

Parallel MR Image Reconstruction using Augmented Lagrangian Methods

Sathish Ramani*, *Member, IEEE*, and Jeffrey A. Fessler, *Fellow, IEEE*

Abstract—Magnetic resonance image (MRI) reconstruction using SENSitivity Encoding (SENSE) requires regularization to suppress noise and aliasing effects. Edge-preserving and sparsity-based regularization criteria can improve image quality, but they demand computation-intensive nonlinear optimization. In this paper, we present novel methods for regularized MRI reconstruction from undersampled sensitivity encoded data—SENSE-reconstruction—using the augmented Lagrangian (AL) framework for solving large-scale constrained optimization problems. We first formulate regularized SENSE-reconstruction as an unconstrained optimization task and then convert it to a set of (equivalent) constrained problems using *variable splitting*. We then attack these constrained versions in an AL framework using an *alternating minimization* method, leading to algorithms that can be implemented easily. The proposed methods are applicable to a general class of regularizers that includes popular edge-preserving (e.g., total-variation) and sparsity-promoting (e.g., ℓ_1 -norm of wavelet coefficients) criteria and combinations thereof. Numerical experiments with synthetic and *in-vivo* human data illustrate that the proposed AL algorithms converge faster than both general-purpose optimization algorithms such as nonlinear conjugate gradient (NCG) and state-of-the-art MFISTA method.

Index Terms—Parallel MRI, SENSE, Image Reconstruction, Regularization, Augmented Lagrangian

I. INTRODUCTION

Parallel MR imaging (pMRI) exploits spatial sensitivity of an array of receiver coils to reduce the number of required Fourier encoding steps, thereby accelerating MR scanning. SENSitivity Encoding (SENSE) [1], [2] is a popular pMRI technique where reconstruction is performed by solving a linear system that explicitly depends on the sensitivity maps of the coil array. While efficient reconstruction methods have been devised for SENSE with Cartesian [1], as well as non-Cartesian k -space trajectories [2], they inherently suffer from SNR degradation in the presence of noise [1] mainly due to k -space undersampling and instability arising from correlation in sensitivity maps [3].

Regularization is an attractive means of restoring stability in the reconstruction mechanism where prior information

can also be incorporated effectively [3]–[9]. Tikhonov-like quadratic regularization [3]–[6] leads to a closed-form solution (under a Gaussian noise model) that can be numerically implemented efficiently. However, with the advent of compressed sensing (CS) theory, sparsity-promoting regularization criteria (e.g., ℓ_1 -based regularization) have gained popularity in MRI [10]. The basic assumption underlying CS-MRI is that many MR images are inherently sparse in some transform domain and can be reconstructed with high accuracy from significantly undersampled k -space data by minimizing transform-domain sparsity-promoting regularization criteria subject to data-consistency. The CS framework is apt for pMRI [11] with undersampled data. This paper investigates the problem of regularized reconstruction from sensitivity encoded data—SENSE-reconstruction—using sparsity-promoting regularizers. We formulate regularized SENSE-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 regularization term, $\Psi(\mathbf{x})$, and a (negative) log-likelihood term corresponding to the noise model. For Ψ , we consider a general class of functionals that includes popular edge-preserving (e.g., total-variation) and sparsity-promoting (e.g., ℓ_1 -norm of wavelet coefficients) criteria and combinations thereof. Such regularization criteria are “non-smooth” (i.e., they may not be differentiable everywhere) and they require solving a nonlinear optimization problem using iterative algorithms.

This paper presents accelerated algorithms for regularized SENSE-reconstruction using the augmented Lagrangian (AL) formalism. The AL framework was originally developed for solving constrained optimization problems [12]; one combines the function to be minimized with a Lagrange multiplier term and a penalty term for the constraints, and minimizes it iteratively (while taking care to update the Lagrange parameters) to solve the original constrained problem. This combination overcomes the shortcomings of the Lagrange multiplier method and penalty-based methods for solving constrained problems [12]. To use the AL formalism for regularized SENSE-reconstruction, we first convert the unconstrained problem in to an equivalent constrained optimization problem using a technique called *variable splitting* where auxiliary variables take the place of linear transformations of \mathbf{x} in the cost function J . Then, we construct a corresponding AL function and minimize it alternatively with respect to one auxiliary variable at a time—this step forms the key ingredient as it decouples the minimization process and simplifies optimization. We investigate different variable-splitting approaches and

Copyright (c) 2010 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

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.

*Sathish Ramani and Jeffrey A. Fessler are with the Department of Electrical Engineering and Computer Science, University of Michigan, 1301 Beal Ave., Ann Arbor, MI 48109-2122, U.S.A. Email: {sramani, fessler}@umich.edu.

correspondingly design different AL algorithms for solving the original unconstrained SENSE-reconstruction problem. We also propose to use a diagonal weighting term in the AL formalism to induce suitable balance between various constraints because the matrix-elements associated with Fourier encoding and the sensitivity maps can be of different orders of magnitude in SENSE. The proposed AL algorithms are applicable for regularized SENSE-reconstruction from data acquired on arbitrary non-Cartesian k -space trajectories. Based on numerical experiments with synthetic and real data, we demonstrate that the proposed AL algorithms converge faster (to an actual solution of the original unconstrained regularized SENSE-reconstruction problem) compared to general-purpose optimization algorithms such as NCG (that has been applied for CS-(p)MRI in [10], [11]), and the recently proposed state-of-the-art Monotone Fast Iterative Shrinkage-Thresholding Algorithm (MFISTA) [13].

The paper is organized as follows. Section II formulates the regularized SENSE-reconstruction problem (with sparsity-based regularization) as an unconstrained optimization task. Next, we concentrate on the development of AL-based algorithms. First, Section III presents a quick overview of AL framework. Then, Section IV applies the AL formalism to regularized SENSE-reconstruction in detail. Here, we discuss various strategies for applying variable splitting and develop different AL algorithms for regularized SENSE-reconstruction. Section V is dedicated to numerical experiments and results. Section VI discusses possible extensions of the proposed AL methods to handle some variations of SENSE-reconstruction such as that proposed in [14]. Finally, we draw our conclusions in Section VII.

II. PROBLEM FORMULATION

We consider the discretized SENSE MR imaging model given by

$$\mathbf{d} = \mathbf{F} \mathbf{S} \mathbf{x} + \boldsymbol{\varepsilon}, \quad (1)$$

where \mathbf{x} is a $N \times 1$ column vector containing the samples of the unknown image to be reconstructed (e.g., a 2-D slice of a 3-D MRI volume), \mathbf{d} and $\boldsymbol{\varepsilon}$ are $ML \times 1$ column vectors corresponding to the data-samples from L coils and noise, respectively, \mathbf{S} is a $NL \times N$ matrix given by $\mathbf{S} = [\mathbf{S}_1^H \cdots \mathbf{S}_L^H]^H$, \mathbf{S}_l is a $N \times N$ (possibly complex) diagonal matrix corresponding to the sensitivity map of the l th coil, $1 \leq l \leq L$, $(\cdot)^H$ represents the Hermitian-transpose, \mathbf{F} is a $ML \times NL$ matrix given by $\mathbf{F} = \mathbf{I}_L \otimes \mathbf{F}_u$, \mathbf{F}_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}_u signifies the fact that the k -space may be undersampled to reduce scan time, i.e., $M \leq N$.

Given an estimate of the sensitivity maps \mathbf{S} , the SENSE-reconstruction problem is to find \mathbf{x} from data \mathbf{d} . Since regularization is an attractive means of reducing aliasing artifacts and the effect of noise in the reconstruction (by incorporating prior knowledge), we formulate the problem in a penalized-likelihood setting where the reconstruction is obtained by minimizing a cost criterion:

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

where \mathbf{K}_{ML} is the inverse of the $ML \times ML$ noise covariance matrix, $\|\mathbf{u}\|_{\mathbf{K}_{ML}}^2 = \mathbf{u}^H \mathbf{K}_{ML} \mathbf{u}$, and Ψ represents a suitable regularizer. We have included \mathbf{K}_{ML} in the data-fidelity term to account for the fact that noise from different coils may be correlated [1], [2]. Assuming that noise is wide-sense stationary and is correlated only over space (i.e., coils) and not over k -space, \mathbf{K}_{ML} can be written as $\mathbf{K}_{ML} = \mathbf{K}_s \otimes \mathbf{I}_M$, where \mathbf{K}_s is a $L \times L$ matrix that corresponds to the inverse of the covariance matrix of the spatial component of noise (from L coils).

The weighting matrix \mathbf{K}_{ML} can be eliminated from J in (2) by applying a noise-decorrelation procedure [2]: Since \mathbf{K}_s is generally positive definite, we write $\mathbf{K}_s = \tilde{\mathbf{K}}_s^H \tilde{\mathbf{K}}_s$, and $\mathbf{K}_{ML} = \tilde{\mathbf{K}}_{ML}^H \tilde{\mathbf{K}}_{ML}$, where $\tilde{\mathbf{K}}_{ML} = \tilde{\mathbf{K}}_s \otimes \mathbf{I}_M$. Then, because of the structures of $\tilde{\mathbf{K}}_{ML}$ and \mathbf{F} , we have that [2]

$$\begin{aligned} \tilde{\mathbf{K}}_{ML} \mathbf{F} &= (\tilde{\mathbf{K}}_s \otimes \mathbf{I}_M) \otimes (\mathbf{I}_L \otimes \mathbf{F}_u) \\ &= (\tilde{\mathbf{K}}_s \mathbf{I}_L \otimes \mathbf{I}_M \mathbf{F}_u) \\ &= (\mathbf{I}_L \tilde{\mathbf{K}}_s \otimes \mathbf{F}_u \mathbf{I}_N) \\ &= (\mathbf{I}_L \otimes \mathbf{F}_u) (\tilde{\mathbf{K}}_s \otimes \mathbf{I}_N) = \mathbf{F} \tilde{\mathbf{K}}_{NL}, \end{aligned}$$

where $\tilde{\mathbf{K}}_{NL} \triangleq \tilde{\mathbf{K}}_s \otimes \mathbf{I}_N$. Letting

$$\tilde{\mathbf{d}} \triangleq \tilde{\mathbf{K}}_{NL} \mathbf{d}, \quad (3)$$

$$\tilde{\mathbf{S}} \triangleq \tilde{\mathbf{K}}_{NL} \mathbf{S}, \quad (4)$$

we therefore get that

$$\frac{1}{2} \|\mathbf{d} - \mathbf{F} \mathbf{S} \mathbf{x}\|_{\mathbf{K}_{ML}}^2 = \frac{1}{2} \|\tilde{\mathbf{d}} - \mathbf{F} \tilde{\mathbf{S}} \mathbf{x}\|_2^2, \quad (5)$$

which is an equivalent unweighted data-fidelity term¹ with a new set of sensitivity maps $\tilde{\mathbf{S}}$ obtained by weighting the original sensitivity maps \mathbf{S} with $\tilde{\mathbf{K}}_{NL}$. In the sequel, we use the r.h.s. of (5) for data-fidelity and drop the $\tilde{\cdot}$ for ease of notation. In the numerical experiments, we used (3) and (4).

We consider sparsity-promoting regularization for Ψ based on the field of compressed sensing for MRI—CS-MRI [10], [11]. We focus on the ‘analysis form’ of the reconstruction problem where the regularization is a function of the unknown image \mathbf{x} . Specifically, we consider a general class of regularizers that use a sum of Q terms given by

$$\Psi(\mathbf{x}) = \sum_{q=1}^Q \lambda_q \sum_{n=1}^{N_q} \Phi_{qn} \left(\sum_{p=1}^{P_q} |[\mathbf{R}_{pq} \mathbf{x}]_n|^{m_q} \right), \quad (6)$$

where q indexes the regularization terms, the parameter $\lambda_q > 0$ controls the strength of the q th regularization term, and $[\mathbf{x}]_n$ or x_n represents the n th element of the vector \mathbf{x} . The $N_q \times N$ matrices \mathbf{R}_{pq} , $p = 1, \dots, P_q \forall q$, represent sparsifying operators. We focus on shift-invariant operators for \mathbf{R}_{pq} (e.g., tight frames, finite-differencing matrices), but the methods can be applied to shift-variant ones such as orthonormal wavelets with only minor modifications. Typically, $P_q \ll M, N \forall q$ (as seen in the examples below). We consider that the values of m_q and the choice of potential functions $\Phi_{qn} \forall q$ and n are such

¹The r.h.s. of (5) automatically includes the special case of $\mathbf{K}_{ML} = \mathbf{I}_{ML}$ with $\tilde{\mathbf{S}} = \mathbf{S}$ and $\tilde{\mathbf{d}} = \mathbf{d}$.

that Ψ is composed of non-quadratic convex regularization terms.

The general class of regularizers (6) includes popular sparsity-promoting regularization criteria such as

- (a) ℓ_1 -norm of wavelet coefficients: $Q = 1$, $P_1 = 1$, $m_1 = 1$, $\mathbf{R}_{11} = \mathbf{W}$ is a wavelet transform (orthonormal or a tight frame), and $\Phi_{1n}(x) = x$ where n indexes the rows of \mathbf{R}_{11} ,
- (b) discrete isotropic total-variation (TV) regularization [15]: $Q = 1$, $P_1 = 2$, $m_1 = 2$, \mathbf{R}_{11} and \mathbf{R}_{21} represent horizontal and vertical finite-differencing matrices, respectively, and $\Phi_{1n}(x) = \sqrt{x}$, where n indexes the rows of \mathbf{R}_{11} ,
- (c) discrete anisotropic total-variation (TV) regularization [15]: $Q = 1$, $P_1 = 2$, $m_1 = 1$, \mathbf{R}_{11} and \mathbf{R}_{21} represent horizontal and vertical finite-differencing matrices, respectively, and $\Phi_{1n}(x) = x$, where n indexes rows of \mathbf{R}_{11} .

The general form (6) also allows the use of a variety of potential functions for Φ . We consider such a generalization because combinations of wavelet-and TV-regularization have been reported to be preferable [10]. The proposed methods can be easily generalized for synthesis-based formulations [16].

The minimization in (2) is a non-trivial optimization task, even for only one regularization term. Although general purpose optimization techniques such as the nonlinear conjugate gradient (NCG) method or iteratively reweighted least squares can be applied to differentiable approximations of $\mathbf{P0}$, they may either be computation-intensive or exhibit slow convergence. This paper describes new techniques based on the augmented Lagrangian (AL) formalism that yield faster convergence per unit computation time.

The basic idea is to break down $\mathbf{P0}$ in to smaller tasks by introducing “artificial” constraints that are designed so that the sub-problems become decoupled and can be solved relatively rapidly [15], [17]–[20]. We first briefly review the AL method and then discuss some strategies for applying it to $\mathbf{P0}$.

III. CONSTRAINED OPTIMIZATION AND AUGMENTED LAGRANGIAN (AL) FORMALISM

Consider the following optimization problem with linear equality constraints:

$$\hat{\mathbf{u}} = \arg \min_{\mathbf{u} \in \Omega^N} f(\mathbf{u}) \text{ subject to } \mathbf{C}\mathbf{u} = \mathbf{b}, \quad (7)$$

where Ω is \mathbb{R} or \mathbb{C} , f is a real convex function, \mathbf{C} is a $M \times N$ (real or complex) matrix that specifies the constraint equations, and $\mathbf{b} \in \Omega^M$. In the augmented Lagrangian (AL) framework (also known as the multiplier method [12]), an AL function is first constructed for problem (7) as

$$\mathcal{L}(\mathbf{u}, \boldsymbol{\gamma}, \mu) = f(\mathbf{u}) + \boldsymbol{\gamma}^H (\mathbf{C}\mathbf{u} - \mathbf{b}) + \frac{\mu}{2} \|\mathbf{C}\mathbf{u} - \mathbf{b}\|_2^2, \quad (8)$$

where $\boldsymbol{\gamma} \in \Omega^M$ represents the vector of Lagrange multipliers, and the quadratic term on the r.h.s. of above equation is called

the “penalty” term² with penalty parameter³ $\mu > 0$. The AL scheme [12] for solving (7) alternates between minimizing $\mathcal{L}(\mathbf{u}, \boldsymbol{\gamma}, \mu)$ with respect to \mathbf{u} for a fixed $\boldsymbol{\gamma}$ and updating $\boldsymbol{\gamma}$, i.e.,

$$\mathbf{u}^{(j+1)} = \arg \min_{\mathbf{u}} \mathcal{L}(\mathbf{u}, \boldsymbol{\gamma}^{(j)}, \mu), \quad (9)$$

$$\boldsymbol{\gamma}^{(j+1)} = \boldsymbol{\gamma}^{(j)} + \mu(\mathbf{C}\mathbf{u}^{(j+1)} - \mathbf{b}), \quad (10)$$

until some stopping criterion is satisfied.

