

# A FAST ALGORITHM FOR TOTAL VARIATION IMAGE RECONSTRUCTION FROM RANDOM PROJECTIONS

YUNHAI XIAO\* AND JUNFENG YANG†

**Abstract.** Total variation (TV) regularization is popular in image restoration and reconstruction due to its ability to preserve image edges. To date, most research activities on TV models concentrate on image restoration from blurry and noisy observations, while discussions on image reconstruction from random projections are relatively fewer. In this paper, we propose, analyze, and test a fast alternating minimization algorithm for image reconstruction from random projections via solving a TV regularized least-squares problem. The per-iteration cost of the proposed algorithm involves a linear time shrinkage operation, two matrix-vector multiplications and two fast Fourier transforms. Convergence, certain finite convergence and  $q$ -linear convergence results are established, which indicate that the asymptotic convergence speed of the proposed algorithm depends on the spectral radii of certain submatrix. Moreover, to speed up convergence and enhance robustness, we suggest an accelerated scheme based on an inexact alternating direction method. We present experimental results to compare with an existing algorithm, which indicate that the proposed algorithm is stable, efficient and competitive with TwIST [3] — a state-of-the-art algorithm for solving TV regularization problems.

**Key words.** Total variation, image restoration, image reconstruction, compressive sensing, alternating direction method

**AMS subject classifications.** 68U10, 65J22, 65K10, 65T50, 90C25

**1. Introduction.** Image restoration and reconstruction play important roles in medical and astronomical imaging, image and video coding, file restoration, and many other applications. Let  $\bar{u} \in \mathbb{R}^{n^2}$  be an original  $n \times n$  image,  $A \in \mathbb{R}^{m \times n^2}$  be a linear operator, and  $f \in \mathbb{R}^m$  be an observation which satisfies the relationship

$$f = \mathbb{N}(A\bar{u}) \in \mathbb{R}^m, \quad (1.1)$$

where  $\mathbb{N}(\cdot)$  represents a noise contamination or corruption procedure. Given  $A$ , image restoration and reconstruction extract  $\bar{u}$  from  $f$ , which is either under-determined ( $m < n^2$ ) or ill-posed (e.g., deconvolution/deblurring), making classical least-squares approximation alone not suitable. To stabilize recovery, regularization technique is frequently used, giving a general reconstruction model of the form

$$\min_u \Phi_{\text{reg}}(u) + \mu \Phi_{\text{fid}}(Au - f), \quad (1.2)$$

where  $\Phi_{\text{reg}}(u)$  promotes solution regularity such as smoothness and sparseness,  $\Phi_{\text{fid}}(Au - f)$  fits the observed data by penalizing the difference between  $Au$  and  $f$ , and  $\mu > 0$  balances the two terms for minimization. The choice of  $\Phi_{\text{fid}}(\cdot)$  depends on different noise, e.g., the squared  $\ell_2$  penalty is usually used for Gaussian additive noise, while the  $\ell_1$  penalty is more appropriate for certain non-Gaussian noise, e.g., salt-and-pepper noise. Throughout this paper, we assume that  $\mathbb{N}(\cdot)$  represents an additive Gaussian noise contamination and thus set  $\Phi_{\text{fid}}(\cdot) = \|\cdot\|_2^2$ . Among other regularization, total variation (TV) has been popular ever since its introduction by Rubin, Osher and Fatemi [27]. The remarkable property of TV is to preserve edges due

---

\*Institute of Applied Mathematics, College of Mathematics and Information Science, Henan University, Kaifeng, 475004, P. R. China. Current Position: Department of Mathematics, Nanjing University, Nanjing, 210093, P. R. China (yhxiao-math@gmail.com). This author's work was supported by the Chinese Post-doctoral Foundation grant 20090461094.

†Department of Mathematics, Nanjing University, Nanjing, 210093, P. R. China (send correspondence to this author at: jfyang@nju.edu.cn). This author's work was supported in part by the Natural Science Foundation of China grant NSFC-10971095 and the Natural Science Foundation of Jiangsu Province BK2008255.

to its linear penalty on differences between adjacent pixels. The most widely studied TV model for image deconvolution (in which case  $A \in \mathbb{R}^{n^2 \times n^2}$  is a convolution matrix) is

$$\min_u \sum_{i=1}^{n^2} \|D_i u\|_2 + \frac{\mu}{2} \|Au - f\|_2^2, \quad (1.3)$$

where  $D_i \in \mathbb{R}^{2 \times n^2}$  denotes the local finite difference operator (with certain boundary conditions) at pixel  $i$ , and  $\sum_i \|D_i u\|_2$  is a discretization of the TV of  $u$ . In this paper, we propose, analyze and test a fast alternating minimization algorithm for solving (1.3), in which  $A$  is a compressive sensing encoding matrix ( $m < n^2$ ) and does not have structures.

In the following of this section, we review briefly compressive sensing ideas and algorithms, which provide theoretical guarantee for image reconstruction via solving (1.3), examine some existing algorithms for relevant TV problems, and describe the contributions and organization of this paper. Throughout this paper, we refer to (1.3) as TV/L<sup>2</sup>.

**1.1. Compressive sensing — ideas and algorithms.** Compressive sensing (CS) is an emerging methodology in digital signal processing brought to the research forefront by Donoho [11], Candès, Romberg and Tao [6, 7], and has attracted intensive research activities in the past few years. In a nutshell, CS first encodes a sparse signal (possibly under certain sparsifying basis) through hardware devices into a relatively small number of linear projections and then reconstructs it from the limited measurements. Let  $\bar{x} \in \mathbb{R}^n$  be the sparse signal that we wish to capture, i.e., the number of nonzeros in  $\bar{x}$  is much less than its length  $n$ , and  $b = A\bar{x}$  represent a set of  $m$  (usually much smaller than  $n$ ) linear projections of  $\bar{x}$ . Under certain desirable conditions, it is shown that with high probability the basis pursuit problem

$$\min_x \|x\|_1 \quad \text{s.t. } Ax = b, \quad (1.4)$$

yields the sparsest solution of the linear system  $Ax = b$ , see [12]. More often than not,  $b$  contains noise, in which case certain relaxation is desirable. For white Gaussian noise, the most widely used models are the basis pursuit denoising problem

$$\min_x \|x\|_1 + \frac{\mu}{2} \|Ax - b\|_2^2, \quad (1.5)$$

where, roughly speaking,  $\mu > 0$  is inversely proportional to the noise level, and its variants. Since most signals of interests are sparse or nearly sparse (called compressible) under certain basis, the CS idea has extremely wide applications. Recent results show that stable reconstruction can be obtained provided that  $A$  possesses certain randomness. It has been clear from [41] that for almost all random matrices the exact recoverability is approximatively identical. Moreover, exact recoverability is attainable when  $A$  contains randomly taken rows from orthonormal matrices, e.g., partial Fourier which arises from magnetic resonance imaging [22].

In the application of CS, matrix  $A$  is large and dense. Furthermore, in certain applications  $A$  contains structures that allow fast matrix-vector multiplication, e.g.,  $A$  is a partial Fourier matrix as in MRI. These features make traditional powerful optimization approaches such as interior point methods not suitable. In comparison, first-order algorithms that depend on merely matrix-vector multiplications are more desirable. Therefore, in the last few years numerous algorithms have been proposed for recovering sparse signals via solving certain  $\ell_1$ -norm regularized problems including (1.4), (1.5) and their variants. Several well-known approaches in this area include the gradient projection method [15], the fixed-point continuation method [19], the spectral projected gradient method [30], and the Bregman iterative method [24, 40, 25, 4, 5]. More recent algorithms can be found in [2, 8, 35, 37].

**1.2. Some existing algorithms for TV/L<sup>2</sup>.** The advantage of TV regularization compared with Tikhonov-like [29] regularization in recovering high quality image is not without a price. The nondifferentiability of TV causes the main difficulty. In addition, problems arising from signal and image reconstruction are usually large scale and ill-posed, which further make TV models difficult to be solved efficiently. Since the introduction of TV regularization, many algorithms have been proposed for solving (1.3) and its variants. In the pioneer work [27], a time-marching scheme was used to solve a partial differential equation system, which in optimization point of view is equivalent to a constant step-length gradient descent method. This time-marching scheme suffers slow convergence especially when the iterate point approaches the solution set. Another well-known method is the linearized gradient method proposed in [31] for denoising and in [32] for deblurring, which solves the Euler-Lagrangian equation via a fixed-point iteration. At each iteration of the linearized gradient method, a linear system needs to be solved, which makes the per-iteration cost extremely expensive especially when the problem becomes more ill-conditioned. To overcome the linear convergence of first-order methods, the authors of [31] incorporated Newton method to solve (1.3), which achieved superlinear convergence at the cost of solving a large linear system at each iteration. Another important approach for TV problems is the iterative shrinkage/thresholding (IST) method [13, 14, 28]. In [3], Bioucas-Dias and Figueiredo introduced a two-step IST (TwIST) algorithm, which exhibits much faster convergence than the primary IST algorithm for ill-conditioned problems. We note that IST-based algorithms require to solve a TV denoising subproblem at each iteration which requires its own iterations.

