed spins in the blood flow into the imaging slice. This causes magnitude fluctuations that are correlated with the cardiac cycle. Another effect is circulation of cerebrospinal fluid (CSF) around the brain due to the cardiac pulsation of the brain.

These effects are conventionally corrected during post-processing of the voxel time series. One such method is RETROICOR [48] or RETROKCOR [49] where the temporal fluctuations are modeled using a low-order Fourier series. The series is formed using the phase of the respiratory and cardiac cycles that are measured at the same time as the fMRI data. It is then used as a regressor to model the physiological artifacts that are superimposed on the actual neuronal activation response. Other corrections include exploratory methods such as PCA or ICA which tend to separate these effects into a component that can be included in the regression model.

2.4 R_2^* Estimation

This section discusses two different types of methods for R_2^* estimation. First we will discuss conventional methods of estimating R_2^* that are based on multiple fully sampled acquisitions that have different TE values followed by a fit to a signal relaxation model. Second we will discuss methods that model more accurately the R_2^* effect in the signal model, requiring nonlinear iterative algorithms to reconstruct R_2^* maps. Both descriptions address the impact of off-resonance correction.

2.4.1 Conventional R_2^* Estimation

Conventional R_2^* estimation uses the following discrete signal equation:

$$s(t; f_{\mathrm{TE}_l}(\vec{r})) = \sum_{n=1}^{N} f_{\mathrm{TE}_l}(\vec{r}_n)\, \mathrm{e}^{-t z_{\mathrm{I}}(\vec{r}_n)}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)}, \; l = 1, \ldots, L, \tag{2.20}$$

where L is the number of acquisitions with different echo times $\mathrm{TE}_1, \ldots, \mathrm{TE}_L$. Using this model to reconstruct R_2^*, one starts by reconstructing $\hat{\boldsymbol{f}}_{\mathrm{TE}_l}$ assuming $\mathrm{e}^{-t z_{\mathrm{I}}(\vec{r}_n)} = \mathrm{e}^{-\mathrm{TE}_l z_{\mathrm{I}}(\vec{r}_n)}$ for all l followed by a voxel-wise fit to the chosen R_2^* relaxation model.

The simplest of such models is the mono-exponential model used in most fMRI studies with R_2^* maps [35–37, 50–52]. Even though this is the simplest model, it has been shown to adequately approximate the signal relaxation after fitting it to acquired MRI magnitude images. This method has problems in areas where there are large spatial variations in susceptibility and in voxels composed of multiple tissue types that have highly varying R_2 relaxation curves.

To deal with the multi-tissue-type problem for some voxels, one can use multi-exponential fits, *e.g.*, a bi-exponential model [53] to fit two of the largest contributions of the signal relaxation. For areas with large spatial variations of susceptibility more complex models have been used such as an exponential with a quadratic exponent [54] or logspline density functions [55].

Most of these studies did not use field map correction in the image reconstruction algorithms. This lack of field map correction could accentuate the model mismatch found when only using a mono-exponential model. Also, to improve the results from the mono-exponential model one could also include the reconstructed R_2^* in (2.20) to correct for R_2^* relaxation during the readout, *i.e.*, instead of using $\mathrm{e}^{-t z_{\mathrm{I}}(\vec{r}_n)}$ use $\mathrm{e}^{-t z(\vec{r}_n)}$ with $t = 0, \ldots, t_{acq}$. This would then be followed by a refined relaxation fit from the resulting magnitude images. This method would initialize the R_2^* map to be $z_{\mathrm{R}}(\vec{r}_n) = 0$ for all n.

However the model chosen for R_2^* relaxation still has two major flaws with this way of estimating R_2^*. The first is that to perform a voxel-wise fit to the signal decay, multiple acquisitions are needed for each slice or 3D volume that is collected. This puts restrictions either on the temporal or spatial resolution of the fMRI data. Secondly is the conventional assumption of instantaneous decay at TE. This simplification ignores the relaxation during the k-space data readout causing model mismatch between the collected signal and does not include any estimation of the R_2^* relaxation during the readout.

2.4.2 Iterative R_2^* Estimation

In this section we use the more accurate discrete signal model given as follows:

$$s(t; z(\vec{r}), f(\vec{r})) = \Phi(\vec{k}(t)) \sum_{n=1}^{N} f(\vec{r}_n)\, \mathrm{e}^{-t z(\vec{r}_n)}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)}, \tag{2.21}$$

with,

$$z(\vec{r}_n) = z_{\mathrm{R}}(\vec{r}_n) + i z_{\mathrm{I}}(\vec{r}_n), \quad z_{\mathrm{R}}(\vec{r}_n) \triangleq R_2^*(\vec{r}_n), \quad z_{\mathrm{I}}(\vec{r}_n) \triangleq \omega(\vec{r}_n).$$

This form of the MR signal equation assumes a mono-exponential signal relaxation model. This formulation combines the effects of R_2^* and ω into a single complex valued rate map $z(\vec{r})$.

To estimate R_2^* one has to resort to iterative methods and, due to temporal fluctuations in $\boldsymbol{z}_{\mathrm{I}}$ during the acquisition of fMRI data, $\boldsymbol{z}_{\mathrm{I}}$ also needs to be estimated. There are a few previously proposed iterative methods that have solved this using (2.21) in an iterative reconstruction algorithm [41–43]. All these algorithms reconstruct $z(\vec{r})$ and [42, 43] also reconstruct $f(\vec{r})$ in addition to $z(\vec{r})$, which has to be otherwise estimated previously. Another algorithm [45] reconstructs R_2^*-weighted images $f_{\mathrm{TE}}(\vec{r}_n)$ along with the field map $z_{\mathrm{I}}(\vec{r}_n)$ for all time frames in an fMRI study. However, all these methods are based on nonlinear reconstruction algorithms that try to solve a nonconvex problem that is much more complex than the methods presented in Section 2.4.1. This has detracted from mainstream use of these algorithms. This thesis provides alternatives that overcome this drawback.

Chapter 3

Fast Joint Reconstruction of Dynamic R_2^* and Field Maps in functional MRI[1]

3.1 Introduction

Here, we propose a method for fast joint reconstruction of R_2^* and field maps from MR data collected during an fMRI study. The method is based on making a linear approximation to the signal equation in (2.9) in terms of the temporal changes in the complex valued rate map $\boldsymbol{z}$ relative to a predefined reference map. The reference map is chosen such that any approximation errors are minimal. This approximation is then used to rewrite the signal equation as a linear mapping of the unknown rate map $\boldsymbol{z}$. This allows us to solve the reconstruction using a penalized least squares estimator, where the minimum is calculated using fast iterative algorithms such as conjugate gradient (CG).

To further minimize any approximation errors, the method uses the first estimate from the linearized model as the reference map and estimates the rate map again. This *refinement* process is repeated multiple times for the current time frame in the fMRI time series until the estimated rate maps have converged. This allows the algorithm to correct for any possible errors introduced from the linear approximation. Once the refinements have converged, one can start to estimate the rate map for the next frame. If the reference map is

---

[1] This chapter is based on [56–58].

chosen close enough to the true rate map there are only 2-3 refinements needed until the reconstructed $\boldsymbol{z}$ has converged.

The structure of this chapter will be as follows. We start by introducing a mono-exponential relaxation model for R_2^* enabling us to combine R_2^* and field map into a single complex-valued map (Section 3.2). We then linearize the MR signal equation relative to the temporal dynamics of this map and employ fast regularized iterative algorithms to reconstruct the complex-valued map (Section 3.3). We then simulate fMRI data for analysis and collect *in vivo* fMRI data for qualitative purposes (Section 3.6 and 3.7). For simplicity, motion is excluded from the reconstruction algorithm and the simulations.

3.2 Joint Reconstruction of R_2^* and Field Map

In fMRI the received complex-valued and discrete T_2^*-weighted MR data $\boldsymbol{y} = [y_1, \ldots, y_M]$, including effects of mono-exponential relaxation and off-resonance, can be modeled as follows [34]:

$$y_m = s(t_m) + \epsilon_m, \quad m = 1, \ldots, M, \tag{3.1}$$

$$s(t) = \int f(\vec{r})\, \mathrm{e}^{-tz(\vec{r})}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}\right)}\, \mathrm{d}\vec{r}, \tag{3.2}$$

where ϵ_m denotes complex-valued independent and identically distributed (iid) Gaussian noise [5] and $s(t_m)$ is a sample of the MR signal equation defined in (3.2). In (3.2), $\vec{r}$ is a 2D or 3D image space coordinate, $f(\vec{r})$ is the magnetization of the object directly after RF excitation, $\vec{k}(t)$ is the k-space trajectory used to acquire the MR data and $z(\vec{r})$ is a complex-valued spatial map:

$$z(\vec{r}) = z_{\mathrm{R}}(\vec{r}) + iz_{\mathrm{I}}(\vec{r})\,,$$

where the R_2^* *map* $z_{\mathrm{R}}(\vec{r})$ and the *field map* $z_{\mathrm{I}}(\vec{r})$ characterize the relaxation rate and off-resonance effects respectively for T_2^*-weighted images. Here, we would like to reconstruct $z(\vec{r})$ from $\boldsymbol{y}$.

Reconstructing the continuous-space map $z(\vec{r})$ from the discrete MR data $\boldsymbol{y}$ using (3.2) is an ill-posed inverse problem. To simplify the problem we parametrize $f(\vec{r})$ and $z(\vec{r})$ with the following approximations [20]:

$$\begin{aligned} z(\vec{r}) &\approx \textstyle\sum_{n=1}^{N} z(\vec{r}_n)\, b(\vec{r}-\vec{r}_n), \\ f(\vec{r}) &\approx \textstyle\sum_{n=1}^{N} f(\vec{r}_n)\, b(\vec{r}-\vec{r}_n), \end{aligned} \tag{3.3}$$

where $b(\cdot)$ is the voxel basis function, chosen here as the 2D or 3D rect function. Using this in (3.2) gives:

$$s(t;\boldsymbol{z},\boldsymbol{f}) \triangleq \Phi(\vec{k}(t)) \sum_{n=1}^{N} f(\vec{r}_n)\, \mathrm{e}^{-tz(\vec{r}_n)}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)}, \tag{3.4}$$

$$\boldsymbol{z} \triangleq [z(\vec{r}_1), \ldots, z(\vec{r}_N)], \ \boldsymbol{f} \triangleq [f(\vec{r}_1), \ldots, f(\vec{r}_N)],$$

where $s(t;\boldsymbol{z},\boldsymbol{f})$ is the discrete-space MR signal equation and $\Phi(\vec{k}(t))$ is the Fourier transform of $b(\cdot)$. This form of the signal equation can now be used to reconstruct a discrete image $\boldsymbol{z}$ from $\boldsymbol{y}$. Note that $\boldsymbol{f}$ is generally not known and thus either needs to be determined before $\boldsymbol{z}$ is reconstructed or jointly reconstructed with $\boldsymbol{z}$ using (3.4).

In an fMRI study, a series of time frames are collected in an MRI scanner, where a frame can either be an image slice (2D) or volume (3D). Using (3.1) and (3.4) we model the received fMRI data for time frame j as follows:

$$\boldsymbol{y}_j = \boldsymbol{s}(\boldsymbol{z}_j,\boldsymbol{f}) + \boldsymbol{\epsilon}_j, \quad j = 1,\ldots,J, \tag{3.5}$$

$$\boldsymbol{s}(\boldsymbol{z}_j,\boldsymbol{f}) \triangleq [s(t_1;\boldsymbol{z}_j,\boldsymbol{f}), \ldots, s(t_M;\boldsymbol{z}_j,\boldsymbol{f})],$$

where $\boldsymbol{y}_j$ is the received MR data, $\boldsymbol{s}(\boldsymbol{z}_j,\boldsymbol{f})$ is the discrete MR signal from (3.4) and $\boldsymbol{\epsilon}_j$ is iid Gaussian noise. An assumption in (3.5) is that $\boldsymbol{f}$ is not time frame dependent, which in the absence of motion is reasonable for single shot MR data acquired using low-flip angles and/or long TRs to control for blood inflow enhancement [35]. Since $\boldsymbol{f}$ is time frame independent, it can be reconstructed from specifically collected MR data $\boldsymbol{y}_0$ prior to the fMRI acquisition. The reconstructed $\boldsymbol{f}$ is denoted $\hat{\boldsymbol{f}}$.

Using $\hat{\boldsymbol{f}}$ we can reconstruct $\boldsymbol{z}_j$ from $\boldsymbol{y}_j$ for $j = 1, \ldots, J$ by minimizing a penalized likelihood cost function as follows:

$$\begin{aligned} \Psi(\boldsymbol{z}_j) &= \frac{1}{2} \left\| \boldsymbol{y}_j - \boldsymbol{s}(\boldsymbol{z}_j) \right\|^2 + \mathrm{R}(\boldsymbol{z}_j) \,, \\ \hat{\boldsymbol{z}}_j &= \arg\min_{\boldsymbol{z}_j} \Psi(\boldsymbol{z}_j) \,, \end{aligned} \tag{3.6}$$

where $\boldsymbol{s}(\boldsymbol{z}_j) \triangleq \boldsymbol{s}\left(\boldsymbol{z}_j, \hat{\boldsymbol{f}}\right)$, $\hat{\boldsymbol{z}}_j$ is the reconstructed $\boldsymbol{z}_j$ and $\mathrm{R}(\boldsymbol{z}_j)$ is an appropriately chosen roughness penalty (see Section 3.3.1) that controls the tradeoff between spatial resolution and noise in $\hat{\boldsymbol{z}}_j$. The cost function $\Psi(\boldsymbol{z}_j)$ is non-convex and would require a nonlinear iterative minimization algorithm, *e.g.*, [42, 43, 45]. Any algorithm used to minimize it is likely to return a local minimum and thus needs to be carefully initialized.

We propose to solve (3.6) by converting it to a sequence of quadratic optimization problems so that fast iterative reconstruction algorithms can be used. The idea is to use a linear approximation to the dynamic temporal changes between $\boldsymbol{z}_j$ and a carefully chosen reference $\check{\boldsymbol{z}}$. The advantage of using a sequence of quadratic approximations is that for each one, we can precompute temporal interpolators required for fast field-corrected image reconstruction [20], and then minimize that quadratic by NUFFT or Toeplitz methods [19]. In contrast, if we apply gradient descent or CG directly to (3.6), each gradient calculation would need new temporal interpolator coefficients, significantly increasing the computation time per iteration.

3.3 Dynamic $\boldsymbol{z}_j$ Reconstruction

This section describes an efficient algorithm for (3.6). Suppose a previously reconstructed reference map $\check{\boldsymbol{z}}$ is available (see discussion later in this section) in addition to $\hat{\boldsymbol{f}}$. Under this assumption, adding and subtracting $\check{\boldsymbol{z}}$ in the exponent containing $\boldsymbol{z}_j$ in (3.4) gives the following:

$$s(t;\boldsymbol{z}_j) = \Phi(\vec{k}(t)) \sum_{n=1}^{N} \hat{f}(\vec{r}_n)\,\mathrm{e}^{-t\check{z}(\vec{r}_n)}\,\mathrm{e}^{-t(z_j(\vec{r}_n)-\check{z}(\vec{r}_n))}\,\mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)}, \tag{3.7}$$

where the exponential has been split into two separate exponentials with one containing the difference of $\check{\boldsymbol{z}}$ and $\boldsymbol{z}_j$. When this difference is small, that term can be approximated using a first-order Taylor expansion, as follows:

$$\mathrm{e}^{-t(z_j(\vec{r}_n)-\check{z}(\vec{r}_n))} \approx 1 - t\left(z_j(\vec{r}_n) - \check{z}(\vec{r}_n)\right). \tag{3.8}$$

Substituting this in (3.7) yields:

$$\begin{aligned} s(t;\boldsymbol{z}_j) &\approx \Phi(\vec{k}(t)) \sum_{n=1}^{N} \hat{f}(\vec{r}_n)\,\mathrm{e}^{-t\check{z}(\vec{r}_n)}\left(1 - t\left(z_j(\vec{r}_n) - \check{z}(\vec{r}_n)\right)\right)\mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)} \\ &= s(t;\check{\boldsymbol{z}}) + \Phi(\vec{k}(t)) \sum_{n=1}^{N} \hat{f}(\vec{r}_n)\,\mathrm{e}^{-t\check{z}(\vec{r}_n)}\cdot(-t)\left(z_j(\vec{r}_n) - \check{z}(\vec{r}_n)\right)\mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)}. \end{aligned} \tag{3.9}$$

Using the approximation in (3.8) the relationship of $s(t;\boldsymbol{z}_j)$ to $\boldsymbol{z}_j$ is now approximately linear (or more precisely affine) according to (3.9). This allows us to rewrite (3.9) in a matrix-vector form as follows:

$$\begin{aligned} \boldsymbol{s}(\boldsymbol{z}_j) &\approx \boldsymbol{s}(\boldsymbol{z}_j;\check{\boldsymbol{z}}), \\ \boldsymbol{s}(\boldsymbol{z}_j;\check{\boldsymbol{z}}) &\triangleq \boldsymbol{s}(\check{\boldsymbol{z}}) + \boldsymbol{A}(\check{\boldsymbol{z}})\left(\boldsymbol{z}_j - \check{\boldsymbol{z}}\right), \end{aligned} \tag{3.10}$$

where $\boldsymbol{A}(\cdot)$ is the *system matrix* of size $M \times N$, with elements written as follows:

$$a_{mn}(\check{\boldsymbol{z}}) = \Phi(\vec{k}(t_m))\, \hat{f}(\vec{r}_n)\, \mathrm{e}^{-t_m \check{z}(\vec{r}_n)}\, (-t_m)\, \mathrm{e}^{-i2\pi\left(\vec{k}(t_m)\cdot\vec{r}_n\right)} . \tag{3.11}$$

Using the approximation in (3.10), we can now approximate the difference of $\boldsymbol{y}_j$ and $\boldsymbol{s}(\boldsymbol{z}_j)$, as follows:

$$\begin{aligned} \boldsymbol{y}_j - \boldsymbol{s}(\boldsymbol{z}_j) &\approx \boldsymbol{y}_j - \boldsymbol{s}(\boldsymbol{z}_j; \check{\boldsymbol{z}}) \\ &= \widetilde{\boldsymbol{y}}_j(\check{\boldsymbol{z}}) - \boldsymbol{A}(\check{\boldsymbol{z}})\, \boldsymbol{z}_j, \end{aligned} \tag{3.12}$$

where,

$$\widetilde{\boldsymbol{y}}_j(\check{\boldsymbol{z}}) \triangleq \boldsymbol{y}_j - \boldsymbol{s}(\check{\boldsymbol{z}}) + \boldsymbol{A}(\check{\boldsymbol{z}})\, \check{\boldsymbol{z}}.$$

Using the approximation in (3.12) we can form a new cost function as follows:

$$\Psi(\boldsymbol{z}_j; \check{\boldsymbol{z}}) = \frac{1}{2} \left\| \widetilde{\boldsymbol{y}}_j(\check{\boldsymbol{z}}) - \boldsymbol{A}(\check{\boldsymbol{z}})\, \boldsymbol{z}_j \right\|^2 + \mathrm{R}(\boldsymbol{z}_j)\,, \tag{3.13}$$

where $\Psi(\boldsymbol{z}_j; \check{\boldsymbol{z}})$ is quadratic in terms of the objective $\boldsymbol{z}_j$. By minimizing $\Psi(\boldsymbol{z}_j; \check{\boldsymbol{z}})$ one can then reconstruct $\boldsymbol{z}_j$ using fast iterative reconstruction algorithms such as [19, 20]. How well that matches to the result of (3.6) depends on the approximation made in (3.8), *i.e.*, we must find a $\check{\boldsymbol{z}}$ that is close enough to the true $\boldsymbol{z}_j$ so that (3.8) does not introduce too much error in (3.12).

As previously discussed, the MR data $\boldsymbol{y}_0$ is already needed to get $\hat{\boldsymbol{f}}$. This data could be acquired so that $\boldsymbol{z}_0$ and $\boldsymbol{f}$ are jointly reconstructed from $\boldsymbol{y}_0$ [42, 43]. Thus, one might choose $\check{\boldsymbol{z}}$ to be the reconstructed map $\hat{\boldsymbol{z}}_0$. This reference map would approximate the *baseline* state of R_2^* and field map. However, in an fMRI study at 3T the voxels showing activation have $\boldsymbol{z}_{\mathrm{R}}$ and $\boldsymbol{z}_{\mathrm{I}}$ showing maximum temporal changes of approximately -2s^{-1} [2] and 3Hz [45] respectively for a 3 minute scan relative to baseline. By using $\boldsymbol{z}_0$ as the reference map and for a typical single shot acquisition with a $\mathrm{TE} = 30$ms the NRMS error for the approximation in (3.8) would at worst be $\sim$15%, which would be undesirably large.

A more appropriate choice is to dynamically update $\check{\boldsymbol{z}}$ using a previous estimate of $\boldsymbol{z}_j$, where the estimate then gets gradually *refined*. To differentiate between the refinements we denote them as $\hat{\boldsymbol{z}}_j^{(l)}$, where l is the refinement index. Thus, we choose $\check{\boldsymbol{z}}$ as $\hat{\boldsymbol{z}}_j^{(l-1)}$ when reconstructing $\hat{\boldsymbol{z}}_j^{(l)}$, and if the total number of refinements are L we set:

$$\hat{\boldsymbol{z}}_j^{(0)} = \hat{\boldsymbol{z}}_{j-1} = \hat{\boldsymbol{z}}_{j-1}^{(L)}.$$

With this choice of $\check{\boldsymbol{z}}$, (3.8) should have a smaller approximation error as l increases, and thus the approximation in (3.12) should improve when used in (3.13).

Including the refinement concept into the reconstruction algorithm, we rewrite (3.12) as follows:

$$\boldsymbol{y}_j - \boldsymbol{s}(\boldsymbol{z}_j) \approx \widetilde{\boldsymbol{y}}_j\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right) - \boldsymbol{A}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)\boldsymbol{z}_j.$$

Using this we reconstruct $\hat{\boldsymbol{z}}_j$ for $j = 1, \ldots, J$ by minimizing a quadratic cost function as follows:

$$\hat{\boldsymbol{z}}_j^{(l)} = \arg\min_{\boldsymbol{z}_j} \Psi\left(\boldsymbol{z}_j; \hat{\boldsymbol{z}}_j^{(l-1)}\right), \quad l = 1, \ldots, L, \tag{3.14}$$

where $\Psi(\boldsymbol{z}_j; \cdot)$ was defined in (3.13) and $\hat{\boldsymbol{z}}_j = \hat{\boldsymbol{z}}_j^{(L)}$. This form of the reconstruction algorithm is very flexible and should approximate well the results of the original nonlinear reconstruction problem given in (3.6).

3.3.1 Roughness Penalty $\mathrm{R}(\boldsymbol{z}_j)$ and Its Implications on $\hat{\boldsymbol{z}}_j$

We must choose a roughness penalty $\mathrm{R}(\boldsymbol{z}_j)$ in (3.14). The conventional choice is given as follows:

$$\frac{1}{2}\beta \left\|\boldsymbol{C}\boldsymbol{z}_j\right\|^2,$$

where β is the regularization parameter and $\boldsymbol{C}$ is a real-valued first-order difference matrix that evaluates the differences of neighboring pixels within a user specified neighborhood.

If $\mathrm{R}(\boldsymbol{z}_j)$ were defined as this penalty the real and imaginary parts of $\boldsymbol{z}_j$ would be penalized equally. However, because the field map $\boldsymbol{z}_{\mathrm{I}}$ is usually smoother than the R_2^* map $\boldsymbol{z}_{\mathrm{R}}$, we choose to regularize these maps separately, as follows:

$$\mathrm{R}(\boldsymbol{z}_j) = \frac{1}{2}\left(\beta_1 \|\boldsymbol{C}\boldsymbol{z}_{\mathrm{R}j}\|^2 + \beta_2 \|\boldsymbol{C}\boldsymbol{z}_{\mathrm{I}j}\|^2\right), \tag{3.15}$$

where $\boldsymbol{C}$ is the first-order difference matrix and $\boldsymbol{z}_{\mathrm{R}j}$ and $\boldsymbol{z}_{\mathrm{I}j}$ are respectively the real and imaginary parts of $\boldsymbol{z}_j$. Note that $\boldsymbol{C}$ can be defined separately for $\boldsymbol{z}_{\mathrm{R}j}$ and $\boldsymbol{z}_{\mathrm{I}j}$. This can have some advantages, *e.g.*, defining $\boldsymbol{C}$ for $\boldsymbol{z}_{\mathrm{I}j}$ as a second-order difference matrix as suggested in [8] while $\boldsymbol{C}$ for $\boldsymbol{z}_{\mathrm{R}j}$ is a first-order difference matrix. However, for simplicity we chose to use the same matrix for both parameters.

As evident from (3.11) the elements of $\boldsymbol{A}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)$ depend on $\hat{\boldsymbol{f}}$, which has values near 0 outside the object. This relationship has the following consequences:

$$\forall n \text{ s.t. } \hat{f}(\vec{r}_n) \approx 0 \Rightarrow \left[\boldsymbol{A}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)\boldsymbol{z}_j\right]_n \approx 0,\ \forall \boldsymbol{z}_j \in \mathbb{C}^N.$$

This implies that the roughness penalty $\mathrm{R}(\boldsymbol{z}_j)$ becomes the dominant factor in the cost function $\Psi\left(\boldsymbol{z}_j; \hat{\boldsymbol{z}}_j^{(l-1)}\right)$ for spatial positions outside the object.

3.3.2 Implementation of the Fast Iterative Algorithm

For fast minimization of $\Psi\left(\boldsymbol{z}_j; \hat{\boldsymbol{z}}_j^{(l-1)}\right)$ in (3.14) we use the Conjugate Gradient (CG) method [21]. To reduce memory we never explicitly form the large matrix $\boldsymbol{A}(\cdot)$, rather we use a software object to represent this matrix.[2] This software uses fast methods such as FFT (for cartesian k-space trajectories) or NUFFT (for noncartesian k-space trajectories) [25] and temporal segmentation [19, 20] to greatly decrease computation time. The total compute time of the reconstruction algorithm is then roughly L times longer than previous fast iterative algorithms used to reconstruct T_2^*-weighted images [19, 20].

---

[2] Software available at http://www.eecs.umich.edu/~fessler/

The convergence of CG depends on how it is initialized. A common CG initialization is to set all the elements of the initial solution to zero. This is a convenient initialization, but does not exploit any prior knowledge of the solution of (3.14). Another CG initialization would be to use the conjugate phase (CP) [17] reconstructed $\boldsymbol{z}_j$. This initializer was shown to improve the convergence of CG, compared to initializing with all zeros, when used to iteratively reconstruct an off-resonance corrected T_2^*-weighted image [20]. However, since $\boldsymbol{A}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)$ also includes R_2^* relaxation, CP is ineffective here.

As previously stated, $\boldsymbol{z}_j$ should change only slightly between neighboring time frames and across refinements. Thus, it is advantageous to exploit this relatively gradual temporal change to initialize the CG algorithm sensibly. Hence, when we reconstruct $\hat{\boldsymbol{z}}_j^{(l)}$ the previous refinement $\hat{\boldsymbol{z}}_j^{(l-1)}$ is used to initialize CG.

3.4 Resolution Properties: Regularization Design

The resolution properties of $\hat{\boldsymbol{z}}_j^{(l)}$ are important to further understand the relationship of the regularization function in (3.15) and the spatial smoothness of $\hat{\boldsymbol{z}}_j^{(l)}$. Local point spread functions (LPSF) [23] have been previously used to analyze this relationship, using the approximate local resolution properties of regularized reconstruction algorithms with parametrized object models [24] as in (3.3). This analysis will be used first, to check if the penalized reconstruction in (3.14) has uniform spatial resolution and, if needed, design a penalty to achieve such uniformity. Then it will be used to set β values to achieve a predetermined resolution that is quantified using the full-width half-max (FWHM) of the LPSF.

3.4.1 Resolution Analysis

We estimate $\hat{\boldsymbol{z}}_j^{(l)}$ by minimizing $\Psi\left(\boldsymbol{z}_j; \hat{\boldsymbol{z}}_j^{(l-1)}\right)$ in (3.14) with separate regularization of the real and imaginary parts per (3.15). Sections 4.2.1 and 4.2.2 derive a *stacked* cost function and its LPSF for such reconstruction problems. If that LPSF can be shown to be shift invariant then the spatial resolution of $\hat{\boldsymbol{z}}_j^{(l)}$ is uniform and the resolution properties of the algorithm can be quantified approximately by evaluating the LPSF in (4.8) at a single spatial location. If both $\boldsymbol{A}'\boldsymbol{A}$ and $\boldsymbol{C}'\boldsymbol{C}$ are Toeplitz, then using (4.8) one can show that the LPSF is approximately locally shift invariant.

For the reconstruction method used in this paper $\boldsymbol{C}'\boldsymbol{C}$ is Toeplitz but $\boldsymbol{A}'\boldsymbol{A}$ is not as seen from (3.11). Thus the LPSF is shift variant, which makes the resolution nonuniform for the usual first-order difference matrix $\boldsymbol{C}$ in (3.15). Next we propose a spatially variant penalty design that leads to approximately uniform local spatial resolution.

3.4.2 Spatially Variant Penalty Design

Although $\boldsymbol{A}'\boldsymbol{A}$ is not Toeplitz, using methods similar to those proposed in [24] we can find an approximation of the form:

$$\boldsymbol{A}'\boldsymbol{A} \approx \boldsymbol{D}'\boldsymbol{G}'\boldsymbol{G}\boldsymbol{D}, \tag{3.16}$$

where $\boldsymbol{G}'\boldsymbol{G}$ is Toeplitz and $\boldsymbol{D}$ is a real-valued and invertible matrix. Having found such an approximation, we introduce a spatially variant differencing matrix $\widetilde{\boldsymbol{C}}$, as follows:

$$\widetilde{\boldsymbol{C}} = \boldsymbol{C}\boldsymbol{D}. \tag{3.17}$$

By replacing $\boldsymbol{C}$ with $\widetilde{\boldsymbol{C}}$ in (3.15) the stacked LPSF in (4.8) becomes as follows:

$$\begin{aligned} \boldsymbol{l}_{\mathrm{S}n} &= \left(\boldsymbol{A}_{\mathrm{S}}'\boldsymbol{A}_{\mathrm{S}} + \widetilde{\boldsymbol{C}}_{\mathrm{S}}'\widetilde{\boldsymbol{C}}_{\mathrm{S}}\right)^{-1} \boldsymbol{A}_{\mathrm{S}}'\boldsymbol{A}_{\mathrm{S}}\boldsymbol{e}_{\mathrm{S}n} && (3.18)\\ &\approx d_n \boldsymbol{D}_{\mathrm{S}}^{-1} \left(\boldsymbol{G}_{\mathrm{S}}'\boldsymbol{G}_{\mathrm{S}} + \boldsymbol{C}_{\mathrm{S}}'\boldsymbol{C}_{\mathrm{S}}\right)^{-1} \boldsymbol{G}_{\mathrm{S}}'\boldsymbol{G}_{\mathrm{S}}\boldsymbol{e}_{\mathrm{S}n}, \\ &\approx \left(\boldsymbol{G}_{\mathrm{S}}'\boldsymbol{G}_{\mathrm{S}} + \boldsymbol{C}_{\mathrm{S}}'\boldsymbol{C}_{\mathrm{S}}\right)^{-1} \boldsymbol{G}_{\mathrm{S}}'\boldsymbol{G}_{\mathrm{S}}\boldsymbol{e}_{\mathrm{S}n}, && (3.19) \end{aligned}$$

where $\boldsymbol{A}_{\mathrm{S}}'\boldsymbol{A}_{\mathrm{S}}$ and $\boldsymbol{e}_{\mathrm{S}n}$ are shown in (4.9), $\widetilde{\boldsymbol{C}}_{\mathrm{S}}$ is a block diagonal matrix with $\sqrt{\beta_1}\widetilde{\boldsymbol{C}}$ and $\sqrt{\beta_2}\widetilde{\boldsymbol{C}}$ forming the diagonal blocks, d_n is the nth diagonal element of $\boldsymbol{D}$ and

$$\boldsymbol{G}_{\mathrm{S}}'\boldsymbol{G}_{\mathrm{S}} = \begin{bmatrix} \Re\left(\boldsymbol{G}'\boldsymbol{G}\right) & -\Im\left(\boldsymbol{G}'\boldsymbol{G}\right) \\ \Im\left(\boldsymbol{G}'\boldsymbol{G}\right) & \Re\left(\boldsymbol{G}'\boldsymbol{G}\right) \end{bmatrix}, \ \boldsymbol{D}_{\mathrm{S}} = \begin{bmatrix} \boldsymbol{D} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{D} \end{bmatrix}.$$

Thus even though $\boldsymbol{A}'\boldsymbol{A}$ is not Toeplitz we can still make the LPSF to be approximately locally shift invariant by introducing a spatially variant penalty of the form (3.17) in the reconstruction.