So-called “penalty methods” [12] correspond to the case where $\boldsymbol{\gamma} = \mathbf{0}$ and (9) is solved repeatedly while increasing $\mu \rightarrow \infty$. The AL scheme (9)-(10) also permits the use of increasing sequences of μ -values, but an important aspect of the AL scheme is that convergence may be guaranteed without the need for changing μ [12].

The AL scheme is also closely related to the Bregman iterations [15, Equations (2.6)-(2.8)] applied to problem (7):

$$\mathbf{u}^{(j+1)} = \arg \min_{\mathbf{u}} D_f(\mathbf{u}, \mathbf{u}^{(j)}, \boldsymbol{\rho}^{(j)}) + \frac{\mu}{2} \|\mathbf{C}\mathbf{u} - \mathbf{b}\|_2^2, \quad (11)$$

$$\boldsymbol{\rho}^{(j+1)} = \boldsymbol{\rho}^{(j)} - \mu \mathbf{C}^H (\mathbf{C}\mathbf{u}^{(j+1)} - \mathbf{b}), \quad (12)$$

where $D_f(\mathbf{u}, \mathbf{v}, \boldsymbol{\rho}) = f(\mathbf{u}) - f(\mathbf{v}) - \boldsymbol{\rho}^H (\mathbf{u} - \mathbf{v})$ is called the “Bregman distance” [15] and $\boldsymbol{\rho}$ is a $N \times 1$ vector in the subgradient of f at \mathbf{u} . The connection between AL method and Bregman iterations is readily established if $\boldsymbol{\rho} = -\mathbf{C}^H \boldsymbol{\gamma}$ [22]. Then, $D_f(\mathbf{u}, \mathbf{u}^{(j)}, \boldsymbol{\rho}^{(j)}) + \frac{\mu}{2} \|\mathbf{C}\mathbf{u} - \mathbf{b}\|_2^2$ is identical to $\mathcal{L}(\mathbf{u}, \boldsymbol{\gamma}^{(j)}, \mu)$ (up to constants irrelevant for optimization) and (9)-(10) become equivalent to (11)-(12) as noted in [22], [23].

The AL function \mathcal{L} in (8) can be rewritten by grouping together the terms involving $\mathbf{C}\mathbf{u} - \mathbf{b}$ as

$$\mathcal{L}(\mathbf{u}, \boldsymbol{\eta}, \mu) = f(\mathbf{u}) + \frac{\mu}{2} \|\mathbf{C}\mathbf{u} - \boldsymbol{\eta}\|_2^2 + C_{\boldsymbol{\gamma}}, \quad (13)$$

where $\boldsymbol{\eta} \triangleq \mathbf{b} - \frac{1}{\mu} \boldsymbol{\gamma}$, and $C_{\boldsymbol{\gamma}}$ is a constant independent of \mathbf{u} that we ignore henceforth. The parameter $\boldsymbol{\gamma}$ can then be replaced by $\boldsymbol{\eta}$ in (10) which results in the following version of AL algorithm for solving (7).

Algorithm AL

1. Select $\mathbf{u}^{(0)}$, $\boldsymbol{\eta}^{(0)}$, and $\mu > 0$; set $j = 0$
- Repeat**
2. $\mathbf{u}^{(j+1)} = \arg \min_{\mathbf{u}} \{f(\mathbf{u}) + \frac{\mu}{2} \|\mathbf{C}\mathbf{u} - \boldsymbol{\eta}^{(j)}\|_2^2\}$
3. $\boldsymbol{\eta}^{(j+1)} = \boldsymbol{\eta}^{(j)} - (\mathbf{C}\mathbf{u}^{(j+1)} - \mathbf{b})$
4. Set $j = j + 1$
- Until** stop-criterion is met

It has been shown in [15, Theorem 2.2] that the Bregman iterations (11)-(12)—equivalently, the AL algorithm under above mentioned conditions—converge to a solution of (7) whenever the minimization in (11)—in turn, Step 2 of the AL algorithm—is performed exactly. However, this step may be computationally expensive and is often replaced in practice by an inexact minimization [12], [15], [17]. Numerical evidence in [15] suggests that inexact minimizations can still be effective in the Bregman/AL scheme.

²A more general version of AL allows for the minimization of non-convex functions subject to nonlinear equality and/or inequality constraints with non-quadratic “penalty” terms [12].

³For non-convex problems, there may exist a lower bound on the possible values of μ for establishing convergence [12, Proposition 1], [21, page 519].

IV. PROPOSED AL ALGORITHMS FOR REGULARIZED SENSE-RECONSTRUCTION

Our strategy is to first transform the unconstrained problem **P0** into a constrained optimization task as follows. We replace linear transformations of \mathbf{x} ($\mathbf{F}\mathbf{S}\mathbf{x}$, and $\mathbf{R}_{pq}\mathbf{x}$) in J with a set of auxiliary variables $\{\mathbf{u}_l\}$. Then, we frame **P0** as a constrained problem where J is minimized as a function of $\{\mathbf{u}_l\}$ subject to the constraint that each auxiliary variable, \mathbf{u}_l , equals the respective linear transformations of \mathbf{x} . We handle the resulting constrained optimization task (that is equivalent to **P0**) in the AL framework described in Section III.

The technique of introducing auxiliary variables $\{\mathbf{u}_l\}$ is also known as *variable splitting*; it has been employed, for instance, in [15], [18]–[20] for image deconvolution, inpainting and CS-MRI with wavelets- and TV-based regularization in a Bregman/AL framework and in [17] for developing a fast penalty-based algorithm for TV image restoration. The purpose of variable splitting is to make the associated AL function \mathcal{L} amenable to alternating minimization methods [15], [17], [24]–[26] which may decouple the minimization of \mathcal{L} with respect to the auxiliary variables. This makes (9) easier to accomplish compared to directly solving the original unconstrained problem **P0**.

The splitting procedures used in [15], [17], [19] introduce auxiliary variables only for decoupling the effect of regularization. In this work, in addition to splitting the regularization, we also propose to use one or more auxiliary variables to separate the terms involving \mathbf{F} and \mathbf{S} (see Section IV-B). The AL-based techniques in [18], [20] also use auxiliary variables for the data-fidelity term, but they pertain to problems of the form

$$\min_{\mathbf{u}, \mathbf{v}} f(\mathbf{v}) \text{ subject to } \mathbf{v} = \mathbf{C}\mathbf{u},$$

where \mathbf{C} is a “tall”, i.e., block-column matrix and are not directly applicable to (7) with some instances of \mathbf{C} investigated in this paper (see Sections IV-B and VI-C). Furthermore, in general, different splitting mechanisms yield different algorithms as they attempt to solve constrained optimization problems (that are equivalent to **P0**) with different constraints. In this paper, we investigate two splitting schemes for **P0**, described below.

A. Splitting the Regularization Term

In the first form, we split the regularization term by introducing $\mathbf{u}_1 = \mathbf{R}\mathbf{x} \in \mathbb{C}^R$, where $\mathbf{R} = [\mathbf{R}_{11}^H \cdots \mathbf{R}_{P_0 Q}^H]^H$ and R is the number of rows in \mathbf{R} . This form is similar to the split-Bregman scheme proposed in [15, Section 4.2]. The resulting constrained formulation of **P0** is given by

$$\mathbf{P1}: \min_{\mathbf{u}_1, \mathbf{x}} J_1(\mathbf{x}, \mathbf{u}_1) \text{ subject to } \mathbf{u}_1 = \mathbf{R}\mathbf{x}$$

where

$$J_1(\mathbf{x}, \mathbf{u}_1) \triangleq \frac{1}{2} \|\mathbf{d} - \mathbf{F}\mathbf{S}\mathbf{x}\|_2^2 + \sum_{q=1}^Q \lambda_q \sum_{n=1}^{N_q} \Phi_{qn} \left(\sum_{p=1}^{P_q} |[\mathbf{u}_{1pq}]_n|^{m_q} \right)$$

and $\mathbf{u}_{1pq} = \mathbf{R}_{pq}\mathbf{x}$, $p = 1, \dots, P_q \forall q$. Problem **P1** can be written in the general form of (7) with

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{x} \end{bmatrix}, f(\mathbf{u}) = J_1(\mathbf{x}, \mathbf{u}_1), \mathbf{C} = [\mathbf{I}_R \quad -\mathbf{R}], \mathbf{b} = \mathbf{0}.$$

The associated AL function (8) is therefore

$$\mathcal{L}_1(\mathbf{u}, \boldsymbol{\gamma}_1, \mu) = J_1(\mathbf{x}, \mathbf{u}_1) + \boldsymbol{\gamma}_1^H \mathbf{C}\mathbf{u} + \frac{\mu}{2} \|\mathbf{C}\mathbf{u}\|_2^2.$$

The AL function \mathcal{L}_1 can be written in the form of (13) (ignoring irrelevant constants) as

$$\mathcal{L}_1(\mathbf{u}, \boldsymbol{\eta}_1, \mu) = J_1(\mathbf{x}, \mathbf{u}_1) + \frac{\mu}{2} \|\mathbf{C}\mathbf{u} - \boldsymbol{\eta}_1\|_2^2, \quad (14)$$

where $\boldsymbol{\eta}_1 = -\frac{1}{\mu}\boldsymbol{\gamma}_1$. Applying the **AL** algorithm to **P1** requires the joint minimization of \mathcal{L}_1 with respect to \mathbf{u}_1 and \mathbf{x} at Step 2. Since this can be computationally challenging, we apply an (inexact) alternating minimization method [15], [17], [19]: We alternatively minimize \mathcal{L}_1 with respect to one variable at a time while holding others constant. This decouples the individual updates of \mathbf{u}_1 and \mathbf{x} and simplifies the optimization task. Specifically, at the j th iteration, we perform the following individual minimizations, taking care to use updated variables for subsequent minimizations [15], [17]:

$$\mathbf{u}_1^{(j+1)} = \arg \min_{\mathbf{u}_1} \mathcal{L}_1(\mathbf{u}_1, \mathbf{x}^{(j)}, \boldsymbol{\eta}_1^{(j)}, \mu), \quad (15)$$

$$\mathbf{x}^{(j+1)} = \arg \min_{\mathbf{x}} \mathcal{L}_1(\mathbf{u}_1^{(j+1)}, \mathbf{x}, \boldsymbol{\eta}_1^{(j)}, \mu). \quad (16)$$

1) *Minimization with respect to \mathbf{x}* : The minimization in (16) is straightforward since the associated cost function is quadratic. Ignoring irrelevant constants, we get that

$$\begin{aligned} \mathbf{x}^{(j+1)} &= \arg \min_{\mathbf{x}} \left\{ \frac{1}{2} \|\mathbf{d} - \mathbf{F}\mathbf{S}\mathbf{x}\|_2^2 + \frac{\mu}{2} \|\mathbf{u}_1^{(j+1)} - \mathbf{R}\mathbf{x} - \boldsymbol{\eta}_1^{(j)}\|_2^2 \right\} \\ &= \mathbf{G}_\mu^{-1} [\mathbf{S}^H \mathbf{F}^H \mathbf{d} + \mu \mathbf{R}^H (\mathbf{u}_1^{(j+1)} - \boldsymbol{\eta}_1^{(j)})], \end{aligned} \quad (17)$$

where⁴

$$\mathbf{G}_\mu = \mathbf{S}^H \mathbf{F}^H \mathbf{F} \mathbf{S} + \mu \mathbf{R}^H \mathbf{R}. \quad (18)$$

Although (17) is an analytical solution, computing \mathbf{G}_μ^{-1} is impractical for large N . Therefore, we apply a few iterations of the conjugate-gradient (CG) algorithm with *warm starting*, i.e., the CG algorithm is initialized with the estimated \mathbf{x} from the previous **AL** iteration.

2) *Minimization with respect to \mathbf{u}_1* : Writing out (15) explicitly (ignoring constants independent of \mathbf{u}_1), we have that

$$\mathbf{u}_1^{(j+1)} = \arg \min_{\mathbf{u}_1} \left\{ \sum_{q=1}^Q \lambda_q \sum_{n=1}^{N_q} \Phi_{qn} \left(\sum_{p=1}^{P_q} |[\mathbf{u}_{1pq}]_n|^{m_q} \right) + \frac{\mu}{2} \|\mathbf{u}_1 - \mathbf{R}\mathbf{x}^{(j)} - \boldsymbol{\eta}_1^{(j)}\|_2^2 \right\}. \quad (19)$$

While (19) is a large-scale problem by itself, the splitting variable \mathbf{u}_1 decouples the different regularization terms so that (19) can be decomposed into smaller minimization tasks as follows. Let $\mathbf{r}^{(j)} = \mathbf{R}\mathbf{x}^{(j)}$; for each q and n , we collect $\mathbf{v}_{qn} =$

⁴We design the regularization Ψ such that the non-trivial null-spaces of $\mathbf{R}^H \mathbf{R}$ and $\mathbf{S}^H \mathbf{F}^H \mathbf{F} \mathbf{S}$ are disjoint. Then, \mathbf{G}_μ is non-singular for $\mu > 0$.

$\{u_{1pqm}\}_{p=1}^{P_q}$, $\boldsymbol{\varrho}_{qn}^{(j)} = \{r_{pqm}^{(j)}\}_{p=1}^{P_q}$, and $\boldsymbol{\beta}_{qn}^{(j)} = \{\eta_{1pqm}^{(j)}\}_{p=1}^{P_q}$, so that \mathbf{v}_{qn} , $\boldsymbol{\varrho}_{qn}^{(j)}$, $\boldsymbol{\beta}_{qn}^{(j)} \in \mathbb{C}^{P_q}$. Then, (19) separates for each q and n as

$$\mathbf{v}_{qn}^{(j+1)} = \arg \min_{\mathbf{v}_{qn}} \left\{ \begin{array}{l} \frac{\lambda_q}{\mu} \Phi_{qn} \left(\|\mathbf{v}_{qn}\|_{m_q}^{m_q} \right) \\ + \frac{1}{2} \|\mathbf{v}_{qn} - (\boldsymbol{\varrho}_{qn}^{(j)} + \boldsymbol{\beta}_{qn}^{(j)})\|_2^2 \end{array} \right\}. \quad (20)$$

This is basically a P_q -dimensional denoising problem with $\boldsymbol{\varrho}_{qn}^{(j)} + \boldsymbol{\beta}_{qn}^{(j)}$ playing the role of the data and where $\|\cdot\|_p$ denotes the ℓ_p -norm. Often (20) has a closed-form solution as discussed below. Otherwise, a gradient-descent-based algorithm such as NCG with warm starting can be applied for obtaining a partial update for $\mathbf{v}_{qn}^{(j+1)}$. Before proceeding, it is useful to compute the gradient of the cost function in (20). Ignoring the indices q and n and setting the gradient of the cost function in (20) to zero, we get for $\mathbf{v} \neq \mathbf{0}$ that

$$(\boldsymbol{\Theta}(\mathbf{v}) + \mathbf{I}_P)\mathbf{v}|_{\mathbf{v}=\mathbf{v}^{(j+1)}} = \boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}, \quad (21)$$

where

$$\boldsymbol{\Theta}(\mathbf{v}) \triangleq \text{diag} \left\{ \frac{\lambda m}{\mu} \Phi'(\|\mathbf{v}\|_m^m) |v_k|^{m-2} \right\}, \quad (22)$$

and Φ' is the first derivative of Φ , and v_k is the k th component of \mathbf{v} . The main obstacles to obtaining a direct solution of (20) are the coupling introduced between different components of \mathbf{v} , i.e., $\Phi'(\|\mathbf{v}\|_m^m)$, and the presence of the $|v_k|^{m-2}$ in $\boldsymbol{\Theta}(\mathbf{v})$. Below we analyze some special cases of practical interest where this problem can be circumvented to obtain simple solutions.

Case of ℓ_1 -regularization: For ℓ_1 -type regularization in (6) we set $m = 1$, $\Phi(x) = x$. Consequently, (21) further decouples in terms of the components of \mathbf{v} as

$$\left(\frac{\lambda}{\mu|v_k|} + 1 \right) v_k = \varrho_k^{(j)} + \beta_k^{(j)},$$

where $v_k \neq 0$, $k = 1, 2, \dots, P$. The minimizer of (20) in this case is given by the shrinkage rule [27]

$$v_k^{(j+1)} = \text{shrink} \left\{ \varrho_k^{(j)} + \beta_k^{(j)}, \frac{\lambda}{\mu} \right\} \quad \forall k,$$

where $\text{shrink}\{d, \lambda\} = \frac{d}{|d|} \max\{|d| - \lambda, 0\}$.

Case of $P = 1$: In this case, (20) reduces to 1-D minimization that can be easily achieved numerically for a general Φ or analytically for $m = 1$ and some specific instances of Φ listed in [28, Section 4].