Despite the progress have been achieved, algorithms for solving (1.3) are still much slower than those for Tikhonov regularization problems. Recently, a fast TV deconvolution (FTVd) method is proposed in [33], which makes full use of problem structures (both  $A$  and finite difference operators have circulant structures under proper boundary conditions) and thus converges very fast. FTVd solves a penalty approximation of (1.3), that is

$$\min_{u, \mathbf{w}} \sum_{i=1}^{n^2} \left( \|\mathbf{w}_i\|_2 + \frac{\beta}{2} \|\mathbf{w}_i - D_i u\|_2^2 \right) + \frac{\mu}{2} \|Au - f\|_2^2, \quad (1.6)$$

where, for each  $i$ ,  $\mathbf{w}_i \in \mathbb{R}^2$  is an auxiliary variable and  $\beta > 0$  is a penalty parameter. The advantage of considering (1.6) is that it leads to fast and efficient alternating minimizations for deconvolution problems. The numerical results given in [33] indicates that FTVd is much faster than the lagged diffusivity method in [31], which is known to be efficient previously. For more details on the FTVd algorithm and its performance, see [33]. Given the practical efficiency of FTVd, this split and penalty idea has been extended to multichannel image restoration in [36], impulsive noise elimination in [38] and medical reconstruction from partial Fourier coefficients in [39]. More algorithms for TV/L<sup>2</sup> problem can be found in [9, 10, 21, 23, 26, 34] and references therein.

**1.3. Contributions.** The purpose of this paper is to develop a fast algorithm for solving (1.3), where  $A$  is a general linear operator. Specifically, we are interested in compressive sensing encoding matrices in which case  $A$  contains smaller or even much smaller number of rows than columns. As is stated above, problem (1.6) admits fast alternating minimization when  $A$  is a convolution matrix. As a matter of fact, the minimization of (1.6) with respect to  $\mathbf{w}_i$ ,  $i = 1, 2, \dots, n^2$ , reduces to  $n^2$  two-dimensional problems (no matter what  $A$  is), which can be solved easily and exactly in linear time. However, different from deconvolution problems,  $A$  does not have structures in our stated case. Consequently, the solution of  $u$ -subproblems can not utilize any fast transforms.

In this paper, we first introduce a fast alternating minimization scheme for solving (1.6), which recurs to linearization and proximal techniques when solving the  $u$ -subproblems. Under quite reasonable technical

assumptions, we show that the proposed algorithm converges globally to a solution of (1.6). Moreover, we establish  $q$ -linear convergence results which indicate that the  $q$ -linear factor depends on the spectral radius of certain submatrix. Clearly, the solution of (1.6) well approximates that of (1.3) only when  $\beta$  is sufficiently large, which causes numerical difficulties in computation. To overcome this drawback, we introduce an inexact alternating direction method, which accelerates the convergence of the alternating minimization approach and converges to a solution of (1.3) without driving  $\beta$  to infinity. Since the proposed algorithms solve (1.6) and (1.3) with a CS encoding matrix, we name the resulting algorithms FTVCS. We present experimental results and compare with TwIST [3]. The comparison results indicate that FTVCS is fast and efficient and performs comparable with the state-of-the art algorithm TwIST.

**1.4. Notation and organization.** Now, we define our notation. For scalars  $\alpha_i$ , vectors  $v_i$ , and matrices  $M_i$  of appropriate sizes,  $i = 1, 2$ , we let  $\alpha = (\alpha_1; \alpha_2) \triangleq (\alpha_1, \alpha_2)^\top$ ,  $v = (v_1; v_2) \triangleq (v_1^\top, v_2^\top)^\top$ , and  $M = (M_1; M_2) \triangleq (M_1^\top, M_2^\top)^\top$ . Let  $D^{(1)}$  and  $D^{(2)}$  be the two first-order finite difference matrices in horizontal and vertical directions, respectively. As is used before,  $D_i \in \mathbb{R}^{2 \times n^2}$  is a two-row matrix formed by stacking the  $i$ th row of  $D^{(1)}$  on that of  $D^{(2)}$ . Throughout this paper, we let  $D = (D^{(1)}; D^{(2)}) \in \mathbb{R}^{2n^2 \times n^2}$ ,  $\rho(T)$  be the spectral radius of matrix  $T$ , and  $\mathcal{P}(\cdot)$  be the projection operator under Euclidean norm. The inner product of two vectors will be denoted by  $\langle u, v \rangle$ . In the rest of this paper, we let  $\|\cdot\| = \|\cdot\|_2$ , and without misleading we abbreviate  $\sum_{i=1}^{n^2}$  as  $\sum_i$ . Additional notation will be introduced when it occurs.

The paper is organized as follows. In Section 2.1, we introduce our alternating minimization algorithm FTVCS and study its convergence properties. An accelerated scheme of FTVCS is proposed in Section 2.2 by incorporating an inexact alternating direction technique. Numerical results in comparison with TwIST are presented in Section 3. Finally, we conclude the paper in Section 4.

**2. Proposed algorithms.** The task of this section is to construct our algorithm for solving (1.3). As is stated above, our interest in this paper concentrates on CS encoding matrices, i.e.,  $A \in \mathbb{R}^{m \times n^2}$  with  $m \ll n$ . The non-smoothness of TV causes the main difficulty. Similar as in [33], we first consider the approximation problem (1.6) and then propose an inexact alternating direction method for the solution of (1.3).

**2.1. Alternating minimization.** The introduction of auxiliary variables  $\mathbf{w}$  in (1.6) makes it easy to apply alternating minimization. It is easy to see that, for fixed  $u$ , the minimization of (1.6) with respect to  $\mathbf{w}$  reduces to the following two-dimensional problems

$$\min_{\mathbf{w}_i \in \mathbb{R}^2} \|\mathbf{w}_i\| + \frac{\beta}{2} \|\mathbf{w}_i - D_i u\|^2, \quad i = 1, 2, \dots, n^2, \quad (2.1)$$

for which the unique minimizers are given by the two-dimensional shrinkage formula

$$\mathbf{w}_i = \max \left\{ \left\| D_i u \right\| - \frac{1}{\beta}, 0 \right\} \frac{D_i u}{\|D_i u\|}, \quad i = 1, \dots, n^2, \quad (2.2)$$

where the convention  $0 \cdot (0/0) = 0$  is followed. On the other hand, for fixed  $\mathbf{w}$ , the minimization of (1.6) with respect to  $u$  is a least squares problem, and the corresponding normal equations are given by

$$\left( \sum_i D_i^\top D_i + \frac{\mu}{\beta} A^\top A \right) u = \sum_i D_i^\top \mathbf{w}_i + \frac{\mu}{\beta} A^\top f,$$

or equivalently,

$$\left( D^\top D + \frac{\mu}{\beta} A^\top A \right) u = D^\top w + \frac{\mu}{\beta} A^\top f, \quad (2.3)$$

where  $w \in \mathbb{R}^{2n^2}$  is an reordering of  $\mathbf{w}_i$ ,  $i = 1, 2, \dots, n^2$ . It is well-known that, under the periodic boundary condition for  $u$ ,  $D^\top D$  is a block-circulant matrix and can be diagonalized by two-dimensional fast Fourier transform (FFT). Unfortunately, the matrix  $A^\top A$  does not have circulant structures for general CS encoding matrices. Therefore, the exact solution of (2.3) is expensive, which causes the main difficulty to apply alternating minimization directly.

To avoid solution of linear system of equations at each iteration, we linearize  $\frac{1}{2}\|Au - f\|^2$  at the current point  $u^k$  and add a proximal term, resulting the following approximation problem

$$\min_{u, \mathbf{w}} \sum_i \left( \|\mathbf{w}_i\| + \frac{\beta}{2} \|\mathbf{w}_i - D_i u\|^2 \right) + \mu \left( g_k^\top (u - u^k) + \frac{1}{2\tau} \|u - u^k\|^2 \right), \quad (2.4)$$

where  $g_k = A^\top (Au^k - f)$  denotes the gradient of  $\frac{1}{2}\|Au - f\|^2$  at  $u^k$ , and  $\tau > 0$  is a parameter. Clearly, problem (2.4) is equivalent to

$$\min_{u, \mathbf{w}} \sum_i \left( \|\mathbf{w}_i\| + \frac{\beta}{2} \|\mathbf{w}_i - D_i u\|^2 \right) + \frac{\mu}{2\tau} \|u - (u^k - \tau g_k)\|^2. \quad (2.5)$$

For fixed  $w$  (or  $\mathbf{w}$ ), the minimization of (2.5) with respect to  $u$  is equivalent to

$$\left( D^\top D + \frac{\mu}{\beta\tau} I \right) u = D^\top w + \frac{\mu}{\beta\tau} (u^k - \tau g_k), \quad (2.6)$$

where we recall that  $D = (D^{(1)}; D^{(2)})$ . Under the periodic boundary conditions for  $u$ , the coefficient matrix in (2.6) can be diagonalized easily by FFT. Consequently, the solution of (2.6) can be accomplished by two FFTs (including one inverse FFT). To sum up, our alternating minimization algorithm, named fast total variation decoding from compressive sensing measurements or FTVCS, is described below.

**ALGORITHM 1** (FTVCS). *Input*  $f$ ,  $A$  and  $\mu, \beta, \tau > 0$ . *Initialize*  $u^0 = f$  and  $k = 0$ .

**While** “not converged”, **Do**

- 1) Compute  $w^{k+1}$  according to (2.2) for fixed  $u = u^k$ .
- 2) Compute  $u^{k+1}$  according to (2.6) for fixed  $w = w^{k+1}$ .
- 3)  $k = k + 1$ .

**End Do**

To establish the convergence of FTVCS, we need the following technical assumption.

**ASSUMPTION 1.**  $\mathcal{N}(A) \cap \mathcal{N}(D) = \{0\}$ , where  $\mathcal{N}(\cdot)$  represents the null space of a matrix.

Assumption 1 is a quite loose condition and commonly used in the convergence analyses of similar studies, see e.g., [33]. Under Assumption 1, we have the following convergence results.

