SPHERICAL FUNCTION REGULARIZATION FOR PARALLEL MRI RECONSTRUCTION

Yonggui Zhu* Tuomo Valkonen†

2018-02-04

Abstract From the optimization point of view, a difficulty with parallel MRI with simultaneous coil sensitivity estimation is the multiplicative nature of the non-linear forward operator: the image being reconstructed and the coil sensitivities compete against each other, causing the optimization process to be very sensitive to small perturbations. This can, to some extent, be avoided by regularizing the unknown in a suitably "orthogonal" fashion. In this paper, we introduce such a regularization based on spherical function bases. To perform this regularization, we represent efficient recurrence formulas for spherical Bessel functions and associated Legendre functions. Numerically, we study the solution of the model with non-linear ADMM. We perform various numerical simulations to demonstrate the efficacy of the proposed model in parallel MRI reconstruction.

Key words Parallel MRI, spherical function, regularization, coil sensitivity, ADMM

1 INTRODUCTION

1.1 PARALLEL MAGNETIC RESONANCE IMAGING

Parallel magnetic resonance imaging (p-MRI) increases acquisition speed by simultaneously using multiple radio frequency (RF) detector coils. This helps avoid some of the time-consuming phase-encoding steps in the MRI process. Although less k-space data is received for each coil, this is compensated by data being available from multiple coils. The first approach to p-MRI was based on an arrangement of J surface coils around the object for MR imaging, one for each k-space line to be acquired [28]. The p-MRI method to achieve routine use was sensitivity encoding, SENSE [44, 46]. In this approach, a discrete Fourier transform is used to reconstruct an aliased image for each element in the array. Then the full full-of-view image is generated from the individual sets of images.

Generally, the MRI signal acquired by receiver coil j is given by

$$s_j(\vec{k}) = \int u(\vec{x})c_j(\vec{x})exp(i\vec{k}\cdot\vec{x})d\vec{x}, \quad j = 1, 2, \ldots, J. \tag{1.1}$$

*School of Science, Communication University of China, P. R. China (ygzhu@cuc.edu.cn)
†Department of Mathematical Sciences, University of Liverpool, United Kingdom (tuomo.valkonen@iki.fi).

Here u is the excited proton density function, c_j the sensitivity profile of the jth coil at $\vec{x}$, and $\vec{k}$ is the chosen k-space trajectory. In discrete form (1.1) can be written

$$s_j(k_m, k_n) = \sum_{m=1}^{N}\sum_{n=1}^{N} u(m,n)c_j(m,n)exp(ik_m m)exp(ik_n n), \quad j = 1, 2, \ldots, J. \tag{1.2}$$

If the coil sensitivity profiles $c_j(m,n)$, $j = 1, 2, \ldots, J$, are known, the system of equations (1.2) can be numerically inverted for u with relative ease [24, 52]. An early direct method to invert (1.2) is to decouple the system of equations in image space under regular sub-sampled pattern like SENSE [36, 45, 46]. Another direct approach is to approximate a sparse inverse by using the coil data in k-space as in SMASH [51] and g-SMASH [11]. SMASH is a partial p-MRI method using multiple coils to speedup acquisition in the course of imaging. Whereas g-SMATH is generalized SMASH method that reconstructs image with the coil data in k-space. However, the MRI signal equation will be increasingly ill-conditioned when the acceleration factor becomes large. The acceleration factor is the ratio of the amount of k-space data required for a fully sampled image to the amount collected in accelerated acquisition. When ill-conditioned, the inversion of the linear system (1.2) will lead to the amplification of noise present in the MRI signal s_l. Therefore regularization methods are required to improve reconstruction quality. Historically employed regularization methods include the truncated singular value decomposition (TSVD) and damped least-squares (DLS) [37].

If the coil sensitivities are not known, it is common to acquire sensitivity information by using a calibration step [23]. For example, the coil sensitivity profiles can be obtained directly from the reference lines in autocalibrating SENSE [41]. The GRAPPA method [22] is the most widely used autocalibrating technique in the determination of coil sensitivities. The coils sensitivities are generally determined from the center of the k-space rather than using all available information. Due to small errors, this leads to residual aliasing artifacts in the reconstruction because. Nonlinear inversion with the joint the estimation of the coil sensitivities c_j and the determination of the proton density image u, can improve reconstruction quality [5, 52, 53, 59].

1.2 NONLINEAR INVERSION FOR P-MRI

Parallel MR imaging can be formulated as a nonlinear inverse problem with a nonlinear forward operator $\mathfrak{F}$, which maps the proton density u and the coil sensitivities $c = (c_j, c_2, \ldots, c_J)^T$ to the measured k-space data g as

$$\mathfrak{F}(u,c) := \left(P\mathcal{F}(u \cdot c_1), P\mathcal{F}(u \cdot c_2), \ldots, P\mathcal{F}(u \cdot c_J)\right)^T = g. \tag{1.3}$$

Here P is the binary sub-sampling mask, $\mathcal{F}$ is the discrete 2D Fourier transform, and $g = (g_1, g_2, \ldots, g_J)^T$ the acquired k-space measurements for J receiver coils. As shown in [34, 52], the problem (1.3) can be solved by the iteratively regularized Gauss-Newton (IRGN) method [4, 9, 19, 26]. The discrepancy principle is used to obtain a suitable level of regularization. In [35] the authors furthermore expanded IRGN method with variational regularization terms to improve reconstruction quality. The method works as follows. Writing $v = (u, c_1, c_2, \ldots, c_J)^T$, and starting from an initial guess v^0, we solve on each step for $\triangle v$ from the linearised problem

$$\min_{\triangle v} \frac{1}{2}\left\|\mathfrak{F}'(v^k)\triangle v + \mathfrak{F}(v^k) - g\right\|_2^2 + \frac{\alpha_k}{2}R_c(c^k + \triangle c) + \beta_k R_u(u^k + \triangle u). \tag{1.4}$$

Then we update $v^{k+1} := v^k + \triangle v$. Here $R_c(c)$ is a regularization functional for penalizing the high Fourier coefficients of the coils $c_j, j = 1, 2, \ldots, J$, and R_u regularizes the image. The regularization parameters α_k and β_k are updated by the formulas $\alpha_{k+1} := q_\alpha \alpha_k$ and $\beta_{k+1} := q_\beta \beta_k$ with $0 < q_\alpha, q_\beta < 1$. More details can be found in [35].

There are many options for the regularisers R_u and R_c in the inverse problems literature. The most basic regularization is the simple L^2 penalty $R_u(u) = \frac{1}{2}\|u\|_2^2$. This is used in [34, 35, 52]. Another conventional choice for the image u is $R_u(u) = TV(u)$, the Total Variation [12, 25, 39, 49, 63, 64]. The are two common variants of the total variation, dependent on the choice of pointwise norm used. Restricting ourselves to the finite-dimensional setting, with the two-norm we obtain the isotropic total variation

$$\mathrm{TV}_I(u) = \sum_{m=1}^{N}\sum_{n=1}^{N}\sqrt{|\nabla_1 u(m,n)|^2 + |\nabla_2 u(m,n)|^2},$$

while with the 1-norm we obtain the computationally easier but anisotropic total variation

$$\mathrm{TV}_{l_1}(u) = \sum_{m=1}^{N}\sum_{n=1}^{N}|\nabla_1 u(m,n)| + |\nabla_2 u(m,n)|.$$

In both cases we have used the forward-differences

$$\nabla_1 u(m,n) = \begin{cases} u(m+1,n) - u(m,n), & m < N \\ 0, & m = N, \end{cases} \qquad \nabla_2 u(m,n) = \begin{cases} u(m,n+1) - u(m,n), & n < N \\ 0 & n = N. \end{cases}$$

If the penalty parameter β_k becomes large, TV regularization will generate staircasing artefacts. This can be avoided through the use of second-order Total Generalized Variation (TGV) [10, 33, 56].

For the regularization term $R_c(c)$, one choice from [4, 9, 19, 26] is to take $R_c(c) = \|w \cdot \mathcal{F}c\|_2^2$, where w is an weighting operator that penalizes high Fourier coefficients. It is well known that coil sensitivities are generally rather smooth functions that vary only slowly and do not have sharp edges. This supports the use of quadratic regularization of the gradients the Tikhonov-regularized model in [7]. Specifically, instead of the iteratively regularized IRGN approach (1.4), the authors directly solve for $\hat{v} = (\hat{u}, \hat{c_1}, \hat{c_2}, \ldots, \hat{c_J})$ the variational model

$$\min_{v=(u,c_1,c_2,\ldots,c_J)^T} \frac{1}{2}\sum_{j=1}^{J}\left\|P\mathcal{F}(G(v))_j - g_j\right\|_2^2 + \alpha_0 R_u(u) + R_c(c), \tag{1.5}$$

where $G(v) = (uc_1, uc_2, \ldots, uc_J)^T$, $R_u(u) = \mathrm{TV}_I(u)$, and $R_c(c) = \sum_{j=1}^{J}\alpha_j\|\nabla c_j\|_{2,2}$.

1.3 CONTRIBUTIONS

From the optimization point of view, a difficulty with both models (1.4) and (1.5) is the multiplicative nature of $G(v)$. It can cause u and c_j to compete against each other. Therefore, besides physical considerations, one goal in the design of the regularisers R_u and R_c would be to try

to make u and c_j in some vague sense "orthogonal", to avoid this competition. One approach to such vague orthogonality is to force u piecewise constant and c_j smooth. This is roughly performed by the TV and H^1 regularisers in [7]. Another approach is for u and c_j have very different sparsity structure. This is what we will do in this paper.

Specifically, we will assume that the coil sensitivities can be sparsely represented in a spherical function basis $\{f_l^+\}$, which we introduce in detail in Section 2. Then, with $c_j = \sum_{l=1}^{L} a_l^{(j)} f_l^+$, we will in the variational model (1.5) promote sparsity by taking

$$R_c(c) = \alpha R_a(a) \quad \text{for} \quad R_a(a) = \sum_{j=1}^{J} \sum_{l=1}^{L} |a_l^{(j)}|. \tag{1.6}$$

Therefore, we consider the model

$$\min_{v=(u,a)} \frac{1}{2} \sum_{j=1}^{J} \left\| P\mathcal{F}(G(v))_j - g_j \right\|_2^2 + \alpha_0 R_u(u) + \alpha R_a(a), \tag{1.7}$$

for an appropriate definition of G that we provide in Section 4.

In order to make this model practical, we propose in Section 3 an efficient approach to compute the spherical basis functions based on spherical Bessel functions of the first kind and spherical harmonics. For the computation of spherical Bessel functions, we develop recurrence formulas. Based on the recurrence formula, all other spherical Bessel functions are efficiently calculated via the first two Bessel functions. For the computation of the spherical harmonics, we also provide a means to efficiently compute the associated Legendre functions by establishing in Section 3 a recurrence formula with only four terms. In Section 4 we then present a numerical method for (1.5) with (1.6), based on the alternating direction method of multipliers (ADMM), and study the practical reconstruction performance in Section 5.

2 SPHERICAL BASIS FUNCTION REPRESENTATION OF COIL SENSITIVITIES

According to the principle of reciprocity [27, 29, 30, 62], the coil sensitivity maps can be evaluated from transmit radio frequency field profiles B_1^+. We therefore start by briefly introducing the theory of B_1^+ fields. Let ω, μ, σ, and ε denote the Larmour frequency, the magnetic permeability, the conductivity, and the dielectric permittivity of the material, respectively. The radio frequency (RF) field is denoted by $\vec{B}(\vec{r}) = (B_x(\vec{r}), B_y(\vec{r}), B_z(\vec{r}))^T$ with $\vec{B}(\vec{r}) \in \mathbb{C}^3$ and $\vec{r} \in \mathbb{R}^3$. In the positively rotating frame given in [27, 30, 50], the transmit RF field is

$$B_1^+(\vec{r}) \equiv \frac{B_x(\vec{r}) + iB_y(\vec{r})}{2}.$$

For the detailed introduction of positively rotating frame, see [27]. These fields an be approximated [18, 32, 43, 58] by

$$B_1^+(\vec{r}) \approx \sum_{l=1}^{L} a_l f_l^+(\vec{r}), \tag{2.1}$$

where $f_l^+(\vec{r})$ are the spherical basis functions, L is a small natural number and a_l are complex coefficients. The overall magnetic field $\vec{B}$ can be reduced to the Helmholtz equation [13]

$$\nabla^2\vec{B}(\vec{r}) + \zeta^2\vec{B}(\vec{r}) = 0, \quad \textit{where} \quad \zeta^2 = \varepsilon\mu\omega^2 - i\sigma\omega\mu. \tag{2.2}$$

In spherical coordinates (ρ, θ, ϕ), the equation (2.2) has the solution [31]

$$B_x(\vec{r}) = \sum_{n=0}^{\infty}\sum_{m=-n}^{n}\alpha_n^m f_n^m(\vec{r}), \quad B_y(\vec{r}) = \sum_{n=0}^{\infty}\sum_{m=-n}^{n}\beta_n^m f_n^m(\vec{r}), \quad B_z(\vec{r}) = \sum_{n=0}^{\infty}\sum_{m=-n}^{n}\gamma_n^m f_n^m(\vec{r}), \tag{2.3}$$

where f_n^m are so-called spherical functions. They can be written

$$f_n^m(\rho, \theta, \phi) \equiv j_n(\zeta\rho)Y_n^m(\theta, \phi), \tag{2.4}$$

where j_n is the spherical Bessel function of the first kind of order n, and Y_n^m is the spherical harmonic of order n and degree m. The spherical functions form a basis for the B_1^+ fields by setting

$$f_l^+(\rho, \theta, \phi) = f_n^m(\rho, \theta, \phi) \quad \text{with} \quad l = n^2 + n + m + 1,\ |m| \le n \text{ and } 0 \le n \le \tilde{n}.$$

When signals or objects of approximately spherical shape are considered, fast convergence is expected, so the complex coefficients α_n^m, β_n^m, and γ_n^m in (2.3) should be negligible for $n > \tilde{n}$ with $\tilde{n}$ being a small natural number. We can therefore also expect fast convergence for the spherical function approximation of the B_1^+ field, and by extension the coil sensitivities.

