An Efficient Augmented Lagrangian Method with Applications to Total Variation Minimization

Chengbo Li[1], Wotao Yin[1], Hong Jiang[2], and Yin Zhang[1]

[1]Department of Computational and Applied Mathematics, Rice University, 6100 Main, Houston, TX 77005 (email: chengbo.li, wotao.yin, and yzhang@rice.edu).
[2]Bell Laboratories, Alcatel-Lucent, 700 Mountain Avenue, Murray Hill, NJ 07974 (email: hong.jiang@alcatel-lucent.com).

August 13, 2012

Abstract

Based on the classic augmented Lagrangian multiplier method, we propose, analyze and test an algorithm for solving a class of equality-constrained non-smooth optimization problems (chiefly but not necessarily convex programs) with a particular structure. The algorithm effectively combines an alternating direction technique with a nonmonotone line search to minimize the augmented Lagrangian function at each iteration. We establish convergence for this algorithm, and apply it to solving problems in image reconstruction with total variation regularization. We present numerical results showing that the resulting solver, called TVAL3, is competitive with, and often outperforms, other state-of-the-art solvers in the field.

1 Introduction

1.1 A Class of Non-Smooth Minimization Problems

In this paper, we consider solving a class of equality-constrained minimization problems of the form

$$\min_{x,y} f(x,y), \quad \text{s.t. } h(x,y) = 0, \tag{1}$$

where $x \in \mathbb{R}^{n_1}$, $y \in \mathbb{R}^{n_2}$, the vector-valued function $h : \mathbb{R}^{n_1+n_2} \to \mathbb{R}^m$ ($m < n_1+n_2$) is differentiable with respect to both x and y, but the function f may or may not be differentiable with respect to y. In addition, we will later impose a special structure on such problems under consideration. For solving this class of problems, we will propose and study an algorithm in the framework of the classic augmented Lagrangian multiplier (ALM) method, first suggested by Hestenes [20] and Powell

[28]. In the ALM framework, one obtains the k-th iterate (x^k, y^k) by minimizing the augmented Lagrangian function

$$\mathcal{L}_A(x, y, \lambda; \mu) = f(x, y) - \lambda^T h(x, y) + \frac{\mu}{2} h(x, y)^T h(x, y), \quad \mu > 0, \tag{2}$$

jointly with respect to both x and y for a given multiplier estimate $\lambda = \lambda_{k-1}$, then updates the multiplier estimate by the formula $\lambda_k = \lambda_{k-1} - \mu h(x_k, y_k)$. In principle, the positive parameter μ in the augmented Lagrangian function, known as the penalty parameter, can also be altered from iteration to iteration.

It is evident that the iteration-complexity of an ALM algorithm depends almost entirely on how the augmented Lagrangian function is minimized jointly with respect to both x and y. In order to solve such subproblems efficiently, one should utilize useful structures existing in the augmented Lagrangian function. Therefore, we concentrate on solving unconstrained minimization problems of the form

$$\min_{x,y} \phi(x, y), \tag{3}$$

where ϕ is differentiable with respect to the block variable x but not necessarily to y.

In this paper, we assume that the objective function $\phi(x, y)$ in (3) has the following qualitative structure, called a "*structure of uneven complexity*"; that is, in some measurement,

the complexity of minimizing $\phi(x, y)$ with respect to y is much lower than that of minimizing with respect to x.

For example, for $x, y \in \mathbb{R}^n$ a function $\phi(x, y)$ has a structure of uneven complexity if the complexity of minimizing $\phi(x, y)$ with respect to y is $O(n)$ while that of minimizing with respect to x is $O(n^2)$ or higher.

With little loss of generality, we assume that

$$y(x) = \arg\min_y \phi(x, y) \tag{4}$$

exists and is unique for each x in a region of interest. Consequently, problem (3) can be reduced to an unconstrained minimization problem with respect to x only; that is,

$$\min_x \psi(x) \triangleq \phi(x, y(x)), \tag{5}$$

where $\psi(x)$ is generally non-differentiable, but $\partial_1 \phi$, the partial derivative of $\phi(x, y)$ with respect to x, is assumed to exist.

To solve the unconstrained minimization problem (5), we will construct a nonmonotone line search algorithm which is a modification of the one [37] proposed by Zhang and Hager. The modification is necessary since, to the best of our knowledge, existing nonmonotone line search algorithms require that objective functions be differentiable, or at least have their sub-differentials

available. In our case, however, we do not require the availability of the sub-differential of $\psi(x)$. Instead, we only use $\partial_1\phi$ — the partial derivative of $\phi(x, y)$ with respect to x.

Problem (1) has a wide range of applications. For example, a large number of problems in physics, mechanics, economics and mathematics can be reduced to variational problems of the form:

$$\min_x f(x) = f_1(x) + f_2(Bx),$$

where both f_1 and f_2 are convex, proper, lower semicontinuous functions, and B is a linear operator. We consider the case that f_1 is differentiable but f_2 is not. In the early 1980s, Glowinski *et al.* studied this type of problems in depth using the ALM and operator-splitting methods [14, 16], which also have close ties to earlier works such as [24]. Clearly, the above unconstrained variational problem is equivalent to

$$\min_x f_1(x) + f_2(y), \quad \text{s.t. } Bx - y = 0. \tag{6}$$

It is evident that problem (6) is a special case of problem (1). As we will see in a concrete example below, whenever f_2 is a separable function, minimizing the augmented Lagrangian function of (6) with respect y is likely to be trivial.

1.2 An Example: Total Variation Minimization for Compressive Sensing

In recent years, a new theory of compressive sensing (CS) [11, 7, 6] — also known under the terminology of compressed sensing or compressive sampling — has drawn considerable research attention. It provides an alternative approach to data acquisition and compression that reduces the number of required samples, which could translate into simpler sensors, short sensing time, and/or reduced transmission/storage costs in suitable applications. The theory indicates that a sparse signal under some basis may still be recovered even though the number of measurements is deemed insufficient by Shannon's criterion. Nowadays, CS has been widely studied and applied to various fields (see [22, 10, 21, 13, 38, 39] for example).

Given measurements b, instead of finding the sparsest solution to $Ax = b$ by a combinatorial algorithm, which is generally NP-hard [25], one often chooses to minimize the ℓ_1-norm or the total variation (abbreviated TV) of x. In the context of CS, sufficient conditions for exact and stable recoveries are given in [12] and [6]. The use of TV regularization instead of the ℓ_1 term makes the reconstructed images sharper by preserving the edges or boundaries more accurately. Instead of assuming the signal is sparse, the premise of TV regularization is that the gradient of the underlying signal or image is sparse. In spite of those remarkable advantages of TV regularization, the properties of non-differentiability and non-linearity make TV minimization computationally more difficult than ℓ_1 minimization.

The use of TV has a long and rich history. It was introduced into imaging denoising problems by Rudin, Osher and Fatemi in 1992 [30]. From then on, TV minimizing models have become one

of the most popular and successful methodologies for image denoising [30, 8], deconvolution [9, 33] and restoration [5, 31, 34, 35], to cite just a few.

Specifically, the noise-free discrete TV model for CS reconstruction can be written as

$$\min_x \mathrm{TV}(x) \triangleq \sum_i \|D_i x\|_p, \quad \text{s.t. } Ax = b, \tag{7}$$

where $x \in \mathbb{R}^n$, or $x \in \mathbb{R}^{s\times t}$ with $s \cdot t = n$, represents an image, $D_i x \in \mathbb{R}^2$ is the discrete gradient of x at pixel i, $A \in \mathbb{R}^{m\times n}$ $(m < n)$ is the measurement matrix, $b \in \mathbb{R}^m$ is the measurement of x, and $p = 1$ or 2. The ℓ_p-norm could either be the ℓ_2-norm corresponding to the *isotropic* TV, or the ℓ_1-norm corresponding to the *anisotropic* TV. For reconstruction from noise-corrupted measurements, one can solve the ROF model instead

$$\min_x \mathrm{TV}(x) + \frac{\mu}{2}\|Ax - b\|^2. \tag{8}$$

For convenience, $\|.\|$ refers to the ℓ_2 norm hereafter.