**THEOREM 2.1.** *Under Assumption 1, for any fixed  $\beta > 0$  and  $0 < \tau < 2/\lambda_{\max}(A^\top A)$ , where  $\lambda_{\max}(A^\top A)$  denotes the spectral radius of  $A^\top A$ , the sequence  $\{(w^k, u^k)\}$  generated by Algorithm 1 from any starting point  $(w^0, u^0)$  converges to a solution  $(w^*, u^*)$  of (1.6).*

**THEOREM 2.2.** *Suppose the sequence  $\{(w^k, u^k)\}$  generated by Algorithm 1 converges to  $(w^*, u^*)$ . Then, we have  $\mathbf{w}_i^k = \mathbf{w}_i^* = 0$ ,  $\forall i \in L$  after a finite number of iterations, where  $L = \{i : \|D_i u^*\| \leq 1/\beta\}$ .*

**THEOREM 2.3.** *Under the conditions of Theorem 2.1, the sequence  $\{u^k\}$  generated by Algorithm 1 converges to  $\{u^*\}$   $q$ -linearly.*

The proofs of Theorems 2.1, 2.2 and 2.3 are given in Appendix A.

**2.2. An accelerated scheme based on inexact alternating direction method.** It is well-known that problem (1.6) well approximates (1.3) only when  $\beta$  is sufficiently large. However, it is generally difficult to determine theoretically how large a  $\beta$  value must be to attain a given accuracy. In this section, we present

an inexact alternating direction method (ADM), which converges to a solution of (1.3) without requiring  $\beta$  goes to infinity.

First, we review briefly the idea of ADM pioneered in [17, 18]. The classical ADM is designed to solve the following structure optimization problem:

$$\min_{y,z} \{\theta_1(y) + \theta_2(z) : Hy - z = 0\}, \quad (2.7)$$

where  $\theta_1 : \mathbb{R}^s \rightarrow \mathbb{R}$  and  $\theta_2 : \mathbb{R}^t \rightarrow \mathbb{R}$  are functions, and  $H$  is a  $s \times t$  matrix. Given  $z^k \in \mathbb{R}^t$  and  $p^k \in \mathbb{R}^s$ , the ADM iterates as follows

$$y^{k+1} \leftarrow \arg \min_y \theta_1(y) - (p^k)^\top (Hy - z^k) + \frac{\sigma}{2} \|Hy - z^k\|^2, \quad (2.8)$$

$$z^{k+1} \leftarrow \arg \min_z \theta_2(z) - (p^k)^\top (Hy^{k+1} - z) + \frac{\sigma}{2} \|Hy^{k+1} - z\|^2, \quad (2.9)$$

$$p^{k+1} \leftarrow p^k - \sigma(Hy^{k+1} - z^{k+1}), \quad (2.10)$$

where  $\sigma > 0$  is a parameter. In (2.8),  $p^k$  is the Lagrangian multiplier and  $\sigma$  serves as a penalty parameter. It can be shown that, under quite reasonable assumption, (2.8) converges to a solution of (2.7) for any fixed  $\sigma > 0$ , see [17, 18].

We now consider the model (1.3) in its equivalent form

$$\min_{u,w} \left\{ \sum_i \|\mathbf{w}_i\| + \frac{\mu}{2} \|Au - f\|^2 : \mathbf{w}_i = D_i u, \forall i \right\}. \quad (2.11)$$

The augmented Lagrangian problem of (2.11) is given by

$$\min_{u,w} \sum_i \left( \|\mathbf{w}_i\| - \lambda_i^\top (\mathbf{w}_i - D_i u) + \frac{\beta}{2} \|\mathbf{w}_i - D_i u\|^2 \right) + \frac{\mu}{2} \|Au - f\|^2, \quad (2.12)$$

where, for each  $i$ ,  $\lambda_i \in \mathbb{R}^2$  is the Lagrangian multiplier attached to  $\mathbf{w}_i = D_i u$ . Inspired by the ADM iterations, for given  $(u^k, w^k, \lambda^k)$ , we obtain the next triplet  $(u^{k+1}, w^{k+1}, \lambda^{k+1})$  as follows. First, for fixed  $u^k$  and  $\lambda^k$ , the minimization of (2.12) with respect to  $\mathbf{w}$  is equivalent to

$$\min_{\mathbf{w}_i \in \mathbb{R}^2} \|\mathbf{w}_i\| + \frac{\beta}{2} \|\mathbf{w}_i - (D_i u^k - \lambda_i^k / \beta)\|^2, \quad i = 1, 2, \dots, n^2,$$

the solutions of which are given by

$$\mathbf{w}_i^{k+1} = \max \left\{ \|D_i u^k - \lambda_i^k / \beta\| - \frac{1}{\beta}, 0 \right\} \frac{D_i u^k - \lambda_i^k / \beta}{\|D_i u^k - \lambda_i^k / \beta\|}, \quad i = 1, 2, \dots, n^2. \quad (2.13)$$

Second, for fixed  $w^{k+1}$ ,  $u^k$  and  $\lambda^k$ , the minimization of (2.12) with respect to  $u$  is approximated by linearizing  $\frac{1}{2} \|Au - f\|^2$  and adding a proximal term as in (2.4), resulting the following problem

$$\min_u \sum_i \left( -(\lambda_i^k)^\top (\mathbf{w}_i^{k+1} - D_i u) + \frac{\beta}{2} \|\mathbf{w}_i^{k+1} - D_i u\|^2 \right) + \frac{\mu}{2\tau} \|u - (u^k - \tau g_k)\|^2, \quad (2.14)$$

where  $g_k$  is defined in (2.4). It is easy to show that the normal equations of (2.14) are of the form

$$\left( D^\top D + \frac{\mu}{\beta\tau} I \right) u = D^\top (w^{k+1} - \lambda^k / \beta) + \frac{\mu}{\beta\tau} (u^k - \tau g_k). \quad (2.15)$$

Under the periodic boundary conditions, the exact solution of (2.15) can be attained by two FFTs. Finally,  $\lambda$  is updated via

$$\lambda^{k+1} = \lambda^k - \beta(w^{k+1} - Du^{k+1}), \quad (2.16)$$

We note that the linearization technique makes the  $u$ -subproblem of (2.12) is solved inexactly. Therefore, we name the above iterative framework as an inexact ADM or IADM, which is summarized below.

ALGORITHM 2 (IADM). *Input*  $f$ ,  $A$  and  $\mu, \beta, \tau > 0$ . *Initialize*  $u^0 = f$  and  $k = 0$ .

**While** “not converged”, **Do**

- 1) Compute  $w^{k+1}$  according to (2.13) for fixed  $\lambda = \lambda^k$  and  $u = u^k$ .
- 2) Compute  $u^{k+1}$  according to (2.15) for fixed  $\lambda = \lambda^k$  and  $w = w^{k+1}$ .
- 3) Update  $\lambda$  via (2.16) and set  $k = k + 1$ .

**End Do**

We have the following convergence results for Algorithm 2.

THEOREM 2.4. *Under Assumption 1, the sequence  $\{(w^k, u^k)\}$  generated by Algorithm 2 from any starting point  $(w^0, u^0)$  converges to a solution of (2.11).*

A closer examination shows that Algorithm 2 is related to the proximal ADM of He et al. [20] for solving monotone variational inequalities. Hence, the global convergence is followed directly, see Appendix B for details.

**3. Numerical experiments.** In this section, we present numerical results to illustrate the feasibility and efficiency of FTVCS and its accelerated variant IAMD. All experiments were accomplished in Matlab 2009a running on a PC (Intel Pentium(R) 4, 1.6 GHz, 1.0GB SDRAM) with Windows XP operating system. As usual, we measure the quality of reconstruction by relative error to the original image  $\bar{u}$ , i.e.,

$$RE = \frac{\|u - \bar{u}\|}{\|\bar{u}\|} \times 100\%.$$

In the following, we first present primary experimental results to show the feasibility of both algorithms, and then compare both algorithms with TwIST — a state-of-the-art algorithm for solving (1.3), to demonstrate their efficiency.

**3.1. Test on FTVCS and IADM.** In the first experiment, we present reconstruction results of both algorithms to illustrate their feasibility for solving (1.3). We used a random matrix with independent identical distributed Gaussian entries as CS encoding matrix and tested the Shepp-Logan phantom image, which has been widely used in simulations for TV models. Due to storage limitations, we tested the image size  $64 \times 64$ . The sample ratio in this test is 30%, which are selected uniformly at random. Besides, we added Gaussian noise of zero mean and standard deviation  $\sigma = 0.001$ . Similar as in FTVd [33], we implemented FTVCS with a continuation scheme on  $\beta$  to speed up convergence. Specifically, we tested the  $\beta$ -sequence  $\{2^4, 2^5, 2^6, 2^7\}$  and used the warm-start technique. In IADM, the value of  $\beta$  is fixed to be 8. In both algorithms, the weighting parameter  $\mu$  was set to be 200. Both algorithms were terminated when the relative change between successive iterates fell below  $10^{-3}$ , i.e.,

$$\|u_k - u_{k-1}\| \leq 10^{-3} \|u_{k-1}\|. \quad (3.1)$$

The original image, the initial guess, and the reconstructed ones by both algorithms are listed in Figure 3.1.