3 EFFICIENT COMPUTATION OF THE SPHERICAL BASIS FUNCTIONS

Our task in the present section is develop efficient recurrence formulas for the computation of the spherical basis functions (2.4). As discussed, this will be based on formulas for the Bessel functions and spherical harmonics.

3.1 A RECURRENCE RELATION FOR THE SPHERICAL BESSEL FUNCTIONS

Following [1], we now develop a recurrence formula for the spherical Bessel functions j_n. We start by recalling that the Bessel function of the first kind, for arbitrary order $\alpha \in \mathbb{R}$, is defined as

$$J_\alpha(x) = \sum_{s=0}^{\infty}\frac{(-1)^s}{s!\Gamma(s+\alpha+1)!}\left(\frac{x}{2}\right)^{\alpha+2s}.$$

Since the Γ satisfies $\Gamma(n) = (n-1)!$ for integral $n \ge 0$ we can in particular write

$$J_n(x) = \sum_{s=0}^{\infty}\frac{(-1)^s}{s!(n+s)!}\left(\frac{x}{2}\right)^{n+2s}.$$

To compute J_n for negative integers, we can use the relationship

$$J_{-n}(x) = (-1)^n J_n(x).$$

In addition, we have the recurrence relationship [1]

$$J_{n-1}(x) + J_{n+1}(x) = \frac{2n}{x} J_n(x). \tag{3.1}$$

We now finally define the spherical Bessel function

$$j_n(x) = \sqrt{\frac{\pi}{2x}} J_{n+1/2}(x).$$

From the recurrence relation (3.1) for the Bessel functions J_n, we then obtain

$$j_{n-1}(x) + j_{n+1}(x) = \frac{2n+1}{x} j_n(x).$$

This can more conveniently be rewritten

$$j_{n+1}(x) = \frac{2n+1}{x} j_n(x) - j_{n-1}(x). \tag{3.2}$$

This is a three-term recurrence relation, so if j_0 and j_1 are known, then any higher-order j_n can be computed from (3.2).

To compute j_0, we recall that Legendre's duplication formula for the Γ function states

$$\Gamma(1+z)\Gamma(z+\frac{1}{2}) = 2^{-2z}\sqrt{\pi}\Gamma(2z+1).$$

For integral z therefore

$$z!(z+\frac{1}{2})! = 2^{-2z-1}\sqrt{\pi}(2z+1)!.$$

Consequently

$$j_n(x) = 2^n x^n \sum_{s=0}^{\infty} \frac{(-1)^s (s+n)!}{s!(2s+2n+1)!} x^{2s}. \tag{3.3}$$

When $n = 0$, we find from (3.3) that

$$j_0(x) = \sum_{s=0}^{\infty} \frac{(-1)^s}{(2s+1)!} x^{2s} = \frac{\sin x}{x}. \tag{3.4}$$

To compute j_1, we set

$$n_n(x) = (-1)^{n+1}\sqrt{\frac{\pi}{2x}} J_{-n-1/2}(x), \quad \text{and} \qquad h_n^{(1)}(x) = j_n(x) + i n_n(x).$$

In [1] it is shown

$$h_n^{(1)}(x) = (-i)^{n+1} \frac{e^{ix}}{x} \sum_{s=0}^{n} \frac{i^s}{s!(2x)^s} \frac{(n+s)!}{(n-s)!},$$

Table 1: The first spherical Bessel functions

n	$j_n(x)$
$n=0$	$j_0(x)=\frac{\sin x}{x}$
$n=1$	$j_1(x)=-\frac{\cos x}{x}+\frac{\sin x}{x^2}$
$n=2$	$j_2(x)=\left(-\frac{1}{x}+\frac{3}{x^3}\right)\sin x-\frac{3}{x^2}\cos x$
$n=3$	$j_3(x)=\left(-\frac{15}{x^4}-\frac{6}{x^2}\right)\sin x+\left(\frac{1}{x}-\frac{15}{x^3}\right)\cos x$

which for $n=1$ gives

$$h_1^{(1)}(x)=e^{ix}\left(-\frac{1}{x}-\frac{i}{x^2}\right).$$

As the real part of $h_1^{(1)}(x)$, we then obtain for $j_1(x)$ the expression

$$j_1(x)=\frac{\sin x}{x^2}-\frac{\cos x}{x}. \tag{3.5}$$

Based on the recurrence (3.2) and the expressions (3.4) and (3.5) for j_0 and j_1, we can now compute any higher-order j_n. We list the first few in Table 1.

3.2 EFFICIENT COMPUTATION OF THE ASSOCIATED LEGENDRE FUNCTIONS

The spherical harmonics Y_n^m are defined in spherical coordinates as [1]

$$Y_n^m(\theta,\phi)\equiv(-1)^m\sqrt{\frac{2n+1}{4\pi}\frac{(n-m)!}{(n+m)!}}P_n^m(\cos\theta)e^{im\phi},\qquad 0\le\theta\le\pi, 0\le\phi\le 2\pi, \tag{3.6}$$

where the associated Legendre function

$$P_n^m(x)=(1-x^2)^{m/2}\frac{d^m}{dx^m}P_n(x),\qquad 0\le m\le n, \tag{3.7}$$

and $P_n(x)$ are the nth-order Legendre polynomials. They are defined as

$$P_n(x)=\sum_{k=0}^{\lfloor n/2\rfloor}(-1)^k\frac{(2n-2k)!}{2^n k!(n-k)!(n-2k)!}x^{n-2k} \tag{3.8}$$

with $\lfloor n/2\rfloor=n/2$ for n even, $(n-1)/2$ for n odd. In particular, it is easy to see that

$$P_0(x)=1\quad\text{and}\quad P_1(x)=x.$$

For $-n\le m<0$, using Leibniz' differentiation formula, we can find that $P_n^m(x)$ and $P_n^{-m}(x)$ are related by [1]

$$P_n^{-m}(x)=(-1)^m\frac{(n-m)!}{(n+m)!}P_n^m(x). \tag{3.9}$$

For $|m| > n$, $P_n^m(x) = 0$.

From (3.7), we have

$$P_n^0(x) = P_n(x).$$

Thus $P_0^0(x) = P_0(x) = 1$, $P_1^0(x) = P_1(x) = x$, $P_1^1(x) = (1-x^2)^{1/2}\frac{d}{dx}P_1(x) = (1-x^2)^{1/2}$. By (3.9), we obtain $P_1^{-1}(x) = -\frac{1}{2}(1-x^2)^{1/2}$. For the sake of the convenience of developing the recurrence relation to find all the P_n^m effectively, we put $P_1^{-1}, P_0^0, P_1^0, P_1^1$ together as

$$P_0^0(x) = 1, \qquad P_1^{-1}(x) = -\frac{1}{2}(1-x^2)^{1/2}, \qquad P_1^0(x) = x, \qquad P_1^1(x) = (1-x^2)^{1/2}. \tag{3.10}$$

Let us define the polynomials $\mathcal{P}_{s+m}^m(x) = P_{s+m}^m(x)(1-x^2)^{-m/2}$, $m \geq 0$. Then the generating function [1, (12.83)]

$$\bar{g}_m(x,t) \equiv \frac{(2m)!}{2^m m!(1-2tx+t^2)^{m+1/2}} = \sum_{s=0}^{\infty} \mathcal{P}_{s+m}^m(x)t^s.$$

Furthermore, following [1, §12.5], we have the recursion

$$\mathcal{P}_{s+m}^m(x) = 2x\mathcal{P}_{s+m-1}^m(x) - \mathcal{P}_{s+m-2}^m(x) + (2m-1)\mathcal{P}_{s+m-1}^{m-1}(x).$$

Therefore

$$\begin{aligned}
P_{s+m}^m(x) &= (1-x^2)^{m/2}\mathcal{P}_{s+m}^m(x) \\
&= (1-x^2)^{m/2}\big(2x\mathcal{P}_{s+m-1}^m(x) - \mathcal{P}_{s+m-2}^m(x) + (2m+1)\mathcal{P}_{s+m-1}^{m-1}(x)\big) \\
&= 2(1-x^2)^{m/2}x\mathcal{P}_{s+m-1}^m(x) - (1-x^2)^{m/2}\mathcal{P}_{s+m-2}^m(x) \\
&\quad + (2m-1)(1-x^2)^{m/2}\mathcal{P}_{s+m-1}^{m-1}(x) \\
&= 2(1-x^2)^{m/2}x(1-x^2)^{-m/2}P_{s-1+m}^m(x) - (1-x^2)^{m/2}(1-x^2)^{-m/2}P_{s-2+m}^m(x) \\
&\quad + (2m-1)(1-x^2)^{m/2}(1-x^2)^{-(m-1)/2}P_{s+m-1}^{m-1}(x) \\
&= 2xP_{s+m-1}^m(x) - P_{s+m-2}^m(x) + (2m-1)(1-x^2)^{1/2}P_{s+m-1}^{m-1}(x).
\end{aligned}$$

That is

$$P_{s+m}^m(x) = 2xP_{s+m-1}^m(x) - P_{s+m-2}^m(x) + (2m-1)(1-x^2)^{1/2}P_{s+m-1}^{m-1}(x). \tag{3.11}$$

Now, for $m \geq 0$, using (3.11), we can compute effectively all the $P_n^m(x)$ by starting with (3.10). For $m < 0$, we can then use (3.9). We list the first few Legendre functions are listed in Table 2. Recalling (3.6), we are in particular interested int the case $x = \cos\theta$, which we also list.

Using the recurrence (3.11) for $P_n^m(\cos\theta)$, and the explicit solutions in Table 2, we now easily find all the spherical harmonics $Y_n^m(\theta,\phi)$ by the formula (3.6). Table 3 shows some of the low-order ones.

Table 2: First few associated Legendre functions as functions of x and of $x = \cos\theta$.

P_n^m	fn. of x	fn. of θ	P_n^m	fn. of x	fn. of θ
$P_0^0(x)$	1	1	$P_1^{-1}(x)$	$-\frac{1}{2}(1-x^2)^{1/2}$	$-\frac{1}{2}\sin\theta$
$P_1^0(x)$	x	$\cos\theta$	$P_1^1(x)$	$(1-x^2)^{1/2}$	$\sin\theta$
$P_2^{-2}(x)$	$\frac{1}{8}(1-x^2)$	$\frac{1}{8}\sin^2\theta$	$P_2^{-1}(x)$	$-\frac{1}{2}x(1-x^2)^{1/2}$	$-\frac{1}{2}\cos\theta\sin\theta$
$P_2^0(x)$	$\frac{3}{2}x^2-\frac{1}{2}$	$\frac{3}{2}\cos^2\theta-\frac{1}{2}$	$P_2^1(x)$	$3x(1-x^2)^{1/2}$	$3\cos\theta\sin\theta$
$P_2^2(x)$	$3(1-x^2)$	$3\sin^2\theta$	$P_3^{-3}(x)$	$-\frac{1}{48}(1-x^2)^{3/2}$	$-\frac{1}{48}\sin^3\theta$
$P_3^{-2}(x)$	$\frac{1}{8}x(1-x^2)$	$\frac{1}{8}\cos\theta\sin^2\theta$	$P_3^{-1}(x)$	$-\frac{1}{8}(5x^2-1)(1-x^2)^{1/2}$	$\frac{1}{8}(5\cos^2\theta-1)\sin\theta$
$P_3^0(x)$	$\frac{1}{2}x(5x^2-3)$	$\frac{1}{2}\cos\theta(5\cos^2\theta-3)$	$P_3^1(x)$	$\frac{3}{2}(5x^2-1)(1-x^2)^{1/2}$	$\frac{3}{2}(5\cos^2\theta-1)\sin\theta$
$P_3^2(x)$	$15x(1-x^2)$	$15\cos\theta\sin^2\theta$	$P_3^3(x)$	$15(1-x^2)^{3/2}$	$15\sin^3\theta$

