Efficient Dynamic Parallel MRI Reconstruction for the Low-Rank Plus Sparse Model

Claire Yilin Lin, *Student Member, IEEE*, and Jeffrey A. Fessler, *Fellow, IEEE*

***Abstract*—The low-rank plus sparse (L+S) decomposition model enables the reconstruction of undersampled dynamic parallel magnetic resonance imaging data. Solving for the low rank and the sparse components involves nonsmooth composite convex optimization, and algorithms for this problem can be categorized into proximal gradient methods and variable splitting methods. This paper investigates new efficient algorithms for both schemes. While current proximal gradient techniques for the L+S model involve the classical iterative soft thresholding algorithm (ISTA), this paper considers two accelerated alternatives, one based on the fast iterative shrinkage-thresholding algorithm (FISTA) and the other with the recent proximal optimized gradient method (POGM). In the augmented Lagrangian (AL) framework, we propose an efficient variable splitting scheme based on the form of the data acquisition operator, leading to simpler computation than the conjugate gradient approach required by existing AL methods. Numerical results suggest faster convergence of the efficient implementations for both frameworks, with POGM providing the fastest convergence overall and the practical benefit of being free of algorithm tuning parameters.**

***Index Terms*—Parallel magnetic resonance imaging (MRI), dynamic MRI, low-rank, sparsity, accelerated algorithms, proximal gradient method (PGM), augmented lagrangian (AL), variable splitting.**

I. Introduction

THE application of compressed sensing (CS) to Magnetic Resonance Imaging (MRI) has been extensively explored to accelerate the data acquisition process [1], [2]. In particular, since dynamic MRI data is inherently under-sampled, it is useful to use a CS-MRI model for image reconstruction. CS has also been combined with parallel MRI techniques such as SENSitivity Encoding (SENSE) [3], aiming to collect more data with multiple receiver coils, thereby possibly improving the spatio-temporal resolution trade-off of the reconstructed images. This combination is especially useful in dynamic MRI, where reconstruction of high spatial and temporal resolution is desired [4]. Compared with models that use coil-by-coil auto-calibration, [5], [6], the SENSE framework uses explicit knowledge of the sensitivity of the receiver coils.

In addition to image sparsity based on CS, the low-rank models of the space-time matrix have also been explored for dynamic MRI, based on assumptions of the similarities between temporal profiles [7]–[10]. In particular, a low-rank plus sparse (L+S) matrix decomposition assumes incoherence between a low-rank component L and a sparse component S, with L modeling the temporally correlated background, and S the dynamic information that lies on top of the background. The corresponding reconstruction problem can be formulated as a convex optimization problem, where the nuclear norm and l_1 norm are used to respectively promote low-rankness and sparsity regularization on L and S. The L+S formulation has various applications, such as motion estimation in dynamic contrast-enhanced MRI, and automated background suppression for angiography [8], [9].

One technique for solving such optimization problems involves the class of proximal gradient methods (PGM), whose iterates are based on the proximal operator [11]–[13]. In particular, [9] solves the L+S decomposition with the iterative soft thresholding algorithm (ISTA). Although accelerated variants of ISTA have been applied to various non-parallel and parallel MRI models with sparsity regularization [14]–[17], to our knowledge, fast PGM has yet to be explored for the L+S reconstruction problem.

Variable splitting is another category of optimization schemes that has been used extensively for various MRI reconstruction models, with formulation in the augmented Lagrangian (AL) framework. In single-coil dynamic MRI, variable splitting has provided efficient alternating update schemes for L+S models [7], [8]. Using the splitting of variables to decouple a cost function into simpler sub-problems, one can also apply accelerated schemes, such as the fast iterative shrinkage-thresholding algorithm (FISTA) [18], to the sub-problems, for more efficient computation [19], [20]. For the L+S model, [8] proposed a splitting scheme for which the AL function leads to sub-problems with quadratic updates. That approach requires inverting a matrix of the form $(\mathbf{E}^*\mathbf{E} + \delta I)^{-1}$, where $\mathbf{E}$ is a data acquisition operator, $\mathbf{E}^*$ is its Hermitian adjoint, and δ denotes a penalty parameter. With non-Cartesian sampling, or with multiple coils in the case of parallel MRI, a computationally demanding iterative approach like the conjugate gradient (CG) method is required for the updates. Efficient formulations of this update have been investigated, including a singular value decomposition (SVD)

Manuscript received July 9, 2018; revised October 19, 2018; accepted November 13, 2018. Date of publication November 19, 2018; date of current version February 7, 2019. The associate editor coordinating the review of this manuscript and approving it for publication was Ivan W. Selesnick. This work was supported in part by the National Institutes of Health under Grant R01 EB023618. *(Corresponding author: Claire Yilin Lin.)*

C. Y. Lin is with the Department of Mathematics, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: yilinlin@umich.edu).

J. A. Fessler is with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: fessler@umich.edu).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors. The material includes additional figures for our experiments, and an additional experimental result. This material is three pages in size.

Digital Object Identifier 10.1109/TCI.2018.2882089

of $\mathbf{E}^*\mathbf{E}$ [21], and pre-multiplication of a Fourier operator [10]. However, these implementations are still computationally intensive with multi-coil data.

For under-sampled dynamic parallel MRI, this paper presents efficient algorithms for the L+S image reconstruction problem. In particular, in the PGM category, we investigate two accelerated alternatives to ISTA, one based on FISTA, and the other the recent proximal optimized gradient method (POGM) [22]. For methods involving variable splitting, we adapt a splitting scheme that uses the matrix structures associated with the under-sampling pattern, the Fourier encoding and the sensitivity maps, leading to faster MRI reconstruction [23]–[25]. In this case, we take advantage of the L+S model structure, leading to efficient updates with only two AL variables. This paper is an extension of our previous conference work [26] that briefly investigated the two accelerated algorithms in the PGM category. Compared with this earlier work, here we discuss the algorithms in more detail, investigate another accelerated algorithm in the variable splitting scheme, and include an extension to non-Cartesian MRI in the Supplement.

The rest of this paper is organized as follows. Section II formulates the corresponding convex optimization problem, and reviews some related methods for solving it. Section III presents the efficient implementations for the two classes of algorithms. Section IV reports experimental results, followed by discussion and conclusion in Sections V and VI respectively.

II. Problem and Related Methods

In the L+S framework for dynamic MRI, the goal is to estimate an unknown image, modeled as a superposition of a low-rank component L and a sparse component S. In parallel MRI, we are provided with under-sampled k-space data $d \in \mathbb{C}^{N_s N_c}$, where N_s is the total number of samples received from each receiver coil (across all frames), and N_c is the number of coils. N_x and N_y denote the image dimensions of each image frame, and N_t is the number of time frames. The L+S formulation [9] uses the following regularized convex optimization scheme:

$$\operatorname*{argmin}_{L,S} \frac{1}{2}\|\mathbf{E}(L+S)-d\|_2^2 + \lambda_L\|L\|_* + \lambda_S\|\mathbf{T}S\|_1, \tag{1}$$

where $L, S \in \mathbb{C}^{N_x N_y \times N_t}$ are the desired dynamic image components, $\mathbf{E} : \mathbb{C}^{N_x N_y \times N_t} \rightarrow \mathbb{C}^{N_s N_c}$ is the data acquisition operator that considers the coil sensitivities and the Fourier transform with under-sampling, and $\mathbf{T} : \mathbb{C}^{N_x N_y \times N_t} \rightarrow \mathbb{C}^{N_x N_y N_t}$ a known sparsifying transform operator based on *a priori* assumptions of the domain of image sparsity. This paper considers the (unitary) temporal Fourier transform operator, with $\mathbf{T}S = (T \otimes I_{N_x N_y})\mathbf{vec}(S)$, where T is the $N_t \times N_t$ unitary temporal discrete Fourier transform matrix. This sparsifying transform has been extensively used to promote sparsity in dynamic MRI reconstruction [4]–[9]. Our accelerated methods adapt readily to other unitary operators and to 3D dynamic MRI problems. Here the data consistency is captured by the vector l_2-norm term, the low-rankness of L by the matrix nuclear norm, and the sparsity of the transformed S by the vector l_1 norm. The contributions between these three terms are balanced by the regularization parameters λ_L and λ_S.

Methods for solving the optimization problem (1) fall into two classes: those based on the proximal gradient methods, and those using AL with variable splitting. Below, we review these two methods and existing implementations for the L+S reconstruction model.

A. Conventional Proximal Gradient Scheme

To implement the classical PGM on the L+S optimization problem, we combine the two unknowns by forming a single "stacked" variable $\mathcal{X} = [\begin{smallmatrix} L \\ S \end{smallmatrix}]$. With this change, (1) can be equivalently expressed as

$$\begin{aligned}
&\min_{\mathcal{X}} g(\mathcal{X}) + h_1(\mathcal{X}) + h_2(\mathcal{X}), \quad \text{where} \\
&g(\mathcal{X}) = \frac{1}{2}\|[\mathbf{E} \quad \mathbf{E}]\mathcal{X} - d\|_2^2, \\
&h_1(\mathcal{X}) = \lambda_L\|[I \quad 0]\mathcal{X}\|_*, \quad \text{and} \quad h_2(\mathcal{X}) = \lambda_S\|[0 \quad \mathbf{T}]\mathcal{X}\|_1.
\end{aligned} \tag{2}$$

Here I and 0 denote respectively the identity and the zero matrices, of size $N_x N_y \times N_x N_y$. To verify the convergence assumptions of PGM, we note that $g(\mathcal{X})$ is a smooth, convex, and continuously differentiable function, whose gradient is Lipschitz continuous with constant $l(\nabla g)$; $h_1(\mathcal{X}), h_2(\mathcal{X})$ are continuous, convex and non-smooth functions. For arbitrary variables $\mathcal{Y}, \mathcal{Z}$, the kth iterate of the PGM is then given by the proximal operator:

$$\begin{aligned}
\mathcal{Y}_k &= \mathbf{prox}_h\left(\mathcal{Y}_{k-1} - t\nabla g(\mathcal{Y}_{k-1})\right), \quad \text{where} \\
\mathbf{prox}_h(\mathcal{Z}) &= \operatorname*{argmin}_{\mathcal{Y}} h(\mathcal{Y}) + \frac{1}{2}\|\mathcal{Y} - \mathcal{Z}\|_2^2.
\end{aligned}$$

Here t is a chosen step size, whose dependence on the Lipschitz constant $l(\nabla g)$ guarantees convergence of the algorithm.

