Published in final edited form as:
IEEE Trans Med Imaging. 2015 February ; 34(2): 578–588. doi:10.1109/TMI.2014.2363034.

Fast Parallel MR Image Reconstruction via B1-based, Adaptive Restart, Iterative Soft Thresholding Algorithms (BARISTA)

Matthew J. Muckley [Student Member, IEEE],
Department of Biomedical Engineering, University of Michigan, Ann Arbor, MI 48109 USA

Douglas C. Noll [Senior Member, IEEE], and
Department of Biomedical Engineering, University of Michigan, Ann Arbor, MI 48109 USA

Jeffrey A. Fessler [Fellow, IEEE]
Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA

Matthew J. Muckley: mmuckley@umich.edu; Douglas C. Noll: dnoll@umich.edu; Jeffrey A. Fessler: fessler@umich.edu

Abstract

Sparsity-promoting regularization is useful for combining compressed sensing assumptions with parallel MRI for reducing scan time while preserving image quality. Variable splitting algorithms are the current state-of-the-art algorithms for SENSE-type MR image reconstruction with sparsity-promoting regularization. These methods are very general and have been observed to work with almost any regularizer; however, the tuning of associated convergence parameters is a commonly-cited hindrance in their adoption. Conversely, majorize-minimize algorithms based on a single Lipschitz constant have been observed to be slow in shift-variant applications such as SENSE-type MR image reconstruction since the associated Lipschitz constants are loose bounds for the shift-variant behavior. This paper bridges the gap between the Lipschitz constant and the shift-variant aspects of SENSE-type MR imaging by introducing majorizing matrices in the range of the regularizer matrix. The proposed majorize-minimize methods (called BARISTA) converge faster than state-of-the-art variable splitting algorithms when combined with momentum acceleration and adaptive momentum restarting. Furthermore, the tuning parameters associated with the proposed methods are unitless convergence tolerances that are easier to choose than the constraint penalty parameters required by variable splitting algorithms.

Index Terms

MR Image Reconstruction; Compressed Sensing; FISTA; Majorize-Minimize; Parallel MRI

Correspondence to: Matthew J. Muckley, mmuckley@umich.edu.

I. Introduction

Magnetic resonance imaging (MRI) is an imaging modality where improving the resolution requires increased acquisition time. As a result, the cost of MRI also increases with higher resolution since the cost is directly proportional to the scan time. In addition to reducing the cost of high-resolution MRI, scanning time reductions can also help accommodate pediatric and elderly patients that have difficulty remaining motionless during long scans. Such scan time reductions are facilitated by combining undersampling and advanced signal processing methods to remove the associated aliasing artifacts. SENSitivity Encoding (SENSE) is an MRI technique that undoes aliasing effects caused by undersampling by exploiting variations in the sensitivity profiles (i.e., B1 maps) of multiple coils placed around the patient [1]. When the image can reasonably be assumed to be sparse in some transform domain, compressed sensing techniques can be applied to facilitate further accelerations [2].

Image estimation that leverages SENSE MRI and compressed sensing assumptions can be mathematically formulated as an ℓ_1-regularized optimization problem [3]. Since the ℓ_1 term is nondifferentiable, these problems are difficult to minimize using standard gradient-based methods. Some methods convert such problems into a different form where fast minimization techniques can be applied. One such class includes variable splitting algorithms, where one forms a constrained optimization problem and then proceeds within the augmented Lagrangian formalism to find the solution to the original ℓ_1-regularized problem [4]–[7]. A difficulty with applying these methods is the tuning of a constraint penalty parameter that heavily affects convergence speed. Sufficient conditions for optimally choosing these parameters are unknown, so current practice is to resort to heuristics for setting these parameters [7]. Since the optimal parameter can change from problem to problem (i.e., patient to patient), robust performance of these methods can be difficult to ascertain.

Alternatives to variable splitting methods are majorize-minimize methods such as fast iterative soft thresholding (FISTA) [8]. FISTA methods converge at a rate that depends on the *Lipschitz constant*, a constant that upper bounds the eigenvalues of the Hessian of the data fit term. This constant is on the order of the maximum of the sum of squared absolute values of the sensitivity coils. As such, the Lipschitz constant can be very loose for low signal regions that occur at the center of the object in SENSE MRI. As a result, majorize-minimize methods such as FISTA have performed poorly relative to their variable splitting counterparts in MRI applications [7].

We address the looseness of the Lipschitz bound by formulating tighter bounds that vary spatially based on the sensitivity coil profiles. The approach requires finding a diagonal majorizer in the range of the regularizing matrix. In this paper we show that for several regularizers of interest (including orthogonal wavelets, anisotropic total variation, and undeci-mated Haar wavelets), such diagonal upper bounds are simple to compute and give large accelerations relative to FISTA with the Lipschitz constant. When combined with adaptive momentum restarting [9], these methods outperform variable splitting methods in all of these cases. The proposed methods also use parameters in the form of convergence

tolerances, but in numerical experiments we found that once a reasonable choice was made for these parameters, further optimization was not necessary.

II. Problem Formulation and General Approach

From compressed sensing theory, one can recover a sparse signal by minimizing a convex cost function with ℓ_1 regularization [3]. Let C denote the number of sensitivity coils, D denote the number of data points, and N denote the number of pixels to be estimated. The ℓ_1-minimization procedure for parallel MR image reconstruction can be mathematically formulated as

$$\hat{\mathbf{x}}=\underset{\mathbf{x}\in\mathcal{M}}{\operatorname{argmin}}\{f(\mathbf{x})+\beta R(\mathbf{x})\},\quad f(\mathbf{x})=\tfrac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2,\ R(\mathbf{x})=\|\mathbf{R}\mathbf{x}\|_1,\ \mathbf{A}=\mathbf{F}\mathbf{S}, \tag{1}$$

where $\mathbf{F}\in\mathbb{C}^{D\times CN}$ is a block-diagonal matrix with each block having the same down-sampled DFT operator and $\mathbf{S}\in\mathbb{C}^{CN\times N}$ is a block-column matrix with diagonal blocks. We include a masking set, $\mathcal{M}$, that constrains elements outside the mask to be zero. We call $f(\mathbf{x})$ the data fit term and $R(\mathbf{x})$ the regularizer. Weighted quadratic data fit terms used for noise correlations between coils can be converted to this unweighted form [7]. The parameter, β, must be selected by the user to balance trade-offs between the data fit term and the regularizer (Monte Carlo techniques have been developed for estimating these parameters that perform well under mean-squared error metrics [10].) Defining $\mathbf{A}=\mathbf{FS}$, we note that $\mathbf{S}$ gives $\mathbf{A}$ a highly shift-variant nature, a property that we will consider in our algorithm design. $\mathbf{R}$ is a sparsifying transform. If $\mathbf{R}$ is left-invertible (i.e., $\mathbf{NR}=\mathbf{I}$ for some matrix, $\mathbf{N}$), we say that (1) is a *synthesis* reconstruction problem since we can define $\mathbf{u}=\mathbf{Rx}$ and rewrite (1) as an optimization problem over $\mathbf{u}$. We assume that $\mathbf{R}\in\mathbb{C}^{M\times N}$. If $\mathbf{R}$ is not left-invertible, then we call (1) an *analysis* reconstruction problem and assume that $\mathbf{R}\in\mathbb{R}^{M\times N}$. This restriction of $\mathbf{R}$ to be real-valued includes important classes of analysis regularizers such as total variation [2] and undecimated Haar wavelets [7]. Each of these regularization forms necessitates different algorithm considerations.

Although solving (1) allows one to obtain high-quality estimates of $\mathbf{x}$ with less data, (1) is typically difficult to minimize. Most methods instead minimize a different problem related to (1). The related problem should be easy to minimize relative to (1), but still offer information relevant to the solution of (1). Two procedures for defining and minimizing related problems are *majorize-minimize* procedures and *variable splitting* procedures. For completeness we note that “corner rounding” has also been proposed for dealing with the nondifferentiability of the ℓ_1 regularizer [2], but this has been found to yield algorithms slower than those of the variable splitting class [7]. Our method is of the majorize-minimize class, but is different from previous majorize-minimize methods in that it carefully considers any coupling of the structures of $\mathbf{A}$ and $\mathbf{R}$. We outline the general approach in the following section.

A. Separable Quadratic Surrogates

Majorize-minimize methods work by forming a surrogate cost function (i.e., a majorizer, $\varphi_k(\mathbf{x})$) and then minimizing the surrogate each iteration to find the minimizer of the original cost function. Any quadratic of the form $\frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2$ can be majorized with a *separable quadratic surrogate* (SQS), a procedure that we briefly review [11], [12]. If a surrogate, $\varphi_k(\mathbf{x})$, satisfies the following two conditions, then decreasing the surrogate will decrease the original cost function [13]:

$$f(\mathbf{x}^{(k)})=\phi_k(\mathbf{x}^{(k)}), \quad (2)$$

$$f(\mathbf{x}) \leq \phi_k(\mathbf{x}). \quad (3)$$

We allow the surrogate to be indexed by k since it can vary with iteration. We form such a surrogate for SENSE MRI by first rewriting $f(\mathbf{x})$ around a current estimate, $\mathbf{x}^{(k)}$, as

$$f(\mathbf{x})=f(\mathbf{x}^{(k)})+\Re\left\{\left(\mathbf{A}^H(\mathbf{A}\mathbf{x}^{(k)}-\mathbf{y})\right)^H(\mathbf{x}-\mathbf{x}^{(k)})\right\}+\frac{1}{2}(\mathbf{x}-\mathbf{x}^{(k)})^H\mathbf{A}^H\mathbf{A}(\mathbf{x}-\mathbf{x}^{(k)}), \quad (4)$$

where $\Re\{\cdot\}$ returns the real part of its argument and $\mathbf{A}^H$ is the Hermitian transpose of $\mathbf{A}$. If we have $\mathbf{A}^H\mathbf{A} \preceq \mathbf{D}_f \in \mathbb{R}^{N\times N}$ for some diagonal matrix, $\mathbf{D}_f$ (where $\mathbf{M} \succeq \mathbf{0}$ implies that $\mathbf{M}$ is positive semidefinite), we can write