Table 3: Some low-order spherical harmonics

$Y_0^0(\theta,\phi)=\frac{1}{\sqrt{4\pi}}$	$Y_1^{-1}(\theta,\phi)=\sqrt{\frac{3}{8\pi}}\sin\theta e^{-i\phi}$
$Y_1^0(\theta,\phi)=\sqrt{\frac{3}{4\pi}}\cos\theta$	$Y_1^1(\theta,\phi)=-\sqrt{\frac{3}{8\pi}}\sin\theta e^{i\phi}$
$Y_2^{-2}(\theta,\phi)=\sqrt{\frac{15}{32\pi}}\sin^2\theta e^{-2i\phi}$	$Y_2^{-1}(\theta,\phi)=\sqrt{\frac{15}{8\pi}}\sin\theta\cos\theta e^{-i\phi}$
$Y_2^0(\theta,\phi)=\sqrt{\frac{5}{4\pi}}\left(\frac{3}{2}\cos^2\theta-\frac{1}{2}\right)$	$Y_2^1(\theta,\phi)=-\sqrt{\frac{5}{24\pi}}3\sin\theta\cos\theta e^{i\phi}$
$Y_2^2(\theta,\phi)=\sqrt{\frac{15}{32\pi}}\sin^2\theta e^{i2\phi}$	$Y_3^{-3}(\theta,\phi)=\sqrt{\frac{35}{64\pi}}\sin^3\theta e^{-i3\phi}$
$Y_3^{-2}(\theta,\phi)=\sqrt{\frac{105}{32\pi}}\cos\theta\sin^2\theta e^{-i2\phi}$	$Y_3^{-1}(\theta,\phi)=\sqrt{\frac{21}{64\pi}}(5\cos^2\theta-1)\sin\theta e^{-i\phi}$
$Y_3^0(\theta,\phi)=\sqrt{\frac{7}{16\pi}}\cos\theta(5\cos^2\theta-3)$	$Y_3^1(\theta,\phi)=-\sqrt{\frac{21}{64\pi}}(5\cos^2\theta-1)\sin\theta e^{i\phi}$
$Y_3^2(\theta,\phi)=\sqrt{\frac{105}{32\pi}}\cos\theta\sin^2\theta e^{i2\phi}$	$Y_3^3(\theta,\phi)=-\sqrt{\frac{35}{64\pi}}\sin^3\theta e^{i3\phi}$

3.3 A METHOD TO COMPUTE THE SPHERICAL BASIS FUNCTIONS

We recall te presentation of f_l^+ and the presentation (2.4) of f_n^m. Because $l = n^2+n+m+1$ and $|m| \le n$, we find that l is bounded by $l_{\max}^{(n)} = (n+1)^2$ for n. From $f_l^+(\rho,\theta,\phi) = f_n^m(\rho,\theta,\phi)$, we easily get for the low-order functions the relantionships in Table 4. Using $f_n^m(\rho,\theta,\phi) = j_n(\zeta\rho)Y_n^m(\theta,\phi)$, we then obtain the basis functions f_l^+. Table 5 lists some f_l^+ corresponding to low-order n.

From the above analysis, we find that if j_n and P_n^m are given only for $n = 0, 1$, then rather than directly evaluating the series (3.3) and (3.8), we can quickly find all the basis functions f_l^+ from (3.2) and (3.11). Starting with $j_0(\zeta\rho) = \frac{\sin(\zeta\rho)}{\zeta\rho}$, $j_1(\zeta\rho) = \frac{\sin(\zeta\rho)}{(\zeta\rho)^2} - \frac{\cos(\zeta\rho)}{\zeta\rho}$, $P_0^0(\cos\theta) = 1$, $P_1^{-1}(\cos\theta) = -\frac{1}{2}\sin\theta$, $P_1^0(\cos\theta) = \cos\theta$, $P_1^1(\cos\theta) = \sin\theta$, we can compute all the f_l^+ with $|m| \le n, l = 1, 2, \ldots, l_{\max}^{(n)}$. The numerical computation is outlined in Algorithm 1. In its practical application, the radius ρ, the polar angle θ and the azimuth angle ϕ are computed by Cartesian coordinates (x, y, z), which are as follows:

$$\rho_{i,j} = \sqrt{x_i^2 + y_j^2 + z_0^2},$$

$$\theta_{i,j} = \arccos(\frac{z_0}{\rho_{i,j}}),$$

Table 4: The relationship between f_l^+ and f_n^m

n	f_n^m	f_l^+ $(l_{\max}^{(n)})$
$n=0$	f_0^0	$f_1^+(l_{\max}=(0+1)^2=1)$
$n=1$	f_1^{-1}, f_1^0, f_1^1	f_2^+, f_3^+, f_4^+ $(l_{\max}^{(n)}=(1+1)^2=4)$
$n=2$	$f_2^{-2}, f_2^{-1}, f_2^0, f_2^1, f_2^2$	$f_5^+, f_6^+, f_7^+, f_8^+, f_9^+$ $(l_{\max}^{(n)}=(2+1)^2=9)$
$n=3$	$f_3^{-3}, f_3^{-2}, f_3^{-1}, f_3^0, f_3^1, f_3^2, f_3^3$	$f_{10}^+, f_{11}^+, f_{12}^+, f_{13}^+, f_{14}^+, f_{15}^+, f_{16}^+$ $(l_{\max}^{(n)}=(3+1)^2=16)$
$\vdots$	$\vdots$	$\vdots$

Table 5: The the basis functions f_l^+ corresponding to order n

n	f_l^+
$n=0$	$f_1^+ = j_0(\zeta\rho)Y_0^0(\theta,\phi)$
$n=1$	$f_2^+ = j_1(\zeta\rho)Y_1^{-1}(\theta,\phi)$, $f_3^+ = j_1(\zeta\rho)Y_1^0(\theta,\phi)$, $f_4^+ = j_1(\zeta\rho)Y_1^1(\theta,\phi)$
$n=2$	$f_5^+ = j_2(\zeta\rho)Y_2^{-2}(\theta,\phi)$, $f_6^+ = j_2(\zeta\rho)Y_2^{-1}(\theta,\phi)$, $f_7^+ = j_2(\zeta\rho)Y_2^0(\theta,\phi)$, $f_8^+ = j_2(\zeta\rho)Y_2^1(\theta,\phi)$, $f_9^+ = j_2(\zeta\rho)Y_2^2(\theta,\phi)$
$n=3$	$f_{10}^+ = j_3(\zeta\rho)Y_3^{-3}(\theta,\phi)$, $f_{11}^+ = j_3(\zeta\rho)Y_3^{-2}(\theta,\phi)$, $f_{12}^+ = j_3(\zeta\rho)Y_3^{-1}(\theta,\phi)$, $f_{13}^+ = j_3(\zeta\rho)Y_3^0(\theta,\phi)$, $f_{14}^+ = j_3(\zeta\rho)Y_3^1(\theta,\phi)$, $f_{15}^+ = j_3(\zeta\rho)Y_3^2(\theta,\phi)$, $f_{16}^+ = j_3(\zeta\rho)Y_3^3(\theta,\phi)$
$\vdots$	$\vdots$

$$\phi_{i,j} = \arctan(\frac{y_j}{x_i}),$$

where z_0 is fixed, x_i, y_j are discrete results of x, y, $i,j = 1,2,\ldots,N$. Hence

$$\left(f_l^+\right)_{i,j} = j_n(\zeta\rho_{i,j})Y_n^m(\theta_{i,j},\phi_{i,j}) \quad \text{with} \quad l = n^2+n+m+1, |m| \le n.$$

The B_1^+ field approximation (2.1) can be expressed in matrix-vector form $B_1^+ \approx FA$, where for $\ell = l_{\max}^{(\tilde{n})}$ we have

$$F = \begin{pmatrix} (f_1^+)_{1,1} & (f_2^+)_{1,1} & \cdots & (f_\ell^+)_{1,1} \\ (f_1^+)_{2,1} & (f_2^+)_{2,1} & \cdots & (f_\ell^+)_{2,1} \\ \vdots & \vdots & \ddots & \vdots \\ (f_1^+)_{N,1} & (f_2^+)_{N,1} & \cdots & (f_\ell^+)_{N,1} \\ (f_1^+)_{1,2} & (f_2^+)_{1,2} & \cdots & (f_\ell^+)_{1,2} \\ (f_1^+)_{2,2} & (f_2^+)_{2,2} & \cdots & (f_\ell^+)_{2,2} \\ \vdots & \vdots & \ddots & \vdots \\ (f_1^+)_{N,2} & (f_2^+)_{N,2} & \cdots & (f_\ell^+)_{N,2} \\ \vdots & \vdots & \ddots & \vdots \\ (f_1^+)_{1,N} & (f_2^+)_{1,N} & \cdots & (f_\ell^+)_{1,N} \\ (f_1^+)_{2,N} & (f_2^+)_{2,N} & \cdots & (f_\ell^+)_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ (f_1^+)_{N,N} & (f_2^+)_{N,N} & \cdots & (f_\ell^+)_{N,N} \end{pmatrix}, \quad \text{and} \quad A = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_{l_{\max}^{(\tilde{n})}} \end{pmatrix}.$$

Algorithm 1 The computation for the basis functions f_l^+

Given $\omega, \mu, \sigma, \varepsilon, \rho, \theta, \phi$ and $\tilde{n}$.
Calculate $\zeta = \sqrt{\varepsilon\mu\omega^2 - i\sigma\omega\mu}$.
Initialize $j_0(\zeta\rho) = \frac{\sin(\zeta\rho)}{\zeta\rho}$, $j_1(\zeta\rho) = \frac{\sin(\zeta\rho)}{(\zeta\rho)^2} - \frac{\cos(\zeta\rho)}{\zeta\rho}$,
$P_0^0(\cos\theta) = 1$, $P_1^{-1}(\cos\theta) = -\frac{1}{2}\sin\theta$,
$P_1^0(\cos\theta) = \cos\theta$, $P_1^1(\cos\theta) = \sin\theta$
$Y_0^0(\theta,\phi) = \frac{1}{\sqrt{4\pi}}$, $Y_1^{-1}(\theta,\phi) = \sqrt{\frac{3}{8\pi}}\sin\theta e^{-i\phi}$,
$Y_1^0(\theta,\phi) = \sqrt{\frac{3}{4\pi}}\cos\theta$, $Y_1^1(\theta,\phi) = -\sqrt{\frac{3}{8\pi}}\sin\theta e^{i\phi}$.
Compute $f_1^+ = j_0(\zeta\rho)Y_0^0(\theta,\phi)$, $f_2^+ = j_1(\zeta\rho)Y_1^{-1}(\theta,\phi)$,
$f_3^+ = j_1(\zeta\rho)Y_1^0(\theta,\phi)$, $f_4^+ = j_1(\zeta\rho)Y_1^1(\theta,\phi)$.
For $n = 2 : \tilde{n}$
- **calculate** $j_n(\zeta\rho)$ and $P_n^m(\cos\theta)$ using (3.2) and (3.11), respectively.
- **calculate** $Y_n^m(\theta,\phi)$ with (3.6).
- **calculate** $f_l^+ = j_n(\zeta\rho)Y_n^m(\theta,\phi)$ with $|m| \le n$ and $l = n^2 + n + m + 1$.

End
Return $\{f_l^+ \mid l = 1, 2, \ldots, l_{\max}^{(\tilde{n})}\}$.

4 THE NEW REGULARIZATION MODEL AND ITS NUMERICAL REALIZATION

We now present in detail our spherical function based regularization model for p-MRI reconstruction, as well as a method for its numerical realization.

4.1 REGULARIZATION BY SPARSE PRESENTATION IN SPHERICAL BASIS

Replacing the coil sensitivities c_j and their regularization R_c in the model (1.5) by the spherical functions representation $\sum_{l=1}^{l_{\max}^{(\tilde{n})}} a_l^{(j)} f_l^+, j = 1, 2, \ldots, J$, and $R_a(a) = \alpha \sum_{j=1}^{J} \sum_{l=1}^{l_{\max}^{(\tilde{n})}} |a_l^{(j)}|$, respectively, we obtain the model (1.7), that is

$$\min_{v=(u;a)} \frac{1}{2} \sum_{j=1}^{J} \left\| P\mathcal{F}(G(v))_j - g_j \right\|_2^2 + \alpha_0 R_u(u) + \alpha R_a(a), \tag{4.1}$$

where

$$G(v) = \left(u \sum_{l=1}^{l_{\max}^{(\tilde{n})}} a_l^{(1)} f_l^+, u \sum_{l=1}^{l_{\max}^{(\tilde{n})}} a_l^{(2)} f_l^+, \ldots, u \sum_{l=1}^{l_{\max}^{(\tilde{n})}} a_l^{(J)} f_l^+ \right)^T, \quad R_a(a) = \sum_{j=1}^{J} \sum_{l=1}^{l_{\max}^{(\tilde{n})}} \left| a_l^{(j)} \right|, \tag{4.2}$$

and

$$a = \left(a_1^{(1)}, a_2^{(1)}, \ldots, a_{l_{\max}^{(\tilde{n})}}^{(1)}, a_1^{(2)}, a_2^{(2)}, \ldots, a_{l_{\max}^{(\tilde{n})}}^{(2)}, \ldots, a_1^{(J)}, a_2^{(J)}, \ldots, a_{l_{\max}^{(\tilde{n})}}^{(J)} \right)^T \in \mathbb{C}^{J l_{\max}^{(\tilde{n})}}. \tag{4.3}$$

For the image u, we use the same isotropic total variation regularization $R_u(u) = \mathrm{TV}_I(u)$ as in the model (1.5) from [7], while our R_a yields a different regularization R_c for the coil sensitivities.