Case of $m = 2$ and a general Φ : For $m = 2$, the solution of (20) is in general determined by a vector-shrinkage rule as explained below. Setting $m = 2$ in (21), we get that

$$\underbrace{\left(\frac{2\lambda}{\mu} \Phi'(\|\mathbf{v}\|_2^2) + 1 \right)}_{\chi(\|\mathbf{v}\|_2)} \mathbf{v} = \boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}. \quad (23)$$

The bracketed term on the l.h.s. is a non-negative scalar (cf. Φ is non-decreasing), so that (23) corresponds to shrinking $\boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}$ by an amount prescribed by $\chi(\|\mathbf{v}\|_2)$ for $\mathbf{v} \neq \mathbf{0}$. The exact value of $\chi(\|\mathbf{v}\|_2)$ depends on \mathbf{v} and in general, there

is no closed form solution to (23). Nevertheless, for given λ and μ values, (23) can be solved numerically⁵ by using a look-up table for Φ' to find the value for the shrinkage factor $\chi(\|\mathbf{v}\|_2)$ such that (23) is satisfied.

Case of TV-type regularization: To obtain a TV-type regularization in (6) we set $m = 2$ and $\Phi(x) = \sqrt{x}$. Correspondingly, (21) becomes

$$\chi(\|\mathbf{v}\|_2)\mathbf{v} = \boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}, \quad (24)$$

where $\chi(\|\mathbf{v}\|_2) = \left(\frac{\lambda}{\mu\|\mathbf{v}\|_2} + 1 \right)$ for $\mathbf{v} \neq \mathbf{0}$. In this case, an exact value for χ can be found as shown below. Taking ℓ_2 -norm of the vectors on both sides of (24) and manipulating, we get that $\|\mathbf{v}\|_2|_{\mathbf{v}=\mathbf{v}^{(j+1)}} = \|\boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}\|_2 - \frac{\lambda}{\mu}$, and

$$\chi(\|\mathbf{v}^{(j+1)}\|_2) = \left(\frac{\|\boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}\|_2}{\|\boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}\|_2 - \frac{\lambda}{\mu}} \right),$$

which leads to the following vector-shrinkage rule [15], [17], [22]

$$\mathbf{v}^{(j+1)} = \text{shrink}_{\text{vec}} \left\{ \boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}, \frac{\lambda}{\mu} \right\},$$

where $\text{shrink}_{\text{vec}}\{\mathbf{d}, \lambda\} \triangleq \frac{\mathbf{d}}{\|\mathbf{d}\|_2} \max\{\|\mathbf{d}\|_2 - \lambda, 0\}$. It is also possible to derive closed-form solutions of (20) for $m = 2$ for some instances of Φ listed in [28, Section 4]. In summary, the minimization problem (20) is fairly simple and fast typically.

3) *AL Algorithm for Problem P1:* Combining the results from Sections IV-A1 and IV-A2, we now present the first AL algorithm (that is similar to the split-Bregman scheme [15]) for solving the constrained optimization problem **P1**, formulated as a tractable alternative to the original unconstrained problem **P0**.

AL-P1: AL Algorithm for solving problem P1

1. Select $\mathbf{x}^{(0)}$ and $\mu > 0$
2. Precompute $\mathbf{S}^H \mathbf{F}^H \mathbf{d}$; set $\boldsymbol{\eta}_1^{(0)} = \mathbf{0}$ and $j = 0$
- Repeat:**
3. Obtain an update $\mathbf{u}_1^{(j+1)}$ using an appropriate technique as described in Section IV-A2
4. Obtain an update $\mathbf{x}^{(j+1)}$ by running few CG iterations on (17)
5. $\boldsymbol{\eta}_1^{(j+1)} = \boldsymbol{\eta}_1^{(j)} - (\mathbf{u}_1^{(j+1)} - \mathbf{R}\mathbf{x}^{(j+1)})$
6. Set $j = j + 1$
- Until** stop-criterion is met

The most complex step of this algorithm is using CG to solve (17). We now present an alternative algorithm that simplifies computation further.

B. Splitting the Fourier Encoding and Spatial Components in the Data-Fidelity Term

Since the data-fidelity term is composed of components (**S** and **F**) that act on the unknown image in different domains (spatial and k -space, respectively) it is natural to introduce

⁵Taking the ℓ_2 -norm of the vectors on both sides of (23), we see that (23) entails solving a 1-D problem of the form $(\lambda\Phi'(x^2) + 1)x = d$, for x .

auxiliary variables to split these two components. Specifically, we now consider the constrained problem

$$\mathbf{P2:} \quad \min_{\mathbf{u}_0, \mathbf{u}_1, \mathbf{u}_2, \mathbf{x}} J_2(\mathbf{u}_0, \mathbf{u}_1) \text{ subject to} \\ \mathbf{u}_0 = \mathbf{S}\mathbf{x}, \mathbf{u}_1 = \mathbf{R}\mathbf{u}_2 \text{ and } \mathbf{u}_2 = \mathbf{x},$$

where $\mathbf{u}_0 \in \mathbb{C}^{NL}$, $\mathbf{u}_1 \in \mathbb{C}^R$, $\mathbf{u}_2 \in \mathbb{C}^N$, and

$$J_2(\mathbf{u}_0, \mathbf{u}_1) \triangleq \frac{1}{2} \|\mathbf{d} - \mathbf{F}\mathbf{u}_0\|_2^2 \\ + \sum_{q=1}^Q \lambda_q \sum_{n=1}^{N_q} \Phi_{qn} \left(\sum_{p=1}^{P_q} |[\mathbf{u}_{1pq}]_n|^{m_q} \right).$$

Clearly, **P2** is equivalent to **P0**. The new variable \mathbf{u}_2 simplifies the implementation by decoupling \mathbf{u}_0 and \mathbf{u}_1 . In terms of the general AL formulation (7), **P2** is written as

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_0 \\ \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{x} \end{bmatrix}, f(\mathbf{u}) = J_2(\mathbf{u}_0, \mathbf{u}_1), \mathbf{C} = \mathbf{\Lambda}\mathbf{B}, \mathbf{b} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

where

$$\mathbf{\Lambda} = \begin{bmatrix} \mathbf{I}_{NL} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sqrt{\nu_1} \mathbf{I}_R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \sqrt{\nu_2} \mathbf{I}_N \end{bmatrix}, \mathbf{B} = \begin{bmatrix} \mathbf{I}_{NL} & \mathbf{0} & \mathbf{0} & -\mathbf{S} \\ \mathbf{0} & \mathbf{I}_R & -\mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_N & -\mathbf{I}_N \end{bmatrix}.$$

We have introduced a diagonal weighting matrix $\mathbf{\Lambda}$ in the constraint equation whose purpose will be explained below. Using $\mathbf{\Lambda}$ does not alter problem **P2** as long as $\nu_{1,2} > 0$. The associated AL function (8) is given by

$$\mathcal{L}_2(\mathbf{u}, \boldsymbol{\gamma}_2, \mu) = J_2(\mathbf{u}_0, \mathbf{u}_1) + \boldsymbol{\gamma}_1^H \mathbf{\Lambda} \mathbf{B} \mathbf{u} + \frac{\mu}{2} \|\mathbf{B} \mathbf{u}\|_{\mathbf{\Lambda}^2}^2,$$

where $\boldsymbol{\gamma}_2 = [\boldsymbol{\gamma}_{20}^H \ \boldsymbol{\gamma}_{21}^H \ \boldsymbol{\gamma}_{22}^H]^H$, one component for each row of \mathbf{B} . Then, we write \mathcal{L}_2 in the form of (13) (without irrelevant constants) as

$$\mathcal{L}_2(\mathbf{u}, \boldsymbol{\eta}_2, \mu) = J_2(\mathbf{u}_0, \mathbf{u}_1) + \frac{\mu}{2} \|\mathbf{B} \mathbf{u} - \boldsymbol{\eta}_2\|_{\mathbf{\Lambda}^2}^2, \quad (25)$$

where $\boldsymbol{\eta}_2 \triangleq [\boldsymbol{\eta}_{20}^H \ \boldsymbol{\eta}_{21}^H \ \boldsymbol{\eta}_{22}^H]^H = -\frac{1}{\mu} \mathbf{\Lambda}^{-1} \boldsymbol{\gamma}_2$. From (25), we see that $\mathbf{\Lambda}$ specifies the relative influence of the constraints individually while μ determines the overall influence of the constraints on \mathcal{L}_2 . Note again that the final solution of **P2** does not depend on any of μ , ν_1 or ν_2 .

We again apply alternating minimization to (25) (ignoring irrelevant constants) to obtain the following sub-problems:

$$\mathbf{u}_0^{(j+1)} = \arg \min_{\mathbf{u}_0} \left\{ \frac{1}{2} \|\mathbf{d} - \mathbf{F}\mathbf{u}_0\|_2^2 + \frac{\mu}{2} \|\mathbf{u}_0 - \mathbf{S}\mathbf{x}^{(j)} - \boldsymbol{\eta}_{20}^{(j)}\|_2^2 \right\}, \quad (26)$$

$$\mathbf{u}_1^{(j+1)} = \arg \min_{\mathbf{u}_1} \left\{ \sum_{q=1}^Q \lambda_q \sum_{n=1}^{N_q} \Phi_{qn} \left(\sum_{p=1}^{P_q} |[\mathbf{u}_{1pq}]_n|^{m_q} \right) + \frac{\mu\nu_1}{2} \|\mathbf{u}_1 - \mathbf{R}\mathbf{u}_2^{(j)} - \boldsymbol{\eta}_{21}^{(j)}\|_2^2 \right\}, \quad (27)$$

$$\mathbf{u}_2^{(j+1)} = \arg \min_{\mathbf{u}_2} \left\{ \frac{\mu\nu_2}{2} \|\mathbf{u}_1^{(j+1)} - \mathbf{R}\mathbf{u}_2 - \boldsymbol{\eta}_{21}^{(j)}\|_2^2 + \frac{\mu\nu_2}{2} \|\mathbf{u}_2 - \mathbf{x}^{(j)} - \boldsymbol{\eta}_{22}^{(j)}\|_2^2 \right\}, \quad (28)$$

$$\mathbf{x}^{(j+1)} = \arg \min_{\mathbf{x}} \left\{ \frac{\mu}{2} \|\mathbf{u}_0^{(j+1)} - \mathbf{S}\mathbf{x} - \boldsymbol{\eta}_{20}^{(j)}\|_2^2 + \frac{\mu\nu_2}{2} \|\mathbf{u}_2^{(j+1)} - \mathbf{x} - \boldsymbol{\eta}_{22}^{(j)}\|_2^2 \right\}. \quad (29)$$

The minimization in (27) is exactly same as the one in (19) except that we now have $\mathbf{R}\mathbf{u}_2$ instead of $\mathbf{R}\mathbf{x}$ in the quadratic part of the cost. Therefore, we apply the techniques described in Section IV-A2 to solve (27).

1) *Minimization with respect to $\mathbf{u}_{0,2}$ and \mathbf{x}* : The cost functions in (26) and (28)-(29) are all quadratic and thus have closed-form solutions as follows:

$$\mathbf{u}_0^{(j+1)} = \mathbf{H}_\mu^{-1} [\mathbf{F}^H \mathbf{d} + \mu(\mathbf{S}\mathbf{x}^{(j)} + \boldsymbol{\eta}_{20}^{(j)})], \quad (30)$$

$$\mathbf{u}_2^{(j+1)} = \mathbf{H}_{\nu_1\nu_2}^{-1} \begin{bmatrix} \mathbf{R}^H (\mathbf{u}_1^{(j+1)} - \boldsymbol{\eta}_{21}^{(j)}) \\ + \frac{\nu_2}{\nu_1} (\mathbf{x}^{(j)} + \boldsymbol{\eta}_{22}^{(j)}) \end{bmatrix}, \quad (31)$$

$$\mathbf{x}^{(j+1)} = \mathbf{H}_{\nu_2}^{-1} \begin{bmatrix} \mathbf{S}^H (\mathbf{u}_0^{(j+1)} - \boldsymbol{\eta}_{20}^{(j)}) \\ + \nu_2 (\mathbf{u}_2^{(j+1)} - \boldsymbol{\eta}_{22}^{(j)}) \end{bmatrix}, \quad (32)$$

where

$$\mathbf{H}_\mu = \mathbf{F}^H \mathbf{F} + \mu \mathbf{I}_{NL}, \quad (33)$$

$$\mathbf{H}_{\nu_1\nu_2} = \mathbf{R}^H \mathbf{R} + \frac{\nu_2}{\nu_1} \mathbf{I}_N, \quad (34)$$

$$\mathbf{H}_{\nu_2} = \mathbf{S}^H \mathbf{S} + \nu_2 \mathbf{I}_N. \quad (35)$$

We show below that these matrices can be inverted efficiently thereby avoiding the more difficult inverse \mathbf{G}_μ^{-1} in (17). We have proposed using $\mathbf{\Lambda}$ to ensure suitable balance between the various constraints (equivalently, the block-rows of \mathbf{B}) since the block-rows of \mathbf{B} may be of different orders of magnitude. We can adjust $\nu_{1,2}$ to regulate the condition numbers of $\mathbf{H}_{\nu_1\nu_2}$, and \mathbf{H}_{ν_2} to ensure stability of the inverses in (31)-(32). Using general positive definite diagonal matrices in place of weighted identity matrices inside $\mathbf{\Lambda}$ is possible but would complicate the structure of the matrices $\mathbf{H}_{\nu_1\nu_2}$, and \mathbf{H}_{ν_2} in (34)-(35), respectively.

2) *Implementing the Matrix Inverses*: When the k -space samples lie on a Cartesian grid, \mathbf{F}_u corresponds to a sub-sampled DFT matrix in which case we solve (30) exactly using FFTs. For non-Cartesian k -space trajectories, computing $\mathbf{u}_0^{(j+1)}$ requires an iterative method. For example, a CG-solver (with warm starting) that implements products with $\mathbf{F}^H \mathbf{F}$ using gridding-based techniques [29] can be used for (30). Alternatively, we can exploit the special structure of $\mathbf{F}^H \mathbf{F}$ (of size $NL \times NL$) to implement (30) using the technique proposed in [30]. We have that

$$\mathbf{F}^H \mathbf{F} = \mathbf{Z}^H \mathbf{Q} \mathbf{Z}, \quad (36)$$

where \mathbf{Z} is a $2NL \times NL$ zero-padding matrix and \mathbf{Q} is a $2NL \times 2NL$ circulant matrix [31]. Then, we write \mathbf{H}_μ as

$$\mathbf{H}_\mu = \left(\mathbf{Z}^H \mathbf{Q}_1 \mathbf{Z} + \frac{\mu}{2} \mathbf{I}_{NL} \right),$$

where $\mathbf{Q}_1 = \mathbf{Q} + \frac{\mu}{2} \mathbf{I}_{2NL}$. We have split the factor $\mu \mathbf{I}_{NL}$ in \mathbf{H}_μ because \mathbf{Q} may have a non-trivial null-space and therefore may not be invertible. Letting \mathbf{w} denote the quantity within the

brackets on the r.h.s. of (30), we apply the Sherman-Morrison-Woodbury matrix inversion lemma (MIL) to \mathbf{H}_μ^{-1} in (30) and obtain

$$\mathbf{u}_0^{(j+1)} = \mathbf{H}_\mu^{-1} \mathbf{w} = \frac{2}{\mu} \mathbf{w} - \frac{4}{\mu^2} \mathbf{Z}^H \boldsymbol{\tau}, \quad (37)$$

where $\boldsymbol{\tau}$ must be obtained by solving

$$\left(\mathbf{Q}_1^{-1} + \frac{2}{\mu} \mathbf{Z} \mathbf{Z}^H \right) \boldsymbol{\tau} = \mathbf{Z} \mathbf{w}. \quad (38)$$

Since \mathbf{Q}_1 is circulant and $\mathbf{Z} \mathbf{Z}^H$ is a diagonal matrix containing either ones or zeros (due to the structure of \mathbf{Z}) [30], we use a circulant preconditioner of the form $(\mathbf{Q}_1^{-1} + \alpha \mathbf{I}_{2NL})^{-1}$ (with $\alpha \approx 0$) to quickly solve (38) using the CG algorithm. The advantage here is that the matrices in the l.h.s. of (38) and the preconditioner are either circulant or diagonal, which simplifies CG-implementation.

When the regularization matrices, \mathbf{R}_{pq} , $p = 1, \dots, P_q \forall q$, are shift-invariant (or circulant), $\mathbf{R}^H \mathbf{R}$ is also shift-invariant. Then, we compute $\mathbf{H}_{\nu_1 \nu_2}^{-1}$ efficiently using FFTs. In the case where a \mathbf{R}_{pq} is not shift-invariant (e.g., an orthonormal wavelet transform), we apply a few CG iterations with warm-starting to solve (31). Finally, since $\mathbf{S}^H \mathbf{S}$ is diagonal, we see that \mathbf{H}_{ν_2} is also diagonal and is therefore easily inverted.