$$\begin{aligned} f(\mathbf{x}) \leq \phi_k(\mathbf{x}) &= f(\mathbf{x}^{(k)})+\Re\left\{\left(\mathbf{A}^H(\mathbf{A}\mathbf{x}^{(k)}-\mathbf{y})\right)^H(\mathbf{x}-\mathbf{x}^{(k)})\right\}+\tfrac{1}{2}\|\mathbf{x}-\mathbf{x}^{(k)}\|_{\mathbf{D}_f}^2 \\ &= \psi(\mathbf{x},\mathbf{x}^{(k)})+\zeta, \\ \psi(\mathbf{x},\mathbf{x}^{(k)}) &:= \tfrac{1}{2}\left\|\mathbf{x}-(\mathbf{x}^{(k)}-\mathbf{D}_f^{-1}\mathbf{A}^H(\mathbf{A}\mathbf{x}^{(k)}-\mathbf{y}))\right\|_{\mathbf{D}_f}^2, \end{aligned} \quad (5)$$

where ζ is a constant that arises from completing the square and is independent of $\mathbf{x}$. Decreasing $\psi(\mathbf{x}, \mathbf{x}^{(k)})$ causes $\varphi_k(\mathbf{x})$ to decrease by the same amount. Standard majorize-minimize procedures use $\mathbf{D}_f = L\mathbf{I}$, where L is the maximum eigenvalue of $\mathbf{A}^H\mathbf{A}$. We instead use a more general $\mathbf{D}_f$ that is a tighter bound for $\mathbf{A}^H\mathbf{A}$. In the case where $\mathbf{A} = \mathbf{F}\mathbf{S}$, we have $\mathbf{A}^H\mathbf{A} = \mathbf{S}^H\mathbf{F}^H\mathbf{F}\mathbf{S}$. In general $\mathbf{F}^H\mathbf{F} \preceq F\mathbf{I}$, where F is the maximum eigenvalue of $\mathbf{F}^H\mathbf{F}$. In the case of Cartesian MRI with unitary DFT matrices, $F = 1$. One can estimate F offline in the non-Cartesian case via power iteration since it does not depend on the object. Noting this, we have

$$\mathbf{D}_f := F\mathbf{S}^H\mathbf{S} \succcurlyeq \mathbf{A}^H\mathbf{A}, \quad (6)$$

where $\mathbf{S}^H\mathbf{S}$ is a diagonal matrix with the sum of the squared absolute values of the sensitivity coils along its diagonal. We could use $\mathbf{D}_f$ to upper bound any SENSE-type quadratic data fit term with a separable quadratic surrogate. We will use this property in the following sections. Furthermore, $\mathbf{D}_f$ is easy to compute once one has determined the coil sensitivities,

and with the recent development of fast algorithms for SENSE map estimation it is quickly available in online settings [14].

B. Proposed Minimization Algorithm

We note through the majorization conditions that solving the following problem will decrease the cost function in (1):

$$\mathbf{x}^{(k+1)} = \underset{\mathbf{x}\in\mathscr{M}}{\operatorname{argmin}} \left\{ \eta(\mathbf{x}, \mathbf{x}^{(k)}) := \psi(\mathbf{x}, \mathbf{x}^{(k)}) + \beta R(\mathbf{x}) \right\}. \quad (7)$$

The minimization problem in (7) is where synthesis and analysis regularizers differ. In the synthesis case, $\eta(\mathbf{x}, \mathbf{x}^{(k)})$ is either fully separable or it can be converted into a fully separable form. When $\eta(\mathbf{x}, \mathbf{x}^{(k)})$ is fully separable, closed-form solutions exist via shrinkage functions. In the analysis case, closed form solutions do not exist and we have to run a few steps of an iterative algorithm to decrease $\eta(\mathbf{x}, \mathbf{x}^{(k)})$. We discuss the synthesis case in detail in Section III and the analysis case in Section IV.

Iteratively applying (7) qualifies as a majorize-minimize procedure, and as such it can be accelerated with momentum techniques [8]. Momentum accelerations can be enhanced with adaptive momentum restarting [9]. This gives a general algorithm, which we call B1-based, Adaptive Restart, Iterative Soft Thresholding Algorithm, or BARISTA, since it has step sizes that depend on the sensitivity or B1 maps. Fig. 1 shows the overall algorithm. Variants of this general form are shown in Fig. 2 for the synthesis case and Fig. 3 for the analysis case. The tracking of the momentum is provided by the $t^{(k)}$ parameter and an auxiliary variable, $\mathbf{z}^{(k)}$. If the algorithm takes a step in a certain direction, then $\mathbf{z}^{(k+1)}$ takes a larger step in the same direction where the size is determined by $t^{(k)}$. The restart is shown with the "if" statement at the end of an algorithm step. If the cosine of the angle between $\mathbf{x}^{(k+1)} - \mathbf{z}^{(k)}$ and $\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}$ is greater than a, then the momentum is wiped away. This helps prevent the generalized gradient and the momentum term from taking the algorithm in opposite directions. In our numerical experiments we found that good values for a are negative and lie near 0; we used $a = -\cos(4\pi/9)$. As stated previously, one challenge is in the minimization of $\eta(\mathbf{x}, \mathbf{z}^{(k)})$. Another associated challenge is the design of matrices similar to $\mathbf{D}_f$, but in the range of the regularizer for both synthesis and analysis regularization. The following sections discuss these topics.

III. Synthesis Regularization

A. Synthesis Algorithm Formulation

We use the term *synthesis regularization* when $\mathbf{R}$ is left-invertible, which allows rewriting the minimization problem in the basis of the regularizer. For notational simplicity in this section, we discuss $\mathbf{R}$ that forms a *tight frame*, i.e., $\mathbf{R}^H\mathbf{R} = \mathbf{I}$. Orthogonal wavelet transforms for SENSE MRI and DFT/DCT regularizers for dynamic MRI are examples of unitary matrices that might be used in synthesis problems in MRI. Defining $\mathbf{u} = \mathbf{R}\mathbf{x}$, we can rewrite (1) as

$$\hat{\mathbf{u}} = \underset{\mathbf{u}\in\mathcal{M}_{\text{synth}}}{\operatorname{argmin}} \left\{ \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{R}^H\mathbf{u}\|_2^2 + \beta\|\mathbf{u}\|_1 \right\}, \quad \hat{\mathbf{x}} = \mathbf{R}^H\hat{\mathbf{u}} \tag{8}$$

where $\mathcal{M}_{\text{synth}}$ is a synthesis mask that restricts a subset of the synthesis coefficients to be zero. It is less natural to use a mask for the synthesis approach than for analysis, so if masking is desired we recommend using the algorithm outlined in Section IV. Now, if we find a diagonal matrix, $\mathbf{D}_R$, such that

$$\mathbf{D}_R \succcurlyeq \mathbf{R}\mathbf{D}_f\mathbf{R}^H \tag{9}$$

and defining $\mathbf{B} = \mathbf{A}\mathbf{R}^H$, the surrogate in (7) is

$$\eta(\mathbf{u}, \mathbf{u}^{(k)}) = \frac{1}{2}\|\mathbf{u}-(\mathbf{u}^{(k)}-\mathbf{D}_R^{-1}\mathbf{B}^H(\mathbf{B}\mathbf{u}^{(k)}-\mathbf{y}))\|_{\mathbf{D}_R}^2 + \beta\|\mathbf{u}\|_1, \tag{10}$$

the constrained minimum of which is a shrinkage solution:

$$\begin{aligned}\mathbf{u}^{(k+1)} &= \underset{\mathbf{u}\in\mathcal{M}_{\text{synth}}}{\operatorname{argmin}}\ \eta(\mathbf{u}, \mathbf{u}^{(k)}) \\ &= P_{\mathcal{M}_{\text{synth}}}(\text{shrink}(\mathbf{b}^{(k)}, \beta\mathbf{d}_R^{-1})),\end{aligned} \tag{11}$$

where $\mathbf{b}^{(k)} = \mathbf{u}^{(k)} - \mathbf{D}_R^{-1}\mathbf{B}^H(\mathbf{B}\mathbf{u}^{(k)}-\mathbf{y})$. $\mathbf{d}_R^{-1}$ is a vector composed of the diagonal elements of $\mathbf{D}_R^{-1}$, and the shrinkage function is defined as shrink($\mathbf{b}$, β) = diag{sign(b_i)}($|\mathbf{b}| - \beta\mathbf{1}$)$_+$. In this case, $|\cdot|$ denotes the absolute value function, $(\cdot)_+$ sets values less than zero to zero, diag{$\cdot$} takes the input elements and arranges them as a diagonal matrix, and sign($\cdot$) returns the complex sign of its argument. The $P_{\mathcal{M}_{\text{synth}}}$ operator projects its argument on to the set, $\mathcal{M}_{\text{synth}}$, which in this case corresponds to setting all elements outside the mask to zero. Fig. 2 shows the synthesis version of BARISTA. Our goal now is to design $\mathbf{D}_R$, which will allow us to take larger step sizes in step 5 of Fig. 2 and apply more aggressive shrinkage in step 6.

B. Diagonal Upper Bounds in Unitary Bases

The challenge in designing $\mathbf{D}_R$ is that it must be constructed in the basis of the regularizer, while $\mathbf{D}_f$ is in the basis of the image. For this purpose we will use Theorem 1, which gives a means of constructing $\mathbf{D}_R$. Theorem 1 can be applied for any $\mathbf{R}$, although it is most useful for unitary regularizing matrices that have rows with compact support. We will use it here since orthogonal wavelets fall into this class.

Theorem 1—Let $\mathbf{R} \in \mathbb{C}^{M\times N}$ be any matrix and let $\mathbf{D}_f \in \mathbb{R}^{N\times N}$ be diagonal with diagonal elements $d_{n,f}$. Let $\mathbf{r}_m$ be the mth row of $\mathbf{R}$ and let $\mathcal{S}_m \subset \{1, \ldots, N\}$ be the support set for $\mathbf{r}_m$. Define $\tau_m = \max_{n\in\mathcal{S}_m}(d_{n,f})$ and $d_{m,R} = \sum_{l=1}^{M} \min(t_m, t_l)|\langle \mathbf{r}_m, \mathbf{r}_l\rangle|$. Let $\mathbf{D}_R$ be a diagonal matrix with diagonal elements $d_{m,R}$, then $\mathbf{D}_R \succeq \mathbf{R}\mathbf{D}_f\mathbf{R}^H$.

