REGULARIZED PARALLEL MRI RECONSTRUCTION USING AN ALTERNATING DIRECTION METHOD OF MULTIPLIERS

Sathish Ramani and Jeffrey A. Fessler

EECS Department, University of Michigan, Ann Arbor, MI 48109-2122, USA.

ABSTRACT

Using sparsity-based regularization to improve magnetic resonance image (MRI) reconstruction quality demands computation-intensive nonlinear optimization. In this paper, we develop an iterative algorithm based on the *method of multipliers*—augmented Lagrangian (AL) formalism—for reconstruction from sensitivity encoded data using sparsity-based regularization. We first convert the unconstrained reconstruction problem into an equivalent constrained optimization task and attack the constrained version in an AL framework using an *alternating direction minimization* method—this leads to an *alternating direction method of multipliers* whose intermediate steps are amenable to parallelization. Numerical experiments with *in-vivo* human brain data illustrate that the proposed algorithm converges faster than both general-purpose optimization algorithms such as nonlinear conjugate gradient (NCG) and state-of-the-art MFISTA.

Index Terms— Parallel MRI, SENSE, Image Reconstruction, Regularization, Multiplier Methods

1. INTRODUCTION

SENSitivity Encoding (SENSE) [1, 2] is a parallel MRI (pMRI) technique that depends on the sensitivity maps of the coil array. Standard reconstruction methods for SENSE suffer from SNR degradation due to k-space undersampling and instability arising from correlation in sensitivity maps [1]. As an attractive means of restoring stability in the reconstruction mechanism, sparsity-promoting regularization criteria have gained popularity in MRI [3, 4] due to advances in compressed sensing (CS) theory. Here, we investigate the problem of regularized reconstruction from sensitivity encoded data—SENSE-Reconstruction—using sparsity-promoting regularization. We formulate image reconstruction as an unconstrained optimization problem where we obtain the reconstructed image, $\hat{\mathbf{x}}$, by minimizing a cost function, $J(\mathbf{x})$, composed of a quadratic data-fidelity term and a sparsity-regularization term, $\Psi(\mathbf{x})$. Sparsity-regularization criteria are "non-smooth" (i.e., they may not be differentiable everywhere) and require solving a nonlinear optimization problem using iterative algorithms.

This paper presents an accelerated algorithm for regularized SENSE-reconstruction based on the *method of multipliers*, specifically, the augmented Lagrangian (AL) formalism [5] for solving large-scale constrained problems: We first convert the unconstrained regularized SENSE-reconstruction problem, **P0**, into an equivalent constrained optimization task, **P1**, using a *variable splitting* scheme (different from those in [6, 7]) that separates the various components of J. Then, we construct an augmented Lagrangian (AL) function [5] (that includes a Lagrange multiplier term) for **P1** and minimize it iteratively (while taking care to update the Lagrange multiplier) to solve **P0**. We apply an *alternating* scheme that decouples the minimization of the AL function and simplifies optimization: The resulting algorithm—*alternating direction method of multipliers* (ADMM)—is a "block"-variant of that developed in [8, Sec. 3] and is convergent. Numerical experiments with *in-vivo* human brain data demonstrate that the proposed ADMM converges faster (to a solution of the unconstrained regularized SENSE-reconstruction problem) than general-purpose optimization algorithms such as NCG (that has been applied for CS-(p)MRI in [3, 4]), and the recently proposed state-of-the-art Monotone Fast Iterative Shrinkage-Thresholding Algorithm (MFISTA) [9].

This work was supported by the Swiss National Science Foundation under fellowship PBELP2-125446 and in part by the National Institutes of Health under grant P01 CA87634. The authors would like to thank Dr. Jon-Fredrik Nielsen, University of Michigan, for providing the *in-vivo* human brain data-set used in the experiments.

2. REGULARIZED SENSE-RECONSTRUCTION

We formulate regularized SENSE-reconstruction as

$$\textbf{P0: } \hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ J(\mathbf{x}) = \frac{1}{2}\|\mathbf{d} - \mathbf{F}\,\mathbf{S}\,\mathbf{x}\|_2^2 + \Psi(\mathbf{x}) \right\}, \qquad (1)$$

where $\mathbf{x}$ is a $N \times 1$ column vector containing the samples of the unknown image to be reconstructed, $\mathbf{d}$ is a $ML \times 1$ column vector corresponding to the data-samples from L coils, $\mathbf{S}$ is a $NL \times N$ matrix given by $\mathbf{S} = [\mathbf{S}_1^{\mathrm{H}} \cdots \mathbf{S}_L^{\mathrm{H}}]^{\mathrm{H}}$, $\mathbf{S}_l$ is a $N \times N$ (possibly complex) diagonal matrix corresponding to the sensitivity map of the lth coil, $1 \le l \le L$, $(\cdot)^{\mathrm{H}}$ represents the Hermitian-transpose, $\mathbf{F} = \mathbf{I}_L \otimes \mathbf{F}_{\mathrm{u}}$, $\mathbf{F}_{\mathrm{u}}$ is a $M \times N$ Fourier encoding matrix, $\mathbf{I}_L$ is the identity matrix of size L and $\otimes$ denotes the Kronecker product. The subscript 'u' in $\mathbf{F}_{\mathrm{u}}$ signifies the fact that the k-space may be undersampled to reduce scan time, i.e., $M \le N$. The formulation **P0** can also be used to deal with correlated noise after applying a suitable noise-decorrelation procedure [2, App. B].