The proximal maps for the nuclear norm in h_1 and the vector l_1 norm in h_2 have closed-form expressions. In particular, $\mathbf{prox}_{h_2}$ is given by the soft thresholding operator

$$\mathbf{\Lambda}_\lambda(\mathcal{Y}) = \text{sign}(\mathcal{Y}) \odot (|\mathcal{Y}| - \lambda)_+,$$

where $\odot$ denotes element-wise multiplication. $\mathbf{prox}_{h_1}$ is the singular value thresholding operator

$$\mathbf{SVT}_\lambda(\mathcal{Y}) = U\mathbf{\Lambda}_\lambda(\Sigma)V^*,$$

where $U\Sigma V^*$ is a singular value decomposition of $\mathcal{Y}$. Since h_1 and h_2 are functions of L and S respectively, the k-th iterate can be written separately for L_k and S_k:

$$\begin{aligned}
L_k &= \mathbf{SVT}_{\lambda_L}\left(L_{k-1} - td(\mathcal{X}_{k-1})\right), \quad \text{and} \\
S_k &= \mathbf{T}^*\Big(\Lambda_{\lambda_S}\left[\mathbf{T}\big(S_{k-1} - td(\mathcal{X}_{k-1})\big)\right]\Big), \quad \text{where} \\
d(\mathcal{X}) &= [I \quad 0]\nabla g(\mathcal{X}) = \mathbf{E}^*([\mathbf{E} \quad \mathbf{E}]\mathcal{X} - d) = [0 \quad I]\nabla g(\mathcal{X}).
\end{aligned}$$

Here $\mathbf{T}^*$ denotes the adjoint operator of $\mathbf{T}$, defined by the corresponding inverse Fourier transform. The expression of S_k uses the fact that $\mathbf{T}$ represents the unitary temporal Fourier transform, and that the l_2 norm is unitary invariant.

Since the gradient $d(\mathcal{X}_{k-1})$ is the same in both expressions of L_k and S_k, only one gradient evaluation is necessary in each iteration. The methods in [9] use this fact, exploiting computational efficiency by jointly estimating L and S. However, Section III shows that fast PGM provides much further acceleration.

B. Conventional Variable Splitting Scheme

It has been empirically observed that in some problem settings, AL-based methods can achieve higher accuracy than PGM in fewer iterations [27]. This has motivated another technique of solving (1), using variable splitting.

A splitting scheme is introduced in [8] to solve the L+S decomposition problem. In particular, (1) is re-formulated with two constraints:

$$\operatorname*{argmin}_{L,S} \min_{U,W} \frac{1}{2}\|\mathbf{E}(L+S)-d\|_2^2 + \lambda_L\|U\|_* + \lambda_S\|W\|_1 \tag{1}$$

$$\text{subject to } \begin{cases} U = L \\ W = \mathbf{T}S. \end{cases} \tag{3}$$

With this formulation, the associated modified AL function is

$$\frac{1}{2}\|\mathbf{E}(L+S)-d\|_2^2 + \lambda_L\|U\|_* + \lambda_S\|W\|_1 + \frac{\delta_1}{2}\|L-U+V_1\|_2^2 + \frac{\delta_2}{2}\|\mathbf{T}S-W+V_2\|_2^2,$$

where V_1, V_2 are Lagrange multiplier arrays, and δ_1, δ_2 are two corresponding AL penalty parameters that affect the convergence rate, but not the final estimates.

This problem can be solved by iterative updates of the four unknowns, followed by updates of the Lagrange multipliers. In particular, each update of L and S is quadratic, requiring computation of $(\mathbf{E}^*\mathbf{E}+\delta_i I)^{-1}$ for $i = 1, 2$. With single-coil Cartesian data, as considered in [8], $\mathbf{E}^*\mathbf{E}$ is circulant and one can use FFT operations for efficient computation. However, in parallel MRI, the operator $\mathbf{E}$ contains additional information of coil sensitivities, so $\mathbf{E}^*\mathbf{E}$ is not circulant, and the updates of the quadratic terms would require an iterative method like the CG approach.

Based on this observation, Section III presents a new AL algorithm that simplifies the computation by considering a more efficient variable splitting scheme for the L+S model.

III. Accelerated Algorithms

This section presents three efficient algorithms for the minimization problem (1). Two of them are in the class of PGM, with additional momentum terms in the updates that help achieve faster convergence rates. The third is an AL method that uses a different variable splitting scheme than (3), exploiting the structure of the data acquisition operator $\mathbf{E}$, improving computation efficiency for parallel MRI.

A. Proximal Gradient Scheme

The ISTA update for L and S, as given in [9], is based on classical PGM, for which the sequence of function values converges to the optimal function value at a rate of $O(1/k)$ [18]. We assume from now on that the operator $\mathbf{E}$ is normalized such that the spectral norm $\|\mathbf{E}\|_2 = 1$ for fully sampled data. Then the Lipschitz constant of $g(\cdot)$ in (2) satisfies

$$l(\nabla g) = \|[\mathbf{E} \quad \mathbf{E}]\|_2^2 = 2\|\mathbf{E}\|_2^2 \leq 2, \tag{4}$$

so ISTA converges for any step size t with $0 < t < \frac{2}{2\|\mathbf{E}\|_2^2} = 1$.

Algorithm 1: Proximal Gradient L+S.

```
Inputs:
  $d$: under-sampled k-t data
  $\mathbf{E}$: data acquisition operator
  $\mathbf{T}$: temporal Fourier transform
  $\lambda_L$: singular value threshold
  $\lambda_S$: sparsity threshold
Initialization:
  $M_0 = L_0 = \mathbf{E}^* d$, $S_0 = 0$
  additional initialization $(I)$ for FISTA or POGM
for $k = 1, 2, \ldots, N$ do
  update $\mathcal{X}_k$ by FISTA or POGM scheme $(\mathcal{X}_k)$
  update $M_k$ by FISTA or POGM scheme $(M_k)$
end for
output: $\mathcal{X}_N$
```

We now introduce two accelerated methods for (1) that have $O(1/k^2)$ convergence rates; their convergence analyses build on the work of Nesterov's fast gradient methods [28]. We use the same algorithm framework for these two accelerated L+S variants of ISTA, formulated as **Algorithm 1**. Computing the gradient is the most expensive step in each iteration; because both the L and S updates involve the same gradient expression, we jointly update them by first computing

$$\mathcal{X}_k = \begin{bmatrix} L_k \\ S_k \end{bmatrix},$$

then evaluating the gradient in a data consistency term, denoted as M_k below.

1) FISTA Update: Built upon the convergence analysis in [28], FISTA achieves the same rate of convergence of $O(1/k^2)$ [18]. In addition to the unknown $\mathcal{X}_k$ in each iteration, the FISTA update involves a secondary sequence $\widetilde{\mathcal{X}}_k = [\begin{smallmatrix} \widetilde{L}_k \\ \widetilde{S}_k \end{smallmatrix}]$ computed by adding a "momentum" term to the original sequence. This addition preserves the computational simplicity of ISTA, as the main computational effort of gradient evaluation remains unchanged from ISTA. Given the L+S framework of **Algorithm 1**, the additional FISTA initialization and updates are:

$$\begin{aligned}
(I) \quad & \widetilde{\mathcal{X}}_0 = \mathcal{X}_0, \quad \theta_0 = 1 \\
(\mathcal{X}_k) \quad & L_k = \mathbf{SVT}_{\lambda_L}(M_{k-1} - \widetilde{S}_{k-1}) \\
& S_k = \mathbf{T}^*\Big(\mathbf{\Lambda}_{\lambda_S}\big[\mathbf{T}(M_{k-1} - \widetilde{L}_{k-1})\big]\Big) \\
& \theta_k = \frac{1 + \sqrt{1 + 4\theta_{k-1}^2}}{2} \\
& \widetilde{\mathcal{X}}_k = \mathcal{X}_k + \frac{\theta_{k-1} - 1}{\theta_k}(\mathcal{X}_k - \mathcal{X}_{k-1}) \\
(M_k) \quad & M_k = \widetilde{L}_k + \widetilde{S}_k - t\mathbf{E}^*\big(\mathbf{E}(\widetilde{L}_k + \widetilde{S}_k) - d\big).
\end{aligned}$$

Based on the Lipschitz constant in (4), convergence for FISTA is guaranteed when the step size satisfies $0 < t \leq \frac{1}{2\|\mathbf{E}\|_2^2} = 0.5$.

2) POGM Update: In the smooth unconstrained setting, the recent optimized gradient method (OGM) achieves a worst-case convergence bound twice as small as that of Nesterov's fast gradient methods (FGM) [29], [30] by optimizing the choice of the coefficients that determine the step size in a first-order algorithm by minimizing a relaxed worst-case performance bound of $f(X_N) - f(X_*)$, the cost function discrepancy at the Nth iteration. This optimization problem is solved by semi-definite programming (SDP) in [31], and an analytical expression of the optimized step size is derived in [29], confirming the numerical observation that the worst-case performance is two times better than FGM's bound. OGM was shown to have optimal complexity for large-scale smooth problems in [31].