We use a diagonal matrix $\boldsymbol{D}$ such that the diagonal of $\boldsymbol{A}'\boldsymbol{A}$ and $\boldsymbol{D}'\boldsymbol{G}'\boldsymbol{G}\boldsymbol{D}$ are equal. This equality constraint when estimating $\hat{\boldsymbol{z}}_j^{(l)}$ can be written as follows:

$$\sum_{m=1}^{M} \left| a_{mn}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)\right|^2 = d_n\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)^2 \cdot \sum_{m=1}^{M} \left|g_{mn}\right|^2. \tag{3.20}$$

To find d_n we need to define g_{mn} such that $\boldsymbol{G}'\boldsymbol{G}$ is Toeplitz and yet ensure (3.16) is a good approximation. The term in $\boldsymbol{A}$ that makes $\boldsymbol{A}'\boldsymbol{A}$ non-Toeplitz is $\hat{f}(\vec{r}_n)\,\mathrm{e}^{-t_m \hat{z}_j^{(l-1)}(\vec{r}_n)}$. Even though $\hat{f}(\vec{r}_n)$ can be separated from $\boldsymbol{A}'\boldsymbol{A}$ as proposed in (3.16) the same cannot be said for $\mathrm{e}^{-t_m \hat{z}_j^{(l-1)}(\vec{r}_n)}$ because of its spatio-temporal structure. However, by approximating the elements of $\hat{\boldsymbol{z}}_j^{(l-1)}$ with the median value of $\hat{\boldsymbol{z}}_0$, denoted $\overline{z}$, we define the elements of $\boldsymbol{G}$ as follows:

$$g_{mn} \triangleq \Phi(\vec{k}(t_m))\ \mathrm{e}^{-t_m \overline{z}}\,(-t_m)\,\mathrm{e}^{-i2\pi\left(\vec{k}(t_m)\cdot\vec{r}_n\right)}, \tag{3.21}$$

This definition of $\boldsymbol{G}$ makes $\boldsymbol{G}'\boldsymbol{G}$ Toeplitz and the diagonal elements of $\boldsymbol{D}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right) = \boldsymbol{D}$

are then given as follows:

$$
\begin{aligned}
d_n\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right) &= \sqrt{\frac{\sum_{m=1}^{M} \left|a_{mn}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)\right|^2}{\sum_{m=1}^{M} |g_{mn}|^2}} \\
&= |f(\vec{r}_n)| \sqrt{\frac{\sum_{m=1}^{M} c_m^2 \,\mathrm{e}^{-2t_m \hat{z}_{\mathrm{R}_j}^{(l-1)}(\vec{r}_n)}}{\sum_{m=1}^{M} c_m^2 \,\mathrm{e}^{-2t_m \bar{z}_{\mathrm{R}}}}},
\end{aligned}
\tag{3.22}
$$

with $c_m \triangleq \Phi(\vec{k}(t_m))\, t_m$. Note that this form of $\boldsymbol{D}$ allows us to use (3.19) to find a single pair of regularization parameters to achieve a desired resolution that depends only on $\hat{\boldsymbol{z}}_0$.

3.4.3 Spatially Variant Penalty Implementation

To implement the penalty designed in (3.22) there were some issues regarding computational speed and stability of the reconstruction that needed to be addressed.

For stability, we find $\boldsymbol{D}$ according to (3.22) at the beginning of the algorithm after estimating $\hat{\boldsymbol{z}}_0$ and use that for subsequent j and l, *i.e.*, $\boldsymbol{D}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right) \triangleq \boldsymbol{D}(\hat{\boldsymbol{z}}_0)$ for all (l, j). Also, calculating (3.22) for all n is computationally expensive due to the spatio-temporal dependence of the numerator. However, since an R_2^* map generally has values within a fairly tight range we approximate the numerator by histograming $\hat{\boldsymbol{z}}_0$ and replace $\hat{z}_{\mathrm{R}_j}^{(l-1)}(\vec{r}_n)$ with the center bin value it falls within. Our implementation of the penalty suggested in (3.17) uses the modification given in equation (35) in [24].

Equation (3.19) is implemented efficiently using FFTs as is derived in Section 4.3. This allows us to evaluate it for multiple β values, calculate the FWHM of the resulting LPSFs and interpolate that to the desired FWHMs and their associated β values in a very fast manner.

3.5 Reconstruction Algorithm – Overview

We can now summarize the proposed reconstruction algorithm as follows:

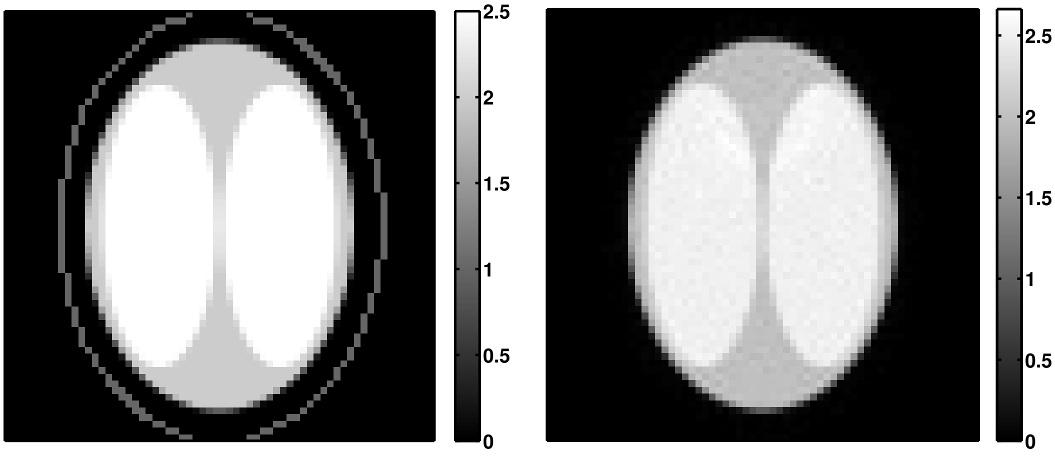
- Find $\hat{\boldsymbol{f}}$ and $\hat{\boldsymbol{z}}_0$ from multi-echo MR data $\boldsymbol{y}_0$ that is collected at the beginning of the fMRI study.
- Using (3.19), find β_1 and β_2 that yield the desired resolution for the real and imaginary parts of $\hat{\boldsymbol{z}}_j$.
- Generate $\boldsymbol{D}$ for the regularizer using (3.22) with $\hat{\boldsymbol{z}}_0$.
- Reconstruct $\hat{\boldsymbol{z}}_j$ for all $j = 1, \ldots, J$ as follows:

```
for $j = 1, \ldots, J$        [time]
    Set $\hat{\boldsymbol{z}}_j^{(0)} = \hat{\boldsymbol{z}}_{j-1}$.
    for $l = 1, \ldots, L$        [refinement]
        Form $\boldsymbol{A}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)$ and generate $\boldsymbol{s}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)$.
        Solve $\hat{\boldsymbol{z}}_j^{(l)} = \arg\min_{\boldsymbol{z}_j} \Psi\left(\boldsymbol{z}_j; \hat{\boldsymbol{z}}_j^{(l-1)}\right)$ using CG.
    end
    Set $\hat{\boldsymbol{z}}_j = \hat{\boldsymbol{z}}_j^{(L)}$.
end
```

3.6 Simulations

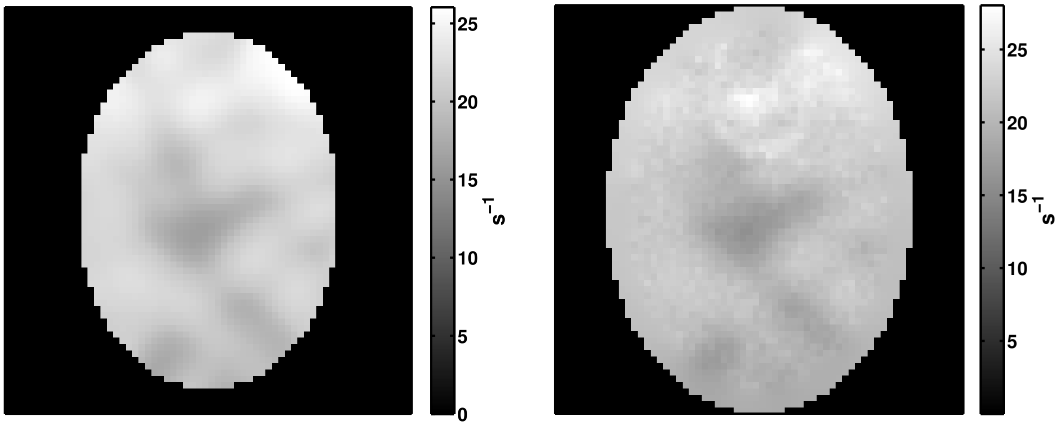
We simulated k-space data using the exact form of the signal equation given in (3.4), with no temporal interpolation. We used a 4713 sample spiral-out k-space trajectory with a readout of 18.8ms, field of view of 22cm and maximum gradient amplitude and slew rate of 22mT/m and 180mT/m/ms respectively. The simulation maps were 128×128 but reconstructed as 64×64, unless otherwise noted, with the baseline maps $\boldsymbol{f}$ and $\boldsymbol{z}_0$ shown in Figure 3.1(a) - 3.1(c). For simulations corrupted by noise, we found the variance of the iid Gaussian noise to make the SNR 80, 55 or 30 for the baseline k-space data with $\mathrm{TE} = 30$ms, where:

$$\mathrm{SNR} = \frac{\|\boldsymbol{s}(\boldsymbol{z}_0, \boldsymbol{f})\|}{\|\boldsymbol{\epsilon}_0\|}.$$



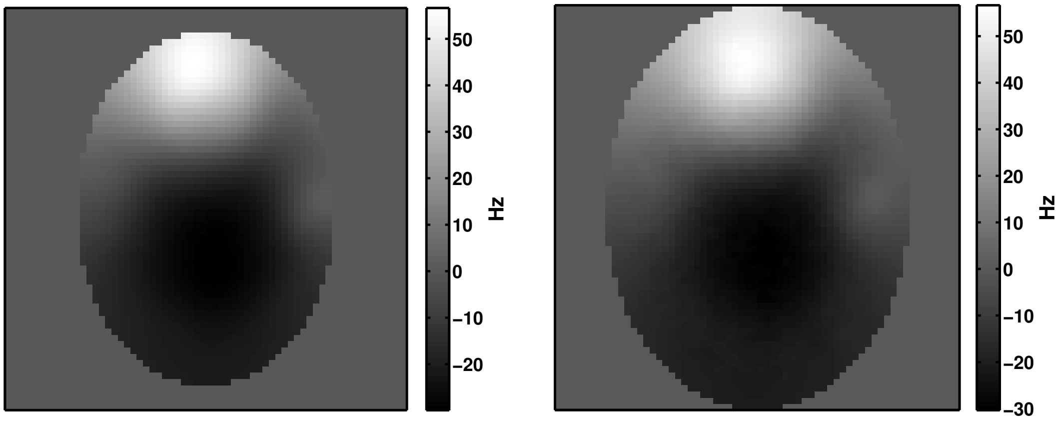
(a)

(d) **NRMSE = $(\mathbf{3.9}, \mathbf{5.3}, \mathbf{8.8})\%$**



(b)

(e) **RMSE = $(\mathbf{0.61}, \mathbf{0.66}, \mathbf{0.85})\mathrm{s}^{-1}$**



(c)

(f) **RMSE = $(\mathbf{0.30}, \mathbf{0.41}, \mathbf{0.70})$Hz**

Figure 3.1: True and reconstructed maps (SNR $= 55$) used in the simulations. (a) $\boldsymbol{f}$ with the edge of the reconstruction mask for reference; (b) $\boldsymbol{z}_{\mathrm{R0}}$; (c) $\boldsymbol{z}_{\mathrm{I0}}$; (d) $\hat{\boldsymbol{f}}$; (e) $\hat{\boldsymbol{z}}_{\mathrm{R0}}$; (f) $\hat{\boldsymbol{z}}_{\mathrm{I0}}$. The reconstructed maps for simulated k-space data are only shown for voxels within the reconstruction mask and the (N)RMSE is the (normalized) RMS error of the reconstructed map relative to the true map within the mask $|\boldsymbol{f}| > 0$ for SNR $= (80, 55, 30)$.

This noise variance was then used to generate iid Gaussian noise for k-space data of other time frames and TEs.

For the iterative algorithm parameters we used 9 segments for the temporal segmentation of $\mathrm{e}^{-t\boldsymbol{z}(\vec{r}_n)}$, which were interpolated using min-max interpolation coefficients [20]. When comparing the exact and interpolated complex exponential the maximum error and normalized RMS error (NRMSE) were on the order of 10^{-7} and 10^{-8} respectively for the simulation maps shown in Figure 3.1(b) - 3.1(c). For the NUFFT parameters we used $2\times$ oversampling and a 5×5 neighborhood [25]. We ran 20 iterations of CG for each (l, j) pair to get $\hat{\boldsymbol{z}}_j^{(l)}$ and used a reconstruction mask to reduce the number of reconstructed voxels from 4096 to 2404. Figure 3.1(a) shows the edge of the reconstruction mask. All reconstructions were run on a 2.13Ghz Intel Core 2 Duo with 2GB of memory.

3.6.1 Initialization: Estimating $\hat{\boldsymbol{f}}$ and $\hat{\boldsymbol{z}}_0$

The proposed reconstruction needs $\hat{\boldsymbol{f}}$ and $\hat{\boldsymbol{z}}_0$ for initialization. These spatial maps were reconstructed from multiple fully sampled readouts, *i.e.*, *multi-echo data*. Here, we simulated 5 echos with noise, where the readouts had $\mathrm{TE} = [6.5, 4.5, 24.3, 44.1, 63.8]$ms. We reconstructed two T_2^*-weighted images from the 4.5ms and 6.5ms readouts using iterative reconstruction [20] and then estimated $\hat{\boldsymbol{z}}_{\mathrm{I}0}$ from the phase difference of these two images [7, 8]. This was repeated two times, where after estimating $\hat{\boldsymbol{z}}_{\mathrm{I}0}$ the first time it was used in the iterative reconstruction to correct for off-resonance during readout.

Using $\hat{\boldsymbol{z}}_{\mathrm{I}0}$, we reconstructed an off-resonance corrected T_2^*-weighted image for each echo of the multi-echo data using the same iterative reconstruction. The reconstructed images have different T_2^*-weighting which is assumed to occur at TE. This allowed us to fit the decay of each voxel in the reconstructed images to a mono-exponential model [50], which gave $\hat{\boldsymbol{z}}_{\mathrm{R}0}$. Since this fit is highly sensitive to noise in voxels with low SNR, spatially weighted smoothing was performed that applied low smoothing to $\hat{\boldsymbol{z}}_{\mathrm{R}0}$ in areas inside the object and higher smoothing where there are signal voids and outside the object, similar to

[8]. We chose the weights as the magnitude image with $\mathrm{TE} = 4.5\mathrm{ms}$. This was repeated three times, where after estimating $\hat{\boldsymbol{z}}_{\mathrm{R}0}$ each time it was used in the iterative reconstruction to correct for R_2^* relaxation during readout.

From [50], we know that $\hat{\boldsymbol{f}}$ could also be reconstructed using the mono-exponential fitting method. However, due to the poor fitting performance in voxels with low SNR, we chose to reconstruct $\hat{\boldsymbol{f}}$ iteratively using the signal model in (3.4) as follows:

$$\hat{\boldsymbol{f}} = \arg\min_{\boldsymbol{f}} \frac{1}{2} \left\| \boldsymbol{y}_0 - \boldsymbol{B}(\hat{\boldsymbol{z}}_0)\, \boldsymbol{f} \right\|^2 + \mathrm{R}(\boldsymbol{f}) \,,$$

where $\boldsymbol{y}_0$ is the concatenated multi-echo data, $\mathrm{R}(\boldsymbol{f})$ is a roughness penalty and the elements of $\boldsymbol{B}(\hat{\boldsymbol{z}}_0)$ are given as follows:

$$b_{mn}(\hat{\boldsymbol{z}}_0) = \Phi(\vec{k}(t_m))\ \mathrm{e}^{-t_m \hat{z}_0(\vec{r}_n)}\ \mathrm{e}^{-i2\pi\left(\vec{k}(t_m)\cdot\vec{r}_n\right)} \,,$$

where t_m are the concatenated time vector samples of the simulated readouts and $\hat{\boldsymbol{z}}_0$ is formed using $\hat{\boldsymbol{z}}_{\mathrm{R}0}$ and $\hat{\boldsymbol{z}}_{\mathrm{I}0}$. We chose the spatial regularization for $\hat{\boldsymbol{f}}$ so that the LPSF of the center voxel had a FWHM of approximately 1.25 voxels.

The results for simulated k-space data with $\mathrm{SNR} = 55$ are shown in Figure 3.1(d) - 3.1(f) and the (N)RMSE shown below the images is the (normalized) RMS error of the reconstructed map relative to the true map within the mask $|\boldsymbol{f}| > 0$ for all SNRs.

3.6.2 Resolution: Properties and Nonuniformity Correction

Here we analyze the performance of the proposed spatially variant penalty. All the analysis is based on the true simulation maps in Figure 3.1(a) - 3.1(c). We started by finding β_1 and β_2 to achieve a desired resolution. We chose the desired resolution such that the LPSF for the center voxel gave a FWHM of 1.35 and 1.5 voxels for the real and imaginary parts of $\hat{\boldsymbol{z}}_j^{(l)}$ respectively. This was done by evaluating the LPSF in (3.19) for 100 pairs of β_1 and

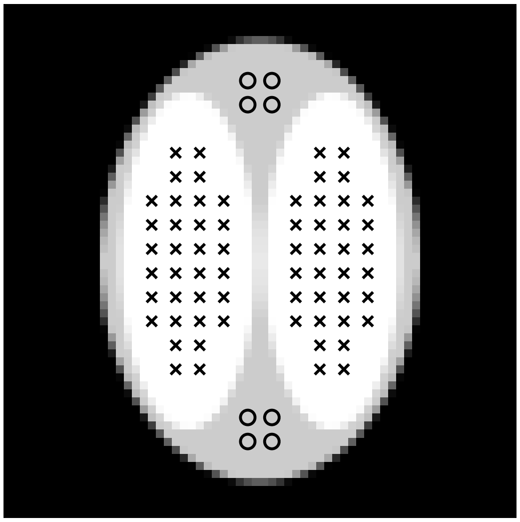
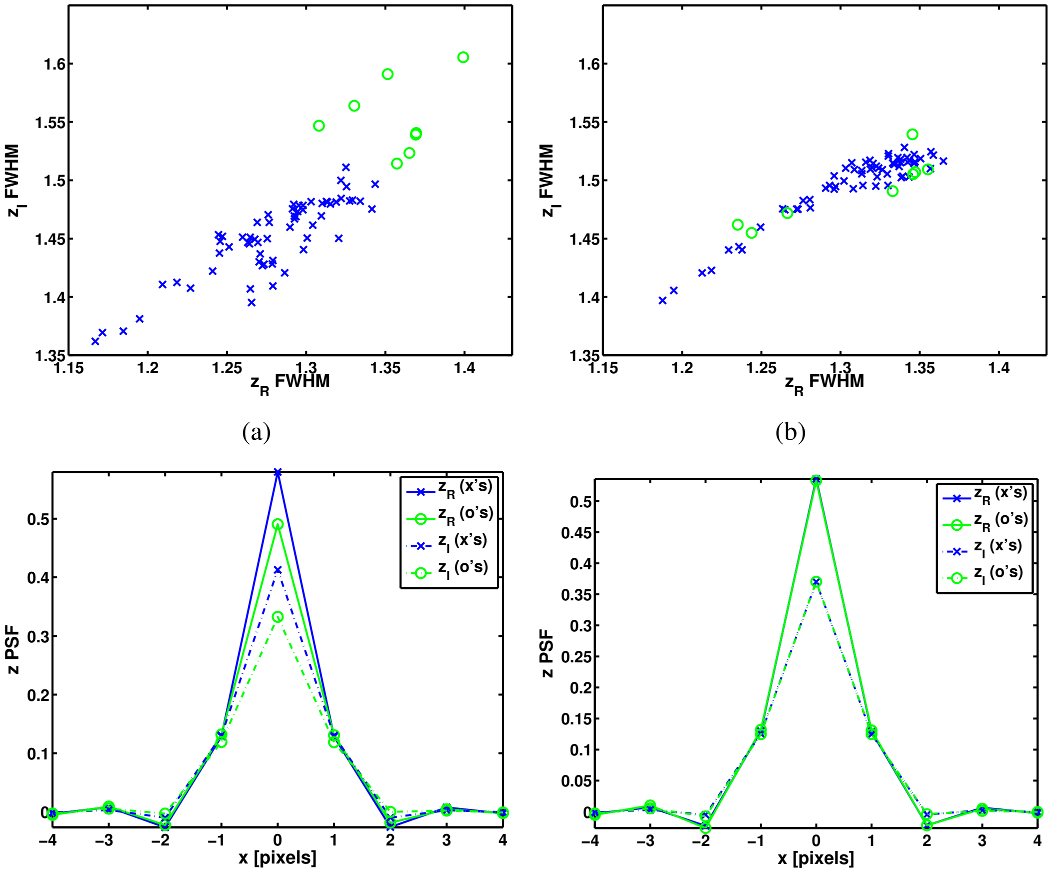


Figure 3.2: Voxel positions where the LPSFs in (3.18) were evaluated. The positions are split into 2 groups, indicated by the x's and o's, where each group are at locations where the value of $\boldsymbol{f}$ is the same.

β_2, where $\overline{z}$ was the median of the true $\boldsymbol{z}_0$ in Figure 3.1(b)-3.1(c). Using FFTs [59] it took 19.2s to evaluate the LPSFs for all 100 pairs. The FWHMs of those LPSFs were then used to interpolate the desired FWHM to their associated β values.

Using those β values, we calculated the exact LPSF using (3.18) at the voxel positions shown in Figure 3.2. We investigated the resolution properties by evaluating (3.18) for $\hat{\boldsymbol{z}}_1$, with $\boldsymbol{A}_{\mathrm{S}}$ formed by stacking $\boldsymbol{A}(\boldsymbol{z}_0)$ and using the true $\boldsymbol{f}$ in place of $\hat{\boldsymbol{f}}$. The spatially variant penalty was formed in 4.7s using (3.22), where the true simulation maps were again used and the numerator was calculated by histograming $\boldsymbol{z}_{\mathrm{R}0}$ with 100 bins. For comparison, we also designed a spatially nonvariant penalty by making a diagonal matrix using the mean of $d_n(\boldsymbol{z}_0)$ in (3.22) across all voxels where $|\boldsymbol{f}| > 0$. This made the nonvariant penalty approximately have the desired FWHM using the previously calculated β values.

Figure 3.3 shows both FWHM scatter plots and the average profile of the LPSF for the voxel positions in Figure 3.2 of the real and imaginary parts of $\hat{\boldsymbol{z}}_1$ for both penalties. The calculated FWHM values are more concentrated around the desired FWHM values when using the spatially variant penalty compared to the nonvariant one. This is further



(a)

(b)

(c)

(d)

Figure 3.3: FWHM scatter plots and average profile of the LPSF for the two groups of voxel positions in Figure 3.2. Results are shown for the real and imaginary parts of $\hat{z}_1$ when using the spatially variant and nonvariant penalty. (a) Scatter plot for spatially nonvariant penalty; (b) Scatter plot for spatially variant penalty; (c) Average profile for spatially nonvariant penalty; (d) Average profile for spatially variant penalty.

Table 3.1: Average FWHM for both $\hat{\boldsymbol{z}}_{\mathrm{R}1}$ and $\hat{\boldsymbol{z}}_{\mathrm{I}1}$ with its standard deviation for both the spatially nonvariant and proposed penalties.

	Spatially Nonvariant Penalty		Spatially Variant Penalty		Desired
	x’s	o’s	x’s	o’s	
$\hat{\boldsymbol{z}}_{\mathrm{R}1}$ FWHM	1.28 ± 0.04	1.36 ± 0.03	1.31 ± 0.04	1.31 ± 0.05	1.35
$\hat{\boldsymbol{z}}_{\mathrm{I}0}$ FWHM	1.45 ± 0.04	1.55 ± 0.03	1.50 ± 0.03	1.50 ± 0.03	1.50

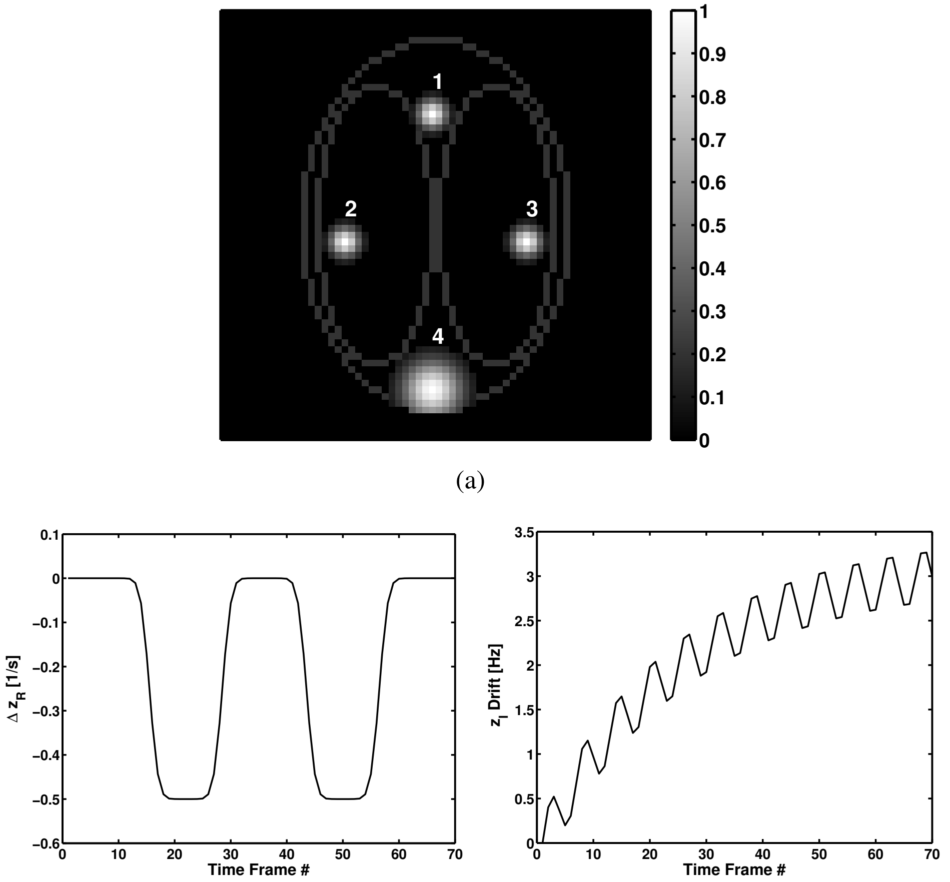
confirmed in the profile plots of the LPSFs averaged over each group for both the real and imaginary parts of $\hat{\boldsymbol{z}}_1$. This indicates the importance of compensating for spatial resolution nonuniformity in the reconstruction algorithm.

Table 3.1 shows the mean and the standard deviation of the FWHM for both the real and imaginary parts of $\hat{\boldsymbol{z}}_1$. It shows further evidence of the effect of $\boldsymbol{f}$ in causing resolution nonuniformity in the reconstructed $\boldsymbol{z}_j$. The mean FWHM of the two groups from Figure 3.2 deviate more when using a spatially nonvariant penalty compared to the proposed penalty. Repeating these measurements for $\boldsymbol{A}(\hat{\boldsymbol{z}}_j)$ for a time frame j during activation the numbers did not change significantly while still using $d_n(\boldsymbol{z}_0)$ to form $\boldsymbol{D}$. This indicates that it is sufficient to find $\boldsymbol{D}$ based on $\hat{\boldsymbol{f}}$ and $\hat{\boldsymbol{z}}_0$ and use that for all time frames.

3.6.3 Simulated fMRI Data

To analyze the proposed reconstruction in (3.14) for fMRI time series we compared its activation detection and R_2^* estimation performance to an iterative T_2^*-weighted reconstruction [20] and a multi-echo R_2^* reconstruction [50]. We simulated a 70 time frame four-echo fMRI data with $\mathrm{TE} = [10.2, 30, 49.8, 69.6]$ms. The $\mathrm{TE} = 30$ms readouts were used for the proposed and the T_2^*-weighted reconstructions. For the multi-echo R_2^* reconstruction we iteratively reconstructed four T_2^*-weighted images [20], one for each readout, and estimated both R_2^* and f by fitting a mono-exponential decay to each voxel of the T_2^*-weighted magnitude images.

Figure 3.4 shows the simulated fMRI spatial activation map and the temporal changes.



(a)

(b) (c)

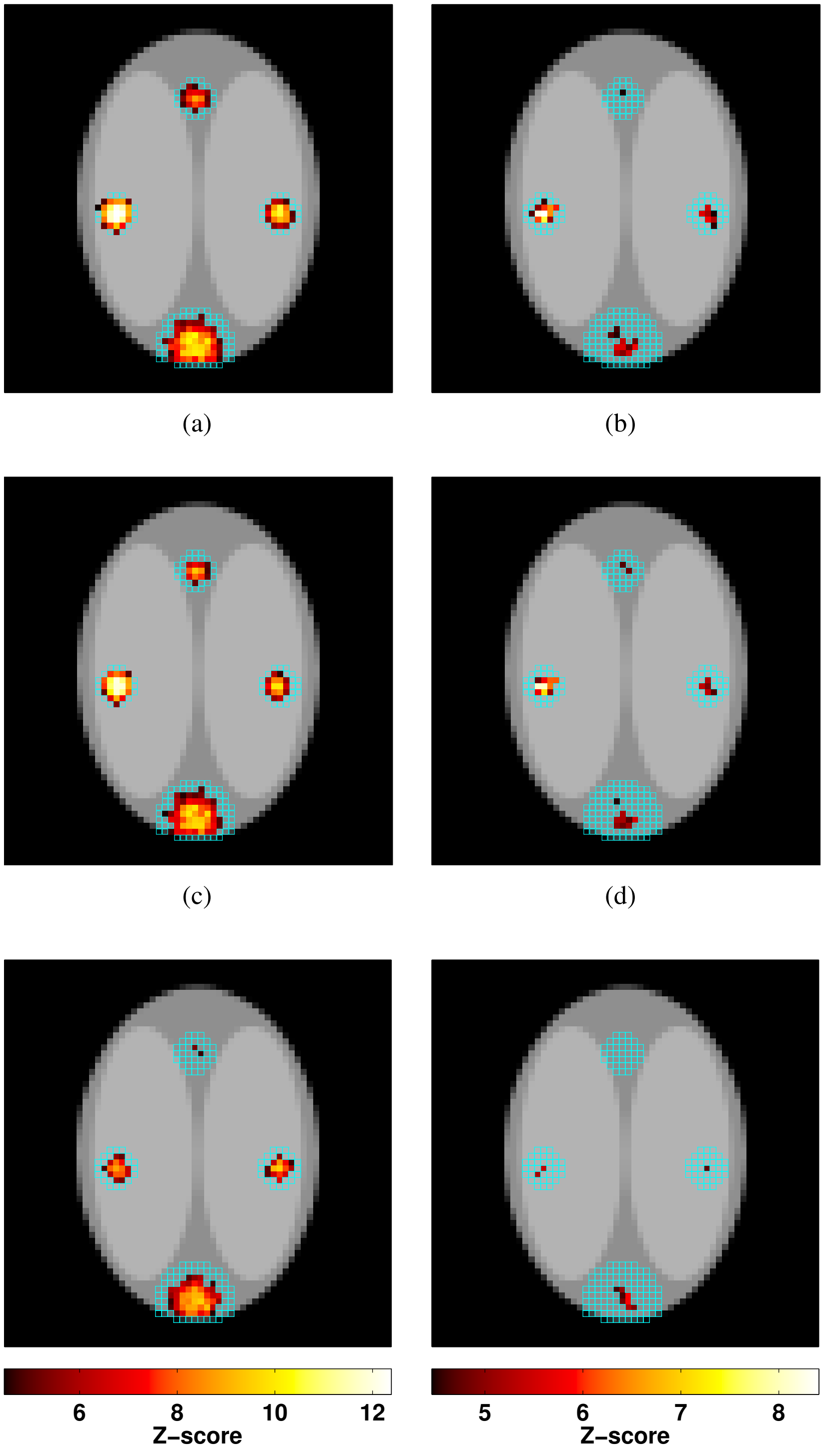
Figure 3.4: Simulated spatial and temporal activations. (a) Map of the spatial weights for four enumerated activation clusters, along with the edges of $\boldsymbol{f}$ in Figure 3.1(a) shown for reference; (b) Additive task waveform for R_2^* for all clusters. Additionally, in clusters 2 and 3 we added task-related changes in $\boldsymbol{f}$ and $\boldsymbol{z}_{\mathrm{I}}$ respectively (maximum change of 1% and 0.15rad/s); (c) Additive spatially global drift in the field map to simulate the effects of magnetic field drift and respiration.

Figure 3.4(a) shows the spatial weights for four enumerated activation clusters, along with the edges of $\boldsymbol{f}$ in Figure 3.1(a) shown for reference. Figure 3.4(b) shows the task-related temporal changes in $\boldsymbol{z}_{\mathrm{R}}$ that were simulated in all the clusters. Additionally, in clusters 2 and 3 we added task correlated changes in $\boldsymbol{f}$ and $\boldsymbol{z}_{\mathrm{I}}$ respectively (maximum change of 1% and 0.15rad/s). Cluster 1 was placed where the in-plane field map gradient of $\boldsymbol{z}_{\mathrm{I}0}$ was large, while cluster 4 was placed along the edge of $\boldsymbol{f}$. Figure 3.4(c) shows the spatially global drift in $\boldsymbol{z}_{\mathrm{I}}$ that simulates the effects of magnetic field drift and respiration. The k-space time series was generated by adding these spatio-temporal changes to $\boldsymbol{z}_0$ and $\boldsymbol{f}$ in Figure 3.1, using the exact MR signal equation (3.4) and noise.

To choose L for the proposed reconstruction, we generated noiseless k-space data from 64×64 images to reconstruct $\hat{\boldsymbol{z}}_j$ with $L = 1, \ldots, 5$ by solving (3.14). The desired resolution was set as described in the first paragraph in Section 3.6.2 and we used the true $\boldsymbol{f}$ and $\boldsymbol{z}_0$ to exclude any effect from the initialization of the reconstruction. With the exception of the first time frame the algorithm had similar temporal RMSE for $L \geq 2$ (average temporal RMSE was 0.197). Hence we chose to use $L = 5$ for the first time frame and $L = 2$ for subsequent time frames when reconstructing the k-space data from 128×128 images. We also initialized the reconstruction using $\hat{\boldsymbol{f}}$ and $\hat{\boldsymbol{z}}_0$ from Section 3.6.1.