The Appendix shows a proof of Theorem 1. Theorem 1 states that any matrix of the form $\mathbf{R}\mathbf{D}_f\mathbf{R}^H$ can be upper bounded with a diagonal matrix by taking maximums over patches of $\mathbf{D}_f$ and scaling those maximums by sums of inner products. These inner product sums increase as $\mathbf{R}$ becomes less unitary, but in our synthesis case we assume unitary $\mathbf{R}$ so $d_{m,R} = t_m$. We have found that this is an effective majorizing matrix for unitary regularizing matrices, and we used Theorem 1 to design $\mathbf{D}_R$ for orthogonal Haar and Daubechies D4 wavelets in our numerical experiments where we ran the algorithm in Fig. 2.

IV. Analysis Regularization

A. Analysis Algorithm Formulation

In the analysis setting $\mathbf{R}$ is not left-invertible and we can no longer define $\mathbf{u} = \mathbf{R}\mathbf{x}$ and rewrite (1) as an optimization problem over $\mathbf{u}$. As such, we must leave (1) in its original form. The forms of $\mathbf{R}$ of this type that are of interest for SENSE MRI include anisotropic total variation regularizers [2] and undecimated wavelets where the approximation coefficients are unregularized [7], [15]. However, we can still form a quadratic surrogate for the data fit term. This gives the *analysis denoising problem*. Since we do not have a closed-form solution of this problem, we run a few iterations of a denoising procedure. Fig. 3 shows the overall analysis algorithm, while the denoising procedure is shown in Fig. 4.

We must decide on a stopping criterion for the iterative algorithm used for the denoising step. Previous methods have used a fixed iteration count for this step [7], [16], but we instead use an $\varepsilon^{(k)}$ criterion. When large steps are being taken in the outer iterations, the denoising step only needs to provide an approximate solution to progress the algorithm, whereas very accurate solutions are beneficial for later iterations where the outer steps are small.

Fig. 3 shows a strategy for choosing $\varepsilon^{(k)}$ that was effective in our numerical experiments. $\varepsilon^{(k)}$ is chosen to be $\varepsilon_{\text{diff}}$ times the norm-difference of the previous iteration, restricted between the upper and lower bounds of $\varepsilon^{(k-1)}$ and ε_{min}. We choose $\varepsilon_{\text{diff}}$ to balance the cost of solving the denoising problem and progressing the outer iterations. In all experiments we used $\varepsilon_{\text{diff}} = 10^{-1}$. We choose ε_{min} based on the precision level of the machine that runs the algorithm; its primary purpose is to prevent the algorithm from stalling as a result of numerical precision. For double precision, we observed that $\varepsilon_{\text{min}} = 10^{-12}$ gave agreeable convergence in later iterations. We set $\varepsilon^{(0)} = 10^{-1}$. We decrease $\varepsilon^{(k)}$ monotonically so that the denoising subproblem is solved more accurately as the algorithm progresses toward the solution.

B. Analysis Denoising Subroutine

We now discuss the so-called *analysis denoising problem* that needs to be solved in step 6 of the algorithm in Fig. 3, which is formulated as follows:

$$\begin{aligned}\mathbf{x}^{(k+1)} &= \underset{\mathbf{x}\in\mathscr{M}}{\operatorname{argmin}}\, \eta(\mathbf{x}, \mathbf{x}^{(k)}),\\ \eta(\mathbf{x}, \mathbf{x}^{(k)}) &= \tfrac{1}{2}\|\mathbf{b}^{(k)} - \mathbf{x}\|_{\mathbf{D}_f}^2 + \|\mathbf{R}\mathbf{x}\|_1, \quad (12)\\ \mathbf{b}^{(k)} &= \mathbf{x}^{(k)} - \mathbf{D}_f^{-1}\mathbf{A}^H(\mathbf{A}\mathbf{x}^{(k)} - \mathbf{y}).\end{aligned}$$

This is equivalent to solving step 4 of the algorithm in Fig. 1. There are many potential approaches to solving this step, including nonlinear CG [2] and split Bregman schemes [4]. As mentioned in the previous section, our goal is to minimize the cost function in (12) to some pre-specified numerical precision. As a result, whatever procedure is chosen should perform well under all numerical precision environments, a property not satisfied by nonlinear CG due to its corner-rounding parameter or split Bregman due to its constraint penalty parameter. Instead, we choose to extend the results in [16] to general ℓ_1 regularizers, adapt it to complex numbers, and reintroduce our diagonal majorizing matrix in the range of the regularizer. This approach meets the numerical precision requirements and gave agreeable convergence speed in numerical experiments. Our derivation requires real-valued $\mathbf{R}$, which includes interesting classes of anisotropic total variation and undecimated Haar wavelet regularizers. Fig. 4 shows the algorithm that arises from extending the results in [16]. The Appendix derives this algorithm.

The $P_{\mathcal{P}^M}(\cdot)$ operator in Fig. 4 projects its argument on to the set, $\mathcal{P}^M$, the ℓ_∞-unit ball. This set arises from our dual formulation discussed in the Appendix. In this case this means that $P_{\mathcal{P}^M}(\cdot)$ examines each element in its input vector and normalizes any elements with an absolute value greater than 1 to an absolute value of 1, preserving the complex sign. For this inner denoising step we include an ε stopping criterion, the choice of which as discussed in Section IV depends on the step sizes of the outer iterations of the algorithm in Fig. 3. Although not noted in Fig. 4, we also included a maximum iteration number to prevent the algorithm from stalling. We did not observe that this was necessary in our numerical experiments, but we wanted to ensure stable convergence in a variety of circumstances. We measure the convergence based on $\mathbf{x}^{(k,j+1)}$, which is calculated from the momentum variable $\mathbf{v}^{(j)}$, although the actual convergence would depend on the dual variable, $\mathbf{q}^{(j)}$. This simplification avoids making extra computations each iteration that would be required to estimate $\mathbf{x}$ from $\mathbf{q}^{(j)}$, and with the adaptive restart we expect $\mathbf{v}^{(j)}$ to be a good approximation for $\mathbf{q}^{(j)}$ near the solution. Lastly, we note that we initialized the analysis denoising algorithm with the last value for $\mathbf{q}$ from the previous run of the algorithm. This warm start greatly helps the convergence speed of the analysis denoising subroutine.

The algorithm in Fig. 4 requires computing a $\mathbf{D}_R$ that satisfies the analysis majorizer condition:

$$\mathbf{D}_R \succcurlyeq \mathbf{R}\mathbf{D}_f^{-1}\mathbf{R}^T, \quad (13)$$

where $\mathbf{R}^T$ is the transpose of $\mathbf{R}$. We discuss $\mathbf{D}_R$ for the cases of anisotropic total variation and undecimated Haar wavelets.

C. Diagonal Majorizers for Analysis Regularizers

One could use Theorem 1 to upper bound any matrix, including $\mathbf{R}\mathbf{D}_f^{-1}\mathbf{R}^T$. However, in practice we have found that bound somewhat loose for the analysis regularizing matrices of anisotropic total variation and undecimated Haar wavelets. We discuss tighter bounds for these two cases. For the case of anisotropic total variation, we choose

$$\mathbf{D}_R = \mathrm{diag}\{\mathrm{abs}(\mathbf{R})\mathbf{D}_f^{-1}\mathrm{abs}(\mathbf{R}^T)\mathbf{1}\}, \quad (14)$$

where abs(·) returns a matrix that has entries that are the absolute value of the input matrix. This is guaranteed to be a majorizer as it is a modification of the techniques of De Pierro [17], and we have found it to be very tight for anisotropic total variation. Its calculation is also simple.

For the case of undecimated Haar wavelets, we present a different approach that builds on Theorem 1 via the polarization identity. The idea is to split up a non-orthogonal **R** into orthogonal pieces for which Theorem 1 will provide tight diagonal majorizers. Consider **R** of the form,

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 \\ \vdots \\ \mathbf{R}_Q \end{bmatrix}, \quad (15)$$

where Q is the number of submatrices of **R**. Defining $\mathbf{c}_i = \mathbf{D}_f^{-1/2}\mathbf{R}_i^T\mathbf{w}_i$ for any arbitrary vector $\mathbf{w}^H = [\mathbf{w}_1^H, \ldots, \mathbf{w}_Q^H]$ (possibly different sized $\mathbf{w}_i$), we then have

$$\begin{aligned} \mathbf{w}^H\mathbf{R}\mathbf{D}_f^{-1}\mathbf{R}^T\mathbf{w} &= \sum_{i=1}^{Q}\|\mathbf{c}_i\|_2^2 + \sum_{i=1}^{Q}\sum_{j=i+1}^{Q} 2\Re\{\langle \mathbf{c}_i, \mathbf{c}_j\rangle\} \\ &\leq \sum_{i=1}^{Q}\|\mathbf{c}_i\|_2^2 + (Q-1)\sum_{i=1}^{Q}\|\mathbf{c}_i\|_2^2 \\ &= \sum_{i=1}^{Q} Q\mathbf{w}_i\mathbf{R}_i\mathbf{D}_f^{-1}\mathbf{R}_i^T\mathbf{w}_i, \end{aligned} \quad (16)$$

where one proceeds from the first to the second step by applying the polarization identity,

$$2|\Re\{\langle \mathbf{c}_i, \mathbf{c}_j\rangle\}| \leq \|\mathbf{c}_i\|_2^2 + \|\mathbf{c}_j\|_2^2, \quad (17)$$

and collecting all inner product pairs. Thus, finding a majorizer for each $\mathbf{R}_i\mathbf{D}_f^{-1}\mathbf{R}_i^T$ submatrix will provide a majorizer for $\mathbf{R}\mathbf{D}_f^{-1}\mathbf{R}^T$. Such a structure applies to the 2-level undecimated Haar wavelet case since a 2-level undecimated Haar wavelet can be written as a cascade:

$$\mathbf{R} = \mathbf{R}_B\mathbf{R}_A. \quad (18)$$

In this case the first step of the cascade, $\mathbf{R}_A$, can be broken up into pieces:

$$\mathbf{R}_A = \begin{bmatrix} \mathbf{R}_{A,1} \\ \mathbf{R}_{A,2} \\ \mathbf{R}_{A,3} \\ \mathbf{R}_{A,4} \end{bmatrix}, \quad (19)$$

where each of the $\mathbf{R}_{A,i}$ is an orthogonal 1-level Haar wavelet transform. We apply the inequality in (16) to majorize $\mathbf{R}_A \mathbf{D}_f^{-1} \mathbf{R}_A^T$ while using Theorem 1 to majorize each $\mathbf{R}_{A,i} \mathbf{D}_f^{-1} \mathbf{R}_{A,i}^T$ term. We applied the procedure recursively to $\mathbf{R}_B$ since it has a similar structure. In the undecimated Haar wavelet case each of the $\mathbf{R}_{A,1}$, …$\mathbf{R}_{A,4}$ is a similar operation, so we expect that each of the $\mathbf{c}_i$ will be approximately linearly dependent and this inequality approach will be fairly tight.