The complex numbers $a_l^{(j)}$ are the coefficients corresponding to the spherical function representation. The spherical basis functions are smooth enough to mainly encode low-frequency information for low l, so should not pick up important image features. Apart from this, we will not need to store the coil sensitives, which are themselves relatively high-dimensional images, or differentiate and perform their updates in a numerical optimization algorithm, as would be the case with the mode (1.7) in [7]. Instead we work with the relatively low-dimensional a. Thus the updated model can be expected to be effective for parallel MRI reconstruction. We will validate this with the numerical experiments in Section 5, but now we need to construct the optimization algorithm to solve the model (4.1), that is (1.7).

4.2 THE ADMM FOR CONVEX PROBLEMS

We now start building a method for solving variational problems of type (1.7), i.e., (4.1). Note that due to the structure of G, these problems are non-convex. We therefore follow the non-linear ADMM approach of [7], itself motivated by the non-linear primal–dual hybrid gradient method (modified; PDHGM) of [14, 54]. Here we simplify the derivations from [7] to our specific problem form, and squared Hilbert space distances in place of the general Bregman distances employed in [7].

To motivate the algorithm that we will use, we start by considering convex problems

$$\inf_{v\in V}\{E(v)+F(Bv)\}, \tag{4.4}$$

where $E : V \longrightarrow]-\infty,+\infty]$ and $F : H \longrightarrow]-\infty,+\infty]$ are two proper lower semi-continuous convex functions, B is a linear operator from V into H, and V and H are the real Hilbert spaces equipped with the inner products $\langle\cdot,\cdot\rangle_V$ and $\langle\cdot,\cdot\rangle_H$. The augmented dual problem for (4.4) is

$$\sup_{\mu}\varphi_\delta(\mu) := \inf_{v\in V}\left\{E(v)+\inf_{q\in H}\left(F(q)+\langle\mu,Bv-q\rangle_H+\frac{\delta}{2}\parallel Bv-q\parallel_H^2\right)\right\}. \tag{4.5}$$

where, $\delta > 0$, and $\varphi_\delta : H \longrightarrow [-\infty,+\infty[$. We can solve (4.4) and the dual (4.5) by finding on $V\times H\times H$ a saddle point of the augmented Lagrangian function defined by

$$\mathscr{L}_\delta(v,q,\mu) = F(q)+E(v)+\langle\mu,Bv-q\rangle_H+\frac{\delta}{2}\parallel Bv-q\parallel_H^2 . \tag{4.6}$$

Theorem 4.1. *If $\{u,p,\lambda\}$ is a saddle-point of $\mathscr{L}_\delta(v,q,\mu)$ on $V\times H\times H$ for $\delta>0$, then u is a solution of (4.4), and we have $p = Bu$.*

Proof. The proof is a direct consequence of Theorem 2.1 in Chapter III [21]. □

In view of Theorem 4.1, in order to calculate the saddle-points of $\mathscr{L}_\delta(v,q,\mu)$ on $V\times H\times H$, we employ an algorithm of the Uzawa type: given $\lambda^0\in H$, determine $\{u^{k+1},p^{k+1}\}$ from

$$\min_{\{v,q\}\in V\times H}\mathscr{L}_\delta(v,q,\lambda^k), \tag{4.7}$$

and then update

$$\lambda^{k+1} = \lambda^k+\delta(Bu^{k+1}-p^{k+1}). \tag{4.8}$$

For more details we refer to [2]. The direct solution to (4.7) is due to the coupling of v and q. For this reason, we are led to the Alternating Direction Method of Multipliers (ADMM) algorithm introduced in [2] to solve (4.7). This algorithm approximates the pair (u^{k+1}, p^{k+1}) for (4.7) via decoupled minimization over v and q as

$$u^{k+1} \in \operatorname*{argmin}_{v \in V} \mathscr{L}_\delta(v, p^k, \lambda^k), \qquad p^{k+1} \in \operatorname*{argmin}_{q \in H} \mathscr{L}_\delta(u^{k+1}, q, \lambda^k). \tag{4.9}$$

The ADMM algorithm for (4.4), described by (4.6), (4.8), and (4.9), can thus be summarized

$$u^{k+1} \in \operatorname*{argmin}_{v \in V} F(p^k) + E(v) + \langle \lambda^k, Bv - p^k \rangle_H + \frac{\delta}{2} \parallel Bv - p^k \parallel_H^2, \tag{4.10a}$$

$$p^{k+1} \in \operatorname*{argmin}_{q \in H} F(q) + E(u^{k+1}) + \langle \lambda^k, Bu^k - q \rangle_H + \frac{\delta}{2} \parallel Bu^k - q \parallel_H^2, \tag{4.10b}$$

$$\lambda^{k+1} = \lambda^k + \delta(Bu^{k+1} - p^{k+1}).$$

4.3 PROXIMAL MINIMIZATION

When V and H are $\mathbb{R}^n$, the sub-problems (4.10a) and (4.10b) can be solved by the proximal minimization algorithm that we now describe. Let us consider the convex problem

$$\min_{x \in \mathbb{R}^n} W(x), \tag{4.11}$$

where $W : \mathbb{R}^n \to]-\infty, +\infty]$ is a proper, lower semi-continuous convex function. The Moreau–Yosida envelope of $W(x)$ is defined as

$$W_\rho(x) = \inf_{y \in \mathbb{R}^n} \left\{ W(y) + \frac{1}{2\rho} \|x - y\|^2 \right\}.$$

As proved in [42], $W_\rho(x)$ is convex and differentiable, and has the same set of minimizers, and the same optimal value, as W. This leads to the proximal minimization algorithm proposed by Martinet in [40]. Namely, we solve (4.11) by iterating

$$x^{k+1} = \operatorname*{argmin}_{x \in \mathbb{R}^n} \left\{ W(x) + \frac{1}{2\rho_{k+1}} \|x - x^k\|^2 \right\}, \tag{4.12}$$

where the initial point $x^0 \in \mathbb{R}^n$, $\{\rho_k\}_{k=1}^\infty$ is a sequence of positive numbers. The convergence of this algorithm has been proved by Rockafellar in [47, 48]. For more discussion on proximal methods, see [15, 38, 55].

4.4 PRECONDITIONED PROXIMAL MINIMIZATION

Following the ideas given in [60], we can improve the performance of the proximal minimization method (4.12) by preconditioning. Specifically, we pick some positive definite symmetric matrix Q, and replace (4.12) by

$$x^{k+1} = \operatorname*{argmin}_{x \in \mathbb{R}^n} \left\{ W(x) + \rho_{k+1}^{-1} \left\|x - x^k\right\|_Q^2 \right\}.$$

Here we define $\| x - x^k \|_Q := \sqrt{\langle Q(x - x^k), x - x^k \rangle}$. Thus, picking Q_v^k and Q_q^k positive semi-definite, and incorporating the corresponding Moreau–Yosida regularization into (4.10), we obtain the preconditioned ADMM (cf. [20, 61])

$$
\begin{aligned}
u^{k+1} &= \operatorname*{argmin}_{v \in \mathbb{R}^n} \; F(p^k) + E(v) + \langle \lambda^k, Bv - p^k \rangle + \frac{\delta}{2} \left\| Bv - p^k \right\|^2 + \rho_{k+1}^{-1} \left\| v - u^k \right\|_{Q_v^k}^2, \\
p^{k+1} &= \operatorname*{argmin}_{q \in \mathbb{R}^n} \; F(q) + E(u^{k+1}) + \langle \lambda^k, Bu^{k+1} - q \rangle + \frac{\delta}{2} \left\| Bu^{k+1} - q \right\|^2 + \rho_{k+1}^{-1} \left\| q - p^k \right\|_{Q_q^k}^2, \\
\lambda^{k+1} &= \lambda^k + \delta(Bu^{k+1} - p^{k+1}).
\end{aligned}
$$

4.5 A COMPUTATIONAL METHOD FOR THE PROPOSED MODEL

In order to cast the model (1.7) in the preconditioned ADMM framework, let us first study the augmented Lagrangian for (1.7). For $v = (u; a)^T$, we define

$$
B(v) := \left(G(v), \nabla u, a \right),
$$

where G and a are as in (4.2) and (4.3), respectively. We represent the image as a vector $u \in \mathbb{R}^{N^2}$, and write $\nabla u = (\nabla_1 u; \nabla_2 u) \in \mathbb{R}^{2N^2}$. with $E = 0$ and

$$
F(B(v)) = \frac{1}{2} \sum_{j=1}^{J} \left\| P\mathcal{F}(G(v))_j - g_j \right\|_2^2 + \alpha_0 R_u(u) + \alpha R_a(a),
$$

the problem (1.7) has the form (4.4) except for B being nonlinear operator. It follows from the above analysis and the preceding sections that an augmented Lagrangian naturally associated with the problem (1.7) is given by

$$
\mathscr{L}_\delta(v, q, \mu) = F(q) + (\mu, B(v) - q) + \frac{\delta}{2} \left\| B(v) - q \right\|^2.
$$

By analogue, following [7], we extend the preconditioned ADMM (4.13) to non-linear B as

$$
u^{k+1} = \operatorname*{argmin}_{v} \; F(p^k) + \langle \lambda^k, B(v) - p^k \rangle + \frac{\delta}{2} \left\| B(v) - p^k \right\|^2 + \rho_{k+1}^{-1} \left\| v - u^k \right\|_{Q_v^k}^2, \tag{4.14a}
$$

$$
p^{k+1} = \operatorname*{argmin}_{q} \; F(q) + \langle \lambda^k, B(u^{k+1}) - q \rangle + \frac{\delta}{2} \left\| B(u^{k+1}) - q \right\|^2 + \rho_{k+1}^{-1} \left\| q - p^k \right\|_{Q_q^k}^2, \tag{4.14b}
$$

$$
\lambda^{k+1} = \lambda^k + \delta(B(u^{k+1}) - p^{k+1}). \tag{4.14c}
$$

For our specific problem, the minimizations are over $v \in \mathbb{R}^{N^2 + J * l_{\max}^{(\tilde{n})}}$, and $q \in \mathbb{R}^{J * N^2 + 2 * N^2 + J * l_{\max}^{(\tilde{n})}}$.

To make the proximal minimizations (4.14a) and (4.14b) easier, we linearise the operator B. Let $\tilde{F}_1(v) = B(v) - p^k$ and $\tilde{F}_2(q) = B(u^{k+1}) - q$. Since these functions are smooth,

$$
\tilde{F}_1(v) \approx \tilde{F}_1(u^k) + \tilde{F}_1{}'(u^k)(v - u^k), \quad \text{and} \quad \tilde{F}_2(q) \approx \tilde{F}_2(p^k) + \tilde{F}_2{}'(p^k)(v - p^k), \tag{4.15}
$$

Algorithm 2 Linearised preconditioned ADMM for (1.7)

Initialization u^0, p^0, λ^0 and δ.
Set $\bar{\lambda}^0 = \mu^0$.
while "stopping criterion is not satisfied" **do**
  $J_v^k = \tilde{F}_1'(u^k)$
  Choose τ_v^k such that $\tau_v^k \delta < \frac{1}{\|J_v^k\|^2}$
  Compute u^{k+1} by (4.17)
  Compute J_q^k by $J_q^k = \tilde{F}_2'(p^k)$
  Choose τ_q^k such that $\tau_q^k \delta < \frac{1}{\|J_q^k\|^2}$
  Compute p^{k+1} by (4.18)
  Compute λ^{k+1} by (4.14c)
  Compute $\bar{\lambda}^{k+1} = 2\lambda^{k+1} - \lambda^k$
end while
Return u^k, p^k, λ^k and $\bar{\lambda}^k$

where $\tilde{F}_1'(u^k)$ and $\tilde{F}_2'(p^k)$ are the Fréchet derivative of $\tilde{F}_1$ at u^k and $\tilde{F}_2$ at p^k. For the sake of clarity, we set $J_v^k := \tilde{F}_1'(u^k)$, $J_q^k := \tilde{F}_2'(p^k)$, $r_v^k := \tilde{F}_1'(u^k)u^k - \tilde{F}_1(u^k)$ and $r_q^k := \tilde{F}_2'(p^k)p^k - \tilde{F}_2(p^k)$. Using (4.15), (4.14a), and (4.14b) are replaced by

$$u^{k+1} = \operatorname*{argmin}_v \langle \lambda^k, J_q^k v\rangle + \frac{\delta}{2}\left\|J_q^k v - r_v^k\right\|^2 + \rho_{k+1}^{-1}\left\|v - u^k\right\|_{Q_v^k}^2, \tag{4.16a}$$

$$p^{k+1} = \operatorname*{argmin}_q \; F(q) + \langle \lambda^k, J_q^k v\rangle + \frac{\delta}{2}\left\|J_q^k v - r_v^k\right\|^2 + \rho_{k+1}^{-1}\left\|q - p^k\right\|_{Q_q^k}^2, \tag{4.16b}$$

where still $v \in \mathbb{R}^{N^2 + J*l_{\max}^{(\tilde{n})}}$, and $q \in \mathbb{R}^{J*N^2+2*N^2+J*l_{\max}^{(\tilde{n})}}$.