POGM extends OGM to the proximal case for nonsmooth composite problems. The numerical worst-case performance bound of POGM is twice better than that of FISTA [22]. In the L+S model, compared with the FISTA iterate, POGM introduces an additional sequence $\overline{\mathcal{X}}_k = [\frac{\overline{L}_k}{\overline{S}_k}]$, whose update involves three momentum terms. The POGM formulation is guaranteed to converge when using the same step size as in FISTA, and it again achieves the same computational simplicity as ISTA. The initialization and updates with POGM for the L+S framework are:

$$
\begin{aligned}
(I) \qquad & \widetilde{\mathcal{X}}_0 = \overline{\mathcal{X}}_0 = \mathcal{X}_0, \quad \theta_0 = \zeta_0 = 1 \\
(\mathcal{X}_k) \qquad & \widetilde{L}_k = M_{k-1} - S_{k-1} \\
& \widetilde{S}_k = M_{k-1} - L_{k-1} \\
& \theta_k = \begin{cases} \frac{1+\sqrt{1+4\theta_{k-1}^2}}{2}, & k < N \\ \frac{1+\sqrt{1+8\theta_{k-1}^2}}{2}, & k = N \end{cases} \\
& \overline{\mathcal{X}}_k = \widetilde{\mathcal{X}}_k + \frac{\theta_{k-1} - 1}{\theta_k}(\widetilde{\mathcal{X}}_k - \widetilde{\mathcal{X}}_{k-1}) \\
& \quad + \frac{\theta_{k-1}}{\theta_k}(\widetilde{\mathcal{X}}_k - \mathcal{X}_{k-1}) + \frac{\theta_{k-1} - 1}{\zeta_{k-1}\theta_k} t(\overline{\mathcal{X}}_{k-1} - \mathcal{X}_{k-1}) \\
& \zeta_k = t\left(1 + \frac{\theta_{k-1} - 1}{\theta_k} + \frac{\theta_{k-1}}{\theta_k}\right) \\
& L_k = \mathbf{SVT}_{\lambda_L}(\overline{L}_k) \\
& S_k = \mathbf{T}^*\Big(\mathbf{\Lambda}_{\lambda_S}\left[\mathbf{T}(\overline{S}_k)\right]\Big) \\
(M_k) \qquad & M_k = L_k + S_k - t\mathbf{E}^*\big(\mathbf{E}(L_k + S_k) - d\big).
\end{aligned}
$$

The empirical results in Section IV show that POGM converges faster than ISTA and FISTA, yet requires essentially the same computation time per iteration (dominated by the M_k update needed in all methods).

B. Variable Splitting Scheme

We now consider variable splitting methods for the L+S reconstruction problem for parallel MRI. In this setting, the data acquisition operator is $\mathbf{E} = \mathbf{\Omega} Q \mathbf{C}$, where $\mathbf{\Omega} : \mathbb{C}^{N_x N_y N_t N_c} \to \mathbb{C}^{N_s N_c}$ contains the under-sampling patterns for all frames, $Q \in \mathbb{C}^{N_x N_y N_t N_c \times N_x N_y N_t N_c}$ represents a Fourier encoding matrix, and $\mathbf{C} : \mathbb{C}^{N_x N_y \times N_t} \to \mathbb{C}^{N_x N_y N_t N_c}$ captures the sensitivity maps of the receiver coils [23]–[25]. While direct extension of the splitting scheme in [23] to the L+S model leads to at least four more variables in the AL function, here we make use of the L and S formulations, and introduce only two AL variables to capture the constrained cost function. For simpler formulation of the algorithm, we assume from now on that $\mathbf{C}$ is normalized such that $\mathbf{C}^*\mathbf{C} = \mathbf{I}$ (identity). This normalization is valid since our model considers sparsity with temporal Fourier transform $\mathbf{T}$, and the spatial scaling does not affect the rank of the low-rank component. After reconstruction, one can undo the image scaling if needed.

With this expression, we represent the following novel reformulation of (1) in the constrained form

$$
\begin{aligned}
& \underset{L,S}{\operatorname{argmin}} \min_{Z,X} \frac{1}{2}\|\mathbf{\Omega} Z - d\|_2^2 + \lambda_L \|L\|_* + \lambda_S \|\mathbf{T} S\|_1 \\
& \text{subject to } \begin{cases} Z = Q\mathbf{C}X \\ X = L + S. \end{cases} \qquad (5)
\end{aligned}
$$

Compared with (3), this splitting scheme also involves four variable updates, but leads to simpler updates, as shown next.

The modified AL function corresponding to (5) is

$$
\begin{aligned}
& \frac{1}{2}\|\mathbf{\Omega} Z - d\|_2^2 + \lambda_L \|L\|_* + \lambda_S \|\mathbf{T} S\|_1 \\
& \quad + \frac{\delta_1}{2}\|Z - Q\mathbf{C}X + V_1\|_2^2 + \frac{\delta_2}{2}\|X - (L + S) + V_2\|_2^2.
\end{aligned}
$$

The L update involves the nuclear norm, and its proximal map is given by singular value thresholding:

$$
\begin{aligned}
& \underset{L}{\operatorname{argmin}}\, \lambda_L \|L\|_* + \frac{\delta_2}{2}\|X - (L + S) + V_2\|_2^2 \\
& = \mathbf{SVT}_{\lambda_L/\delta_2}(X - S + V_2). \qquad (6)
\end{aligned}
$$

The S update contains a vector l_1-norm term, whose proximal operator is soft thresholding, where we use the fact that $\mathbf{T}$ is a unitary operator, with the change of variables $\tilde{S} = \mathbf{T} S$:

$$
\begin{aligned}
& \underset{S}{\operatorname{argmin}}\, \lambda_S \|\mathbf{T} S\|_1 + \frac{\delta_2}{2}\|X - (L + S) + V_2\|_2^2 \\
& = \mathbf{T}^*\Big(\underset{\tilde{S}}{\operatorname{argmin}}\, \lambda_S \|\tilde{S}\|_1 + \frac{\delta_2}{2}\|\mathbf{T}(X - L + V_2) - \tilde{S}\|_2^2\Big) \\
& = \mathbf{T}^* \mathbf{\Lambda}_{\lambda_S/\delta_2}\big(\mathbf{T}(X - L + V_2)\big). \qquad (7)
\end{aligned}
$$

The updates for Z and X involve quadratic terms:

$$
\begin{aligned}
& \underset{Z}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{\Omega} Z - d\|_2^2 + \frac{\delta_1}{2}\|Z - Q\mathbf{C}X + V_1\|_2^2 \\
& = (\mathbf{\Omega}^*\mathbf{\Omega} + \delta_1 \mathbf{I})^{-1}\big(\mathbf{\Omega}^* d + \delta_1(Q\mathbf{C}X - V_1)\big), \qquad (8)
\end{aligned}
$$

```
Algorithm 2: Variable Splitting L+S.
Inputs:
  $d$: under-sampled multi-coil k-t data
  $\mathbf{\Omega}$: under-sampling mask
  $Q$: Fourier encoding operator
  $\mathbf{C}$: coil sensitivity maps
  $\mathbf{T}$: temporal Fourier tranform
  $\lambda_L$: singular value threshold
  $\lambda_S$: sparsity threshold
  $\delta_1, \delta_2$: AL penalty parameters
Initialization: $X_0 = L_0 = \mathbf{C}^* Q^* \mathbf{\Omega}^* d$,
  $S_0 = V_{1,0} = V_{2,0} = 0$
for $k = 1, 2, \ldots, N$ do
  compute $Z_k$ by efficient inverse (8)
  compute $X_k$ by efficient inverse (9)
  compute $L_k$ by singular value thresholding (6)
  compute $S_k$ by soft thresholding (7)
  $V_{1,k} \leftarrow V_{1,k-1} + (Z_k - Q\mathbf{C}X_k)$
  $V_{2,k} \leftarrow V_{2,k-1} + \big(X_k - (L_k + S_k)\big)$
end for
output: $L_N, S_N$
```

$$
\begin{aligned}
&\operatorname*{argmin}_X \frac{\delta_1}{2}\|Z - Q\mathbf{C}X + V_1\|_2^2 + \frac{\delta_2}{2}\|X - (L+S) + V_2\|_2^2 \\
&= \left(\mathbf{C}^*\mathbf{C} + \frac{\delta_2}{\delta_1}\mathbf{I}\right)^{-1} \left(\mathbf{C}^* Q^*(Z + V_1) + \frac{\delta_2}{\delta_1}(L + S - V_2)\right) \\
&= \frac{\delta_1}{\delta_1 + \delta_2}\left(\mathbf{C}^* Q^*(Z + V_1) + \frac{\delta_2}{\delta_1}(L + S - V_2)\right), \qquad (9)
\end{aligned}
$$

where we use the fact that Q is the unitary Fourier encoding matrix, and that $\mathbf{C}^*\mathbf{C} = \mathbf{I}$ by assumption.

Compared with the splitting scheme in (3), which involves the inverse $(\mathbf{E}^*\mathbf{E} + \delta_i\mathbf{I})^{-1}$, our proposed variable splitting scheme in (5) only involves computing $(\mathbf{\Omega}^*\mathbf{\Omega} + \delta_1\mathbf{I})^{-1}$. Representing the under-sampling mask matrix as a Kronecker product $\mathbf{\Omega} = I_{N_c} \otimes \widetilde{\mathbf{\Omega}}$, we note that $\widetilde{\mathbf{\Omega}}^*\widetilde{\mathbf{\Omega}}$ is diagonal, hence the inverse $(\mathbf{\Omega}^*\mathbf{\Omega} + \delta_1\mathbf{I})^{-1}$ is easy to compute.

Algorithm 2 summarizes the implementation of these updates, as well as updates for the updates of the Lagrange multipliers.