To separate the non-differentiable ℓ_p-norm term, we can split variables by introducing $y_i = D_i x$. Then models (7) and (8) are equivalent to, respectively,

$$\min_{y_i,x} \sum_i \|y_i\|_p, \quad \text{s.t. } Ax = b \text{ and } D_i x = y_i \quad \forall i, \tag{9}$$

and

$$\min_{y_i,x} \sum_i \|y_i\|_p + \frac{\mu}{2}\|Ax - b\|^2, \quad \text{s.t. } D_i x = y_i \quad \forall i. \tag{10}$$

Both models (9) and (10) can be regarded as special forms of (1), while the non-differentiable parts of their augmented Lagrangian functions are easy to solve due to separability.

1.3 A Brief Overview of Related Works

For the method proposed in this paper, the ALM method plays a key role, which has been well researched for decades. The general quadratic penalty method turns a constrained optimization problem into a series of unconstrained problems by penalizing constraint violations. However, in theory it requires the penalty parameter to go to infinity to guarantee convergence, which may cause a deterioration in the numerical conditioning of the method. In 1969, Hestenes [20] and Powell [28] independently proposed the ALM method which, by introducing and adjusting Lagrangian multiplier estimates, no longer requires the penalty parameter to go to infinity for the method to converge. The augmented Lagrangian function differs from the standard Lagrangian function with an additional square penalty term, and differs from the quadratic penalty function with an additional term involving the multiplier λ times the constraints.

Numerically, it is usually impossible to find an exact minimizer of unconstrained minimization subproblem (2) at each iteration. For convex optimization, Rockafellar [29] proved the global convergence of the ALM method for any positive penalty parameter value, as long as the subproblems are solved to prescribed tolerances of increasing accuracy. In addition, the convergence theorem holds without assuming the objective function f to be differentiable.

Extending the classic ALM method, Glowinski *et al.* [17, 15] also suggested the so-called alternating direction method (often abbreviated as ADM). Instead of jointly minimizing the augmented Lagrangian function (2) at each iteration, ADM only demands minimizers with respect to x and y respectively before updating the multiplier, which may produce computationally more affordable iterations.

ADM is most effective when both subproblems can be efficiently and accurately solved, which is not always possible. For example, in TV minimization model (9) or (10), one of the subproblems is usually quadratic minimization that dominates the computation. Without further special structures, accurately solving such a quadratic minimization problem at each iteration can be excessively expensive.

Recently, it has been discovered that ALM can also be derived through an alternative approach called Bregman regularization [27, 36]. In particular, Goldstein and Osher [18] applied Bregman regularization to a split formulation in [33] to derive an algorithm for TV minimization called split Bregman, which is equivalent to ALM. In their computational experiments, however, they just used one sweep of alternating direction iteration to approximately minimize the augmented Lagrangian, resulting in a numerically efficient implementation that turns out to be equivalent to ADM.

Several solvers have been developed to solve TV minimization problem (7) or (8), or other variants. Among them, ℓ_1-Magic [6, 7], TwIST [3, 4] and NESTA [2] have been widely used. Specifically, ℓ_1-Magic solves a second-order cone reformulation of TV models. TwIST implements a two-step iterative shrinkage/thresholding (IST) algorithms, which exhibits much faster convergence rate than IST itself when the linear observation operator is ill-conditioned. NESTA is based on Nesterov's smoothing technique [26], extended to TV minimization by modifying the smooth approximation of the objective function. In Section 4, we will apply our proposed method to TV minimization with comparison to the above three state-of-the-art solvers.

1.4 Contributions

The classic ALM method is a fundamental and effective approach in constrained optimization. However, to apply it to realistic applications, it is critically important to design subproblem solvers capable of taking advantages of useful problem structures. In this work, we consider a rather common structure that in minimizing a non-smooth augmented Lagrangian function of two block variables, it is much easier to minimize with respect to one of the variables (in which the function may not be differentiable) than with respect to the other. This *structure of uneven complexity*

exists in a wide range of application problems.

We construct an efficient method for problems with the uneven structure that integrates two existing algorithmic ideas: (a) alternating direction and (b) nonmonotone line search. The former enables taking full advantages of the low-cost minimization in the "easy" direction; and the latter allows relatively quick and large steps in the "hard" direction. We are able to establish convergence for this algorithm by extending existing theoretical results.

Our numerical results on image reconstruction with TV regularization show that the proposed algorithm is robust and efficient, significantly outperforming several state-of-the-art solvers on most tested problems. The resulting MATLAB solver, called TVAL3, has been posted online [23].

2 Algorithm Construction

In this section, we first describe a first-order algorithm for solving the non-smooth, unconstrained minimization problem (5). The algorithm is an extension to the one in [37] designed for minimizing smooth functions. This non-smooth unconstrained minimization algorithm is then embedded into the classic ALM framework to form the backbone of an algorithm for solving the equality-constrained optimization problem (1). The motivation for the proposed algorithms is to take full advantages of the *structure of uneven complexity*, explained above, so that the derived algorithm has a relatively low iteration-complexity.

2.1 Nonmonotone Line Search for Smooth Functions

Grippo, Lampariello and Lucidi [19] proposed a nonmonotone line search scheme in 1986. In stead of requiring a monotone decrease in the objective function value as in the classic line search schemes, it only enforces a decrease in the maximum of previous k function values. More recently, Zhang and Hager [37] modified their line search scheme by replacing the "maximum" by a "weighted average" of all the previous function values, and showed that their scheme required fewer function and gradient evaluations on a large set of test problems. Convergence results for both schemes were established under the assumption that the objective function is differentiable.

In the nonmonotone line search algorithm (NLSA) given in [37], at each iteration the step length α_k is chosen to be uniformly bounded above and to satisfy the *nonmonotone Armijo condition*:

$$\psi(x_k + \alpha d_k) \leq C_k + \alpha\delta\nabla\psi(x_k)^T d_k, \tag{11}$$

where d_k is a descent direction, and C_k is a linear combination of all the previous function values, updated by the formulas

$$Q_{k+1} = \eta_k Q_k + 1, \qquad C_{k+1} = (\eta_k Q_k C_k + f(x_{k+1}))/Q_{k+1}, \tag{12}$$

where $\eta_k \geq 0$ controls the degree of nonmonotonicity. Specifically, the larger η_k is, the more nonmonotone the scheme is allowed to be. Additionally, one may also require that the *Wolfe*

conditions:

$$\nabla\psi(x_k + \alpha d_k)^T d_k \geq \sigma \nabla\psi(x_k)^T d_k, \tag{13}$$

be satisfied as well. Under these settings, they proved global convergence for smooth functions, and R-linear convergence rate for strongly convex functions.

In Section 3, we will extend the convergence proof in [37] to the case of minimizing the non-differentiable function $\psi(x)$ defined in (5).

2.2 Algorithm NADA

We now describe a nonmonotone line search algorithm for solving the non-smooth unconstrained minimization problem (5), which is equivalent to (3). From the standpoint of solving (3), the algorithm has a flavor of alternating minimization or block coordinate descent, but it does not require a monotone descent. For convenience, we call this approach *Nonmonotone Alternating Direction Algorithm*, or simply NADA.

Since our objective function $\psi(x) = \phi(x, y(x))$ is non-differentiable, the nonmonotone line search algorithm described in [37] is not directly applicable. The main modification is to replace the search direction $d_k = -\nabla\psi(x_k)$, which does not exist in our case, by $d_k = -\partial_1\phi(x_k, y(x_k))$. Such a modification can be justified as follows. Suppose that all the involved subdifferentials exist, then it follows from the chain rule that

$$d\psi(x) = \partial_1\phi(x, y(x)) + \partial_2\phi(x, y(x))dy(x).$$

By the construction of $y(x)$, we have $0 \in \partial_2\phi(x, y(x))$, then $\partial_1\phi(x, y(x)) \in d\psi(x)$. Hence, the search direction $d_k = -\partial_1\phi(x_k, y(x_k))$ can be regarded as a subgradient direction for $\psi(x)$. However, since we do not require that $y(x)$ be sub-differentiable, a vigorous convergence proof is still necessary for such a modification.