All the T_2^*-weighted reconstructed readouts used 20 CG iterations, with CG initialized using the conjugate phase reconstruction [17], and corrected for off-resonance in all time frames using $\hat{\boldsymbol{z}}_{\mathrm{I}0}$ in Figure 3.1(f). The regularization parameter was chosen to have a LPSF with FWHM of 1.35 so that the resolution was comparable to $\hat{\boldsymbol{z}}_{\mathrm{R}j}$ from the proposed reconstruction.

We reconstructed all the time frames for all SNRs. We did the time series analysis using a GLM model with the task waveform as a regressor and generated z-score maps that were thresholded with a Bonferroni corrected P-value of 0.01. Figure 3.5 shows the overlaid z-score of the voxels inside $\boldsymbol{f}$ that were above the threshold for SNR $= 80$ (left) and SNR $= 30$ (right). Voxels with true positives are shown with a color coded z-score



(a) (b)

(c) (d)

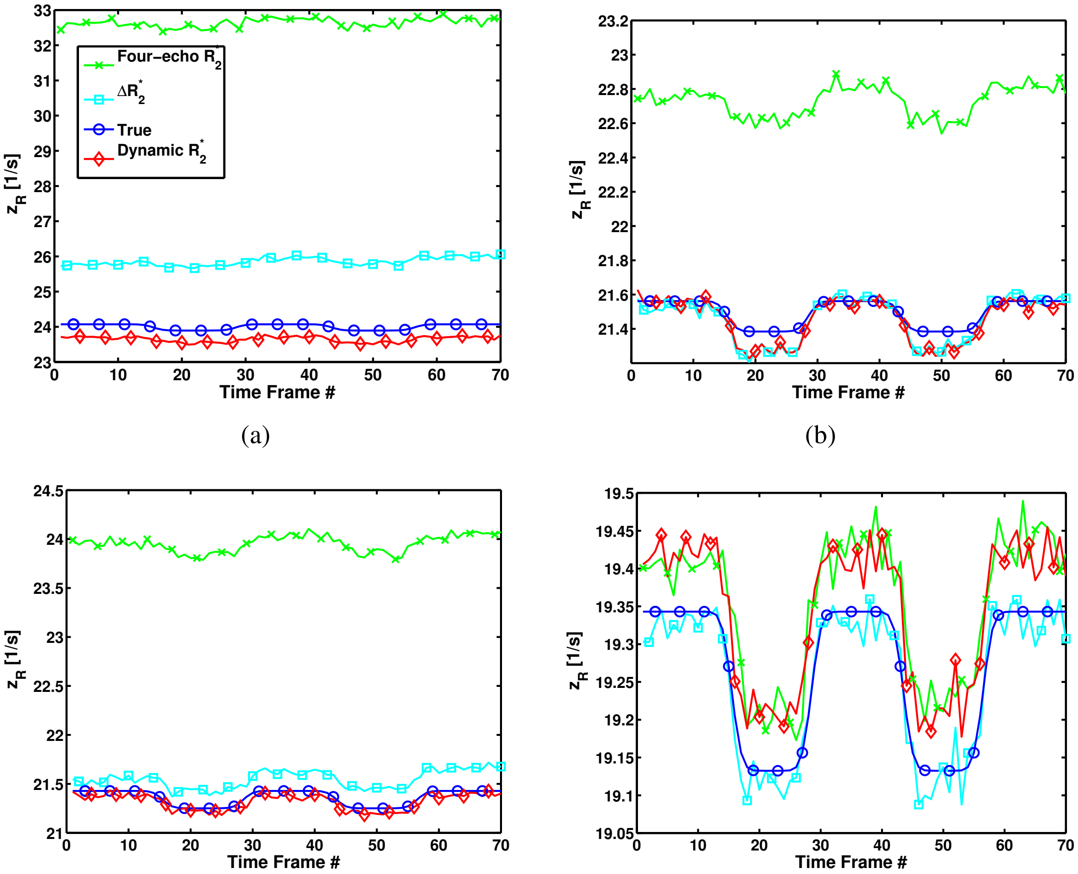
(e) (f)

Figure 3.5: Overlaid z-score of the voxels inside $\boldsymbol{f}$ that were above the threshold (Bonferroni corrected P-value of 0.01) for SNR $= 80$ (left) and SNR $= 30$ (right). (a)&(b) Dynamic $\hat{\boldsymbol{z}}_{\mathrm{R}j}$ reconstruction; (c)&(d) T_2^*-weighted reconstruction; (e)&(f) Multi-echo R_2^* reconstruction.

and false negatives with a square. Figure 3.5(a) - 3.5(b) shows the results from the dynamically reconstructed $\hat{\boldsymbol{z}}_j$, Figure 3.5(c) - 3.5(d) from the T_2^*-weighted reconstruction and Figure 3.5(e) - 3.5(f) from the multi-echo R_2^* reconstruction.

Figure 3.5 shows that in terms of total number of true positives the four-echo R_2^* reconstruction performs the worst. This is especially apparent in cluster 1 due to the high in-plane field map gradient. Compared to the T_2^*-weighted reconstruction, the proposed reconstruction performs slightly better for both SNRs, especially for voxels that have low functional CNR. This is particularly evident in cluster 3, that has the task correlated changes in the field map, and cluster 4. However, both these reconstructions are sensitive to the task correlated changes in $\boldsymbol{f}$ for cluster 2. The multi-echo R_2^* reconstruction is more robust to this effect. Similar trends are seen for $\mathrm{SNR} = 55$ as for the other SNRs.

Figure 3.6 shows R_2^* time series from the three reconstructions, spatially averaged over cluster 1-4 as shown in Figure 3.6(a) - 3.6(d) respectively. For reference the plots show the true spatially averaged $\boldsymbol{z}_{\mathrm{R}j}$ time series. To convert the T_2^*-weighted time series to R_2^* we calculated its ΔR_2^* time series using $\Delta R_2^* \approx -\left(S_j - S_1\right)/S_1/\mathrm{TE}$, where S_j is the T_2^*-weighted magnitude image of time frame j. This time series was then shifted by $\hat{\boldsymbol{z}}_{\mathrm{R}0}$. The plots show that the multi-echo R_2^* reconstruction performs the worst and the proposed reconstruction the best in estimating the R_2^* time series. This is obvious in Figure 3.6(a) due to the high in-plane field map gradient of cluster 1. Figure 3.6(b) shows how the simulated inflow changes in cluster 2 affect the R_2^* estimates during activation for both the proposed and T_2^*-weighted reconstructions. Additionally, all plots indicate a slight linear drift in the ΔR_2^* time series, as clearly seen in Figure 3.6(c). All the clusters have $<2\%$ error in estimating R_2^* time series for the proposed reconstruction.



(a) (b)

(c) (d)

Figure 3.6: Estimated R_2^* time series (SNR $= 55$) for all the reconstructions, which was spatially averaged over each cluster, along with the true $\boldsymbol{z}_{\mathrm{R}j}$. (a) Results from cluster 1; (b) Results from cluster 2; (c) Results from cluster 3; (d) Results from cluster 4.

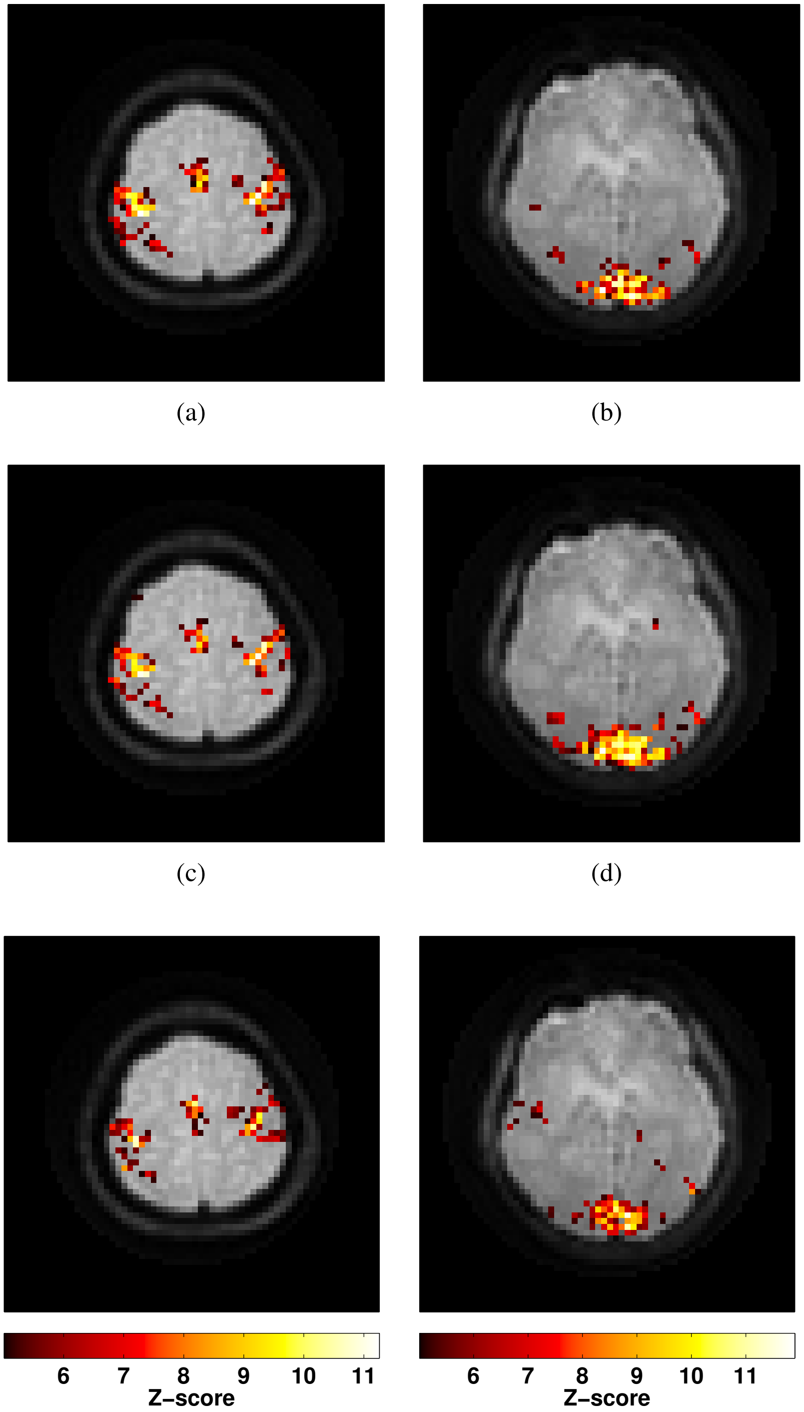
3.7 Experimental Data

We scanned a single subject in an fMRI experiment to demonstrate the proposed reconstruction for *in vivo* data and compared with T_2^*-weighted and multi-echo R_2^* reconstructions. The data had 102 time frames, four-echo spiral-out readouts, $\mathrm{FOV} = 24\mathrm{cm}$, $\mathrm{TR} = 3.2\mathrm{s}$ and $\mathrm{FA} = 90°$. The first time frame had $\mathrm{TE} = [4.6, 23.2, 41.8, 60.4]\mathrm{ms}$, the second time frame the same TEs shifted by 2ms and subsequent time frames had $\mathrm{TE} = [11.4, 30, 48.6, 67.2]\mathrm{ms}$. The subject was instructed to repeat 5 times 32s of rest followed by 32s of bilateral finger tapping prompted by a flickering checkerboard. We collected 26 axial slices that covered the visual and motor cortices.

The four readouts from the first time frame and the first readout of the second time frame were used to form a five-echo k-space data with $\mathrm{TE} = [4.6, 6.6, 23.2, 41.8, 60.4]\mathrm{ms}$. This five-echo data was used to estimate $\hat{\boldsymbol{f}}$ and $\hat{\boldsymbol{z}}_0$ identically to the procedure described in Section 3.6.1. For the proposed and T_2^*-weighted reconstructions we used only the $\mathrm{TE} = 30\mathrm{ms}$ readout while the multi-echo R_2^* reconstruction used all the readouts. All the reconstructions were set up identically to what was described in Section 3.6.

The fMRI analysis was applied to all the reconstructions, using a GLM model with a gamma-variate regressor [60] for the task and a linear regressor for the linear drift. We generated z-score maps that were thresholded using a Bonferroni corrected P-value of 0.01 followed by a clustering constraint [61] of at least one neighboring voxel above the threshold. Figure 3.7 shows 2 slices with overlaid z-scores of voxels above the threshold for all the reconstructions. The left column of Figure 3.7 shows a superior slice with motor activation and the right column shows an inferior slice with visual activation. Figure 3.7(a) - 3.7(b) shows the results from the proposed reconstruction, Figure 3.7(c) - 3.7(d) from the T_2^*-weighted reconstruction and Figure 3.7(e) - 3.7(f) from the four-echo R_2^* reconstruction.

Figure 3.7 shows clear activations in the motor cortex, supplementary motor area and the visual cortex for all the reconstructions. Also, $\boldsymbol{f}$ estimates from the four-echo R_2^* reconstruction showed only 3 voxels in the visual cortex having minor task correlated inflow and



(a) (b) (c) (d) (e) (f)

Figure 3.7: Z-scores of voxels above the threshold for 2 slices overlaid on anatomical images for all the reconstructions. Left column shows the results of a superior slice with motor activation and right column shows the results of an inferior slice with visual activation. (a)&(b) Dynamic $\hat{\boldsymbol{z}}_j$ reconstruction; (c)&(d) T_2^*-weighted reconstruction; (e)&(f) Four-echo R_2^* reconstruction.

none in the motor cortex. The results from the superior slice follows the trend shown in the simulations, where the proposed reconstruction has the most number of voxels classified as active and the four-echo R_2^* reconstruction the fewest. However, this is not the case for the inferior slice, where the T_2^*-weighted reconstruction has the most active voxels. This difference may be due to the stronger field gradients in the inferior slice that currently are not included in the signal model in (3.4).

3.8 Conclusion and Discussion

We have proposed a method for reconstructing dynamic R_2^* and field maps for fMRI data with the same temporal resolution as T_2^*-weighted imaging. This was done using a linear approximation to the temporal changes in R_2^* and field maps relative to a previously determined reference map. Simulations showed the reconstruction outperformed both T_2^*-weighted and four-echo R_2^* reconstructions in detecting active voxels. For the quantitative R_2^* estimation the proposed reconstruction did considerably better than the four-echo R_2^* reconstruction but about the same as the T_2^*-weighted reconstruction after converting it to ΔR_2^* and excluding the $\hat{\boldsymbol{z}}_{\mathrm{R0}}$ shift. However, there was a slight drift present in the ΔR_2^* time series, which may explain the lower detection performance of the T_2^*-weighted reconstruction compared to the proposed reconstruction. Adding a linear drift regressor into the GLM model did improve the performance of the T_2^*-weighted reconstruction, but still not to the level of the proposed reconstruction.

For the *in vivo* data all the reconstructions showed a similar trend to the simulation results for the superior slice, less so for the inferior slice. Since the inferior slice is closer to the sinuses it is more affected by field gradients than the superior slice. The effects of the gradients are not in the signal equation for any of the reconstructions used here. However, when compared to T_2^*-weighted reconstruction, the proposed reconstruction does rely on a mono-exponential relaxation model which can introduce model bias in areas with high gradients [54, 55]. Adding gradients in the signal equation in (3.4) [47] may also alleviate this model mismatch for the proposed reconstruction. This is something that needs further investigation. Any further comparisons in terms of detection performance for the *in vivo* data would need more acquisitions and estimation of test-retest reliability [62].

The compute time and the performance of the algorithm depends mainly on the number of refinements used to find $\hat{\boldsymbol{z}}_j$. It was shown in the simulations that $L = 5$ for the first time frame and $L = 2$ for other time frames was adequate. The compute time for the first time frame was 60.3s, which includes forming $\boldsymbol{D}$ (4.7s) and finding the β values to satisfy our desired resolution (19.2s), and subsequent time frames took 17.7s. This time could be reduced by reusing common parts of $\boldsymbol{A}$ for all time frames. The reconstruction is then two times longer than T_2^*-weighted reconstruction since $L = 2$, but with the added benefit of correcting for field drift and getting R_2^* estimates.

Currently, the algorithm does not include any motion correction. Since all the simulations were done without motion the performance of the algorithm with motion has not been assessed. However, since it relies on the reconstructed map $\hat{\boldsymbol{f}}$, one would assume that any motion in the data is going to translate into changes in $\hat{\boldsymbol{f}}$. This may make the algorithm more sensitive to motion induced errors than T_2^*-weighted image reconstruction, where all the frames can be reconstructed independent of past or future time frames.

One simple method for motion correction would be to estimate rigid body motion parameters from T_2^*-weighted images, using the first time frame as the reference frame. We would then use those motion parameters when reconstructing $\hat{\boldsymbol{z}}_j$ for the same data to correct $\hat{\boldsymbol{f}}$ in the system matrix for any motion. This method and others need to be investigated further but ultimately motion correction must be included in the algorithm for it to be robust.

In addition to excluding motion there is also an assumption of blood inflow being limited. Under this assumption $\hat{\boldsymbol{f}}$ should be time frame invariant when reconstructing $\hat{\boldsymbol{z}}_j$ for all j. One way to limit the effects of inflow enhancement is to acquire the data by either increasing TR or decreasing the flip angle [35], which respectively puts limitations on the data acquisition and lowers SNR. This effect could also be decreased by jointly reconstructing $\hat{\boldsymbol{f}}$ and $\hat{z}_j$. However, due to the higher dimensionality of this ill-conditioned reconstruction of both spatial maps, longer readouts and further regularization of $\hat{\boldsymbol{f}}$ would be needed. This would potentially be at a cost of higher compute times due to the increased complexity.

In our nonuniform regularization design there are mainly three limitations. Firstly, we form the spatially nonuniform penalty once and then only based on $\hat{\boldsymbol{z}}_{\mathrm{R}0}$. This can result in time frame varying resolution for voxels with significant temporal changes in $\boldsymbol{z}_j$. Additionally, any spatial resolution variations due to $\hat{\boldsymbol{z}}_{\mathrm{I}j}$ are not compensated since we only account for spatial variations in R_2^*. Secondly, we chose the regularization parameter arbitrarily to satisfy a desired FWHM. There are alternative methods available to choose this parameter [22], such as cross validation. Thirdly, since $\left|\hat{\boldsymbol{f}}\right|$ is multiplicative in $\boldsymbol{D}$, voxels with low $\left|\hat{f}(\vec{r}_n)\right|$ will have very little regularization. This could be alleviated by smoothly extending high $\left|\hat{f}(\vec{r}_n)\right|$ voxels over the low valued voxels to decrease the variance of the estimates. These design limitations need further investigation, especially with respect to resolution uniformity, reconstruction compute time and bias/variance trade-off.

The simulated and *in vivo* k-space data was acquired using a single shot spiral-out trajectory. The accuracy of the algorithm does depend on the readout length of the trajectory that is used to acquire the data. For instance, a fully sampled single shot spiral-in followed by an undersampled spiral-out was shown to have higher correlation values than using only spiral-in data [63]. Further investigation of the behavior of the algorithm for various trajectories and readout lengths is needed.

To initialize the algorithm we used data collected at the start of the fMRI study to reconstruct $\boldsymbol{z}_{\mathrm{I}0}$, then $\boldsymbol{z}_{\mathrm{R}0}$, and finally $\boldsymbol{f}$. The reconstruction of $\hat{\boldsymbol{f}}$ may be sensitive to errors in $\hat{\boldsymbol{z}}_0$, which may especially come from $\hat{\boldsymbol{z}}_{\mathrm{R}0}$ since log fitting is very sensitive to noise. An alternative solution would be to use a joint reconstruction algorithm [42, 43], where $\hat{\boldsymbol{z}}_0$ and

$\hat{\boldsymbol{f}}$ are simultaneously reconstructed by minimizing one regularized cost function instead of minimizing separate cost functions for each spatial map.

Chapter 4

Spatial Resolution Analysis of Quadratic Penalized Least-Squares with a Separate Real and Imaginary Roughness Penalty[1]

4.1 Introduction

Iterative image reconstruction methods are growing in popularity for many imaging devices. This usually involves iteratively optimizing a cost function that fits the acquired data to a physics-based signal model. Since the data received from the imaging device is generally contaminated with noise, the reconstructed images can have errors due to noise if the problem is badly conditioned. One way to improve the conditioning and thus reduce variance is to add a regularization to the cost function at the price of additional bias to the reconstructed image. For example, one common regularization function is the spatial roughness penalty, where the bias materializes as a reduction in spatial resolution.

For complex valued images, the conventional roughness penalty penalizes equally the real and imaginary parts. Sometimes, unconventional roughness penalties are needed, for example in separate regularization of magnitude and phase [64, 65] or the real and imaginary parts, which is the topic of this chapter. This type of regularization has been used in digital holography [66] and functional magnetic resonance imaging (fMRI) as seen in

---

[1] This chapter is based on [59].

Chapter 3.

The *regularization parameters* of the roughness penalty dictate how smooth the reconstructed image becomes and it can be challenging to choose these parameters sensibly. One method is to base that choice on a desired spatial resolution for the reconstructed image. Spatial resolution can be quantified as the full width half max (FWHM) of a *point spread function* (PSF), sometimes referred to as an *impulse response function* (IRF). To find the regularization parameters that achieve the desired resolution, the PSF and its FWHM is calculated for a wide range of values of the regularization parameters. This is followed by an interpolation from the desired FWHM to the resulting regularization parameters. By using those parameters in the reconstruction algorithm it should give images that have the desired resolution.

For computational efficiency it would be advantageous to only have to compute the PSF for one spatial location. This inherently assumes that the PSF is shift invariant, *i.e.*, that the PSF, and hence its FWHM, is uniform across the reconstructed image. However, for some image reconstruction algorithms the PSF is shift variant. This is undesirable since the reconstruction now introduces spatially varying resolution in the reconstructed images. By making the penalty appropriately spatially variant this resolution nonuniformity can be counteracted. But one is still left with evaluating the PSF multiple times to find the regularization parameters that achieve the desired resolution.

Some methods have been proposed to design spatially varying penalties based on *local impulse responses* [23, 24] to correct for spatially varying resolution. This has been used previously for a quadratic penalized least-squares (QPLS) cost function using the conventional roughness penalty [24]. By using a very reasonable approximation it was also shown that the local impulse response can be calculated using FFT to further reduce compute time. However, this approach is not directly applicable to the case when the real and imaginary parts of the image are to be regularized separately.

This chapter presents a fast method to calculate an approximate local impulse response for a separate real and imaginary roughness penalty. This allows us to use a computationally efficient algorithm to evaluate the local impulse response when finding the regularization parameters needed for a desired spatial resolution. This analysis can also be used to design a spatially varying roughness penalty, as was shown for the reconstruction in Chapter 3 to impose an approximately uniform resolution.

4.2 Local Impulse Response Using Separate Real and Imaginary Regularization

We assume the following discrete and linear model for the noisy acquired data vector $\boldsymbol{y}$:

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\epsilon}, \tag{4.1}$$

where vector $\boldsymbol{x}$ contains the unknown object that is being imaged, $\boldsymbol{A}$ is a linear transform from image space to acquired data space, sometimes called the system matrix, and vector $\boldsymbol{\epsilon}$ contains samples of noise in the acquired data. Quadratic penalized least-squares (QPLS) is a common method to estimate $\boldsymbol{x}$ from $\boldsymbol{y}$ using the model in (4.1) and can be solved efficiently using an iterative gradient descent algorithm, like conjugate gradient. A QPLS reconstruction problem can be described as follows:

$$\begin{aligned} \Psi(\boldsymbol{x}) &= \frac{1}{2}\left\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\right\|^2 + \mathrm{R}(\boldsymbol{x}) \\ \hat{\boldsymbol{x}} &= \arg\min_{\boldsymbol{x}} \Psi(\boldsymbol{x}), \end{aligned} \tag{4.2}$$

where $\frac{1}{2}\left\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\right\|^2$ is called the data fit term and $\mathrm{R}(\cdot)$ is a quadratic penalty, used to improve the overall conditioning of the problem.

Here, we would like to use a spatial roughness penalty that separately penalizes the real and imaginary parts of $\boldsymbol{x}$ as follows:

$$\mathrm{R}(\boldsymbol{x}) = \frac{1}{2}\left(\beta_1 \left\|\boldsymbol{C}_1\boldsymbol{x}_{\mathrm{R}}\right\|^2 + \beta_2 \left\|\boldsymbol{C}_2\boldsymbol{x}_{\mathrm{I}}\right\|^2\right), \tag{4.3}$$

where the subscripts R and I refer to the real and imaginary part of $\boldsymbol{x}$ respectively. Also, $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ are real valued differencing matrices that compute some order of finite differences (most common are first and second order) for $\boldsymbol{x}_{\mathrm{R}}$ and $\boldsymbol{x}_{\mathrm{I}}$ respectively and β_1 and β_2 are *regularization parameters* that control the tradeoff between noise and spatial resolution in $\hat{\boldsymbol{x}}$. We would like to develop a computationally fast framework that chooses β_1 and β_2 to achieve a desired spatial resolution for $\hat{\boldsymbol{x}}_{\mathrm{R}}$ and $\hat{\boldsymbol{x}}_{\mathrm{I}}$ in (4.2) based on its *local impulse response*.

Finding the local impulse response of a QPLS reconstruction involves calculating the first and second order gradients of $\Psi(\boldsymbol{x})$ [24]. Since (4.3) involves $\boldsymbol{x}_{\mathrm{R}}$ and $\boldsymbol{x}_{\mathrm{I}}$, it is not clear how these gradients would be formed when using that penalty. We address this problem by first considering a previous resolution analysis using the conventional roughness penalty term presented in [24] and secondly reformulate $\Psi(\boldsymbol{x})$ with (4.3) so that analysis can be used to easily derive the local impulse response of a QPLS reconstruction with the regularization in (4.3).

To review the resolution analysis in [24], lets suppose $\mathrm{R}(\cdot)$ is the *conventional* roughness penalty, given as follows:

$$\mathrm{R}(\boldsymbol{x}) = \frac{1}{2}\beta \left\|\boldsymbol{C}\boldsymbol{x}\right\|^2 = \frac{1}{2}\beta\left(\left\|\boldsymbol{C}\boldsymbol{x}_{\mathrm{R}}\right\|^2 + \left\|\boldsymbol{C}\boldsymbol{x}_{\mathrm{I}}\right\|^2\right), \tag{4.4}$$

where $\boldsymbol{C}$ is a differencing matrix and β is the regularization parameter. This penalty puts equal weight on the real and imaginary parts of $\boldsymbol{x}$ and thus can be considered a special case of (4.3) with $\beta_1 = \beta_2 = \beta$ and $\boldsymbol{C}_1 = \boldsymbol{C}_2 = \boldsymbol{C}$. Using this roughness penalty in (4.2), the gradients of $\Psi(\boldsymbol{x})$ are easily found and it has been shown in [24] that the local impulse response at spatial position n is given as follows:

$$\boldsymbol{l}_n = \left(\boldsymbol{A}'\boldsymbol{A} + \beta\boldsymbol{C}'\boldsymbol{C}\right)^{-1} \boldsymbol{A}'\boldsymbol{A}\boldsymbol{e}_n, \quad n = 1, \ldots, N \tag{4.5}$$

where $\boldsymbol{l}_n$ is the local impulse response and $\boldsymbol{e}_n$ is a column vector with 1 at vector element position n and zeros elsewhere (Kronecker impulse).

The method used to derive the local impulse response presented in (4.5) does not apply when using our preferred penalty in (4.3). Instead, we propose to form a *stacked* cost function with separate real and imaginary regularization. This stacked cost function looks like $\Psi(\boldsymbol{x})$ with the conventional penalty in (4.4) and thus allows us to derive the local impulse response of the stacked cost function using the type of analysis presented in [24].

4.2.1 Stacked Cost Function With Separate Real and Imaginary Regularization

We first rewrite all the matrices and vectors in a stacked format, as follows:

$$\boldsymbol{y}_{\mathrm{S}} = \begin{bmatrix} \boldsymbol{y}_{\mathrm{R}} \\ \boldsymbol{y}_{\mathrm{I}} \end{bmatrix}, \quad \boldsymbol{x}_{\mathrm{S}} = \begin{bmatrix} \boldsymbol{x}_{\mathrm{R}} \\ \boldsymbol{x}_{\mathrm{I}} \end{bmatrix},$$

$$\boldsymbol{A}_{\mathrm{S}} = \begin{bmatrix} \boldsymbol{A}_{\mathrm{R}} & -\boldsymbol{A}_{\mathrm{I}} \\ \boldsymbol{A}_{\mathrm{I}} & \boldsymbol{A}_{\mathrm{R}} \end{bmatrix}, \quad \boldsymbol{C}_{\mathrm{S}} = \begin{bmatrix} \boldsymbol{C}_{\mathrm{S1}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{C}_{\mathrm{S2}} \end{bmatrix},$$

where $\boldsymbol{C}_{\mathrm{S}}$ is block diagonal with $\boldsymbol{C}_{\mathrm{S1}} = \sqrt{\beta_1}\boldsymbol{C}_1$ and $\boldsymbol{C}_{\mathrm{S2}} = \sqrt{\beta_2}\boldsymbol{C}_2$. Note that β_1 and β_2 can be chosen independently of each other. Using these definitions for the stacked matrices and vectors we can form a new *stacked cost function* $\Psi_{\mathrm{S}}(\boldsymbol{x}_{\mathrm{S}})$ as follows:

$$\Psi_{\mathrm{S}}(\boldsymbol{x}_{\mathrm{S}}) = \frac{1}{2}\left\|\boldsymbol{y}_{\mathrm{S}} - \boldsymbol{A}_{\mathrm{S}}\boldsymbol{x}_{\mathrm{S}}\right\|^2 + \frac{1}{2}\left\|\boldsymbol{C}_{\mathrm{S}}\boldsymbol{x}_{\mathrm{S}}\right\|^2. \tag{4.6}$$

The original goal was to solve (4.2) with a roughness penalty that separately penalized the real and imaginary parts of $\boldsymbol{x}$. The question is, are we still effectively doing that by minimizing the stacked cost function in (4.6). For this to be true, we first need to show that the data fit terms of both cost functions are equivalent, *i.e.*, that $\|\boldsymbol{y}_{\mathrm{S}} - \boldsymbol{A}_{\mathrm{S}}\boldsymbol{x}_{\mathrm{S}}\|^2 = \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|^2$, and secondly that the stacked penalty in (4.6) is separately penalizing the real and imaginary parts of $\boldsymbol{x}$.

To show that $\|\boldsymbol{y}_{\mathrm{S}} - \boldsymbol{A}_{\mathrm{S}}\boldsymbol{x}_{\mathrm{S}}\|^2 = \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|^2$ we need two steps. Step one is to relate the error of the stacked vectors $\boldsymbol{y}_{\mathrm{S}} - \boldsymbol{A}_{\mathrm{S}}\boldsymbol{x}_{\mathrm{S}}$ to the error of the complex valued vectors $\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}$ as follows:

$$
\begin{aligned}
\boldsymbol{y}_{\mathrm{S}} - \boldsymbol{A}_{\mathrm{S}}\boldsymbol{x}_{\mathrm{S}} &= \begin{bmatrix} \boldsymbol{y}_{\mathrm{R}} \\ \boldsymbol{y}_{\mathrm{I}} \end{bmatrix} - \begin{bmatrix} \boldsymbol{A}_{\mathrm{R}} & -\boldsymbol{A}_{\mathrm{I}} \\ \boldsymbol{A}_{\mathrm{I}} & \boldsymbol{A}_{\mathrm{R}} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_{\mathrm{R}} \\ \boldsymbol{x}_{\mathrm{I}} \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{y}_{\mathrm{R}} - (\boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{R}} - \boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{I}}) \\ \boldsymbol{y}_{\mathrm{I}} - (\boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{R}} + \boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{I}}) \end{bmatrix} \\
&= \begin{bmatrix} \Re\left(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\right) \\ \Im\left(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\right) \end{bmatrix},
\end{aligned}
$$

where $\Re(\cdot)$ and $\Im(\cdot)$ are the real and imaginary parts respectively. From this we see that the error of the stacked vectors equals the stacked error of the original complex valued vectors. Step two is to show that the squared l_2 norm of a complex valued vector $\boldsymbol{r} = \boldsymbol{r}_{\mathrm{R}} + i\boldsymbol{r}_{\mathrm{I}}$ equals the squared l_2 norm of its stacked version $\boldsymbol{r}_{\mathrm{S}}$. This can be shown as follows:

$$
\begin{aligned}
\|\boldsymbol{r}\|^2 &= \left(\boldsymbol{r}_{\mathrm{R}}^{\mathrm{T}} - i\boldsymbol{r}_{\mathrm{I}}^{\mathrm{T}}\right)\left(\boldsymbol{r}_{\mathrm{R}} + i\boldsymbol{r}_{\mathrm{I}}\right) \\
&= \boldsymbol{r}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{r}_{\mathrm{R}} + \boldsymbol{r}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{r}_{\mathrm{I}} + i\left(\boldsymbol{r}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{r}_{\mathrm{I}} - \boldsymbol{r}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{r}_{\mathrm{R}}\right) \\
&= \boldsymbol{r}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{r}_{\mathrm{R}} + \boldsymbol{r}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{r}_{\mathrm{I}}, \quad \text{since } \boldsymbol{r}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{r}_{\mathrm{I}} = \boldsymbol{r}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{r}_{\mathrm{R}} \\
&= \|\boldsymbol{r}_{\mathrm{S}}\|^2.
\end{aligned}
$$

Using these two relationships between the stacked vectors and the complex valued vectors we see that:

$$\|\boldsymbol{y}_{\mathrm{S}} - \boldsymbol{A}_{\mathrm{S}}\boldsymbol{x}_{\mathrm{S}}\|^2 = \left\| \begin{bmatrix} \Re(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) \\ \Im(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) \end{bmatrix} \right\|^2 = \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|^2,$$

and hence that the data fit terms of $\Psi(\boldsymbol{x})$ and $\Psi_{\mathrm{S}}(\boldsymbol{x}_{\mathrm{S}})$ are equivalent.