For Ψ, we consider the following ℓ_1-regularization ("analysis" form): $\Psi(\mathbf{x}) = \lambda\|\mathbf{W}\mathbf{x}]\|_1$, where $\lambda > 0$ is the regularization parameter and $\mathbf{W}\mathbf{x}$ represents coefficients of the undecimated Haar-wavelet transform excluding the approximation level. The proposed method can be extended easily to deal with synthesis formulations and to include other regularizers involving orthonormal wavelets, total-variation and combinations thereof.

3. ADMM FOR REGULARIZED SENSE-RECONSTRUCTION

This section develops a reconstruction algorithm using the method of multipliers [5] for rapid minimization of J in (1). The basic idea is to break down **P0** into smaller tasks by introducing "artificial" constraints that are designed so that the sub-problems become decoupled and can be solved relatively easily.

3.1. Variable Splitting

Our strategy is to apply variable splitting to **P0** where we replace linear transformations of $\mathbf{x}$ (such as $\mathbf{S}\mathbf{x}$ and $\mathbf{W}\mathbf{x}$) in J with a set of auxiliary variables $\{\mathbf{u}_l\}$. Then, we frame **P0** as an equivalent constrained problem:

$$\textbf{P1:}\quad \min_{\mathbf{u}_0,\mathbf{u}_1,\mathbf{u}_2,\mathbf{x}} \left\{ f(\mathbf{u}_0,\mathbf{u}_1) \triangleq \frac{1}{2}\|\mathbf{d}-\mathbf{F}\mathbf{u}_0\|_2^2 + \lambda\|\mathbf{u}_1\|_1 \right\}$$
$$\text{subject to } \mathbf{u}_0 = \mathbf{S}\mathbf{x}, \mathbf{u}_1 = \mathbf{W}\mathbf{u}_2, \mathbf{u}_2 = \mathbf{x}, \tag{2}$$

which can be concisely rewritten as

$$\mathbf{P1}: \min_{\mathbf{u}} f(\mathbf{u}) \text{ subject to } \mathbf{C}\mathbf{u} = \mathbf{0}, \tag{3}$$

where $\mathbf{C}$ is a $N_1 \times N_2$ matrix given by

$$\mathbf{C} \triangleq \begin{bmatrix} \mathbf{I}_{NL} & \mathbf{0} & \mathbf{0} & -\mathbf{S} \\ \mathbf{0} & \sqrt{\nu_1}\mathbf{I}_W & -\sqrt{\nu_1}\mathbf{W} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \sqrt{\nu_2}\mathbf{I}_N & -\sqrt{\nu_2}\mathbf{I}_N \end{bmatrix}, \ \mathbf{u} \triangleq \begin{bmatrix} \mathbf{u}_0 \\ \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{x} \end{bmatrix},$$

$\mathbf{u}_0 \in \mathbb{C}^{NL}$, $\mathbf{u}_1 \in \mathbb{C}^W$, $\mathbf{u}_2, \mathbf{x} \in \mathbb{C}^N$, W is the number of rows of $\mathbf{W}$, $N_1 = NL + W + N$, $N_2 = NL + W + 2N$, and $\nu_{1,2} > 0$ are scalars (whose purpose is explained in Section 3.3.1) that do not affect the solution of **P1**. The splitting in (2) differs from those in [6, 7] in that it not only separates the regularization and data-fidelity terms (using $\mathbf{u}_1$), but also splits the components inside the data-fidelity i.e., $\mathbf{F}$ and $\mathbf{S}$ (using $\mathbf{u}_0$) and decouples $\mathbf{S}$ and $\mathbf{W}$ (using $\mathbf{u}_2$). This form of splitting (using $\mathbf{u}_{0,1,2}$) leads to simple matrix-inverses in the optimization [see Section 3.3.1)].

3.2. Augmented Lagrangian (AL) Formalism

Constrained problems of the type **P1** can be effectively handled in a multiplier-method-framework [5]. Specifically, we use the augmented Lagrangian (AL) formalism where an AL function is constructed for **P1** as

$$\mathcal{L}_1(\mathbf{u},\boldsymbol{\gamma},\mu) \triangleq f(\mathbf{u}) + \boldsymbol{\gamma}^{\mathrm{H}}\mathbf{C}\mathbf{u} + \frac{\mu}{2}\|\mathbf{C}\mathbf{u}\|_2^2 \tag{4}$$

with penalty parameter $\mu > 0$ and Lagrange multiplier $\boldsymbol{\gamma} \in \mathbb{C}^{N_1}$. The AL scheme for solving **P1** alternates between minimizing $\mathcal{L}_1$ with respect to $\mathbf{u}$ for a fixed $\boldsymbol{\gamma}$ and updating $\boldsymbol{\gamma}$:

$$\mathbf{u}^{(j+1)} = \arg\min_{\mathbf{u}} \mathcal{L}_1(\mathbf{u},\boldsymbol{\gamma}^{(j)},\mu), \tag{5}$$
$$\boldsymbol{\gamma}^{(j+1)} = \boldsymbol{\gamma}^{(j)} + \mu\,\mathbf{C}\mathbf{u}^{(j+1)}, \tag{6}$$

until some stopping criterion is satisfied. Importantly, (5)-(6) converge to a solution of **P1** (and **P0**) without the need for changing μ [5]. The AL function $\mathcal{L}_1$ in (4) can be simplified by grouping together the terms involving $\mathbf{C}\mathbf{u}$ as

$$\mathcal{L}_1(\mathbf{u},\boldsymbol{\eta}^{(j)},\mu) = f(\mathbf{u}) + \frac{\mu}{2}\|\mathbf{C}\mathbf{u} - \boldsymbol{\eta}^{(j)}\|_2^2 + C_{\boldsymbol{\gamma}}, \tag{7}$$

where $C_{\boldsymbol{\gamma}}$ is an irrelevant constant that we ignore henceforth and $\boldsymbol{\eta} \triangleq -\frac{1}{\mu}\boldsymbol{\gamma}$ so that (6) becomes $\boldsymbol{\eta}^{(j+1)} = \boldsymbol{\eta}^{(j)} - \mathbf{C}\mathbf{u}^{(j+1)}$.

Applying (5)-(6) to **P1** requires the joint minimization of $\mathcal{L}_1$ with respect to $\mathbf{u}_{0,1,2}$ and $\mathbf{x}$ in (5). Since this can be computationally challenging, we develop below an alternating minimization method similar to [8] that decouples the problem with respect to $\mathbf{u}_{0,1,2}$ and $\mathbf{x}$, thereby simplifying optimization.

3.3. An Alternating Minimization Scheme

We rewrite **P1** in the spirit of [8, Sec. 3] as

$$\mathbf{P2:}\ \min_{\mathbf{u},\mathbf{v}} f(\mathbf{u}) + g(\mathbf{v}) \text{ subject to } [\mathbf{A} \ \ -\mathbf{I}_{2N_1}] \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} = \mathbf{0}, \tag{8}$$

where[1] $\mathbf{A}$ is a $2N_1 \times N_2$ matrix with full column-rank given by

[1] Note that (9) is a "block"-variant of [8, Eq. (18)].

ADMM for Regularized SENSE-Reconstruction
1. Select $\mathbf{u}^{(0)}$, $\mathbf{v}^{(0)}$, $\boldsymbol{\eta}^{(0)}$, and μ, ν_1, $\nu_2 > 0$; set $j = 0$
Repeat:
2. $\mathbf{u}_0^{(j+1)} = \arg\min_{\mathbf{u}_0} \left\{ \frac{1}{2}\|\mathbf{d}-\mathbf{F}\mathbf{u}_0\|_2^2 + \frac{\mu}{2}\|\mathbf{u}_0 - \mathbf{v}_0^{(j)} - \boldsymbol{\eta}_0^{(j)}\|_2^2 \right\}$
3. $\mathbf{u}_1^{(j+1)} = \arg\min_{\mathbf{u}_1} \left\{ \lambda\|\mathbf{u}_1\|_1 + \frac{\mu}{2}\|\sqrt{\nu_1}\mathbf{u}_1 - \mathbf{v}_2^{(j)} - \boldsymbol{\eta}_2^{(j)}\|_2^2 \right\}$
4. $\mathbf{u}_2^{(j+1)} = \arg\min_{\mathbf{u}_2} \left\{ \begin{array}{l} \|\sqrt{\nu_1}\mathbf{W}\mathbf{u}_2 + \mathbf{v}_3^{(j)} + \boldsymbol{\eta}_3^{(j)}\|_2^2 \\ +\|\sqrt{\nu_2}\mathbf{u}_2 - \mathbf{v}_4^{(j)} - \boldsymbol{\eta}_4^{(j)}\|_2^2 \end{array} \right\}$
5. $\mathbf{x}^{(j+1)} = \arg\min_{\mathbf{x}} \left\{ \begin{array}{l} \|\mathbf{S}\mathbf{x} + \mathbf{v}_1^{(j)} + \boldsymbol{\eta}_1^{(j)}\|_2^2 \\ +\|\sqrt{\nu_2}\mathbf{x} + \mathbf{v}_5^{(j)} + \boldsymbol{\eta}_5^{(j)}\|_2^2 \end{array} \right\}$
6. $\mathbf{v}^{(j+1)} = \arg\min_{\mathbf{v}\in\Omega} \frac{\mu}{2}\|\mathbf{A}\mathbf{u}^{(j+1)} - \mathbf{v} - \boldsymbol{\eta}^{(j)}\|_2^2$
7. $\boldsymbol{\eta}^{(j+1)} = \boldsymbol{\eta}^{(j)} - (\mathbf{A}\mathbf{u}^{(j+1)} - \mathbf{v}^{(j+1)})$
8. Set $j = j + 1$
Until stop criterion is met