To suite our situation we need to modify the nonmonotone Armijo condition into the following form:

$$\phi(x_k + \alpha d_k, y_k) \leq C_k + \alpha\,\delta\,\partial_1\phi(x_k, y_k)^T d_k, \tag{14}$$

which is just (11) applied to the function $\phi(x, y_k)$ as a function of x at every iteration. In our implementation, we always choose the search direction $d_k = -\partial_1\phi(x_k, y_k)$, even though the convergence theorem in the next section actually allows a wider range of choices.

Algorithm 1 (Nonmonotone Alternating Direction Algorithm)**.**

Initialize $0 < \delta < 1 < \rho$, $0 \leq \eta_{\min} \leq \eta_{\max} \leq 1$, $\alpha_{\max} > 0$, *tol* > 0, *and* $Q_0 = 1$.

Choose starting points (x_0, y_0), *and set* $C_0 = \phi(x_0, y_0)$ *and* $k = 0$.

While $\|\partial_1\phi(x_k, y_k)\| >$ *tol* **Do**

(1) Compute $d_k = -\partial_1\phi(x_k, y_k)$, *and an initial trial step* $\bar{\alpha}_k > 0$.
(2) Choose $\alpha_k = \bar{\alpha}_k \rho^{-\theta_k}$ *where* θ_k *is the largest integer such that*
$\alpha_k \le \alpha_{\max}$ *and the nonmonotone Armijo condition* (14) *holds.*
(3) Set $x_{k+1} = x_k + \alpha_k d_k$.
(4) For some $\eta_k \in [\eta_{\min}, \eta_{\max}]$, *compute* Q_{k+1} *and* C_{k+1} *by (12).*
(5) Compute $y_{k+1} = y(x_{k+1}) \triangleq \operatorname{argmin}_y \phi(x_{k+1}, y)$.
(6) Increment k *and continue.*

End Do

Some additional remarks are in order.

Remark 1. *In our implementation, the initial trial step* $\bar{\alpha}_k$ *is chosen according to the Barzilai-Borwein method [1] to achieve good practical performance. For given* y_k, *applying BB method on minimizing* $\phi(x, y_k)$ *with respect to* x *leads to a trial step*

$$\bar{\alpha}_k \quad = \quad \frac{s_k^T s_k}{s_k^T z_k}, \tag{15}$$

or alternatively $\bar{\alpha}_k = s_k^T z_k / z_k^T z_k$, *where* $s_k = x_k - x_{k-1}$ *and* $z_k = \partial_1\phi(x_k, y_k) - \partial_1\phi(x_{k-1}, y_k)$.

Remark 2. *The integer parameter* θ_k *is not necessarily positive. In practical implementations, starting from the BB step, one can increase or decrease the step length by forward or backward tracking until the nonmonotone Armijo condition is satisfied.*

Compared to popular existing approaches, NADA differs from the traditional alternating minimization approach since it does not require exact (or high-precision) minimizers in all directions, and it differs from the block coordinate descent (BCD) approach since it does not require a monotone decease in the objective function. The convergence of Algorithm NADA will be analyzed in Section 3.

2.3 TVAL3

Embedding the unconstrained minimization algorithm NADA into the ALM framework, we obtain the following algorithm for solving the equality-constrained minimization problem (1).

Algorithm 2 (Augmented Lagrangian Multiplier)**.**

Initialize (x_0, y_0), $\mu_0 > 0$, *and* $\lambda_0 = \mathbf{0} \in \mathbb{R}^m$. *Set* $k = 0$.

While *"not converged"*, **Do**

(1) Call NADA to minimize $\phi(x, y) \triangleq \mathcal{L}_A(x, y, \lambda_k; \mu_k)$
starting from (x_k, y_k), *giving the output* (x_{k+1}, y_{k+1}).
(2) Update the multiplier: $\lambda_{k+1} = \lambda_k - \mu_k h(x_{k+1}, y_{k+1})$.
(3) If necessary, update the penalty parameter to get μ_{k+1}.

(4) Increment k and continue.

End Do

In order to achieve low-cost minimization with respect to y (the non-smooth part), a variable-splitting technique is usually coupled with this algorithm. The idea of variable-splitting has been used in various fields for years, and has recently been introduced into image deconvolution and TV minimization in [33]. For instance, we could split the ℓ_1-norm term from the finite difference operation as illustrated in Section 1.2.

Specifically, we apply NADA to the variants of TV regularized linear inverse problems (9) and (10) presented in Section 1.2, resulting in a solver that we call TVAL3 [23] (Total Variation Augmented Lagrangian ALternating-direction ALgorithm). In the particular case of solving (9), we have

$$\begin{aligned}\phi(x,y) &\triangleq \mathcal{L}_A(x,y,\nu,\lambda;\beta,\mu)\\ &= \sum_i \left(\|y_i\|_p - \nu_i^T(D_i x - y_i) + \frac{\beta}{2}\|D_i x - y_i\|^2 \right) + \frac{\mu}{2}\|Ax - b - \lambda/\mu\|^2.\end{aligned}$$

Then we can easily derive

$$\partial_1 \phi(x,y) = \sum_i (\beta D_i^T(D_i x - y) - D_i^T \nu_i) + \mu A^T(Ax - b - \lambda/\mu), \tag{16}$$

and, when $p = 2$ (isotropic TV),

$$y_i(x) \triangleq \operatorname*{argmin}_{y_i} \phi(x,y) = \max\left\{ \|D_i x - \nu_i/\beta\|_2 - \frac{1}{\beta}, 0 \right\} \frac{D_i x - \nu_i/\beta_i}{\|D_i x - \nu_i/\beta_i\|}, \tag{17}$$

which is the so-called *2D shrinkage formula*. When $p = 1$, one would apply an equally simple 1D shrinkage formula to obtain $y_i(x)$. As we can see, the computation of $y(x)$ is indeed very low in comparison to that of updating x from solving (16) at a fixed y value. Hence, the structure of uneven complexity is present.

In our implementation of NADA in TVAL3, d_k is chosen as the negative partial gradient given in (16), and $y(x)$ is computed using (17). A similar derivation is applicable to solving problem (10).

3 Convergence Results

As discussed earlier, the convergence of ALM has been thoroughly studied, so the convergence of the proposed Algorithm 2 relies on the convergence of Algorithm 1 (i.e., NADA) for the unconstrained subproblems. By extending Zhang and Hager's proof in [37], we present a convergence result for Algorithm-NADA in this section (its proof is given in the appendix). The fundamental extension is to allow the objective function, previously assumed to be continuously differentiable, to take the form $\phi(x, y(x))$ where we do not assume any differentiability of $y(x)$. As such, NADA is no longer a standard gradient or subgradient method previously studied in nonmonotone line search frameworks

3.1 Assumptions and Main Result

First recall the definition that $\psi(x) = \phi(x, y(x))$ for $y(x) = \text{argmin}_y \phi(x, y)$. We need to impose the following assumption on the function $\phi(x, y)$.

Assumption 1. *The function $\phi(x, y)$ is continuously differentiable with respect to x, and sub-differentiable with respect to y. Furthermore, $y(x) = \text{argmin}_y \phi(x, y)$ uniquely exists for each x.*

We note that the above assumption does not imply the sub-differentiability of $\psi(x) = \phi(x, y(x))$ since $y(x)$ is not assumed to be sub-differentiable. In addition, $\phi(x, y)$ does not have to be convex with respect to y for $y(x)$ to uniquely exist.

Let $\partial_1\phi$ and $\partial_2\phi$ refer to the sub-differentials of ϕ with respect to x and y, respectively. The convergence proof of NADA makes use of the following assumptions.

Assumption 2 (Direction Assumption)**.** *There exist $c_1 > 0$ and $c_2 > 0$ such that*

$$\begin{cases} \partial_1\phi(x_k, y_k)^T d_k \leq -c_1\|\partial_1\phi(x_k, y_k)\|^2, \\ \|d_k\| \leq c_2\|\partial_1\phi(x_k, y_k)\|. \end{cases} \tag{18}$$

This assumption obviously holds for $d_k = -\partial_1\phi_k(x_k)$ with $c_1 = c_2 = 1$. However, this assumption allows more generality. For instance, certain approximations to $-\partial_1\phi_k(x_k)$ would become permissible.