In order to simplify the linearisation (4.16a) and (4.16b), we will seek Q_v^k and Q_q^k by J_v^k and J_q^k. For $\tau_v^k\delta < \frac{1}{\|J_v^k\|^2}$ and $\tau_q^k\delta < \frac{1}{\|J_q^k\|^2}$, we specifically let $Q_v^k := \tau_v^k I - \delta J_v^{k*} J_v^k$ and $Q_q^k := \tau_q^k I - \delta J_q^{k*} J_q^k$. We also set $\bar{\lambda}^k := 2\lambda^k - \lambda^{k-1}$ and $\rho_{k+1} = 2$. It follows easily from (4.16a) and (4.16b)

$$u^{k+1} = u^k - \tau_v^k J_v^{k*} \bar{\lambda}^k, \tag{4.17}$$

$$p^{k+1} = \left(I + \tau_q^k \partial F\right)^{-1}\left(p^k - \tau_q^k J_q^{k*}\left(\lambda^k + \delta\left(B(u^{k+1}) - p^k\right)\right)\right). \tag{4.18}$$

Finally, (4.17), (4.18) and (4.14c) yields Algorithm 2 for the solution of (1.7). Its convergence is studied in [7] based on the results of [54].

5 NUMERICAL EXPERIMENTS

We now study the reconstruction performance of (1.7) in comparison to the model (1.5) from [7].

5.1 TECHNICAL DETAILS

In the numerical simulations, we introduce regularization parameters α_j into the fidelity term $\frac{1}{2}\sum_{j=1}^{J}\left\|P\mathcal{F}(G(v))_j - g_j\right\|_2^2$ in the model (1.7). With this, the objective functional becomes

$$\begin{aligned} F(Bv) &= \frac{1}{2}\sum_{j=1}^{J}\alpha_j\left\|P\mathcal{F}(G(v))_j - g_j\right\|_2^2 + \alpha_0 R_u(u) + \alpha R_a(a) \\ &= \frac{1}{2}\sum_{j=1}^{J}\alpha_j\left\|P\mathcal{F}(G(v))_j - g_j\right\|_2^2 + \alpha_0 TV_I(u) + \alpha\sum_{j=1}^{J}\sum_{l=1}^{l_{\max}^{(\tilde{n})}}\left|a_l^{(j)}\right|. \end{aligned}$$

We decompose $F(Bv)$ into $F_j(v) = \frac{1}{2}\alpha_j\left\|P\mathcal{F}(G(v))_j - g_j\right\|_2^2$ for $j = 1, 2, \ldots, J$, as well as $F_{J+1}(u) = \alpha_0 TV_I(u)$, and $F_{J+2}(a) = \alpha\sum_{j=1}^{J}\sum_{l=1}^{l_{\max}^{(\tilde{n})}}\left|a_l^{(j)}\right|$. That is $F = \sum_{j=1}^{J}F_j + F_{J+1} + F_{J+2} = \sum_{j=1}^{J+2}F_j$. We compute the corresponding resolvents explicitly as

$$\begin{aligned} \left(I + \tau_q^k \partial F_j\right)^{-1}(x) &= \mathcal{F}^{-1}\left(\frac{\mathcal{F}x_j + \alpha_j\tau_q^k P^T g_j}{1 + \alpha_j\tau_q^k \operatorname{diag}(P^T P)}\right),\ j = 1, 2, \ldots, J, \\ \left(I + \tau_q^k \partial F_{J+1}\right)^{-1}(x) &= \frac{x_{J+1}}{TV_I\left(x_{J+1}\right)}\max\left(TV_I\left(x_{J+1}\right) - \alpha_0\tau_q^k, 0\right), \\ \left(I + \tau_q^k \partial F_{J+1}\right)^{-1}(x) &= \frac{(x_{J+2})_i}{\left|(x_{J+2})_i\right|}\max\left(\left|(x_{J+2})_i\right| - \alpha\tau_q^k, 0\right),\ i = 1, 2, \ldots, J * l_{\max}^{(\tilde{n})}. \end{aligned}$$

5.2 EXPERIMENTAL SETUP

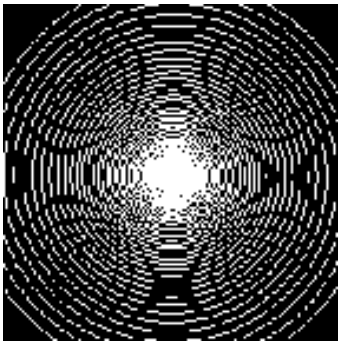
Our numerical experiments are based on the synthetic brain phantom from [3, 8], depicted in Section 5.2 and of dimension 190×190. It contains several tissues, such as cerebrospinal fluid (CSF), gray matter (GM), white matter (WM) and cortical bone. In the numerical simulations, we set the number of coils $J = 8$. For the generation of k-space measurement data g_j, $j = 1, 2, \ldots, J$, we use the approach of [7]. We generate 8 coil sensitivity maps, based on a measurement of a water bottle with an 8-channel head coil array. These measurements are in Figure 6. We then multiply the brain phantom with each of these coil sensitivity maps separately, and convert the result to k-space data with the Fourier transform. Then we apply the 25% subsampling mask shown in Section 5.2. Finally, we add Gaussian noise with standard deviation $\tilde{\sigma}$ to the sub-sampled data.

We also demonstrate the robustness of the proposed approach in the p-MRI reconstruction by perturbing the coil sensitivity maps obtained from the water bottle. This is done by adding the 1st spherical basis function multiplied by factor $\gamma = 4$ to the water bottle measurements. The resulting maps are shown in Figure 12.

In numerical experiments, for the number of spherical basis functions "levels", we choose either $\tilde{n} = 2$ or $\tilde{n} = 5$. So the number of spherical basis functions is either $l_{\max}^{(\tilde{n})} = 9$ or $l_{\max}^{(\tilde{n})} = 36$. In the Cartesian coordinate system, we set $z_0 = 0.5$, and x, y in MATLAB are discretised by

$$\frac{2 * step * (1:190)}{190} - 10,$$

(a) Brain phantom



(b) k-space sampling mask

Figure 1: (a) shows the brain phantom described in Section 5.2. (b) shows the spiral-shaped 25% k-space sub-sampling mask.

where $step = 10$. We plot in Figure 2 the first 36 spherical basis functions corresponding to $\tilde{n} = 5$. For $\tilde{n} = 2$ only the 9 first are used.

5.3 Quality measures and parameter selection

All algorithms have been implemented in MATLAB, and the test hardware is an Intel Core i7-6700 HQ CPU 2.60GHz with 8GB RAM. We evaluate the results of the proposed approach in terms of the peak signal-to-noise (PSNR) that is available in the image processing toolbox in MATLAB and the Structural SIMilarity (SSIM) given in [57]. In the computation of the spherical basis function f_l^+, we use the Larmour frequency $\omega = 42.58$. The conductivity σ and the dielectric permittivity ε are the optimal (σ, ε) for the heterogeneous model in [50] with $\sigma = 0.6, \varepsilon = 50$. The magnetic permeability for water is $\mu = 1.2566 \times 10^{-6}$.

5.4 Numerical reconstructions and comparison between (1.7) and (1.5)

We perform experiments with the noise level $\tilde{\sigma} = 0.05$. We initialize the Algorithm 2 with $u^0 = 0 \in \mathbb{R}^{N^2}$, $a^0 = (1, \ldots, 1) \in \mathbb{R}^{Jl_{\max}^{(n)}}$, and take as step lengths $\tau_v^k = 1/8$, $\tau_q^k = 23$, and $\delta = 1/24$ and the remaining v^0, λ^0, $\bar{\lambda}^0$ are all also initialized to zero in the two algorithms. For numerical reconstruction corresponding to (1.5), we use the codes from [7], available from [6].

We take as regularization parameters $\alpha_j = 0.4018$, $(j = 1, \ldots, J)$, $\alpha_0 = 0.0062$ and $\alpha = 0.2149$. We perform a fixed number of iterations of Algorithm 2. With $\tilde{n} = 2$, i.e., $l_{\max} = 9$, stopping after 1000, 1200, and 1500 iterations, the reconstruction results for the model (1.7) are shown in Figure 3. We also perform the numerical simulations with $\tilde{n} = 5$, and the same number of iterations 1000, 1200, and 1500. The reconstruction results for the model (1.7) are shown in Figure 4, and for the model (1.5) in Figure 5. In Table 6 we report the PNSR and SSIM [57] values. From these results we can observe that the reconstruction quality of the model (1.7) is much better than the model (1.5) when iterations are 1000, 1200, and 1500. The reconstruction results can be further improved due to the regularization parameters α not being optimally chosen; for a truly fair comparison of the potential of the two models distinct, parameter learning strategies

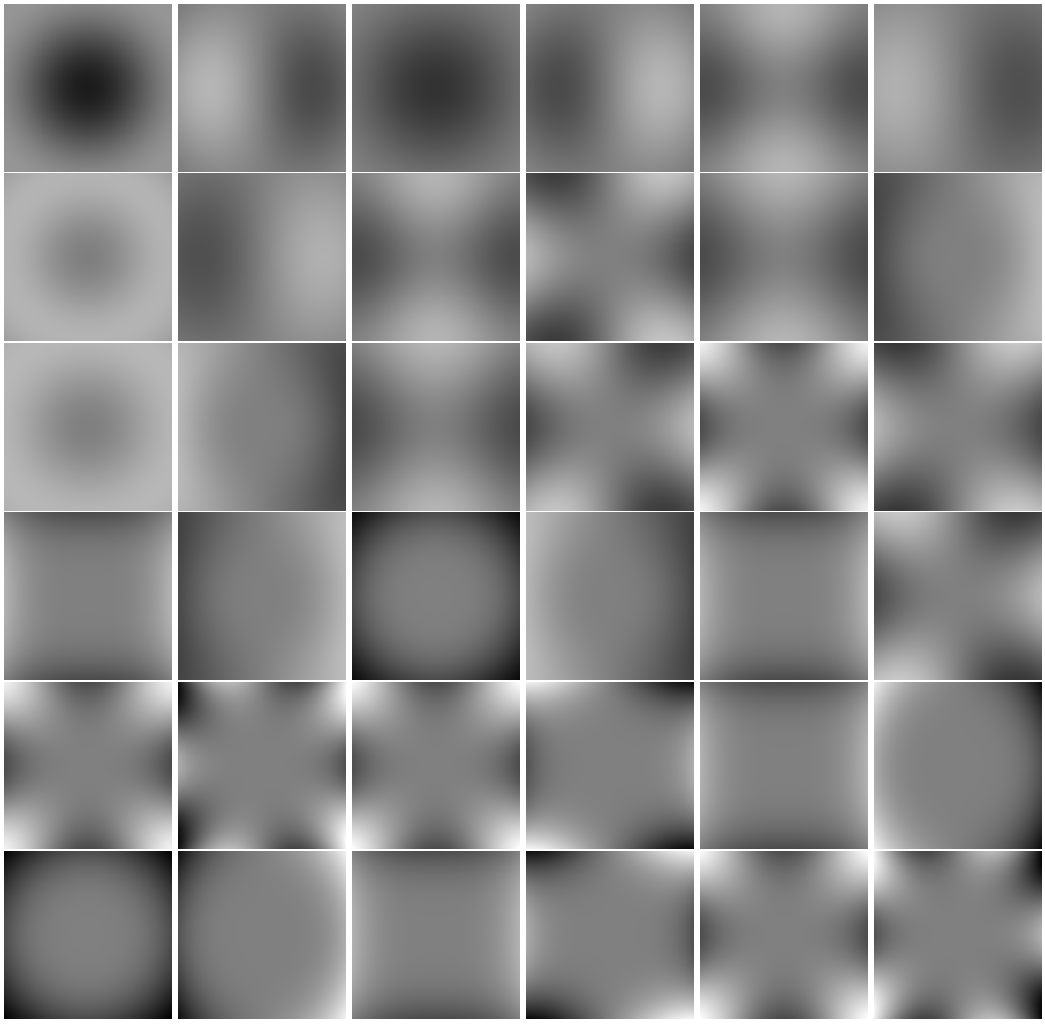
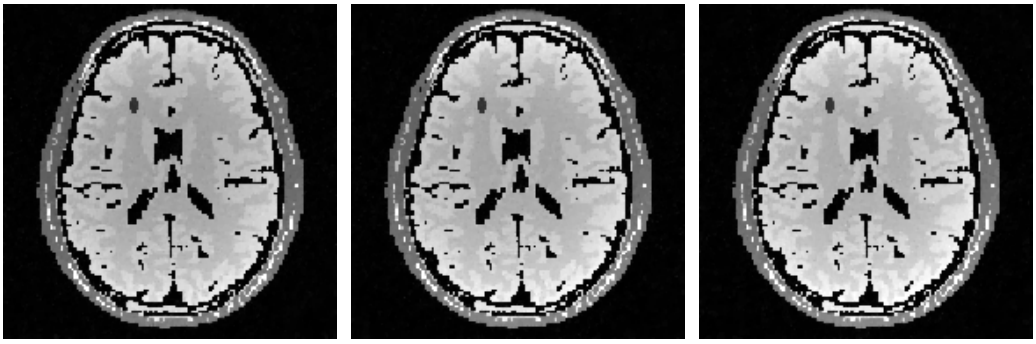


Figure 2: The 36 first spherical basis functions corresponding to $\tilde{n} = 5$. For $\tilde{n} = 2$ only the 9 first are used.

should be used [16, 17]. What we can with reasonable confidence say based on our experiments here is that the non-linear ADMM converges faster for the model (1.7) than for (1.5). This is important in practical applications.