V. Experiments

A. Experimental Setup

In the interest of reproducible research, MATLAB code for implementing these methods will be uploaded to the Image Reconstruction Toolbox at web.eecs.umich.edu/~fessler/. All experiments were run on a machine with an Intel Xeon E31230 Processor that had four cores with each core running at 3.2 GHz. The machine had 16 GB of memory. All experiments used $\alpha = -\cos(4\pi/9)$.

We compared the convergence speed of BARISTA to state-of-the-art variable splitting methods in several experiments on four data sets. We present *in vivo* brain results in the main paper body and include results for a numerical brain phantom, a breast phantom, and an American College of Radiology phantom in the supplementary material. The variable splitting methods were each of the AL-P1 or split Bregman type [7]. The AL-P2 method in [7] uses condition number heuristics to tune AL penalty parameters, but we found that these condition number heuristics could change between different regularizers. Tuning AL-P2 for each regularizer would have required setting multiple condition number parameters. AL-P1 has only one constraint penalty parameter, μ, and it had comparable speed to AL-P2, so we used AL-P1 with careful manual tuning of μ as a representative of AL-based methods. We also investigated dynamically updating the μ parameter using update rules and parameters proposed by Boyd (Section 3.4.1 of [18]), which helps mitigate tuning difficulties. We initialized such AL methods with dynamic μ updates with one of the manually-tuned μ values. In the plots this method with dynamic μ updates is denoted as "AL, dynamic μ." We also introduced a diagonal preconditioner for the conjugate gradient (CG) subroutine in step 4 of AL-P1. We used $\mathbf{P} = (\mathbf{S}^H\mathbf{S} + \mu\mathbf{I})^{-1}$ for all wavelet regularizers and $\mathbf{P} = (\mathbf{S}^H\mathbf{S})^{-1}$ for the total variation regularizer. These preconditioners were not mentioned in [7], but we observed that they accelerated AL-P1 on the order of 50% time to reach the same point of convergence. The AL-P1 methods all used 5 preconditioned CG (PCG) iterations for step 4 of AL-P1.

To track convergence, we computed the following normalized residual as a function of iteration:

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

where $\mathbf{x}^{(\infty)}$ is a "converged" solution obtained by running many thousands of iterations of AL-P1. Note that even though $\mathbf{R}$ is not full column rank in the total variation case, the AL-P1 method is still convergent [19]. In our convergence plot comparisons to AL-based methods we set the lower bound for $\xi(k)$ at −140 dB. We chose to do this for two reasons: 1) our raw MRI data were less precise than single precision and 2) BARISTA vastly outperformed all other methods in reaching double precision, so these parts of the plots were less interesting. We also stored the time at which the kth estimate was computed and in our figures and we plot $\xi(\cdot)$ as a function of elapsed CPU time instead of iteration. We choose to do this since iterations of the proposed majorize-minimize methods and the variable splitting methods have drastically different compute times due to the PCG subroutine and the analysis denoising step in the proposed methods. All methods used identical subroutines for matrix multiplications.

We selected regularization parameter, β, to give visually appealing solutions for each regularizer. In practice the regularization parameter could be estimated via Monte Carlo SURE methods [10].

In our plots we only show BARISTA from the classes of majorize-minimize algorithms as opposed to other methods such as FISTA with $\mathbf{D}_f = L\mathbf{I}$ since BARISTA was always the fastest majorize-minimize method. Fig. 5 shows an example of the relative convergence speed of majorize-minimize methods in the case of orthogonal Haar wavelet regularization. In this case, BARISTA was twice as fast as RFISTA (restart FISTA), three times as fast as NRBARISTA, (non-restart BARISTA) and over five times as fast as FISTA in reaching −120 dB. Although RFISTA converges rapidly to double precision, in early iterations it is not competitive with BARISTA or variable splitting methods. In a practical setting, the algorithms may not even be run to convergence, so early-iteration convergence speed is critical for general adoption of the proposed methods. Furthermore, negligible time is required to use the majorizing matrices discussed in this paper, so the factor of two speed-up over RFISTA more or less comes for free. We also observed the factor of two speed-up or greater with the orthogonal Daubechies D4 regularizer and the undecimated Haar wavelet regularizer. Speed-up of BARISTA vs. RFISTA in the anisotropic total variation case was negligible. We are unsure why this occurred, but it may be that a shift-variant majorizer makes the analysis denoising problem more difficult to solve in the total variation case.

For the *in vivo* experiment, a 3D data set was acquired on a GE 3T scanner with an 8-channel head coil with acquisition parameters T_R = 25 ms, T_E = 5.172 ms, and voxel size 1 mm × 1.35 mm × 1 mm. The data matrix size was 256 × 144 × 128 uniformly spaced samples. Sensitivity maps were estimated using a quadratic regularized least squares routine [14]. The data were retrospectively undersampled in the Fourier domain using a Poisson disk sampling scheme [20] with a fully sampled center (32-by-32 block), which has been demonstrated to be useful in compressed sensing MRI applications [21]. This sampling pattern simulates one slice of a 3D MRI experiment where the sampling pattern in Fig. 6b is

in the phase encode plane [7], [21] (this sampling pattern would be impractical for 2D MRI). Only 20% of the full DFT sampling was used for reconstruction. Fig. 6a shows **x** estimated from fully sampled data, while Fig. 6b shows the Poisson-disc sampling pattern with a densely-sampled center used in all the *in vivo* experiments.

B. Synthesis Regularizer Results

As stated earlier, we performed numerical experiments with orthogonal Haar and Daubechies D4 wavelet regularizers to examine the convergence speed of the proposed method in the synthesis setting. We set the regularization parameter to zero for the approximation coefficients since a sparse model does not fit these coefficients as well as the detail coefficients [15]. Fig. 7a shows an example of the diagonal majorizing elements in the Haar wavelet basis. Fig. 7b shows the majorizer for the Daubechies D4 wavelets. The majorizer for the Daubechies D4 wavelet case is smoother than the Haar case since it requires taking maximums over larger patches.

Fig. 8a and Fig. 8b show the convergence results for Haar and Daubechies D4 wavelets, respectively. BARISTA converges faster than the other methods. The time positions when BARISTA undergoes restart are visible in the stair step pattern in the convergence plots. Several AL parameters are shown to demonstrate the range of speeds of AL-based methods, although we can make no theoretical guarantees on the optimal speed of AL-based methods since we do not know any theoretically optimal way to tune the penalty parameter.

C. Analysis Regularizer Results

We performed numerical experiments with total variation and 2-level undecimated Haar wavelet regularization to examine the convergence speed of the proposed methods in the analysis setting. Our anisotropic total variation implementation took differences in vertical, horizontal, and diagonal directions. We did not regularize the approximation coefficients of the 2-level undecimated Haar wavelet transform [15]. Fig. 9 shows examples of elements from $\mathbf{D}_R$ for the analysis cases. Since for the analysis case we design $\mathbf{D}_R \succcurlyeq \mathbf{R}\mathbf{D}_f^{-1}\mathbf{R}^T$, the sensitivity elements are now inverted relative to the synthesis case. Our analysis algorithm formulation required setting the $\varepsilon_{\text{diff}}$, $\varepsilon_{\min}$, and $\varepsilon^{(0)}$ parameters. We chose $\varepsilon_{\min} = 10^{-12}$, $\varepsilon_{\text{diff}} = 10^{-1}$, and $\varepsilon^{(0)} = 10^{-1}$. We note that although these convergence criteria parameters require some tuning, we were able to use the same convergence criteria for all regularizers in all experiments. Conversely, we had to tune the constraint penalty parameters for the AL-P1 method each time when changing regularizers or data sets.

Fig. 10 shows results for the analysis regularizers. BARISTA matches the other methods in early iterations and outperforms all other methods in later iterations. As previously, the time steps at which the algorithm restarts are shown in the stair step pattern in the convergence plots. We also observed that all algorithms converged slower with the total variation regularizer than the other regularizers. Results with analysis regularizers with an image domain mask were similar and are shown in the supplementary material. Notably, BARISTA converged about twice as fast when using a mask than without a mask.

VI. Discussion

A. Convergence Speed of BARISTA vs. AL Methods

BARISTA was observed to converge faster than the AL-based methods in both early and late iterations. The early iteration speed of BARISTA is due to its tight approximation of the Hessian of the cost function via the diagonal majorizers developed in this paper and the use of Nesterov momentum acceleration. Nesterov momentum has been added to AL algorithms in some cases [22], although those algorithms require an estimate of the Lipschitz constant, so the diagonal majorizers presented here may be useful for those methods.

The late-iteration speed of BARISTA is due to the use of adaptive restart. We are unaware of a means to apply adaptive restart to AL-based methods. We attempted to recover some of the benefits of adaptive restart through the use of dynamic AL parameter updates, but this did not give the same large convergence speed boost as adaptive restart.