Assumption 3 (Lipschitz Condition)**.** *There exists $L > 0$, such that at any given y, and for all x and $\tilde{x}$,*

$$\|\partial_1\phi(x, y) - \partial_1\phi(\tilde{x}, y)\| \leq L\|x - \tilde{x}\|. \tag{19}$$

Assumption 4 (Boundedness from below)**.** *The function $\phi(x, y)$ is bounded below, i.e.,*

$$\phi(x, y) \geq -M$$

for some $M > 0$ and for all (x, y).

We note that Lipschitz continuity and the boundedness from below are widely assumed in the analysis of convergence of gradient-type methods.

The main convergence result of this paper is as follows:

Theorem 1. *Under Assumptions 1-4, the iterate sequence $\{(x_k, y_k)\}$ generated by Algorithm-NADA, which is well defined, satisfies*

$$\begin{cases} \lim\limits_{k\to 0} \partial_1\phi(x_k, y_k) = 0, \\ \partial_2\phi(x_k, y_k) \ni 0. \end{cases} \tag{20}$$

Our proof of this theorem follows a similar path as the convergence proof in [37], with extra steps to connect the non-differentiable part with the differentiable part by means of alternating minimization. We include the detailed proof in the appendix.

Based on Theorem 1, we can easily deduce global convergence of Algorithm-NADA under a further convexity assumption. We state the following corollary without a proof.

Corollary 1. *If $\phi(x, y)$ is jointly convex, lower semi-continuous and coercive, then under Assumptions 1-4 the sequence $\{(x_k, y_k)\}$ generated by Algorithm-NADA converges to a minimizer (x^*, y^*) of problem (3).*

As indicated before, the convergence of the overall ALM algorithm, Algorithm 2, follows from that of NADA.

4 Numerical Results in Image Reconstruction

To demonstrate the performance of the proposed method on problems with the structure of uneven complexity, we conducted numerical experiments on TV minimization problems from image reconstruction in CS; that is, solving model (9) or (10). Our implementation of Algorithm 2 is the solver TVAL3, which was compared to three other state-of-the-art TV minimization solvers: ℓ_1-Magic [6, 7], TwIST [3, 4] and NESTA [2]. All experiments were performed on a Lenovo X301 laptop with a 1.4GHz Intel Core 2 Duo SU9400 and 2GB of DDR3 memory, running Windows XP and MATLAB R2009a (32-bit).

Throughout the experiments, we always used a default set of parameter values for TVAL3. Specifically, we set $\eta = .9995$, $\rho = 5/3$, $\delta = 10^{-5}$ and $\alpha_{\max} = 10^4$ (see Algorithm 1), and initialized multiplier estimate to the zero vector as presented in Algorithm 2. Additionally, the penalty parameter might vary in a range of 2^5 to 2^9 according to distinct noise level and required accuracy. In spite of a lack of theoretical guidance, we have found that it is not particularly difficult to choose adequate values for penalty parameters since the algorithm is not overly sensitive to such values as long as they fall into some appropriate but reasonably wide range. A few trial-and-error attempts are usually needed to find good penalty parameter values, judged by observed convergence speed.

In an effort to make comparisons fair, for all other tested solvers mentioned above, we did tune their parameters and try to make them perform in a as near optimal fashion as we could.

4.1 Tests on Synthetic Data

The first three tests are base on synthetic data. In the first test, the data x is an $n = 64 \times 64$ phantom image from which an observation $b = Ax$ is generated without additive noise. The matrix $A \in \mathbb{R}^{m \times n}$, where $m = 0.3n$, is generated from orthogonalizing the rows of a Gaussian random matrix by QR factorization. Then we apply TVAL3, TwIST, NESTA and ℓ_1-Magic to model (7) and obtain a recovered image from each solver. The quality of recovered images is measured by the

signal-to-noise ratio (SNR). Parameters were extensively tuned to achieve a near-best performance possible. The test results are presented in Figure 1.

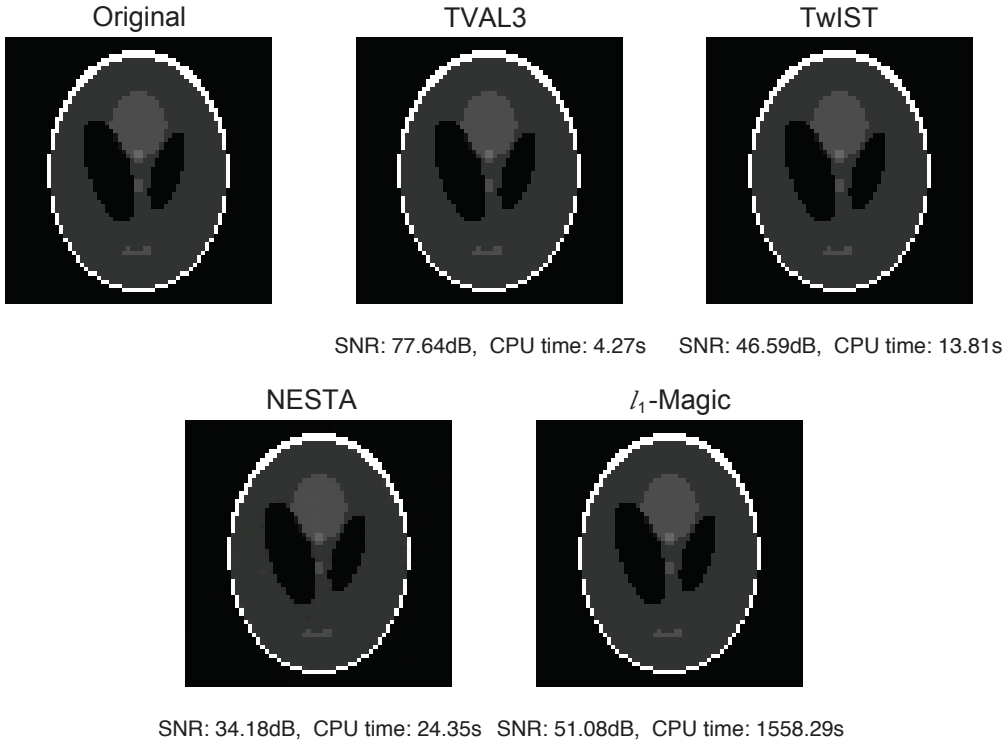

Figure 1: Recovery of a 64×64 phantom image (shown in the top-left) from 30% noiseless measurements. **Top-middle:** reconstructed by TVAL3. **Top-right:** reconstructed by TwIST. **Bottom-left:** reconstructed by NESTA. **Bottom-right:** reconstructed by ℓ_1-Magic.

From Figure 1, we observe that TVAL3 achieved the highest SNR at 77.6dB, while taking the shortest running time (4.3 seconds). The second highest SNR was obtained by ℓ_1-Magic at 51.1dB at the cost of taking an unacceptable amount of time (1558.3 seconds). TwIST and NESTA attained relatively medium-quality images (SNR around 46.6dB and 34.2dB respectively) within reasonable running times (13.8 and 24.4 seconds, respectively). This test suggests that TVAL3 is capable of achieving high accuracy within an affordable running time on the tested image reconstruction problems, outperforming some state-of-the-art solvers.

The second test is much more challenging with a larger image and rather high-level Gaussian noise. Specifically, the data is a 256×256 MR brain image contained rather complicated features. This time, the measurement matrix A is a column permuted, partial Walsh Hadamard matrix with only 10% rows selected at random. To the observation vector $b = Ax$ we added Gaussian noise at the level of 10% in magnitude. Specifically, a noisy observation is synthesized by the formula (in

the Matlab format)

$$b = b + \sigma * \texttt{mean}\,(\texttt{abs}\,(b)) * \texttt{randn}\,(m, 1), \tag{21}$$

where $b \in \mathbb{R}^m$ on the right-hand side is the noiseless observation, and σ represents the noise level.

From the first test on the phantom image, we know that ℓ_1-Magic, though producing good quality solutions, can become excessively expensive on relatively large-scale problems. For this reason, we excluded it from the second test. In Figure 2, we present test results for TVAL3, TwIST and NESTA solving ROF model (8). We observe from Figure 2 that TVAL3 produced the best recovery quality with the shortest amount of running time, TwIST produced the poorest recovery quality with the longest amount of running time, while NESTA is in the middle on both accounts. These results indicate that TVAL3 is likely to be more efficient and more robust in solving certain highly difficult problems.