The stacked penalty in (4.6) can be easily shown to equal our preferred penalty in (4.3), as follows:

$$\begin{aligned} \frac{1}{2}\|\boldsymbol{C}_{\mathrm{S}}\boldsymbol{x}_{\mathrm{S}}\|^2 &= \frac{1}{2}\boldsymbol{x}_{\mathrm{S}}'\boldsymbol{C}_{\mathrm{S}}'\boldsymbol{C}_{\mathrm{S}}\boldsymbol{x}_{\mathrm{S}} \\ &= \frac{1}{2}\begin{bmatrix} \boldsymbol{x}_{\mathrm{R}}' & \boldsymbol{x}_{\mathrm{I}}' \end{bmatrix} \begin{bmatrix} \beta_1\boldsymbol{C}_1'\boldsymbol{C}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \beta_2\boldsymbol{C}_2'\boldsymbol{C}_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_{\mathrm{R}} \\ \boldsymbol{x}_{\mathrm{I}} \end{bmatrix} \\ &= \frac{1}{2}\left(\beta_1\boldsymbol{x}_{\mathrm{R}}'\boldsymbol{C}_1'\boldsymbol{C}_1\boldsymbol{x}_{\mathrm{R}} + \beta_2\boldsymbol{x}_{\mathrm{I}}'\boldsymbol{C}_2'\boldsymbol{C}_2\boldsymbol{x}_{\mathrm{I}}\right) \\ &= \frac{1}{2}\left(\beta_1\|\boldsymbol{C}_1\boldsymbol{x}_{\mathrm{R}}\|^2 + \beta_2\|\boldsymbol{C}_2\boldsymbol{x}_{\mathrm{I}}\|^2\right). \end{aligned} \tag{4.7}$$

This shows that by minimizing $\Psi_{\mathrm{S}}(\boldsymbol{x}_{\mathrm{S}})$ in terms of $\boldsymbol{x}_{\mathrm{S}}$ we are able to enforce a separate real and imaginary regularization and still maintain the same data fit as when minimizing $\Psi(\boldsymbol{x})$. As before we could minimize the stacked cost function in (4.6) using an iterative gradient descent algorithm such as the conjugate gradient algorithm. This can be done efficiently as shown in Appendix A, but our focus here is to characterize the spatial resolution properties of (4.6).

4.2.2 Stacked Local Impulse Response

Since the stacked cost function in (4.6) looks just like the QPLS cost function using the conventional penalty in (4.4), we can use exactly the same methods as were used in [24] to derive the stacked local impulse response. This is written as follows:

$$\boldsymbol{l}_{\mathrm{S}n} = \left(\boldsymbol{A}_{\mathrm{S}}'\boldsymbol{A}_{\mathrm{S}} + \boldsymbol{C}_{\mathrm{S}}'\boldsymbol{C}_{\mathrm{S}}\right)^{-1}\boldsymbol{A}_{\mathrm{S}}'\boldsymbol{A}_{\mathrm{S}}\boldsymbol{e}_{\mathrm{S}n}, \tag{4.8}$$

where,

$$\begin{aligned} \boldsymbol{l}_{\mathrm{S}n} &= \begin{bmatrix} \boldsymbol{l}_{\mathrm{R}n} \\ \boldsymbol{l}_{\mathrm{I}n} \end{bmatrix}, \quad \boldsymbol{e}_{\mathrm{S}n} = \begin{bmatrix} (1-\alpha)\,\boldsymbol{e}_n \\ \alpha \boldsymbol{e}_n \end{bmatrix}, \quad \alpha \in \{0,1\} \\ \boldsymbol{A}_{\mathrm{S}}'\boldsymbol{A}_{\mathrm{S}} &= \begin{bmatrix} \Re\left(\boldsymbol{A}'\boldsymbol{A}\right) & -\Im\left(\boldsymbol{A}'\boldsymbol{A}\right) \\ \Im\left(\boldsymbol{A}'\boldsymbol{A}\right) & \Re\left(\boldsymbol{A}'\boldsymbol{A}\right) \end{bmatrix}, \quad \boldsymbol{C}_{\mathrm{S}}'\boldsymbol{C}_{\mathrm{S}} = \begin{bmatrix} \beta_1 \boldsymbol{C}_1'\boldsymbol{C}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \beta_2 \boldsymbol{C}_2'\boldsymbol{C}_2 \end{bmatrix}, \end{aligned} \tag{4.9}$$

where α is a variable that we set to 0 to calculate the point spread function of the real part and 1 for the imaginary part. Using (4.8) we can now characterize the spatial properties of a QPLS reconstruction that separately penalizes the real and imaginary parts.

4.3 Approximate Stacked Local Impulse Response

Evaluating (4.8) directly to calculate the real and imaginary local impulse responses could involve taking an inverse of $\boldsymbol{A}_{\mathrm{S}}'\boldsymbol{A}_{\mathrm{S}} + \boldsymbol{C}_{\mathrm{S}}'\boldsymbol{C}_{\mathrm{S}}$ which would be extremely time consuming and memory intensive. We would like to make this evaluation faster on the assumption that both $\boldsymbol{A}_{\mathrm{S}}'\boldsymbol{A}_{\mathrm{S}}$ and $\boldsymbol{C}_{\mathrm{S}}'\boldsymbol{C}_{\mathrm{S}}$ are locally circulant. This assumption allows us to approximate the inverse with an FFT followed by an inverse of a sparse matrix and finally an IFFT. This section describes the details of this fast approximation and verifies that the stability of the inverse is not compromised.

4.3.1 Fast Calculation

Suppose $\boldsymbol{C}_1'\boldsymbol{C}_1$, $\boldsymbol{C}_2'\boldsymbol{C}_2$ and $\boldsymbol{A}'\boldsymbol{A}$ are circulant, which makes $\Re\left(\boldsymbol{A}'\boldsymbol{A}\right)$ and $\Im\left(\boldsymbol{A}'\boldsymbol{A}\right)$ also circulant. It is well known that circulant matrices are diagonalizable using FFT transforms as follows:

$$\Re\left(\boldsymbol{A}'\boldsymbol{A}\right) = \boldsymbol{Q}\boldsymbol{\Lambda}_1\boldsymbol{Q}^{-1}, \quad \Im\left(\boldsymbol{A}'\boldsymbol{A}\right) = \boldsymbol{Q}\boldsymbol{\Lambda}_2\boldsymbol{Q}^{-1},$$
$$\boldsymbol{C}_1'\boldsymbol{C}_1 = \boldsymbol{Q}\boldsymbol{\Omega}_1\boldsymbol{Q}^{-1}, \quad \boldsymbol{C}_2'\boldsymbol{C}_2 = \boldsymbol{Q}\boldsymbol{\Omega}_2\boldsymbol{Q}^{-1}$$

where, $\boldsymbol{Q}^{-1}$ is the orthonormal 2D DFT matrix and $\boldsymbol{\Lambda}_1$, $\boldsymbol{\Lambda}_2$, $\boldsymbol{\Omega}_1$ and $\boldsymbol{\Omega}_2$ are diagonal matrices given by,

$$\begin{aligned}\boldsymbol{\Lambda}_1 &= \operatorname{diag}\left\{\boldsymbol{Q}^{-1}\Re\left(\boldsymbol{A}'\boldsymbol{A}\right)\boldsymbol{e}_1\right\}, \quad \boldsymbol{\Lambda}_2 = \operatorname{diag}\left\{\boldsymbol{Q}^{-1}\Im\left(\boldsymbol{A}'\boldsymbol{A}\right)\boldsymbol{e}_1\right\}, \\ \boldsymbol{\Omega}_1 &= \operatorname{diag}\left\{\boldsymbol{Q}^{-1}\boldsymbol{C}_1'\boldsymbol{C}_1\boldsymbol{e}_1\right\}, \quad \boldsymbol{\Omega}_2 = \operatorname{diag}\left\{\boldsymbol{Q}^{-1}\boldsymbol{C}_2'\boldsymbol{C}_2\boldsymbol{e}_1\right\}.\end{aligned} \tag{4.10}$$

Column n of a circulant matrix is a copy of its first column that has been circularly shifted n places. Using this in conjunction with the circular shift property of the DFT transform, the diagonal matrices in (4.10) can be equivalently calculated for any $\boldsymbol{e}_n$, by accounting for the added phase due to the shift, as follows:

$$\begin{aligned}\boldsymbol{\Lambda}_1 &= \operatorname{diag}\left\{\operatorname{diag}\left\{\mathrm{e}^{-i\angle\left(\boldsymbol{Q}^{-1}\boldsymbol{e}_n\right)}\right\}\boldsymbol{Q}^{-1}\Re\left(\boldsymbol{A}'\boldsymbol{A}\right)\boldsymbol{e}_n\right\}, \\ \boldsymbol{\Lambda}_2 &= \operatorname{diag}\left\{\operatorname{diag}\left\{\mathrm{e}^{-i\angle\left(\boldsymbol{Q}^{-1}\boldsymbol{e}_n\right)}\right\}\boldsymbol{Q}^{-1}\Im\left(\boldsymbol{A}'\boldsymbol{A}\right)\boldsymbol{e}_n\right\}, \\ \boldsymbol{\Omega}_1 &= \operatorname{diag}\left\{\operatorname{diag}\left\{\mathrm{e}^{-i\angle\left(\boldsymbol{Q}^{-1}\boldsymbol{e}_n\right)}\right\}\boldsymbol{Q}^{-1}\boldsymbol{C}_1'\boldsymbol{C}_1\boldsymbol{e}_n\right\}, \\ \boldsymbol{\Omega}_2 &= \operatorname{diag}\left\{\operatorname{diag}\left\{\mathrm{e}^{-i\angle\left(\boldsymbol{Q}^{-1}\boldsymbol{e}_n\right)}\right\}\boldsymbol{Q}^{-1}\boldsymbol{C}_2'\boldsymbol{C}_2\boldsymbol{e}_n\right\}.\end{aligned} \tag{4.11}$$

Using these FFT formulations we get for $\boldsymbol{A}_{\mathrm{S}}'\boldsymbol{A}_{\mathrm{S}}$ the following:

$$
\begin{aligned}
\boldsymbol{A}_{\mathrm{S}}'\boldsymbol{A}_{\mathrm{S}} &= \begin{bmatrix} \Re\left(\boldsymbol{A}'\boldsymbol{A}\right) & -\Im\left(\boldsymbol{A}'\boldsymbol{A}\right) \\ \Im\left(\boldsymbol{A}'\boldsymbol{A}\right) & \Re\left(\boldsymbol{A}'\boldsymbol{A}\right) \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{Q}\boldsymbol{\Lambda}_1\boldsymbol{Q}^{-1} & -\boldsymbol{Q}\boldsymbol{\Lambda}_2\boldsymbol{Q}^{-1} \\ \boldsymbol{Q}\boldsymbol{\Lambda}_2\boldsymbol{Q}^{-1} & \boldsymbol{Q}\boldsymbol{\Lambda}_1\boldsymbol{Q}^{-1} \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{Q} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{Q} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Lambda}_1 & -\boldsymbol{\Lambda}_2 \\ \boldsymbol{\Lambda}_2 & \boldsymbol{\Lambda}_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{Q}^{-1} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{Q}^{-1} \end{bmatrix} \\
&= \boldsymbol{Q}_{\mathrm{D}}\boldsymbol{\Lambda}_{\mathrm{S}}\boldsymbol{Q}_{\mathrm{D}}^{-1},
\end{aligned}
$$

where,

$$
\boldsymbol{Q}_{\mathrm{D}} \triangleq \begin{bmatrix} \boldsymbol{Q} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{Q} \end{bmatrix}, \quad \boldsymbol{\Lambda}_{\mathrm{S}} \triangleq \begin{bmatrix} \boldsymbol{\Lambda}_1 & -\boldsymbol{\Lambda}_2 \\ \boldsymbol{\Lambda}_2 & \boldsymbol{\Lambda}_1 \end{bmatrix}.
$$

In a similar manner we get for $\boldsymbol{C}_{\mathrm{S}}'\boldsymbol{C}_{\mathrm{S}}$ the following:

$$
\begin{aligned}
\boldsymbol{C}_{\mathrm{S}}'\boldsymbol{C}_{\mathrm{S}} &= \begin{bmatrix} \beta_1\boldsymbol{C}_1'\boldsymbol{C}_1 & \mathbf{0} \\ \mathbf{0} & \beta_2\boldsymbol{C}_2'\boldsymbol{C}_2 \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{Q} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{Q} \end{bmatrix} \begin{bmatrix} \beta_1\boldsymbol{\Omega}_1 & \mathbf{0} \\ \mathbf{0} & \beta_2\boldsymbol{\Omega}_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{Q}^{-1} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{Q}^{-1} \end{bmatrix} \\
&= \boldsymbol{Q}_{\mathrm{D}}\boldsymbol{\Omega}_{\mathrm{S}}\boldsymbol{Q}_{\mathrm{D}}^{-1},
\end{aligned}
$$

where,

$$
\boldsymbol{\Omega}_{\mathrm{S}} = \begin{bmatrix} \beta_1\boldsymbol{\Omega}_1 & \mathbf{0} \\ \mathbf{0} & \beta_2\boldsymbol{\Omega}_2 \end{bmatrix}.
$$

Using these equalities we can rewrite the stacked local impulse response in (4.8) as follows:

$$\begin{aligned}
\boldsymbol{l}_{\mathrm{S}n} &= \left(\boldsymbol{A}'_{\mathrm{S}}\boldsymbol{A}_{\mathrm{S}} + \boldsymbol{C}'_{\mathrm{S}}\boldsymbol{C}_{\mathrm{S}}\right)^{-1}\boldsymbol{A}'_{\mathrm{S}}\boldsymbol{A}_{\mathrm{S}}\boldsymbol{e}_{\mathrm{S}n} \\
&= \left(\boldsymbol{Q}_{\mathrm{D}}\boldsymbol{\Lambda}_{\mathrm{S}}\boldsymbol{Q}_{\mathrm{D}}^{-1} + \boldsymbol{Q}_{\mathrm{D}}\boldsymbol{\Omega}_{\mathrm{S}}\boldsymbol{Q}_{\mathrm{D}}^{-1}\right)^{-1}\boldsymbol{Q}_{\mathrm{D}}\boldsymbol{\Lambda}_{\mathrm{S}}\boldsymbol{Q}_{\mathrm{D}}^{-1}\boldsymbol{e}_{\mathrm{S}n} \\
&= \boldsymbol{Q}_{\mathrm{D}}\left(\boldsymbol{\Lambda}_{\mathrm{S}} + \boldsymbol{\Omega}_{\mathrm{S}}\right)^{-1}\boldsymbol{\Lambda}_{\mathrm{S}}\boldsymbol{Q}_{\mathrm{D}}^{-1}\boldsymbol{e}_{\mathrm{S}n}. \qquad (4.12)
\end{aligned}$$

Thus for the case when $\boldsymbol{C}'_1\boldsymbol{C}_1$, $\boldsymbol{C}'_2\boldsymbol{C}_2$ and $\boldsymbol{A}'\boldsymbol{A}$ are circulant, we have reduced the calculation of the stacked local impulse response considerably since it now involves only the inverse of $\boldsymbol{\Lambda}_{\mathrm{S}} + \boldsymbol{\Omega}_{\mathrm{S}}$, which are both sparse block matrices with diagonal blocks, and an FFT and IFFT operations.

For most imaging problems $\boldsymbol{C}'_1\boldsymbol{C}_1$, $\boldsymbol{C}'_2\boldsymbol{C}_2$ and $\boldsymbol{A}'\boldsymbol{A}$ are not exactly circulant. However, if they are *locally* circulant then (4.12) can be shown to be *locally* approximate. For these matrices to be locally circulant implies that their stacked local impulse response of one image voxel is similar to the response of its neighboring voxels. For example, a Toeplitz[2] matrix is locally circulant due to its circulant-like structure. Here, $\boldsymbol{C}'_1\boldsymbol{C}_1$ and $\boldsymbol{C}'_2\boldsymbol{C}_2$ are Toeplitz and although $\boldsymbol{A}'\boldsymbol{A}$ is generally not Toeplitz it is often approximately so in many MRI problems. Thus we can now locally approximate $\boldsymbol{C}'_1\boldsymbol{C}_1$, $\boldsymbol{C}'_2\boldsymbol{C}_2$ and $\boldsymbol{A}'\boldsymbol{A}$, and hence $\Re\left(\boldsymbol{A}'\boldsymbol{A}\right)$ and $\Im\left(\boldsymbol{A}'\boldsymbol{A}\right)$, as follows:

$$\begin{gathered}
\boldsymbol{C}'_1\boldsymbol{C}_1\boldsymbol{e}_n \approx \boldsymbol{Q}\boldsymbol{\Omega}_1\boldsymbol{Q}^{-1}\boldsymbol{e}_n, \quad \boldsymbol{C}'_2\boldsymbol{C}_2\boldsymbol{e}_n \approx \boldsymbol{Q}\boldsymbol{\Omega}_2\boldsymbol{Q}^{-1}\boldsymbol{e}_n, \\
\boldsymbol{A}'\boldsymbol{A}\boldsymbol{e}_n \approx \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^{-1}\boldsymbol{e}_n, \qquad (4.13) \\
\Re\left(\boldsymbol{A}'\boldsymbol{A}\right)\boldsymbol{e}_n \approx \boldsymbol{Q}\boldsymbol{\Lambda}_1\boldsymbol{Q}^{-1}\boldsymbol{e}_n, \quad \Im\left(\boldsymbol{A}'\boldsymbol{A}\right)\boldsymbol{e}_n \approx \boldsymbol{Q}\boldsymbol{\Lambda}_2\boldsymbol{Q}^{-1}\boldsymbol{e}_n,
\end{gathered}$$

where $\boldsymbol{\Omega}_1$, $\boldsymbol{\Omega}_1$. $\boldsymbol{\Lambda}_1$ and $\boldsymbol{\Lambda}_2$ are defined in (4.11) and $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}_1 + i\boldsymbol{\Lambda}_2$. We note that (4.13)

---

[2] For simplicity we use "Toeplitz" instead of "block Toeplitz with Toeplitz block"

implies we can approximate the stacked local impulse response as given in (4.12) such that:

$$\boldsymbol{l}_{\mathrm{S}n} \approx \boldsymbol{Q}_{\mathrm{D}} \left(\boldsymbol{\Lambda}_{\mathrm{S}} + \boldsymbol{\Omega}_{\mathrm{S}}\right)^{-1} \boldsymbol{\Lambda}_{\mathrm{S}} \boldsymbol{Q}_{\mathrm{D}}^{-1} \boldsymbol{e}_{\mathrm{S}n}. \tag{4.14}$$

4.3.2 Stability Analysis of the Stacked Impulse Response

We know that $\boldsymbol{C}_1'\boldsymbol{C}_1$, $\boldsymbol{C}_2'\boldsymbol{C}_2$ and $\boldsymbol{A}'\boldsymbol{A}$ are nonnegative definite, and thus have real and nonnegative eigenvalues. The matrix $\boldsymbol{\Omega}_{\mathrm{S}}$ as defined in (4.11) is thus approximately nonnegative definite since its elements are approximately the eigenvalues of $\boldsymbol{C}_1'\boldsymbol{C}_1$ and $\boldsymbol{C}_2'\boldsymbol{C}_2$. However, it is not obvious if $\boldsymbol{\Lambda}_{\mathrm{S}}$ is nonnegative definite and if that were not the case, there would exist β_1 and β_2 such that the inverse of $\boldsymbol{\Lambda}_{\mathrm{S}} + \boldsymbol{\Omega}_{\mathrm{S}}$ would be undefined. This could cause stability issues when calculating the stacked local impulse response. To investigate this we need to relate the eigenvalues of $\boldsymbol{\Lambda}_{\mathrm{S}}$ to the eigenvalues of $\boldsymbol{A}'\boldsymbol{A}$ under the local impulse response approximation given in (4.13).

We first need to express the elements of $\boldsymbol{\Lambda}_1$ and $\boldsymbol{\Lambda}_2$ as functions of the elements of $\boldsymbol{\Lambda}$. Using the approximation in (4.13), we have:

$$\begin{aligned}
\boldsymbol{Q}^{-1}\Re\left(\boldsymbol{A}'\boldsymbol{A}\right)\boldsymbol{e}_n &= \boldsymbol{Q}^{-1}\Re\left(\boldsymbol{A}'\boldsymbol{A}\boldsymbol{e}_n\right) \\
&\approx \boldsymbol{Q}^{-1}\Re\left(\boldsymbol{Q}^{-1}\boldsymbol{\Lambda}\boldsymbol{Q}^{-1}\boldsymbol{e}_n\right) \\
&= \boldsymbol{Q}^{-1}\frac{\boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^{-1}\boldsymbol{e}_n + \left(\boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^{-1}\boldsymbol{e}_n\right)^*}{2} \\
&= \frac{1}{2}\boldsymbol{v} + \frac{1}{2}\boldsymbol{Q}^{-1}\left(\boldsymbol{Q}\boldsymbol{v}\right)^*,
\end{aligned} \tag{4.15}$$

where $\boldsymbol{\Lambda} = \mathrm{diag}\{\boldsymbol{\lambda}\}$ and $\boldsymbol{v}$ is a vector with kth element $v_k = \lambda_k\, \mathrm{e}^{i\angle\left(\boldsymbol{Q}^{-1}\boldsymbol{e}_n\right)_k}$ and * is the conjugate operator. The expression above involves a conjugate of the IDFT of $\boldsymbol{v}$, which for element n in the vector simplifies as follows:

$$\left[(\boldsymbol{Q}\boldsymbol{v})^*\right]_n = \left(\sum_{k=0}^{K-1} Q_{nk}v_k\right)^* = \sum_{k=0}^{K-1} Q_{nk}^* v_k^* = \sum_{k=0}^{K-1} Q_{nk} v_{\underline{k}}^* = \left[\boldsymbol{Q}\boldsymbol{v}_-^*\right]_n,$$

where $\underline{k}$ is the index of the conjugate pair that goes with k and $\boldsymbol{v}_-$ has those elements. Thus $(\boldsymbol{Q}\boldsymbol{v})^*$ can be rewritten as an IDFT of a circularly shifted version of $\boldsymbol{v}$. Using this in (4.15), we now get for the kth element,

$$\begin{aligned}\left[\boldsymbol{Q}^{-1}\Re\left(\boldsymbol{A}'\boldsymbol{A}\right)\boldsymbol{e}_n\right]_k &\approx \frac{1}{2}\left(v_k + v_{\underline{k}}^*\right) \\ &= \frac{1}{2}\left(\lambda_k \, \mathrm{e}^{i\angle\left(\boldsymbol{Q}^{-1}\boldsymbol{e}_n\right)_k} + \lambda_{\underline{k}} \, \mathrm{e}^{-i\angle\left(\boldsymbol{Q}^{-1}\boldsymbol{e}_n\right)_{\underline{k}}}\right).\end{aligned}$$

Using this expression, the diagonal elements of $\boldsymbol{\Lambda}_1$ (4.11), which we write here as λ_{1k}, can be expressed in terms of the eigenvalues of $\boldsymbol{A}'\boldsymbol{A}$ as follows,

$$\lambda_{1k} \approx \mathrm{e}^{-i\angle\left(\boldsymbol{Q}^{-1}\boldsymbol{e}_n\right)_k} \frac{1}{2}\left(v_k + v_{\underline{k}}^*\right) = \frac{1}{2}\left(\lambda_k + \lambda_{\underline{k}}\right). \tag{4.16}$$

This ensures the eigenvalues of $\Re\left(\boldsymbol{A}'\boldsymbol{A}\right)$ are approximately real and nonnegative under the local impulse response approximation. Similarly, the eigenvalues of $\Im\left(\boldsymbol{A}'\boldsymbol{A}\right)$ under the local impulse response approximation can be written in terms of the eigenvalues of $\boldsymbol{A}'\boldsymbol{A}$ as follows:

$$\lambda_{2k} \approx \frac{1}{2i}\left(\lambda_k - \lambda_{\underline{k}}\right), \tag{4.17}$$

which makes the eigenvalues of $\Im\left(\boldsymbol{A}'\boldsymbol{A}\right)$ to be approximately purely imaginary.

Now we use (4.16) and (4.17) to form $\boldsymbol{\Lambda}_{\mathrm{S}}$, which is a 2×2 block matrix where each block is a diagonal matrix and the main diagonal is real and off-diagonals are strictly imaginary. However, this does not tell us yet if the eigenvalues of $\boldsymbol{\Lambda}_{\mathrm{S}}$ are real and non-negative. By using a well know property that eigenvalues of a matrix are invariant to elementary row and column operation, we can use those operations to make $\boldsymbol{\Lambda}_{\mathrm{S}}$ into a block diagonal matrix with K matrices that are each 2×2 matrices, where matrix number k is given as follows:

$$\begin{bmatrix} \lambda_{1k} & -\lambda_{2k} \\ \lambda_{2k} & \lambda_{1k} \end{bmatrix}.$$

Since all the submatrices along the diagonal are the same, we only need to find the eigenvalues of that to see if those are real and nonnegative and hence all the eigenvalues of $\mathbf{\Lambda}_{\mathrm{S}}$ are also. We find the eigenvalues α by solving for the characteristic equations as follows:

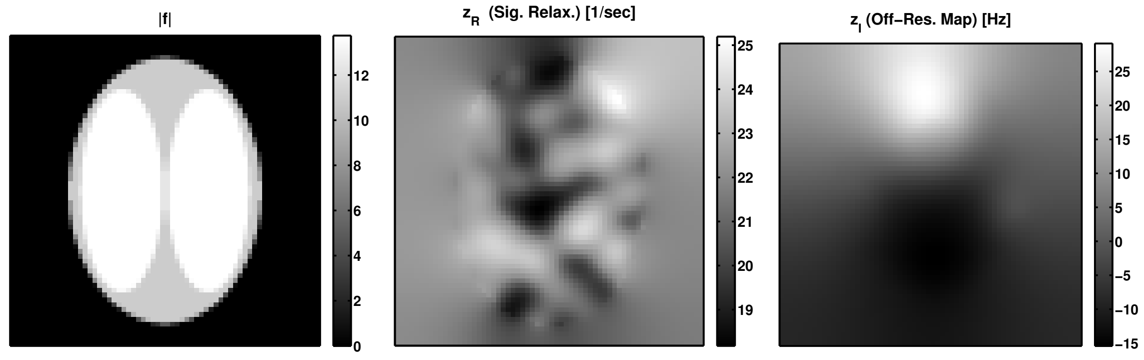
$$\begin{gathered}\begin{vmatrix} \lambda_{1k} - \alpha & -\lambda_{2k} \\ \lambda_{2k} & \lambda_{1k} - \alpha \end{vmatrix} = 0 \\ \Rightarrow \alpha^2 - 2\lambda_{1k}\alpha + \lambda_{1k}^2 + \lambda_{2k}^2 = 0.\end{gathered}$$

Using (4.16) and (4.17) the two roots $\alpha_{1,2}$ that solve this equality are given as follows:

$$\begin{aligned}\alpha_{1,2} &= \frac{2\lambda_{1k} \pm \sqrt{4\lambda_{1k}^2 - 4\left(\lambda_{1k}^2 + \lambda_{2k}^2\right)}}{2} \\ &= \frac{2\lambda_{1k} \pm \sqrt{-4\lambda_{2k}^2}}{2} \\ &\approx \frac{1}{2}\left(\lambda_k + \lambda_{\underline{k}}\right) \pm \frac{1}{2}\sqrt{\left(\lambda_k - \lambda_{\underline{k}}\right)^2} \\ &= \frac{1}{2}\left(\lambda_k + \lambda_{\underline{k}}\right) \pm \frac{1}{2}\left(\lambda_k - \lambda_{\underline{k}}\right).\end{aligned}$$

If $\boldsymbol{A}'\boldsymbol{A}$ has real and nonnegative eigenvalues λ_k under the local impulse response approximation, we see that $\alpha_{1,2}$ and hence the eigenvalues of $\mathbf{\Lambda}_{\mathrm{S}}$ are also approximately real and nonnegative. This makes the matrix sum $\mathbf{\Lambda}_{\mathrm{S}} + \mathbf{\Omega}_{\mathrm{S}}$ invertible and thus the local impulse response exists using the proposed fast calculation.

This implies that to ensure the stability of the approximation in (4.14) we may need to enforce real and nonnegative λ_k values. This is usually done in local impulse response calculations. In addition to this the same may also need to be enforced for the elements of $\mathbf{\Omega}_{\mathrm{S}}$.



(a) (b) (c)

Figure 4.1: Simulation maps used to evaluate the accuracy of the local point spread function (LPSF) approximation when $\boldsymbol{A}'\boldsymbol{A}$ is not Toeplitz but approximately so. (a) Initial magnetization $\boldsymbol{f}$; (b) R_2^* map $\boldsymbol{z}_{\mathrm{R}}$; (c) Field map $\boldsymbol{z}_{\mathrm{I}}$.

4.3.3 Implementation Details

To implement a fast version of the approximate stacked local impulse response approximation in (4.14), we need to form $\boldsymbol{\Lambda}_{\mathrm{S}}$ and $\boldsymbol{\Omega}_{\mathrm{S}}$ fast and efficiently. The key to that is to use (4.11) to form $\boldsymbol{\Omega}_1$ and $\boldsymbol{\Omega}_2$. However, to form $\boldsymbol{\Lambda}_{\mathrm{S}}$, we first find $\boldsymbol{\Lambda}$ by taking an FFT of $\boldsymbol{A}'\boldsymbol{A}\boldsymbol{e}_n$ as per (4.13). We can then form $\boldsymbol{\Lambda}_1$ and $\boldsymbol{\Lambda}_2$ using (4.16) and (4.17) respectively. Since $\boldsymbol{\Lambda}_{\mathrm{S}}$ and $\boldsymbol{\Omega}_{\mathrm{S}}$ are both sparse the inverse can be performed very quickly.

To maintain the stability of the algorithm, we enforce the FFT coefficients of $\boldsymbol{C}_1'\boldsymbol{C}_1$, $\boldsymbol{C}_2'\boldsymbol{C}_2$ and $\boldsymbol{A}'\boldsymbol{A}$ to be strictly real and nonnegative. As per the previous stability discussion, this ensures that $\boldsymbol{\Lambda}_{\mathrm{S}}$ and $\boldsymbol{\Omega}_{\mathrm{S}}$ have also real and nonnegative eigenvalues.

4.4 Simulations

To evaluate the accuracy of the local impulse response approximation proposed in (4.12) we use as an example a system matrix from the reconstruction discussed in Chapter 3. There we wanted to reconstruct changes in R_2^* relaxation rate map $\boldsymbol{z}_{\mathrm{R}}$ and field map $\boldsymbol{z}_{\mathrm{I}}$ using a stacked cost function as in (4.6). There, $\boldsymbol{A}$ in the signal model (4.1) has the following elements:

$$a_{mn} = f(\vec{r}_n)\, \mathrm{e}^{-t_m \check{z}(\vec{r}_n)}\, t_m\, \mathrm{e}^{-i2\pi(k_m \cdot \vec{r}_n)}\,,$$

$$\boldsymbol{f} \triangleq [f(\vec{r}_1)\,, \ldots, f(\vec{r}_N)]\,, \quad \check{\boldsymbol{z}} \triangleq [\check{z}(\vec{r}_1)\,, \ldots, \check{z}(\vec{r}_N)]\,,$$

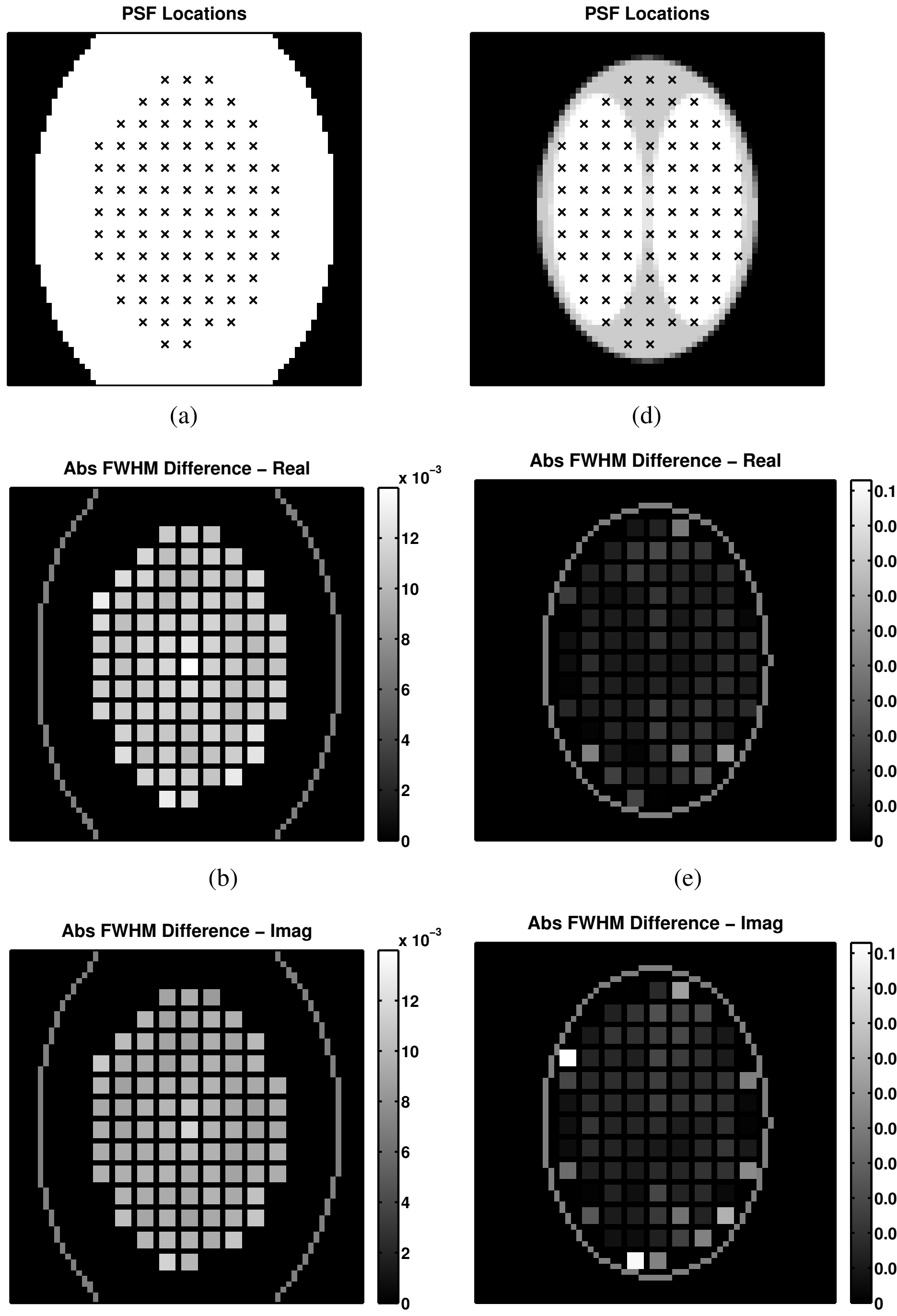
where $\boldsymbol{f}$ is the magnetization after excitation, $\check{\boldsymbol{z}}$ (assumed to be known) is the complex valued spatial map of the *reference* R_2^* (real part) and field map (imaginary part) and k_m is sample number m of the k-space trajectory used to acquire the MR data, which in this case is a spiral.