B. Selection of AL Penalty Parameters

We manually optimized the AL penalty parameters for speed. For some cases, such as total variation and 2-level undecimated Haar wavelet regularizers, we observed a tradeoff between early and late iteration convergence speed, with smaller parameters favoring early iteration speed and larger parameters favoring late iteration speed. In our tests we chose the small parameters that gave reasonable convergence to −120 dB; however, this behavior suggests that changing the penalty parameter in a dynamic fashion may improve the convergence speed of AL-based methods. The dynamic tuning method from [18] helped in some cases, but not consistently, and we still observed faster convergence with BARISTA in both early and late iterations.

From a theoretical point of view, analysis of AL-based methods considers static penalty parameters [23]. The fact that AL theory considers static penalty parameters is considered one of the primary motivations for using AL methods instead of penalty methods in the first place [7], [23]; adaptively changing the parameter removes this advantage. Conversely, our use of the $\varepsilon^{(k)}$ parameter falls within the MFISTA theory provided monotonicity checks are used, and although [16] does not cover the case of adaptive restart we observed stable convergence of BARISTA in all numerical experiments.

C. Surrogate Tightness and Sensitivity Coil Smoothness

In our data set, the sum of squares of absolute values of the sensitivity coils exhibited high variability across the object. As the sensitivity coils become more uniform, our proposed $\mathbf{D}_f$ and $\mathbf{D}_R$ matrices will more closely approximate their Lipschitz counterparts (i.e., $\mathbf{D}_f = L\mathbf{I}$). In these cases, the advantages of BARISTA will diminish relative to that of RFISTA. However, we typically expect RFISTA to be a lower bound for the speed of BARISTA. Furthermore, as the sensitivity coils become smoother, the proposed surrogate functions actually become better approximations to the original cost function in (1), so we expect the speed of the proposed methods to be superior with smooth sensitivity coil profiles than our case with large sensitivity coil variability.

D. Tuning the Restart Criterion

The restart criterion in [9] used $\alpha = 0$ in all of their experiments. We found that this choice led to too infrequent restarts because using $\alpha = 0$ allows the momentum and generalized gradient to begin to point in different directions before restarting. Instead, in our experiments we used $\alpha = -\cos(4\pi/9)$. We found that this choice gave very good early-iteration convergence in 24 numerical experiments with four different regularizers (see supplementary material).

E. Near-monotonicity of BARISTA

In [16] it is stated that when an iterative procedure is applied to minimize the surrogate cost function, one should apply a monotonicity check to ensure stable convergence of the algorithm in the analysis total variation setting. The primary cause of non-monotonicity with FISTA algorithms is when the momentum takes the algorithm in a bad direction near the solution [9]. In our numerical experiments we observed that the combination of the $\varepsilon^{(k)}$ parameter and adaptive restart made the monotonicity checks in [16] unnecessary and the proposed method performed as a monotone algorithm. Nonetheless, the monotonicity checks of [16] could be included in a practical setting if monotonicity is still deemed to be an issue.

F. Relations to Proximal Newton Methods

The methods outlined in this paper have some relations to proximal Newton methods (e.g., [24]), which use alternative methods to approximate the Hessian. One issue with such methods is that the memory storage requirements can be undesirably large for medical imaging problems. Low-memory versions of these methods also exist (e.g., L-BFGS, see [24], [25]). BARISTA can also be thought of as having a low-memory approximation to the Hessian due to its diagonal structure, which may be more accurate if the SENSE maps dominate the behavior of the Hessian. Comparisons between our proposed method and these more general proximal Newton methods are an avenue for future investigation. One could even potentially modify BARISTA to use an L-BFGS Hessian approximation update, although proximal Newton methods are often developed for real numbers and may require adaptations for the complex numbers in MRI reconstruction.

VII. Conclusion

We have introduced generalizations of the FISTA algorithm, which we call BARISTA, for SENSE-type MR imaging with compressed sensing regularizers that compensate for the shift-variant aspects of the sensitivity coils. The methods gave superior convergence speed relative to state-of-the-art variable splitting methods in numerical experiments. Furthermore, the proposed methods avoid the penalty parameter tuning associated with variable splitting methods, instead relying on unitless convergence tolerance parameters. We have provided heuristics for selecting these parameters and found that the same values worked well across 24 numerical experiments conducted with four different regularizers on four different data sets. We expect that the proposed methods will give fast, high-quality reconstructions across a wide variety of data sets and will aid in the adoption of compressed sensing methods in a clinical setting.

Acknowledgments

This work was supported by the University of Michigan MCubed program and NIH grants 1P01 CA87634 and R01 NS 058576.

The authors thank Hung Nien for the suggestion to incorporate adaptive restart. We would also like to thank Daniel S. Weller for a useful discussion that led to step 2 of (16). Lastly, we would like to thank Gopal Nataraj, Hao Sun, and Paige Castle for their careful proofreading of the manuscript.

References

1. Pruessmann KP, Weiger M, Scheidegger MB, Boesiger P. SENSE: sensitivity encoding for fast MRI. Mag Res Med. Nov; 1999 42(5):952–62.
2. Lustig M, Donoho D, Pauly JM. Sparse MRI: The application of compressed sensing for rapid MR imaging. Mag Res Med. Dec; 2007 58(6):1182–95.
3. Candès EJ, Romberg J, Tao T. Stable signal recovery from incomplete and inaccurate measurements. Comm Pure Appl Math. 2006; 59(8):1207–23.
4. Goldstein T, Osher S. The split Bregman method for L1-regularized problems. SIAM J Imaging Sci. 2009; 2(2):323–43.
5. Yang J, Zhang Y, Yin W. A fast alternating direction method for TVL1-L2 signal reconstruction from partial Fourier data. IEEE J Sel Top Sig Proc. Apr; 2010 4(2):288–97.
6. Afonso José MV, Bioucas-Dias M, Figueiredo Mário AT. Fast image recovery using variable splitting and constrained optimization. IEEE Trans Im Proc. Sep; 2010 19(9):2345–56.
7. Ramani S, Fessler JA. Parallel MR image reconstruction using augmented Lagrangian methods. IEEE Trans Med Imag. Mar; 2011 30(3):694–706.
8. Beck A, Teboulle M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J Imaging Sci. 2009; 2(1):183–202.
9. O'Donoghue B, Candès E. Adaptive restart for accelerated gradient schemes. Found Computational Math. 2014
10. Ramani S, Liu Z, Rosen J, Nielsen JF, Fessler JA. Regularization parameter selection for nonlinear iterative image restoration and MRI reconstruction using GCV and SURE-based methods. IEEE Trans Im Proc. Aug; 2012 21(8):3659–72.
11. Erdoğan H, Fessler JA. Ordered subsets algorithms for transmission tomography. Phys Med Biol. Nov; 1999 44(11):2835–51. [PubMed: 10588288]
12. Hunter DR, Lange K. A tutorial on MM algorithms. American Statistician. Feb; 2004 58(1):30–7.
13. Jacobson MW, Fessler JA. An expanded theoretical treatment of iteration-dependent majorize-minimize algorithms. IEEE Trans Im Proc. Oct; 2007 16(10):2411–22.
14. Allison MJ, Ramani S, Fessler JA. Accelerated regularized estimation of MR coil sensitivities using augmented Lagrangian methods. IEEE Trans Med Imag. Mar; 2013 32(3):556–64.
15. Selesnick, IW.; Figueiredo, MAT. Signal restoration with overcomplete wavelet transforms: comparison of analysis and synthesis priors. Proc. SPIE 7446 Wavelets XIII; 2009. p. 74460Dwavelets XIII
16. Beck A, Teboulle M. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. IEEE Trans Im Proc. Nov; 2009 18(11):2419–34.
17. De Pierro AR. A modified expectation maximization algorithm for penalized likelihood estimation in emission tomography. IEEE Trans Med Imag. Mar; 1995 14(1):132–7.
18. Boyd S, Parikh N, Chu E, Peleato B, Eckstein J. Distributed optimization and statistical learning via the alternating direction method of multipliers. Found & Trends in Machine Learning. 2010; 3(1):1–122.
19. Nien, H.; Fessler, JA. A convergence proof of the split Bregman method for regularized least-squares problems. 2014. arxiv 1402.4371. [Online]. Available: http://arxiv.org/abs/1402.4371
20. Dunbar D, Humphreys G. A spatial data structure for fast Poisson-disk sample generation. ACM Trans on Graphics. Jul; 2006 25(3):503–8. sIGGRAPH.

21. Lustig M, Alley M, Vasanawala S, Donoho DL, Pauly JM. L1 SPIR-iT: Autocalibrating parallel imaging compressed sensing. Proc Intl Soc Mag Res Med. 2009:334.
22. Ouyang, Y.; Chen, Y.; Lan, G.; Pasiliao, E. An accelerated linearized alternating direction method of multipliers. 2014. arxiv 1401.6607. [Online]. Available: http://arxiv.org/abs/1401.6607
23. Eckstein J, Bertsekas DP. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. Mathematical Programming. Apr; 1992 55(1–3):293–318.
24. Lee J, Sun Y, Saunders M. Proximal Newton-type methods for convex optimization. NIPS. 2012; 25:827–35. [Online]. Available: http://papers.nips.cc/paper/4740-proximal-newton-type-methods-for-convex-optimization.
25. Zhu C, Byrd RH, Lu P, Nocedal J. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. ACM Trans Math Software. Dec; 1997 23(4):550–60.
26. Sion M. On general minimax theorems. Pacific J Math. 1958; 8(1):171–6.