Finally, in the third test we fix the Gaussian noise level to 10% and repeat the experiment for 90 different sampling ratios ranging from 9% to 98% with 1% increment. All the parameters are set as the same as in the second test. The results are plotted in Figure 3, showing the recovery quality and running time for TVAL3, TwIST and NESTA. Figure 3 indicates that on these test cases TVAL3 always achieves the best quality (highest SNR) with the shortest running time among the three tested solvers. TwIST and NESTA attain similar accuracy, but TwIST is much slower especially when the sampling ratio is relatively low. These facts are consistent with what we have discovered from Figure 2.

4.2 A Test with Hardware-Measured Data

To see the performance of the solvers under a more realistic environment, we did a test using data collected by Rice's *single-pixel camera* [32]. Simply speaking, it is a compressive sensing camera using digital micro-mirror device to generate measurements (for more details see [32]) . In this test, we focused on reconstructing infrared data captured by this single-pixel camera.

As is shown in Figure 4, a canvas board with two letters "IR" written on it by charcoal pencil is entirely covered by blue oil paint, which makes the letters "IR" invisible to human eyes. This board was illuminated by a 150-watt halogen lamp and measurements were gathered by the single-pixel camera equipped with an infrared sensor. We applied TVAL3, TwIST, NESTA and ℓ_1-Magic to ROF model (8) in order to recover the image, respectively from top to bottom in Figure 5, where the recovered images, from left to right, are corresponding to 15%, 35%, and 50% sampling ratios, respectively.

In this test, measurements were not synthesized, but collected from hardware. Hence, there is no "ground-truth" solution available. As such, recovery quality can only be judged by subjective visual examinations. It is perhaps agreeable that, in Figure 5, the results by TwIST (on the second row) are visually inferior to others. Another unmistakable observation is that ℓ_1-Magic took at least 10 times longer running time than others, while the others required much less running times.

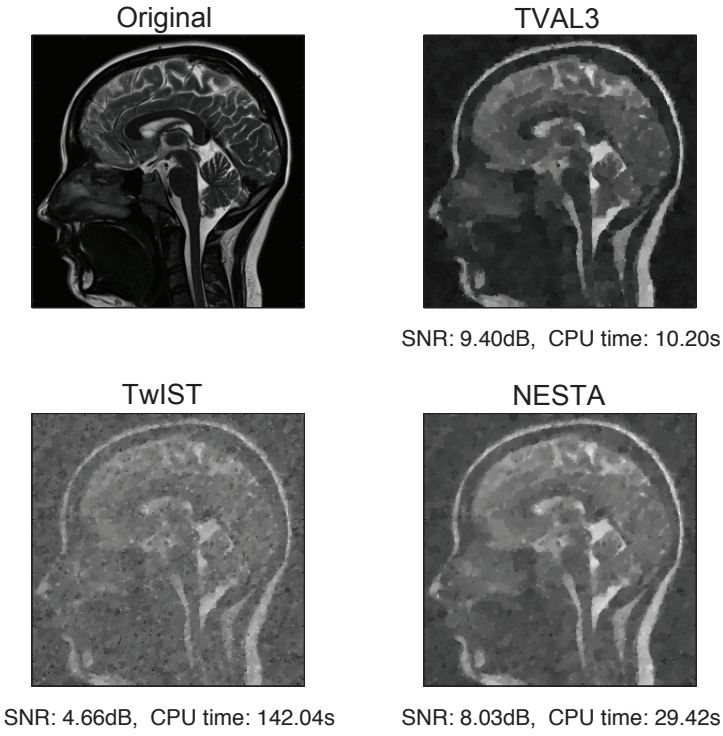

Figure 2: Recovery of a 256×256 MR brain image (top-left) from 10% measurements with noise at 10% level. **Top-right:** reconstructed by TVAL3. **Bottom-left:** reconstructed by TwIST. **Bottom-right:** reconstructed by NESTA.

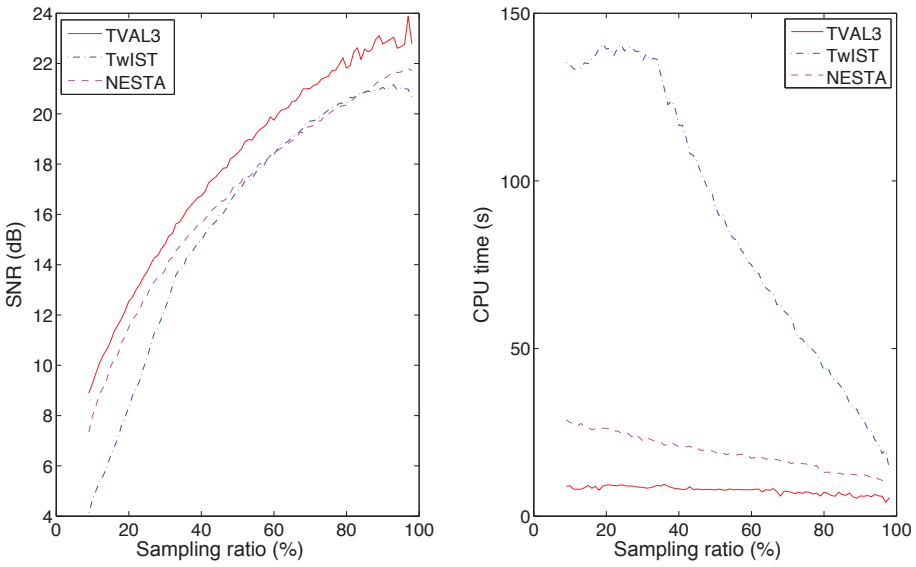

Figure 3: Recoverability for 256×256 MR brain image. The noise level is 10%. **Left:** average SNR. **Right:** average running time. SNR and running time are measured simultaneously with the growth of the sampling ratio.

In short, numerical results indicate that TVAL3 is at least competitive to other state-of-the-art TV solvers for CS reconstruction. In fact, it seems to be more efficient and more robust in most tests using synthetic data.

5 Conclusions

In this paper, we have proposed, analyzed and tested an algorithm for solving non-smooth unconstrained optimization problems with a structure called uneven complexity in terms of minimizing the objective with respect to two groups of variables. Such a structure widely exists in application problems including some total-variation minimization problems in various image processing tasks. The proposed algorithm effectively integrates the ideas of alternating direction and nonmonotone line search to take advantages of both techniques, leading to relatively low-cost iterations for suitably structured problems.

We have established convergence for this algorithm by extending convergence results in [37] that are applicable only to smooth objective functions. When embedded into the ALM framework as the subproblem solver, the proposed approach leads to efficient ALM implementations for solving targeted equality-constrained optimization problems. Based on this approach, a TV minimization solver called TVAL3 is constructed. Extensive experiments demonstrate that TVAL3 compares

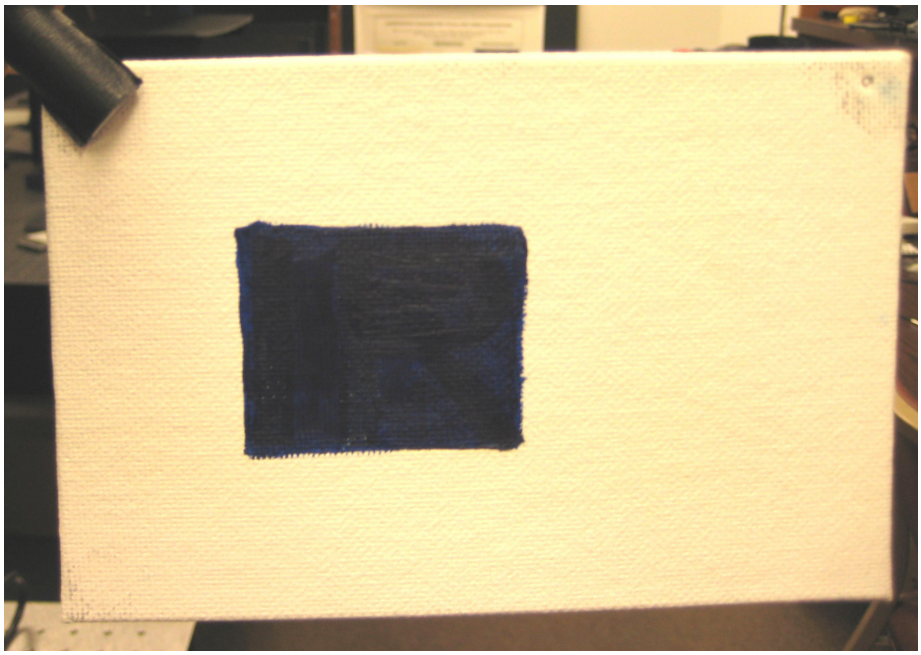

Figure 4: Target image under visible light with letters IR covered by paint.

competitively, and often favorably, with several state-of-the-art TV solvers in the field.