IV. Results

To compare the algorithms, we first performed experiments on two dynamic MRI datasets examined in [9]. Each dataset includes Cartesian under-sampled multi-coil data d, the k-space under-sampling mask $\mathbf{\Omega}$, and coil sensitivity maps $\mathbf{C}$. We compared the results of the three accelerated algorithms with ISTA [9] and the AL-based method that requires CG for parallel MRI [8]. We then tested our methods on the physiologically improved nonuniform cardiac torso (PINCAT) numerical phantom used in [7]. In this case, the data is under-sampled with a pseudo-radial scheme, as in the original implementation [32]. To compare algorithms in the parallel MRI setting, we included simulated coil sensitivity maps based on [33], using the Michigan Image Reconstruction Toolbox (MIRT) [34]. In the Supplement, we also explore a non-Cartesian MRI dataset from [9], where we compare methods in the PGM scheme. All our experiments used MATLAB R2018a, with a 2.7-GHz dual-core Intel Core i5. The MATLAB code that reproduces the experiments with our efficient algorithms will be available as part of the MIRT.

For each of the three datasets, we kept the regularization parameters λ_L, λ_S consistent for all algorithms. For the *in vivo* data, we set them to align as closely as possible with those in the original code provided by [9]; Section V discusses further details of this procedure. Similarly, we set a stopping criterion for our ISTA implementation that provides analogous results to the reconstructed images in [9]. To ensure fast convergence, we used a step size t of 0.99 for ISTA, and 0.5 for FISTA and POGM, as provided by the convergence theory of those methods. In addition, FISTA and POGM used an adaptive restart scheme [35]; we explored both the function and the gradient restart schemes, and report the results with the function scheme due to its slightly faster convergence with both datasets. We tuned the penalty parameters δ_1, δ_2 for the AL-based methods by sweeping across a range of values and choosing the ones that achieve the fastest convergence among them. For both datasets, we applied 3 inner CG iterations for each outer iteration of the AL scheme (3), with warm-starting; i.e., each CG call starts with the estimate from the previous AL iteration. No such inner iterations are needed for the proposed AL approach (5). We examine convergence rate by computing the normalized root-mean-squared difference (NRMSD) of each iterate to a converged image, defined by $\|X_k - X_\infty\|_2 / \|X_\infty\|_2$, where $X_k = L_k + S_k$, and $\|\cdot\|_2$ denotes the vector l_2 norm. We obtained $X_\infty = L_\infty + S_\infty$ as a reference by averaging $X_\infty^{\text{AL-2}}$ and X_∞^{POGM}, as discussed below for each dataset, then computed the distance to the minimizer.

A. Cardiac Perfusion Dataset

Images for this dataset have size $N_x \times N_y = 128 \times 128$, with $N_t = 40$ temporal profiles and $N_c = 12$ coils. Data were retrospectively under-sampled by a factor of 10, using the sampling pattern from [9], with fully sampled low spatial frequencies and low-density-sampled outer k-space. We used $\lambda_L = \lambda_S = 0.01$ as in [9], with scalings to match the original implementation, as discussed below in Section V. For the variable splitting framework, the penalty parameters were empirically tuned to achieve fast convergence, with $\delta_1 = \delta_2 = 1/5$ for AL with CG, and $\delta_1 = 1/5, \delta_2 = 1/50$ for the efficient AL method. We ran the efficient AL and the POGM implementations for 24,000 seconds to obtain $X_\infty^{\text{AL-2}}$ and X_∞^{POGM}. In this case, the cost function values $f_\infty^{\text{AL-2}}$ and f_∞^{POGM} are within 10^{-16} relative difference from each other, and we averaged the results to obtain X_∞ and f_∞. In this case, the NRMSD between $X_\infty^{\text{AL-2}}$ and X_∞^{POGM} is approximately $8.9 \times 10^{-13}\%$. As shown in Fig. 1, POGM converges the fastest overall. ISTA converges faster than the other three methods at the beginning, a phenomenon that could be due to the chosen step size. FISTA converges faster than the AL-based methods in this case, with the CG implementation being the slowest of all. The supplement contains additional figures showing the long-run behavior of the algorithms.

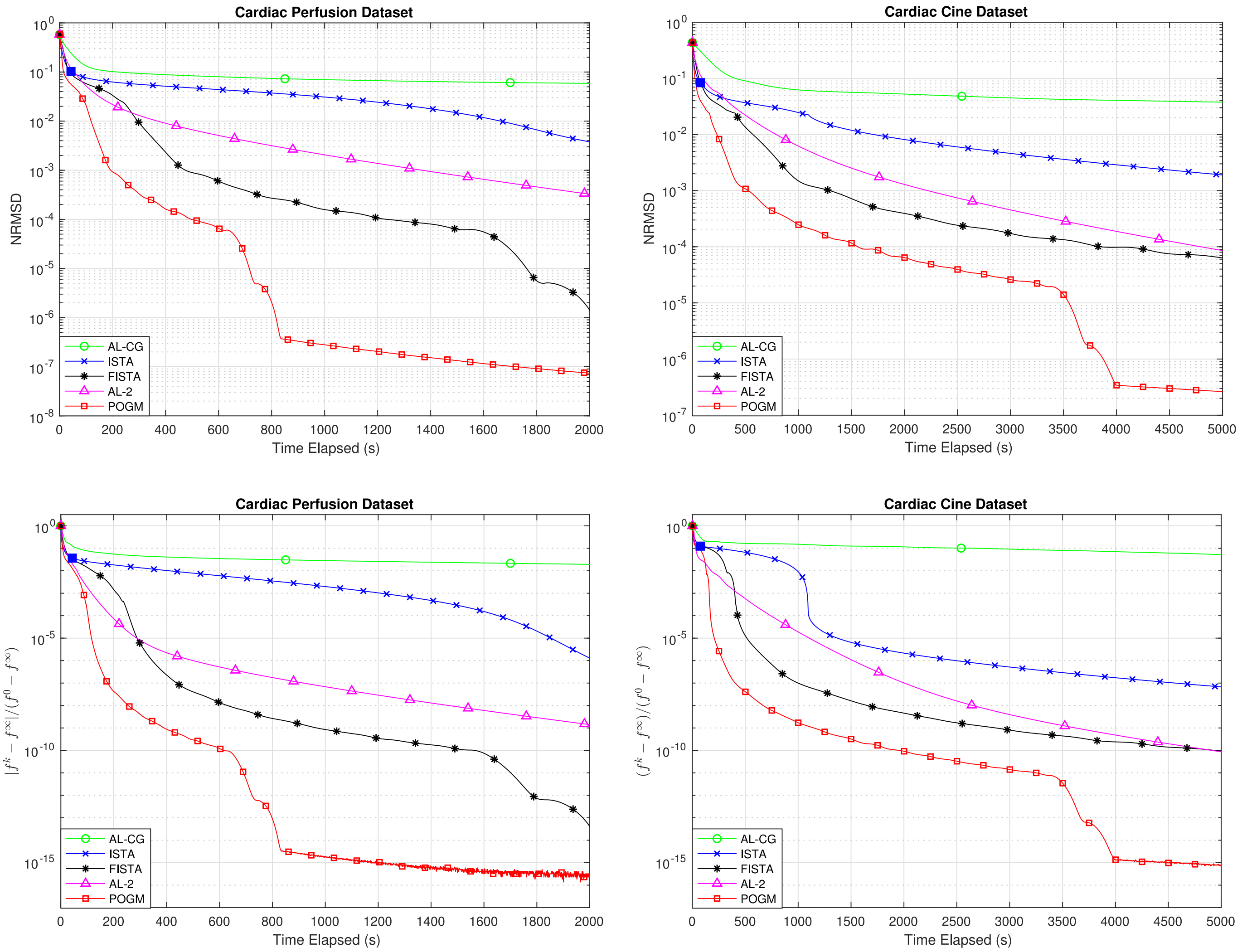

Fig. 1. Convergence of the five algorithms for the two datasets, in terms of NRMSD to the minimizer (top row), as well as the cost function values (bottom row). AL-CG and AL-2 refer respectively to the AL methods with CG (3) and with **Algorithm 2** implementations. Every 100th iteration is marked by a dot, indicating their relative speeds. The blue square markers show when ISTA reaches the approximate stopping criteria corresponding to the implementation in [9].

For this dataset, ISTA reached its stopping criterion from [9] at $k = 53$ iterations, after 30 seconds of elapsed compute time. Fig. 2 shows the magnitude of the reconstructed X_∞, as well as images of all 5 algorithms at 30 seconds time elapsed.

To help visualize the reconstructed image in the spatial-temporal domain, the Supplement includes y-t images for a selected y-slice in the center, with comparison to the fully sampled case. The AL-2 and the POGM updates provide significantly faster convergence than the other methods in their corresponding algorithmic schemes.

B. Cardiac Cine Dataset

This dataset corresponds to images of size 256 × 256, with 24 temporal frames and 12 coils, and a retrospective under-sampling factor of 8. As in the cadiac perfusion case, we used $\lambda_L = 0.01, \lambda_S = 0.0025$ with additional scalings. For the AL-based methods, we used $\delta_1 = 1/10, \delta_2 = 1/20$ for AL-CG, and $\delta_1 = 1/10, \delta_2 = 1/100$ for AL-2. To obtain X_∞ and f_∞, we ran the efficient AL and the POGM implementations for 30,000 seconds and averaged the results. The cost function values are within 10^{-16} relative difference from each other, and the NRMSD between $X_\infty^{\text{AL-2}}$ and X_∞^{POGM} is $1.4 \times 10^{-7}\%$. Fig. 1 illustrates that AL-2 achieving faster convergence than the FISTA update, but slower than POGM. Fig. 3 shows results with the same run time cut-off of 48 seconds, with ISTA taking $k = 30$ iterations to reach the stopping criterion. We again observed superior rates of convergence of the efficient implementations, in both the proximal gradient and the variable splitting schemes.