$$\mathbf{A} \triangleq \begin{bmatrix} \mathbf{I}_{NL} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{S} \\ \mathbf{0} & \sqrt{\nu_1}\mathbf{I}_W & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\sqrt{\nu_1}\mathbf{W} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \sqrt{\nu_2}\mathbf{I}_N & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\sqrt{\nu_2}\mathbf{I}_N \end{bmatrix}, \tag{9}$$

$$g(\mathbf{v}) \triangleq \begin{cases} 0, & \mathbf{v} \in \Omega \\ +\infty, & \mathbf{v} \notin \Omega \end{cases}, \qquad \Omega \triangleq \{\mathbf{v} \in \mathbb{C}^{2N_1} \mid \mathbf{B}\mathbf{v} = \mathbf{0}\},$$

$\mathbf{v} = [\mathbf{v}_0^{\mathrm{H}}\, \mathbf{v}_1^{\mathrm{H}} \cdots \mathbf{v}_5^{\mathrm{H}}]^{\mathrm{H}}$, the component $\mathbf{v}_i$ corresponds to the block-row $\mathbf{A}_{i\cdot}$ of $\mathbf{A}$, $i = 0, 1, \cdots, 5$, and $\mathbf{B}$ is a $N_1 \times 2N_1$ matrix:

$$\mathbf{B} \triangleq \begin{bmatrix} \mathbf{I}_{NL} & \mathbf{I}_{NL} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_W & \mathbf{I}_W & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_N & \mathbf{I}_N \end{bmatrix}$$

that satisfies

$$\mathbf{B}\mathbf{A} = \mathbf{C}, \tag{10}$$
$$\mathbf{B}\mathbf{B}^{\mathrm{T}} = 2\mathbf{I}_{N_1}. \tag{11}$$

From (8)-(11) and the definitions of $g(\cdot)$ and Ω, it is verified that **P2** is equivalent to **P1**. We now apply the AL formalism to the constrained problem **P2** and construct an AL function similar to (7):

$$\mathcal{L}_2(\mathbf{u},\mathbf{v},\boldsymbol{\eta},\mu) \triangleq f(\mathbf{u}) + g(\mathbf{v}) + \frac{\mu}{2}\|\mathbf{A}\mathbf{u} - \mathbf{v} - \boldsymbol{\eta}\|_2^2, \tag{12}$$

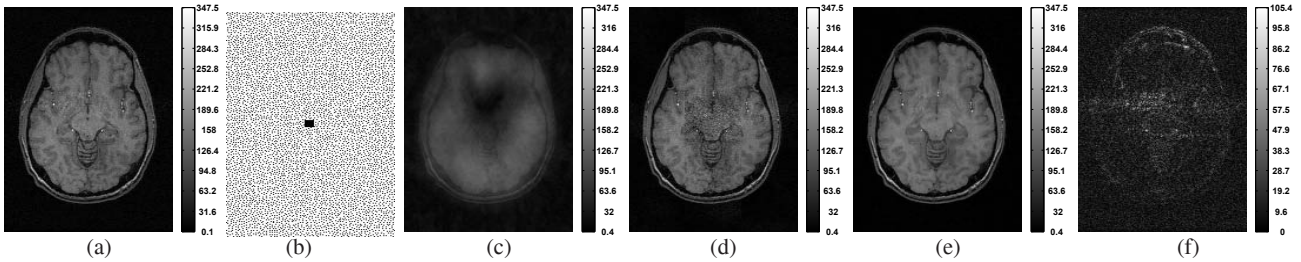
where $\boldsymbol{\eta} \triangleq [\boldsymbol{\eta}_0^{\mathrm{H}}\, \boldsymbol{\eta}_1^{\mathrm{H}} \cdots \boldsymbol{\eta}_5^{\mathrm{H}}]^{\mathrm{H}} \in \mathbb{C}^{2N_1}$ one component for each block-row of $\mathbf{A}$. However, instead of carrying out the conventional AL steps (similar to (5)-(6))

$$(\mathbf{u}^{(j+1)},\mathbf{v}^{(j+1)}) = \arg\min_{\mathbf{u},\mathbf{v}} \mathcal{L}_2(\mathbf{u},\mathbf{v},\boldsymbol{\eta}^{(j)},\mu), \tag{13}$$
$$\boldsymbol{\eta}^{(j+1)} = \boldsymbol{\eta}^{(j)} - (\mathbf{A}\mathbf{u}^{(j+1)} - \mathbf{v}^{(j+1)}), \tag{14}$$

we apply alternating direction minimization [8]:

$$\mathbf{u}^{(j+1)} = \arg\min_{\mathbf{u}} \mathcal{L}_2(\mathbf{u},\mathbf{v}^{(j)},\boldsymbol{\eta}^{(j)},\mu), \tag{15}$$
$$\mathbf{v}^{(j+1)} = \arg\min_{\mathbf{v}} \mathcal{L}_2(\mathbf{u}^{(j+1)},\mathbf{v},\boldsymbol{\eta}^{(j)},\mu), \tag{16}$$
$$\boldsymbol{\eta}^{(j+1)} = \boldsymbol{\eta}^{(j)} - (\mathbf{A}\mathbf{u}^{(j+1)} - \mathbf{v}^{(j+1)}). \tag{17}$$