Acknowledgments

The work of the first author was supported in part by NSF Grant DMS-0811188. The work of the second author was supported in part by NSF grants DMS-07-48839 and ECCS-1028790, as well as ONR Grant N00014-08-1-1101. The work of the fourth author was supported in part by NSF Grant DMS-0811188, ONR Grant N00014-08-1-1101, and NSF Grant DMS-1115950. The first and the fourth authors also appreciate a gift fund from Bell Labs, Alcatel-Lucent to Rice University that partially supported their travels to international conferences.

References

[1] J. Barzilai and J. M. Borwein, *Two-point step size gradient methods*, IMA J. Numer. Anal., vol. 8, pp. 141–148, 1988.

[2] S. Becker, J. Bobin and E. Candès, *NESTA: A fast and accurate first-order method for sparse recovery*, SIAM J. Imaging Sci., vol. 4, pp. 1–39, 2011.

[3] J. Bioucas-Dias and M. Figueiredo, *A new TwIST: Two-step iterative thresholding algorithm for image restoration*, IEEE Trans. Imag. Process., vol. 16, no. 12, pp. 2992–3004, 2007.

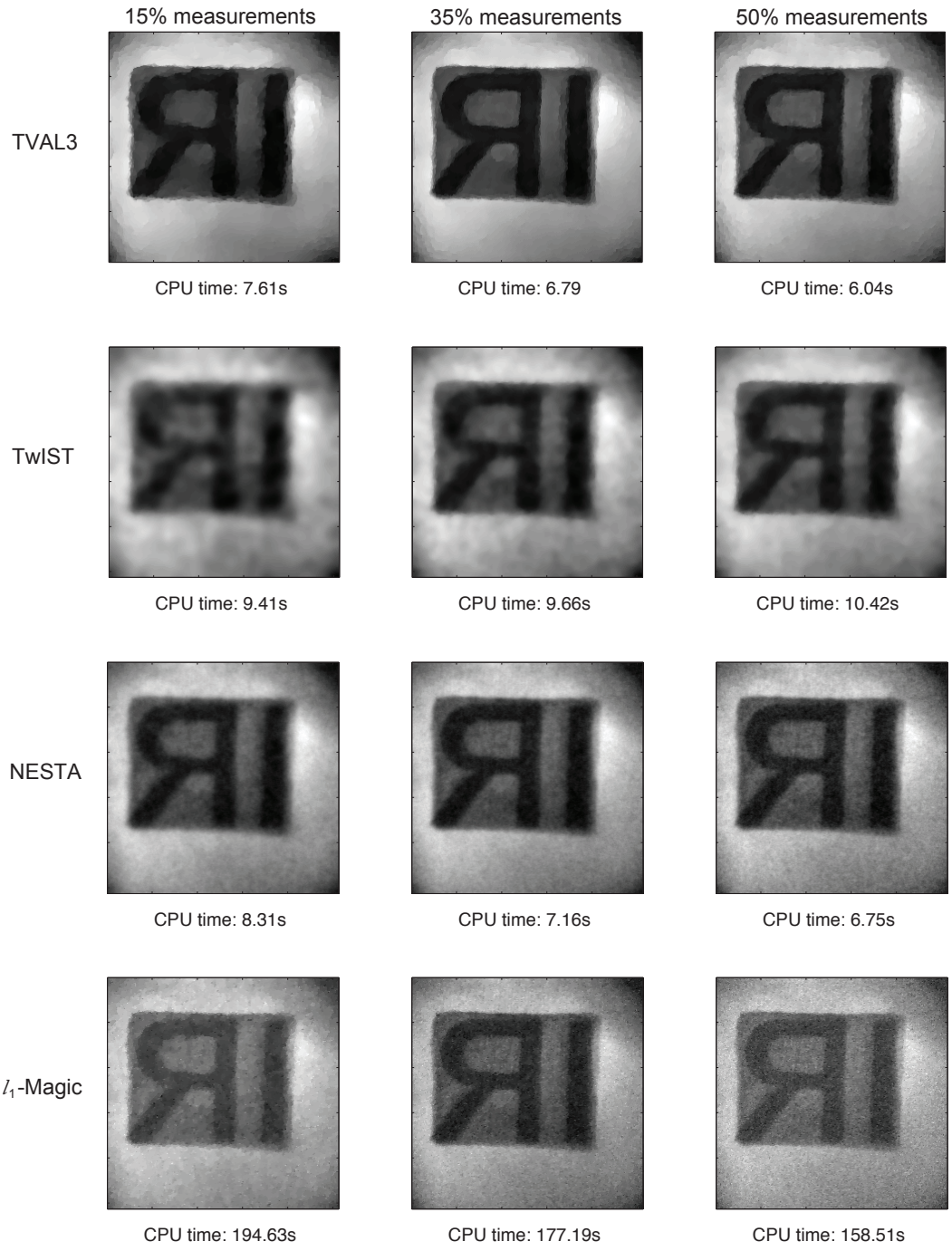

Figure 5: Recovery of a 256×256 infrared image. The four rows (top to bottom) are reconstructed by TVAL3, TwIST, NESTA and ℓ_1-Magic, respectively, for sampling ratios (left to right) 15%, 35%, and 50%, respectively.

[4] J. Bioucas-Dias and M. Figueiredo, *Two-step algorithms for linear inverse problems with non-quadratic regularization*, IEEE International Conference on Image Processing–ICIP 2007, San Antonio, TX, USA, September 2007.

[5] Y. Boykov, O. Veksler and R. Zabih, *Fast approximate energy minimization via graph cuts*, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 23, no. 11, pp. 1222–1239, 2001.

[6] E. Candès, J. Romberg, and T. Tao, *Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Inform. Theory, vol. 52, no. 2, pp. 489–509, 2006.

[7] E. Candès, and T. Tao, *Near optimal signal recovery from random projections: Universal encoding strategies*, IEEE Trans. on Inform. Theory, vol. 52, no. 12, pp. 5406–5425, 2006.

[8] A. Chambolle, *An algorithm for total variation minimization and applications*, Journal of Mathematical Imaging and Vision, vol. 20, 89–97, Jan. 2004.

[9] T. Chan and C. K. Wong, *Total variation blind deconvolution*, IEEE Trans. Image Processing, vol. 7, no. 3, pp. 370–375, 1998.

[10] T. Chang, L. He and T. Fang, *MR image reconstruction from sparse radial samples using bregman iteration*, ISMRM, 2006.

[11] D. Donoho, *Compressed sensing*, IEEE Transactions on Information Theory, vol. 52, no. 4, pp. 1289–1306, 2006.

[12] D. Donoho, *Neighborly polytopes and sparse solution of underdetermined linear equations*, IEEE. Trans. Info. Theory, 2006.

[13] M. F. Duarte, S. Sarvotham, D. Baron, M. B. Wakin and R. G. Baraniuk, *Distributed compressed sensing of jointly sparse signals*, in 39th Asilomar Conference on Signals, Systems and Computers, pp. 1537–1541, 2005.

[14] M. Fortin and R. Glowinski, *Méthodes de Lagrangien Augmenté*, Application à la résolution numérique de problèmes aux limites, Dunod-Bordas, Paris, 1982 (in French).

[15] D. Gabay and B. Mercier, *A dual algorithm for the solution of nonlinear variational problems via finite element approximations*, Comp. Math. Appl., vol. 2, pp. 17–40, 1976.

[16] R. Glowinski, *Numerical Methods for Nonlinear Variational Problems*, Springer-Verlag, New York, Berlin, Heidelberg, Tokyo, 1984.