C. PINCAT Phantom Dataset

The ground truth phantom data provided by [7] has spatial dimension 128 × 128 with 50 temporal frames. To compare the algorithms in the multi-coil setting, we added simulated coil sensitivity maps of 32 coils (4 rings of 8 coils), with coil compression to reduce to $N_c = 8$ coils. Following the setup in [7], we applied a pseudo-radial under-sampling mask $\mathbf{\Omega}$, i.e., a Cartesian trajectory that closely approximates a radial trajectory, with 24 spokes per frame, corresponding to a acceleration

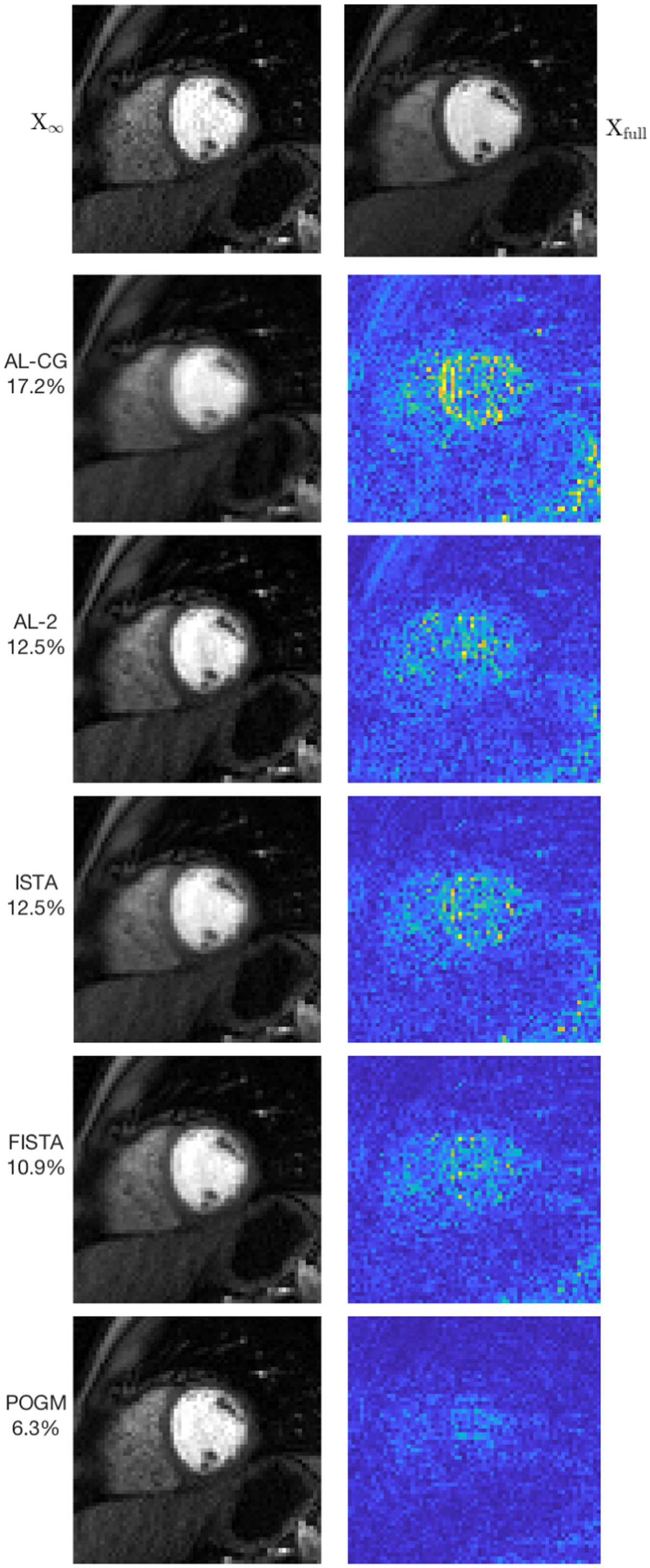

Fig. 2. First row: X_∞, taken as the average of $X_\infty^{\text{AL-2}}$ and X_∞^{POGM}, and X_{full}, reconstructed using fully sampled data without regularization. Left column: reconstructed images on a scale of [0, 1], of one temporal frame of the Cardiac Perfusion Dataset, after the closest run time less than when ISTA reaches its stopping criterion (30 seconds). Right column: the residual images are plotted on a scale of [0, 0.2], with their corresponding NRMSD shown on the left of each row.

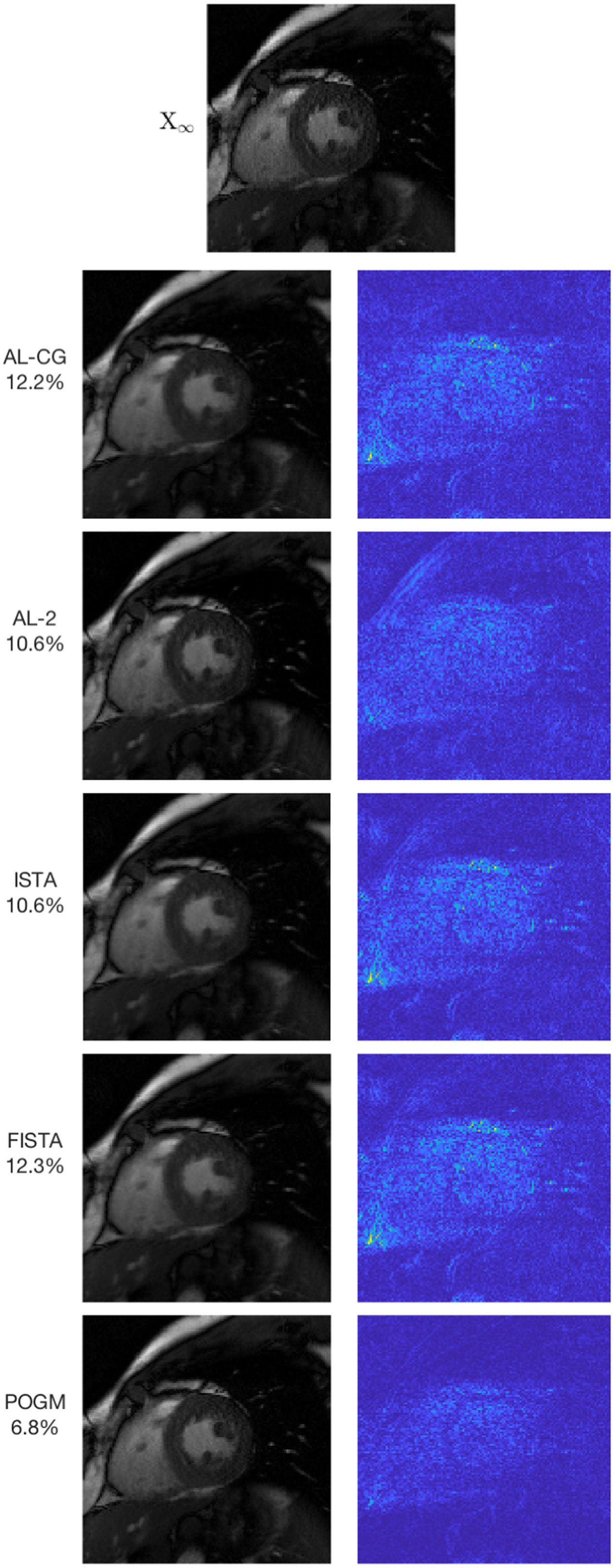

Fig. 3. First row: X_∞, taken as the average of $X_\infty^{\text{AL-2}}$ and X_∞^{POGM}. Left column: reconstructed images on a scale of [0, 1], on one temporal frame for the Cardiac Cine Dataset, after the closest run time less than when ISTA reaches its stopping criterion (48 seconds). Right column: the residual images are plotted on a scale of [0, 0.2], with their corresponding NRMSD shown on the left of each row.

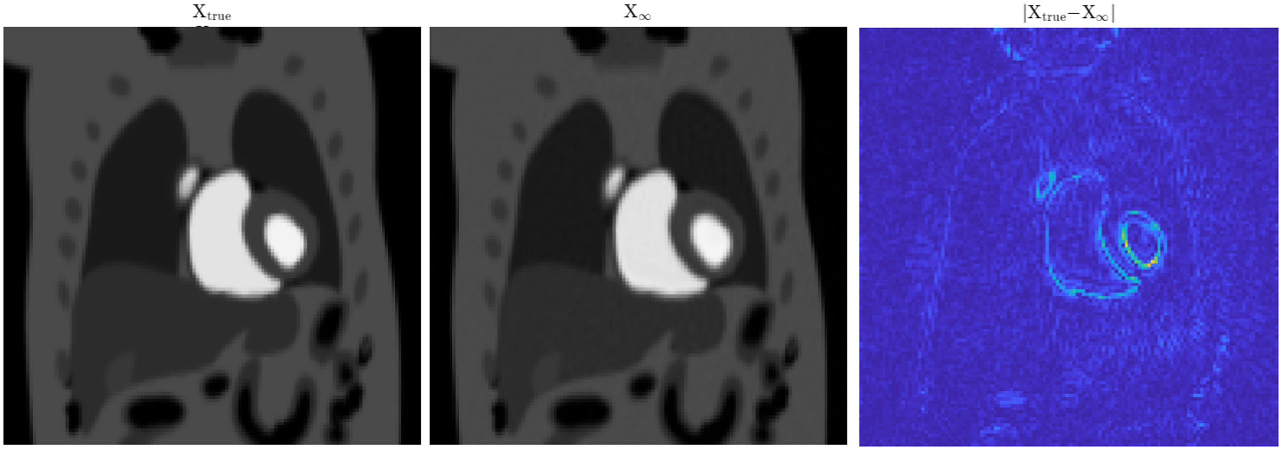

Fig. 4. From left to right: X_{true} of one temporal frame from the ground truth PINCAT phantom data, and its corresponding reconstruction X_∞ as the average of $X_\infty^{\text{AL-2}}$ and X_∞^{POGM}, all on a scale of [0, 1]. The residual image $|X_{\text{true}} - X_\infty|$ is on a scale of [0, 0.1].