As is shown in Figure 3.1, both algorithms perform favorably and produce faithful recovery results in a few seconds. We note that the per-iteration cost of both algorithms is one shrinkage operation, two matrix-vector multiplications and two FFTs. The results also indicate that the inexact ADM approach described in Algorithm 2 is indeed more efficient than the penalty approach FTVCS described in Algorithm 1 in the sense that better recovery results were obtained in less CUP seconds. To closely examine the convergence behavior of both algorithms, we present in Figure 3.2 the decreasing of objective function values and relative errors

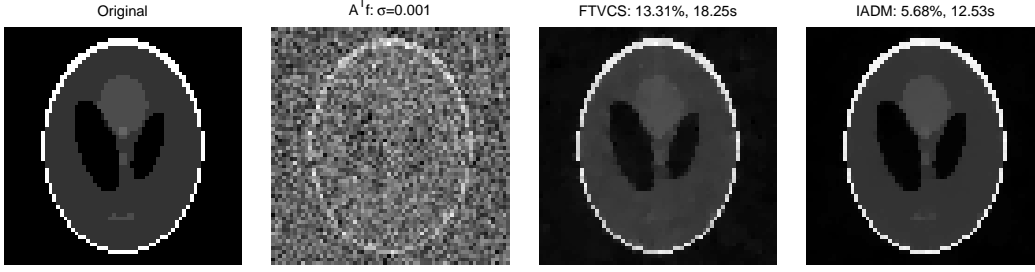


FIG. 3.1. *Reconstruction results of FTVCS and IADM. Original (left,  $64 \times 64$ ); Initial guess (middle left); Recovered by FTVCS (middle right,  $RE=13.31\%$ , CPU time 18.25s); Recovered by IADM (right,  $RE=5.68\%$ , CPU time 12.53s)*

as CPU time proceeded. It is clear from Figure 3.2 that both algorithms generated decreasing sequences of function values. From the right-hand plot, IADM achieved a solution of lower relative error. In both plots, the curves of IADM fall below those of FTVCS throughout the whole iteration process.

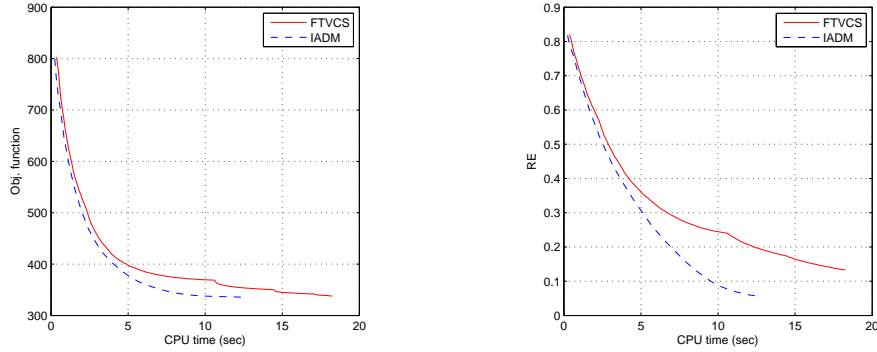


FIG. 3.2. *Convergence behavior of FTVCS and IADM. Left: objective function; Right: relative error. In both plots, the horizontal axes denote CPU time in seconds.*

**3.2. Comparison with TwIST.** In this subsection, we present extensive numerical results to compare IADM with TwIST [3] — a two-step iterative shrinkage/thresholding algorithm for solving a class of optimization problems arising from image restoration, reconstruction and linear inverse problems<sup>1</sup>. Specifically, TwIST is designed to solve

$$\min_u \mathcal{J}(u) + \frac{\mu}{2} \|Au - b\|^2, \quad (3.2)$$

where  $\mathcal{J}(\cdot)$  is a general regularizer, which can be either the  $\ell_1$ -norm or the TV semi-norm, as well as others. In the comparison, we used partial discrete cosine transform (DCT) matrix as CS encoder, i.e., the  $m$  rows of  $A$  were chosen uniformly at random from the  $n \times n$  DCT matrix. Since the DCT matrix is implicitly stored as fast transforms, this enables us to test larger images. We used the default parametric settings for TwIST and terminated it as the relative change in objective function values fell below  $tol = 10^{-3}$ . The parameters in IADM were set as follows:  $\tau = 1.9$  and  $\beta = 2^6$ . To obtain higher quality images, we used more stringent stopping tolerance and terminated IADM when  $\|u_k - u_{k-1}\| \leq 5 \times 10^{-5} \|u_{k-1}\|$  was satisfied.

We first compared IADM with TwIST using the Shepp-Logan phantom benchmark image of size  $128 \times 128$ . We randomly selected 30% DCT coefficients and added Gaussian noise of mean zero and standard

<sup>1</sup>The Matlab code of TwIST can be obtained from <http://www.lx.it.pt/~bioucas/TwIST/TwIST.htm>

TABLE 3.1  
Comparison results of IADM and TwIST with different  $\mu$ .

$\mu$	TwIST				IADM			
	RE	Obj	Iter	Time	RE	Obj	Iter	Time
100	3.79%	736.10	40	26.51	4.80%	721.95	140	12.73
500	3.74%	804.20	55	29.23	3.37%	786.99	219	19.26
600	3.93%	810.80	56	29.02	3.43%	794.20	247	21.47
700	3.97%	814.32	62	33.27	3.56%	798.19	271	23.34
800	3.90%	815.41	69	33.17	3.59%	800.68	297	23.16
900	4.06%	818.17	65	31.92	3.51%	803.09	310	26.97
1000	4.71%	817.52	71	29.72	3.81%	803.00	354	25.86
2000	4.45%	831.86	91	43.41	3.99%	817.08	602	53.22
3000	5.40%	926.04	86	34.55	4.54%	816.49	845	75.22
4000	4.40%	830.97	122	47.23	4.23%	817.81	1066	84.11
5000	4.54%	873.51	146	61.06	4.23%	821.52	1303	116.03
6000	4.51%	858.31	182	77.19	4.33%	822.30	1581	132.27
7000	4.61%	852.42	200	89.05	4.39%	821.55	1742	157.08
8000	4.10%	832.80	288	139.17	4.42%	821.31	1929	177.05
9000	4.22%	830.45	315	135.16	4.59%	819.54	2165	170.30
10000	4.10%	828.36	336	156.98	4.47%	817.46	2405	245.77

deviation 0.001. Table 3.2 reports the detailed results of both algorithms for different values of  $\mu$ , where RE, Obj, Iter and Time represent, respectively, the relative error of the reconstructed image to the original one, the final objective function value, the number of iterations, and the consumed CPU time in seconds.

It can be seen from Table 3.2 that, for both algorithms, the number of iterations becomes larger and larger as  $\mu$  increases, and as a result longer CPU time is consumed. For larger  $\mu$ , the performance of both algorithms deteriorates, while the resulting relative errors were not improved. For  $\mu$  between 500 and 7000, IADM always obtained comparable or higher recovery quality than TwIST. For  $\mu$  between 500 and 1000, IADM is also faster than TwIST. In terms of final function values, IADM obtained slightly smaller ones than those of TwIST.

Besides the Shepp-Logan phantom image, we also tested Cameraman, Lena, Boat, Sailboat, as well as two brain images. In this experiment, we simply set  $\mu = 500$  and keep all other parameters unchanged. The original and the recovered images by TwIST and IADM are given in Figures 3.3 and 3.4, and detailed results including relative errors (RE), CPU time (Time), final objective function values (Obj), and the number of iterations (Iter) are presented in Table 3.2. It can be seen from Table 3.2 that IADM attained comparable or better image quality in less CPU seconds. For each test, IADM consumed more iterations while the CPU time is less because the per-iteration cost of IADM is much less than that of TwIST. Specifically, the per-iteration cost of IADM contains two matrix-vector multiplications and two FFTs, while TwIST needs to solve a TV denoising problem at each iteration. In addition, IADM always attained smaller function values. In summary, the comparison results indicate that IADM performs favorably and can be competitive with the state-of-the-art algorithm TwIST.

**4. Concluding Remarks.** In this paper, we proposed a Fast alternating minimization algorithm for Total Variation image reconstruction from Compressive Sensing data (FTVCS). The per-iteration cost of FTVCS includes a linear time shrinkage operation, two matrix-vector multiplications and two FFTs. To overcome the difficulty caused by large penalty parameter in FTVCS, we have also developed an Inexact Alternating Direction Method (IADM) based on linearization and proximal techniques. Our experimental results indicate that IADM indeed performs better than FTVCS and is comparable with the state-of-the-art algorithm TwIST for solving TV reconstruction models.