Appendix

A. Proof of Theorem 1

To prove Theorem 1 we first define $\mathbf{R}^H = [\mathbf{r}_1, \ldots, \mathbf{r}_M]$, where $\mathbf{r}_m$ denotes the mth column of $\mathbf{R}^H$. We then recognize that the inner product of two compactly-supported vectors can be computed over either vector's support, i.e.,

$$\langle \mathbf{r}_m, \mathbf{r}_l \rangle = \sum_{n \in \mathscr{S}_m} r_{m,n} r^*_{l,n} = \sum_{n \in \mathscr{S}_l} r_{m,n} r^*_{l,n}. \quad (21)$$

The entries in $\mathbf{V} := \mathbf{R}\mathbf{D}_f \mathbf{R}^H$ are weighted inner products of the form,

$$v_{m,l} = \langle \mathbf{r}_m, \mathbf{r}_l \rangle_{\mathbf{D}_f}, \quad (22)$$

where $v_{m,l}$ is the mth, lth entry of $\mathbf{V}$. We recall that if $\mathbf{D}_1 \succeq \mathbf{D}_f$, then $\mathbf{R}\mathbf{D}_1\mathbf{R}^H \succeq \mathbf{R}\mathbf{D}_f\mathbf{R}^H$. One such $\mathbf{D}_1$ is a diagonal matrix where the diagonal entries are defined as

$$d_{1,n} = \begin{cases} t_1, & \text{if } n \in \mathscr{S}_1 \\ d_{n,f}, & \text{otherwise} \end{cases} \quad (23)$$

We also note that finding a $\mathbf{D}_R \succeq \mathbf{R}\mathbf{D}_f\mathbf{R}^H$ is equivalent to finding a $\mathbf{D}_R$ such that $\mathbf{w}^H \mathbf{D}_R \mathbf{w} \geq \mathbf{w}^H \mathbf{R}\mathbf{D}_f\mathbf{R}^H\mathbf{w}$ for any vector $\mathbf{w}$. To accomplish this, we make the partition, $\mathbf{R}^H = [\mathbf{r}_1, \mathbf{R}_1^H]$, where $\mathbf{R}_1^H = [\mathbf{r}_2, \ldots, \mathbf{r}_M]$. We also make the partition, $\mathbf{w} = [w_1, \mathbf{w}_2]$. We now have

$$\begin{aligned} \mathbf{w}^H \mathbf{R}\mathbf{D}_1\mathbf{R}^H\mathbf{w} &= t_1 |w_1|^2 \|\mathbf{r}_1\|_2^2. \\ + t_1 \sum_{m=2}^{M} 2\Re\{w_1 w_m^* \langle \mathbf{r}_1, \mathbf{r}_m \rangle\} &+ \mathbf{w}_2^H \mathbf{R}_1 \mathbf{D}_1 \mathbf{R}_1^H \mathbf{w}_2 \\ &\leq t_1 |w_1|^2 \|\mathbf{r}_1\|_2^2 \\ + t_1 \sum_{m=2}^{M} (|w_1|^2 + |w_m|^2) |\langle \mathbf{r}_1, \mathbf{r}_m \rangle| &+ \mathbf{w}_2^H \mathbf{R}_1 \mathbf{D}_1 \mathbf{R}_1^H \mathbf{w}_2, \end{aligned} \quad (24)$$

which comes from applying (21) and (22). This implies that

$$\mathbf{R}\mathbf{D}_f\mathbf{R}^H \preccurlyeq \begin{bmatrix} t_1\sum_{m=1}^{M}|\langle \mathbf{r}_1, \mathbf{r}_m\rangle| & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_1\mathbf{D}_1\mathbf{R}_1^H \end{bmatrix} + \begin{bmatrix} 0 & & & \\ & t_1|\langle \mathbf{r}_1, \mathbf{r}_2\rangle| & & \\ & & \ddots & \\ & & & t_1|\langle \mathbf{r}_1, \mathbf{r}_M\rangle| \end{bmatrix}. \quad (25)$$

Without loss of generality, we can assume that $t_1 \le t_2 \le \ldots \le t_M$. If this is not satisfied, then the appropriate permutation can be applied to $\mathbf{V}$ to make it so. The procedure can again be applied to $\mathbf{R}_1\mathbf{D}_1\mathbf{R}_1^H$, and then again recursively. Applying this procedure recursively through M gives Theorem 1.

B. Analysis Denoising Derivation

For the extension of the results in [16], we assume $\mathbf{R} \in \mathbb{R}^{M\times N}$, which includes the classes of total variation and undecimated wavelet regularizers that are of interest to us. The difficulty in minimizing (12) is the fact that $\mathbf{R}$ mixes different elements of $\mathbf{x}$. To decouple the mixing effects, we will introduce dual variables. Let $\gamma \in \mathbb{R}$ and $\nu \in \mathbb{R}$ be two variables and define $\mathcal{P} = \{(\gamma, \nu) \in \mathbb{R}^2 : \gamma^2 + \nu^2 \le 1\}$. Then for any $c \in \mathbb{C}$ we have

$$|c| = \max_{(\gamma,\nu)\in\mathcal{P}} \{\gamma\Re\{c\} + \nu\Im\{c\}\}, \quad (26)$$

where $\Im\{\cdot\}$ returns the imaginary part of its argument. Noting this, we now have

$$\|\mathbf{R}\mathbf{x}\|_1 = \max_{(\boldsymbol{\gamma},\boldsymbol{\nu})\in\mathcal{P}^M} \{\boldsymbol{\gamma}^T\mathbf{R}\Re\{\mathbf{x}\} + \boldsymbol{\nu}^T\mathbf{R}\Im\{\mathbf{x}\}\}, \quad (27)$$

where $\mathcal{P}^M$ is a Cartesian product of M sets of the form of $\mathcal{P}$. Note that $\mathcal{P}^M$ is the ℓ_∞-unit ball in $\mathbb{C}^M$. We also have

$$\begin{aligned} \mathbf{x}^{(k+1)} &= \operatorname*{argmin}_{\mathbf{x}\in\mathcal{M}} \max_{(\boldsymbol{\gamma},\boldsymbol{\nu})\in\mathcal{P}^M} \theta_k((\boldsymbol{\gamma},\boldsymbol{\nu}),\mathbf{x}), \\ \theta_k((\boldsymbol{\gamma},\boldsymbol{\nu}),\mathbf{x}) &= \tfrac{1}{2}\|\mathbf{b}^{(k)} - \mathbf{x}\|_{\mathbf{D}_f}^2 + \beta(\boldsymbol{\gamma}^T\mathbf{R}\Re\{\mathbf{x}\} + \boldsymbol{\nu}^T\mathbf{R}\Im\{\mathbf{x}\}). \end{aligned} \quad (28)$$

To simplify notation, we will now drop the "arg" and implicitly take $\mathbf{x}$ from wherever the critical point of the cost function is. Since $\mathcal{M}$ is a convex set, $\mathcal{P}^M$ is a compact, convex set, and (28) is convex in $\mathbf{x}$ and concave in $(\boldsymbol{\gamma}, \boldsymbol{\nu})$, we apply Sion's Theorem [26] to exchange the order of maximization and minimization, which gives

$$\max_{(\boldsymbol{\gamma},\boldsymbol{\nu})\in\mathcal{P}^M} \min_{\mathbf{x}\in\mathcal{M}} \theta_k((\boldsymbol{\gamma},\boldsymbol{\nu}),\mathbf{x}). \quad (29)$$

Now we use the fact that since $\mathbf{D}_f \in \mathbb{R}^{N\times N}$ and is diagonal, the weighted 2-norm squared is separable into its real and imaginary parts, i.e.,

$$\|\mathbf{b}^{(k)}-\mathbf{x}\|_{\mathbf{D}_f}^2=\|\Re\{\mathbf{b}^{(k)}\}-\Re\{\mathbf{x}\}\|_{\mathbf{D}_f}^2+\|\Im\{\mathbf{b}^{(k)}\}-\Im\{\mathbf{x}\}\|_{\mathbf{D}_f}^2. \quad (30)$$

Defining $\mathbf{q} = \gamma + i\nu$, the inner minimization in (29) has a solution where $\mathbf{x}=P_{\mathcal{M}}(\mathbf{b}^{(k)}-\beta\mathbf{D}_f^{-1}\mathbf{R}^T\mathbf{q})$. As stated previously, the $P_{\mathcal{M}}(\cdot)$ operator simply sets elements outside the mask to zero. Plugging this back into (29) reveals a new maximization problem:

$$\begin{aligned}&\max_{\mathbf{q}\in\mathscr{P}^M}\left\{-\tfrac{1}{2}\|\mathbf{b}^{(k)}-\beta\mathbf{D}_f^{-1}\mathbf{R}^T\mathbf{q}\|_{\mathbf{D}_f}^2+\tfrac{1}{2}\|P_{\mathcal{M}}(\mathbf{b}^{(k)}-\beta\mathbf{D}_f^{-1}\mathbf{R}^T\mathbf{q})-(\mathbf{b}^{(k)}-\beta\mathbf{D}_f^{-1}\mathbf{R}^T\mathbf{q})\|_{\mathbf{D}_f}^2\right\}\\&=\min_{\mathbf{q}\in\mathscr{P}^M}\left\{\tfrac{1}{2}\|\mathbf{b}^{(k)}-\beta\mathbf{D}_f^{-1}\mathbf{R}^T\mathbf{q}\|_{\mathbf{D}_f}^2-\tfrac{1}{2}\|P_{\mathcal{M}}(\mathbf{b}^{(k)}-\beta\mathbf{D}_f^{-1}\mathbf{R}^T\mathbf{q})-(\mathbf{b}^{(k)}-\beta\mathbf{D}_f^{-1}\mathbf{R}^T\mathbf{q})\|_{\mathbf{D}_f}^2\right\}\\&=\min_{\mathbf{q}\in\mathscr{P}^M}\left\{\tfrac{1}{2}\|P_{\mathcal{M}}(\mathbf{b}^{(k)}-\beta\mathbf{D}_f^{-1}\mathbf{R}^T\mathbf{q})\|_{\mathbf{D}_f}^2\right\}.\end{aligned} \quad (31)$$