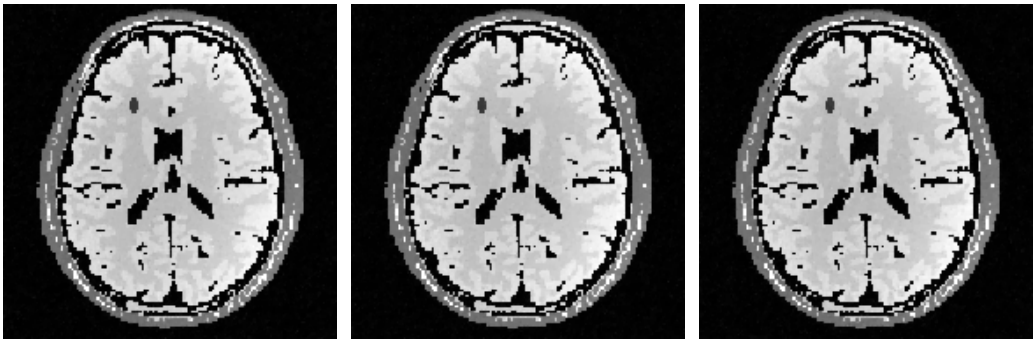
We also report the absolute values of the coefficients $a_l^{(j)}$ in Table 7 when the stopping number of iterations is 1500 for $\tilde{n} = 2$. Similarly, Table 8 shows the absolute values for the resulted coefficients $a_l^{(j)}$ for $\tilde{n} = 5$ when we stop at 1500 iterations. While for $\tilde{n} = 2$ the last rows of the coefficient pyramid for each coil still have high coefficient values, for $\tilde{n} = 5$ the coefficients on the last row have decayed to below 1% of the main coefficient on the first row; often 0.1% or less. This supports our starting intuition that a sparse approximation of the coil sensitivities with relatively few coefficients is sufficient for a high-quality reconstruction.

Using the discovered coefficients $a_l^{(j)}$, and the known spherical basis functions, we can recon-



(a) 1200 iterations (b) 1500 iterations (c) 1800 iterations

Figure 3: Reconstructed brain slice using (1.7) and $\tilde{n} = 2$.

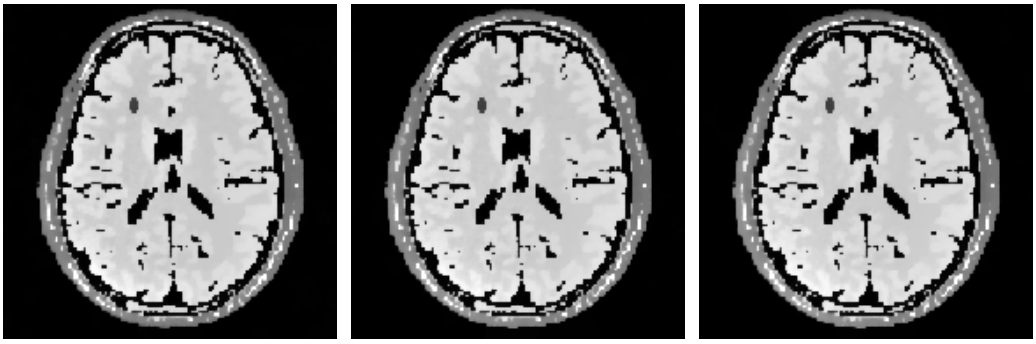


(a) 1200 iterations (b) 1500 iterations (c) 1800 iterations

Figure 4: Reconstructed brain slice using (1.7) and $\tilde{n} = 5$.

struct the approximation of the coil sensitivities c_j. These are in Figure 7 and Figure 8 for $\tilde{n} = 2$ and $\tilde{n} = 5$ with 1500 iterations. In order to do the further comparison between (1.7) and (1.5), we also give the approximation of c_j with 1500 iterations for (1.5) in Figure 9. The PSNR and SSIM values for reconstruction of coils using (1.7) and (1.5) for 1500 iterations are reported in Table 9. Visually, the coil sensitivities constructed with our model (1.5) are significantly smoother than those constructed with the model (1.7), and indeed appear to very well approximate the "true" coil sensitivities in Figure 6.

To test robustness, we show in Figures 10 and 11 for $\tilde{n} = 2$ and $\tilde{n} = 5$, respectively, the reconstructions results for the alternative coil sensitivity maps in Figure 12. The number of iterations is 1500. The PSNR and SSIM values are reported in Table 10. Comparing to Figures 3 and 4 and Table 6, we can see that the results remain stable under this perturbation of coil sensitivities, being virtually identical. By contrast, the reconstructed coil sensitivities in Figures 13 and 14 have changed, corresponding to the change in true coil sensitivities.



(a) 1200 iterations (b) 1500 iterations (c) 1800 iterations

Figure 5: Reconstructed brain slice using (1.5).

Table 6: Reconstruction quality comparison between (1.7) with $\tilde{n} = 2, 5$ and (1.5).

Method	stopping itr. k	PSNR(dB)	SSIM
using (1.7) with $\tilde{n} = 2$	1200	25.2750	0.9996
	1500	25.6878	0.9997
	1800	25.1069	0.9996
using (1.7) with $\tilde{n} = 5$	1200	26.0725	0.9997
	1500	25.5883	0.9997
	1800	25.8730	0.9997
using (1.5)	1200	24.7173	0.9996
	1500	24.1524	0.9996
	1800	23.6702	0.9995

6 CONCLUSIONS

In this paper, we have established a new model for parallel MRI reconstruction based on sparse regularization of coil sensitivities in spherical basis function bases. We have developed efficient recurrence formulas for the computation of these functions. We have then applied the nonlinear ADMM from [7] to numerically solve our model (1.7). By numerical reconstructions and comparison between (1.7) and (1.5), we think that the reconstruction quality for proposed model (1.7) is better than the model (1.5). In additional, the reconstruction for our model (1.7) for the alternative coils sensitivity maps is very robust. That has an important significance in practical applications. In the future, we will study the optimal choice among the regularization parameters α_j, α_0, and α to improve reconstruction quality furthermore via parameter learning strategies in [16, 17].

ACKNOWLEDGMENTS

Y. Zhu was supported by the National Natural Science Foundation of China No. 11571325, Science Research Project of CUC No. 3132016XNL1612. Towards the end of this research, T. Valkonen

Table 7: The absolute values of coefficients of f_l^+ with $\tilde{n} = 2$ for 1500 iterations. j is the ordinal number of coils.

j	f_1^+ f_2^+, f_3^+, f_4^+ $f_5^+, f_6^+, f_7^+, f_8^+, f_9^+$	j	f_1^+ f_2^+, f_3^+, f_4^+ $f_5^+, f_6^+, f_7^+, f_8^+, f_9^+$
1	5.4801 0.9382, 0.0092, 4.7704 1.0213, 0.0075, 5.3347, 0.0034, 3.8693	5	3.4487 0.2859, 0.0096, 3.6825 1.1578, 0.0075, 3.1660, 0.0059, 3.2907
2	4.8840 0.3632, 0.0059, 4.7055 1.2463, 0.0085, 4.6682, 0.0083, 4.3803	6	4.3978 0.1909, 0.0085, 3.4326 0.6340, 0.0060, 3.7131, 0.0048, 3.2575
3	3.4124 0.2777, 0.0058, 4.3643 1.7627, 0.0072, 2.2884, 0.0066, 4.1581	7	3.7714 0.2415, 0.0016, 3.0790 1.4061, 0.0070, 4.1169, 0.0045, 3.8595
4	2.6890 0.1031, 0.0094, 4.2809 1.4676, 0.0065, 2.5603, 0.0058, 3.7494	8	3.3760 0.2552, 0.0065, 3.2423 1.6636, 0.0093, 2.7457, 0.0075, 3.7089

has been supported by the EPSRC First Grant EP/P021298/1, "PARTIAL Analysis of Relations in Tasks of Inversion for Algorithmic Leverage".

We would like to thank Florian Knoll for the water bottle measurement, and Martin Benning for making his codes [6] available.

A DATA STATEMENT FOR THE EPSRC

All data and source codes will be publicly deposited when the final accepted version of the manuscript is submitted.

REFERENCES

[1] Arfken & Weber, *Mathematical methods for physicists*, San Diego: Hartcourt/Academic Press, 2001.

[2] Arrow, Hurwicz & Uzawa, *Studies in Linear and Non-Linear Programming*, Stanford University Press, 1958.

[3] Aubert-Broche, Evans & Collins, A new improved version of realistic digital brain phantom, *NeuroImage* 32 (2006), 138–145.

[4] Bakushinsky & Kokurin, *Iterative methods for approximate solution of inverse problems*, Vol. 577. Springer, 2004.

[5] Bauer & Kannengiesser, An Alternative Approach to the Image Reconstruction for Parallel Data Acquisition in MRI, *Math Meth Appl Sci.* 30 (2007), 1437–1451.

[6] Benning, Knoll, Schönlieb & Valkonen, Research data supporting: Preconditioned ADMM with nonlinear operator constraint, 2016, DOI: 10.17863/CAM.163.

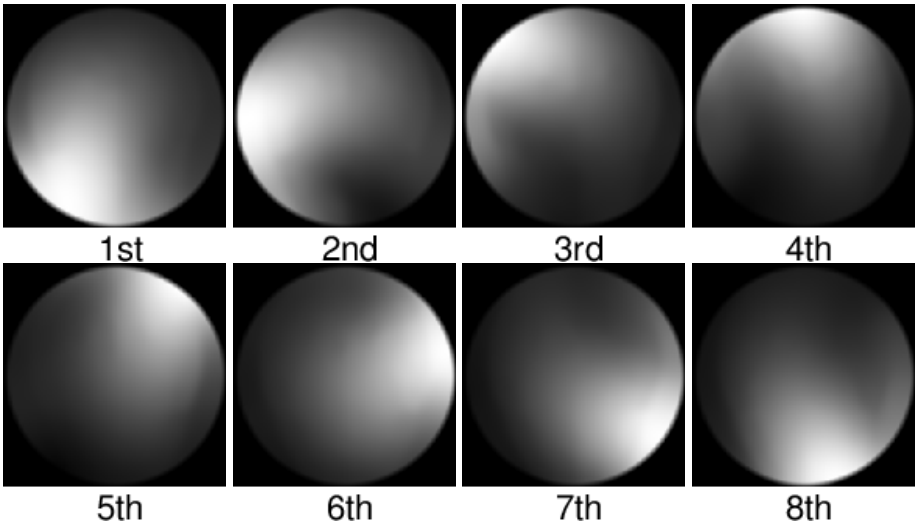


Figure 6: Coil sensitivities generated by the measurements of a water bottle with 8-channel head coil array.

[7] Benning, Knoll, Schönlieb & Valkonen, Preconditioned ADMM with Nonlinear Operator Constraint, in: *System Modeling and Optimization: 27th IFIP TC 7 Conference, CSMO 2015, Sophia Antipolis, France, June 29–July 3, 2015, Revised Selected Papers*, Springer International Publishing, 2016, 117–126, DOI: 10.1007/978-3-319-55795-3_10.

[8] Bernstein, King & Zhou, *Handbook of MRI Pulse Sequences*, Elsevier, 2004.

[9] Blaschke, Neubauer & Scherzer, On convergence rates for the iteratively regularized Gauss-Newton method, *IMA J Numer Anal.* 17 (1997), 421–436.

[10] Bredies, Kunisch & Pock, Total generalized variation, *SIAM J. on Imaging Sci.* 3 (2010), 492–526.

[11] Bydder, Larkman & Hajnal, Generalized SMASH Imaging, *Magn. Reson. Med.* 47 (2002), 160–170.

[12] Chambolle, An algorithm for total variation minimization and applications, *J. Math. Imag. Vis.* 20 (2004), 89–97.

[13] Cheng, *Time varying fields and Maxwell's equations. In: Cheng DK. Field and Wave Electromagnetics*, Chevy Chase, MD:Addison-Wesley, 1989.

[14] Clason & Valkonen, Primal-dual extragradient methods for nonlinear nonsmooth PDE-constrained optimization, *SIAM Journal on Optimization* 27 (2017), 1313–1339.

[15] Combettes & Wajs, Signal recovery by proximal forward-backward splitting, *Multiscale Model. Simul.* 4 (2005), 1168–1200.

[16] De Los Reyes & Schönlieb, Image denoising: Learning noise distribution via PDE-constrained optimization, *Inverse Problems and Imaging* 7 (2013), 1183–1214, arXiv: 1207.3425.

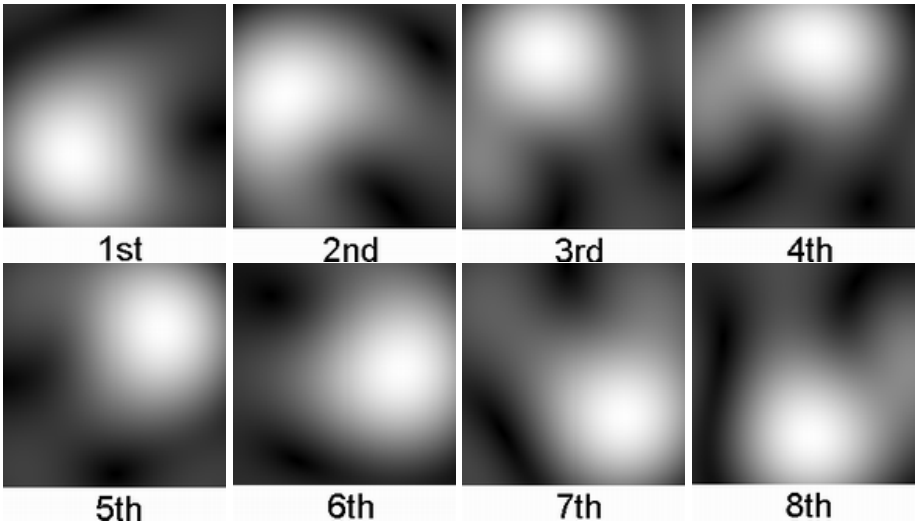


Figure 7: Reconstructed coil sensitivities for the 8 different coils when the number of iterations is 1500 for $\tilde{n} = 2$.