Due to the structure of $\mathbf{A}$, (15) further decouples into the minimization of $\mathcal{L}_2(\cdot,\mathbf{v}^{(j)},\boldsymbol{\eta}^{(j)},\mu)$ with respect to $\mathbf{u}_0$, $\mathbf{u}_1$, $\mathbf{u}_2$ and $\mathbf{x}$ individually. This leads to the alternating direction method of multipliers (**ADMM**) [8] given above (where we have ignored constants irrelevant for optimization in Steps 2-6). Because Steps 2-6 accomplish (15)-(16), a variant of [8, Th. 3.1] ensures convergence of **ADMM** to a solution of **P2** (and **P0**).



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

Fig. 1. Experiment with *in-vivo* human brain data (Slice 38): (a) Body-coil image (from fully-sampled phase-encodes); (b) Poisson-disk-based k-space sampling pattern (on a Cartesian grid) with reduction factor $\gtrapprox 6$; (c) Square-root of SoS (SRSoS) of coil images (from undersampled data); (d) Conventional SENSE-reconstruction [1] corresponding to Cartesian sub-sampling by 3×2; (e) The solution $\mathbf{x}^{(\infty)}$ to **P0** obtained by running **MFISTA-20**; (f) Absolute difference between (a) and (e).

3.3.1. Minimization with respect to $\mathbf{u}$*—Steps 2-5 of* **ADMM**

The cost functions corresponding to $\mathbf{u}_0$, $\mathbf{u}_2$, and $\mathbf{x}$ in Steps 2, 4, and 5, respectively, are all quadratic in nature and have closed form solutions:

$$\mathbf{u}_0^{(j+1)} = \mathbf{H}_\mu^{-1}(\mathbf{F}^{\mathrm{H}}\mathbf{d} + \mu(\mathbf{v}_0^{(j)} + \boldsymbol{\eta}_0^{(j)})), \quad (18)$$

$$\mathbf{u}_2^{(j+1)} = \mathbf{H}_{\nu_1\nu_2}^{-1}\begin{bmatrix} \sqrt{\nu_2}(\mathbf{v}_4^{(j)} + \boldsymbol{\eta}_4^{(j)}) \\ -\sqrt{\nu_1}\mathbf{W}^{\mathrm{H}}(\mathbf{v}_3^{(j)} + \boldsymbol{\eta}_3^{(j)}) \end{bmatrix}, \quad (19)$$

$$\mathbf{x}^{(j+1)} = \mathbf{H}_{\nu_2}^{-1}\begin{bmatrix} -\sqrt{\nu_2}(\mathbf{v}_5^{(j)} + \boldsymbol{\eta}_5^{(j)}) \\ -\mathbf{S}^{\mathrm{H}}(\mathbf{v}_1^{(j)} + \boldsymbol{\eta}_1^{(j)}) \end{bmatrix}, \quad (20)$$

where $\mathbf{H}_\mu = (\mathbf{F}^{\mathrm{H}}\mathbf{F} + \mu\mathbf{I}_{ML})$, $\mathbf{H}_{\nu_1\nu_2} = (\nu_1\mathbf{W}^{\mathrm{H}}\mathbf{W} + \nu_2\mathbf{I}_N)$, and $\mathbf{H}_{\nu_2} = (\mathbf{S}^{\mathrm{H}}\mathbf{S} + \nu_2\mathbf{I}_N)$. Although μ, ν_1 and ν_2 do not affect the solution to which **ADMM** converges, they determine its convergence speed. We adjust μ, ν_1 and ν_2 so as to obtain condition numbers $\kappa(\mathbf{H}_\mu)$, $\kappa(\mathbf{H}_{\nu_1\nu_2})$, $\kappa(\mathbf{H}_{\nu_2})$ of $\mathbf{H}_\mu$, $\mathbf{H}_{\nu_1\nu_2}$, and $\mathbf{H}_{\nu_2}$, respectively, that yield fast convergence of **ADMM**.

When the k-space samples lie on a Cartesian grid, $\mathbf{F}_{\mathrm{u}}$ corresponds to a sub-sampled DFT matrix in which case we solve (18) exactly using FFTs. For non-Cartesian k-space trajectories, computing $\mathbf{u}_0^{(j+1)}$ requires an iterative method. For example, a conjugate gradient (CG) solver (with *warm starting*, i.e., to obtain $\mathbf{u}_0^{(j+1)}$, the CG algorithm is initialized with $\mathbf{u}_0^{(j)}$) can be used for Step 2. Since $\mathbf{W}$ constitutes an undecimated wavelet transform, $\mathbf{W}^{\mathrm{H}}\mathbf{W}$ is circulant under periodic boundary conditions. Then, we compute $\mathbf{u}_2^{(j+1)}$ exactly using FFTs. Finally, $\mathbf{H}_{\nu_2}$ is diagonal (due to the diagonal nature of $\mathbf{S}^{\mathrm{H}}\mathbf{S}$) and is therefore exactly inverted to yield $\mathbf{x}^{(j+1)}$.