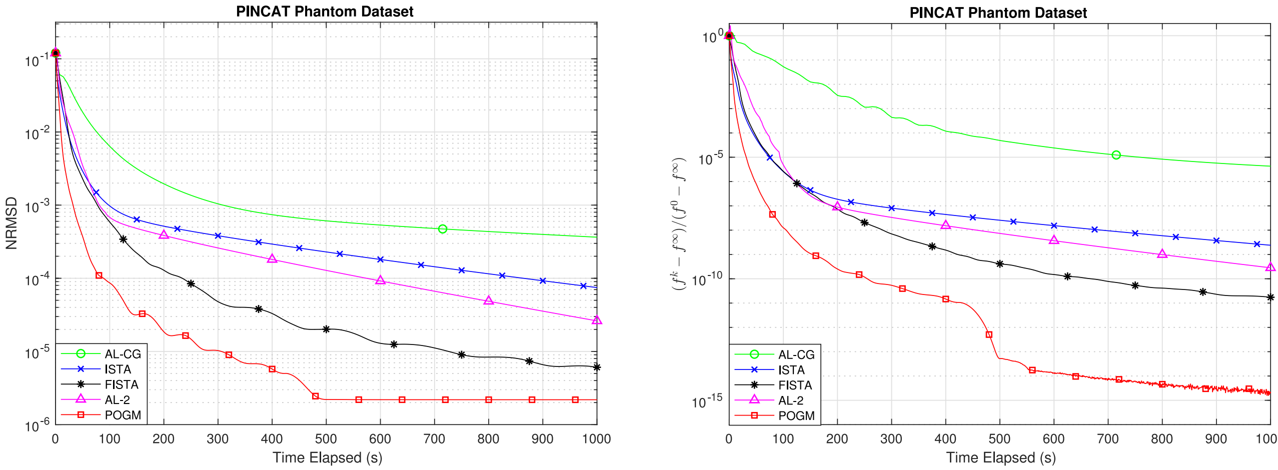

Fig. 5. Convergence of the five algorithms for the PINCAT phantom dataset, in terms of NRMSD and cost to the minimizer. Every 100th iteration of each algorithm is marked by a dot, indicating their relative speeds.

factor of 128/24 ≈ 5.3. We added zero mean Gaussian noise such that the signal to noise ratio is 46 dB. We tuned the regularization parameters λ_L to 0.0025 multiplied by the top singular value of L_0, and λ_S to 0.05, each divided by a constant that captures the square root of the sum of squares of the coil sensitivitiy maps before the normalization $\mathbf{C}^*\mathbf{C} = \mathbf{I}$. The penalty parameters in the AL-based methods were empirically tuned to achieve fast convergence, with $\delta_1 = \delta_2 = 1/3$ for AL-CG, and $\delta_1 = 1/5, \delta_2 = 1/20$ for AL-2. We ran AL-2 and POGM for 30,000 seconds, and averaged the results to obtain X_∞ and f_∞. The NRMSD between $X_\infty^{\text{AL-2}}$ and X_∞^{POGM} is approximately $4.4 \times 10^{-4}\%$, and the cost function values $f_\infty^{\text{AL-2}}$ and f_∞^{POGM} are within 10^{-12} relative difference from each other. The convergence behavior is similar to the *in vivo* case; Fig. 5 demonstrates again the superior convergence speed of AL-2 and POGM in the two schemes. For unknown reasons, AL-2 reaches a final cost that is about 10^{-10} higher than the proximal algorithms. This behavior is unimportant practically but still somewhat curious; it is unique to the PINCAT data. To visualize the results, Fig. 4 shows images of the ground truth X_{true} and the under-sampled reconstruction $X_\infty = L_\infty + S_\infty$. See the Supplement for y-t images compared to the ground truth.

V. Discussion

A. Alternative Variable Splittings

With under-sampled multi-coil data, we have expressed the data acquisition operator as $\mathbf{E} = \mathbf{\Omega} Q \mathbf{C}$, and our proposed AL approach (5) splits the sampling mask $\mathbf{\Omega}$ from the Fourier encoding together with the coil sensitivity maps $Q\mathbf{C}$. An alternative is to split the Fourier encoding operator with under-sampling, $\mathbf{\Omega} Q$, from the coil sensitivities $\mathbf{C}$, as proposed in [23]. Compared with (5), this splitting introduces a slight variation:

$$\min_{Z,X,L,S} \frac{1}{2}\|\mathbf{\Omega} Q Z - d\|_2^2 + \lambda_L \|L\|_* + \lambda_S \|\mathbf{T}S\|_1$$

$$\text{subject to} \begin{cases} Z = \mathbf{C}X, \\ X = L + S. \end{cases}$$

In this case, the update for Z involves the inverse of $Q^*\mathbf{\Omega}^*\mathbf{\Omega}Q + \delta_1 I$, which is block circulant and can be diagonalized by pre- and post-multiplication by FFT operators [25]. Hence with this splitting, efficient implementation of the inverse is again possible. We chose to implement the splitting (5) because of the simpler expressions (8), (9) for the Z and X updates, with diagonal matrix inverses due to the unitary property of Q. This leads to simpler updates with only inversion of diagonal matrices.

Although the proposed AL scheme shows empirical convergence, it does not have analytical convergence guarantee as in generalized Alternating Direction Method of Multipliers (ADMM). To compare the variable splitting (5) to the ADMM scheme, we note that it is equivalent to a formulation in the monotropic programming framework [36]:

$$\min_U f(U)$$
$$\text{subject to } AU = b,$$

$$\text{where } U = \begin{bmatrix} Z \\ X \\ L \\ S \end{bmatrix}, \; A = \begin{bmatrix} I & -\mathbf{QC} & 0 & 0 \\ 0 & I & -I & -I \end{bmatrix}, \text{ and}$$

$b = 0$. Since the last two columns of A are linearly dependent, this splitting scheme does not satisfy the sufficient conditions for the convergence guarantee of ADMM [36]. To satisfy those conditions, one could introduce an alternative variable splitting that meets the convergence criteria of ADMM, but at the cost of more variables, and thus potentially slightly slower convergence [25]. We did not investigate that approach here since AL-2 empirically converged well, despite not satisfying the sufficient conditions in [36]. In addition, POGM converged faster, and is practically preferable because it does not require any AL-type tuning parameters.

B. ISTA Implementation in Comparison With [9]

Our ISTA implementation is based on the algorithm discussed mathematically in [9]. However, we did not directly use the MATLAB code provided for [9] at http://cai2r.net/resources/software/ls-reconstruction-matlab-code, because that code is slightly inconsistent with the math in [9].

The first inconsistency is the implementation of the operator $\mathbf{E}$. The MATLAB code for the Hermitian adjoint $\mathbf{E}^*$ operation contains an additional division by the sum of squares of the coil sensitivity maps that causes inconsistency between the forward and the adjoint operations, preventing convergence to the minimizer of the stated cost function. We modified the code so that $\mathbf{E}^*$ is the exact adjoint of $\mathbf{E}$, so that all of the algorithms, including ISTA, can converge to the same cost function. To obtain similar images as those in [9], we pre-process by dividing the given coil sensitivities $\mathbf{C}$ by the square root of its sum of squares. Since this factor is close to being constant across the image, we absorb it into the regularization parameters λ_L and λ_S, to ensure a consistent setup with [9].

Another implementation difference involves the singular value threshold. In the cost function (1), the nuclear norm regularization parameter λ_L is a fixed constant, but in the provided code, λ_L changes across iterations, with a factor that depends on the leading singular value of L. This “moving target” cost function would make it impossible to compare the convergence rates of different algorithms. To ensure fair comparison of all the algorithms, while maintaining similar overall regularization as in [9], we fix λ_L by considering the leading singular value of L_∞, produced by running the original implementation until convergence.

The provided implementation has stopping criteria based on the maximum number of iterations and the tolerance of the change in updates. With the above modifications, we stop our ISTA implementation when it reaches the same cost function value as at the stopped points, and compare the NRMSD at these points with other algorithms, as indicated by the blue square markers in Fig. 1.

VI. Conclusion

This paper presents efficient algorithms for the L+S reconstruction of dynamic parallel MRI. Within the proximal gradient category, in place of using ISTA to solve the optimization problem, we consider updates by FISTA and POGM. Both methods can be efficiently formulated within the L+S framework, preserving the computational simplicity of the original ISTA implementation. Experiments with two cardiac datasets in [9] and a phantom dataset in [7] verify their accelerated rates of convergence.

For AL-based approaches, we also proposed an efficient variable splitting scheme that considers the structure of the data acquisition operator. In particular, we split the variables based on the under-sampling mask, the Fourier transform operator and the coil sensitivity maps. While the existing splitting scheme for the L+S model requires CG approach to solve for the quadratic updates [8], our proposed formulation leads to a diagonal matrix inverse that can be easily computed. Numerical experiments again confirms its superior convergence rate, compared with the existing implementation.

Although there is no strict convexity guarantee for the L+S optimization problem, our experimental results suggest high similarities between the reconstructed images by the AL and the PGM schemes, due to the observed low NRMSD of both $X_\infty^{\text{AL-2}}$ and X_∞^{POGM}. In the implementation perspective, however, AL-based methods in the L+S model requires the tuning of two additional penalty parameters, whereas POGM has no extra tuning parameters. This practical benefit, combined with the empirical faster convergence of POGM seen in the examples, make POGM our recommended approach for solving L+S reconstruction problems for dynamic MRI.