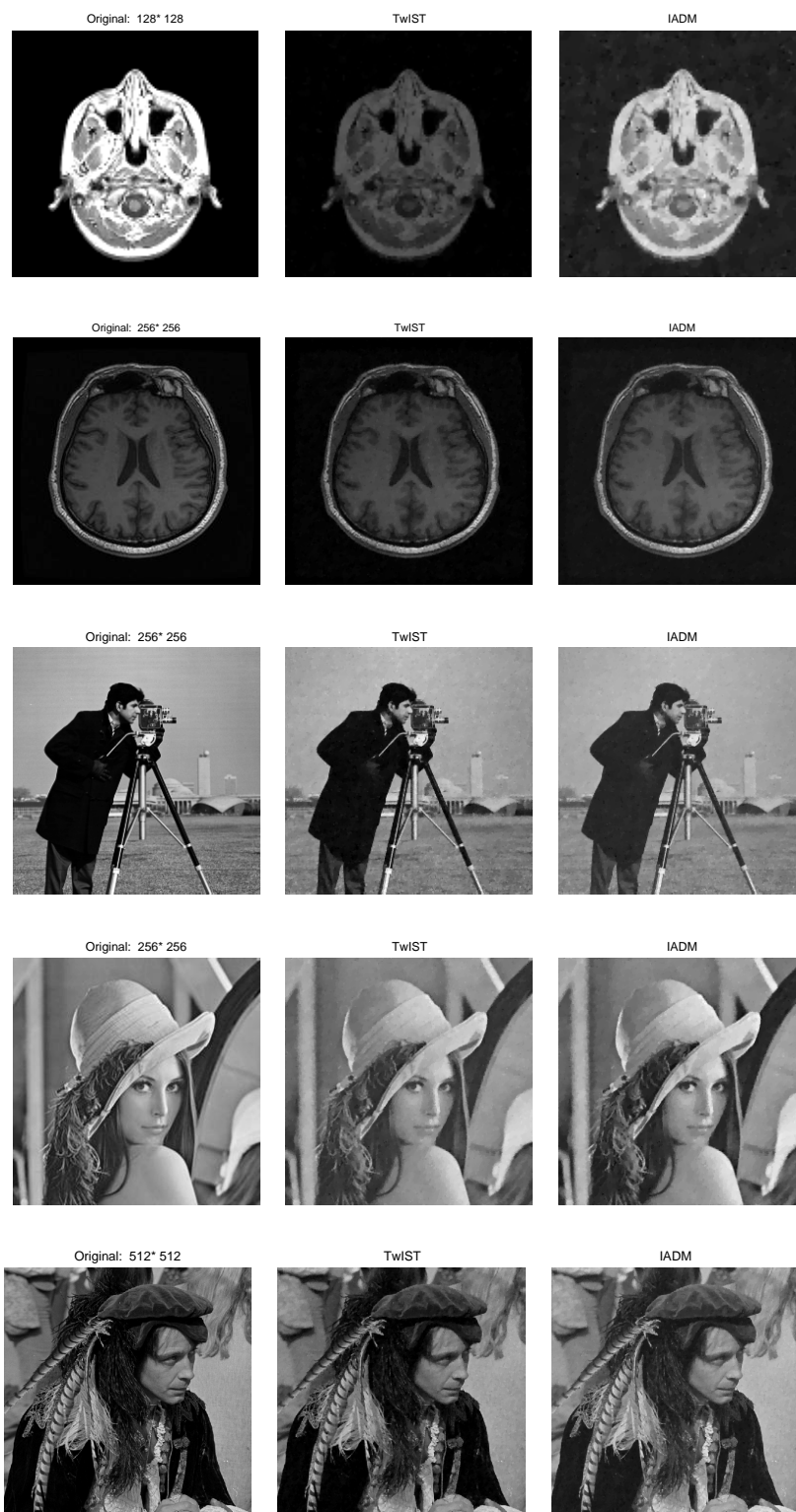


FIG. 3.3. Original and recovered images by IADM and TwiST. From top to bottom: brain 1, brain 2, cameraman, lena and man.

TABLE 3.2  
Comparison results of IADM and TwIST with different images.

Images	Size	TwIST				IADM			
		Iter	RE	Time	Obj	Iter	Re	Time	Obj
brain 1	$128 \times 128$	52	14.01%	34.25s	4.7831e+002	208	13.64%	20.41s	4.5478e+002
brain 2	$256 \times 256$	48	9.59%	90.22s	1.6397e+003	176	9.45%	61.67s	1.5665e+003
cameraman	$256 \times 256$	56	5.71%	122.22s	2.9068e+003	257	5.59%	118.67s	2.7822e+003
lena	$256 \times 256$	53	4.93%	121.06s	2.3656e+003	205	5.01%	92.11s	2.2627e+003
man	$512 \times 512$	59	8.54%	423.38s	1.0617e+004	262	8.57%	400.81s	1.0122e+004
sailboat	$450 \times 450$	57	4.91%	361.58s	7.9220e+003	245	4.98%	260.73s	7.5960e+003
sheppon	$512 \times 512$	42	2.62%	335.36s	4.4496e+003	135	2.09%	217.08s	4.2317e+003
boat	$512 \times 512$	53	4.37%	477.17s	8.7306e+003	200	4.34%	384.61s	8.3312e+003
barbara	$512 \times 512$	56	9.83%	493.20s	1.3469e+004	292	9.80%	550.39s	1.2814e+004

Given the promising performance of IADM and the wide applications of TV models, we believe that it is worthwhile to further accelerate IADM via certain line search strategy. In both FTVCS and IADM, we used a linearization technique and FFTs to obtain a new point. A possible improvement is to solve the  $u$ -subproblem of (2.12) by using certain gradient methods, e.g., gradient descent method with BB steplengths [1] and non-monotone line search. This should be interesting for further investigations.

## REFERENCES

- [1] J. BARZILAI AND J.M. BORWEIN, *Two point step size gradient method*, IMA J. Numer. Anal., 8 (1988), 141-148.
- [2] A. BECK AND M. TEOULLE, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM J. Imaging Sci., 2(2009), 183-202.
- [3] J. BIOUCAS-DIAS, AND M. FIGUEIREDO, *A new TwIst: Two-step iterative thresholding algorithm for image restoration*, IEEE Trans. Image. Process., 16 (2007), 2992-3004.
- [4] J.F. CAI, S. OSHER, AND Z. SHEN, *Linear Bregman iterations for compressed sensing*, Math. Comput., 78(2009), 1515-1536.
- [5] J.F. CAI, S. OSHER, AND Z. SHEN, *Convergence of the linearized Bregman iteration for  $\ell_1$ -norm minimization*, Math. Comput., 78(2009), 2127-2136.
- [6] E. CANDÈS, J. ROMBERG, AND T. TAO, *Stable signal recovery from incomplete and inaccurate information*, Communications on Pure and Applied Math., 59 (2005), 1207-1233.
- [7] E. CANDÈS, J. ROMBERG, AND T. TAO, *Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Inform. Theory, 52 (2006), 489-509.
- [8] P.L. COMBETTES AND J.C. PESQUET, *Proximal thresholding algorithm for minimization over orthonormal bases*, SIAM J. Optim., 18 (2008), 1351-1376.
- [9] T.F. CHAN AND K. CHEN, *An optimization-based multilevel algorithm for total variation image denoising*, Multiscale Model. Simul., 5 (2006), 615-645.
- [10] T.F. CHAN, G.H. GOLUB, AND P. MULET, *A nonlinear diffusivity fixed point method in total variation based image restoration*, SIAM J. Sci. Comput., 20 (1999), 1964-1977.
- [11] D. DONOHO, *Compressed sensing*, IEEE Trans. Inform. Theory, 52 (2006), 1289-1306.
- [12] D. DONOHO, *For most large underdetermined systems of linear equations, the minimal  $\ell_1$ -norm solution is also the sparsest solution*, Communications on Pure and Applied Mathematics, 59 (2006), 907-934.
- [13] M. ELAD, *Why simple shrinkage is still relevant for redundant representations?*, IEEE Transactions on Information Theory, 52 (2006), pp. 5559-5569.
- [14] M. ELAD, B. MATALON, AND M. ZIBULEVSKY, *Image denoising with shrinkage and redundant representations*, in Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, New York, 2006.
- [15] M. FIGUEREDO, R. NOWAK, AND S.J. WRIGHT, *Gradient projection for sparse reconstruction, Application to compressed sensing and other inverse problems*, IEEE Journal of Selected Topics in Signal Processing: Special Issue on Convex Optimization Methods for Signal Processing, 1 (2007), 586-598.
- [16] D. GABAY AND B. MERCIER, *A dual algorithm for the solution of nonlinear variational problems via finite element approximation*, Computer Math. Appl., 2 (1976), 17-40.

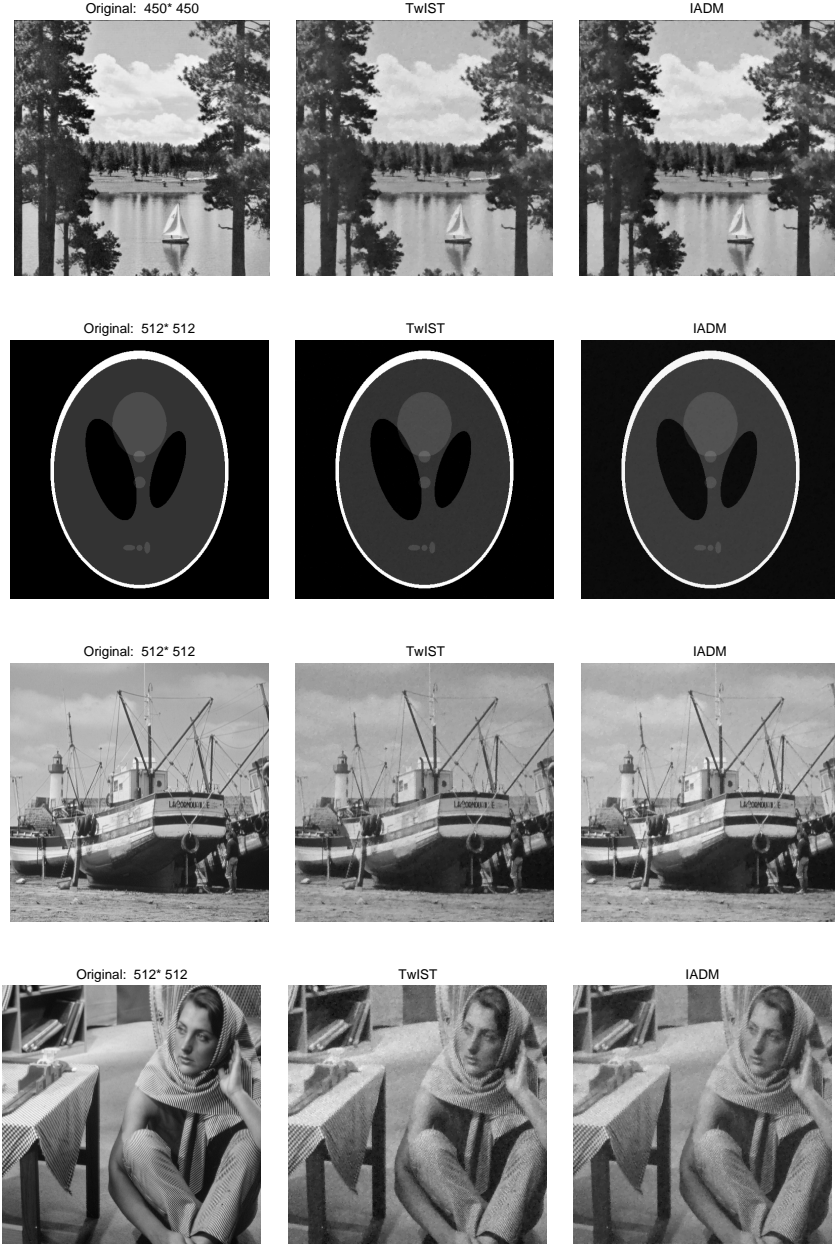


FIG. 3.4. Original and recovered images by IADM and TwIST. From top to bottom: sailboat, sheppon, boat and barbara.