[17] R. Glowinski and A. Marrocco, *Sur l'approximation par éléments finis d'ordre un et la résolution par pénalisation-dualité d'une classe de problèmes de Dirichlet nonlinéaires*, C. R. Acad. Sci. Paris, 278A, 1649–1652, 1974 (in French).

[18] T. Goldstein and S. Osher, *The split Bregman method for L1 regularized problems*, SIAM J. Imag. Sci., vol. 2, no. 2, pp. 323–343, April 2009.

[19] L. Grippo, F. Lampariello and S. Lucidi, *A nonmonotone line search technique for Newton's method*, SIAM J. Numer. Anal., vol. 23, pp. 707–716, 1986.

[20] M. R. Hestenes, *Multiplier and gradient methods*, Journal of Optimization Theory and Applications, vol. 4, pp. 303–320, and in Computing Methods in Optimization Problems, 2 (Eds L.A. Zadeh, L.W. Neustadt and A.V. Balakrishnan), Academic Press, New York, 1969.

[21] J. Laska, S. Kirolos, M. Duarte, T. Ragheb, R. Baraniuk and Y. Massoud, *Theory and implementaion of an analog-to-information converter using random demodulation*, In Proceedings of the IEEE International Symposium on Circuites and Systems (ISCAS), New Orleans, Louisiana, 2007.

[22] C. Li, T. Sun, K. Kelly and Y. Zhang, *A compressive sensing and unmixing scheme for hyperspectral data processing*, IEEE Transactions on Image Processing, vol. 21, pp. 1200–1210, 2012.

[23] C. Li, Y. Zhang and W. Yin,
http://www.caam.rice.edu/~optimization/L1/TVAL3/.

[24] P. L. Lions and B. Mercier, *Splitting algorithms for the sum of two nonlinear operators*, SIAM J. Numer. Anal., vol. 16, pp. 964–979, 1979.

[25] B. K. Natarajan, *Sparse approximate solutions to linear systems*, SIAM Journal on Computing, vol. 24, pp. 227–234, 1995.

[26] Yu. Nesterov, *Smooth minimization of non-smooth functions*, Math. Program., Ser. A 103, pp. 127–152, 2005.

[27] S. Osher, M. Burger, D. Goldfarb, J. Xu and W. Yin, *An iterated regularization method for total variation based image restoration*, SIAM Journal on Multiscale Modeling and Simulation, vol. 4, pp. 460–489, 2005.

[28] M. J. D. Powell, *A Method for Nonlinear Constraints in Minimization Problems*, Optimization (Ed. R. Fletcher), Academic Press, London, New York, pp. 283–298, 1969.

[29] R. T. Rockafellar, *The multiplier method of Hestenes and Powell applied to convex programming*, Journal of Optimization Theory and Applications, vol. 12, no. 6, pp. 555–562, 1973.

[30] L. Rudin, S. Osher and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Physica D, pp. 259–268, 1992.

[31] T. Sun, G. L. Woods, M. F. Duarte, K. F. Kelly, C. Li and Y. Zhang, *OBIC Measurements without Lasers or Raster-Scanning Based on Compressive Sensing*, in Proceedings of the 35th International Symposium for Testing and Failure Analysis, 2009.

[32] D. Takhar, J. N. Laska, M. B. Wakin, M. F. Duarte, D. Baron, S. Sarvotham, K. F. Kelly and R. G. Baraniuk, *A new compressive imaging camera architecture using optical-domain compression*, Computational Imaging IV, vol. 6065, pp. 43–52, Jan. 2006.

[33] Y. Wang, J. Yang, W. Yin and Y. Zhang, *A new alternating minimization algorithm for total variation image reconstruction*, SIAM J. Imag. Sci., vol. 1, no. 4, pp. 248–272, 2008.

[34] J. Yang, W. Yin and Y. Zhang, *A fast alternating direction method for TVL1-L2 signal reconstruction from partial Fourier data*, Technical Report, TR08-27, CAAM, Rice University, 2008.

[35] W. Yin, S. Morgan, J. Yang and Y. Zhang, *Practical compressive sensing with Toeplitz and circulant matrices*, In proceedings of Visual Communications and Image Processing (VCIP), 2010.

[36] W. Yin, S. Osher, D. Goldfarb and J. Darbon, *Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing*, SIAM Journal on Imaging Sciences, vol 1, pp. 143–168, 2008.

[37] H. Zhang and W. W. Hager, *A nonmonotone line search technique and its application to unconstrained optimization*, SIAM J. Optim., vol. 14, pp. 1043–1056, 2004.

[38] C. Li, H. Jiang, P.A., Wilford, Y. Zhang, *Video coding using compressive sensing for wireless communications*, IEEE Wireless Communications and Networking Conference (WCNC), 10.1109/WCNC.2011.5779474, pp. 2077–2082, 2011.

[39] H. Jiang, C. Li, R. Haimi-Cohen, P. Wilford and Y. Zhang, *Scalable Video Coding using Compressive Sensing*, Bell Labs Technical Journal, **16**, No.4, 2012.

Appendix: Proof of Theorem 1

For notational simplicity, let us define

$$\phi_k(\cdot) = \phi(\cdot, y_k) \quad \text{and} \quad \nabla\phi_k(\cdot) = \partial_1\phi(\cdot, y_k). \tag{22}$$

The proof of the theorem relies on two lemmas. The two lemmas are modifications of their counterparts in [37]. Since our objective may contain a non-differentiable part, the key modification is to connect this non-differentiable part to the differentiable part by means of alternating minimization. Otherwise, the line of proofs follows closely that given in [37].

The first lemma presents some basic properties and established that the algorithm is well-defined.

Lemma 1. *If $\nabla\phi_k(x_k)^T d_k \le 0$ holds for each k, then for the sequences generated by Algorithm-NADA, we have $\phi_k(x_k) \le \phi_{k-1}(x_k) \le C_k$ for each k and $\{C_k\}$ is monotonically non-increasing. Moreover, if $\nabla\phi_k(x_k)^T d_k < 0$, a step length $\alpha_k > 0$ always exists so that the nonmonotone Armijo condition (11) holds.*

Proof. Define real-valued function

$$D_k(t) = \frac{tC_{k-1} + \phi_{k-1}(x_k)t + 1}{} \text{ for } t \ge 0,$$

then

$$D_k'(t) = \frac{C_{k-1} - \phi_{k-1}(x_k)}{(t+1)^2} \qquad \text{for } t \ge 0.$$

Due to the nonmonotone Armijo condition (11) and $\nabla\phi_k(x_k)^T d_k \le 0$, we have

$$C_{k-1} - \phi_{k-1}(x_k) \ge -\delta\alpha_{k-1}\nabla\phi_{k-1}(x_{k-1})^T d_{k-1} \ge 0.$$

Therefore, $D_k'(t) \ge 0$ holds for any $t \ge 0$, and then D_k is non-decreasing.
Since

$$D_k(0) = \phi_{k-1}(x_k) \quad \text{and} \quad D_k(\eta_{k-1}Q_{k-1}) = C_k,$$

we have

$$\phi_{k-1}(x_k) \le C_k, \quad \forall k.$$

As is defined in Algorithm-NADA,

$$y_k = \operatorname*{argmin}_y \phi(x_k, y).$$

Therefore,

$$\phi(x_k, y_k) \le \phi(x_k, y_{k-1}).$$

Hence, $\phi_k(x_k) \le \phi_{k-1}(x_k) \le C_k$ holds for any k.

Furthermore,

$$C_{k+1} = \frac{(\eta_k Q_k C_k + \phi_k(x_{k+1}))}{Q_{k+1}} \leq \frac{(\eta_k Q_k C_k + C_{k+1})}{Q_{k+1}},$$

i.e.,

$$(\eta_k Q_k + 1) C_{k+1} \leq (\eta_k Q_k C_k + C_{k+1}),$$

i.e.,

$$C_{k+1} \leq C_k.$$

Thus, $\{C_k\}$ is monotonically non-increasing.