Splitting the k -space and spatial (i.e., \mathbf{F} and \mathbf{S} , respectively) components in the data-fidelity term has led to separate matrix inverses— \mathbf{H}_μ^{-1} and $\mathbf{H}_{\nu_2}^{-1}$ involving the components $\mathbf{F}^H \mathbf{F}$ and $\mathbf{S}^H \mathbf{S}$, respectively. Without \mathbf{u}_0 , one would have ended up with a term $\mathbf{S}^H \mathbf{F}^H \mathbf{F} \mathbf{S}$ (as in \mathbf{G}_μ) that is more difficult to handle using MIL compared to $\mathbf{F}^H \mathbf{F}$. Using \mathbf{u}_2 decouples the terms $\mathbf{R}^H \mathbf{R}$ and $\mathbf{S}^H \mathbf{S}$, thereby replacing a numerically intractable matrix inverse of the form $(\mathbf{S}^H \mathbf{S} + \alpha \mathbf{R}^H \mathbf{R})^{-1}$ with tractable ones such as $\mathbf{H}_{\nu_1 \nu_2}^{-1}$ and $\mathbf{H}_{\nu_2}^{-1}$.

3) *AL Algorithm for Problem P2*: Combining the results from Sections IV-B1 and IV-B2, we present our second AL algorithm that solves problem P2, and thus P0.

AL-P2: AL Algorithm for solving problem P2

1. Select $\mathbf{x}^{(0)}$, $\mathbf{u}_2^{(0)} = \mathbf{x}^{(0)}$, $\nu_{1,2} > 0$, and $\mu > 0$
2. Precompute $\mathbf{F}^H \mathbf{d}$; set $\boldsymbol{\eta}_{20,21,22}^{(0)} = \mathbf{0}$ and $j = 0$
- Repeat:**
3. Compute $\mathbf{u}_0^{(j+1)}$ from (30) using FFTs on (37)
4. Compute $\mathbf{u}_1^{(j+1)}$ using an appropriate technique as described in Section IV-A2 for problem (27)
5. Compute $\mathbf{u}_2^{(j+1)}$ using (31)
6. Compute $\mathbf{x}^{(j+1)}$ using (32)
7. $\boldsymbol{\eta}_{20}^{(j+1)} = \boldsymbol{\eta}_{20}^{(j)} - (\mathbf{u}_0^{(j+1)} - \mathbf{S} \mathbf{x}^{(j+1)})$
8. $\boldsymbol{\eta}_{21}^{(j+1)} = \boldsymbol{\eta}_{21}^{(j)} - (\mathbf{u}_1^{(j+1)} - \mathbf{R} \mathbf{u}_2^{(j+1)})$
9. $\boldsymbol{\eta}_{22}^{(j+1)} = \boldsymbol{\eta}_{22}^{(j)} - (\mathbf{u}_2^{(j+1)} - \mathbf{x}^{(j+1)})$
10. Set $j = j + 1$
- Until** stop-criterion is met

With the possible exception of Steps 3 and 4, all updates in **AL-P2** are exact (for circulant $\{\mathbf{R}_{pq}\}$) unlike **AL-P1** because of the way we split the variables in **P2**.

Although Steps 2-4 of **AL-P1** and Steps 2-6 of **AL-P2** do not exactly accomplish Step 2 of **AL**, we found in our experiments that both **AL-P1** and **AL-P2** work well, corroborating the numerical evidence from [15].

TABLE I
COMPUTATION TIME PER OUTER ITERATION OF VARIOUS ALGORITHMS
FOR THE EXPERIMENTS IN SECTION V

Algorithm	Time Taken (in seconds)	
	(Section V-B)	(Section V-C)
AL-P1-4	0.21	0.17
AL-P1-6	0.27	0.22
AL-P1-10	0.35	0.30
AL-P2	0.15	0.12
NCG-1	0.21	0.17
NCG-5	0.30	0.26
MFISTA-1	0.22	0.18
MFISTA-5	0.53	0.43
MFISTA-20	1.51	1.18

C. Choosing μ - and ν -values for the **AL** algorithms

Although μ - and ν -values do not affect the final solution to **P0**, they can affect the convergence rate of **AL-P1** and **AL-P2**. For **AL-P2**, we set the parameters μ , ν_1 , and ν_2 so as to achieve condition numbers— $\kappa(\mathbf{H}_\mu)$, $\kappa(\mathbf{H}_{\nu_1 \nu_2})$, and $\kappa(\mathbf{H}_{\nu_2})$ of \mathbf{H}_μ , $\mathbf{H}_{\nu_1 \nu_2}$, and \mathbf{H}_{ν_2} , respectively—that result in fast convergence of the algorithm. Because of the presence of identity matrices in (33)-(35), $\kappa(\mathbf{H}_\mu)$, $\kappa(\mathbf{H}_{\nu_1 \nu_2})$, and $\kappa(\mathbf{H}_{\nu_2})$ are decreasing functions of μ , $\frac{\nu_2}{\nu_1}$, and ν_2 , respectively. Choosing μ such that $\kappa(\mathbf{H}_\mu) \rightarrow 1$ would require a large μ and accordingly, the influence of self-adjoint component $\mathbf{F}^H \mathbf{F}$ in \mathbf{H}_μ diminishes— \mathbf{H}_μ becomes “over-regularized”; we observed in our experiments that this phenomenon would result in slow convergence of **AL-P2**. On the other hand, taking $\mu \rightarrow 0$ would increase $\kappa(\mathbf{H}_\mu)$ making \mathbf{H}_μ^{-1} numerically unstable (because $\mathbf{F}^H \mathbf{F}$ may have a non-trivial null-space). The same trend also applies to $\kappa(\mathbf{H}_{\nu_1 \nu_2})$ and $\kappa(\mathbf{H}_{\nu_2})$ as functions of $\frac{\nu_2}{\nu_1}$ and ν_2 , respectively. 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 **AL-P2** in all our experiments.

In the case of **AL-P1**, the components $\mathbf{S}^H \mathbf{F}^H \mathbf{F} \mathbf{S}$ and $\mathbf{R}^H \mathbf{R}$ balance each other in preventing \mathbf{G}_μ (18) from having a non-trivial null-space—the condition number $\kappa(\mathbf{G}_\mu)$ of \mathbf{G}_μ therefore exhibits a minimum for some $\mu_{\min} > 0$: $\mu_{\min} = \arg \min_{\mu} \kappa(\mathbf{G}_\mu)$. It was suggested in [15] that μ_{\min} can be used for split-Bregman-like schemes such as **AL-P1** for ensuring quick convergence of the CG algorithm applied to (17) (Step 4 of **AL-P1**). However, we observed that selecting $\mu = \mu_{\min}$ did not consistently yield⁶ fast convergence of the **AL-P1** algorithm in our experiments (see Section VI-B). So, we resorted to a manual selection of μ for **AL-P1** for reconstructing one slice of a 3-D MRI volume, but applied the same μ -value for reconstructing other slices.

V. EXPERIMENTS

A. Experimental Setup

In all our experiments, we considered k -space samples on a Cartesian⁷ grid, so \mathbf{F}_u corresponds to an undersampled version of the DFT matrix. We used Poisson-disk-based sub-sampling

⁶A possible explanation for this phenomenon is presented in a supplementary downloadable material available, along with the paper, at <http://ieeexplore.ieee.org>.

⁷The proposed algorithms also apply to non-Cartesian k -space trajectories.

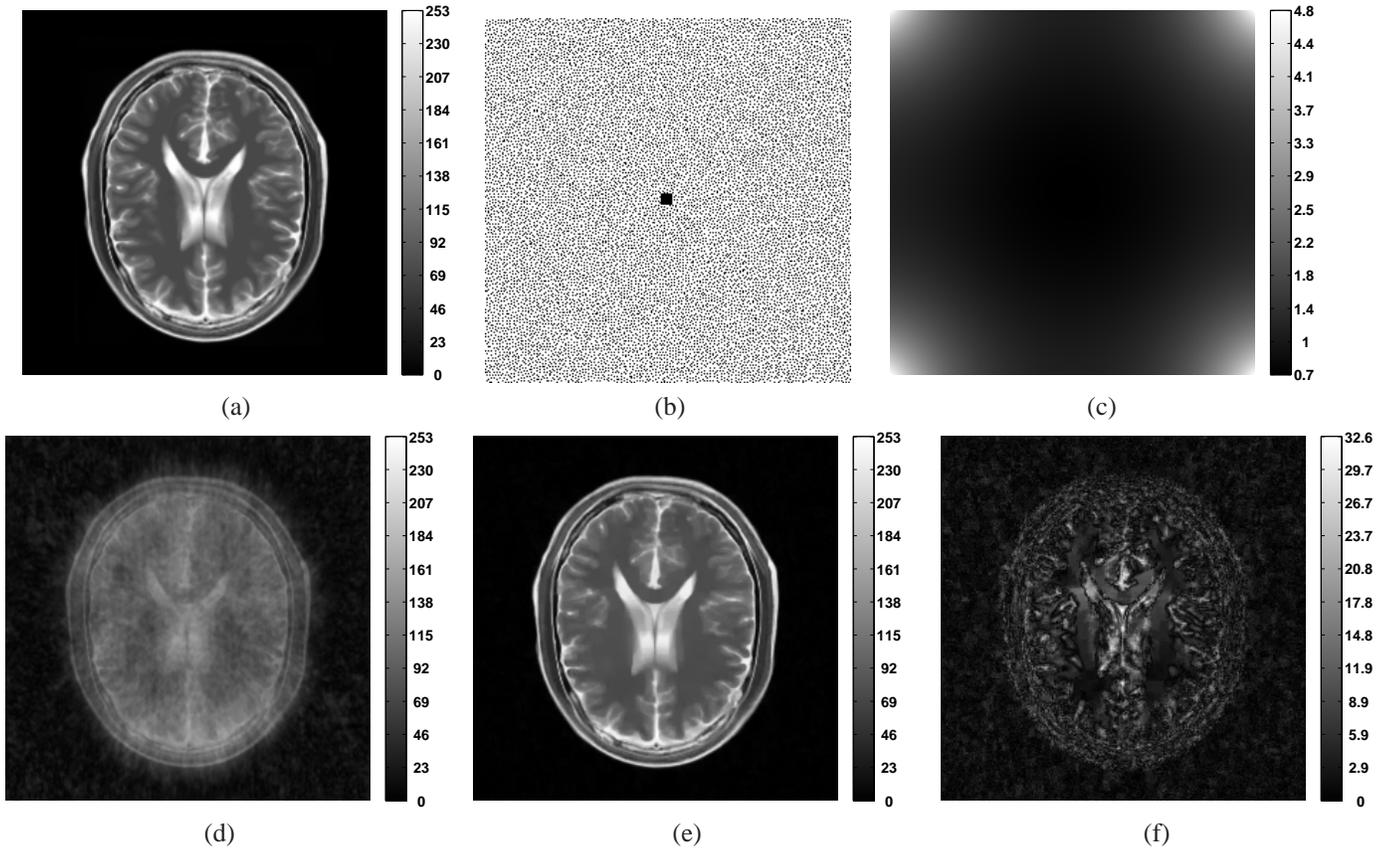


Fig. 1. Experiment with synthetic data: (a) Noise-free T2-weighted MR image used for the experiment; (b) Poisson-disk-based sampling pattern (on a Cartesian grid) in the phase-encode plane with 80% undersampling (black spots represent sample locations); (c) SoS of sensitivity maps ($\mathbf{S}^H \mathbf{S}$) of coils; (d) Square-root of SoS of coil images (SNR = 9.52 dB) obtained by taking inverse Fourier transform of the undersampled data after filling the missing k -space samples with zeros (also the initial guess $\mathbf{x}^{(0)}$); (e) the solution $\mathbf{x}^{(\infty)}$ (SNR = 24.52 dB) obtained by running **MFISTA-20**; (f) Absolute difference between (a) and (e). The goal of this work is to converge to the image $\mathbf{x}^{(\infty)}$ in (e) quickly.

[32] which provides random, but nearly uniform sampling that is advantageous for CS-MRI [33].

We compared the proposed AL methods to NCG (which has been used for CS-(p)MRI [10], [11]) and to the recently proposed MFISTA [13]—a monotone version of the state-of-the-art Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [34]. For the minimization step [13, Equation 5.3] in MFISTA, we applied the Chambolle-type algorithm developed in [35] that accommodates general regularizers of the form (6). We used the line-search described in [36] for NCG that guarantees monotonic decrease of $J(\mathbf{x})$. NCG also requires a positive “smoothing” parameter, ϵ (as indicated in [10, Appendix A]) to round-off “corners” of non-smooth regularization criteria; we set $\epsilon = 10^{-8}$ which seemed to yield good convergence speed for NCG without compromising the resulting solution too much (see Section VI-A). We implemented the following algorithms in MATLAB:

- **MFISTA- N** with N iterations of [35, Equation 6],
- **NCG- N** with N line-search iterations,
- **AL-P1- N** with N CG iterations at Step 4, and
- **AL-P2**.

We conducted the experiments on a dual quad-core Mac Pro with 2.67 GHz Intel processors. Table I shows the per-iteration computation time of the above algorithms for each experiment.

Since our goal is to minimize the cost function J (which

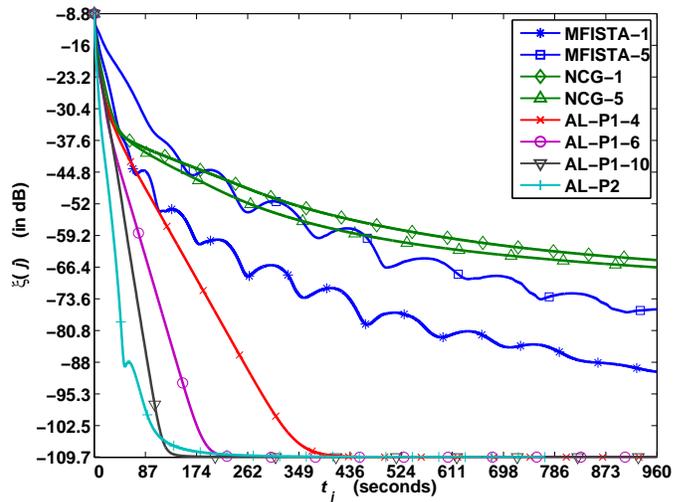


Fig. 2. Experiment with synthetic data: Plot of $\xi(j)$ as a function time t_j for NCG, MFISTA, and **AL-P1** and **AL-P2**. Both AL algorithms converge much faster than NCG and MFISTA.

determines the image quality), we focused on the speed of convergence to a solution of **P0**. For all algorithms, we quantified convergence rate by computing the normalized ℓ_2 -distance between $\mathbf{x}^{(j)}$ and the limit $\mathbf{x}^{(\infty)}$ (that represents a