Minimization with respect to $\mathbf{u}_1$—Step 3—can be decoupled in terms of its components, i.e., for $n = 1, 2, \ldots, W$,

$$u_{1n}^{(j+1)} = \arg\min_{u_{1n}} \left\{ \lambda|u_{1n}| + \frac{\mu}{2}(\sqrt{\nu_1}u_{1n} - v_{2n}^{(j)} - \eta_{2n}^{(j)})^2 \right\},$$

whose solution is given by the soft-thresholding rule:

$$u_{1n}^{(j+1)} = \mathrm{soft}\left\{ (v_{2n}^{(j)} + \eta_{2n}^{(j)})/\sqrt{\nu_1},\ \ \lambda/(\mu\nu_1) \right\}, \quad (21)$$

where $\mathrm{soft}\{d, \lambda\} = \mathrm{sign}(d)\max\{|d| - \lambda, 0\}$, and $u_{1n}^{(j+1)}$, $v_{2n}^{(j)}$, and $\eta_{2n}^{(j)}$ are the nth component of $\mathbf{u}_1^{(j+1)}$, $\mathbf{v}_2^{(j)}$, $\boldsymbol{\eta}_2^{(j)}$, respectively.

3.3.2. Minimization with respect to $\mathbf{v}$*—Step 6 of* **ADMM**

We handle the constrained problem in Step 6 using a Lagrangian multiplier term $\boldsymbol{\alpha} \in \mathbb{C}^{N_1}$:

$$(\boldsymbol{\alpha}^{(j+1)}, \mathbf{v}^{(j+1)}) = \arg\min_{(\boldsymbol{\alpha},\mathbf{v})} \frac{\mu}{2}\|\mathbf{A}\mathbf{u}^{(j+1)} - \mathbf{v} - \boldsymbol{\eta}^{(j)}\|_2^2 + \boldsymbol{\alpha}^{\mathrm{H}}\mathbf{B}\mathbf{v},$$

whose solution can be found using (11) as [8]

$$\boldsymbol{\alpha}^{(j+1)} = \frac{\mu}{2}(\mathbf{C}\mathbf{u}^{(j+1)} - \mathbf{B}\boldsymbol{\eta}^{(j)}), \quad (22)$$

$$\mathbf{v}^{(j+1)} = (\mathbf{A}\mathbf{u}^{(j+1)} - \boldsymbol{\eta}^{(j)}) - \frac{1}{\mu}\mathbf{B}^{\mathrm{T}}\boldsymbol{\alpha}^{(j+1)}. \quad (23)$$

From (23) and Step 7 of **ADMM**, we see that

$$\boldsymbol{\eta}^{(j)} = -\frac{1}{\mu}\mathbf{B}^{\mathrm{T}}\boldsymbol{\alpha}^{(j)}\ \forall\ j. \quad (24)$$

Using (11), (22), and (24), we thus get the update rule

$$\boldsymbol{\alpha}^{(j+1)} = \boldsymbol{\alpha}^{(j)} + \frac{\mu}{2}\mathbf{C}\mathbf{u}^{(j+1)}. \quad (25)$$

Substituting (24) in (23), and using (25), we get that

$$\mathbf{v}^{(j+1)} = \left(\mathbf{A} - \frac{1}{2}\mathbf{B}^{\mathrm{T}}\mathbf{C}\right)\mathbf{u}^{(j+1)}. \quad (26)$$

Since products with $\mathbf{A}$, $\mathbf{B}^{\mathrm{T}}$ and $\mathbf{C}$ amount to simple linear operations, we propose to store the lower-dimensional vectors $\boldsymbol{\alpha}^{(\cdot)}$ and $\mathbf{u}^{(\cdot)}$ in place of higher-dimensional vectors $\boldsymbol{\eta}^{(\cdot)}$ and $\mathbf{v}^{(\cdot)}$, respectively, and employ (24)-(26) to find $\boldsymbol{\eta}^{(\cdot)}$ and $\mathbf{v}^{(\cdot)}$ for use in (18)-(21). In summary, with the possible exception of Step 2, all steps in **ADMM** are exact and can be implemented efficiently. Moreover, the updates in (18)-(21) are independent of each other and may therefore be computed in parallel.

4. EXPERIMENTAL RESULTS

We compared the proposed **ADMM** to NCG (which has been used for CS-(p)MRI [3, 4]) and to the recently proposed MFISTA [9]. For the minimization step [9, Eq. 3.13] in MFISTA, we applied the Chambolle-type algorithm developed in [10] as it does not require "rounding" the "corners" of $\Psi(\mathbf{x})$ unlike NCG [3, App. A]. We implemented the following algorithms in MATLAB: **MFISTA-*N*** with ***N*** iterations of [10, Eq. 6], **NCG-*N*** with ***N*** line-search iterations, and **ADMM**. We conducted the experiments on a 8-core PC with 2.67 GHz Intel Xeon processors.