[17] De Los Reyes, Schönlieb & Valkonen, Bilevel parameter learning for higher-order total variation regularisation models, *Journal of Mathematical Imaging and Vision* 57 (2017), 1–25, DOI: 10.1007/s10851-016-0662-8.

[18] Devaney, Multipole expansion and plane wave representations of the electromagnetic field, *J. Math. Phys.* 15 (1974), 234–244.

[19] Engl, Hanke & Neubauer, *Regularization of inverse problems*, Vol. 375. Kluwer Academic Publishers Group, 1996.

[20] Esser, Zhang & Chan, A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science, *SIAM Journal on Imaging Sciences* 3 (2010), 1015–1046, DOI: 10.1137/09076934X.

[21] Gabay, *Applications of the method of multipliers to variational inequalities. In: Augmented Lagrangian Methods:Applications to the Solution of Boundary-Value Problems*, North-Holland, 1983.

[22] Griswold et al., Generalized Autocalibrating Partially Parallel Acquisitions (GRAPPA), *Magn. Reson. Med.* 47 (2002), 1202–1210.

[23] Griswold et al., Autocalibrated Coil Sensitivity Estimation for Parallel Imaging, *NMR Biomed* 19 (2006), 316–324.

[24] Hoge, Brooks, Maddore & Kyriakos, A Tour of Accelerated Parallel MR Imaging from A Linear Systems Perspective, *Concepts Magn Reson Part A* 27 (2005), 17–37.

[25] Hoge, Brooks, Madore & Kyriakos, On the regularization of SENSE and space-RIP in parallel MR imaging, *Proc. IEEE Int Biomedical Imaging: Nano to Macro Symp.* 51 (2004), 241–244.

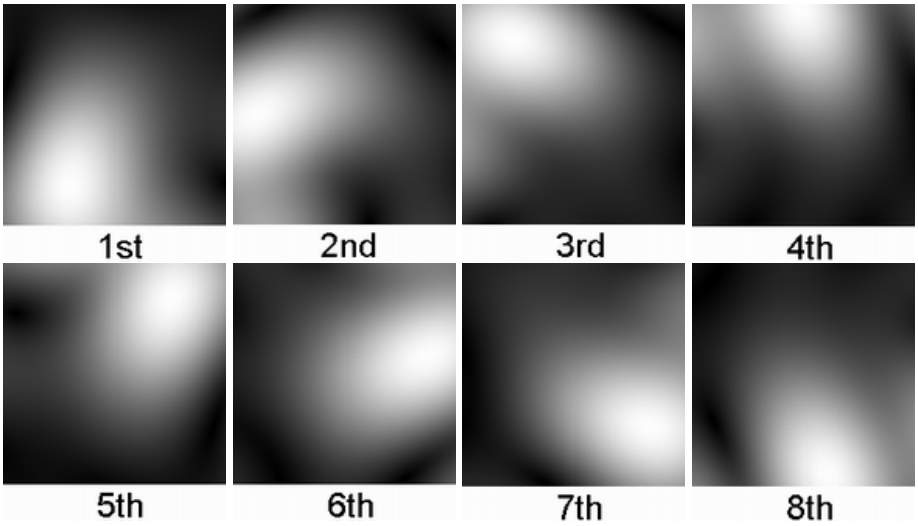


Figure 8: Reconstructed coil sensitivities for the 8 different coils when the number of iterations is 1500 for $\tilde{n} = 5$.

[26] Hohage, Logarithmic Convergence Rates of the Iteratively Regularized Gauss-Newton Method for an Inverse Potential and an Inverse Scattering Problem, *Inverse Problems* 13 (1997), 1279–1299.

[27] Hoult, The Principle of Reciprocity in Signal Strength Calculations-A Mathematical Guide, *Concepts Magn Reson Part A* 12 (2000), 173–187.

[28] Hutchinson & Raff, Fast MRI Data Acquisition Using Multiple Detectors, *Magn. Reson. Med.* 6 (1988), 87–91.

[29] Ibrahim, Analytical approach to the MR Signal, *Magn. Reson. Med.* 54 (2005), 677–682.

[30] Jin et al., An Electromagnetic reverse method of coil sensitivity mapping for parallel MRI-theoretical framework, *J. Magn. Reson.* 207 (2010), 59–68.

[31] Jones, *Acoustic and electromagnetic waves*, New York: Oxford University Press, 1989.

[32] Kelner et al., Electromagnetic fields of surface coil in vivo NMR at high frequencies, *Magn. Reson. Med.* 22 (1991), 467–480.

[33] Knoll, Bredies, Pock & Stollberger, Second Order Total Generalized Variation (TGV) for MRI, *Magn. Reson. Med.* 65 (2011), 480–491.

[34] Knoll, Clason, Uecker & Stollberger, *Improved Reconstruction in Non-cartesian Parallel Imaging by Regularized Nonlinear Inversion*, Proceedings of the 17th Scientific Meeting & Exhibition of ISMRM, 2009.

[35] Knoll et al., Parallel Imaging with Nonlinear Reconstruction using Variational Penalties, *Magn. Reson. Med.* 67 (2012), 34–41.

[36] Kyriakos et al., Sensitivity Profiles from An Array of Coils for Encoding and Reconstruction in Parallel (SPACE RIP), *Magn. Reson. Med.* 44 (2000), 301–308.

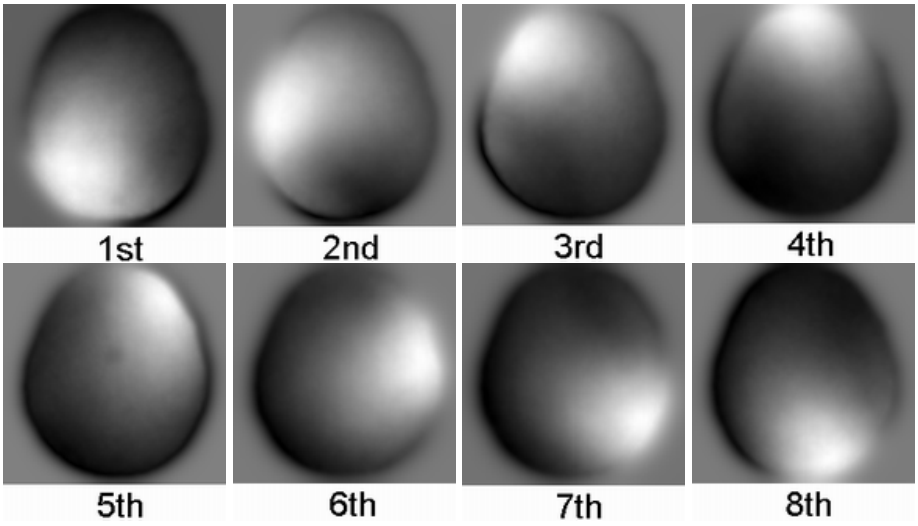
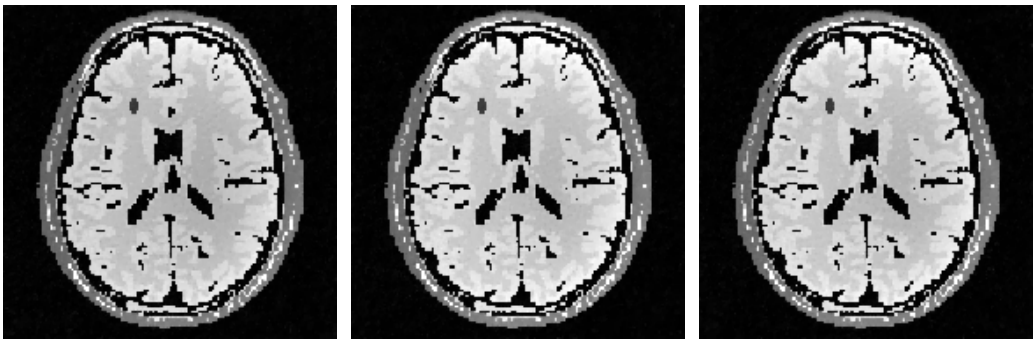


Figure 9: Reconstructed coil sensitivities for the 8 different coils when the number of iterations is 1500 for (1.5).



(a) 1200 iterations (b) 1500 iterations (c) 1800 iterations

Figure 10: Reconstruction when data is generated with the alternative coil sensitivities from Figure 12 with $\tilde{n} = 2$.

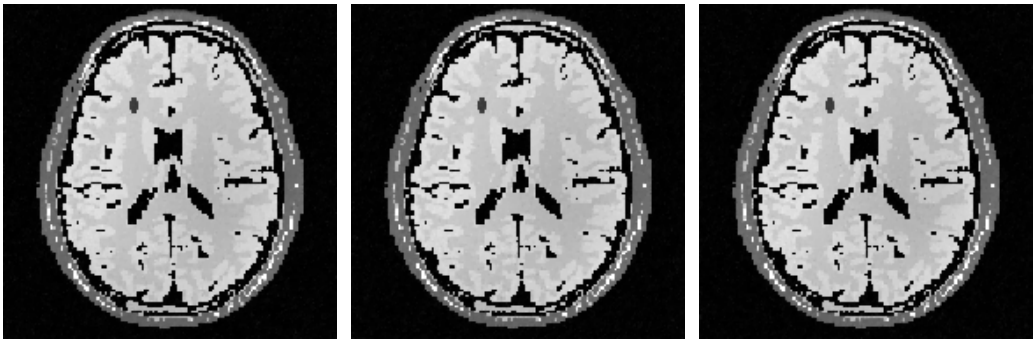
[37] Larkman & Nunes, Parallel Magnetic Resonance Imaging, *Phys. Med. Biol.* 52 (2007), R15–R55.

[38] Lemaire, *in International Series of Numberical Mathematics*, Birkhäuser-Verlag, 1989, 73–87.

[39] Lin, Kwong, Belliveau & Wald, Parallel imaging reconstruction using automatic regularization, *Magn. Reson. Med.* 51 (2004), 559–567.

[40] Martinet, Pertubation des methodes d'optimisation: Application, *RAIRO Anal. Numer.* 12 (1978), 153–171.

[41] McKenzie et al., Self-calibrating Parallel Imaging with Automatic Coil Sensitivity Extraction, *Magn. Reson. Med.* 47 (2002), 529–538.



(a) 1200 iterations (b) 1500 iterations (c) 1800 iterations

Figure 11: Reconstruction when data is generated with the alternative coil sensitivities from Figure 12 with $\tilde{n} = 5$.

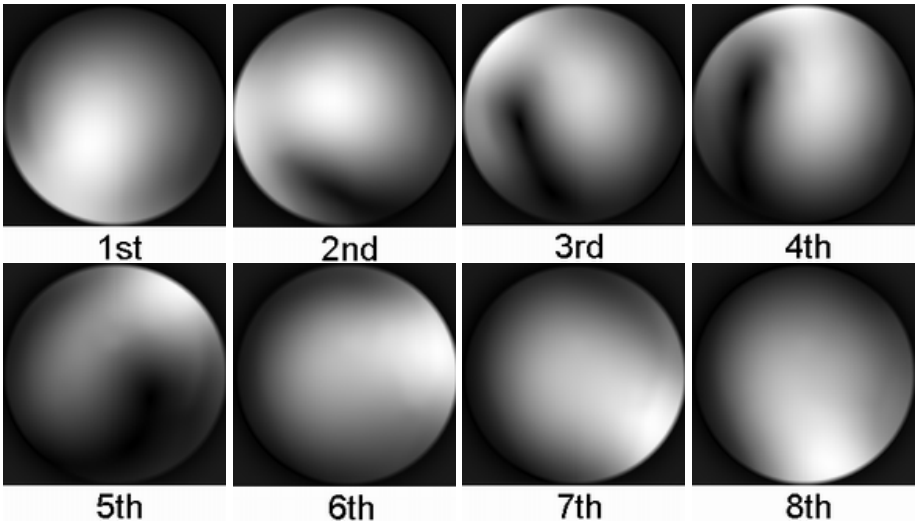


Figure 12: Alternative Perturbed coil sensitivity maps.

[42] Moreau, Proximité et dualite dans un espace Hilbertien, *Bull. Soc. Math. France* 93 (1965), 273–299.

[43] Ocali & Atalar, Ultimate intrinsic signal-to-noise ratio in MRI, *Magn. Reson. Med.* 39 (1998), 462–473.

[44] Pruessmann, *Sensitivity Encoded Magnetic Resonance Imaging, Ph. D. thesis*, Swiss Federal Insititute of Technology, 2000.

[45] Pruessmann, Weiger, Bornert & Boesiger, Advances in Sensitivity Encoding with Arbitrary k-Space Trajectories, *Magn. Reson. Med.* 46 (2001), 638–651.