- [17] R. GLOWINSKI AND P. LE TALLEC, *Augmented Lagrangian and operator-splitting methods*, in: Nonlinear Mechanics, SIAM Studies in Applied Mathematics, Philadelphia, PA, 1989.
- [18] R. GLOWINSKI, *Numerical methods for nonlinear variational problems*, Springer-Verlat, New York, 1984.
- [19] E.T. HALE, W. YIN, AND Y. ZHANG, *A fixed-point continuation method for l1-regularized minimization with applications to compressed sensing*, SIAM J. Optim., 19 (2008), 1107-1130.
- [20] B. HE, L.Z. LIAO, D. HAN, AND H. YANG, *A new inexact alteratin directions method for monotone variational inequalities*, Math. Program., 92 (2002), 103-118.
- [21] Y. HUANG, M.K. NG, AND Y.W. WEN, *A fast total variation minimization method for image restoration*, Multiscale Model. Simul., 7 (2008), 775-795.
- [22] M. LUSTIG, D. DONOHO, AND J. M. PAULY, *Sparse MRI: the application of compressed sensing for rapid MR Imaging*,

- Magnetic Resonance in Medicine, 58 (2007), 1182-1195.
- [23] M. NG, L. QI, Y. YANG, AND Y. HUANG, *On semismooth Newton's methods for total variation minimization*, J. Math. Imaging Vision, 27 (2007), 265-276.
  - [24] S. OSHER, M. BURGER, D. GOLDFARB, J. XU, AND W. YIN, *An iterated regularization method fro total variation-based image restoration*, Multiscale Model. Simul., 4 (2005), 460-489.
  - [25] S. OSHER, Y. MAO, B. DONG, AND W. YIN, *Fast linearized Bregman iteration for compressed sensing and sparse denoising*, TR08-07, CAAM, Rice University.
  - [26] N. PARAGIOS, C. CHEN, AND O. FAUGERAS, *Handbook of mathematical models in computer vision*, Springer, New York, 2006.
  - [27] L. RUDIN, S. OSHER, AND E. FATEMI, *Nonlinear total variation based noise removal algorithms*, Phys. D, 60 (1992), 259-268.
  - [28] J. L. STARCK, M. NGUYEN, AND F. MURTAGH, *Wavelets and curvelets for image deconvolution: a combined approach*, Signal Processing, 83 (2003), pp. 2279-2283.
  - [29] A. TIKHONOV AND V. ARSEININ, *Solution of Ill-Posed Problems*, Winston, Washington, DC, 1977.
  - [30] E. VAN DEN BERG AND M.P. FRIEDLANDER, *Probing the pareto frontier for basis pursuit solutions*, SIAM J. Sci. Comput., 31 (2008), 890-912.
  - [31] C. R. VOGEL AND M. E. OMAN, *Iterative methods for total variation denoising*, SIAM J. Sci. Comput., 17 (1996), pp.227-238.
  - [32] —, *A fast, robust total variation based reconstruction of noisy, blurred images*, IEEE Trans. Image Process., 7 (1998), pp. 813-824.
  - [33] Y. WANG, J. YANG, W. YIN, AND Y. ZHANG, *A new alternating minimization algorithm for total variation image reconstruction*, SIAM J. Imaging Sci., 1 (2008), 248-272.
  - [34] Y.W. WEN, M.K. NG, AND W.K. CHING, *Iterative algorithms based on decoupling of deblurring and denoising for image restoration*, SIAM J. Sci. Comput., 30 (2008), 2655-2674.
  - [35] Z. WEN, W. YIN, D. GOLDFARB, AND Y. ZHANG, *A fast algorithm for sparse resconstruction based on shrinkage: subspace optimization and continuations*, TR09-01, CAAM, Rice University.
  - [36] J. YANG, W. YIN, Y. ZHANG, AND Y. WANG, *A fast algorithm for edge-preserving variational multichannel image restoration*, SIAM J. Imaging Sci., 2 (2009), 569-592.
  - [37] J. YANG AND Y. ZHANG, *Alternating direction algorithms for  $\ell_1$ -problems in compressive sensing*, TR09-37, CAAM, Rice University.
  - [38] J. YANG, Y. ZHANG, AND W. YIN, *An efficient TVL1 algorithm for deblurring multichannel images corrupted by impulsive noise*, SIAM J. Sci. Comput., 31 (2009), 2842-2865.
  - [39] J. YANG, Y. ZHANG, AND W. YIN, *A fast TVL1-L2 minimization algorithm for signal reconstruction from partial Fourier data*, IEEE J. Special Topics Signal Processing, to appear.
  - [40] W. YIN, S. OSHER, D. GOLDFARB, AND J. DARBON, *Bregman Iterative Algorithms for  $l_1$ -Minimization with Applications to compressed sensing*, SIAM J. Imaging Sci., 1 (2008), 143-168.
  - [41] Y. ZHANG, *On the theory of compressed sensing via  $\ell_1$ -minimization: simple derivations and extensions*, TR08-11, CAAM, Rice University.

## Appendix A. Proof of Theorems 2.1, 2.2 and 2.3.

The purpose of this appendix is to establish convergence properties of Algorithm 1 for a fixed  $\beta > 0$ . For convenience, we define some notation. For fixed  $\beta > 0$ , the 2-dimensional (2D) shrinkage operator  $s : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  is defined as

$$s(\alpha) \triangleq \alpha - \mathcal{P}_{\mathcal{B}}(\alpha) = \max \{ \|\alpha\| - 1/\beta, 0 \} \frac{\alpha}{\|\alpha\|}, \quad (\text{A.1})$$

where  $\mathcal{P}_{\mathcal{B}}(\cdot) : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  is the projection onto the closed disc  $\mathcal{B} \triangleq \{ \alpha \in \mathbb{R}^2 : \|\alpha\| \leq 1/\beta \}$ , and the convention  $0 \cdot (0/0) = 0$  is followed. For vectors  $u, v \in \mathbb{R}^N$ ,  $N \geq 1$ , we define  $S(u; v) : \mathbb{R}^{2N} \rightarrow \mathbb{R}^{2N}$  by

$$S(u; v) = (s(\alpha_1); \dots; s(\alpha_N)), \quad \text{where } \alpha_i = \begin{pmatrix} u_i \\ v_i \end{pmatrix},$$

i.e.,  $S$  applies 2D shrinkage to each pair  $(u_i; v_i)$ , for  $i = 1, 2, \dots, N$ . From the definition of  $s(\cdot)$ , it is easy to

see that (2.1) or (2.2) can be rewritten as  $\mathbf{w}_i = s(D_i u)$ . The following result shows that the operator  $s$  is non-expansive.

LEMMA A.1 ([33]). *For any  $a, b \in \mathbb{R}^2$ , it holds that*

$$\|s(a) - s(b)\|^2 \leq \|a - b\|^2 - \|\mathcal{P}_{\mathcal{B}}(a) - \mathcal{P}_{\mathcal{B}}(b)\|^2,$$

Furthermore, if  $\|s(a) - s(b)\| = \|a - b\|$ , then  $s(a) - s(b) = a - b$ .