We simulated two cases to evaluate the accuracy and performance of the approximate stacked local impulse response in (4.14) compared to its exact form in (4.8). The first case had $\boldsymbol{f} = \mathbf{1}$ within the reconstruction mask and $\check{\boldsymbol{z}} = \mathbf{0}$ so that $\boldsymbol{A}'\boldsymbol{A}$ was Toeplitz, and hence locally circulant. The second case had a more realistic $\boldsymbol{f}$ and $\check{\boldsymbol{z}}$ as shown in Figure 4.1(a) - 4.1(c), for which $\boldsymbol{A}'\boldsymbol{A}$ is only approximately Toeplitz.

We calculate the exact stacked local impulse response by fully forming $\boldsymbol{A}_{\mathrm{S}}'\boldsymbol{A}_{\mathrm{S}}$ and $\boldsymbol{C}_{\mathrm{S}}'\boldsymbol{C}_{\mathrm{S}}$ and computed the approximate one as described in Section 4.3.3 where $\boldsymbol{A}$ was formed in a fast and memory efficient version manner [20, 25]. We chose the regularization to be first-order differences for both the real and imaginary parts. The regularization parameters β_1 and β_2 were chosen so that the fast approximate stacked local impulse response gave a FWHM of 1.35 and 1.5 voxels for the real and imaginary parts respectively.

For the case when $\boldsymbol{A}'\boldsymbol{A}$ is Toeplitz, Figure 4.2(a) shows the locations where the stacked local impulse response and its approximation were evaluated. Since FWHM of the impulse response is commonly used for resolution evaluation, the absolute difference in FWHM calculated using the exact and approximate stacked local impulse response is shown in Figure 4.2(b) - 4.2(c) for the real and imaginary parts respectively. Figure 4.2(d) - 4.2(f) shows the same for the case when $\boldsymbol{A}'\boldsymbol{A}$ is approximately Toeplitz.

For the exact method the average FWHM for the real and imaginary parts was 1.36



(a) (d)

(b) (e)

(c) (f)

Figure 4.2: FHWM difference when evaluating it based on the stacked local impulse response and its approximation. The results when $\boldsymbol{A}'\boldsymbol{A}$ is Toeplitz are shown in (a)-(c) and when it is approximately Toeplitz in (d)-(f). (a)&(d) Spatial positions where the stacked local impulse response and its approximation were evaluated rendered on top of $\boldsymbol{f}$; Absolute FWHM difference between the local impulse response and its approximation for (b)&(e) $\boldsymbol{z}_{\mathrm{R}}$ and (c)&(f) $\boldsymbol{z}_{\mathrm{I}}$, with the outline of $\boldsymbol{f}$ shown for reference.

and 1.51 voxels respectively when $\boldsymbol{A}'\boldsymbol{A}$ was Toeplitz and 1.36 and 1.51 voxels respectively when $\boldsymbol{A}'\boldsymbol{A}$ was approximately Toeplitz. Comparing Figure 4.2(b) - 4.2(c) and Figure 4.2(e) - 4.2(f) we do note that the FWHM difference between the approximation and exact impulse response is greater when $\boldsymbol{A}'\boldsymbol{A}$ is approximately Toeplitz. This is also apparent from the root mean squared (RMS) FWHM difference of the real and imaginary parts, which is respectively only 0.013 and 0.009 when $\boldsymbol{A}'\boldsymbol{A}$ is Toeplitz but 0.018 and 0.028 when $\boldsymbol{A}'\boldsymbol{A}$ is approximately Toeplitz. Despite this, the FWHM difference for both cases is under 3%.

In terms of compute performance, the exact method took about 2200 seconds to calculate the local impulse response for all spatial positions shown in Figure 4.2 but the approximate method took only about 13 seconds. This is a huge compute acceleration and is even more impressive considering that on average the accuracy is within 3% of the FWHM of the exact local impulse response.

4.5 Discussion

We presented a fast method to approximate the stacked local impulse response for an iterative reconstruction algorithm, where we separately penalize the real and the imaginary parts of the image. This method was shown to be stable, fast and accurate in estimating the local impulse response at various spatial positions. This was indicated by the comparisons of simulation data for the approximate and exact stacked local impulse response, where the approximation was shown to have a RMS error of around 2% and approximately 170 times faster.

From the simulation results in Figure 4.2 we note that the biggest errors in the FWHM values seem to be located close to the edges of $\boldsymbol{f}$. This would indicate that the approximate local impulse response does not hold well there, *i.e.*, $\boldsymbol{A}'\boldsymbol{A}$ is not locally circulant in those areas. This seems to indicate a weakness in the approximation. This is further supported from the fact that if we exclude the FWHM differences that are closest to the edge of $\boldsymbol{f}$ the

RMS FWHM difference reduces from 0.018 to 0.015 for the real part and from 0.028 to 0.018 for the imaginary part.

At first this would seem to be a problem that is unavoidable. However, from the resolution analysis of the reconstruction presented in Chapter 3, we note that by using a spatially variant penalty, any spatial nonuniformity caused by $\boldsymbol{f}$ can be accounted for. In fact we can generally state that using a spatially variant penalty allows for an evaluation of the local impulse response using a system matrix that is Toeplitz and hence the problem of non-Toeplitzness can be circumvented.

Finally it should be mentioned that a further analysis of other separate penalties also needs to be analyzed, such as separate magnitude and phase penalties.

Chapter 5

Fast and Motion Robust Dynamic R_2^* Reconstruction for Functional MRI[1]

5.1 Introduction

This chapter builds on the work presented in Chapter 3, where we introduced a fast iterative reconstruction of the complex valued rate map $\boldsymbol{z}$, whose real and imaginary parts are R_2^* and field maps, for functional MRI (fMRI) data. In its original form the signal equation in terms of $\boldsymbol{z}$ has a nonlinear relationship to the acquired k-space data. To make the reconstruction fast we made a linear approximation to the signal equation around a known reference map $\check{\boldsymbol{z}}$. The approximation was then used to form a new cost function that was quadratic in terms of the current rate map. This cost function could be minimized using a fast iterative algorithm such as conjugate gradient (CG). Once a new rate map had been estimated, we could use that as a new reference and repeat the minimization. This could be repeated until the original cost function converged. If the data consists of a time series of images, such as in fMRI, we can use the $\boldsymbol{z}$ estimated for a prior time frame as an initial reference map for the current time frame.

One of the major faults of this method was that it assumed that the initial magnetization $\boldsymbol{f}$ of the object was static for all the time frames. This is only true in the absence of subject motion and blood inflow changes (both flow and volume) while acquiring the fMRI data.

---

[1] This chapter is based on [67, 68].

Although these factors can be controlled to some degree, they are close to impossible to completely eliminate. For this reason we developed a fast image reconstruction algorithm that builds on the work in Chapter 3, but also reconstructs $\boldsymbol{f}$ dynamically to control for motion and inflow effects.

The structure of this chapter is as follows. We start by defining the MR signal equation and form a penalized least squares cost function that we would like to minimize in terms of $\boldsymbol{f}$ and $\boldsymbol{z}$. As previously stated this cost function is nonlinear in terms of $\boldsymbol{z}$, but is linear in terms of $\boldsymbol{f}$ (see Section 5.2). Next we use the linear approximation to the MR signal equation and from that form a new penalized cost function that approximates the original cost function. We propose to minimize this new cost function using an alternating minimization algorithm and review the quadratic penalties and the dual echo k-space trajectory we propose to use for this reconstruction (see Section 5.3). Then we simulate fMRI data and investigate the performance of the proposed joint reconstruction and compare it to the conventional iterative reconstruction usually used for fMRI data (see Section 5.4). Finally we look at some phantom MR data and human fMRI data corrupted by motion to evaluate the performance of the reconstruction (see Section 5.5).

5.2 Model Based Joint Reconstruction of R_2^*, Field Map and Magnetization

In functional magnetic resonance imaging (fMRI) the MR signal $s(t; z(\vec{r}), f(\vec{r}))$, including R_2^* relaxation and off-resonance, is conventionally modeled as follows:

$$s(t; z(\vec{r}), f(\vec{r})) = \int f(\vec{r})\, \mathrm{e}^{-tz(\vec{r})}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}\right)}\, \mathrm{d}\vec{r}, \tag{5.1}$$

where $f(\vec{r})$ is a map of the initial magnetization after excitation, $\vec{k}(t)$ is the k-space trajectory, $\vec{r}$ is the spatial coordinate and $z(\vec{r})$ is the complex valued rate map:

$$z(\vec{r}) = z_{\mathrm{R}}(\vec{r}) + iz_{\mathrm{I}}(\vec{r}) \,,$$

where the real valued $z_{\mathrm{R}}(\vec{r})$ and $z_{\mathrm{I}}(\vec{r})$ are the R_2^* relaxation and field maps respectively. The measured MR data $\boldsymbol{y} = [y_1, \ldots, y_M]$ from an MRI scanner is modeled as (5.1) contaminated with additive iid Gaussian noise [6] as follows:

$$y_m = s(t_m; z(\vec{r}) \,, f(\vec{r})) + \epsilon_m, \quad m = 1, \ldots, M. \tag{5.2}$$

We would like to jointly reconstruct $f(\vec{r})$ and $z(\vec{r})$ from $\boldsymbol{y}$ by minimizing a regularized cost function. This allows us to estimate an R_2^* map that is robust to noise, motion, blood inflow and off-resonance. The final goal is then to use the resulting R_2^* maps for fMRI analysis.

Jointly reconstructing the continuous-space $f(\vec{r})$ and $z(\vec{r})$ maps from the discrete $\boldsymbol{y}$ is an ill-posed inverse problem. By parametrizing both maps as follows:

$$f(\vec{r}) \approx \sum_{n=1}^{N} f(\vec{r}_n) \, b(\vec{r} - \vec{r}_n) \,, \quad z(\vec{r}) \approx \sum_{n=1}^{N} z(\vec{r}_n) \, b(\vec{r} - \vec{r}_n) \,, \tag{5.3}$$

where $b(\cdot)$ is a spatial basis function, this reconstruction is simplified to the point of being computationally tractable. Instead of estimating spatially continuous maps, the problem is reduced to reconstructing all the complex valued parameters $f(\vec{r}_n)$ and $z(\vec{r}_n)$. Here, $b(\cdot)$ is assumed to be the 2D or 3D rect function (the voxel basis), making $f(\vec{r}_n)$ and $z(\vec{r}_n)$ the height of a voxel basis with its center at $\vec{r}_n$.

If (5.3) is used in (5.1), we get the discrete-space MR signal $s(t; \boldsymbol{z}, \boldsymbol{f})$ as follows:

$$s(t; \boldsymbol{z}, \boldsymbol{f}) = \Phi(\vec{k}(t)) \sum_{n=1}^{N} f(\vec{r}_n)\, \mathrm{e}^{-t z(\vec{r}_n)}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)}, \tag{5.4}$$
$$\boldsymbol{z} = [z(\vec{r}_1), \ldots, z(\vec{r}_N)],\ \boldsymbol{f} = [f(\vec{r}_1), \ldots, f(\vec{r}_N)],$$

where $\Phi(\vec{k}(t))$ is the Fourier transform of the basis function $b(\cdot)$. This form of the signal equation can now be rewritten in a matrix-vector format as follows:

$$\boldsymbol{s}(\boldsymbol{z}, \boldsymbol{f}) = \boldsymbol{A}(\boldsymbol{z})\, \boldsymbol{f}, \tag{5.5}$$

where the elements $a_{mn}(\boldsymbol{z})$ of the *system matrix* $\boldsymbol{A}(\boldsymbol{z})$ are:

$$a_{mn}(\boldsymbol{z}) = \Phi(\vec{k}(t_m))\, \mathrm{e}^{-t_m z(\vec{r}_n)}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t_m)\cdot\vec{r}_n\right)}, \quad m = 1, \ldots, M,\ n = 1, \ldots, N. \tag{5.6}$$

Using (5.5), we can now write the acquired MR data from (5.2) in a matrix-vector format:

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{z})\, \boldsymbol{f} + \boldsymbol{\epsilon}. \tag{5.7}$$

From (5.7) we note that the acquired data $\boldsymbol{y}$ that we get during an fMRI experiment is an iid Gaussian random vector. This implies that the maximum likelihood estimator (MLE) of $\boldsymbol{f}$ and $\boldsymbol{z}$ from the MR data is equivalent to a least squares optimization of the residual error of $\boldsymbol{y}$ and $\boldsymbol{A}(\boldsymbol{z})\, \boldsymbol{f}$. However, to make the estimator more robust to noise, especially in areas where $\boldsymbol{f}$ is small, this estimator will be regularized using a roughness penalty. The reconstruction is thus formed by minimizing a penalized log-likelihood cost function $\Psi_{\mathrm{PLL}}(\boldsymbol{f}, \boldsymbol{z})$ as follows:

$$\begin{aligned} \Psi_{\mathrm{PLL}}(\boldsymbol{f}, \boldsymbol{z}) &= \frac{1}{2}\left\|\boldsymbol{y} - \boldsymbol{A}(\boldsymbol{z})\, \boldsymbol{f}\right\|^2 + \mathrm{R}_1(\boldsymbol{f}) + \mathrm{R}_2(\boldsymbol{z}), \\ \left\{\hat{\boldsymbol{f}}, \hat{\boldsymbol{z}}\right\} &= \arg\min_{\boldsymbol{f}, \boldsymbol{z}} \Psi_{\mathrm{PLL}}(\boldsymbol{f}, \boldsymbol{z}), \end{aligned} \tag{5.8}$$

where the *data fit* term is the l_2 norm of the residual error of data and signal model and $\mathrm{R}_1(\boldsymbol{f})$ and $\mathrm{R}_2(\boldsymbol{z})$ are quadratic spatial *roughness penalties*, to be defined later.

A minimization problem that has two parameters that need to be estimated, as stated in (5.8), can be solved using an alternating minimization. There, the cost function is minimized relative to one parameter while the other is kept constant in an alternating fashion until the cost function has converged if both steps are monotonic. The cost function in (5.8) in terms of $\boldsymbol{f}$ while keeping $\boldsymbol{z}$ constant is quadratic, which can be solved fast and efficiently using the conjugate gradient (CG) algorithm. However, when it comes to $\boldsymbol{z}$, the problem becomes nonlinear and non-convex, which calls for potentially slower optimization algorithms.

In Chapter 3 we developed a method where estimating $\boldsymbol{z}$ was simplified and sped up considerably by a linear approximation using a reference rate map $\check{\boldsymbol{z}}$ [56]. Although this method does not guarantee monotonic descent for the cost function in (5.8), one could use the majorizer derived in Appendix B as a fallback option. Here we use a similar procedure to simplify the alternating minimization in (5.8) so that CG can be used to solve both minimization problems.

5.3 Fast and Motion Robust R_2^* Reconstruction

Suppose we acquire some additional time frames of k-space data at the beginning of an fMRI run. From this additional data we can estimate a complex valued reference rate map $\check{\boldsymbol{z}}$ using conventional methods of R_2^* and field map estimation. These can either be fast but less accurate conventional methods or slow and accurate iterative methods.

Assuming this reference rate map is a good approximation of the rate map that we want to estimate, we now form the following first-order approximation as was proposed in

Chapter 3:

$$\begin{aligned} \mathrm{e}^{-tz(\vec{r}_n)} &= \mathrm{e}^{-t\check{z}(\vec{r}_n)}\,\mathrm{e}^{-t(z_j(\vec{r}_n)-\check{z}(\vec{r}_n))} \\ &\approx \mathrm{e}^{-t\check{z}(\vec{r}_n)}\,(1+(-t)\,(z_j(\vec{r}_n)-\check{z}(\vec{r}_n))) \\ &= \mathrm{e}^{-t\check{z}(\vec{r}_n)} - (-t)\,\mathrm{e}^{-t\check{z}(\vec{r}_n)}\,\check{z}(\vec{r}_n) + (-t)\,\mathrm{e}^{-t\check{z}(\vec{r}_n)}\,z(\vec{r}_n)\,. \end{aligned} \tag{5.9}$$

This affine approximation allows the MR signal equation $s(t;\boldsymbol{z},\boldsymbol{f})$ to be approximated as follows:

$$\begin{aligned} s(t;\boldsymbol{z},\boldsymbol{f}) \approx \Phi(\vec{k}(t)) &\sum_{n=1}^{N} f(\vec{r}_n)\,\mathrm{e}^{-t\check{z}(\vec{r}_n)}\,\mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)} - \\ (-t)\,\Phi(\vec{k}(t)) &\sum_{n=1}^{N} f(\vec{r}_n)\,\mathrm{e}^{-t\check{z}(\vec{r}_n)}\,\mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)}\,\check{z}(\vec{r}_n) + \\ (-t)\,\Phi(\vec{k}(t)) &\sum_{n=1}^{N} f(\vec{r}_n)\,\mathrm{e}^{-t\check{z}(\vec{r}_n)}\,\mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)}\,z(\vec{r}_n)\,, \end{aligned}$$

which can be written in a more compact matrix-vector format:

$$\begin{aligned} \boldsymbol{s}(\boldsymbol{z},\boldsymbol{f}) &= \boldsymbol{A}(\boldsymbol{z})\,\boldsymbol{f} \approx \boldsymbol{s}(\boldsymbol{z},\boldsymbol{f};\check{\boldsymbol{z}}) \\ \boldsymbol{s}(\boldsymbol{z},\boldsymbol{f};\check{\boldsymbol{z}}) &\triangleq \boldsymbol{A}(\check{\boldsymbol{z}})\,\boldsymbol{f} - \boldsymbol{D}(-\boldsymbol{t})\,\boldsymbol{A}(\check{\boldsymbol{z}})\,\boldsymbol{D}(\boldsymbol{f})\,\check{\boldsymbol{z}} + \boldsymbol{D}(-\boldsymbol{t})\,\boldsymbol{A}(\check{\boldsymbol{z}})\,\boldsymbol{D}(\boldsymbol{f})\,\boldsymbol{z}, \end{aligned}$$

where $\boldsymbol{D}(\cdot)$ is a diagonal matrix where the argument is on the diagonal and the elements of $\boldsymbol{A}(\cdot)$ are given by (5.6). This allows us to approximate the MR signal so that both $\boldsymbol{f}$ and $\boldsymbol{z}$ have a linear relationship to the signal equation.

This approximation can now be used to form a new cost function, which is an approximation to the cost function in (5.8). This cost function is referred to here as an affine approximation to the penalized log likelihood, and $\hat{\boldsymbol{z}}$ and $\hat{\boldsymbol{f}}$ are found by minimizing it as follows:

$$
\begin{aligned}
\Psi_{\mathrm{APLL}}(\boldsymbol{f},\boldsymbol{z};\check{\boldsymbol{z}}) &= \frac{1}{2}\left\|\boldsymbol{y}-\boldsymbol{A}(\check{\boldsymbol{z}})\,\boldsymbol{f}+\boldsymbol{D}(-\boldsymbol{t})\,\boldsymbol{A}(\check{\boldsymbol{z}})\,\boldsymbol{D}(\boldsymbol{f})\,\check{\boldsymbol{z}}-\boldsymbol{D}(-\boldsymbol{t})\,\boldsymbol{A}(\check{\boldsymbol{z}})\,\boldsymbol{D}(\boldsymbol{f})\,\boldsymbol{z}\right\|^2 \\
&\quad + \mathrm{R}_1(\boldsymbol{f})+\mathrm{R}_2(\boldsymbol{z})\,, \\
\left\{\hat{\boldsymbol{f}},\hat{\boldsymbol{z}}\right\} &= \operatorname*{arg\,min}_{\boldsymbol{f},\boldsymbol{z}} \Psi_{\mathrm{APLL}}(\boldsymbol{f},\boldsymbol{z};\check{\boldsymbol{z}})
\end{aligned}
\tag{5.10}
$$

which can be minimized using an alternating minimization algorithm. Unlike in (5.8), both minimizations in the alternating minimizations are now quadratic so fast reconstruction algorithms such as CG can be used.

The accuracy of the reconstruction depends on the accuracy of the affine approximation in (5.9). Similar to what was proposed in Chapter 3, we *refine* $\boldsymbol{z}$ by updating $\check{\boldsymbol{z}}$ using the latest estimate of $\boldsymbol{z}$ in the alternating minimization algorithm. This will gradually improve the affine approximation and lead to a more accurate $\hat{\boldsymbol{f}}$ and $\hat{\boldsymbol{z}}$. This refinement process can be repeated until the cost function in (5.8) has converged.

5.3.1 Alternating Minimization of $\Psi_{\mathrm{APLL}}(\boldsymbol{f},\boldsymbol{z};\check{\boldsymbol{z}})$ for fMRI Data

The alternating minimization algorithm for a single time series of fMRI data consists of a one time initialization and three steps that are repeated until all time frames have been reconstructed. What follows is a general description of that process. Here we use j and l as the time frame and refinement indexes respectively and the estimate of $\boldsymbol{z}$ for time frame j and refinement l is then noted as $\hat{\boldsymbol{z}}_j^{(l)}$.

To initialize the algorithm, we need to estimate a $\boldsymbol{z}$ that can be used as a $\check{\boldsymbol{z}}$ for the first time frame of the fMRI data. For that purpose, additional feature-rich k-space data can be acquired at the start of each scan of an fMRI study. We then estimate an initial $\check{\boldsymbol{z}}$ either by using separate estimation of R_2^* [35–37, 50–52, 69] and field maps [7, 70] or slow nonlinear optimization algorithms [42, 43] of that k-space data. After reconstructing this image we repeat on three general steps to reconstruct all the time frames for the fMRI data. In the context of that general description, the initial $\check{\boldsymbol{z}}$ is referred to as $\hat{\boldsymbol{z}}_1^{(0)}$ and j and l are initialized as $j = l = 1$.

The first step is to minimize the cost function in (5.10) with respect to $\boldsymbol{f}$. This is done by setting $\boldsymbol{z} = \check{\boldsymbol{z}} = \hat{\boldsymbol{z}}_j^{(l-1)}$ and formulating a new *quadratic* cost function $\Psi_{\mathrm{APLL},1}\left(\boldsymbol{f}; \hat{\boldsymbol{z}}_j^{(l-1)}\right)$ as follows:

$$\begin{aligned} \Psi_{\mathrm{APLL},1}\left(\boldsymbol{f}; \hat{\boldsymbol{z}}_j^{(l-1)}\right) &\triangleq \Psi_{\mathrm{APLL}}\left(\boldsymbol{f}, \hat{\boldsymbol{z}}_j^{(l-1)}; \hat{\boldsymbol{z}}_j^{(l-1)}\right) - \mathrm{R}_2\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right) \\ &= \frac{1}{2}\left\|\boldsymbol{y} - \boldsymbol{A}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)\boldsymbol{f}\right\|^2 + \mathrm{R}_1(\boldsymbol{f}), \end{aligned} \tag{5.11}$$

where any constants have been subtracted from the original cost function in (5.10). Note that this cost function is the same if this problem were formulated using the original cost function in (5.8) instead of its affine approximation. The estimate $\hat{\boldsymbol{f}}_j$ is then equal to the minimum of this cost function, which can be found using fast NUFFT [20] or Toeplitz [19] based CG reconstructions.

The second step is to minimize the cost function in (5.10) with respect to $\boldsymbol{z}$. This is done by setting $\boldsymbol{f} = \hat{\boldsymbol{f}}_j$ and formulating a new *quadratic* cost function $\Psi_{\mathrm{APLL},2}\left(\boldsymbol{z}; \hat{\boldsymbol{f}}_j, \hat{\boldsymbol{z}}_j^{(l-1)}\right)$ as follows:

$$\begin{aligned} \Psi_{\mathrm{APLL},2}\left(\boldsymbol{z}; \hat{\boldsymbol{f}}_j, \hat{\boldsymbol{z}}_j^{(l-1)}\right) &\triangleq \Psi_{\mathrm{APLL}}\left(\hat{\boldsymbol{f}}_j, \boldsymbol{z}; \hat{\boldsymbol{z}}_j^{(l-1)}\right) \\ &= \frac{1}{2}\left\|\widetilde{\boldsymbol{y}}\left(\hat{\boldsymbol{f}}_j, \hat{\boldsymbol{z}}_j^{(l-1)}\right) - \boldsymbol{B}\left(\hat{\boldsymbol{f}}_j, \hat{\boldsymbol{z}}_j^{(l-1)}\right)\boldsymbol{z}\right\|^2 + \mathrm{R}_2(\boldsymbol{z}), \end{aligned} \tag{5.12}$$

where the data vector $\widetilde{\boldsymbol{y}}\left(\hat{\boldsymbol{f}}_j, \hat{\boldsymbol{z}}_j^{(l-1)}\right)$ and system matrix $\boldsymbol{B}\left(\hat{\boldsymbol{f}}_j, \hat{\boldsymbol{z}}_j^{(l-1)}\right)$ are defined as follows:

$$\begin{aligned} \widetilde{\boldsymbol{y}}\left(\hat{\boldsymbol{f}}_j, \hat{\boldsymbol{z}}_j^{(l-1)}\right) &\triangleq \boldsymbol{y} - \boldsymbol{A}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)\hat{\boldsymbol{f}}_j + \boldsymbol{D}(-\boldsymbol{t})\,\boldsymbol{A}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)\boldsymbol{D}\left(\hat{\boldsymbol{f}}_j\right)\hat{\boldsymbol{z}}_j^{(l-1)} \\ \boldsymbol{B}\left(\hat{\boldsymbol{f}}_j, \hat{\boldsymbol{z}}_j^{(l-1)}\right) &\triangleq \boldsymbol{D}(-\boldsymbol{t})\,\boldsymbol{A}\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)\boldsymbol{D}\left(\hat{\boldsymbol{f}}_j\right). \end{aligned}$$

The minimum of this cost function is referred to as $\hat{\boldsymbol{z}}_j^{(l)}$, which can be found using the same previously mentioned fast CG based reconstructions [19, 20]. One thing to note here is that in this cost function there is a multiplicative relationship between $\hat{\boldsymbol{f}}_j$ and the parameter of interest $\boldsymbol{z}$. This could potentially cause a coupling between the final estimates $\hat{\boldsymbol{f}}_j$ and $\hat{\boldsymbol{z}}_j$, which we need to be aware of.

The third step is to evaluate if the joint reconstruction for this time frame has converged and if so move on to the next time frame if one exists. If the cost function in (5.8) has not converged, we then redefine $l \triangleq l + 1$ and go back to the first step. However, if it has converged and j is not at the final time frame J we define:

$$\hat{\boldsymbol{z}}_{j+1}^{(0)} \triangleq \hat{\boldsymbol{z}}_j^{(l)} = \hat{\boldsymbol{z}}_j,$$

and move to the next time frame ($j \triangleq j + 1$), reset the refinement index ($l \triangleq 1$) and go back to the first step of the algorithm. If j is at the final time frame the algorithm is done.

Before using $\hat{\boldsymbol{z}}_j^{(l)}$ to estimate a new $\boldsymbol{f}_j$, we enforced an non-negativity constraint on the real part of $\hat{\boldsymbol{z}}_j^{(l)}$ to make certain that the R_2^* values used in the system matrix were positive. If this is not done there a possibility that the negative R_2^* values could cause instability issues in the reconstruction.

5.3.2 K-Space Trajectory Design

To improve the accuracy of the estimated relaxation rate map it has been shown to be beneficial to oversample near the center of k-space [43]. This is due to the MR signal being higher close to the center of k-space, so having a lot of samples in that area should help separate the effects of $\boldsymbol{f}$ and $\boldsymbol{z}$ in the signal equation.

A trajectory that revisits often the center of k-space as it samples the data would be ideal, such as radial or rosette [71] trajectories. The rosette trajectory was strongly advocated and used in a previous paper where both $\boldsymbol{f}$ and $\boldsymbol{z}$ were estimated using nonlinear optimization [43]. However, the rosette trajectory puts a very high strain on the gradient system and is susceptible to eddy currents and trajectory distortions. This can cause model mismatch in the reconstruction of data acquired from an MRI scanner and can lead to distortions in image space.

The spiral trajectory is much better behaved in terms of trajectory distortions than the rosette trajectory. Even though it does sample heavily the center of k-space, there is not enough information from one spiral readout to estimate both $\boldsymbol{f}$ and $\boldsymbol{z}$ without resorting to a non-quadratic regularization and thus increasing computation time significantly. Here we propose to acquire the data using two spiral readouts where the first one is acquired right after the RF excitation and the other is acquired with BOLD weighting. This results in a T_2-weighted and T_2^*-weighted readouts respectively which should result in a rich enough data set to estimate $\boldsymbol{f}$ and $\boldsymbol{z}$ accurately.

Even though two spiral readouts give an adequate amount of samples to estimate both maps, there is still a lot less samples from the center of k-space compared to if the data were collected using the rosette trajectory. This was demonstrated in [43], where four consecutive spiral readouts were shown to give worse results than a rosette trajectory of the same readout length. However, in that work the reconstruction did not use any regularization, which if used reduces the data dependence of the reconstruction. The regularization is generally based on a prior assumption on some properties the reconstructed images. Thus, if regularization is used it has to be taken into consideration when deciding on an acquisition method.

5.3.3 Regularization Design

The proposed reconstruction method uses regularization for both estimates of $\boldsymbol{f}$ and $\boldsymbol{z}$. This allows the reconstruction to make use of some prior information about the characteristics of either image and thus can work well with less data. This comes at a price of a user controlled bias in the reconstructed images.

For this reconstruction we chose to use a quadratic roughness penalty for both parameter maps, defined as:

$$\begin{aligned} \mathrm{R}_1(\boldsymbol{f}) &= \frac{1}{2}\beta_1 \left\|\boldsymbol{C}_1 \boldsymbol{f}\right\|^2 \\ \mathrm{R}_2(\boldsymbol{z}) &= \frac{1}{2}\left(\beta_{21} \left\|\boldsymbol{C}_1 \boldsymbol{z}_{\mathrm{R}}\right\|^2 + \beta_{22} \left\|\boldsymbol{C}_2 \boldsymbol{z}_{\mathrm{I}}\right\|^2\right), \end{aligned}$$

where $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ are first and second order differencing matrices and β_1, β_{21} and β_{22} are user selected tuning parameters that control the trade-off between noise and resolution. There are various ways of choosing these parameters and here we do that based on a predetermined desired resolution.

In Section 3.4 we showed how a quadratic spatial roughness penalty produced spatially varying local resolution for the resulting $\boldsymbol{z}$ estimates. In our case this is particularly problematic for $\boldsymbol{f}$ and the R_2^* map $\boldsymbol{z}_{\mathrm{R}}$ since both of these maps need to have uniform resolution for functional analysis purposes. However, for the field map $\boldsymbol{z}_{\mathrm{I}}$ non-uniform resolution is an advantage if the regularization smooths less where $\boldsymbol{f}$ is large and more where $\boldsymbol{f}$ is small [70].

In Section 3.4 we proposed an approximate correction to the resolution non-uniformity. This was done by adding a diagonal weighting matrix $\boldsymbol{D}$ with diagonal elements given by (3.22). It turns out that this analysis is still relevant to the proposed alternating minimization reconstruction.