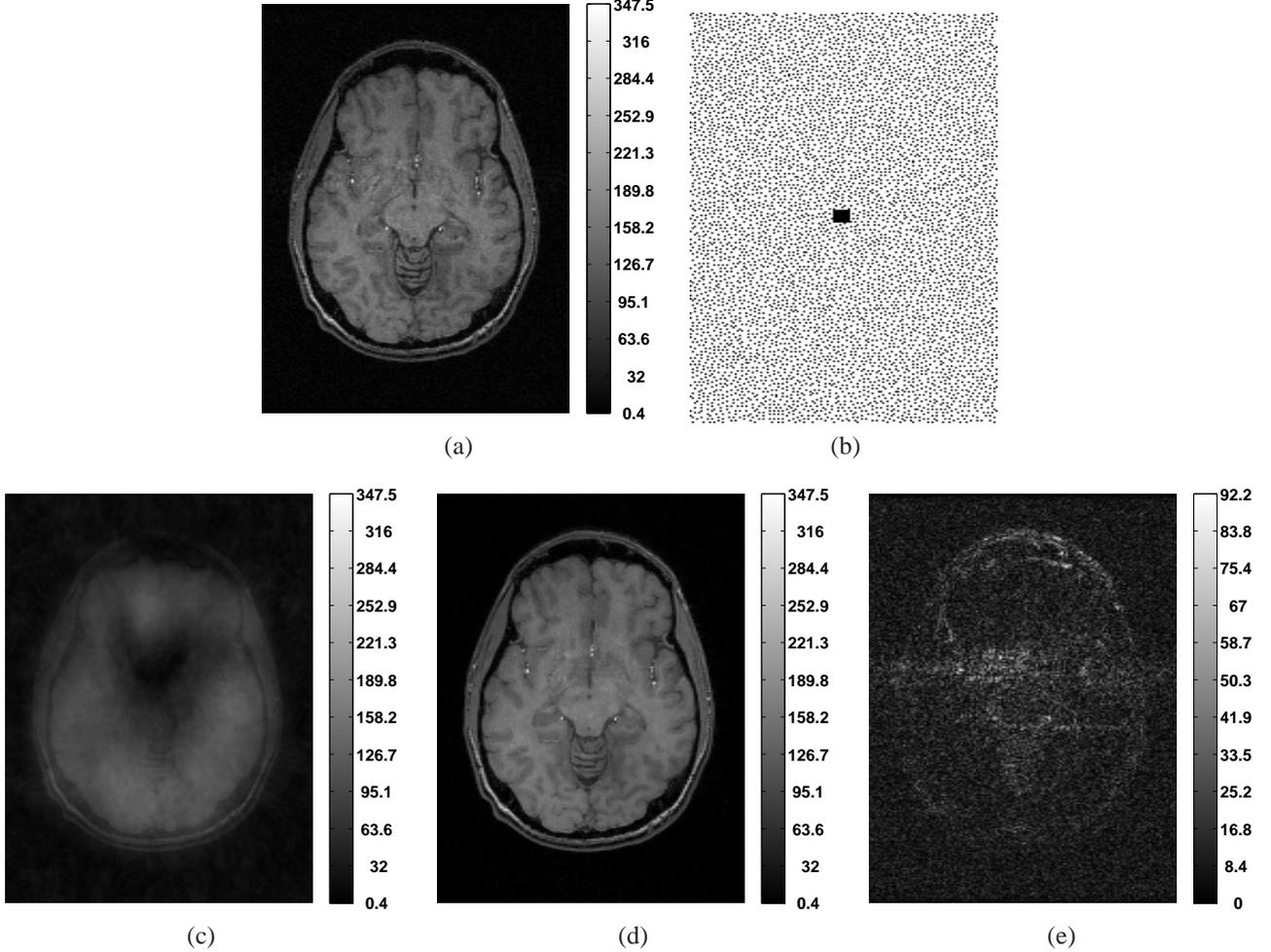


Fig. 3. Experiment with *in-vivo* human brain data (Slice 38): (a) Body-coil image corresponding to fully-sampled phase-encode; (b) Poisson-disk-based k -space sampling pattern (on a Cartesian grid) with 84% undersampling (black spots represent sample locations); (c) Square-root of SoS of coil images obtained by taking inverse Fourier transform of the undersampled data after filling the missing k -space samples with zeros (also the initial guess $\mathbf{x}^{(0)}$); (d) the solution $\mathbf{x}^{(\infty)}$ to $\mathbf{P0}$ in (2) obtained by running **MFISTA-20**; (e) Absolute difference between (a) and (d) indicates that aliasing artifacts and noise have been suppressed considerably in the reconstruction (d).

solution of $\mathbf{P0}$ given by

$$\xi(j) = 20 \log_{10} \left(\frac{\|\mathbf{x}^{(j)} - \mathbf{x}^{(\infty)}\|_2}{\|\mathbf{x}^{(\infty)}\|_2} \right). \quad (39)$$

We obtained $\mathbf{x}^{(\infty)}$ in each experiment by running thousands of iterations of **MFISTA-20** because our implementation of MFISTA (with Chambolle-type inner iterations [35]) does not require rounding the corners of non-smooth regularization unlike NCG, and therefore converges to a solution of $\mathbf{P0}$. Since the algorithms have different computational loads per outer-iteration, we evaluated $\xi(j)$ as a function of algorithm run-time⁸ t_j (time elapsed from start until iteration j). 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. For the purpose of illustration, we selected the regularization parameters $\{\lambda_q\}$ such that minimizing the corresponding J in (2) resulted in

a visually appealing solution $\mathbf{x}^{(\infty)}$. In practice, quantitative methods such as the discrepancy principle or cross-validation-based schemes may be used for automatic tuning [37] of regularization parameters. We adjusted μ for **AL-P1** and $(\nu_1$ and $\nu_2)$ for **AL-P2** as described in Section IV-C: In particular, we universally set

$$\kappa(\mathbf{H}_\mu) = 24, \quad \kappa(\mathbf{H}_{\nu_1\nu_2}) = 12, \quad (40)$$

$$\kappa(\mathbf{H}_{\nu_2}) = \min\{0.9\kappa(\tilde{\mathbf{S}}^H\tilde{\mathbf{S}}), 12\} \quad (41)$$

for **AL-P2** in all our experiments, which provided good results for different undersampling rates and regularization settings (such as ℓ_1 -norm of wavelet coefficients, TV and their combination) as demonstrated next.

B. Experiments with Synthetic Data

We considered a noise-free 256×256 T2-weighted MR image obtained from the Brainweb database [38]. We used a Poisson-disk-based sampling scheme where we fully sampled the central 8×8 portion of the k -space; the resulting sampling pattern (shown in Figure 1b) corresponded to 80%

⁸In timing MFISTA, we ignored the computation time spent on estimating the maximum eigenvalue of $\tilde{\mathbf{S}}^H\mathbf{F}^H\tilde{\mathbf{S}}$ necessary for its implementation.

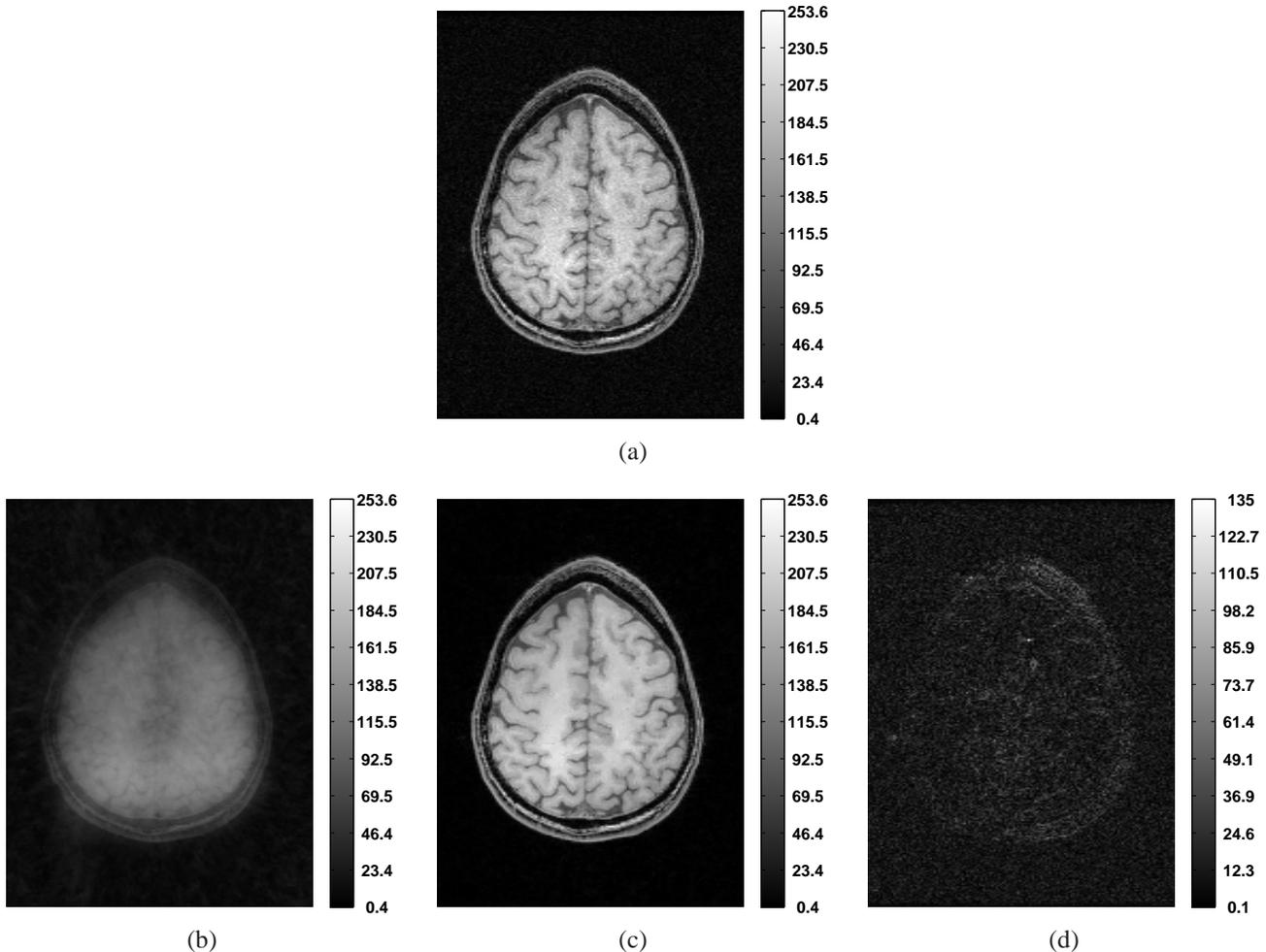


Fig. 4. Experiment with *in-vivo* human brain data (Slice 90): (a) Body-coil image corresponding to fully-sampled phase-encodes; (b) Square-root of SoS of coil images obtained by taking inverse Fourier transform of the undersampled data after filling the missing k -space samples with zeros (also the initial guess $\mathbf{x}^{(0)}$); (c) the solution $\mathbf{x}^{(\infty)}$ to $\mathbf{P0}$ in (2) obtained by running **MFISTA-20**; (d) Absolute difference between (a) and (c) indicates that aliasing artifacts and noise have been suppressed considerably in the reconstruction (c).

undersampling of the k -space. We simulated data from $L = 4$ coils whose sensitivities were generated using the technique developed in [39] (SoS of coil sensitivities is shown in Figure 1c). We added complex zero-mean white Gaussian noise (with a $1/r$ -type correlation between coils) to simulate noisy correlated coil data of 30 dB SNR. This setup simulates data acquisition corresponding to one 2-D slice of a 3-D MRI volume where the k -space sampling pattern in Figure 1b is in the phase-encode plane.

We utilized the true sensitivities and inverse noise covariance matrix (i.e., those employed for simulating data generation) to compute $\tilde{\mathbf{S}}$ in (4). We chose $\Psi(\mathbf{x}) = \|\mathbf{W}\mathbf{x}\|_{\ell_1}$, where \mathbf{W} represents 2 levels of the undecimated Haar-wavelet transform (with periodic boundary conditions) excluding the ‘scaling’ coefficients. Using ℓ_1 -regularization has reduced aliasing artifacts and restored most of the fine structures in the regularized reconstruction $\mathbf{x}^{(\infty)}$ (Figure 1e) compared to the SRSoS image (Figure 1d). Figure 2 compares NCG, MFISTA and the proposed **AL-P1** and **AL-P2** schemes in terms of speed of convergence to $\mathbf{x}^{(\infty)}$, showing $\xi(j)$ as a function of t_j for the above algorithms. Both AL methods converge significantly faster than NCG and MFISTA.

C. Experiments with In-Vivo Human Brain Data

In our next experiment, we used a 3-D *in-vivo* human brain data-set acquired from a GE 3T scanner ($T_R = 25$ ms, $T_E = 5.172$ ms, and voxel-size = $1 \times 1.35 \times 1$ mm³), 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. Two slices—Slice 38 and 90—(along x - y direction) of the reference body-coil image-volume are shown in Figures 3a and 4a, respectively. To estimate the sensitivity maps \mathbf{S} corresponding to a slice, we separately optimized a quadratic-regularized least-squares criterion (similar to [40]) that encouraged smooth maps which “closely” fit the body-coil image to the head-coil images. We estimated the inverse of noise covariance matrix \mathbf{K}_s from data collected during a dummy scan where only the static magnetic-field (and no RF excitations) was applied and computed $\tilde{\mathbf{S}}$ using (4).

We then performed regularized SENSE-reconstruction of 2-D slices (x - y plane)—Slice 38 and 90—from undersampled

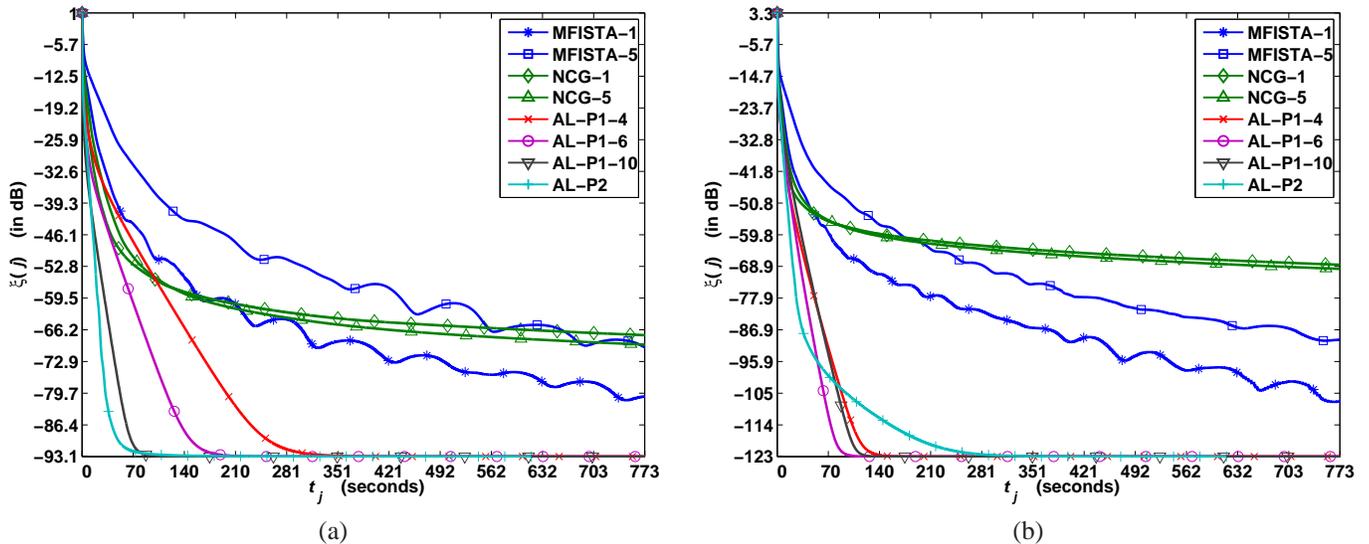


Fig. 5. Experiment with *in-vivo* human brain data: Plot of $\xi(j)$ as a function time t_j for NCG, MFISTA, **AL-P1**, and **AL-P2** for the reconstruction of (a) Slice 38, and (b) Slice 90. The AL penalty parameter μ was manually tuned for fast convergence of **AL-P1** for reconstructing Slice 38, while the same μ -value was used in **AL-P1** for reconstructing Slice 90. For **AL-P2**, the “universal” setting (40)-(41) was used for reconstructing both slices. It is seen that the AL algorithms converge much faster than NCG and MFISTA in both cases. These results also indicate that the proposed condition-number-setting (40)-(41) provides agreeably fast convergence of **AL-P2** for reconstructing multiple slices of a 3-D volume.

phase-encodes: For experiments with both slices, we applied the Poisson-disk-sampling pattern in Figure 3b (corresponding to 16% of the original 256×144 k -space samples) in the phase-encode plane and used a regularizer that combined ℓ_1 -norm of 2-level undecimated Haar-wavelet coefficients (excluding the ‘scaling’ coefficients) and TV-regularization. The reconstructions, $\mathbf{x}^{(\infty)}$, corresponding to Slice 38 and 90 were obtained by running several thousands of iterations of **MFISTA-20** and are shown in Figures 3d and 4c, respectively. Aliasing artifacts and noise have been suppressed considerably in the regularized reconstructions compared to corresponding SRSoS images (Figures 3c and 4b, respectively). We manually adjusted μ for **AL-P1** for reconstructing Slice 38 and used the same μ -value for reconstructing Slice 90 using **AL-P1**. For **AL-P2**, we used the “universal” setting (40)-(41) for reconstructing both slices. We also ran NCG and MFISTA in both cases and computed ξ . Figures 5a and 5b plot $\xi(j)$ for the all algorithms as a function of t_j . The AL algorithms converge faster than NCG and MFISTA in both cases. These figures also illustrate that choosing μ , ν_1 and ν_2 using the proposed condition-number-setting (40)-(41) provides agreeably fast convergence of **AL-P2** for reconstructing multiple slices of a 3-D volume. We also obtained results (not shown) in favor of **AL-P2** similar to those in Figures 3-5 when we repeated the above experiment (with Slices 38 and 90) with the same sampling and regularization setup but using sensitivity maps estimated from low-resolution body-coil and head-coil images obtained from iFFT-reconstruction of corresponding central 32×32 phase-encodes.