Since the objective function in (1.6) is convex, bounded below, and coercive (i.e., its goes to infinity as  $\|(w, u)\| \rightarrow \infty$ ), it has at least one minimizer  $(w^*, u^*)$  that cannot be decreased by the alternating minimization scheme (2.2)-(2.3) and thus must satisfy

$$\begin{cases} w^* = S(D^{(1)}u^*; D^{(2)}u^*) \ (\triangleq S(Du^*)), \\ (D^\top D + \frac{\mu}{\beta} A^\top A)u^* = D^\top w^* + \frac{\mu}{\beta} A^\top f. \end{cases} \quad (\text{A.2})$$

By using the shrinkage operator, we can rewrite the iteration of Algorithm 1 as

$$\begin{cases} w^{k+1} = S(D^{(1)}u^k; D^{(2)}u^k) \ (\triangleq S(Du^k)), \\ (D^\top D + \frac{\mu}{\beta\tau} I)u^{k+1} = D^\top w^{k+1} + \frac{\mu}{\beta\tau}(u^k - \tau g_k). \end{cases} \quad (\text{A.3})$$

In the following, we show that (A.3) converges to (A.2). The following matrices will be used in our analysis:

$$M = D^\top D + \frac{\mu}{\beta} A^\top A, \quad H = D^\top D + \frac{\mu}{\beta\tau} I, \quad \text{and} \quad T = I - \tau A^\top A.$$

Assumption 1 ensures the non-singularity of  $M$ , while  $H^{-1}$  is always well defined under the circumstance. Simple manipulation shows that  $H - M = \eta^2 T$ , where  $\eta \triangleq \sqrt{\frac{\mu}{\beta\tau}}$ . With these definitions, (A.2) and (A.3) can be, respectively, simplified as

$$\begin{cases} w^* = S(Du^*), \\ v^* = \eta T u^* + \eta \tau A^\top f, \\ H u^* = D^\top w^* + \eta v^*, \end{cases} \quad \text{and} \quad \begin{cases} w^{k+1} = S(Du^k), \\ v^{k+1} = \eta T u^k + \eta \tau A^\top f, \\ H u^{k+1} = D^\top w^{k+1} + \eta v^{k+1}. \end{cases}$$

To further simplify the above equations, we define

$$h(w; v) = D H^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix}^\top \begin{pmatrix} w \\ v \end{pmatrix} \quad \text{and} \quad p(w; v) = \eta T H^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix}^\top \begin{pmatrix} w \\ v \end{pmatrix}.$$

Hence, the solution and iteration systems can be, respectively, rewritten as

$$\begin{cases} w^* = S \circ h(w^*; v^*), \\ v^* = p(w^*; v^*) + \eta \tau A^\top f, \\ H u^* = D^\top w^* + \eta v^*, \end{cases} \quad (\text{A.4})$$

and

$$\begin{cases} w^{k+1} = S \circ h(w^k; v^k), \\ v^{k+1} = p(w^k; v^k) + \eta \tau A^\top f, \\ H u^{k+1} = D^\top w^{k+1} + \eta v^{k+1}, \end{cases} \quad (\text{A.5})$$

where “ $\circ$ ” denotes operator composition. Furthermore, we define

$$q(w; v) = \begin{pmatrix} S \circ h(w; v) \\ p(w; v) \end{pmatrix} + \begin{pmatrix} 0 \\ \eta \tau A^\top f \end{pmatrix}.$$

Then (A.4) and (A.5) become

$$\begin{cases} (w^*; v^*) = q(w^*; v^*) \\ Hu^* = D^\top w^* + \eta v^*, \end{cases} \quad \text{and} \quad \begin{cases} (w^{k+1}; v^{k+1}) = q(w^k; v^k) \\ Hu^{k+1} = D^\top w^{k+1} + \eta v^{k+1}, \end{cases}$$

LEMMA A.2.  $q(w; v)$  is non-expansive.

*Proof.* Given  $(w^1; v^1)$  and  $(w^2; v^2)$ , it holds that

$$\begin{aligned} \|q(w^1; v^1) - q(w^2; v^2)\|^2 &= \left\| \begin{pmatrix} S \circ h(w^1; v^1) - S \circ h(w^2; v^2) \\ p(w^1; v^1) - p(w^2; v^2) \end{pmatrix} \right\|^2 \\ &\leq \|h(w^1; v^1) - h(w^2; v^2)\|^2 + \|p(w^1; v^1) - p(w^2; v^2)\|^2 \\ &= \left\| R \begin{pmatrix} w^1 - w^2 \\ v^1 - v^2 \end{pmatrix} \right\|^2, \end{aligned}$$

where “ $\leq$ ” comes from the non-expansive of  $s(\cdot)$  and

$$R \triangleq \begin{pmatrix} D \\ \eta T \end{pmatrix} H^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix}^\top.$$

It is easy to verify that

$$\begin{aligned} R^\top R &= \begin{pmatrix} D \\ \eta I \end{pmatrix} H^{-1} (D^\top D + \eta^2 T^2) H^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix}^\top \\ &= \begin{pmatrix} D \\ \eta I \end{pmatrix} H^{-1} \left( H - \frac{\mu}{\beta} (2A^\top A - \tau(A^\top A)^2) \right) H^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix}^\top \\ &= \begin{pmatrix} D \\ \eta I \end{pmatrix} H^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix}^\top - \frac{\mu}{\beta} \begin{pmatrix} D \\ \eta I \end{pmatrix} H^{-1} (2A^\top A - \tau(A^\top A)^2) H^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix}^\top. \end{aligned}$$

Recall that we require  $0 < \tau < 2/\lambda_{\max}(A^\top A)$ , which ensures the positive semi-definiteness of  $A^\top A - \tau(A^\top A)^2$ . Therefore,

$$\|q(w^1; v^1) - q(w^2; v^2)\|^2 \leq \begin{pmatrix} w^1 - w^2 \\ v^1 - v^2 \end{pmatrix}^\top \begin{pmatrix} D \\ \eta I \end{pmatrix} H^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix}^\top \begin{pmatrix} w^1 - w^2 \\ v^1 - v^2 \end{pmatrix} \leq \left\| \begin{pmatrix} w^1 - w^2 \\ v^1 - v^2 \end{pmatrix} \right\|^2, \quad (\text{A.6})$$

which shows that  $q(w; v)$  is non-expansive.  $\square$

LEMMA A.3. Equality holds in (A.6) if and only if

$$q(w^1; v^1) - q(w^2; v^2) = \begin{pmatrix} w^1 - w^2 \\ v^1 - v^2 \end{pmatrix}.$$

*Proof.* We note that in the proof of Lemma A.2 there exist three “ $\leq$ ”. Thus, equality holds in (A.6) only when all the three inequalities become “ $=$ ”. For simplicity, we let  $dw = w^1 - w^2$  and  $dv = v^1 - v^2$ .

1. The first “ $\leq$ ” becomes “ $=$ ” if and only if

$$S \circ h(w^1; v^1) - S \circ h(w^2; v^2) = h(w^1; v^1) - h(w^2; v^2) = DH^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix}^\top \begin{pmatrix} dw \\ dv \end{pmatrix}.$$

2. The second “ $\leq$ ” becomes “ $=$ ” if and only if

$$\begin{pmatrix} dw \\ dv \end{pmatrix}^\top \begin{pmatrix} D \\ \eta I \end{pmatrix} H^{-1} (2A^\top A - \tau(A^\top A)^2) H^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix}^\top \begin{pmatrix} dw \\ dv \end{pmatrix} = 0.$$

3. Let  $U$  be orthonormal and

$$\begin{pmatrix} D \\ \eta I \end{pmatrix}^\top H^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix} = U^\top \Lambda U$$

be its eigenvalue decomposition. The third “ $\leq$ ” becomes “ $=$ ” if and only if

$$\sum_i \lambda_i \left( U \begin{pmatrix} dw \\ dv \end{pmatrix} \right)_i^2 = \sum_i \left( U \begin{pmatrix} dw \\ dv \end{pmatrix} \right)_i^2.$$

Since  $0 \leq \lambda_i \leq 1$ , the above equality holds only when

$$\lambda_i \left( U \begin{pmatrix} dw \\ dv \end{pmatrix} \right)_i^2 = \left( U \begin{pmatrix} dw \\ dv \end{pmatrix} \right)_i^2, \quad \forall i.$$

Therefore,

$$\Lambda U \begin{pmatrix} dw \\ dv \end{pmatrix} = U \begin{pmatrix} dw \\ dv \end{pmatrix},$$

and thus

$$\begin{pmatrix} D \\ \eta I \end{pmatrix} H^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix}^\top \begin{pmatrix} dw \\ dv \end{pmatrix} = \begin{pmatrix} dw \\ dv \end{pmatrix}. \quad (\text{A.7})$$

From 1 and (A.7), we have

$$S \circ h(w^1; v^1) - S \circ h(w^2; v^2) = D H^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix}^\top \begin{pmatrix} dw \\ dv \end{pmatrix} = dw. \quad (\text{A.8})$$

From (A.7), the equality in 2 is equivalent to

$$dv^\top (2A^\top A - \tau(A^\top A)^2) dv = 0.$$

Let  $U^\top \Lambda U = A^\top A$  be the eigenvalue decomposition of  $A^\top A$ . The above equation is equivalent to

$$dv^\top U^\top (2\Lambda - \tau\Lambda^2) U dv = 0 \quad \text{or} \quad \sum_i (2\lambda_i - \tau\lambda_i^2) (U dv)_i^2 = 0.$$

Since  $2\lambda_i - \tau\lambda_i^2 \geq 0$ , we have  $(2\lambda_i - \tau\lambda_i^2) (U dv)_i^2 = 0, \forall i$ . If  $\lambda_i \neq 0$ , then from the choice of  $\tau$  we have  $2\lambda_i - \tau\lambda_i^2 > 0$ , and thus  $(U dv)_i = 0$ . Therefore,  $\Lambda U dv = 0$  and  $A^\top A dv = U^\top \Lambda U dv = 0$ . Sum the above discussions up, we have

$$\begin{aligned} q(w^1; v^1) - q(w^2; v^2) &= \begin{pmatrix} S \circ h(w^1; v^1) - S \circ h(w^2; v^2) \\ p(w^1; v^1) - p(w^2; v^2) \end{pmatrix} \\ &= \begin{pmatrix} dw \\ T \cdot dv \end{pmatrix} = \begin{pmatrix} dw \\ dv - A^\top A dv \end{pmatrix} = \begin{pmatrix} w^1 - w^2 \\ v^1 - v^2 \end{pmatrix}, \end{aligned}$$

where the first equality is from the definition of  $q(\cdot; \cdot)$ ; the second one is from (A.8), the definition of  $p$  and (A.7); the third one is from the definition of  $T$ ; and the final one is from  $A^\top dv = 0$ . This completes the proof.  $\square$

COROLLARY A.4. *Suppose  $(w^*; v^*)$  is a fixed point of  $q$ , i.e.,  $(w^*; v^*) = q(w^*; v^*)$ . Then for any  $(w; v)$  it holds*

$$\|q(w; v) - q(w^*; v^*)\| < \|(w; v) - (w^*; v^*)\|$$

unless  $(w; v)$  is also a fixed point of  $q(\cdot; \cdot)$ .