We quantified the speed of convergence to a solution of **P0**, i.e., the limit $\mathbf{x}^{(\infty)}$, by computing the normalized ℓ_2-distance between $\mathbf{x}^{(j)}$ and $\mathbf{x}^{(\infty)}$: $\xi(j) = 20\log_{10}(\|\mathbf{x}^{(j)} - \mathbf{x}^{(\infty)}\|_2/\|\mathbf{x}^{(\infty)}\|_2)$. We obtained $\mathbf{x}^{(\infty)}$ numerically by running 5000 iterations of **MFISTA-20**. We evaluated $\xi(j)$ as a function of algorithm run-time t_j (time elapsed from start until iteration j) because the algorithms have different computational loads per outer-iteration. We used the square-root of sum of squares (SRSoS) of coil images (obtained by taking inverse Fourier transform of the undersampled data after filling the missing k-space samples with zeros) as our initial guess $\mathbf{x}^{(0)}$ for all algorithms. We found empirically that choosing μ, ν_1 and ν_2 such that $\kappa(\mathbf{H}_\mu), \kappa(\mathbf{H}_{\nu_1\nu_2}), \kappa(\mathbf{H}_{\nu_2}) \in [10,\ 36]$ generally provided good convergence speeds for **ADMM**, so we simply used $\kappa(\mathbf{H}_\mu) = 24$, $\kappa(\mathbf{H}_{\nu_1\nu_2}) = 12$, $\kappa(\mathbf{H}_{\nu_2}) = \min\{0.9\kappa(\mathbf{S}^{\mathrm{H}}\mathbf{S}),\ 12\}$ in all our experiments. We set $\mathbf{u}_0^{(0)} = \mathbf{S}\mathbf{x}^{(0)}$, $\mathbf{u}_2^{(0)} = \mathbf{x}^{(0)}$, $\mathbf{u}_1^{(0)} = \mathbf{W}\mathbf{u}_2^{(0)}$, $\mathbf{v}^{(0)} =$

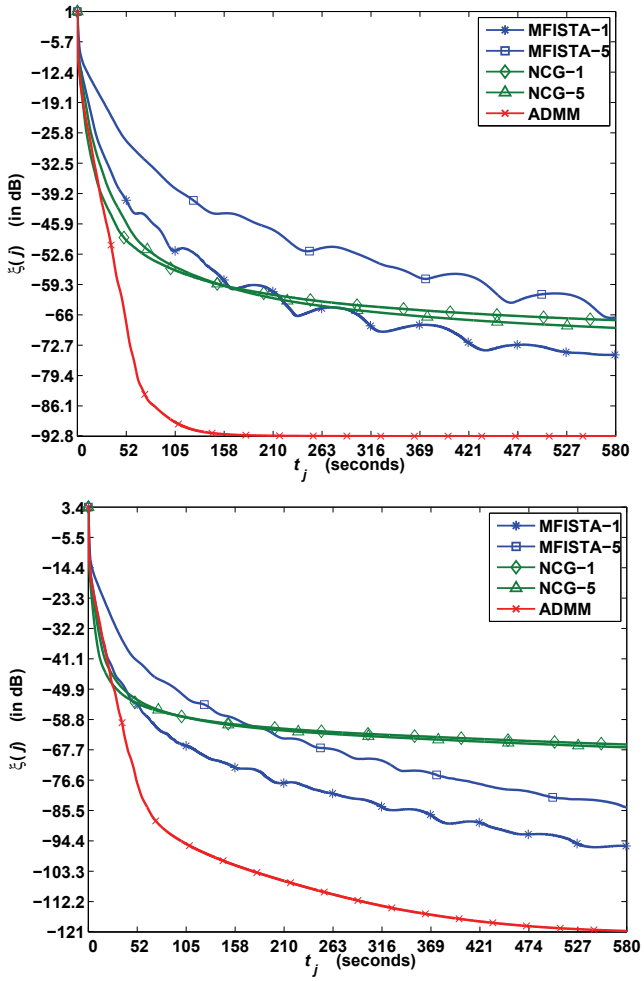


Fig. 2. Plot of $\xi(j)$ as a function time t_j for NCG, MFISTA, and **ADMM** for Slice 38 (top) and Slice 90 (bottom).

$(\mathbf{A} - \frac{1}{2}\mathbf{B}^{\mathrm{T}}\mathbf{C})\mathbf{u}^{(0)}$, $\boldsymbol{\alpha}^{(0)} = \frac{\mu}{2}\mathbf{C}\mathbf{u}^{(0)}$, and $\boldsymbol{\eta}^{(0)} = -\frac{1}{\mu}\mathbf{B}^{\mathrm{T}}\boldsymbol{\alpha}^{(0)}$ in accordance with (2), (24)-(26) and executed the steps of **ADMM** sequentially.