VI. DISCUSSION

A. Influence of Corner-Smoothing Parameter on NCG

Section V-A mentioned that implementing NCG requires a parameter $\epsilon > 0$ to round-off the “corners” of non-smooth

regularizers. While ϵ is usually set to a “small” value in practice, we observed in our experiments that varying ϵ over several orders of magnitude yielded a trade-off (results not shown) between the convergence speed of NCG and the limit to which it converged. Smaller ϵ yielded slow convergence speeds, probably because $\|\nabla J\|_2$ (norm of the gradient of the cost function in (2)) is large for non-smooth regularization criteria with sparsifying operators and correspondingly, many NCG-iterations may have to be executed before a satisfactory decrease of $\|\nabla J\|_2$ can be achieved. For sufficiently small ϵ , running numerous NCG-iterations would approach a solution of **P0**. On the other hand, increasing ϵ accordingly decreases the gradient-norm thereby accelerating convergence. However, for larger ϵ -values, the gradient no longer corresponds to the actual ∇J and NCG converges to something that is not a solution of **P0** (e.g., Figure 5). In our experiments, we found that $\epsilon \in [10^{-8}, 10^{-4}]$ provided reasonable balance in the above trade-off. No such ϵ is needed in MFISTA and AL methods.

B. AL-P1 versus AL-P2

Increasing the number of CG iterations, N in **AL-P1- N** , leads to a more accurate update $\mathbf{x}^{(j+1)}$ at Step 4 of **AL-P1** thereby decreasing **AL-P1**’s run-time to convergence (e.g., Figures 2 and 5a). However, at some point the computation load dominates the accuracy gained resulting in longer run-time to achieve convergence—this is illustrated in Figure 5b where **AL-P1-6** is slightly faster than **AL-P1-10**.

Selecting $\mu = \mu_{\min}$ did not consistently provide fast convergence of the split-Bregman-like **AL-P1** algorithm in our experiments as remarked in Section IV-C. Our understanding of this phenomenon is that μ_{\min} can be extremely large or small whenever the elements of $\mathbf{S}^H \mathbf{F}^H \mathbf{F} \mathbf{S}$ and $\mathbf{R}^H \mathbf{R}$ in \mathbf{G}_μ (18) are of different orders of magnitude (because \mathbf{S} can

vary arbitrarily depending on the scanner or noise level). Correspondingly, $\frac{\lambda_q}{\mu_{\min}}$ in (20) becomes very small or large, which does not favor the convergence speed of **AL-P1**.

In devising **AL-P2**, we circumvented the above problem by introducing additional splitting variables that lead to simpler matrices \mathbf{H}_μ , $\mathbf{H}_{\nu_1\nu_2}$, and \mathbf{H}_{ν_2} whose condition numbers $\kappa(\mathbf{H}_\mu)$, $\kappa(\mathbf{H}_{\nu_1\nu_2})$, and $\kappa(\mathbf{H}_{\nu_2})$, can be adjusted individually to account for differing orders of magnitude of \mathbf{F} , \mathbf{R} , and \mathbf{S} , respectively. Choosing (μ, ν_1, ν_2) based on condition numbers (40)-(41) provided good convergence speeds for **AL-P2** in our experiments (including those in Sections V-B and V-C) with different synthetic data-sets and a real breast-phantom data-set acquired with a Philips 3T scanner (results not shown). Furthermore, almost all the steps of **AL-P2** are exact which makes it more appealing for implementation. With proper code-optimization, we believe the computation-time of **AL-P2** can be reduced more than that of **AL-P1**.

C. Constraint Involving the Data

Recently, Liu *et al* [14] applied a Bregman iterative scheme to TV-regularized SENSE-reconstruction, which converges to a solution of the constrained optimization problem

$$\min_{\mathbf{x}} \Psi(\mathbf{R}\mathbf{x}) \text{ subject to } \mathbf{F}\mathbf{S}\mathbf{x} = \mathbf{d} \quad (42)$$

for some regularization Ψ . Although this paper has focused on faster algorithms for solving the unconstrained problem (**P0**), we can extend the proposed approaches to solve (42) by including a constraint involving the data. For instance, (42) can be reformulated as

$$\min_{\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3, \mathbf{x}} \Psi(\mathbf{u}_2) \text{ subject to} \\ \mathbf{d} = \mathbf{F}\mathbf{u}_1, \mathbf{u}_1 = \mathbf{S}\mathbf{x}, \mathbf{u}_2 = \mathbf{R}\mathbf{u}_3, \text{ and } \mathbf{u}_3 = \mathbf{x}, \quad (43)$$

where we have introduced auxiliary variables to decouple the data-domain components \mathbf{F} and \mathbf{S} , and the regularization component \mathbf{R} . The AL technique (Section III) can then be applied to (43) noting that it can be written in the general form of (7) as

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \\ \mathbf{x} \end{bmatrix}, \mathbf{b} = \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \mathbf{C} = \Lambda_1 \begin{bmatrix} \mathbf{F} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{I}_{NL} & \mathbf{0} & \mathbf{0} & -\mathbf{S} \\ \mathbf{0} & \mathbf{I}_R & -\mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_N & -\mathbf{I}_N \end{bmatrix},$$

$f(\mathbf{u}) = \Psi(\mathbf{u}_2)$, where Λ_1 is a suitable weighting matrix similar to Λ administered in **P2**, respectively. The **AL** algorithm (Section III) applied to (43) will converge to a solution that satisfies the constraint in (42).

VII. SUMMARY AND CONCLUSIONS

The augmented Lagrangian (AL) framework constitutes an attractive class of methods for solving constrained optimization problems. In this paper, we investigated the use of AL-based methods for MR image reconstruction from undersampled data using sensitivity encoding (SENSE) with a general class of regularization functional. Specifically, we formulated regularized SENSE-reconstruction as an unconstrained optimization problem in a penalized-likelihood framework

and investigated two constrained versions—equivalent to the original unconstrained problem—using variable splitting. The first version, **P1**, is similar to the split-Bregman approach [15] where we split only the regularization term. In the second version, **P2**, we proposed to split the components of the data-fidelity term as well. These constrained problems were then tackled in the AL framework. We applied alternating schemes to decouple the minimization of the associated AL functions and developed AL algorithms **AL-P1** and **AL-P2**, respectively, thereof.

The convergence speeds of the above AL algorithms is chiefly determined by the AL penalty parameter μ . Automatically selecting μ for fast convergence of **AL-P1** still remains to be addressed for regularized SENSE-reconstruction. This is a significant practical drawback of **AL-P1**. However, for **AL-P2** we provided an empirical condition-number-rule to select μ for fast convergence. In our experiments with synthetic and real data, the proposed AL algorithms—**AL-P1** and **AL-P2** (with μ determined as above)—converged faster than conventional (NCG) and state-of-the-art (MFISTA) methods. The algebraic developments and numerical results in this paper indicate the potential of using variable splitting and alternating minimization in the AL formalism for solving other large-scale constrained/unconstrained optimization problems.

ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewer for suggesting the reformulation of the problem in equation (5), Dr. Jon-Fredrik Nielsen, University of Michigan, for providing the *in-vivo* human brain data-set used in the experiments and Michael Allison, University of Michigan, for carefully proofreading the manuscript.

REFERENCES

- [1] K. P. Pruessmann, M. Weiger, M. B. Scheidegger, and P. Boesiger, "SENSE: Sensitivity Encoding for Fast MRI," *Magnetic Resonance in Medicine*, vol. 42, pp. 952–962, 1999.
- [2] K. P. Pruessmann, M. Weiger, P. Börner, and P. Boesiger, "Advances in Sensitivity Encoding with Arbitrary k -Space Trajectories," *Magnetic Resonance in Medicine*, vol. 46, pp. 638–651, 2001.
- [3] F. H. Lin, K. K. Kwong, J. W. Belliveau, and L. L. Wald, "Parallel Imaging Reconstruction Using Automatic Regularization," *Magnetic Resonance in Medicine*, vol. 51, pp. 559–567, 2004.
- [4] P. Qu, J. Luo, B. Zhang, J. Wang, and G. X. Shen, "An Improved Iterative SENSE Reconstruction Method," *Magnetic Resonance Engineering*, vol. 31, pp. 44–50, 2007.
- [5] F. H. Lin, F. N. Wang, S. P. Ahlfors, M. S. Hamalainen, and J. W. Belliveau, "Parallel MRI Reconstruction Using Variance Partitioning Regularization," *Magnetic Resonance in Medicine*, vol. 58, pp. 735–744, 2007.
- [6] M. Bydder, J. E. Perthen, and J. Du, "Optimization of Sensitivity Encoding with Arbitrary k -Space Trajectories," *Magnetic Resonance Imaging*, vol. 25, pp. 1123–1129, 2007.
- [7] K. T. Block, M. Uecker, and J. Frahm, "Undersampled Radial MRI with Multiple Coils. Iterative Image Reconstruction Using a Total Variation Constraint," *Magnetic Resonance in Medicine*, vol. 57, pp. 1086–1098, 2007.
- [8] A. Raj, G. Singh, R. Zabih, B. Kressler, Y. Wang, N. Schuff, and M. Weiner, "Bayesian Parallel Imaging With Edge-Preserving Priors," *Magnetic Resonance in Medicine*, vol. 57, pp. 8–21, 2007.
- [9] L. Ying, B. Liu, M. C. Steckner, G. Wu, M. Wu, and S. L. Li, "A Statistical Approach to SENSE Regularization with Arbitrary k -Space Trajectories," *Magnetic Resonance in Medicine*, vol. 60, pp. 414–421, 2008.

- [10] M. Lustig, D. L. Donoho, and J. M. Pauly, "Sparse MRI: The Application of Compressed Sensing for Rapid MR Imaging," *Magnetic Resonance in Medicine*, vol. 58, pp. 1182–1195, 2007.
- [11] D. Liang, B. Liu, J. Wang, and L. Ying, "Accelerating SENSE Using Compressed Sensing," *Magnetic Resonance in Medicine*, vol. 62, pp. 1574–1584, 2009.
- [12] D. P. Bertsekas, "Multiplier Methods: A Survey," *Automatica*, vol. 12, pp. 133–145, 1976.
- [13] 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.
- [14] B. Liu, K. King, M. Steckner, J. Xie, J. Sheng, and L. Ying, "Regularized Sensitivity Encoding (SENSE) Reconstruction Using Bregman Iterations," *Magnetic Resonance in Medicine*, vol. 61, pp. 145–152, 2009.
- [15] 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.
- [16] S. Ma, W. Yin, Y. Zhang, and A. Chakraborty, "An Efficient Algorithm for Compressed MR Imaging Using Total Variation and Wavelets," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, 2008.
- [17] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A New Alternating Minimization Algorithm for Total Variation Image Reconstruction," *SIAM J. Imaging Sciences*, vol. 1, no. 3, pp. 248–272, 2008.
- [18] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "An Augmented Lagrangian Approach to the Constrained Optimization Formulation of Imaging Inverse Problems," *IEEE Trans. Image Processing*, <http://arxiv.org/abs/0912.3481> accepted 2009.
- [19] M. V. Afonso, J. M. Bioucas-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.
- [20] M. A. T. Figueiredo and J. M. Bioucas-Dias, "Restoration of Poissonian Images Using Alternating Direction Optimization," *IEEE Trans. Image Processing*, <http://dx.doi.org/10.1109/TIP.2010.2053941> accepted 2010.
- [21] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer, New York, 1999.
- [22] E. Esser, "Applications of Lagrangian-Based Alternating Direction Methods and Connections to Split Bregman," Computational and Applied Mathematics Technical Report 09-31, University of California, Los Angeles, 2009.
- [23] W. Yin, S. Osher, D. Goldfarb, and J. Darbon, "Bregman Iterative Algorithms for ℓ_1 Minimization with Applications to Compressed Sensing," *SIAM J. Imaging Sciences*, vol. 1, pp. 143–168, 2008.
- [24] R. Glowinski and A. Marroco, "Sur l'Approximation par Element Finis d'Ordre un, et la Resolution, par Penalisation-Dualité, d'une Classe de Problemes de Cirichlet Nonlineares," *Revue Française d'Automatique, Informatique et Recherche Opérationnelle* 9, vol. R-2, pp. 41–76, 1975.
- [25] D. Gabay and B. Mercier, "A Dual Algorithm for the Solution of Nonlinear Variational Problems via Finite Element Approximations," *Computers and Mathematics with Applications* 2, pp. 17–40, 1976.
- [26] M. Fortin and R. Glowinski, "On Decomposition-Coordination Methods Using an Augmented Lagrangian," in *Augmented Lagrangian Methods: Applications to the Solution of Boundary-Value Problems*, M. Fortin and R. Glowinski, Eds. North-Holland, Amsterdam, 1983.
- [27] A. Chambolle, R. A. DeVore, N. Y. Lee, and B. J. Lucier, "Nonlinear Wavelet Image Processing: Variational Problems, Compression, and Noise Removal Through Wavelet Shrinkage," *IEEE Trans. Image Processing*, vol. 7, no. 3, pp. 319–335, March 1998.
- [28] C. Chau, P. L. Combettes, J. C. Pesquet, and V. R. Wajs, "A Variational Formulation for Frame-Based Inverse Problems," *Inverse Problems*, vol. 23, pp. 1496–1518, 2007.
- [29] J. I. Jackson, C. H. Meyer, D. G. Nishimura, and A. Macovski, "Selection of a Convolution Function for Fourier Inversion Using Gridding," *IEEE Trans. Medical Imaging*, vol. 10, no. 3, pp. 473–478, 1991.
- [30] A. E. Yagle, "New Fast Preconditioners for Toeplitz-like Linear Systems," *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 1365–1368, 2002.
- [31] R. H. Chan and M. K. Ng, "Conjugate gradient methods for Toeplitz Systems," *SIAM Review*, vol. 38, pp. 427–482, 1996.
- [32] Daniel Dunbar and Greg Humphreys, "A Spatial Data Structure for Fast Poisson-Disk Sample Generation," *Proceedings of SIGGRAPH*, Boston, MA, USA, 30 July-3 August, 2006 (<http://www.cs.virginia.edu/~gfx/pubs/antimony/>).
- [33] M. Lustig, M. Alley, S. Vasanawala, D. L. Donoho, and J. M. Pauly, "Autocalibrating Parallel Imaging Compressed Sensing using L_1 SPIRiT with Poisson-Disk Sampling and Joint Sparsity Constraints," *Proceedings of International Society for Magnetic Resonance in Medicine*, p. 334, 2009.
- [34] A. Beck and M. Teboulle, "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems," *SIAM J. Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.
- [35] 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.
- [36] J. A. Fessler and S. D. Booth, "Conjugate-Gradient Preconditioning Methods for Shift-Variant PET Image Reconstruction," *IEEE Trans. Image Processing*, vol. 8, no. 5, pp. 688–699, May 1999.
- [37] W. C. Karl, "Regularization in Image Restoration and Reconstruction," in *Handbook of Image & Video Processing*, A. Bovik, Ed., pp. 183–202. ELSEVIER, 2nd edition, 2005.
- [38] "Brainweb: Simulated MRI Volumes for Normal Brain," McConnell Brain Imaging Centre, <http://www.bic.mni.mcgill.ca/brainweb/>.
- [39] M. I. Grivich and D. P. Jackson, "The Magnetic Field of Current-Carrying Polygons: An Application of Vector Field Rotations," *American Journal of Physics*, vol. 68, no. 5, pp. 469–474, May 2000.
- [40] S. L. Keeling and R. Bammer, "A Variational Approach to Magnetic Resonance Coil Sensitivity Estimation," *Applied Mathematics and Computation*, vol. 158, no. 2, pp. 53–82, 2004.

Parallel MR Image Reconstruction using Augmented Lagrangian Methods: Supplementary Material

Sathish Ramani*, *Member, IEEE*, and Jeffrey A. Fessler, *Fellow, IEEE*

In this transcript, we discuss the quantitative selection of the augmented Lagrangian (AL) parameters μ , ν_1 , and ν_2 associated with the AL algorithms (**AL-P1** and **AL-P2**) that we developed in “Parallel MR Image Reconstruction using Augmented Lagrangian Methods”, *IEEE Transactions on Medical Imaging*. These parameters govern only the convergence speed of the above AL algorithms and does not affect the solution of the regularized SENSE-reconstruction problem. References to equations, sections, tables, figures, etc., given here are with respect to the paper unless stated otherwise.