Since the target cost function is now a constrained minimization over a quadratic, we can once again apply the separable quadratic surrogates techniques outlined in Section II. We choose to do this instead of developing other quadratic minimization routines due to the presence of the constraint. Applying this procedure gives the minimization problem over a surrogate:

$$\begin{aligned}\mathbf{q}^{(j+1)}&=\operatorname*{argmin}_{\mathbf{q}\in\mathscr{P}^M}\left\{\tfrac{1}{2}\|\mathbf{q}-(\mathbf{q}^{(j)}-\beta^{-1}\mathbf{D}_R^{-1}\mathbf{R}\mathbf{x}^{(k,j+1)})\|_{\mathbf{D}_R}^2\right\}\\\mathbf{x}^{(k,j+1)}&:=P_{\mathcal{M}}(\mathbf{b}^{(k)}-\beta\mathbf{D}_f^{-1}\mathbf{R}^T\mathbf{q}^{(j)})\end{aligned} \quad (32)$$

for $\mathbf{D}_R \succcurlyeq \mathbf{R}\mathbf{D}_f^{-1}\mathbf{R}^T$. This is obtained by recognizing that the $P_{\mathcal{M}}(\cdot)$ operator where $\mathcal{M}$ is a masking set can be formulated as a projection matrix, $\mathbf{M}$, where $P_{\mathcal{M}}(\cdot) = \mathbf{M}(\cdot)$. The Hessian in (31) arising from the inclusion of this linear projection matrix is $\beta^2\mathbf{R}\mathbf{D}_f^{-1}\mathbf{M}^T\mathbf{D}_f\mathbf{M}\mathbf{D}_f^{-1}\mathbf{R}^T$, which is upper bounded by $\beta^2\mathbf{R}\mathbf{D}_f^{-1}\mathbf{R}^T$. The majorize-minimize algorithm arising from using this surrogate with momentum acceleration and adaptive momentum restart is shown in Fig. 4.

initialize $k = 0, \mathbf{z}^{(0)} = \mathbf{x}^{(0)}, \mathbf{D}_f, \alpha$
while $k < K$ **do**
  $\tau^{(k+1)} = (1 + \sqrt{1 + 4(\tau^{(k)})^2})/2$
  $\mathbf{x}^{(k+1)} = \underset{\mathbf{x} \in \mathcal{M}}{\operatorname{argmin}}\ \eta(\mathbf{x}, \mathbf{z}^{(k)})$
  $\kappa = \left\|\mathbf{z}^{(k)} - \mathbf{x}^{(k+1)}\right\|_2 \left\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\right\|_2$
  if $\Re\left\{\left\langle \mathbf{z}^{(k)} - \mathbf{x}^{(k+1)}, \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} \right\rangle\right\} > \alpha\kappa$ **then**
    $\mathbf{z}^{(k+1)} = \mathbf{x}^{(k+1)}$
    $\tau^{(k+1)} = 1$
  else
    $\mathbf{z}^{(k+1)} = \mathbf{x}^{(k+1)} + \frac{\tau^{(k)} - 1}{\tau^{(k+1)}}(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)})$
  end if
  $k = k + 1$
end while
$\hat{\mathbf{x}} = \mathbf{x}^{(K)}$

Fig. 1.
BARISTA: B1-based, Adaptive Restart, Iterative Soft Thresholding Algorithm

initialize $k = 0, \mathbf{u}^{(0)} = \mathbf{R}\mathbf{x}^{(0)}, \mathbf{z}^{(0)} = \mathbf{u}^{(0)}, \mathbf{D}_f, \alpha$
calculate $\mathbf{D}_R \succeq \mathbf{R}\mathbf{D}_f\mathbf{R}^H$ according to Theorem 1
while $k < K$ **do**
    $\tau^{(k+1)} = (1 + \sqrt{1 + 4(\tau^{(k)})^2})/2$
    $\mathbf{b}^{(k)} = \mathbf{z}^{(k)} - \mathbf{D}_R^{-1}\mathbf{B}^H(\mathbf{B}\mathbf{z}^{(k)} - \mathbf{y})$
    $\mathbf{u}^{(k+1)} = P_{\mathcal{M}_{\text{synth}}}(\text{shrink}(\mathbf{b}^{(k)}, \beta\mathbf{d}_R^{-1}))$
    $\kappa = \left\|\mathbf{z}^{(k)} - \mathbf{u}^{(k+1)}\right\|_2 \left\|\mathbf{u}^{(k+1)} - \mathbf{u}^{(k)}\right\|_2$
    if $\Re\left\{\left\langle \mathbf{z}^{(k)} - \mathbf{u}^{(k+1)}, \mathbf{u}^{(k+1)} - \mathbf{u}^{(k)} \right\rangle\right\} > \alpha\kappa$ **then**
        $\mathbf{z}^{(k+1)} = \mathbf{u}^{(k+1)}$
        $\tau^{(k+1)} = 1$
    else
        $\mathbf{z}^{(k+1)} = \mathbf{u}^{(k+1)} + \frac{\tau^{(k)}-1}{\tau^{(k+1)}}(\mathbf{u}^{(k+1)} - \mathbf{u}^{(k)})$
    end if
    $k = k + 1$
end while
$\hat{\mathbf{x}} = \mathbf{R}^H\mathbf{u}^{(K)}$

Fig. 2.
BARISTA for synthesis

initialize $k = 0, \mathbf{z}^{(0)} = \mathbf{x}^{(0)}, \mathbf{D}_f, \alpha, \epsilon^{(0)}$
calculate $\mathbf{D}_R \succeq \mathbf{R}\mathbf{D}_f^{-1}\mathbf{R}^T$ as outlined in Section IV-C
while $k < K$ **do**
    $\tau^{(k+1)} = (1 + \sqrt{1 + 4(\tau^{(k)})^2})/2$
    Run Fig. 4 algorithm to $\epsilon^{(k)}$ convergence to get $\mathbf{x}^{(k+1)}$
    $\kappa = \left\|\mathbf{z}^{(k)} - \mathbf{x}^{(k+1)}\right\|_2 \left\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\right\|_2$
    if $\Re\left\{\left\langle \mathbf{z}^{(k)} - \mathbf{x}^{(k+1)}, \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} \right\rangle\right\} > \alpha\kappa$ **then**
        $\mathbf{z}^{(k+1)} = \mathbf{x}^{(k+1)}$
        $\tau^{(k+1)} = 1$
    else
        $\mathbf{z}^{(k+1)} = \mathbf{x}^{(k+1)} + \frac{\tau^{(k)}-1}{\tau^{(k+1)}}(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)})$
    end if
    $\epsilon^{(k+1)} = \max\left(\min\left(\epsilon_{\text{diff}} \frac{\left\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\right\|_2}{\left\|\mathbf{x}^{(k)}\right\|_2}, \epsilon^{(k)}\right), \epsilon_{\min}\right)$
    $k = k + 1$
end while
$\hat{\mathbf{x}} = \mathbf{x}^{(K)}$

Fig. 3.
BARISTA for analysis

initialize $j = 0, \mathbf{q}^{(0)}, \mathbf{v}^{(0)} = \mathbf{v}^{(0)}, \mathbf{D}_f, \mathbf{D}_R, \epsilon, \alpha$
repeat
  $\tau^{(j+1)} = (1 + \sqrt{1 + 4(\tau^{(j)})^2})/2$
  $\mathbf{x}^{(k,j+1)} = P_{\mathcal{M}}(\mathbf{b}^{(k)} - \beta \mathbf{D}_f^{-1} \mathbf{R}^T \mathbf{v}^{(j)})$
  $\mathbf{q}^{(j+1)} = P_{\mathcal{P}^M}(\mathbf{v}^{(j)} - \beta^{-1} \mathbf{D}_R^{-1} \mathbf{R} \mathbf{x}^{(k,j+1)})$
  $\kappa = \left\| \mathbf{v}^{(k)} - \mathbf{q}^{(k+1)} \right\|_2 \left\| \mathbf{q}^{(k+1)} - \mathbf{q}^{(k)} \right\|_2$
  if $\Re \left\{ \left\langle \mathbf{v}^{(k)} - \mathbf{q}^{(k+1)}, \mathbf{q}^{(k+1)} - \mathbf{q}^{(k)} \right\rangle \right\} > \alpha\kappa$ **then**
    $\mathbf{v}^{(k+1)} = \mathbf{q}^{(k+1)}$
    $\tau^{(k+1)} = 1$
  else
    $\mathbf{v}^{(k+1)} = \mathbf{q}^{(k+1)} + \frac{\tau^{(k)} - 1}{\tau^{(k+1)}} (\mathbf{q}^{(k+1)} - \mathbf{q}^{(k)})$
  end if
  $j = j + 1$
until $\frac{\left\| \mathbf{x}^{(k,j)} - \mathbf{x}^{(k,j-1)} \right\|_2}{\left\| \mathbf{x}^{(k,j-1)} \right\|_2} \leq \epsilon$
$\mathbf{x}^{(k+1)} = P_{\mathcal{M}}(\mathbf{b}^{(k)} - \beta \mathbf{D}_f^{-1} \mathbf{R}^T \mathbf{q}^{(j)})$