In particular, the elements of a diagonal matrix $\boldsymbol{D}_1$ that approximately makes the minimum of (5.11) have spatially uniform resolution are given as follows:

$$\left[\boldsymbol{d}_1\left(\hat{\boldsymbol{z}}_j^{(l-1)}\right)\right]_n = \sqrt{\frac{\sum_{m=1}^M c_m^2 \, \mathrm{e}^{-2t_m \hat{z}_{\mathrm{R}}{}_j^{(l-1)}(\vec{r}_n)}}{\sum_{m=1}^M c_m^2 \, \mathrm{e}^{-2t_m \overline{z}_{\mathrm{R}}}}},$$

where $c_m \triangleq \Phi(\vec{k}(t_m))\, t_m$ and $\overline{z}_{\mathrm{R}}$ is the median R_2^* value within the brain. The elements of the diagonal matrix $\boldsymbol{D}_{21}$ that makes the estimated R_2^* from minimizing (5.12) have spatially uniform resolution are given as follows:

$$\left[\boldsymbol{d}_{21}\left(\hat{\boldsymbol{f}}_j, \hat{\boldsymbol{z}}_j^{(l-1)}\right)\right]_n = \left|\hat{f}_j(\vec{r}_n)\right| \sqrt{\frac{\sum_{m=1}^{M} c_m^2 \, \mathrm{e}^{-2t_m \hat{z}_{\mathrm{R}_j}^{(l-1)}(\vec{r}_n)}}{\sum_{m=1}^{M} c_m^2 \, \mathrm{e}^{-2t_m \overline{z}_{\mathrm{R}}}}}.$$

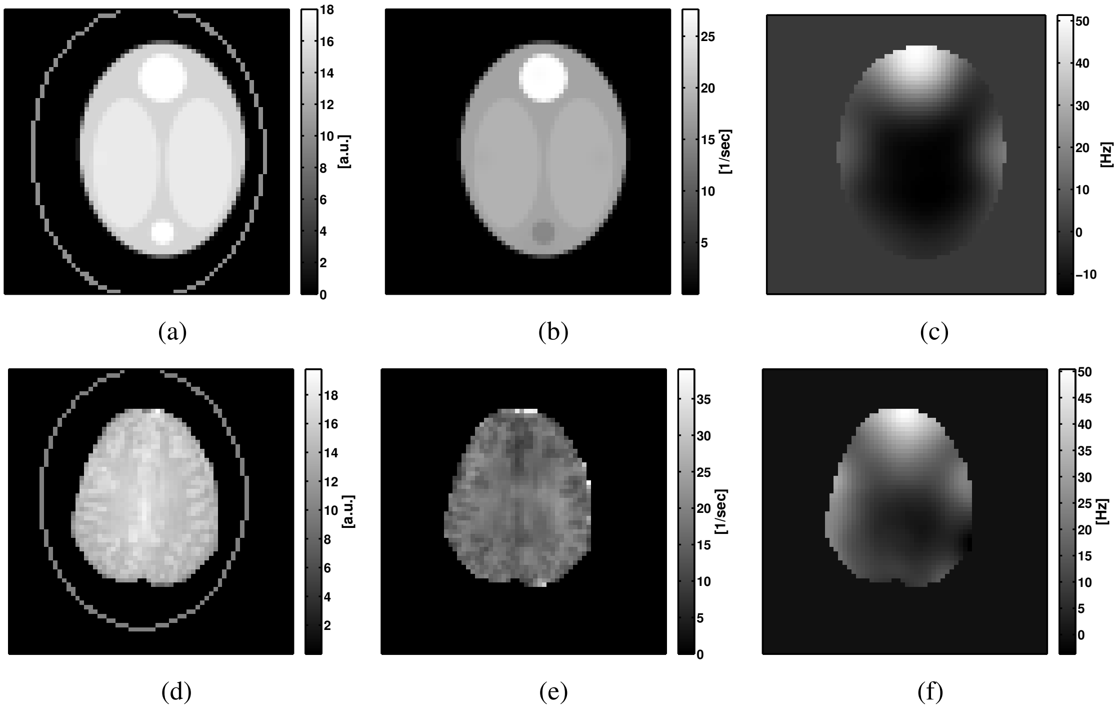
When it comes to the field map estimate we want to smooth more where $\boldsymbol{f}$ is small and smooth less where its large. Looking at the correction that is needed for resolution uniformity in R_2^* we note that this exact behavior would be seen if we would *not* use $\boldsymbol{D}_{21}$ to correct for the resolution nonuniformity. Thus, $\boldsymbol{D}_{22}$ to reconstruct the field map is a constant diagonal matrix defined as:

$$\left[\boldsymbol{d}_{22}\left(\hat{\boldsymbol{f}}_j, \hat{\boldsymbol{z}}_j^{(l-1)}\right)\right]_n = \mathrm{median}\left(\boldsymbol{d}_{21}\left(\hat{\boldsymbol{f}}_j, \hat{\boldsymbol{z}}_j^{(l-1)}\right)\right)$$

Note that all the diagonal weighting matrices $\boldsymbol{D}$ are dependent on the previous estimates of $\boldsymbol{z}$ or/and $\boldsymbol{f}$. However, if we were to update these matrices every time a new estimate of either map is found, the cost functions would be a moving target. Thus we implemented the reconstruction such that we only updated these matrices once the cost function has converged as per a predefined stopping rule. For the first time frame we update all the matrices three times since the initial $\check{\boldsymbol{z}}$ is usually not very close to $\boldsymbol{z}$. For all other time frames we update the matrices only once.

The stopping rule that we use here to decide when we have run enough refinements so that all weighting matrices can be updated is quite conservative. Here, we chose it such that if the normalized error of the current refinement of $\boldsymbol{f}$ relative to its previous refinement increased compared to the similarly defined normalized error of the previous refinement we deem the cost function as converged. This can be expressed with the following inequality:

$$\frac{\left\|\hat{\boldsymbol{f}}^{(l)} - \hat{\boldsymbol{f}}^{(l-1)}\right\|}{\left\|\hat{\boldsymbol{f}}^{(l)}\right\|} > \frac{\left\|\hat{\boldsymbol{f}}^{(l-1)} - \hat{\boldsymbol{f}}^{(l-2)}\right\|}{\left\|\hat{\boldsymbol{f}}^{(l-1)}\right\|},$$



(a) (b) (c)

(d) (e) (f)

Figure 5.1: Baseline simulation maps from digital phantoms ((a) and (b)) and real human images ((c)-(f)). (a) Digital $\boldsymbol{f}$ phantom with the edge of the reconstruction mask for reference; (b) Digital $\boldsymbol{z}_{\mathrm{R}0}$ phantom; (c) $\boldsymbol{z}_{\mathrm{I}0}$ from real data; (d) $\boldsymbol{f}$ with the edge of the reconstruction mask for reference; (e) $\boldsymbol{z}_{\mathrm{R}0}$; (f) $\boldsymbol{z}_{\mathrm{I}0}$.

where l is the refinement index. Then we either updated all $\boldsymbol{D}$ and continue estimating the images for the current time frame or move to the next one.

5.4 Simulations

We simulated a time series of k-space data to further analyze the proposed joint reconstruction of $\boldsymbol{f}_j$ and $\boldsymbol{z}_j$. The k-space data was generated using the exact signal equation as given in (5.4). Here, we chose to use a dual echo spiral out k-space trajectory, where the length of each readout is 4633 samples or 18.5ms long. The echo time for the second readout (TE_2) was chosen to be 30ms, unless stated otherwise, to give an adequate BOLD weighting.

All the simulation maps are 64×64 as is the size of the reconstructed images. Figure 5.1 shows the baseline simulation maps for $\boldsymbol{f}$, R_2^* and field maps for both a digital and collected human data based phantoms. Figure 5.2 shows the spatial maps and time series used to gen-

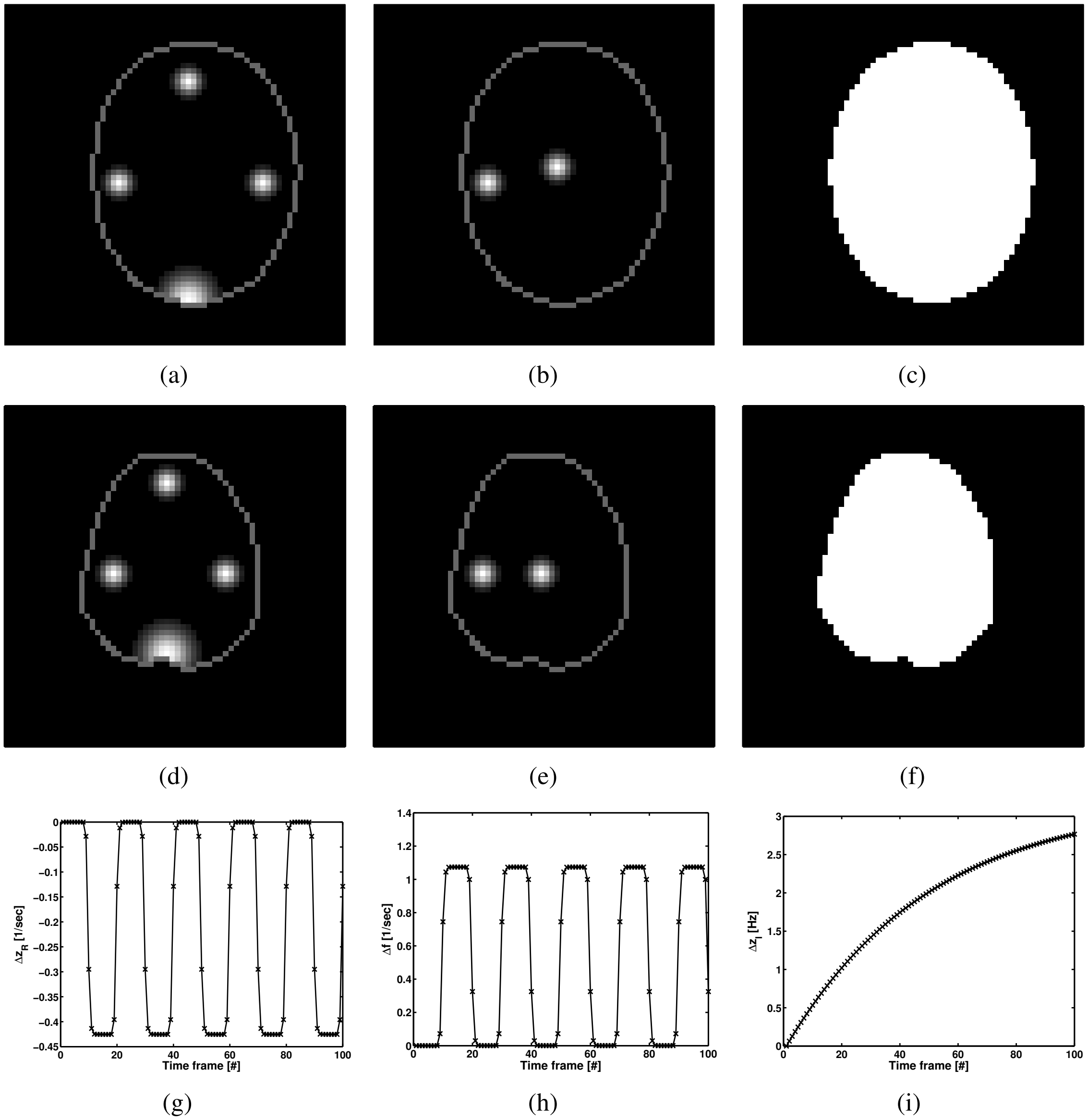


Figure 5.2: Simulation maps and graphs to generate the functional time series for both digital and collected data based phantoms. Spatial weights to generate functional changes in: (a) Digital $\boldsymbol{z}_{\mathrm{R}}$ phantom; (b) Digital $\boldsymbol{f}$ phantom; (c) $\boldsymbol{z}_{\mathrm{I}}$ associated with digital phantoms; (d) $\boldsymbol{z}_{\mathrm{R}}$ from collected data; (e) $\boldsymbol{f}$ from collected data; (f) $\boldsymbol{z}_{\mathrm{I}}$ based on collected data. Simulated functional temporal changes in (g) $\boldsymbol{z}_{\mathrm{R}}$ and (h) $\boldsymbol{f}$. Simulated hardware related drift in (i) $\boldsymbol{z}_{\mathrm{I}}$

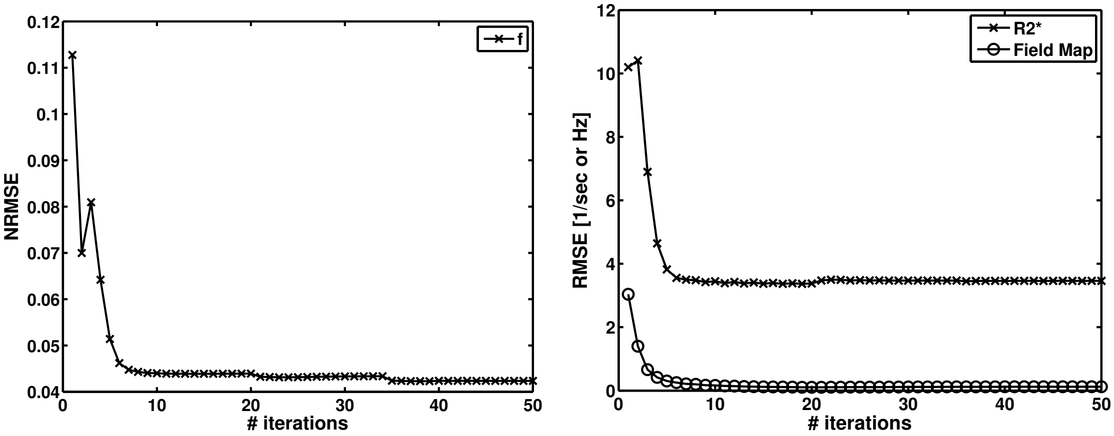
erate the fMRI simulations for both types of phantoms. Figure 5.2(a) and 5.2(d) shows the four “sites” of activation along with the edge of the phantom for reference, all simulated by changing R_2^* as given by the activation waveform in Figure 5.2(g). The top “site” was in a place of higher off-resonance than the others and the bottom “site” has active voxels close to the edges of the simulated phantom. The bilateral activations are to replicate bilateral motor activation. To make the simulations more realistic we also include a global drift in off-resonance, which is regularly seen in fMRI studies [45] and task correlated changes in $\boldsymbol{f}$ that simulate in-flow changes. Figure 5.2(b) and 5.2(e) shows the two “sites” where that occurs. One of the “sites” was intentionally put exactly in the same spatial position as the left bilateral activation.

5.4.1 Initialization: Robustness to Motion and Functional Changes

A possible issue with the reconstruction is how sensitive it is to the choice of the first $\check{\boldsymbol{z}}$ for some time frame. This is especially a concern if there has been motion, inflow changes or functional changes between the time the data for $\check{\boldsymbol{z}}$ was collected and the time the data that is being reconstructed with this $\check{\boldsymbol{z}}$. To investigate this, we simulated 3 time frames, with $\mathrm{TE}_1 = 4.59$ms and $\mathrm{TE}_2 = 30$ms. The first two time frames were used to estimate $\check{\boldsymbol{z}}$ by delaying the echo times by 2ms for the first time frame relative to the second one using conventional methods to estimate field map [7] and R_2^* [50] as discussed in Section 5.3.1.

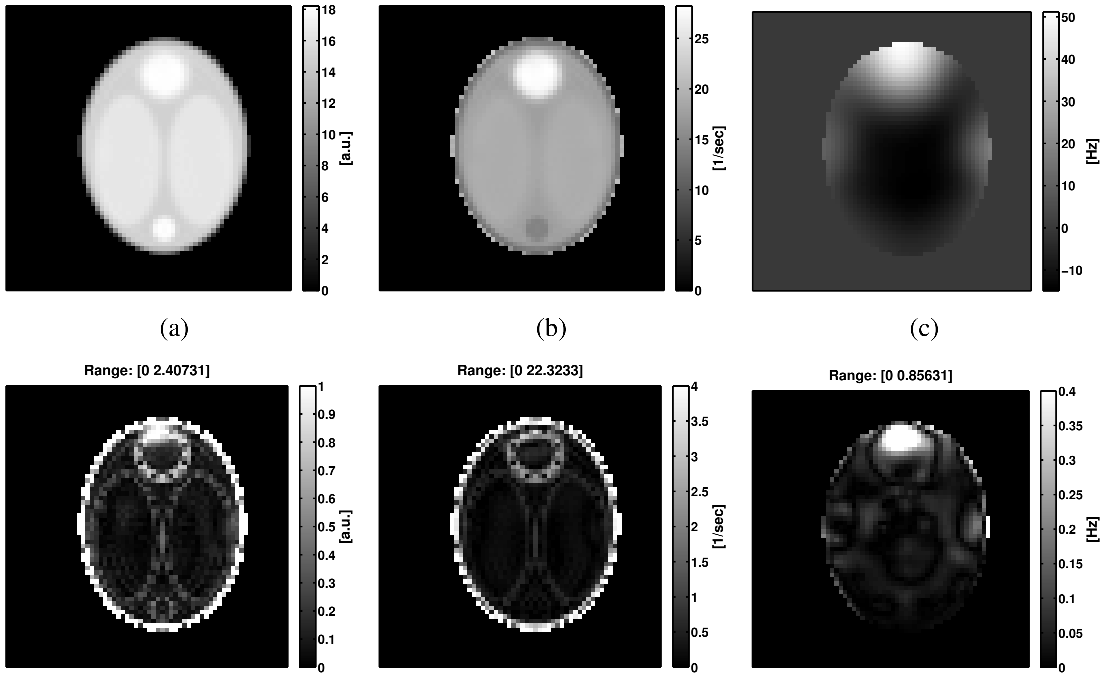
To simulate motion, the third time frame was moved in-plane by 3 voxels in x and 1 voxel in y directions. To simulate inflow and functional changes in the third time frame we also had an additive peak change of 1% in $\boldsymbol{f}$, 2.5% in $\boldsymbol{z}_{\mathrm{R}}$ and 3Hz in $\boldsymbol{z}_{\mathrm{I}}$ relative to their respective median values. The spatial location of these changes is shown in Figure 5.2(a) - 5.2(b). We performed 50 alternating minimizations of (5.11) and (5.12) and updated the penalty with new weights to maintain uniform resolution every time the reconstruction was deemed as having converged as described in Section 5.3.3.

Figure 5.3 show the NRMSE and RMSE plots for $\hat{\boldsymbol{f}}$, $\hat{\boldsymbol{z}}_{\mathrm{R}}$ and $\hat{\boldsymbol{z}}_{\mathrm{I}}$ for the third time



(a) (b)

Figure 5.3: Error plots over 50 alternating minimizations for $\hat{\boldsymbol{f}}$ and $\hat{\boldsymbol{z}}$. The error plots are all relative to the respective simulation maps. (a) Normalized root mean squared error (NRMSE) for $\hat{\boldsymbol{f}}$; (b) Root mean squared error (RMSE) for $\hat{\boldsymbol{z}}_{\mathrm{R}}$ and $\hat{\boldsymbol{z}}_{\mathrm{I}}$.



(a) (b) (c)

(d) (e) (f)

Figure 5.4: Reconstructed (a) $\hat{\boldsymbol{f}}$, (b) $\hat{\boldsymbol{z}}_{\mathrm{R}}$ and (c) $\hat{\boldsymbol{z}}_{\mathrm{I}}$ after 50 alternating minimizations and the absolute error of (d) $\hat{\boldsymbol{f}}$, (e) $\hat{\boldsymbol{z}}_{\mathrm{R}}$ and (f) $\hat{\boldsymbol{z}}_{\mathrm{I}}$.

frames. From the plots we see that the reconstruction has stabilized after 10 iterations. Also we note that the NRMSE goes further down for $\hat{\boldsymbol{f}}$ as we update the penalty with new weights at iteration 21 and 35.

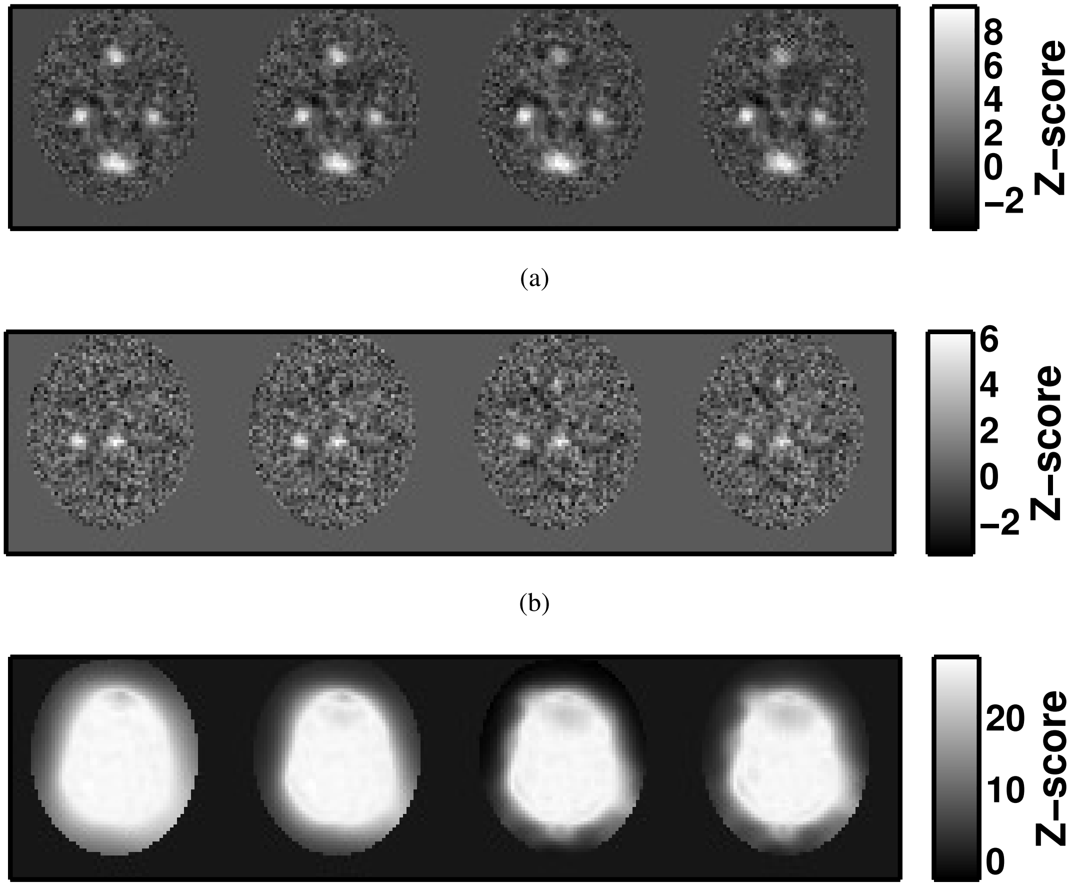
Figure 5.4 shows $\hat{\boldsymbol{f}}$ and $\hat{\boldsymbol{z}}$ after 50 alternating minimizations and the absolute error of that estimate relative to the respective simulation map. The absolute error maps indicate that even though the initial $\check{\boldsymbol{z}}$ was shifted by several voxels in both x- and y-direction, the reconstruction was able to a not too unreasonable estimate of all the parameters. The error maps also show how both $\hat{\boldsymbol{f}}$ and $\hat{\boldsymbol{z}}_{\mathrm{R}}$ are more sensitive to estimation error close to sharp edges. This is something that could possibly be improved by incorporating an edge aware roughness penalty.

5.4.2 Echo Time Dependence of the Dual Echo Spiral Out K-Space Trajectory

When using a dual echo spiral out where the second echo has a fixed echo time $\mathrm{TE}_2 = 30$ms we can still choose when the first echo is acquired. Since that choice could impact the performance of the reconstruction, we would like to look at the effects of possible choices for the echo time of the first echo. More specifically we will focus on how our choice affects the eventual fMRI analysis when using the estimated images from the joint reconstruction.

We simulated four variations of the fMRI time series as described above, where each one had a different echo time TE_1 for the first echo. Our choices for TE_1 were 4.59ms, 6.86ms, 9.14ms and 11.4ms, *i.e.*, from the earliest to the latest echo time possible. Additive iid Gaussian noise was included in all four simulation data sets. The amount of noise corruption was set such that the second k-space readout at $\mathrm{TE}_2 = 30$ms for the first time frame had SNR=55. From that we calculated the variance of the noise and used that to generate the noise for the simulated dual echo readout for all time frames.

To evaluate the effectiveness of each TE_1 choice we reconstructed the 100 time frame



(a)

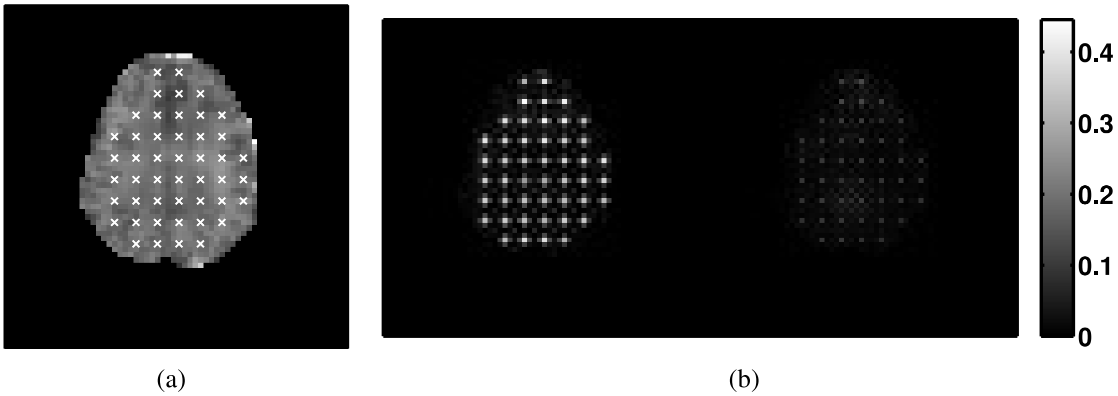
(b)

(c)

Figure 5.5: Results from GLM analysis of the simulated functional changes and drift from Figure 5.2(g)-5.2(i). Z-scores for reconstructed time series of (a) $\boldsymbol{z}_{\mathrm{R}}$, (b) $\boldsymbol{f}$ and (c) $\boldsymbol{z}_{\mathrm{I}}$ for the four choices of TE_1 (left to right) at 4.59ms, 6.86ms, 9.14ms and 11.4ms.

fMRI data using the proposed reconstruction, and performed a general linear model (GLM) analysis on the resulting time series of images. We analyzed $\boldsymbol{f}$, R_2^* and field map time series separately. For $\boldsymbol{f}$ and R_2^* time series we used the activation waveform as a regressor but for the field map we had the drift waveform as a regressor.

Figure 5.5 shows the resulting z-score maps for each image across all the echo time differences. We see that a longer time between the two echos results in less coupling in the activation maps of R_2^* and the inflow maps of $\boldsymbol{f}$. Also, there is a minor smoothing effect in the R_2^* maps and the field map drift effect is captured better as the echo times get closer. Both of these observations indicate that as the echos are further apart, the estimated $\boldsymbol{f}$ and



(a) (b)

Figure 5.6: The effect on the reconstructed $\boldsymbol{f}$ when adding an impulse in $\boldsymbol{z}_{\mathrm{R}}$. (a) Spatial positions of impulses shown with a white 'x' on top of the baseline $\boldsymbol{z}_{\mathrm{R}}$; (b) Resulting impulse responses in the reconstructed $\boldsymbol{z}_{\mathrm{R}}$ (left) and the effect on $\boldsymbol{f}$ (right).

$\boldsymbol{z}_{\mathrm{R}}$ maps are less coupled. This indicates that the accuracy of the three estimates improves as the echo times are further apart.

To further evaluate how much interaction there is between the $\boldsymbol{f}$ and $\boldsymbol{z}_{\mathrm{R}}$ estimates, we looked at what the effect of adding an impulse in $\boldsymbol{z}_{\mathrm{R}}$ had on the estimated $\boldsymbol{f}$. To do this, we generated simulation data that was 47 time frames. Each time frame had one impulse in $\boldsymbol{z}_{\mathrm{R}}$ and across all time frames the spatial position of that impulse was varied. We also generated another data set with 47 time frames with no impulses. These noiseless data sets were reconstructed using the joint reconstruction. To find the impulse response of the reconstruction the resulting $\boldsymbol{z}_{\mathrm{R}}$ and $\boldsymbol{f}$ images with no impulse were subtracted from the ones that had an impulse.

Figure 5.6(a) shows the impulse spatial positions and Figure 5.6(b) resulting impulse responses for both $\boldsymbol{z}_{\mathrm{R}}$ and $\boldsymbol{f}$. From this we see that there is some coupling between the two estimated images. The average ratio of the $\boldsymbol{z}_{\mathrm{R}}$ and $\boldsymbol{f}$ impulse responses for all the voxels where the impulse response peaks is about 3.79. This implies that for a maximal change in $\boldsymbol{z}_{\mathrm{R}}$ of 2.5% from the median value that is often seen in fMRI (0.4254 1/sec in this simulation), results in a possible 0.1123 change in the simulated $\boldsymbol{f}$, or about 0.75% change from its median value. However, as is supported by the simulation results in Figure 5.5, this percent change is low enough that it is generally below the noise level in fMRI data.

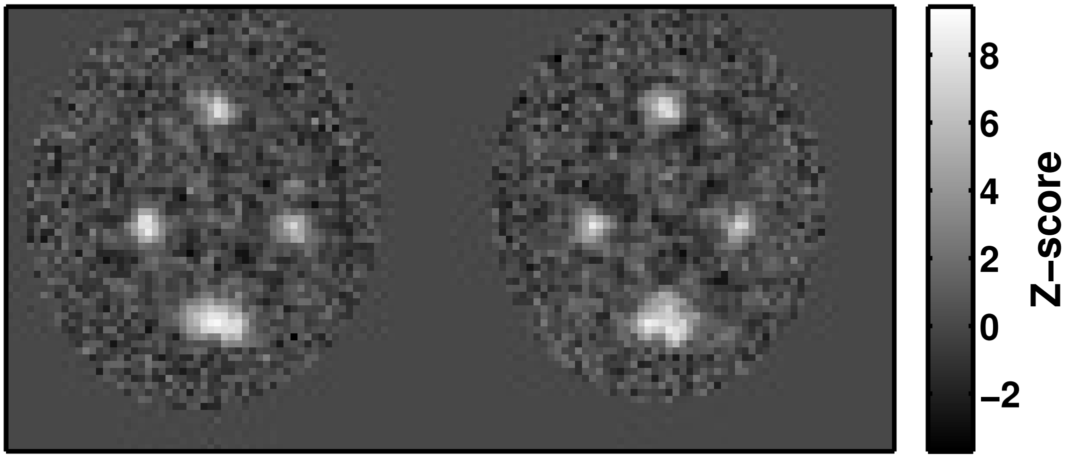


Figure 5.7: Results of the GLM analysis for the simulated functional data, with and without in-plane motion. The z-score maps of the simulated brain function contrast are shown $\boldsymbol{z}_{\mathrm{R}}$ (a) with in-plane motion and (b) without in-plane motion.

Even though we only analyzed how changes in $\boldsymbol{z}_{\mathrm{R}}$ can influence estimates of $\boldsymbol{f}$ a similar relationship exists with changes in $\boldsymbol{f}$ influencing the estimates of $\boldsymbol{z}_{\mathrm{R}}$.

5.4.3 Performance of Joint Reconstruction with Motion Corrupted fMRI Data

To evaluate the performance of the joint reconstruction when dealing with motion corrupted fMRI data, we simulated an fMRI experiment where the data was corrupted by in-plane motion. To get a perfect registration the in-plane motion was deliberately made to be an integer multiple of the voxel width. This was to avoid any performance issues of the registration algorithm. At time frames $[1, 26, 51, 76]$ we shifted the image by $[0, -1, -2, -3]$ voxels in x direction and $[0, -1, 0, 1]$ voxels in y direction relative to the initial position.

The time series was analyzed using GLM with the in-plane shift undone. The GLM was setup to account for any linear drift in the time series, by including a linear regressor. For comparison we also reconstructed the same functional simulation but now without any in-plane motion.

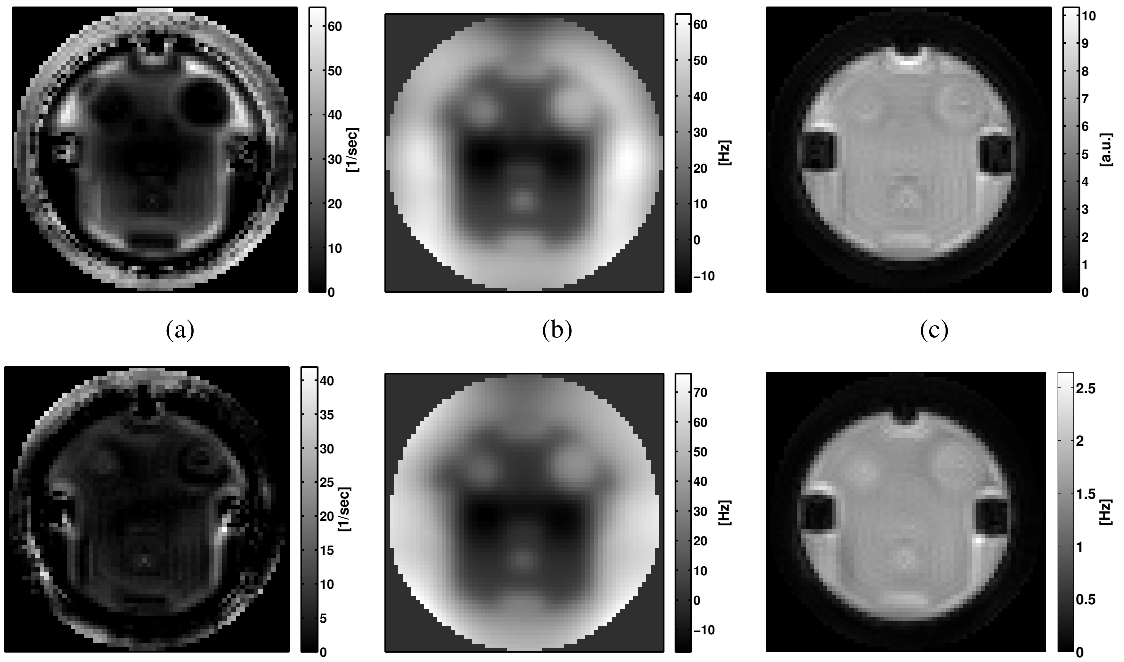
Figure 5.7 shows the results of the GLM analysis. It shows the z-score maps of the simulated brain function contrast for $\boldsymbol{z}_{\mathrm{R}}$ with and without the simulated in-plane motion. Overall the performance of the reconstruction seems to be similar when reconstructing data with and without motion corruption. This is further shown when we threshold the maps with an uncorrected p-value of 0.001. For that threshold we get 92 true positives (TP) and 2 false positives (FP) when reconstructing data with in-plane motion but 100 TP and 1 FP for data without in-plane motion. This indicates the joint reconstruction deals well with motion corrupted data.