As described in Section IV-C, for **AL-P1**, μ is related to the condition number $\kappa(\mathbf{G}_\mu)$ of \mathbf{G}_μ (18), while for **AL-P2**, μ , ν_1 , and ν_2 are related to $\kappa(\mathbf{H}_\mu)$, $\kappa(\mathbf{H}_{\nu_1\nu_2})$ and $\kappa(\mathbf{H}_{\nu_2})$ of \mathbf{H}_μ , $\mathbf{H}_{\nu_1\nu_2}$ and \mathbf{H}_{ν_2} (33)-(35), respectively. So we proposed to select these parameters by adjusting $\kappa(\cdot)$ for fast convergence of the AL algorithms in the paper. The **AL-P1** algorithm is similar to the split-Bregman algorithm [15] as it is based on splitting the regularization term alone (Section IV-A). It was suggested in [15] that one can select $\mu = \mu_{\min} \triangleq \arg \min_{\mu} \kappa(\mathbf{G}_\mu)$ for split-Bregman-like schemes such as **AL-P1**, so as to minimize the condition number $\kappa(\mathbf{G}_\mu)$ of \mathbf{G}_μ thereby ensuring fast convergence of the conjugate gradient (CG) algorithm for solving¹ (17). We observed in our experiments that this rule did not consistently yield good convergence speeds for **AL-P1**: μ_{\min} and μ_{opt} (the μ -value that provides best convergence speed for **AL-P1**) differed at least by an order of magnitude in all our experiments and the convergence speed of **AL-P1** achieved using μ_{\min} was far less compared to that obtained using μ_{opt} . We provide a possible explanation for this behavior at the end of Section 4 of this note. Table I at the end of this note succinctly summarizes these results. In Figures 1, 3, 4, 5, and 7, the vertical black-dashed line indicates μ_{\min} , while the vertical red-dashed lines indicate μ_{opt} for **AL-P1-4**, **AL-P1-6** and **AL-P1-10**, respectively.

We also illustrate the effectiveness of an empirical condition number rule for selecting μ , ν_1 , and ν_2 for fast convergence of **AL-P2** in Section 5 of this note.

1. EXPERIMENT WITH SYNTHETIC DATA-SET

For the experiment described in Section V-B, Figure 1 in this note plots the normalized ℓ_2 -distance ξ (equation (39)) which quantifies the convergence speed) to the solution $\mathbf{x}^{(\infty)}$ of original regularized SENSE-reconstruction problem for various run-times of **AL-P1** and the corresponding² $\kappa(\mathbf{G}_\mu)$ as functions of μ . For this experiment $\xi(\mu_{\min}) = -15.48$ dB for **AL-P1** which is far from $\xi(\mu_{\text{opt}}) = -102.23$ dB.

2. EXPERIMENT WITH REAL BREAST-PHANTOM DATA-SET

We also performed a similar experiment with a breast-phantom data-set acquired from a Philips 3T scanner with a 4-channel coil: The k -space data corresponded to $800 \times 394 \times 94$ uniformly-spaced samples in the k_x (read-out), k_y , and k_z (phase-encode) directions, respectively. From the fully sampled 3-D data, we computed the square-root of SoS (SRSoS) reconstruction which served as a reference for quality; Slice 418 (along y - z direction) of the reference SRSoS volume is shown in Figure 2a. To estimate the sensitivity maps, we truncated the fully sampled phase-encodes by applying a 48×12 cosine-squared window centered at the origin to generate smoothed coil images and computed the ratios of these smooth images to their SRSoS reconstruction. We estimated the noise covariance matrix from data collected during a dummy scan where the phantom was magnetized but no RF excitations were applied.

¹In the figures, **AL-P1-N** stands for **AL-P1** with N CG iterations applied to (17).

²In all experiments, we estimated $\kappa(\mathbf{G}_\mu)$ using the Power method applied to \mathbf{G}_μ and \mathbf{G}_μ^{-1} , where we implemented \mathbf{G}_μ^{-1} using 500 iterations of the CG algorithm.

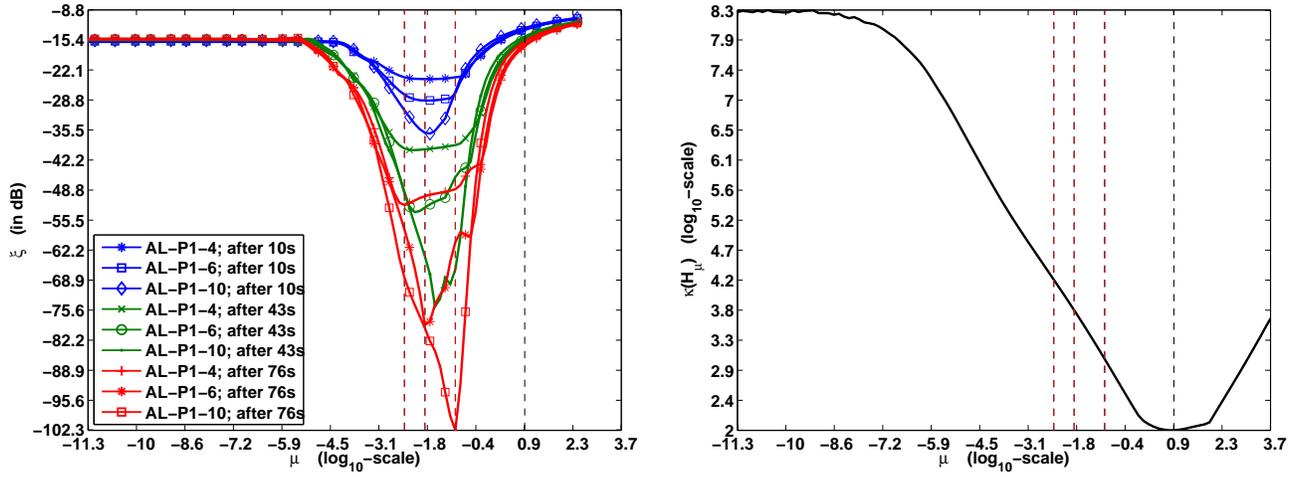


Fig. 1. **Experiment with synthetic data-set** (corresponding to Section V-B in the paper): Plots of ξ for various run-times of **AL-P1** and condition number $\kappa(\mathbf{G}_\mu)$ as functions of μ . It is seen that μ_{\min} (indicated by a black-dashed line) does not provide good convergence speed for **AL-P1** in this example.

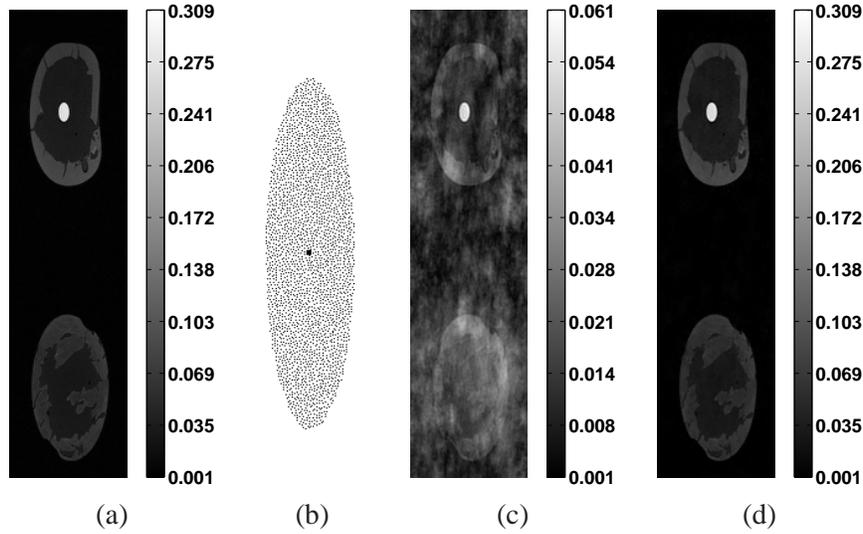


Fig. 2. **Experiment with real breast-phantom data-set (Slice 418)**: (a) Square-root of SoS (SRSoS) of coil images corresponding to fully sampled phase-encodes; (b) Poisson-disk-based k -space sampling pattern (on a Cartesian grid); (c) SRSoS of coil images obtained by taking inverse Fourier transform of zero-filled undersampled data; (d) the solution $\mathbf{x}^{(\infty)}$ obtained by running **MFISTA-20**.

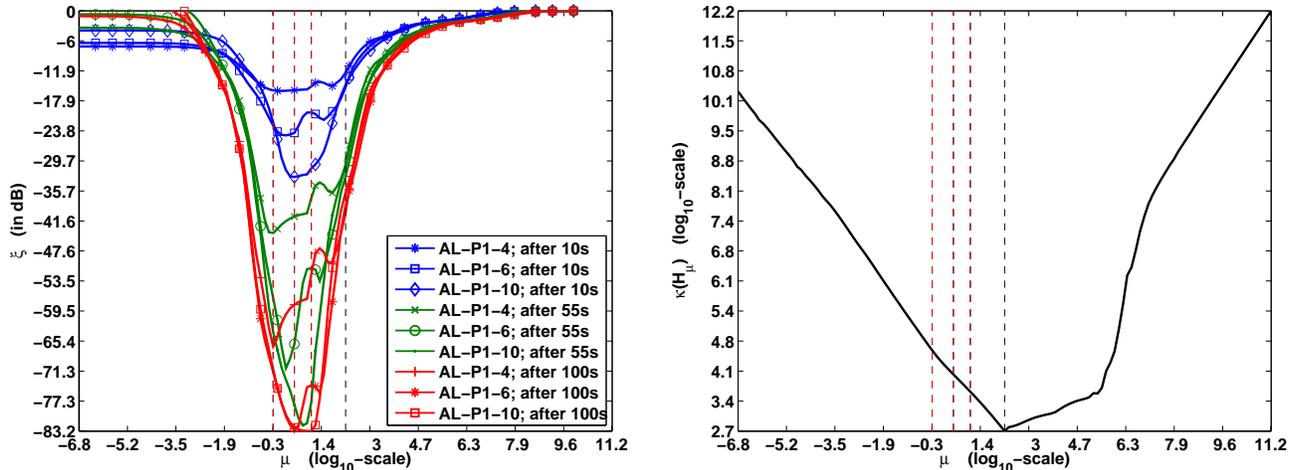


Fig. 3. **Experiment with real breast-phantom data-set (Slice 418)**: Plots of ξ for various run-times of **AL-P1** and condition number $\kappa(\mathbf{G}_\mu)$ as functions of μ . It is seen that **AL-P1** using μ_{\min} (indicated by a black-dashed line) converges relatively slowly compared to using μ_{opt} .

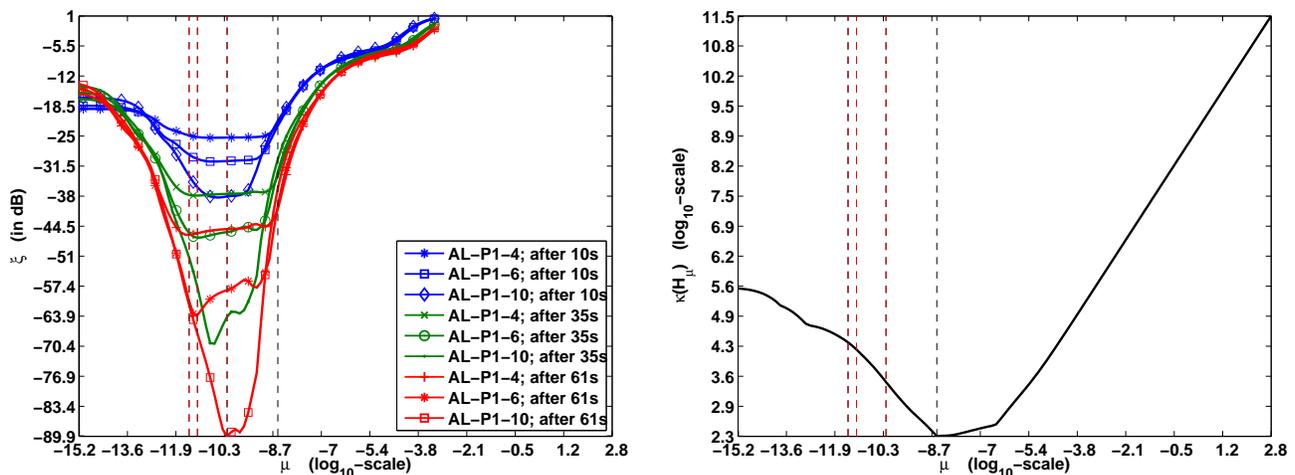


Fig. 4. **Experiment with Slice 38 of *in-vivo* human brain data-set** (corresponding to Section V-C): Plots of ξ for various run-times of **AL-P1** and condition number $\kappa(\mathbf{G}_\mu)$ as functions of μ . It is seen that **AL-P1** using μ_{\min} (indicated by a black-dashed line) converges relatively slowly compared to using μ_{opt} .

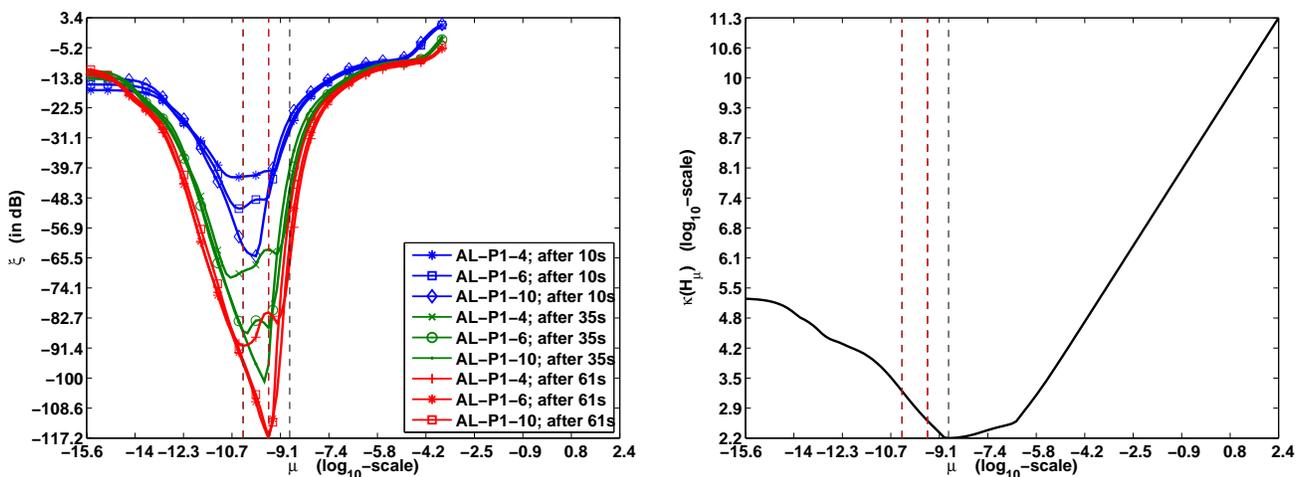


Fig. 5. **Experiment with Slice 90 of *in-vivo* human brain data-set** (corresponding to Section V-C): Plots of ξ for various run-times of **AL-P1** and condition number $\kappa(\mathbf{G}_\mu)$ as functions of μ . It is seen that **AL-P1** using μ_{\min} (indicated by a black-dashed line) converges relatively slowly compared to using μ_{opt} .

We then performed SENSE-reconstruction of Slices 418 from undersampled phase-encodes using **AL-P1**. We used a Poisson-disk-sampling pattern (confined to a Cartesian grid where we fully sampled the central 4×4 portion) in the phase-encode plane, corresponding to 7% of the original 394×94 phase-encodes (see Figure 2b). We used a regularizer that combined ℓ_1 -norm of 2-level undecimated Haar wavelet coefficients (excluding the ‘scaling’ coefficients) and total-variation regularization. The reconstruction $\mathbf{x}^{(\infty)}$ corresponding to Slice 418 was obtained by running several thousands of iterations of **MFISTA-20** (as explained in Section V-A of the paper) and is shown in Figure 2d. We ran **AL-P1** for various μ and computed ξ . Figure 3 in this note plots ξ for various run-times of **AL-P1** and the corresponding $\kappa(\mathbf{G}_\mu)$ as functions of μ . For this experiment $\xi(\mu_{\min}) = -36.28$ dB which is sub-optimal compared to $\xi(\mu_{\text{opt}}) = -83.18$ dB. We obtained similar results in favor of μ_{opt} (under the same experimental setup) for reconstructing other slices of the real breast-phantom data-set using **AL-P1**.

3. EXPERIMENT WITH *In-Vivo* HUMAN BRAIN DATA-SET

For the experiment (with Slices 38 and 90 of *in-vivo* human brain data-set) described in Section V-C, Figures 4 and 5 plot ξ for various run-times of **AL-P1** and $\kappa(\mathbf{G}_\mu)$ as functions of μ , for Slices 38 and 90, respectively. In this experiment too, μ_{\min} yields sub-optimal convergence speeds ($\xi(\mu_{\min}) = -36.20$ dB for Slice 38 and $\xi(\mu_{\min}) = -51.99$ dB for Slice 90) for **AL-P1** compared to μ_{opt} ($\xi(\mu_{\text{opt}}) = -89.82$ dB for Slice 38 and $\xi(\mu_{\text{opt}}) = -116.90$ dB for Slice 90).