Fig. 4.
Analysis denoising algorithm

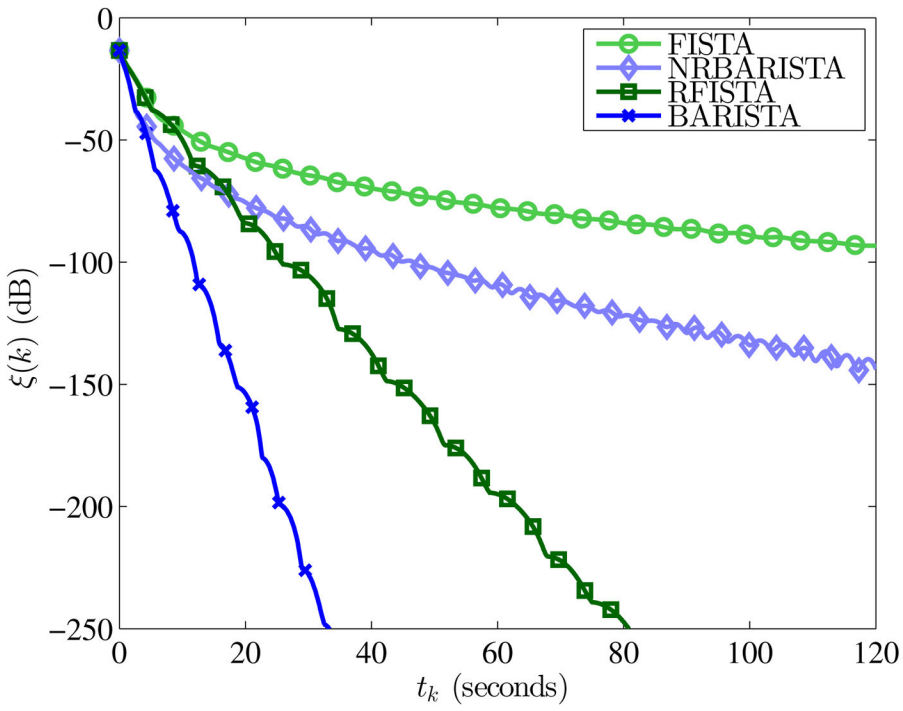
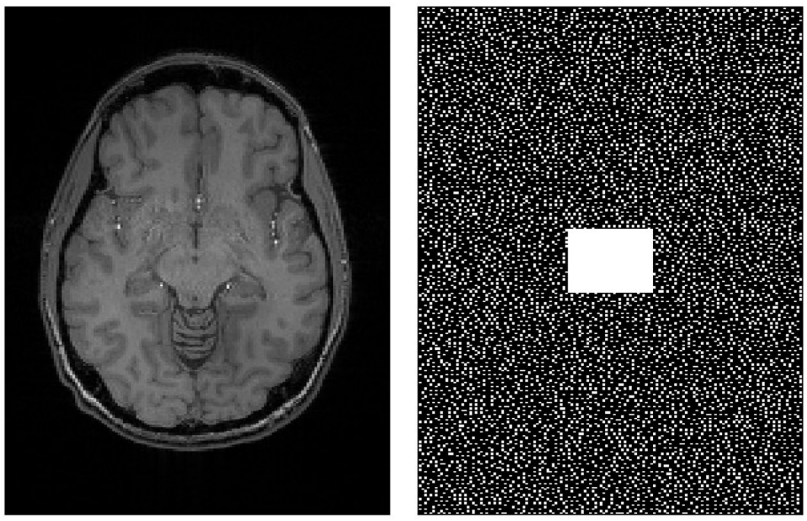


Fig. 5.

Comparison of different majorize-minimize methods with orthogonal Haar wavelet regularization. Markers are placed at 50 iteration intervals. FISTA used $\mathbf{D}_f = \mathbf{D}_R = L\mathbf{I}$ while BARISTA and NRBARISTA (non-restart BARISTA) used the proposed $\mathbf{D}_f$ and $\mathbf{D}_R$. BARISTA is the fastest method; this was also observed for the other experiments with varying degrees of acceleration. Both restart methods exhibit a stair step pattern, where new "steps" arise when the momentum is restarted.



(a) (b)

Fig. 6.

Images corresponding to the *in vivo* experiments. (a) **x** estimated from fully sampled data. Some residual noise is present at the center. (b) Sampling pattern for the *in vivo* experiments with a densely sampled 32 × 32 center.

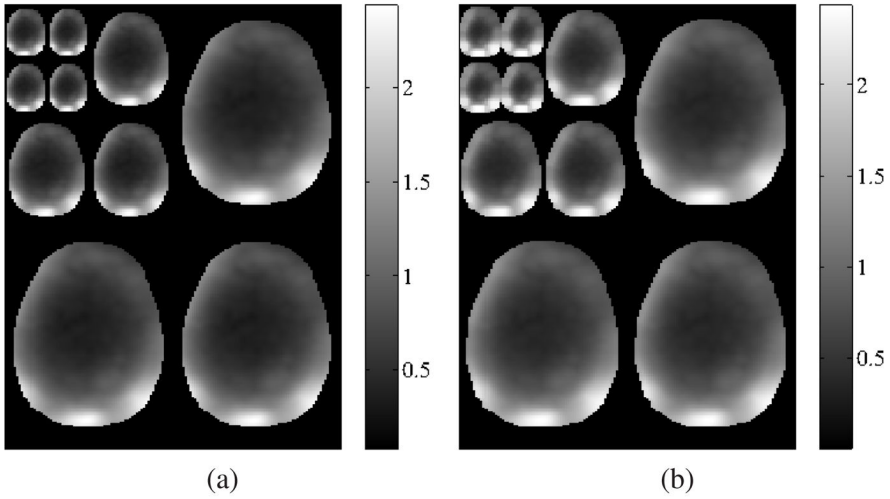
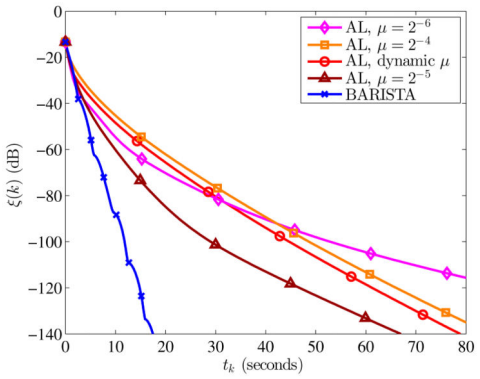
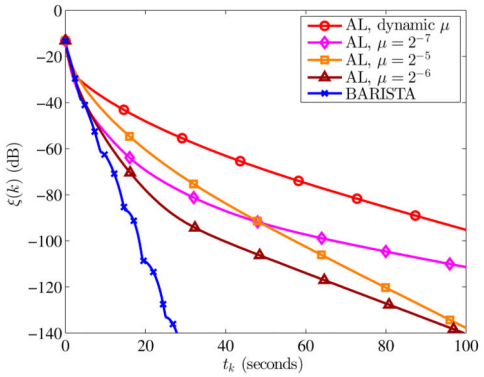


Fig. 7.

Examples of diagonal elements of $\mathbf{D}_R$ for synthesis regularizers, rearranged into an image. (a) Elements of the diagonal of $\mathbf{D}_R$ in the Haar wavelet basis. Areas outside the brain have been masked for presentation. (b) Elements of $\mathbf{D}_R$ for the Daubechies D4 basis. Since the rows of a Daubechies D4 matrix have larger support than those of the Haar, the majorizer is smoother. For both cases color bars are shown to give a sense of the variation across the image caused by the sensitivity coils. The Lipschitz constant was 2.98, while the maximum value of the squared absolute values of the sensitivity coils was 3.36. Many of the entries in $\mathbf{D}_R$ are smaller than the Lipschitz constant.



(a) Orthogonal Haar Wavelet Regularization



(b) Orthogonal Daubechies D4 Wavelet Regularization

Fig. 8.

Summary of convergence results for two different synthesis regularizers. Markers are placed at 30 iteration intervals for all algorithms. (a) Convergence plot comparing the proposed method to variable splitting methods for orthogonal Haar wavelets. The proposed method with momentum restarting is faster than the other methods. (b) Another convergence plot with orthogonal Daubechies D4 wavelets.

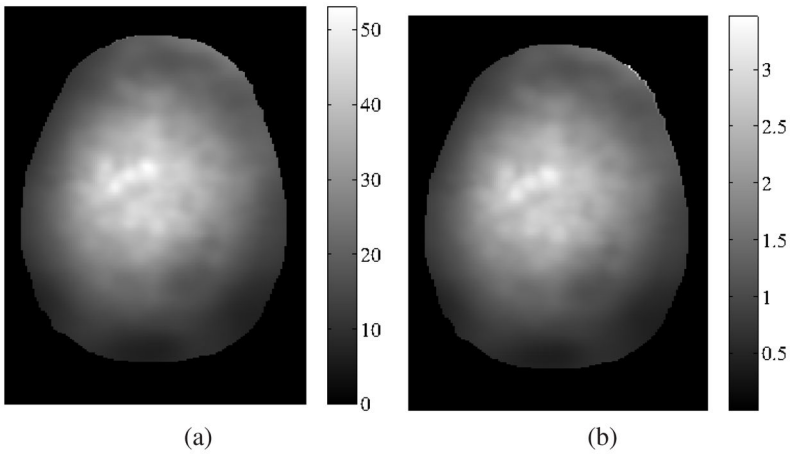
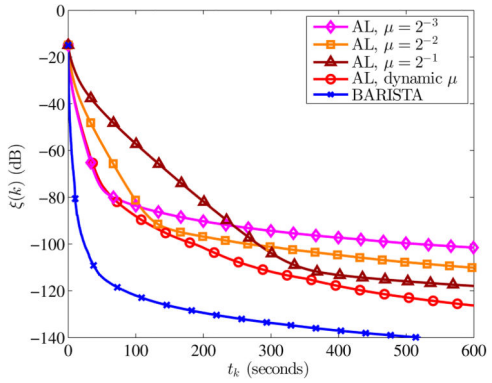
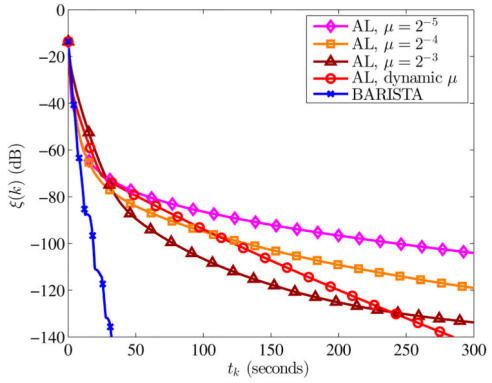


Fig. 9.

Examples of diagonal elements of $\mathbf{D}_R$ for analysis regularizers rearranged into an image. (a) A subset of the elements of $\mathbf{D}_R$ for the total variation case with areas outside the brain masked for presentation. Since this matrix must upper bound $\mathbf{R}\mathbf{D}_f^{-1}\mathbf{R}^T$, the sensitivity elements have been inverted. (b) A subset of the elements of $\mathbf{D}_R$ for the undecimated Haar wavelet case.



(a) Total Variation Regularization



(b) Undecimated Haar Wavelet Regularization

Fig. 10.

Summary of convergence results for two different analysis regularizers. (a) Convergence plot comparing the proposed method to variable splitting methods for total variation regularization. Markers are placed at 100 iteration intervals. The proposed method with momentum restarting is faster than the other methods. (b) Convergence plot comparing BARISTA to variable splitting methods with undecimated Haar wavelet regularization. Markers are placed at 30 iteration intervals. The proposed method with momentum restarting is faster than the other methods, especially in later iterations.