5.5 Experimental Data

We acquired data from a 3T Signa GE scanner to evaluate the reconstruction for data from an MRI scanner. We collected images of both phantoms and humans to evaluate two things in particular. First, the phantom data was used to evaluate the temporal stability of the reconstructed R_2^* images, caused by magnetic field drifts frequently observed in fMRI studies and was discussed in Section 2. Secondly, the human data was used to evaluated the performance of the joint reconstruction for fMRI analysis. Here, we specifically focused on its performance under severe subject head movement during an fMRI study.

5.5.1 Temporal Stability under MRI Hardware Related Magnetic Field Drift

The GE resolution phantom was used to collect the images to evaluate how magnetic field drift affects BOLD weighted data both in conventional T_2^*-weighted images and R_2^* images from the joint reconstruction. To understand the impact of not accounting for field drift when reconstructing fMRI data, we looked at a slice that had considerable in-plane off-resonance. Also, to look at the role through-plane gradient has we acquired the slice with two different slice thicknesses. The first one was 4mm which is pretty standard in whole brain fMRI studies and the second one 1mm.



(a) (b) (c)

(d) (e) (f)

Figure 5.8: Estimated baseline maps from the joint reconstruction for the slice of interest and across both slice thicknesses. The reconstructed images for the 4mm slice thickness are shown for (a) $\boldsymbol{z}_{\mathrm{R}}$, (b) $\boldsymbol{z}_{\mathrm{I}}$ and (c) $\boldsymbol{f}$. The results for the 1mm slice are shown for (d) $\boldsymbol{z}_{\mathrm{R}}$, (e) $\boldsymbol{z}_{\mathrm{I}}$ and (f) $\boldsymbol{f}$.

When acquiring the dual echo spirals, the second echo time was set to 35ms but the first was set to $\mathrm{TE} = 6.85$ms and $\mathrm{TE} = 4.59$ms for the 1mm and 4mm slice thickness respectively. These were the earliest we could acquire the first echo. We acquired 100 time frames, 18 slices (although we only looked at one of them) with $\mathrm{TR} = 1.5$s and 90° flip angle. We then ran the joint reconstruction on both data sets and performed an iterative T_2^*-weighted image reconstruction [19] for each readout. Both reconstructions had a roughness penalty that was setup to give the same spatial resolution for the reconstructed R_2^* images and T_2^* images.

After reconstruction, we performed a GLM analysis on the time series of $\boldsymbol{f}$, R_2^* and field maps from the joint reconstruction and the T_2^*-weighted images from both echos. In all cases we estimated a linear drift effect along with a baseline using GLM and converted those results to z-scores maps. The baseline maps were used to visually evaluate the effects

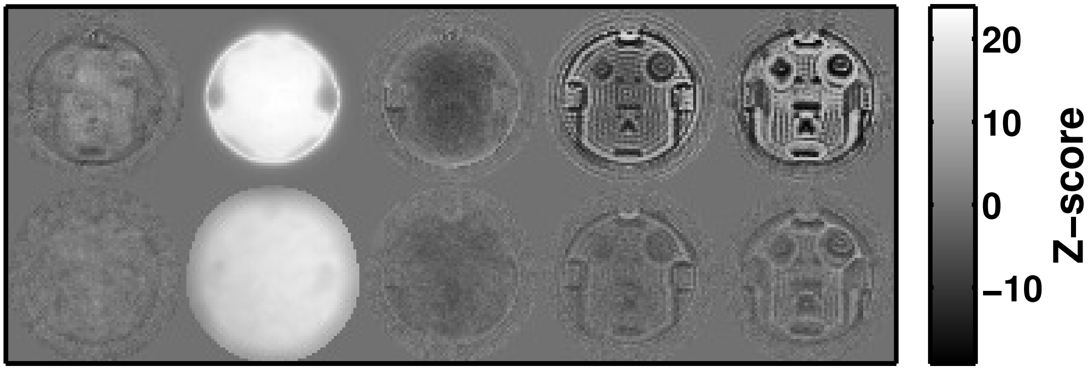
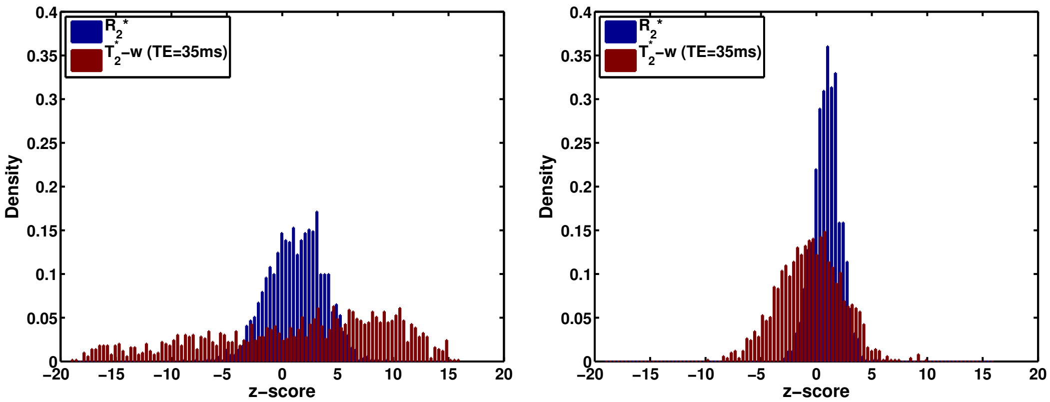


Figure 5.9: Linear drift effect z-scores from the GLM analysis of both slice thickness and for all the reconstructed images. (left to right) Results shown for the reconstructed images of $\boldsymbol{z}_{\mathrm{R}}$, $\boldsymbol{z}_{\mathrm{I}}$, $\boldsymbol{f}$, T_2^*-weighted image for the first echo and T_2^*-weighted images for the second image. The upper row shows results for 4mm slice and the lower row for 1mm slice.

of different slice thickness for the joint reconstruction. The z-score maps of the linear drift effect were used to evaluate the size of that effect and to form a density histogram.

Figure 5.8 shows the estimated baseline maps from the joint reconstruction for the slice of interest and across both slice thicknesses. We see that for the estimated baseline $\boldsymbol{z}_{\mathrm{R}}$ there are higher R_2^* values in the 4mm slice close the edges of the phantom compared to the 1mm slice thickness. This indicates that those areas are dominated by through-plane gradients, which are not accounted for in the reconstruction, and are explained by the reconstruction model as an increase in R_2^* values to account for the loss in signal in the second echo compared to the first one. This indicates that the reconstruction is sensitive to estimation errors in R_2^* for areas with a high through-plane gradient, such as close to the sinuses.

Figure 5.9 shows linear drift effect z-scores from the GLM analysis of both slice thickness and for all the reconstructed images. From these results we see that a linear drift effect is clearly visible for the T_2^*-weighted images for both echos. This effect is even visible for the first echo for the 1mm slice thickness. If we compare this to the effect size seen in $\boldsymbol{z}_{\mathrm{R}}$ estimates we see that R_2^* maps from the joint reconstruction are not as greatly affected by the field drift. The linear drift is correctly captured by the $\boldsymbol{z}_{\mathrm{I}}$ images, as clearly seen from Figure 5.9.



(a)

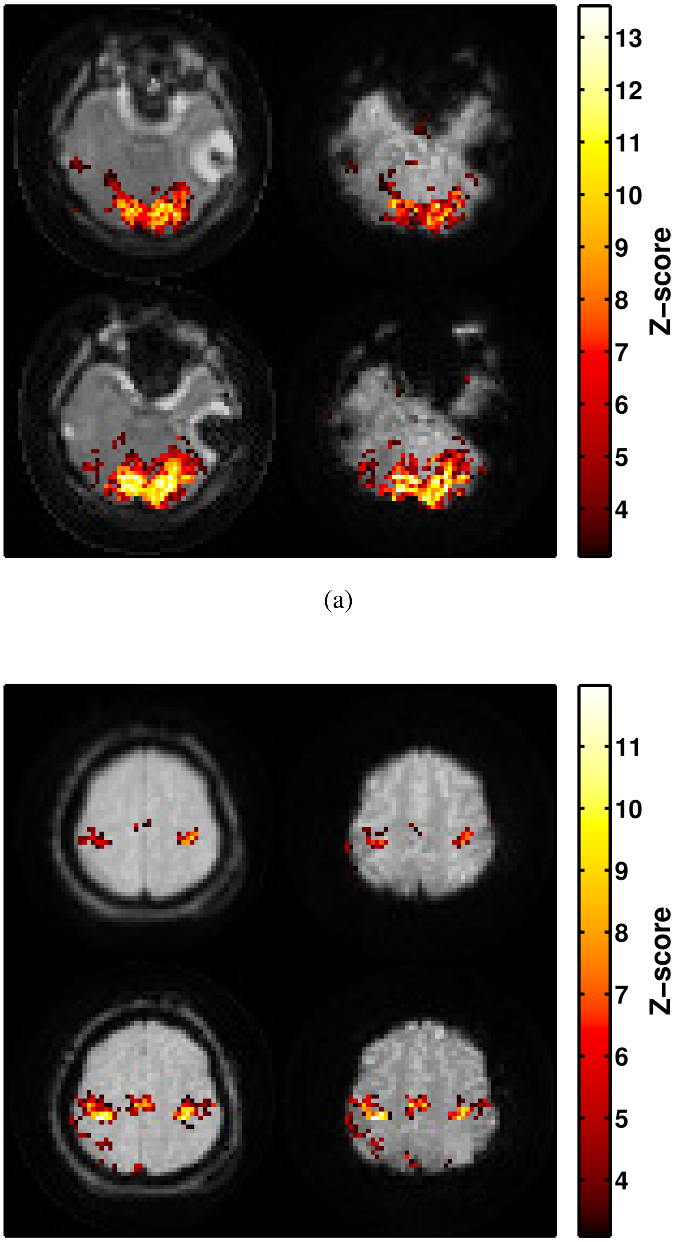
(b)

Figure 5.10: Density histograms for the linear drift z-scores of $\boldsymbol{z}_{\mathrm{R}}$ and the T_2^*-weighted image for the second echo ($\mathrm{TE} = 35\mathrm{ms}$) for (a) 4mm and (b) 1mm slice thickness. The histograms were produced only from voxels within the phantom.

To compare the potential effect the linear drift can have in fMRI studies, we also made a density histograms for the linear drift z-scores of $\boldsymbol{z}_{\mathrm{R}}$ and the T_2^*-weighted image for the second echo ($\mathrm{TE} = 35\mathrm{ms}$). Figure 5.10 shows these results for both slice thicknesses for voxels within the phantom. In both cases there is a clear indication that the R_2^* maps perform better in both cases. That being said, for both slice thicknesses there still seems to be lingering linear drift effect in the $\boldsymbol{z}_{\mathrm{R}}$ estimates, where the mean of the curves are 1.32 for 4mm and 1.00 for 1mm. This can also be seen in the z-score maps in Figure 5.9. This seems to be scanner related, since these drifts were not noticeable in the simulations where we simulated field map drift. Any drift in the scanner amplifiers could explain this effect, due to the previously shown coupling between $\boldsymbol{z}_{\mathrm{R}}$ and $\boldsymbol{f}$ estimates.

5.5.2 *In Vivo* fMRI Experiment with Intentional Head Motion

To evaluate the joint reconstruction under motion corrupted fMRI data, we performed an fMRI experiment where we instructed the subject to move their head randomly while performing a task. This experiment was repeated without motion for comparison. The selected task was a visually cued finger tapping task, where the subject was instructed to lie still for 20s and tap their fingers for 20s, repeated five times. Here the echo times were



(a)

(b)

Figure 5.11: Voxels above an uncorrected threshold with p-value 0.001 for two different slices and for data with and without intentional head movement. In (a) is a representative slice that shows visual activation and (b) motor activation. For both (a) and (b) the first and second columns show results for $\boldsymbol{z}_{\mathrm{R}j}$ and T_2^*-weighted images respectively and they are overlaid on top of the first time frame of $\boldsymbol{f}$ and the T_2^*-weighted image respectively. For (a) and (b) the first and second rows show results for intentional and non-intentional motion respectively.

$\mathrm{TE} = 4.59\mathrm{ms}$ and $\mathrm{TE} = 30\mathrm{ms}$ for the first and second echos respectively, 24 4mm slices and $\mathrm{TR} = 1.6\mathrm{s}$.

We reconstructed the images using the joint reconstruction and also used an iterative reconstruction [19] to estimate the T_2^*-weighted images from the second echo. We slice time and motion corrected both time series using tools from the FSL 4.0 software package.[2] The registration was performed directly to the T_2^*-weighted images but for the $\boldsymbol{z}_\mathrm{R}$ we first estimated the registration parameters from the estimated $\boldsymbol{f}$ images and applied those registration parameters to the $\boldsymbol{z}_\mathrm{R}$. For activation detection we used GLM analysis for both images, where we controlled for linear drift and timing offsets using a linear regressor and the activation regressor derivative respectively.

Figure 5.11 shows the voxels above an uncorrected p-value of 0.001 ($z_{thr} = 3.07$) for two slices from both reconstructions of the fMRI data with intentional and non-intentional head motion. The anatomical overlay for the R_2^* and T_2^*-weighted based results used the first frame of $\boldsymbol{f}$ and T_2^*-weighted magnitude image respectively. For both the intentional motion and non-intentional motion and across slices, $\boldsymbol{z}_{\mathrm{R}j}$ has more voxels above the threshold than the conventional T_2^*-weighted images. In terms of robustness to motion corruption $\boldsymbol{z}_\mathrm{R}$ gives slightly better results for both slices, but not categorically so. This would have to be investigated further.

5.6 Conclusion and Discussion

We proposed a joint reconstruction method that estimates the initial magnetization image $\boldsymbol{f}$, R_2^* and field map for a time series of MR data such as used in functional MRI. The reconstructed images have been shown to be robust to motion and blood in-flow, *i.e.*, temporal changes in $\boldsymbol{f}$, both in simulations and real data. Also, the model was able to capture magnetic field drift as a true drift in the field map. This was shown in phantom data that

---

[2] Software available at http://www.fmrib.ox.ac.uk/fsl/

was collected from an MRI scanner and, although not shown, the same can be said of the simulations that included field map drift. In terms of activation detection, the reconstructed R_2^* maps produced improved results when compared to T_2^*-weighted images, both in simulations and for real data.

We found that the method performed well by using a dual echo spiral-out readout. One thing that we did notice was that the further apart the echo times, the smaller of a coupling effect we got in the reconstructed $\boldsymbol{z}_{\mathrm{R}}$ and $\boldsymbol{f}_j$ estimates. In our discussion we put a fixed echo time for the second echo at 30ms. If we would move that further back, that could potentially reduce this coupling even further. However, even though that would work well in the simulations presented here, any real data would have to take into consideration the additional signal loss from through-plane gradients. Ultimately this is a compromise that needs further investigation.

Through-plane gradients were not specifically included in the signal model used for the joint reconstruction. However, to get an accurate R_2^* estimate, this is something that needs to be added to the model. It would make the reconstruction more robust in areas of higher susceptibility differences such as close to the sinuses or ear canals. The question of how we would first estimate a gradient map and also how we would update that gradient map due to head motion while acquiring the data needs further investigation. An idea of how that could be done is presented in future work section.

The apparent coupling between the reconstructed $\boldsymbol{f}_j$ and $\boldsymbol{z}_{\mathrm{R}}$ is a concern for the joint reconstruction. Even though we showed in simulations that changes in R_2^* cause changes in $\boldsymbol{f}$ the same could be said of changes in $\boldsymbol{f}$ affecting R_2^* estimates. This degrades the quantitative and qualitative results we get for the R_2^* estimates using the joint reconstruction. There are two possible things we can look at to potentially address this coupling effect. The first is a redesign of the penalty that we use for $\boldsymbol{f}$. The second is to consider different k-space trajectory designs.

One problem of using a quadratic roughness penalty for both $\boldsymbol{z}_{\mathrm{R}}$ and $\boldsymbol{f}$ with the same effective resolution is that any separation of the two parameter maps has to be done by the data fit in the cost function. However, as we see from (5.12) and the definition of $\boldsymbol{B}\left(\hat{\boldsymbol{f}}_j, \hat{\boldsymbol{z}}_j^{(l-1)}\right)$, there is a multiplicative relationship between $\boldsymbol{z}$ and $\hat{\boldsymbol{f}}$, which we cannot avoid with the proposed joint reconstruction. This implies that rethinking the penalty term for $\boldsymbol{f}$ is needed to decrease this coupling further. One such idea would be to use a sparsity penalty on the dynamic changes in $\boldsymbol{f}$ relative to some baseline $\boldsymbol{f}$ map. This would however make the the cost function in (5.11) non-quadratic and could potentially have an impact on the reconstruction speed. Also, for the areas where there are changes in $\boldsymbol{f}$ it is not guaranteed that there would be a reduction in this coupling effect.

In [43] it was suggested that a rosette trajectory would be a more effective trajectory than multi-echo spirals for a non-penalized estimate of all three parameters in the joint reconstruction. Since we now have a penalty to impose restrictions on the solution of the reconstruction algorithm, this conclusion may not apply any longer. However, it is far from being clear that the proposed two echo spiral is the best trajectory to use to estimate the parameters. We noted in simulations that by a simple shift in the echo time of the first echo, we were able to reduce this coupling in fMRI results. This indicates that there could be different trajectories that reduce it further. For example a four echo spiral where each spiral is twofold undersampled, so that all four echos combined have the same number of samples as the two fully sampled echos used here, could be a better option. This is because such a sampling pattern would allow us to sample the center of k-space more often than two echos, but not put too much strain on the gradient hardware to induce eddy currents. This trajectory and others for use with the joint reconstruction would need further investigation.

The performance of the joint reconstruction for fMRI data improves when compared to T_2^*-weighted images. However, further investigation is needed to assess how reliable and consistent the estimated R_2^* images are. This is with a specific focus on the functional detection, *i.e.*, how reliable do areas of activation show up in human scanning when we use the estimated R_2^* images. Methods such as an estimate of test-retest reliability of human fMRI data proposed in [72] and applied in [62] would be a good tool to use for further assessment. This would require repeated scans from several subjects.

Chapter 6

Conclusion and Future Work

6.1 Dynamic Reconstruction of R_2^* Images for fMRI

The initial motivation for the work presented in this dissertation was to develop a fast model based iterative image reconstruction of R_2^* and field map to be used in fMRI. However, both of these parameters have a nonlinear relationship to the MR signal equation that would require slow optimization algorithms to estimate. It turns out that the temporal changes in R_2^* and field map as we acquire each time frame in fMRI is relatively slow. This allowed us to make a linear approximation to the signal equation around a reference R_2^* and field map thus significantly reducing the compute time of the estimate as was shown in Chapter 3. This allowed us to reconstruct an R_2^* map from one fully sampled k-space trajectory, thus maintaining the same temporal and spatial resolution of conventional fMRI acquisition methods. This was at the minor cost of having to acquire some small amounts of additional data at the start of each fMRI experiment to estimate an initial reference R_2^* and field map.

Any data collected from the MRI scanner is going to be contaminated with noise. To make the reconstruction robust to noise we added a quadratic roughness penalty to the cost function. Here we chose to separately penalize R_2^* and field maps. This was because the field map is generally more spatially smooth compared to R_2^*. The R_2^* maps is more an inherent property of the tissue in each voxel of the image and thus follows brain structure but the field map is associated with the magnetic field which is more smooth than the underlying tissue structure. By stacking the vectors in the cost function, we were able to reformulate the problem as a quadratic cost function while maintaining the separate regularization of both maps. By including this regularization we were able to reduce the data dependence of the reconstruction, which allowed us to use more commonly used trajectories such as spirals to estimate R_2^*.

One of the side effects of using a regularization function for this problem, was that it introduced spatially nonuniform resolution for the R_2^* maps. Since these estimates are intended for fMRI analysis it was preferred that the spatial resolution of the R_2^* estimates be nearly uniform. To enforce this property we developed a nonuniform roughness penalty for R_2^* that accounted for this spatially varying resolution. As was shown in Chapter 3 this was quite effective. To set the regularization parameter to the right value to achieve some desired resolution, we also developed a fast approximate method of evaluating the local impulse response of the reconstruction algorithm. Chapter 4 concluded that this approximation was accurate and fast. This allowed us to quickly evaluate the resolution properties of the spatially nonuniform roughness penalty so that we could tune the regularization in a computationally efficient manner.

Finally, the last part of the work is presented in Chapter 5. There we improved the reconstruction in Chapter 3 by also estimating the magnetization of the object. This was feasible by acquiring some data right after RF excitation in addition to the conventional BOLD weighted readout. We were able to show that the reconstruction was robust to motion and any temporal changes in the magnetization such as inflow of fresh blood that is generally a confound in conventional fMRI images. This potentially opens the door to also analyzing the magnetization for temporal structures associated with the functional changes. This made the reconstruction produce even more viable and robust estimates of R_2^*, thus paving the way for a fast method to estimate quantitative measure of functional activation.

6.2 Future Work

The future work section can be split into two categories. The first category are ideas that can further improve the reconstruction algorithm to produce more accurate estimates of R_2^*. The second category are ideas that relate to specifically evaluating the possible improvements of using R_2^* maps for fMRI analysis both in terms of detection and quantification. Below we highlight some of the major ideas but do note this is not a complete list. There are other smaller ideas that are presented after each chapter.

6.2.1 Future Improvements to the Joint Reconstruction Algorithm

Incorporating through-plane gradient in the signal model for the joint reconstruction

One of the things that is apparent is that the reconstruction does need to be improved in areas of high susceptibility. Those areas are generally dominated by high through-plane gradients that cause faster signal decay. Since the model used in this reconstruction does not account for the source of this additional decay, the R_2^* values can be higher than they actually are in these areas. There has already been prior work that first derives a 2D MR signal model that includes effects of the slice profile and through-plane gradient and then proposes an efficient method to include these effects in the MR signal model [47]. That model could be used in the joint reconstruction proposed in this dissertation and possibly maintain a similar computational speed as the current version of the reconstruction. Since there is a possibility that R_2^* is correlated with through-plane, we also have to assess the impact on the R_2^* estimates when adding through-plane to the signal model.

Updating through-plane gradient maps for fMRI time series

Connected with the previous suggested future work is the fact that if through-plane gradients are used in the reconstruction for fMRI data, motion will affect the through-plane map at each time frame. If we were to also jointly estimate through-plane gradients we would require more data which may reduce the temporal resolution of the fMRI time series. However, it has been suggested that the field map has a strong correlational relationship to the through-plane gradients [73]. Such a relationship could be modeled and estimated from some additional data at the start of the fMRI study. Since we are estimating field maps for every time frame we could use this relationship to update the through-plane gradient map for every time frame.

K-space trajectory design for the joint reconstruction

Even though double echo fully sampled spiral-out was used to acquire the data in the joint reconstruction there is no indication that it is in some sense the best choice. For example, a four echo undersampled spiral-out would distribute measurements close to the center of k-space more evenly over the total readout than the double echo fully sampled spiral would. However, there is no obvious way of seeing if that gives a more accurate estimate of R_2^*. Other choices need to be explored and compared in terms of the quality of the resulting R_2^* estimates.

Further enhancement of a spatially uniform roughness penalty

The roughness penalty that was used for estimating the rate map $\boldsymbol{z}$ tried to keep the spatial resolution uniform across the object. However, the design that we developed is an approximation. Also, it does not account for any influence of spatial variability due to the field map, only the R_2^* map is accounted for. Some empirical simulation did indicate that the field map plays some role in causing spatial resolution nonuniformity. This needs to be investigated further and addressed if possible.

6.2.2 Future Applications for R_2^* Maps in fMRI

Test-retest reliability of R_2^* in humans

Simulating fMRI data can only go so far in assessing the performance of a reconstruction for fMRI data. The real test is to see if there is tangible evidence that this is also the case when scanning humans. This needs to be investigated further than was done in this dissertation. More subjects and repeated scans for each subject is needed to formulate a test-retest reliability for the reconstruction. We can form ROC curves using that analysis which allows us to quantify any potential improvements in detection performance when using R_2^* maps.

Longitudinal stability of R_2^* as a measure of activation

Many fMRI applications call for repeated scans of subjects to monitor long-term changes in the functional responses to certain tasks. To be able to compare all the results it is important to separate changes due to physiology (this is of interest) and hardware stability (this is a nuisance effect). Since R_2^* can be directly associated to actual physiological changes related to functional activity, it would make for an ideal candidate for these longitudinal studies. However, we still need to assess the long term temporal stability of R_2^* before it can be used regularly in such studies.

Multi-modal imaging using R_2^*

The future of functional imaging undoubtedly lies in trying to merge the information from the multiple sources that are available today to measure functional activity. To do this it helps to have quantifiable measurements when merging this information across different modalities, *e.g.*, EEG and fMRI or near infrared spectroscopy (NIRS) and fMRI. One thing that is needed is to assess how viable R_2^*, as a measure of physiological changes associated with brain function, can help in this regard. This needs to be looked into further.

Appendices

Appendix A

Fast Gradient Based Optimization of a Quadratic Stacked Cost Function

A.1 Introduction

Suppose the acquired complex-valued MR data vector $\boldsymbol{y}$ can be modeled as follows:

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\epsilon},$$

where $\boldsymbol{A}$ is the system matrix, $\boldsymbol{x}$ is the parameter of interest and $\boldsymbol{\epsilon}$ is iid Gaussian noise. A common method of estimating $\boldsymbol{x}$ is to minimize a quadratic penalized negative log-likelihood cost function, that is given as follows:

$$\Psi(\boldsymbol{x}) = \frac{1}{2} \left\| \boldsymbol{y} - \boldsymbol{A}\boldsymbol{x} \right\|^2 + \mathrm{R}(\boldsymbol{x}), \tag{A.1}$$

where $\mathrm{R}(\boldsymbol{x})$ is usually a quadratic spatial roughness penalty. If the roughness penalty that is needed is supposed to separately penalize the real part $\boldsymbol{x}_\mathrm{R}$ and imaginary part $\boldsymbol{x}_\mathrm{I}$ of $\boldsymbol{x}$ as follows:

$$\mathrm{R}(\boldsymbol{x}) = \frac{1}{2} \left(\beta_1 \left\| \boldsymbol{C}_1 \boldsymbol{x}_\mathrm{R} \right\|^2 + \beta_2 \left\| \boldsymbol{C}_2 \boldsymbol{x}_\mathrm{I} \right\|^2 \right),$$

the cost function can be formulated in an equivalent quadratic stacked format as follows:

$$\Psi_{\mathrm{S}}(\boldsymbol{x}_{\mathrm{S}}) = \frac{1}{2} \left\| \boldsymbol{y}_{\mathrm{S}} - \boldsymbol{A}_{\mathrm{S}} \boldsymbol{x}_{\mathrm{S}} \right\|^2 + \mathrm{R}_{\mathrm{S}}(\boldsymbol{x}_{\mathrm{S}}) \,, \tag{A.2}$$

where the definitions of the stacked vectors and matrices is given in Section 4.2.1. Estimating $\boldsymbol{x}$ is now performed by minimizing $\Psi_{\mathrm{S}}(\boldsymbol{x}_{\mathrm{S}})$ in terms of the real valued stacked vector $\boldsymbol{x}_{\mathrm{S}}$. The goal is then to perform this using an iterative optimization algorithm in a memory efficient and computationally fast manner. Our choice was to use the conjugate gradient (CG) optimization algorithm.

Using CG to minimize the quadratic cost functions in (A.1) and (A.2) involves finding the negative gradient of the cost function. In both cases the cost function gradient is a sum of the data fit and the penalty gradients. Calculating the penalty gradient is fast for both cost functions but the same cannot be said of the data fit gradient.

For (A.1) it involves the forward- and backprojections of the system matrix $\boldsymbol{A}$, *e.g.*, $\boldsymbol{A}\boldsymbol{x}$ and $\boldsymbol{A}'\boldsymbol{y}$ respectively. There have been very efficient ways developed to calculate the forward- and backprojections [19, 20] using FFT or NUFFT [25] and $\boldsymbol{A}$ can be stored using only the components that form the matrix. However, for (A.2) the $\boldsymbol{A}_{\mathrm{S}}$ is twice the size of $\boldsymbol{A}$ and also involves the forward- and backprojection of the real and imaginary parts of $\boldsymbol{A}$ which are not easily accelerated using similar methods as for $\boldsymbol{A}$. We propose to accelerate the data fit gradient calculation of (A.2) using the already accelerated data fit gradient calculation of (A.1), by linking the result of the latter with the former.

A.2 Fast Gradient Calculation of a Stacked Cost Function

We will start by analyzing how the real and imaginary parts of the negative gradient of the data fit term of (A.1) looks like. It is given as a sum of two terms, as follows:

$$-\nabla_{\boldsymbol{x}} \frac{1}{2} \left\| \boldsymbol{y} - \boldsymbol{A}\boldsymbol{x} \right\|^2 = \boldsymbol{A}'\boldsymbol{y} - \boldsymbol{A}'\boldsymbol{A}\boldsymbol{x}.$$

Those two terms can now be written in their real and imaginary parts. The first term comes as follows:

$$\begin{aligned}\boldsymbol{A}'\boldsymbol{y} &= \left(\boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}} - i\boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\right)\left(\boldsymbol{y}_{\mathrm{R}} + i\boldsymbol{y}_{\mathrm{I}}\right) \\ &= \left(\boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{y}_{\mathrm{R}} + \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{y}_{\mathrm{I}}\right) + i\left(\boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{y}_{\mathrm{I}} - \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{y}_{\mathrm{R}}\right), \end{aligned} \tag{A.3}$$

and the second term as:

$$\begin{aligned}\boldsymbol{A}'\boldsymbol{A}\boldsymbol{x} &= \left(\boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}} - i\boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\right)\left(\boldsymbol{A}_{\mathrm{R}} + i\boldsymbol{A}_{\mathrm{I}}\right)\left(\boldsymbol{x}_{\mathrm{R}} + i\boldsymbol{x}_{\mathrm{I}}\right) \\ &= \left(\boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}} - i\boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\right)\left[\left(\boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{R}} - \boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{I}}\right) + i\left(\boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{I}} + \boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{R}}\right)\right] \\ &= \left[\boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{R}} - \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{I}} + \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{I}} + \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{R}}\right] + \\ &\quad i\left[\boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{I}} + \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{R}} - \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{R}} + \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{I}}\right]. \end{aligned} \tag{A.4}$$

The negative gradient of the data fit term of (A.2) is also given with a sum of two terms, as follows:

$$-\nabla_{\boldsymbol{x}_{\mathrm{S}}}\frac{1}{2}\left\|\boldsymbol{y}_{\mathrm{S}} - \boldsymbol{A}_{\mathrm{S}}\boldsymbol{x}_{\mathrm{S}}\right\|^2 = \boldsymbol{A}_{\mathrm{S}}'\boldsymbol{y}_{\mathrm{S}} - \boldsymbol{A}_{\mathrm{S}}'\boldsymbol{A}_{\mathrm{S}}\boldsymbol{x}_{\mathrm{S}}.$$

The first term can be written as follows:

$$\begin{aligned}\boldsymbol{A}_{\mathrm{S}}'\boldsymbol{y}_{\mathrm{S}} &= \begin{bmatrix} \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}} & \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}} \\ -\boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}} & \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}} \end{bmatrix}\begin{bmatrix} \boldsymbol{y}_{\mathrm{R}} \\ \boldsymbol{y}_{\mathrm{I}} \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{y}_{\mathrm{R}} + \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{y}_{\mathrm{I}} \\ \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{y}_{\mathrm{I}} - \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{y}_{\mathrm{R}} \end{bmatrix}, \end{aligned} \tag{A.5}$$

and the second term as follows:

$$\begin{aligned}
\boldsymbol{A}'\boldsymbol{A}\boldsymbol{x} &= \begin{bmatrix} \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}} & \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}} \\ -\boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}} & \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_{\mathrm{R}} & -\boldsymbol{A}_{\mathrm{I}} \\ \boldsymbol{A}_{\mathrm{I}} & \boldsymbol{A}_{\mathrm{R}} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_{\mathrm{R}} \\ \boldsymbol{x}_{\mathrm{I}} \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}} & \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}} \\ -\boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}} & \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{R}} - \boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{I}} \\ \boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{I}} + \boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{R}} \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{R}} - \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{I}} + \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{I}} + \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{R}} \\ \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{I}} + \boldsymbol{A}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{R}} - \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{R}} + \boldsymbol{A}_{\mathrm{I}}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{I}}\boldsymbol{x}_{\mathrm{I}} \end{bmatrix}.
\end{aligned} \tag{A.6}$$

Comparing (A.3) to (A.5) and (A.4) to (A.6), we can state the following:

$$-\nabla_{\boldsymbol{x}_{\mathrm{S}}} \frac{1}{2} \left\| \boldsymbol{y}_{\mathrm{S}} - \boldsymbol{A}_{\mathrm{S}}\boldsymbol{x}_{\mathrm{S}} \right\|^2 = \begin{bmatrix} \Re\left(-\nabla_{\boldsymbol{x}} \frac{1}{2} \left\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\right\|^2\right) \\ \Im\left(-\nabla_{\boldsymbol{x}} \frac{1}{2} \left\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\right\|^2\right) \end{bmatrix}.$$

This allows us to calculate the negative gradient of the stacked data fit term using the real and imaginary parts of the negative gradient of the complex valued data fit term. This allows us to use any previously developed acceleration methods.