[46] Pruessmann, Weiger, Scheidegger & Boesiger, SENSE: Sensitivity Encoding for Fast MRI, *Magn. Reson. Med.* 42 (1999), 952–962.

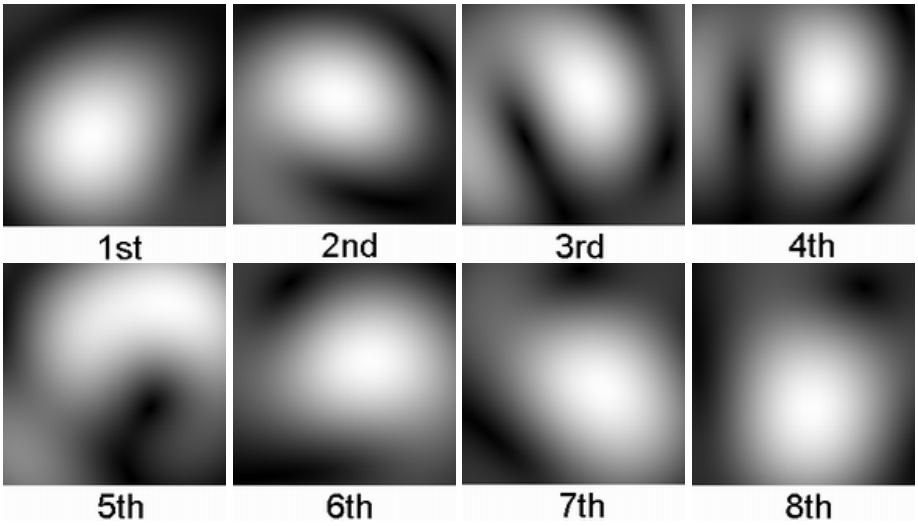


Figure 13: Reconstructed coil sensitivities when data is generated with the alternative coil sensitivities from Figure 12. The number of iterations is 1500 and $\tilde{n} = 2$.

[47] Rockafellar, Augmented Lagrangians and applications of the proximal point algorithm in convex programming, *Math. Oper. Res.* 1 (1976), 97–116.

[48] Rockafellar, Monotone operators and the proximal point algorithm, *SIAM J. Control Optim.* 14 (1976), 877–898.

[49] Rudin, Osher & Fatemi, Nonlinear total variation based noise removal algorithm, *Phys. D* 60 (1992), 259–268.

[50] Sbrizzi et al., Robust reconstruction of B_1^+ maps by projection into a spherical functions space, *Magn. Reson. Med.* 71 (2014), 394–401.

[51] Sodickson & Manning, Simulataneous Acquisition of Spatial Harmonics (SMASH): Fast Imaging with Radiofrequency Coil Arrays, *Magn. Reson. Med.* 38 (1997), 591–603.

[52] Uecker, Hohage, Block & Frahm, Image Reconstruction by Regularized Nonlinear Inversion-Joint Estimation of Coil Sensitivities and Image Content, *Magn. Reson. Med.* 60 (2008), 674–682.

[53] Uecker, Karaus & Frahm, Inverse Reconstruction Method for Segmented Multishot Diffusion-weighted MRI with Multiple Coils, *Magn. Reson. Med.* 62 (2009), 1342–1348.

[54] Valkonen, A primal-dual hybrid gradient method for non-linear operators with applications to MRI, *Inverse Problems* 30 (2014), doi: 10.1088/0266-5611/30/5/055012.

[55] Valkonen, Testing and non-linear preconditioning of the proximal point method (2017), url: https://arxiv.org/abs/1703.05705v2.

[56] Valkonen, Bredies & Knoll, Total generalized variation in diffusion tensor imaging, *SIAM J. on Imaging Sci.* 6 (2013), 487–525.

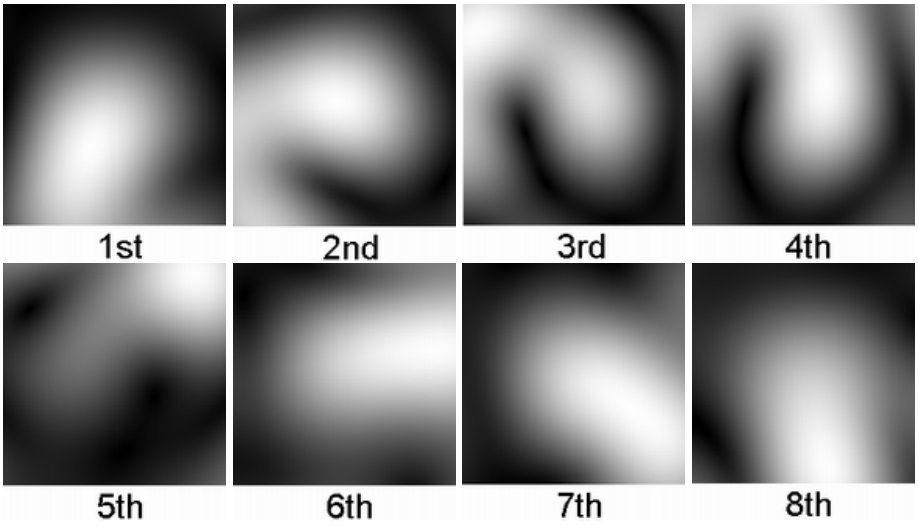


Figure 14: Reconstructed coil sensitivities when data is generated with the alternative coil sensitivities from Figure 12. The number of iterations is 1500 and $\tilde{n} = 5$.

[57] Wang, Bovik, Sheikh & Simoncelli, Image quality assessment: From error visibility to structural similarity, *IEEE Transactions on Image Processing* 13 (2004), 600–612, DOI: 10.1109/TIP.2003.819861.

[58] Wiesinger, Boesiger & Pruessmann, Electrodynamics and ultimate SNR in parallel MR imaging, *Magn. Reson. Med.* 52 (2004), 376–390.

[59] Ying & Sheng, Joint Image Reconstruction and Sensitivity Estimation in SENSE (JSENSE), *Magn. Reson. Med.* 57 (2007), 1196–1202.

[60] Zhang, Burger & Osher, A unified primal-dual algorithm framework based on Bregman iteration, *J Sci Comput* 46 (2011), 20–46.

[61] Zhang, Burger & Osher, A Unified Primal-Dual Algorithm Framework Based on Bregman Iteration, *Journal of Scientific Computing* 46 (2011), 20–46, DOI: 10.1007/s10915-010-9408-8.

[62] Zhu, Parallel Excitation with An Array of Transmit Coils, *Magn. Reson. Med.* 51 (2004), 775–784.

[63] Zhu & Shi, A Fast Method for Reconstruction of Total-Variation MR Images With a Periodic Boundary Condition, *IEEE Signal Processing Letters* 20 (2013), 291–294.

[64] Zhu, Shi, Zhang & Yu, Weighted-average Alternating Minimization Method For Magnetic Resonance Image Reconstruction Based On Compressive Sensing, *Inverse Problems and Imaging* 8 (2014), 925–935.

Table 8: The absolute values of coefficients of f_l^+ with $\tilde{n} = 5$ for 1500 iterations. j is the ordinal number of coils.

j	f_1^+ f_2^+, f_3^+, f_4^+ $f_5^+, f_6^+, f_7^+, f_8^+, f_9^+$ $f_{10}^+, f_{11}^+, f_{12}^+, f_{13}^+, f_{14}^+, f_{15}^+, f_{16}^+$ $f_{17}^+, f_{18}^+, f_{19}^+, f_{20}^+, f_{21}^+, f_{22}^+, f_{23}^+, f_{24}^+, f_{25}^+$ $f_{26}^+, f_{27}^+, f_{28}^+, f_{29}^+, f_{30}^+, f_{31}^+, f_{32}^+, f_{33}^+, f_{34}^+, f_{35}^+, f_{36}^+$	j	f_1^+ f_2^+, f_3^+, f_4^+ $f_5^+, f_6^+, f_7^+, f_8^+, f_9^+$ $f_{10}^+, f_{11}^+, f_{12}^+, f_{13}^+, f_{14}^+, f_{15}^+, f_{16}^+$ $f_{17}^+, f_{18}^+, f_{19}^+, f_{20}^+, f_{21}^+, f_{22}^+, f_{23}^+, f_{24}^+, f_{25}^+$ $f_{26}^+, f_{27}^+, f_{28}^+, f_{29}^+, f_{30}^+, f_{31}^+, f_{32}^+, f_{33}^+, f_{34}^+, f_{35}^+, f_{36}^+$
1	4.9921 .6829, .0438, 4.3006 .8681,.0165, 5.4029,.0174, 3.9672 .9972, .0101, 1.9630, .0107, 1.5569, .0051, 5.2987 .0767, .0075, .0324, .0087, .0344, .0344, .0089, .0135, 5.3953 .0009, .0085, .0078, .0054, .0242, .0053, .0490, .0074, .0038, .0033, .0069	5	3.1244 .2371, .0362, 3.0439 .9841, .0108, 3.2107, .0073, 2.8668 .8269, .0041, .0164, .0065, .5001, .0153, 4.3171 .0811, .0057, .0055, .0057, .0239, .0266, .0237, .0025, 4.6823 .0025, .0061, .0072, .0077, .0125, .0088, .0353, .0091, .0052, .0111, .0058
2	4.4692 .1920, .0465, 4.2089 1.2692, .0071, 4.6202, .0165, 4.1568 .9978, .0050, .0230, .0072, 1.5304, .0038, 4.2382 .6541, .0098, .0122, .0070, .0428, .0284, .0251, .0071, 5.4117 .0031, .0075, .0048, .0064, .0066, .0054, .0288, .0059, .0052, .0059, .0040	6	3.9430 .3079, .0451, 3.1557 .6159, .0090, 4.0539, .0148, 2.8496 1.3257, .0072, .0080, .0128, 1.5127, .0052, 3.1355 .3157, .0061, .0212, .0080, .0327, .0422, .0358, .0122, 4.5738 .0047, .0071, .0085, .0083, .0087, .0076, .0341, .0062, .0099, .0074, .0064
3	3.0346 .3015, .0392, 3.7538 1.6553, .0018, 2.9996, .0041, 4.1566 .9641, .0057, .3390, .0085, 2.2158, .0087, 3.9728 .0021, .0080, .0025, .0139, .0255, .0333, .0106, .0030, 4.4230 .0013, .0075, .0064, .0072, .0030, .0075, .0261, .0075, .0059, .0073, .0048	7	3.4755 .2479, .0338, 2.9170 1.3750, .0061, 3.7187, .0110, 3.4468 .8354, .0115, .1450, .0138, 1.4890, .0046, 3.9267 .0038, .0074, .0169, .0107, .0299, .0389, .0244, .0157, 4.2940 .0028, .0078, .0054, .0075, .0021, .0071, .0275, .0055, .0043, .0068, .0051
4	2.4995 .1197, .0360, 3.3852 1.4643, .0034, 2.4634, .0056, 3.4846 1.0066, .0118, .1241, .0061, 1.5655, .0049, 4.3484 .9035, .0058, .0011, .0142, .0158, .0383, .0126, .0048, 5.1540 .0043, .0079, .0081, .0076, .0034, .0074, .0268, .0071, .0078, .0080, .0095	8	2.9650 .2303, .0268, 2.9961 1.5204, .0118, 3.1554, .0206, 3.4432 1.3120, .0106, .1190, .0056, 1.3044, .0064, 4.4748 .6673, .0071, .0079, .0051, .0222, .0196, .0165, .0123, 4.9405 .0031, .0080, .0065, .0070, .0114, .0076, .0177, .0087, .0153, .0102, .0076

Table 9: Reconstruction quality comparison between (1.7) with $\tilde{n} = 2, 5$ and (1.5) for 1500 iterations.

Method	coil No.	PSNR(dB)	SSIM
using (1.7) with $\tilde{n} = 2$	coil 1	8.5038	0.9864
	coil 2	8.9329	0.9878
	coil 3	8.9895	0.9873
	coil 4	10.7526	0.9918
	coil 5	10.2633	0.9900
	coil 6	10.6595	0.9917
	coil 7	9.0644	0.9869
	coil 8	11.2573	0.9929
using (1.7) with $\tilde{n} = 5$	coil 1	7.2807	0.9849
	coil 2	8.2247	0.9878
	coil 3	7.4537	0.9850
	coil 4	9.3979	0.9903
	coil 5	9.1837	0.9891
	coil 6	10.3233	0.9919
	coil 7	8.5967	0.9876
	coil 8	10.1097	0.9920
using (1.5)	coil 1	5.9826	0.9759
	coil 2	5.9328	0.9754
	coil 3	5.9124	0.9748
	coil 4	6.3114	0.9779
	coil 5	6.1844	0.9762
	coil 6	6.1754	0.9765
	coil 7	6.3423	0.9772
	coil 8	6.4319	0.9774

Table 10: The PSNR and SSIM values for reconstruction with alternative coil sensitivities using (1.7) with $\tilde{n} = 2$ and $\tilde{n} = 5$

Method	stopping itr. k	PSNR(dB)	SSIM
using (1.7) with $\tilde{n} = 2$	1200	27.0202	0.9998
	1500	24.2239	0.9995
	1800	24.9633	0.9996
using (1.7) with $\tilde{n} = 5$	1200	30.8382	0.9999
	1500	30.7620	0.9999
	1800	30.6448	0.9999