We used a 3D *in-vivo* human brain data-set acquired from a GE 3T scanner ($T_R = 25$ ms, $T_E = 5.172$ ms, and voxel-size = $1 \times 1.35 \times 1$ mm^3), with a 8-channel head-coil. The k-space data corresponded to $256 \times 144 \times 128$ uniformly-spaced samples in the k_x and k_y (phase-encode plane), and k_z (read-out) directions, respectively. We used the iFFT-reconstruction of fully-sampled data collected simultaneously from a body-coil as a reference for quality. Slice 38 (along x-y direction) of the reference body-coil image-volume is shown in Figure 1a. To estimate the sensitivity maps $\mathbf{S}$ associated with a slice, we separately optimized a quadratic-regularized least-squares criterion that encouraged smooth maps that "closely" fit a low-resolution body-coil image to low-resolution head-coil images, all obtained from iFFT-reconstruction of corresponding central 32×32 phase-encodes. We estimated the inverse of noise covariance matrix from data collected during a dummy scan where only the static magnetic-field (and no RF excitations) was applied and carried out noise-decorrelation of data as described in [2, App. B].

We then performed regularized SENSE-reconstruction of 2-D slices (x-y plane)—Slices 38 and 90—from undersampled phase-encodes: For experiments with both slices, we applied the Poisson-disk-sampling pattern with reduction factor $\gtrapprox 6$ in Figure 1b in the phase-encode plane (as a means of conducting 3D CS-(p)MRI). We obtained the reconstructions ($\mathbf{x}^{(\infty)}$) corresponding to Slices 38 and 90 by running **MFISTA-20**. Figure 1e shows $\mathbf{x}^{(\infty)}$ for Slice 38 where aliasing artifacts and noise have been suppressed considerably compared to SRSoS image (Figure 1c) and conventional SENSE-reconstruction [1] (Figure 1d) corresponding to a reduction factor of 6. We ran **NCG-1**, **NCG-5**, **MFISTA-1**, **MFISTA-5**, and **ADMM** and computed ξ for experiments with both slices. Figure 2 plots $\xi(j)$ for the all algorithms as a function of t_j. **ADMM** converges faster than NCG and MFISTA in both cases.

5. SUMMARY & CONCLUSION

Based on the augmented Lagrangian (AL) framework, we developed an alternating direction method of multipliers (**ADMM**) for regularized SENSE-reconstruction: First, we introduced a set of constraint variables and converted the original unconstrained regularized SENSE-reconstruction problem, **P0**, into an equivalent constrained task, **P1**, that is a "block"-variant of [8, Eq. (15)-(18)]. We then applied the AL framework to **P1** with an alternating direction minimization scheme that decouples and simplifies the optimization of the associated AL function with respect to these constraint variables. We demonstrated based on numerical experiments with real MR data that **ADMM** converges to a solution of **P0** faster than conventional (NCG) and state-of-the-art (MFISTA) methods. The proposed **ADMM** can be applied to Cartesian/nonCartesian pMRI, is amenable to parallelization and is easy to implement.

6. REFERENCES

[1] K. P. Pruessmann, M. Weiger, M. B. Scheidegger, and P. Boesiger, "SENSE: Sensitivity Encoding for Fast MRI," *Mag. Res. Med.*, vol. 42, pp. 952–962, 1999.

[2] K. P. Pruessmann, M. Weiger, P. Börnert, and P. Boesiger, "Advances in Sensitivity Encoding with Arbitrary k-Space Trajectories," *Mag. Res. Med.*, vol. 46, pp. 638–651, 2001.

[3] M. Lustig, D. L. Donoho, and J. M. Pauly, "Sparse MRI: The Application of Compressed Sensing for Rapid MR Imaging," *Mag. Res. Med.*, vol. 58, pp. 1182–1195, 2007.

[4] D. Liang, B. Liu, J. Wang, and L. Ying, "Accelerating SENSE Using Compressed Sensing," *Mag. Res. Med.*, vol. 62, pp. 1574–1584, 2009.

[5] D. P. Bertsekas, "Multiplier Methods: A Survey," *Automatica*, vol. 12, pp. 133–145, 1976.

[6] T. Goldstein and S. Osher, "The Split Bregman Method for L_1-Regularized Problems," *SIAM J. Imaging Sciences*, vol. 2, no. 2, pp. 323–343, 2009.

[7] M. V. Afonso, J. M. Biouscas-Dias, and M. A. T. Figueiredo, "Fast Image Recovery Using Variable Splitting and Constrained Optimization," *IEEE Trans. Image Processing*, vol. 19, no. 9, pp. 2345–2356, 2010.

[8] J. Eckstein, "Parallel Alternating Direction Multiplier Decomposition of Convex Programs," *Journal of Optimization Theory and Applications*, vol. 80, no. 1, pp. 39–62, 1994.

[9] A. Beck and M. Teboulle, "Fast Gradient-Based Algorithms for Constrained Total Variation Image Denoising and Deblurring Problems," *IEEE Trans. Image Processing*, vol. 18, no. 11, pp. 2419–2434, 2009.

[10] I. W. Selesnick and M. A. T. Figueiredo, "Signal Restoration with Overcomplete Wavelet Transforms: Comparison of Analysis and Synthesis Priors," *Proceedings of SPIE (Wavelets XIII)*, vol. 7446, 2009.