Based on the above lemmas, now we are ready to give the proofs of Theorems 2.1, 2.2 and 2.3.

*Proof.* (Theorem 2.2) First, the convergence of  $(w^k, u^k)$  to  $(w^*, u^*)$  can be established using exactly the same arguments as in Theorem 3.4 in [33]. The convergence of  $u^k$  to  $u^*$  follows from the convergence of  $w^k$  to  $w^*$  and  $v^k$  to  $v^*$ . Therefore, we omit the details.  $\square$

For any  $i$ , we let  $h_i(w; v) = D_i H^{-1}(D^\top w + \eta v) : \mathbb{R}^{3n^2} \rightarrow \mathbb{R}^2$ ,  $E = \{1, \dots, n^2\}/L$ , where we recall that  $L = \{i : \|D_i u^*\| = \|h_i(w^*, v^*)\| \leq 1/\beta\}$ , and

$$w = \min\{1/\beta - \|D_i u^*\| : i \in L\} > 0. \quad (\text{A.9})$$

*Proof.* (THEOREM 2.2) From the non-expansive of  $s(\cdot)$ , for each  $i$ , it holds

$$\|\mathbf{w}_i^{k+1} - \mathbf{w}_i^*\| = \|s \circ h_i(w^k; v^k) - s \circ h_i(w^*; v^*)\| \leq \|h_i(w^k; v^k) - h_i(w^*; v^*)\|. \quad (\text{A.10})$$

Suppose that at iteration  $k$  there exist at least one index  $i \in L$  such that  $\mathbf{w}_i^{k+1} = s \circ h_i(w^k; v^k) \neq 0$ . Then  $\|h_i(w^*; v^*)\| \leq 1/\beta$ ,  $\|h_i(w^k; v^k)\| > 1/\beta$ , and  $\mathbf{w}_i^* = s \circ h_i(w^*; v^*) = 0$ . Therefore,

$$\begin{aligned} \|\mathbf{w}_i^{k+1} - \mathbf{w}_i^*\| &= \|s \circ h_i(w^k; v^k)\|^2 = (\|h_i(w^k; v^k)\| - 1/\beta)^2 \\ &\leq [\|h_i(w^k; v^k) - h_i(w^*; v^*)\| - (1/\beta - \|h_i(w^*; v^*)\|)]^2 \\ &\leq \|h_i(w^k; v^k) - h_i(w^*; v^*)\|^2 - (1/\beta - \|h_i(w^*; v^*)\|)^2 \\ &\leq \|h_i(w^k; v^k) - h_i(w^*; v^*)\|^2 - \omega^2, \end{aligned} \quad (\text{A.11})$$

where the first “ $\leq$ ” is the triangular inequality, the second one follows from the fact that  $\|h_i(w^k; v^k) - h_i(w^*; v^*)\| \geq 1/\beta - \|h_i(w^*; v^*)\| > 0$ , and the last one used the definition of  $\omega$  in (A.9). Combining with (A.9) and (A.11), we obtain

$$\begin{aligned} \left\| \begin{matrix} w^{k+1} - w^* \\ v^{k+1} - v^* \end{matrix} \right\|^2 &= \sum_i \|\mathbf{w}_i^{k+1} - \mathbf{w}_i^*\|^2 + \|v^{k+1} - v^*\|^2 \\ &\leq \sum_i \|h_i(w^k; v^k) - h_i(w^*; v^*)\|^2 - \omega^2 + \|v^{k+1} - v^*\|^2 \\ &= \|h_i(w^k; v^k) - h_i(w^*; v^*)\|^2 - \omega^2 + \|p(w^k; v^k) - p(w^*; v^*)\| \\ &\leq \left\| \begin{matrix} w^k - w^* \\ v^k - v^* \end{matrix} \right\|^2 - \omega^2, \end{aligned}$$

where the second “ $\leq$ ” comes from the non-expansiveness of

$$\phi(w; v) = \begin{pmatrix} h(w; v) \\ p(w; v) \end{pmatrix},$$

which can be easily derived. Therefore, the number of iterations  $k$  with  $\mathbf{w}_i^{k+1} \neq 0$  does not exceed

$$\frac{1}{\omega^2} \left\| \begin{pmatrix} w^0 - w^* \\ v^0 - v^* \end{pmatrix} \right\|^2.$$

This completes the proof of Theorem 2.2.  $\square$

*Proof.* (Theorem 2.3) From the iteration formulae for  $u$  and  $(w; v)$ , there holds

$$u^{k+1} - u^* = H^{-1} \begin{pmatrix} D \\ \eta I \end{pmatrix}^\top \begin{pmatrix} w^{k+1} - w^* \\ v^{k+1} - v^* \end{pmatrix},$$

and

$$\left\| \begin{pmatrix} w^{k+1} - w^* \\ v^{k+1} - v^* \end{pmatrix} \right\|^2 = \|q(w^k; v^k) - q(w^*; v^*)\|^2 \leq \left\| \begin{pmatrix} D(u^k - u^*) \\ \eta T(u^k - u^*) \end{pmatrix} \right\|^2 = \left\| R \begin{pmatrix} w^k - w^* \\ v^k - v^* \end{pmatrix} \right\|^2.$$

Considering the finite convergence of  $\mathbf{w}_i$ ,  $i \in L$ , we have

$$\left\| \begin{pmatrix} w_E^{k+1} - w_E^* \\ v^{k+1} - v^* \end{pmatrix} \right\|^2 \leq \rho((R^\top R)_{EE}) \left\| \begin{pmatrix} w_E^k - w_E^* \\ v^k - v^* \end{pmatrix} \right\|^2,$$

where  $(R^\top R)_{EE}$  is a sub-matrix of  $R^\top R \in \mathbb{R}^{3n^2 \times 3n^2}$  formed by throwing away certain rows (with indexes  $\cup_{i \in L} \{i, i + n^2\}$ ) and corresponding columns. Multiplying  $(D; \eta T)$  to the recursion of  $u^{k+1} - u^*$ , we get

$$\begin{aligned} \|u^{k+1} - u^*\|_{D^\top D + \eta^2 T^2}^2 &= \begin{pmatrix} w^{k+1} - w^* \\ v^{k+1} - v^* \end{pmatrix}^\top R^\top R \begin{pmatrix} w^{k+1} - w^* \\ v^{k+1} - v^* \end{pmatrix} \\ &= \rho((R^\top R)_{EE}) \left\| \begin{pmatrix} w^{k+1} - w^* \\ v^{k+1} - v^* \end{pmatrix} \right\|^2 \leq \rho((R^\top R)_{EE}) \|u^k - u^*\|_{D^\top D + \eta^2 T^2}^2, \end{aligned}$$

which shows that  $\{u^k\}$  converges q-linearly.  $\square$

**Appendix B. Convergence of Algorithm 2.** In this section, we clarify the relationship between Algorithm 2 and the proximal ADM approach proposed in [20]. The convergence of Algorithm 2 follows directly.

We briefly review the proximal ADM approach in [20] for structured variational inequality (SVI) problems. Let  $M$  and  $N$  be, respectively,  $l \times n$  and  $l \times m$  matrixes,  $\mathcal{X} \subset \mathbb{R}^n$  and  $\mathcal{Y} \subset \mathbb{R}^m$  be nonempty closed convex sets, and  $f, g$  be given monotone operators. The SVI problem is to find  $u^* \in \Omega$  such that

$$(u - u^*)^\top F(u^*) \geq 0, \quad \forall u \in \Omega, \quad (\text{B.1})$$

where  $\Omega = \{(x, y) : x \in \mathcal{X}, y \in \mathcal{Y}, Mx + Ny = 0\}$ ,

$$u = \begin{pmatrix} x \\ y \end{pmatrix}, \quad \text{and} \quad F(u) = \begin{pmatrix} f(x) \\ g(y) \end{pmatrix}.$$

Given  $(x^k, y^k, \lambda^k)$ , the proximal ADM proposed in [20] iterates as follows

1. Compute  $x^{k+1} \in \mathcal{X}$  via solving

$$(x' - x)^\top \{f(x) - M^\top [\lambda^k - h(Mx + Ny^k)] + R_k(x - x^k)\} \geq 0, \quad \forall x' \in \mathcal{X}. \quad (\text{B.2})$$

2. Compute  $y^{k+1} \in \mathcal{Y}$  via solving

$$(y' - y)^\top \{g(y) - N^\top [\lambda^k - h(Mx^{k+1} + Ny)] + S_k(y - y^k)\} \geq 0, \quad \forall y' \in \mathcal{Y}. \quad (\text{B.3})$$

3. Update  $\lambda^{k+1}$  via

$$\lambda^{k+1} = \lambda^k - h(Mx^{k+1} + Ny^{k+1}),$$

where  $h > 0$  is a parameter,  $R_k$  and  $S_k$  are symmetric positive semidefinite matrices. Under mild assumptions, global convergence of this proximal ADM approach was established in [20]. Simple manipulation shows that Algorithm 2 is a special case of the proximal ADM approach described above by setting  $R_k \equiv 0$  in (B.2), i.e., the  $w$ -subproblems are solved exactly in Algorithm 2, and  $S_k = \frac{1}{\tau}I - A^\top A$  in (B.3). Hence, the global convergence of Algorithm 2 to a solution of (2.11) follows from [20, Theorem 4].