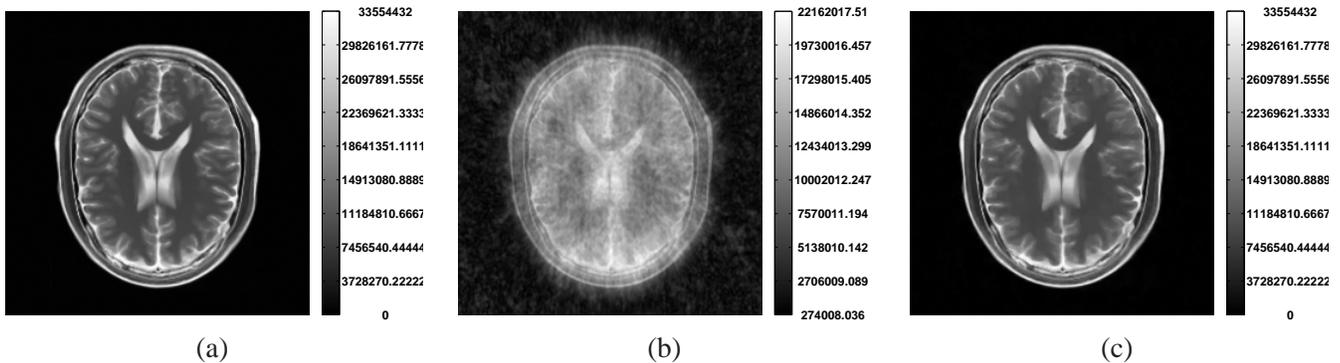


Fig. 6. **Experiment with modified synthetic data-set:** (a) Scaled noise-free T2-weighted MR image; (b) SRSoS of coil images obtained by taking inverse Fourier transform of the zero-filled undersampled data; (c) the solution $\mathbf{x}^{(\infty)}$ obtained by running **MFISTA-20**.

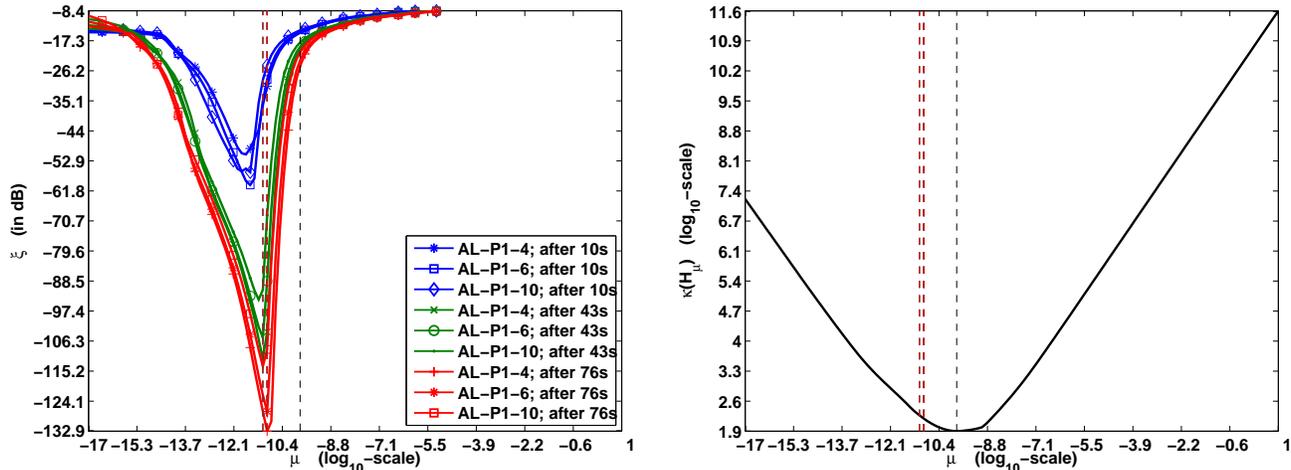


Fig. 7. **Experiment with modified synthetic data-set:** Plots of ξ for various run-times of **AL-P1** and condition number $\kappa(\mathbf{G}_\mu)$ as functions of μ . The μ that minimizes $\kappa(\mathbf{G}_\mu)$ does not yield the best convergence speed for **AL-P1**.

4. EXPERIMENT WITH A DIFFERENT SYNTHETIC DATA-SET

In another experiment, we generated a synthetic data-set from the T2-weighted noise-free MR image (in Section 1) after scaling it to increase its dynamic range from $[0, 253]$ to $[0, 33554432]$. We used the same experimental setting described in Section V-B and Figure 1 in the paper (i.e., same sensitivity maps, noise level, sampling pattern and regularizer with appropriately scaled regularization parameter) and generated noisy data with 30 dB SNR. This resulted in large values in the inverse noise-covariance matrix and correspondingly the spectra of $\tilde{\mathbf{S}}^H \mathbf{F}^H \mathbf{F} \tilde{\mathbf{S}}$ (where we utilized the true sensitivities and inverse noise covariance matrix to compute $\tilde{\mathbf{S}}$) and $\mathbf{R}^H \mathbf{R}$ were several magnitudes apart. For this experiment, we again obtained $\mathbf{x}^{(\infty)}$ using **MFISTA-20** (show in Figure 6c). We ran **AL-P1**, computed ξ and $\kappa(\mathbf{G}_\mu)$ and plotted them as functions of μ in Figure 7. Similar to the situations encountered earlier, μ_{\min} is comparably close to μ_{opt} but still does not provide a good convergence speed.

Our understanding of the phenomenon encountered in Sections 1-4 is that the proposed AL algorithms, **AL-P1** and **AL-P2**, are sensitive to the threshold-values $\tau_1 = \frac{\lambda}{\mu}$ and $\tau_2 = \frac{\lambda}{\mu \nu_1}$, respectively and that τ_1 and τ_2 need to be carefully set (by fixing μ and ν_1 properly) to ensure rapid convergence of the AL algorithms. However, τ_1 is more sensitive to μ compared to τ_2 because τ_1 depends only on μ —even a small deviation from the optimal value μ_{opt} becomes detrimental to the convergence speed of **AL-P1** as seen from Figures 1, 3, 4, 5, and 7, where **AL-P1** exhibits good convergence speeds only in a narrow μ -window.

Table I succinctly summarizes the results from Sections 1-4 of this note where we compare the ratio of maximum eigenvalues of $\tilde{\mathbf{S}}^H \mathbf{F}^H \mathbf{F} \tilde{\mathbf{S}}$ and $\mathbf{R}^H \mathbf{R}$, μ_{\min} and μ_{opt} . In all experiments, it is seen that μ_{\min} is of the same order of magnitude and is relatively close to $\frac{\text{maxeigval}\{\tilde{\mathbf{S}}^H \mathbf{F}^H \mathbf{F} \tilde{\mathbf{S}}\}}{\text{maxeigval}\{\mathbf{R}^H \mathbf{R}\}}$ which is quite expected since μ_{\min} balances

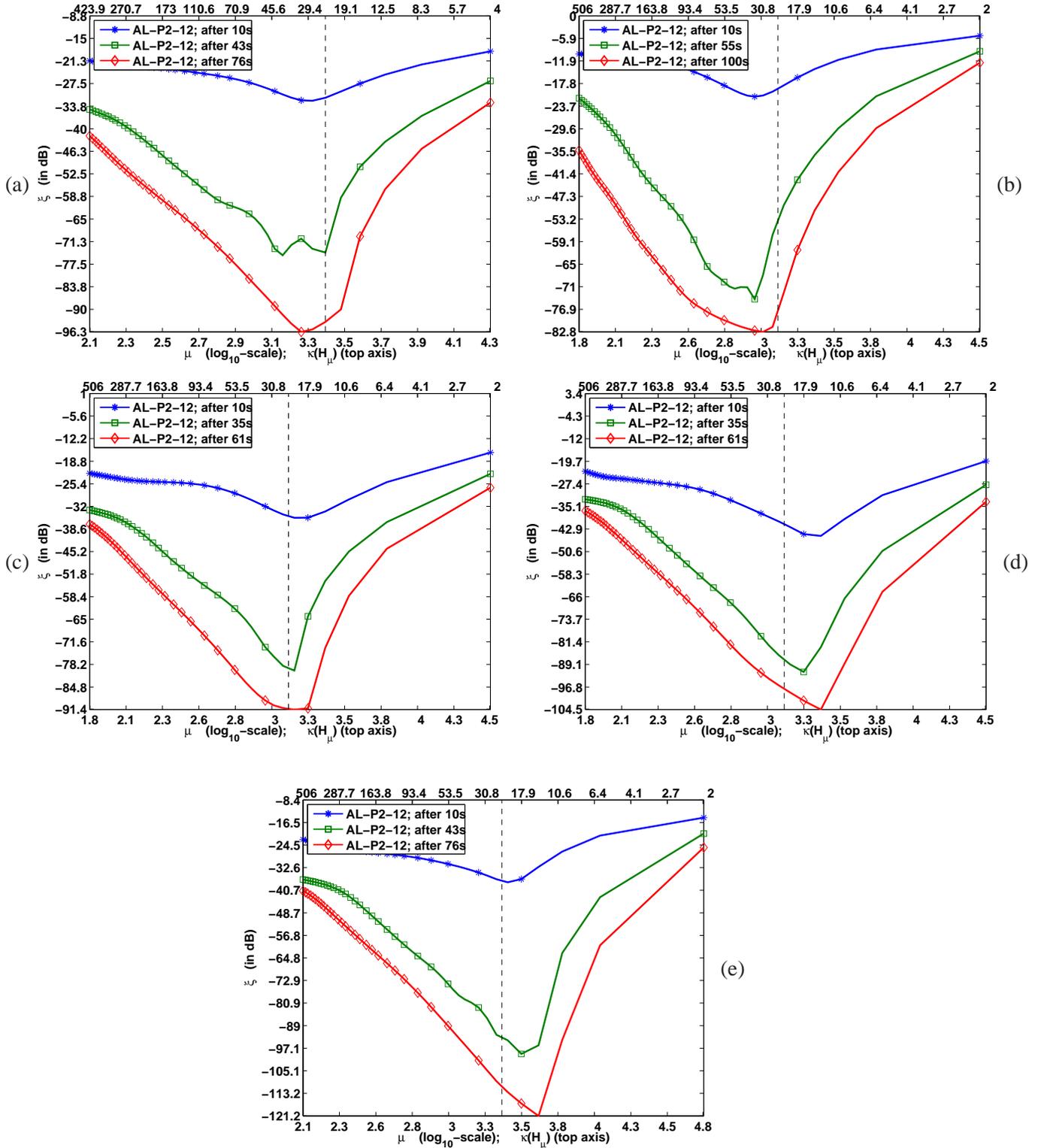


Fig. 8. Plots of ξ for various run-times of AL-P2 as a function of μ for (a) Experiment with synthetic data-set (Section 1); (b) Experiment with Slice 418 of real breast-phantom data-set (Section 2); (c) Experiment with Slice 38 of *in-vivo* human brain data-set (Section 3); (d) Experiment with Slice 90 of *in-vivo* human brain data-set (Section 3); (e) Experiment with modified synthetic data-set (Section 4). The vertical black-dashed line indicates $\mu = \mu_0$ for which $\kappa(\mathbf{H}_{\mu_0}) = 24$ used in our experiments. It is seen that $\kappa(\mathbf{H}_\mu) \in [10, 36]$ provides agreeable results for all data-sets even for cases (c)-(e) where corresponding μ_{\min} did not yield comparable results for AL-P1.

TABLE I
COMPARISON OF RATIO OF MAXIMUM EIGENVALUES OF $\tilde{\mathbf{S}}^H \mathbf{F}^H \tilde{\mathbf{F}} \tilde{\mathbf{S}}$ AND $\mathbf{R}^H \mathbf{R}$, $\mu_{\min} = \arg \min_{\mu} \kappa(\mathbf{G}_{\mu})$, μ_{opt} (μ -VALUE THAT PROVIDES BEST CONVERGENCE SPEED FOR **AL-P1**), NORMALIZED ℓ_2 -DISTANCE TO $\mathbf{x}^{(\infty)}$, ξ , OBTAINED BY RUNNING **AL-P1** WITH μ_{\min} AND μ_{opt}

Experiment	$\frac{\text{maxeigval}\{\tilde{\mathbf{S}}^H \mathbf{F}^H \tilde{\mathbf{F}} \tilde{\mathbf{S}}\}}{\text{maxeigval}\{\mathbf{R}^H \mathbf{R}\}}$	μ_{\min}	μ_{opt}	$\xi(\mu_{\min})$ (in dB)	$\xi(\mu_{\text{opt}})$ (in dB)
Section 1	2.286	10.109	0.112	-15.48	-102.23
Section 2	152.618	178.438	12.397	-36.28	-83.18
Section 3 (Slice 38)	4.999×10^{-9}	3.567×10^{-9}	6.887×10^{-11}	-36.20	-89.82
Section 3 (Slice 90)	2.291×10^{-9}	1.985×10^{-9}	3.831×10^{-10}	-51.99	-116.90
Section 4	3.793×10^{-11}	1.703×10^{-10}	9.450×10^{-12}	-20.20	-113.71

TABLE II
COMPARISON OF μ_0 WHERE $\kappa(\mathbf{H}_{\mu=\mu_0}) = 24$, μ_{opt} FOR **AL-P2**, NORMALIZED ℓ_2 -DISTANCE TO $\mathbf{x}^{(\infty)}$, ξ , OBTAINED BY RUNNING **AL-P2** WITH μ_0 AND μ_{opt}

Experiment	μ_0	μ_{opt}	$\xi(\mu_0)$ (in dB)	$\xi(\mu_{\text{opt}})$ (in dB)
Section 1	2849.391	2114.065	-93.48	-96.25
Section 2	1755.429	1271.1724	-72.45	-82.73
Section 3 (Slice 38)	1755.429	1755.427	-91.31	-91.31
Section 3 (Slice 90)	1755.427	2835.692	-98.58	-104.44
Section 4	3120.762	5041.231	-112.69	-121.19

the spectra of $\tilde{\mathbf{S}}^H \mathbf{F}^H \tilde{\mathbf{F}} \tilde{\mathbf{S}}$ and $\mathbf{R}^H \mathbf{R}$ in the process of minimizing $\kappa(\mathbf{G}_{\mu})$. As mentioned earlier, it is seen that **AL-P1** using μ_{\min} does not lead to a significant decrease of ξ as compared to that using μ_{opt} .

In summary, \mathbf{G}_{μ} is a non-circulant matrix without any special structure whose condition number $\kappa(\mathbf{G}_{\mu})$ cannot be computed in a straightforward manner. Moreover, since $\tilde{\mathbf{S}}$ can vary arbitrarily in SENSE-reconstruction problems depending on the scanner or noise level, $\min_{\mu} \kappa(\mathbf{G}_{\mu})$ has to be performed on a case-by-case basis which can be tedious. Finally, as illustrated in Figures 1-7, minimizing $\kappa(\cdot)$ may not be robust for selecting μ for **AL-P1** in regularized SENSE-reconstruction problems.

5. SELECTION OF μ , ν_1 , ν_2 FOR **AL-P2**

In devising **AL-P2** in the paper, we circumvented the above issues by introducing additional splitting variables that lead to simpler matrices \mathbf{H}_{μ} , $\mathbf{H}_{\nu_1 \nu_2}$, and \mathbf{H}_{ν_2} (33)-(35), which have special structures (toeplitz, circulant and diagonal, respectively) and whose condition numbers can be computed easily and controlled individually. Moreover, in the case of **AL-P2**, the threshold-value $\tau_2 = \frac{\lambda}{\mu \nu_1}$ depends on μ and ν_1 so that a deviation of μ from a corresponding μ_{opt} for **AL-P2** can be balanced appropriately by adjusting ν_1 . We observed 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 **AL-P2** in all our experiments, so we simply set $\kappa(\mathbf{H}_{\nu_1 \nu_2}) = 12$, $\kappa(\mathbf{H}_{\nu_2}) = \min\{0.9\kappa(\tilde{\mathbf{S}}^H \tilde{\mathbf{S}}), 12\}$ universally in all our experiments (40)-(41). Figure 8 illustrates that $\kappa(\mathbf{H}_{\mu}) \in [10, 36]$ provides good convergence speeds for **AL-P2** for all data-sets even when μ_{\min} did not yield comparable results for **AL-P1**. In the paper, we universally set $\kappa(\mathbf{H}_{\mu}) = 24$ for all experiments and obtained promising results in favor of **AL-P2**. Table II summarizes our observations for **AL-P2** where we see that $\xi(\mu_0)$ is agreeably close to $\xi(\mu_{\text{opt}})$ compared to the corresponding values in Table I. Also, μ_0 and μ_{opt} are of the same order of magnitude; the deviation of μ_0 from μ_{opt} in τ_2 is compensated by ν_1 which is empirically selected (using the above condition number setting) for obtaining good convergence speeds.

ACKNOWLEDGEMENTS

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. Marko Ivancevic, Philips Healthcare, for providing the breast phantom data-set used in the experiments.