Appendix B

Majorizing A Nonlinear Least Squares Problem to Estimate R_2^* and Field Map

Estimating a rate map $\boldsymbol{z}$ using MR signal equation for a fixed magnetization $\boldsymbol{f}$ can be setup as a nonlinear least squares problem of the following cost function:

$$\Psi(\boldsymbol{z}) = \frac{1}{2} \left\| \boldsymbol{y} - \boldsymbol{s}(\boldsymbol{z}) \right\|^2 , \tag{B.1}$$

where $\boldsymbol{s} : \mathbb{C}^N \to \mathbb{C}^M$ is the discrete valued MR signal equation:

$$\begin{aligned} \boldsymbol{s}(\boldsymbol{z}) &= \left[s(t_1; \boldsymbol{z}) , \ldots , s(t_M; \boldsymbol{z}) \right] \\ s(t; \boldsymbol{z}) &\triangleq \Phi(\vec{k}(t)) \sum_{n=1}^{N} f(\vec{r}_n) \, \mathrm{e}^{-t z(\vec{r}_n)} \, \mathrm{e}^{-i 2\pi \left(\vec{k}(t) \cdot \vec{r}_n \right)} , \\ \boldsymbol{z} &= \left[z(\vec{r}_1) , \ldots , z(\vec{r}_N) \right] , \; \boldsymbol{f} = \left[f(\vec{r}_1) , \ldots , f(\vec{r}_N) \right] . \end{aligned} \tag{B.2}$$

It would be advantageous to find a majorizer for this cost function since the linear approximation discussed in Chapter 3 and Chapter 5 does not guarantee monotonic descent for the cost function in (B.1), which is what we would like to minimize.

To find the majorizer we first examine the gradient and then the Hessian of the cost function. The row gradient ∇ of $\Psi(\boldsymbol{z})$ is given as follows:

$$\nabla \Psi(\boldsymbol{z}) = \left(\boldsymbol{s}(\boldsymbol{z}) - \boldsymbol{y} \right) \nabla \boldsymbol{s}(\boldsymbol{z}) ,$$

and the Hessian as follows:

$$\nabla^2 \Psi(\boldsymbol{z}) = \sum_{m=1}^{M} \left(s(t_m; \boldsymbol{z}) - y_m\right) \boldsymbol{H}_m + \boldsymbol{\nabla} \boldsymbol{s}(\boldsymbol{z}) \, \boldsymbol{\nabla} \boldsymbol{s}(\boldsymbol{z}) \,,$$

where $\boldsymbol{\nabla}$ is the column gradient. For the MR signal equation in (B.2) we now get:

$$\begin{aligned}
\left[\boldsymbol{\nabla} \boldsymbol{s}(\boldsymbol{z})\right]_{mn} &= \left[\boldsymbol{A}(\boldsymbol{z})\right]_{mn} = \Phi(\vec{k}(t_m)) \sum_{n=1}^{N} f(\vec{r}_n) \, (-t_m) \, \mathrm{e}^{-t_m z(\vec{r}_n)} \, \mathrm{e}^{-i2\pi\left(\vec{k}(t_m) \cdot \vec{r}_n\right)} \\
\left[\boldsymbol{\nabla} \boldsymbol{s}(\boldsymbol{z})\right]_{lm} &= \left[\boldsymbol{A}(\boldsymbol{z})'\right]_{lm} = \Phi(\vec{k}(t_m))^{\,*} \sum_{n=1}^{N} f(\vec{r}_l)^* \, (-t_m) \, \mathrm{e}^{-t_m z(\vec{r}_l)^*} \, \mathrm{e}^{i2\pi\left(\vec{k}(t_m) \cdot \vec{r}_l\right)} \\
\left[\boldsymbol{\nabla} \boldsymbol{s}(\boldsymbol{z}) \, \boldsymbol{\nabla} \boldsymbol{s}(\boldsymbol{z})\right]_{ln} &= \left[\boldsymbol{A}(\boldsymbol{z})' \, \boldsymbol{A}(\boldsymbol{z})\right]_{ln} = \sum_{m=1}^{M} \left|\Phi(\vec{k}(t_m))\right|^2 t_m^2 f(\vec{r}_l)^* \, f(\vec{r}_n) \, \mathrm{e}^{-t_m (z(\vec{r}_l)^* + z(\vec{r}_n))} \\
&\qquad \cdot \mathrm{e}^{-i2\pi\left(\vec{k}(t_m) \cdot (\vec{r}_n - \vec{r}_l)\right)} \\
\boldsymbol{H}_m &= \Phi(\vec{k}(t_m)) \, t_m^2 \cdot \operatorname{diag}\{[h_1, \ldots, h_N]\} \\
h_n &\triangleq f(\vec{r}_n) \, \mathrm{e}^{-t_m z(\vec{r}_n)} \, \mathrm{e}^{-i2\pi\left(\vec{k}(t_m) \cdot \vec{r}_n\right)} \,,
\end{aligned}$$

where * is the complex conjugate.

If we can find matrices $\boldsymbol{U}_m$ and $\boldsymbol{U}$ for all $\boldsymbol{z}$ such that:

$$\left(s(t_m; \boldsymbol{z}) - y_m\right) \boldsymbol{H}_m \preceq \boldsymbol{U}_m, \qquad \boldsymbol{\nabla} \boldsymbol{s}(\boldsymbol{z}) \, \boldsymbol{\nabla} \boldsymbol{s}(\boldsymbol{z}) \preceq \boldsymbol{U}, \tag{B.3}$$

where $\preceq$ is defined as:

$$\boldsymbol{M}(\boldsymbol{z}) \preceq \boldsymbol{K} \;\Leftrightarrow\; \boldsymbol{x}' \boldsymbol{M}(\boldsymbol{z}) \, \boldsymbol{x} \leq \boldsymbol{x}' \boldsymbol{K} \boldsymbol{x}, \; \forall \boldsymbol{x} \in \mathbb{C}, \tag{B.4}$$

we can find a *quadratic surrogate* that majorizes (B.1) with the following curvature:

$$\boldsymbol{J} = \sum_{m} \boldsymbol{U}_m + \boldsymbol{U}.$$

The objective is then to find the matrices $\boldsymbol{U}_m$ and $\boldsymbol{U}$ that satisfy (B.3) and are not dependent on $\boldsymbol{z}$. To do this we utilize the following inequality:

$$\boldsymbol{x}'\boldsymbol{M}(\boldsymbol{z})\,\boldsymbol{x} \leq \|\boldsymbol{M}(\boldsymbol{z})\|_{Frob}\,\|\boldsymbol{x}\|^2, \tag{B.5}$$

and the following assumptions:

$$f_{\max} \triangleq \max(|f(\cdot)|) \tag{B.6}$$

$$z_{\mathrm{R}}(\cdot) \geq 0 \;\Rightarrow\; \mathrm{e}^{-t_m(z_{\mathrm{R}}(\vec{r}_l)+z_{\mathrm{R}}(\vec{r}_n))} \leq 1. \tag{B.7}$$

First we would like to find a matrix $\boldsymbol{U}$, such that $\boldsymbol{x}'\left(\nabla\boldsymbol{s}(\boldsymbol{z})\,\nabla\boldsymbol{s}(\boldsymbol{z})\right)\boldsymbol{x} \leq \boldsymbol{x}'\boldsymbol{U}\boldsymbol{x}$. To do this we use (B.5). We now have the following:

$$\|\nabla\boldsymbol{s}(\boldsymbol{z})\,\nabla\boldsymbol{s}(\boldsymbol{z})\|_{Frob} = \sqrt{\sum_{l=1}^{N}\sum_{n=1}^{N}\left|\left[\boldsymbol{A}(\boldsymbol{z})'\,\boldsymbol{A}(\boldsymbol{z})\right]_{ln}\right|^2}. \tag{B.8}$$

We can show using the triangle inequality, (B.6) and (B.7) that:

$$\begin{aligned}
\left|\left[\boldsymbol{A}(\boldsymbol{z})'\,\boldsymbol{A}(\boldsymbol{z})\right]_{ln}\right| &= \left|\sum_{m=1}^{M}\Phi(\vec{k}(t_m))^{2}t_m^2 f(\vec{r}_l)^*\,f(\vec{r}_n)\,\mathrm{e}^{-t_m(z(\vec{r}_l)^*+z(\vec{r}_n))}\,\mathrm{e}^{-i2\pi\left(\vec{k}(t_m)\cdot(\vec{r}_n-\vec{r}_l)\right)}\right| \\
&\leq \sum_{m=1}^{M}\left|\Phi(\vec{k}(t_m))^{2}t_m^2 f(\vec{r}_l)^*\,f(\vec{r}_n)\,\mathrm{e}^{-t_m(z(\vec{r}_l)^*+z(\vec{r}_n))}\,\mathrm{e}^{-i2\pi\left(\vec{k}(t_m)\cdot(\vec{r}_n-\vec{r}_l)\right)}\right| \\
&= \sum_{m=1}^{M}\left|\Phi(\vec{k}(t_m))\right|^2 t_m^2\,|f(\vec{r}_l)|\,|f(\vec{r}_n)|\,\mathrm{e}^{-t_m(z_{\mathrm{R}}(\vec{r}_l)+z_{\mathrm{R}}(\vec{r}_n))} \\
&\leq \sum_{m=1}^{M}\left|\Phi(\vec{k}(t_m))\right|^2 t_m^2 f_{\max}^2,
\end{aligned}$$

and if we define the constant α_1 as:

$$\alpha_1 \triangleq \sum_{m=1}^{M}\left|\Phi(\vec{k}(t_m))\right|^2 t_m^2 f_{\max}^2,$$

we can use that and (B.8) to show that the following inequalities hold:

$$\left\|\boldsymbol{\nabla} \boldsymbol{s}(\boldsymbol{z})\, \nabla \boldsymbol{s}(\boldsymbol{z})\right\|_{Frob} \leq \sqrt{\sum_{l=1}^{N} \sum_{n=1}^{N} \alpha_1^2} \leq \sqrt{N^2 \alpha_1^2} = N\alpha_1.$$

If we define $\alpha \triangleq N\alpha_1$ we can use this to show the following:

$$\begin{aligned} \boldsymbol{x}' \left(\boldsymbol{\nabla} \boldsymbol{s}(\boldsymbol{z})\, \nabla \boldsymbol{s}(\boldsymbol{z})\right) \boldsymbol{x} &\leq \left\|\boldsymbol{\nabla} \boldsymbol{s}(\boldsymbol{z})\, \nabla \boldsymbol{s}(\boldsymbol{z})\right\|_{Frob} \|\boldsymbol{x}\|^2 \\ &\leq \alpha \cdot \|\boldsymbol{x}\|^2 = \boldsymbol{x}' \left(\alpha \boldsymbol{I}\right) \boldsymbol{x}. \end{aligned}$$

From this we see that $\boldsymbol{U} = \alpha \boldsymbol{I}$ where $\boldsymbol{I}$ is the identity matrix.

Next is to find the series of matrices $\boldsymbol{U}_m$ such that $\boldsymbol{x}' \left(\left(s(t_m; \boldsymbol{z}) - y_m\right) \boldsymbol{H}_m\right) \boldsymbol{x} \leq \boldsymbol{x}' \boldsymbol{U}_m \boldsymbol{x}$. Again we use the inequality in (B.5). The Frobenius norm is as follows:

$$\left\|\left(s(t_m; \boldsymbol{z}) - y_m\right) \boldsymbol{H}_m\right\|_{Frob} = \left|\left(s(t_m; \boldsymbol{z}) - y_m\right)\right| \cdot \left\|\boldsymbol{H}_m\right\|_{Frob}. \tag{B.9}$$

We now get using the triangle inequality, (B.6) and (B.7) that:

$$\begin{aligned} |s(t_m; \boldsymbol{z}) - y_m| &\leq |s(t_m; \boldsymbol{z})| + |y_m| \\ &= \left|\Phi(\vec{k}(t)) \sum_{n=1}^{N} f(\vec{r}_n)\, \mathrm{e}^{-tz(\vec{r}_n)}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)}\right| + |y_m| \\ &\leq \left|\Phi(\vec{k}(t))\right| \cdot \left|\sum_{n=1}^{N} f(\vec{r}_n)\, \mathrm{e}^{-tz(\vec{r}_n)}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)}\right| + |y_m| \\ &\leq \left|\Phi(\vec{k}(t))\right| \cdot \left(\sum_{n=1}^{N} \left|f(\vec{r}_n)\, \mathrm{e}^{-tz(\vec{r}_n)}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t)\cdot\vec{r}_n\right)}\right|\right) + |y_m| \\ &\leq \left|\Phi(\vec{k}(t))\right| \cdot \left(\sum_{n=1}^{N} f_{\max}\right) + |y_m| \\ &= \left|\Phi(\vec{k}(t))\right| N f_{\max} + |y_m|. \end{aligned}$$

The Frobenius norm of $\boldsymbol{H}_m$ is given as follows:

$$\|\boldsymbol{H}_m\|_{Frob} = \sqrt{\sum_{l=1}^{N}\sum_{n=1}^{N} \left|[\boldsymbol{H}_m]_{ln}\right|^2} = \sqrt{\sum_{n=1}^{N} \left|[\boldsymbol{H}_m]_{nn}\right|^2},$$

since $\boldsymbol{H}_m$ is a diagonal matrix. We now get for the diagonal elements:

$$\begin{aligned}\left|[\boldsymbol{H}_m]_{nn}\right| &= \left|\Phi(\vec{k}(t_m))\, t_m^2 f(\vec{r}_n)\, \mathrm{e}^{-t_m z(\vec{r}_n)}\, \mathrm{e}^{-i2\pi\left(\vec{k}(t_m)\cdot\vec{r}_n\right)}\right| \\ &\leq \left|\Phi(\vec{k}(t_m))\right| t_m^2 f_{\max}.\end{aligned}$$

To simplify, we define the following two constants:

$$\beta_{1,m} \triangleq \left|\Phi(\vec{k}(t))\right| N f_{\max} + |y_m|\,, \quad \beta_{2,m} \triangleq \left|\Phi(\vec{k}(t_m))\right| t_m^2 f_{\max}.$$

Using these definitions we can now find an upper bound on Frobenius norm in (B.9) that is not dependent on $\boldsymbol{z}$ as follows:

$$\|(s(t_m;\boldsymbol{z}) - y_m)\,\boldsymbol{H}_m\|_{Frob} \leq \beta_{1,m}\sqrt{\sum_{n=1}^{N} \beta_{2,m}^2} = \sqrt{N}\beta_{1,m}\beta_{2,m}.$$

If we define $\beta_m \triangleq \sqrt{N}\beta_{1,m}\beta_{2,m}$, we can show that:

$$\begin{aligned}\boldsymbol{x}'\left((s(t_m;\boldsymbol{z}) - y_m)\,\boldsymbol{H}_m\right)\boldsymbol{x} &\leq \|(s(t_m;\boldsymbol{z}) - y_m)\,\boldsymbol{H}_m\|_{Frob}\,\|\boldsymbol{x}\|^2 \\ &\leq \beta_m\,\|\boldsymbol{x}\|^2 = \boldsymbol{x}'\left(\beta_m \boldsymbol{I}\right)\boldsymbol{x}.\end{aligned}$$

From this we can see that if $\boldsymbol{U}_m = \beta_m \boldsymbol{I}$.

From this discussion we see that there does exist a *quadratic surrogate* that majorizes (B.1) with the following curvature:

$$\boldsymbol{J} = \sum_m \boldsymbol{U}_m + \boldsymbol{U} = \left(\sum_m \beta_m + \alpha\right)\boldsymbol{I},$$

that implies that the following quadratic surrogate function:

$$Q\big(\boldsymbol{z};\boldsymbol{z}^{(n)}\big) = \Psi(\boldsymbol{z}) + \nabla_z \Psi\big(\boldsymbol{z}^{(n)}\big)\left(\boldsymbol{z}-\boldsymbol{z}^{(n)}\right) + \frac{\left(\sum_m \beta_m + \alpha\right)}{2}\left\|\boldsymbol{z}-\boldsymbol{z}^{(n)}\right\|^2,$$

where $\boldsymbol{z}^{(n)}$ is iteration n when optimizing (B.1). The update of $\boldsymbol{z}$ in the minimization is then:

$$\boldsymbol{z}^{(n+1)} = \boldsymbol{z}^{(n)} + \left(\sum_m \beta_m + \alpha\right)\nabla_z \Psi\big(\boldsymbol{z}^{(n)}\big).$$

If we use this surrogate it guarantees that $\Psi(\boldsymbol{z})$ in (B.1) descends monotonically.

Bibliography

[1] D. G. Nishimura, *Principles of magnetic resonance imaging*. Department of EE: Stanford university, 1996.

[2] J. P. Wansapura, S. K. Holland, R. S. Dunn, and W. S. Ball, Jr., "NMR relaxation times in the human brain at 3.0 Tesla," *J. Mag. Res. Im.*, vol. 9, no. 4, pp. 531–8, Apr. 1999.

[3] K. P. Whittall, A. L. Mackay, D. A. Graeb, R. A. Nugent, D. K. B. Li, and D. W. Paty, "*In vivo* measurement of T_2 distributions and water contents in normal human brain," *Mag. Res. Med.*, vol. 37, no. 1, pp. 34–43, Jan. 1997.

[4] G. J. Stanisz, E. E. Odrobina, J. Pun, M. Escaravage, S. J. Graham, M. J. Bronskill, and R. M. Henkelman, "T_1, T_2 relaxation and magnetization transfer in tissue at 3T," *Mag. Res. Med.*, vol. 54, no. 3, pp. 507–12, Sep. 2005.

[5] E. M. Haacke, R. W. Brown, M. R. Thompson, and R. Venkatesan, *Magnetic resonance imaging: Physical principles and sequence design*. New York: Wiley, 1999.

[6] E. R. McVeigh, R. M. Henkelman, and M. J. Bronskill, "Noise and filtration in magnetic resonance imaging," *Med. Phys.*, vol. 12, no. 5, p. 586, Sep. 1985.

[7] E. Schneider and G. Glover, "Rapid in vivo proton shimming," *Mag. Res. Med.*, vol. 18, no. 2, pp. 335–47, Apr. 1991.

[8] J. A. Fessler, D. Yeo, and D. C. Noll, "Regularized fieldmap estimation in MRI," in *Proc. IEEE Intl. Symp. Biomed. Imag.*, 2006, pp. 706–9.

[9] A. Maeda, K. Sano, and T. Yokoyama, "Reconstruction by weighted correlation for MRI with time-varying gradients," *IEEE Trans. Med. Imag.*, vol. 7, no. 1, pp. 26–31, Mar. 1988.

[10] V. Rasche, R. Proksa, R. Sinkus, P. Börnert, and H. Eggers, "Resampling of data between arbitrary grids using convolution interpolation," *IEEE Trans. Med. Imag.*, vol. 18, no. 5, pp. 385–92, May 1999.

[11] C. H. Meyer, B. S. Hu, D. G. Nishimura, and A. Macovski, "Fast spiral coronary artery imaging," *Mag. Res. Med.*, vol. 28, no. 2, pp. 202–13, Dec. 1992.

[12] D. C. Noll, J. A. Fessler, and B. P. Sutton, "Conjugate phase MRI reconstruction with spatially variant sample density correction," *IEEE Trans. Med. Imag.*, vol. 24, no. 3, pp. 325–36, Mar. 2005.

[13] H. Sedarat and D. G. Nishimura, "On the optimality of the gridding reconstruction algorithm," *IEEE Trans. Med. Imag.*, vol. 19, no. 4, pp. 306–17, Apr. 2000.

[14] H. Choi and D. C. Munson, "Analysis and design of minimax-optimal interpolators," *IEEE Trans. Sig. Proc.*, vol. 46, no. 6, pp. 1571–79, Jun. 1998.

[15] J. I. Jackson, C. H. Meyer, D. G. Nishimura, and A. Macovski, “Selection of a convolution function for Fourier inversion using gridding,” *IEEE Trans. Med. Imag.*, vol. 10, no. 3, pp. 473–8, Sep. 1991.

[16] P. J. Beatty, D. G. Nishimura, and J. M. Pauly, “Rapid gridding reconstruction with a minimal oversampling ratio,” *IEEE Trans. Med. Imag.*, vol. 24, no. 6, pp. 799–808, Jun. 2005.

[17] D. C. Noll, C. H. Meyer, J. M. Pauly, D. G. Nishimura, and A. Macovski, “A homogeneity correction method for magnetic resonance imaging with time-varying gradients,” *IEEE Trans. Med. Imag.*, vol. 10, no. 4, pp. 629–37, Dec. 1991.

[18] L.-C. Man, J. M. Pauly, and A. Macovski, “Multifrequency interpolation for fast off-resonance correction,” *Mag. Res. Med.*, vol. 37, no. 5, pp. 785–92, May 1997.

[19] J. A. Fessler, S. Lee, V. T. Olafsson, H. R. Shi, and D. C. Noll, “Toeplitz-based iterative image reconstruction for MRI with correction for magnetic field inhomogeneity,” *IEEE Trans. Sig. Proc.*, vol. 53, no. 9, pp. 3393–402, Sep. 2005.

[20] B. P. Sutton, D. C. Noll, and J. A. Fessler, “Fast, iterative image reconstruction for MRI in the presence of field inhomogeneities,” *IEEE Trans. Med. Imag.*, vol. 22, no. 2, pp. 178–88, Feb. 2003.

[21] M. Hestenes and E. Stiefel, “Methods of conjugate gradients for solving linear systems,” *J. Research Nat. Bur. Standards*, vol. 49, pp. 409–36, 1952.

[22] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. New York: Springer–Verlag, 2001.

[23] D. W. Wilson and B. M. W. Tsui, “Spatial resolution properties of FB and ML-EM reconstruction methods,” in *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, vol. 2, 1993, pp. 1189–93.

[24] J. A. Fessler and W. L. Rogers, “Spatial resolution properties of penalized-likelihood image reconstruction methods: Space-invariant tomographs,” *IEEE Trans. Im. Proc.*, vol. 5, no. 9, pp. 1346–58, Sep. 1996.

[25] J. A. Fessler and B. P. Sutton, “Nonuniform fast Fourier transforms using min-max interpolation,” *IEEE Trans. Sig. Proc.*, vol. 51, no. 2, pp. 560–74, Feb. 2003.

[26] D. C. Noll, J. M. Pauly, C. H. Meyer, D. G. Nishimura, and A. Macovski, “Deblurring for non-2D fourier transform magnetic resonance imaging,” *Mag. Res. Med.*, vol. 25, no. 2, pp. 319–333, Jun. 1992.

[27] E. Ahunbay and J. G. Pipe, “Rapid method for deblurring spiral MR images,” *Mag. Res. Med.*, vol. 44, no. 3, pp. 491–4, Sep. 2000.

[28] S. Ogawa, T.-M. Lee, A. S. Nayak, and P. Glynn, “Oxygenation-sensitive contrast in magnetic resonance image of rodent brain at high magnetic fields,” *Mag. Res. Med.*, vol. 14, no. 1, pp. 68–78, Nov. 1990.

[29] S. Ogawa, T.-M. Lee, A. R. Kay, and D. W. Tank, "Brain magnetic resonance imaging with contrast dependent on blood oxygenation," *Proc. Natl. Acad. Sci. USA*, vol. 87, no. 24, pp. 9868–72, Dec. 1990.

[30] K. K. Kwong, J. W. Belliveau, D. A. Chesler, I. E. Goldberg, R. M. Weisskoff, B. P. Poncelet, D. N. Kennedy, B. E. Hoppel, M. S. Cohen, R. Turner, H.-M. Cheng, T. J. Brady, and B. R. Rosen, "Dynamic magnetic resonance imaging of human brain activity during primary sensory stimulation," *Proc. Natl. Acad. Sci. USA*, vol. 89, no. 12, pp. 5675–79, Jun. 1992.

[31] S. Ogawa, D. W. Tank, R. Menon, J. M. Ellermann, S. Kim, H. Merkle, and K. Ugurbil, "Intrinsic signal changes accompanying sensory stimulation: Functional brain mapping with magnetic resonance imaging," *Proc. Natl. Acad. Sci. USA*, vol. 89, no. 13, pp. 5951–55, Jul. 1992.

[32] P. A. Bandettini, E. C. Wong, R. S. Hinks, R. S. Tikofsky, and J. S. Hyde, "Time course EPI of human brain function during task activation," *Mag. Res. Med.*, vol. 25, no. 2, pp. 390–97, Dec. 1992.

[33] C. S. Springer, C. S. Patlak, I. Pályka, and W. Huang, "Principles of susceptibility constrast-based functional MRI: The sign of the functional MRI response," in *Functional MRI*, C. T. W. Moonen and P. A. Bandettini, Eds. Springer-Verlag, 1999, pp. 91–102.

[34] S. Ogawa, R. S. Menon, D. W. Tank, S. G. Kim, H. Merkle, J. M. Ellermann, and K. Ugurbil, "Functional brain mapping by blood oxygenation level-dependent contrast magnetic resonance imaging. A comparison of signal characteristics with a biophysical model," *Biophys. J.*, vol. 64, no. 3, pp. 803–12, Mar. 1993.

[35] J. S. Gati, R. S. Menon, K. Ugurbil, and B. K. Rutt, "Experimental determination of the BOLD field strength dependence in vessels and tissue," *Mag. Res. Med.*, vol. 38, no. 2, pp. 296–302, Aug. 1997.

[36] A. B. A. Wennerberg, T. Jonsson, H. Forssberg, and T.-Q. Li, "A comparative fMRI study: T_2^*-weighted imaging versus R_2^* mapping," *NMR in Biomedicine*, vol. 14, no. 1, pp. 41–7, Feb. 2001.

[37] G. E. Hagberg, M. Bianciardi, F. Patria, and I. Indovina, "Quantitative NumART$_2^*$ mapping in functional MRI studies at 1.5 T," *Mag. Res. Im.*, vol. 21, no. 10, pp. 1241–9, Dec. 2003.

[38] H. Schomberg, "Off-resonance correction of MR images," *IEEE Trans. Med. Imag.*, vol. 18, no. 6, pp. 481–95, Jun. 1999.

[39] P.-M. Robitaille and L. Berliner, *Ultra High Field Magnetic Resonance Imaging*, 1st ed., ser. Biological Magnetic Resonance. Springer, Oct. 2006.

[40] X. Hu and S.-G. Kim, "Reduction of signal fluctuation in functional MRI using navigator echoes," *Mag. Res. Med.*, vol. 31, no. 5, pp. 495–503, May 1994.

[41] S. Lee, J. A. Fessler, and D. Noll, "A simultaneous estimation of field inhomogeneity and R2* maps using extended rosette trajectory," in *Proc. Intl. Soc. Mag. Reson. Med*, 2002, p. 2327.

[42] 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," in *Proc. Intl. Soc. Mag. Reson. Med*, 2002, p. 1323.

[43] D. B. Twieg, "Parsing local signal evolution directly from a single-shot MRI signal: A new approach for fMRI," *Mag. Res. Med.*, vol. 50, no. 5, pp. 1043–52, Nov. 2003.

[44] V. Roopchansingh, R. W. Cox, A. Jesmanowicz, B. D. Ward, and J. S. Hyde, "Single-shot magnetic field mapping embedded in echo-planar time-course imaging," *Mag. Res. Med.*, vol. 50, no. 4, pp. 839–43, Oct. 2003.

[45] B. P. Sutton, D. C. Noll, and J. A. Fessler, "Dynamic field map estimation using a spiral-in / spiral-out acquisition," *Mag. Res. Med.*, vol. 51, no. 6, pp. 1194–204, Jun. 2004.

[46] ——, "Compensating for withinvoxel susceptibility gradients in BOLD fMRI," in *Proc. Intl. Soc. Mag. Reson. Med*, vol. 11, 2004, p. 349.

[47] J. A. Fessler and D. C. Noll, "Model-based MR image reconstruction with compensation for through-plane field inhomogeneity," in *Proc. IEEE Intl. Symp. Biomed. Imag.*, Apr. 2007, pp. 920–3.

[48] G. H. Glover, T.-Q. Li, and D. Ress, "Image-based method for retrospective correction of physiological motion effects in fMRI: RETROICOR," *Mag. Res. Med.*, vol. 44, no. 1, pp. 162–7, Jul. 2000.

[49] X. Hu, T. H. Le, T. Parrish, and P. Erhard, "Retrospective estimation and correction of physiological fluctuation in functional MRI," *Mag. Res. Med.*, vol. 34, no. 2, pp. 201–12, Aug. 1995.

[50] O. Speck and J. Hennig, "Functional imaging by I_0 and T_2^*-parameter mapping using multi-image EPI," *Mag. Res. Med.*, vol. 40, no. 2, pp. 243–8, Aug. 1998.

[51] M. Grüne, F. Pillekamp, W. Schwindt, and M. Hoehn, "Gradient echo time dependence and quantitative parameter maps for somatosensory activation in rats at 7 T," *Mag. Res. Med.*, vol. 42, no. 1, pp. 118–26, Jul. 1999.

[52] M. Barth, A. Metzler, M. Klarhöfer, S. Röll, E. Moser, and D. Leibfritz, "Functional MRI of the human motor cortex using single-shot multiple gradient-echo spiral imaging," *Mag. Res. Med.*, vol. 17, no. 9, pp. 1239–43, Nov. 1999.

[53] O. Speck, T. Ernst, and L. Chang, "Biexponential modeling of multigradient-echo MRI data of the brain," *Mag. Res. Med.*, vol. 45, no. 6, pp. 1116–21, Jun. 2001.

[54] D. A. Yablonskiy, “Quantitation of intrinsic magnetic susceptibility-related effects in a tissue matrix. Phantom study,” *Mag. Res. Med.*, vol. 39, no. 3, pp. 417–28, Mar. 1998.

[55] J. Raz, E. J. Fernandez, and J. Gillespie, “Modeling NMR lineshapes using logspline density functions,” *J. Mag. Res.*, vol. 127, no. 2, pp. 173–83, Aug. 1997.

[56] V. T. Olafsson, D. C. Noll, and J. A. Fessler, “Fast joint reconstruction of dynamic R_2^* and field maps in functional MRI,” *IEEE Trans. Med. Imag.*, vol. 27, no. 9, pp. 1177–88, Sep. 2008.

[57] ——, “New approach for estimating ΔR2* in fMRI,” in *Proc. Intl. Soc. Mag. Reson. Med*, 2003, p. 132.

[58] V. Olafsson, J. A. Fessler, and D. C. Noll, “Dynamic update of R_2^* and field map in fMRI,” in *Proc. Intl. Soc. Mag. Reson. Med*, 2004, p. 45.

[59] V. Olafsson, D. C. Noll, and J. A. Fessler, “Spatial resolution analysis of iterative image reconstruction with separate regularization of real and imaginary parts,” in *Proc. IEEE Intl. Symp. Biomed. Imag.*, 2006, pp. 5–8.

[60] M. S. Cohen, “Parametric analysis of fMRI data using linear systems methods,” *Neuroimage*, vol. 6, no. 2, pp. 93–103, Aug. 1997.

[61] S. D. Forman, J. D. Cohen, M. Fitzgerald, W. F. Eddy, M. A. Mintun, and D. C. Noll, “Improved assessment of significant activation in functional magnetic resonance imaging (fMRI): use of a cluster-size threshold,” *Mag. Res. Med.*, vol. 33, no. 5, pp. 636–47, May 1995.

[62] D. C. Noll, C. R. Genovese, L. E. Nystrom, A. L. Vazquez, S. D. Forman, W. F. Eddy, and J. D. Cohen, “Estimating test-retest reliability in functional MR imaging. II: Application to motor and cognitive activation studies,” *Mag. Res. Med.*, vol. 38, no. 3, pp. 508–17, Sep. 1997.

[63] V. Olafsson, J. A. Fessler, and D. C. Noll, “Dynamic updates of R2* and field map in fMRI using a spiral-in quick-spiral-out k-space trajectory,” in *Proc. Intl. Soc. Mag. Reson. Med*, 2006, p. 2838.

[64] J. A. Fessler and D. C. Noll, “Iterative image reconstruction in MRI with separate magnitude and phase regularization,” in *Proc. IEEE Intl. Symp. Biomed. Imag.*, 2004, pp. 209–12.

[65] M. Cetin and W. C. Karl, “Feature-enhanced synthetic aperture radar image formation based on nonquadratic regularization,” *IEEE Trans. Im. Proc.*, vol. 10, no. 4, pp. 623–31, Apr. 2001.

[66] S. Sotthivirat and J. A. Fessler, “Penalized-likelihood image reconstruction for digital holography,” *J. Opt. Soc. Am. A*, vol. 21, no. 5, pp. 737–50, May 2004.

[67] V. Olafsson, J. A. Fessler, and D. C. Noll, “Fast and motion robust r2* reconstruction for functional MRI,” in *Proc. Intl. Soc. Mag. Reson. Med*, 2009, p. 2831.

[68] V. Olafsson, M. Ulfarsson, J. A. Fessler, and D. C. Noll, “Reducing effects of drift in fMRI data using joint reconstruction of R2* and field maps,” in *Proc. Intl. Soc. Mag. Reson. Med*, 2007, p. 1978.

[69] R. S. Menon, S. Ogawa, D. W. Tank, and K. Uğurbil, “4 Tesla gradient recalled echo characteristics of photic stimulation-induced signal changes in the human primary visual cortex,” *Mag. Res. Med.*, vol. 30, no. 3, pp. 380–6, Sep. 1993.

[70] A. K. Funai, J. A. Fessler, D. T. B. Yeo, V. T. Olafsson, and D. C. Noll, “Regularized field map estimation in MRI,” *IEEE Trans. Med. Imag.*, vol. 27, no. 10, pp. 1484–1494, Oct. 2008.

[71] D. C. Noll, “Multishot rosette trajectories for spectrally selective MR imaging,” *IEEE Trans. Med. Imag.*, vol. 16, no. 4, pp. 372–7, Aug. 1997.

[72] C. R. Genovese, D. C. Noll, and W. F. Eddy, “Estimating test-retest reliability in functional MR imaging. I: Statistical methodology.” *Mag. Res. Med.*, vol. 38, no. 3, pp. 497–507, Sep. 1997.

[73] C. Y. Yip, D. Yoon, V. Olafsson, S. Lee, W. A. Grissom, J. A. Fessler, and D. C. Noll, “Spectral-spatial pulse design for through-plane phase precompensatory slice selection in T_2^*-weighted functional MRI,” *Mag. Res. Med.*, vol. 61, no. 4, Apr. 2009.