Acknowledgment

The authors thank D. Noll and M. Muckley for discussions of L+S models for fMRI data. The authors also thank R. Otazo for sharing the cardiac MRI data.

References

[1] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magn. Res. Med.*, vol. 58, no. 6, pp. 1182–95, Dec. 2007.

[2] J. P. Haldar, D. Hernando, and Z.-P. Liang, "Compressed-sensing MRI with random encoding," *IEEE Trans. Med. Imag.*, vol. 30, no. 4, pp. 893–903, Apr. 2011.

[3] K. P. Pruessmann, M. Weiger, M. B. Scheidegger, and P. Boesiger, "SENSE: Sensitivity encoding for fast MRI," *Magn. Reson. Med.*, vol. 42, no. 5, pp. 952–962, Nov. 1999.

[4] H. Jung, K. Sung, K. S. Nayak, E. Y. Kim, and J. C. Ye, "k-t FOCUSS: A general compressed sensing framework for high resolution dynamic MRI," *Magn. Reson. Med.*, vol. 61, no. 1, pp. 103–116, Jan. 2009.

[5] M. A. Griswold *et al.*, "Generalized autocalibrating partially parallel acquisitions (GRAPPA)," *Magn. Reson. Med.*, vol. 47, no. 6, pp. 1201–1210, Jun. 2002.

[6] M. Lustig and J. M. Pauly, "SPIRiT: Iterative self-consistent parallel imaging reconstruction from arbitrary k-space," *Magn. Reson. Med.*, vol. 64, no. 2, pp. 457–471, Aug. 2010.

[7] S. G. Lingala, Y. Hu, E. V. R. DiBella, and M. Jacob, "Accelerated dynamic MRI exploiting sparsity and low-rank structure: k-t SLR," *IEEE Trans. Med. Imag.*, vol. 30, no. 5, pp. 1042–1054, May 2011.

[8] B. Trémoulhéac, N. Dikaios, D. Atkinson, and S. R. Arridge, "Dynamic MR image reconstruction–separation from undersampled (k, t)-space via low-rank plus sparse prior," *IEEE Trans. Med. Imag.*, vol. 33, no. 8, pp. 1689–1701, Aug. 2014.

[9] R. Otazo, E. Candès, and D. K. Sodickson, "Low rank plus sparse matrix decomposition for accelerated dynamic MRI with separation of background and dynamic components," *Magn. Res. Med.*, vol. 73, no. 3, pp. 1125–1136, Mar. 2015.

[10] S. F. Roohi, D. Zonoobi, A. A. Kassim, and J. L. Jaremko, "Multi-dimensional low rank plus sparse decomposition for reconstruction of under-sampled dynamic MRI," *Pattern Recognit.*, vol. 63, pp. 667–679, Mar. 2017.

[11] P. L. Combettes and J. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*. Berlin, Germany: Springer, 2011, pp. 185–212.

[12] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, vol. 408. Berlin, Germany: Springer, 2011.

[13] N. Parikh and S. Boyd, "Proximal algorithms," in *Foundations and Trends® in Optimization*, vol. 1, no. 3. Delft, The Netherlands: Now Publishers, 2014, pp. 127–239.

[14] Ç. Bilen *et al.*, "On compressed sensing in parallel MRI of cardiac perfusion using temporal wavelet and TV regularization," in *Proc. IEEE In. Conf. Acoustics, Speech, Signal Process.*, 2010, pp. 630–633.

[15] L. B. Montefusco, D. Lazzaro, S. Papi, and C. Guerrini, "A fast compressed sensing approach to 3D MR image reconstruction," *IEEE Trans. Med. Imag.*, vol. 30, no. 5, pp. 1064–1075, May 2011.

[16] N. Cai, W. Xie, Z. Su, S. Wang, and D. Liang, "Sparse parallel MRI based on accelerated operator splitting schemes," *Comput. Math. Methods Med.*, vol. 2017, 2017, Art. no. 1724630.

[17] J. Huang, L. Zhu, L. Wang, and W. Song, "Compressed sensing MRI using sparsity averaging and FISTA," *Appl. Mag. Res.*, vol. 48, no. 8, pp. 749–760, Aug. 2017.

[18] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[19] X. Ye, Y. Chen, W. Lin, and F. Huang, "Fast MR image reconstruction for partially parallel imaging with arbitrary k-space trajectories," *IEEE Trans. Med. Imag.*, vol. 30, no. 3, pp. 575–585, Mar. 2011.

[20] S. Kamesh Iyer , T. Tasdizen, D. Likhite, and E. DiBella, "Split Bregman multicoil accelerated reconstruction technique: A new framework for rapid reconstruction of cardiac perfusion MRI," *Med. Phys.*, vol. 43, no. 4, pp. 1969–1981, Apr. 2016.

[21] Ç. Bilen, Y. Wang, and I. W. Selesnick, "High-speed compressed sensing reconstruction in dynamic parallel MRI using augmented Lagrangian and parallel processing," *IEEE Trans. Emerg. Sel. Topics Circuits Syst.*, vol. 2, no. 3, pp. 370–379, Sep. 2012.

[22] A. B. Taylor, J. M. Hendrickx, and F. Glineur, "Exact worst-case performance of first-order methods for composite convex optimization," *SIAM J. Optim.*, vol. 27, no. 3, pp. 1283–1313, Jul. 2017.

[23] S. Ramani and J. A. Fessler, "Parallel MR image reconstruction using augmented Lagrangian methods," *IEEE Trans. Med. Imag.*, vol. 30, no. 3, pp. 694–706, Mar. 2011.

[24] Y. Chen, W. Hager, F. Huang, D. Phan, X. Ye, and W. Yin, "Fast algorithms for image reconstruction with application to partially parallel MR imaging," *SIAM J. Imag. Sci.*, vol. 5, no. 1, pp. 90–118, Jan. 2012.

[25] M. Le, and J. A. Fessler, "Efficient, convergent SENSE MRI reconstruction for nonperiodic boundary conditions via tridiagonal solvers," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 11–12, Mar. 2017.

[26] C. Y. Lin and J. A. Fessler, "Accelerated methods for low-rank plus sparse image reconstruction," in *Proc. IEEE Int. Symp. Biomed. Imag.*, 2018, pp. 48–51.

[27] E. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 3, pp. 1–37, May 2011.

[28] Y. Nesterov, "A method of solving a convex programming problem with convergence rate $O(1/k^2)$," *Sov. Math. Doklady*, vol. 27, no. 2, pp. 372–376, Feb. 1983.

[29] D. Kim and J. A. Fessler, "Optimized first-order methods for smooth convex minimization," *Math. Program.*, vol. 159, no. 1–2, pp. 81–107, Sep. 2016.

[30] D. Kim and J. A. Fessler, "On the convergence analysis of the optimized gradient method," *J. Optim. Theory Appl.*, vol. 172, no. 1, pp. 187–205, Jan. 2017.

[31] Y. Drori and T. Teboulle, "Performance of first-order methods for smooth convex minimization: A novel approach," *Math. Program.*, vol. 145, no. 1–2, pp. 451–482, Jun. 2014.

[32] "Matlab codes for k-t SLR," 2011. [Online]. Available: https://research.engineering.uiowa.edu/cbig/content/matlab-codes-k-t-slr, Accessed on: May 20, 2018.

[33] M. I. Grivich and D. P. Jackson, "The magnetic field of current-carrying polygons: An application of vector field rotations," *Amer. J. Phys.*, vol. 68, no. 5, pp. 469–474, May 2000.

[34] "Michigan Image Reconstruction Toolbox (MIRT)," [Online]. Available: http://web.eecs.umich.edu/fessler/code. Accessed on: May 20, 2018.

[35] B. O'Donoghue and E. Candès, "Adaptive restart for accelerated gradient schemes," *Found. Comput. Math.*, vol. 15, no. 3, pp. 715–732, Jun. 2014.

[36] J. Eckstein, "Parallel alternating direction multiplier decomposition of convex programs," *J. Optim. Theory Appl.*, vol. 80, no. 1, pp. 39–62, Jan. 1994.

Authors' photographs and biographies not available at the time of publication.

Supplementary Material for Efficient Dynamic Parallel MRI Reconstruction for the Low-Rank Plus Sparse Model

Claire Yilin Lin, *Student Member, IEEE,* and Jeffrey A. Fessler, *Fellow, IEEE*

This supplement presents additional figures and experiment that could not be included in the main body of the manuscript [1] due to page restrictions.

VIII. y-t Images

To help visualize the reconstruction results for different time frames, we provide figures of the reconstructed images in the y-t domain, compared to their reference images, as shown in Fig. 1 and 2.

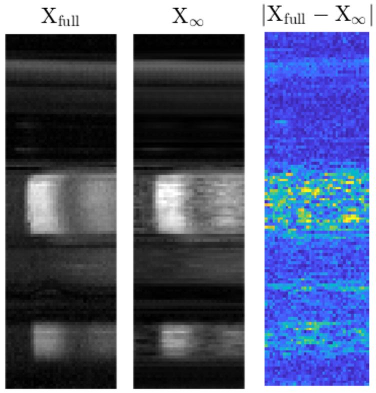
$\mathrm{X}_{\mathrm{full}}$
X_{∞}
$|\mathrm{X}_{\mathrm{full}} - \mathrm{X}_{\infty}|$

Fig. 1. The reconstructed images on a scale of [0,1] from the fully sampled and under-sampled Perfusion Dataset in the y-t domain, where the central slice for y is taken. The difference image is plotted on a scale of [0,0.2].