If C_k is replaced by $\phi_k(x_k)$ in (11), the nonmonotone Armijo condition becomes the standard Armijo condition. It is well-known that $\alpha_k > 0$ exists for the standard Armijo condition while $\nabla\phi_k(x_k)^T d_k < 0$ and ϕ is bounded below. Since $\phi_k(x_k) \leq C_k$, it follows $\alpha_k > 0$ exists as well for the nonmonotone Armijo condition:

$$\phi_k(x_k + \alpha_k d_k) \leq C_k + \delta\alpha_k \nabla\phi_k(x_k)^T d_k.$$

Now we defining the quantity A_k by

$$A_k = \frac{1}{k+1} \sum_{i=0}^{k} \phi_k(x_k). \tag{23}$$

By induction, it is easy to show that C_k is bounded above by A_k. Together with the facts that C_k is also bounded below by $\phi_k(x_k)$ and $\alpha_k > 0$ always exists, it is clear that Algorithm-NADA is well-defined. □

In the next lemma, a lower bound for the step length generated by Algorithm-NADA will be given.

Lemma 2. *Assume that $\nabla\phi_k(x_k)^T d_k \leq 0$ for all k and that Lipschitz condition (19) holds with constant L. Then*

$$\alpha_k \geq \min\left\{\frac{\alpha_{\max}}{\rho}, \frac{2(1-\delta)}{L\rho}\frac{|\nabla\phi_k(x_k)^T d_k|}{\|d_k\|^2}\right\}. \tag{24}$$

Proof. It is noteworthy that $\rho > 1$ is required in Algorithm-NADA. If $\rho\alpha_k \geq \alpha_{\max}$, then the lemma already holds. Otherwise,

$$\rho\alpha_k = \bar{\alpha_k}\rho^{\theta_k+1} < \alpha_{\max},$$

which indicates that θ_k is not the largest integer to make the k-th step length less than $\alpha_{\max}$. According to Algorithm-NADA, θ_k must be the largest integer satisfying the nonmonotone Armijo condition (11), which leads to

$$\phi_k(x_k + \rho\alpha_k d_k) \geq C_k + \delta\rho\alpha_k \nabla\phi_k(x_k)^T d_k.$$

Lemma 1 showed $C_k \geq \phi_k(x_k)$, so

$$\phi_k(x_k + \rho\alpha_k d_k) \geq \phi_k(x_k) + \delta\rho\alpha_k \nabla\phi_k(x_k)^T d_k. \tag{25}$$

On the other hand, for $\alpha > 0$ we have

$$\int_0^\alpha (\nabla\phi_k(x_k + td_k) - \nabla\phi_k(x_k))\, d_k\, dt = \phi_k(x_k + \alpha d_k) - \phi_k(x_k) - \alpha\nabla\phi_k(x_k)^T d_k.$$

Together with the Lipschitz condition, we obtain

$$\begin{aligned}
\phi_k(x_k + \alpha d_k) &= \phi_k(x_k) + \alpha\nabla\phi_k(x_k)^T d_k + \int_0^\alpha (\nabla\phi_k(x_k + td_k) - \nabla\phi_k(x_k))\, d_k\, dt \\
&\leq \phi_k(x_k) + \alpha\nabla\phi_k(x_k)^T d_k + \int_0^\alpha tL\|d_k\|^2\, dt \\
&= \phi_k(x_k) + \alpha\nabla\phi_k(x_k)^T d_k + \frac{1}{2}L\alpha^2\|d_k\|^2.
\end{aligned}$$

Let $\alpha = \rho\alpha_k$, then

$$\phi_k(x_k + \rho\alpha_k d_k) \leq \phi_k(x_k) + \rho\alpha_k\nabla\phi_k(x_k)^T d_k + \frac{1}{2}L\rho^2\alpha_k^2\|d_k\|^2. \tag{26}$$

Comparing (25) to (26), we deduce that

$$(\delta - 1)\nabla\phi_k(x_k)^T d_k \leq \frac{1}{2}L\rho\alpha_k\|d_k\|^2.$$

Since $\nabla\phi_k(x_k)^T d_k \leq 0$,

$$\alpha_k \geq \frac{2(1-\delta)}{L\rho}\frac{|\nabla\phi_k(x_k)^T d_k|}{\|d_k\|^2}.$$

Therefore, the step length α_k is bounded below as in (24). □

With the aid of the lower bound (24), we now are ready to prove Theorem 1. We need to establish the two relationships given in (20).

Proof. First, by definition in Algorithm-NADA,

$$y_k = \operatorname*{argmin}_y \phi(x_k, y).$$

Hence, it always holds true that

$$0 \in \partial_2\phi(x_k, y_k).$$

Now it suffices to show that the limit holds true in (20). Consider the nonmonotone Armijo condition:

$$\phi_k(x_k + \alpha_k d_k) \leq C_k + \delta\alpha_k\nabla\phi_k(x_k)^T d_k. \tag{27}$$

If $\rho\alpha_k < \alpha_{\max}$, in view of the lower bound (24) on α_k in Lemma 2 and the direction assumption (18),

$$\begin{aligned}\phi_k(x_k+\alpha_k d_k) &\leq C_k - \delta\frac{2(1-\delta)}{L\rho}\frac{|\nabla\phi_k(x_k)^T d_k|^2}{\|d_k\|^2}\\ &\leq C_k - \frac{2\delta(1-\delta)}{L\rho}\frac{c_1^2\|\nabla\phi_k(x_k)\|^4}{c_2^2\|\nabla\phi_k(x_k)\|^2}\\ &= C_k - \left[\frac{2\delta(1-\delta)c_1^2}{L\rho c_2^2}\right]\|\nabla\phi_k(x_k)\|^2.\end{aligned}$$

On the other hand, if $\rho\alpha_k \geq \alpha_{\max}$, the lower bound (24), together with the direction assumption (18), gives

$$\begin{aligned}\phi_k(x_k+\alpha_k d_k) &\leq C_k + \delta\alpha_k\nabla\phi_k(x_k)^T d_k\\ &\leq C_k - \delta\alpha_k c_1\|\nabla\phi_k(x_k)\|^2\\ &\leq C_k - \frac{\delta\alpha_{\max}c_1}{\rho}\|\nabla\phi_k(x_k)\|^2.\end{aligned}$$

Introducing a constant

$$\tilde{\tau} = \min\left\{\frac{2\delta(1-\delta)c_1^2}{L\rho c_2^2}, \frac{\delta\alpha_{\max}c_1}{\rho}\right\},$$

we can combine the above inequalities into

$$\phi_k(x_k+\alpha_k d_k) \leq C_k - \tilde{\tau}\|\nabla\phi_k(x_k)\|^2. \tag{28}$$

Next we show by induction that for all k

$$\frac{1}{Q_k} \geq 1 - \eta_{max}, \tag{29}$$

which obviously holds for $k=0$ given that $Q_0 = 1$. Assume that (29) holds for $k=j$. Then

$$Q_{j+1} = \eta_j Q_j + 1 \leq \frac{\eta_j}{1-\eta_{max}} + 1 \leq \frac{\eta_{max}}{1-\eta_{max}} + 1 = \frac{1}{1-\eta_{max}},$$

implying that (29) also holds for $k = j+1$. Hence, (29) holds for all k.

It follows from (28) and (29) that

$$\begin{aligned}C_k - C_{k+1} &= C_k - \frac{\eta_k Q_k C_k + \phi_k(x_{k+1})}{Q_{k+1}}\\ &= \frac{C_k(\eta_k Q_k + 1) - (\eta_k Q_k C_k + \phi_k(x_{k+1}))}{Q_{k+1}}\\ &= \frac{C_k - \phi_k(x_{k+1})}{Q_{k+1}}\\ &\geq \frac{\tilde{\tau}\|\nabla\phi_k(x_k)\|^2}{Q_{k+1}}\\ &\geq \tilde{\tau}(1-\eta_{max})\|\nabla\phi_k(x_k)\|^2.\end{aligned} \tag{30}$$

Since ϕ is bounded below by assumption, $\{C_k\}$ is also bounded below. In addition, by Lemma 1, $\{C_k\}$ is monotonically non-increasing, hence convergent. Therefore, the left-hand side of (30) tends to zero, so does the right-hand side; i.e., $\|\nabla\phi_k(x_k)\| \to 0$. Finally, by definition (22),

$$\lim_{k\to 0} \partial_1\phi(x_k, y_k) = 0,$$

which completes the proof. □