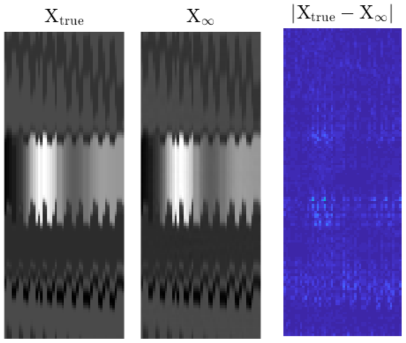
$\mathrm{X}_{\mathrm{true}}$
X_{∞}
$|\mathrm{X}_{\mathrm{true}} - \mathrm{X}_{\infty}|$

Fig. 2. The ground truth and the reconstructed image on a scale of [0,1] from the under-sampled PINCAT Dataset in the y-t domain, where the central slice for y is taken. The difference image is plotted on a scale of [0,0.2].

VII. Long-run Behaviors

Figures 3 and 4 show the results of running many more iterations of the algorithms investigated to illustrate the long-run behaviors of the methods, for the Cardiac Perfusion, Cardiac Cine and the PINCAT phantom datasets.

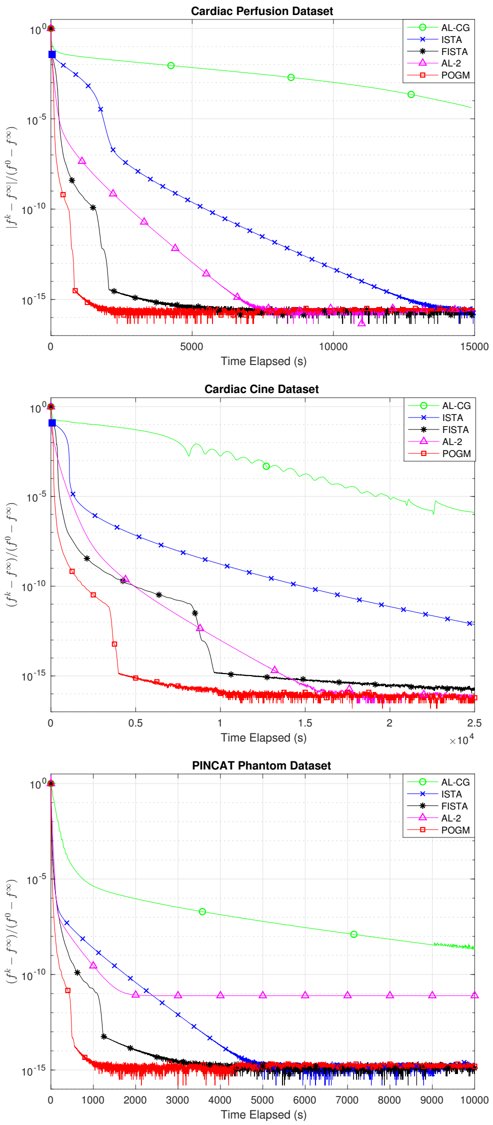

Fig. 3. Long-run convergence behaviors of the five algorithms for the three datasets, in terms of the cost function values. Every 500th iteration is marked by a dot, indicating their relative speeds.

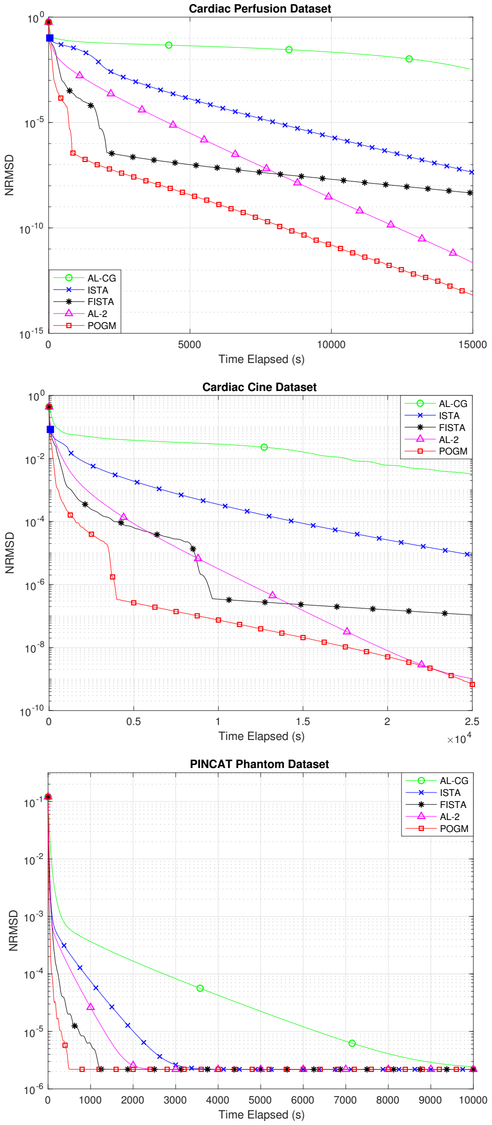

Fig. 4. Long-run convergence behaviors of the five algorithms for the three datasets, in terms of the NRMSD to the minimizer. Every 500th iteration is marked by a dot, indicating their relative speeds.

IX. Non-Cartesian Experiment

We performed an additional experiment on the Abdominal dynamic contrast-enhanced (DCE) MRI dataset examined in [9]. This non-Cartesian dataset uses a golden-angle radial sampling pattern, and corresponds to images of size 384×384, with 28 temporal frames and 12 receiver coils, having an acceleration factor of 12. As in the setup in the MATLAB code provided by [9], we used $\lambda_L = 0.025, \lambda_S =$ 2e-5, with adjusted scaling as described in Section V.B. Here, the step size for PGM depends on the maximum eigenvalue of $\mathbf{E}^*\mathbf{E}$, and we estimated it using power iteration. Due to the observed faster convergence of POGM than the AL methods, and because an additional variable splitting would needed for an AL approach for the non-Cartesian case, we focused on the proximal methods for this experiment. To obtain X_∞ and f_∞, we ran the POGM implementations for 1e5 seconds. Fig. 5 illustrates that POGM achieves the fastest convergence among the three methods. Fig. 6 shows reconstructed image results with the same run time cut-off of 153 seconds, with ISTA taking k = 9 iterations to reach the stopping criterion.

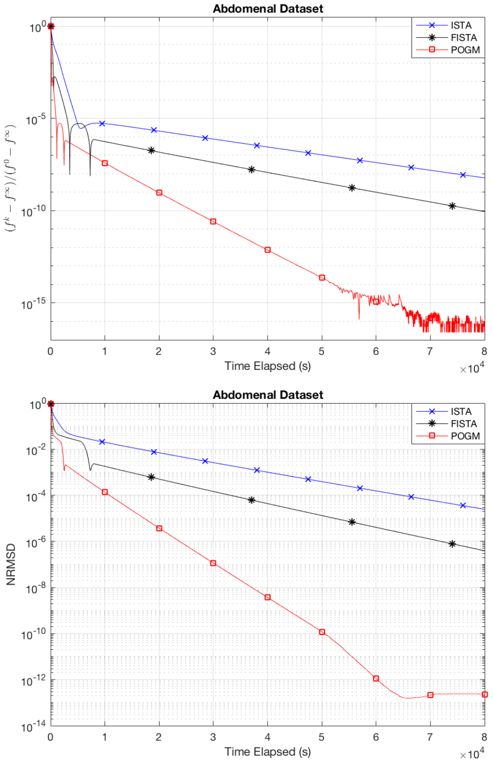

Fig. 5. Convergence of the three proximal algorithms for the Abdominal DCE phantom dataset, in terms of NRMSD and cost to the minimizer. Every 500th iteration of each algorithm is marked by a dot, indicating their relative speeds.

References

[1] C. Y. Lin and J. A. Fessler, "Efficient Dynamic Parallel MRI Reconstruction for the Low-Rank Plus Sparse Model," in *IEEE Trans. Comput. Imag.*, 2018, to appear.

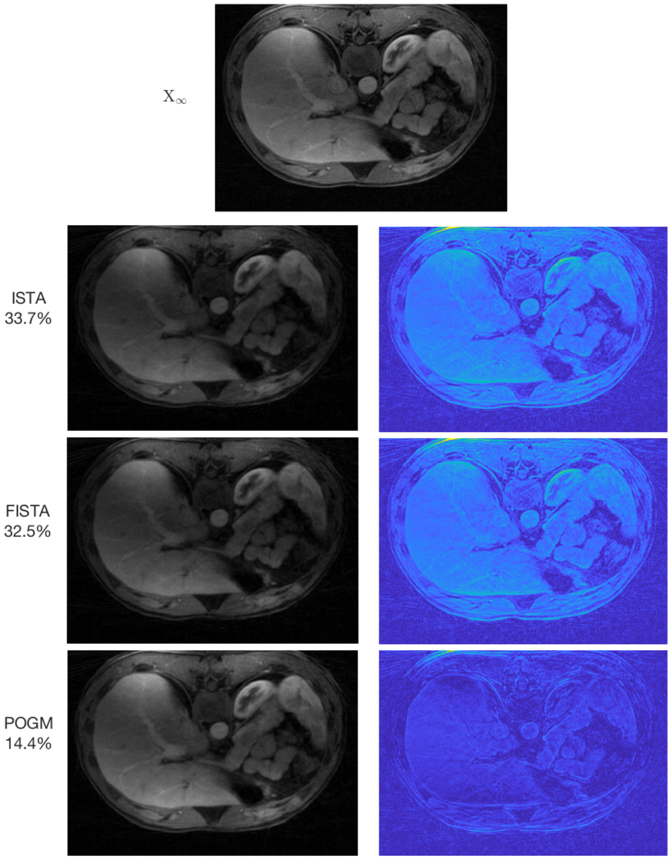

Fig. 6. First row: X_∞^{POGM}. Left column: reconstructed images on a scale of [0,5e-4], on one temporal frame for the Abdominal DCE Dataset, after the closest run time less than when ISTA reaches its stopping criterion (153 seconds). Right column: the residual images are plotted on a scale of [0,2e-4], with their corresponding NRMSD shown on the left of each row.