ANALYSIS AND GENERALIZATIONS OF THE LINEARIZED BREGMAN METHOD*

WOTAO YIN†

Abstract. This paper analyzes and improves the linearized Bregman method for solving the basis pursuit and related sparse optimization problems. The analysis shows that the linearized Bregman method has the exact regularization property, namely, it converges to an exact solution of the basis pursuit problem whenever its smooth parameter α is greater than a certain value. The analysis is based on showing that the linearized Bregman algorithm is equivalent to gradient descent applied to a certain dual formulation. This result motivates generalizations of the algorithm enabling the use of gradient-based optimization techniques such as line search, Barzilai-Borwein, L-BFGS, and nonlinear conjugate gradient methods. In the numerical simulations, the two proposed implementations, one using Barzilai-Borwein steps with nonmonotone line search and the other using L-BFGS, gave more accurate solutions in much shorter times than the existing basic implementation of the linearized Bregman method (with a so-called kicking technique).

Key words. Bregman, linearized Bregman, compressed sensing, l1 minimization, basis pursuit.

AMS subject classifications. 68U10, 65K10, 90C25, 90C51.

1. Introduction. Let $A \in \mathbb{R}^{m\times n}$ for $m < n$ (and sometimes, $m \ll n$ in compressed sensing), $b \in \mathbb{R}^m$, and $x \in \mathbb{R}^m$. The linearized Bregman method is introduced in [**?**] and extended or analyzed in [**?**, **?**, **?**] to approximately solve the basis pursuit problem

$$\min\{\|x\|_1 : Ax = b\}, \tag{1.1}$$

which determines an ℓ_1-minimal solution x_{opt} of the underdetermined linear system $Ax = b$. This problem arises in many applications, and in particular, in the recently emerging application of compressed sensing, which was brought to the forefront by Donoho [**?**] and Candes, Romberg, and Tao [**?**].

The *linearized* Bregman method is a variant of the *original* Bregman method introduced in [**?**, **?**], and both Bregman methods can be applied to (**??**). They are briefly reviewed in subsection **??** below. Previous analysis in [**?**, **?**] shows that the linearized Bregman method generates a sequence of points converging to x_α, the unique solution of $\min\{\|x\|_1 + \frac{1}{2\alpha}\|x\|^2 : Ax = b\}$, where $\|x\| := \|x\|_2$ is the Euclidean norm of x (in the view of objective function smoothing, it is related to the Moreau-Yosida regularization [**?**]). This paper analyzes the primal–dual problem pair

$$\begin{aligned}
\mathbf{P}(\alpha): &\quad \min_x\{\|x\|_1 + \frac{1}{2\alpha}\|x\|^2 : Ax = b\}\\
\mathbf{D}(\alpha): &\quad \min_{y,z}\{-b^\top y + \frac{\alpha}{2}\|A^\top y - z\|^2 : z \in [-1,1]^n\},
\end{aligned}$$

and studies how their solutions vary in terms of α.

1.1. Contributions. The first contribution of this report is an exact regularization property: there exists a finite α_∞ so that whenever $\alpha > \alpha_\infty$, the solution of $\mathbf{P}(\alpha)$ is a solution of (**??**). Similar exact regularization results were introduced by Mangasarian and Meyer [**?**] and studied in [**?**, **?**] in the context of linear programming. In [**?**], Ferris and Mangasarian studied such results for nondifferentiable and strongly convex objective functions. We recently become aware of the work of M.Friedlander and P.Tseng [**?**], which proves the same result for a large class of optimization problems. Specifically, the necessary and sufficient

*This research was supported in part by NSF CAREER Award DMS-07-48839, ONR Grants N00014-08-1-1101, the U.S. Army Research Laboratory and the U. S. Army Research Office grant W911NF-09-1-0383, and an Alfred P. Sloan Research Fellowship.

†Department of Computational and Applied Mathematics, Rice University, 6100 Main Street, MS-134, Houston, Texas, 77005, U.S.A. (wotao.yin@rice.edu).

condition for exact regularization to hold is provided, especially for problems with polyhedral objective functions including the ℓ_1 norm. The exact regularization result of this paper can be obtained by applying their results. However, in the context of the linearized Bregman method, this paper obtains the result by taking a different proof approach based on analyzing $\mathbf{D}(\alpha)$, which leads to results for checking α.

The second contribution is to show that the linearized Bregman iteration is equivalent to the gradient descent iteration applied to the y–minimization problem below. Specifically, in $\mathbf{D}(\alpha)$, the optimal z is given by $z = \text{Proj}_{[-1,1]^n}(A^\top y)$, so z can be eliminated, which reduces $\mathbf{D}(\alpha)$ to

$$\mathbf{D'}(\alpha): \quad \min_y \ -b^\top y + \frac{\alpha}{2}\left\|A^\top y - \text{Proj}_{[-1,1]^n}(A^\top y)\right\|^2.$$

Since the second term poses quadratic penalty to $-e \le A^\top y \le e$, $\mathbf{D'}(\alpha)$ can be viewed as a quadratic penalty problem for the Lagrange dual of (**??**):

$$\min\{-b^\top y : \|A^\top y\|_\infty \le 1\}. \tag{1.2}$$

The above result allows us to apply standard optimization techniques to accelerate gradient descents and obtain much faster convergence. Specifically, $\mathbf{D'}(\alpha)$ has a Lipschitz continuous objective function, on which techniques such as line search, Barzilai-Borwein [**?**], quasi-Newton, L-BFGS [**?**], and nonlinear conjugate gradient methods naturally apply. Numerical simulations were performed to demonstrate an significant improvement in speed and accuracy.

We also show that the solution x_α of $\mathbf{P}(\alpha)$ can be obtained from any solution y_α of $\mathbf{D}(w)$ as

$$x_\alpha := \alpha\,\text{shrink}(A^\top y_\alpha, 1), \tag{1.3}$$

where shrink is the soft–thresholding or shrinkage operator defined component-wise by $\text{shrink}(s_i, \beta) = \text{SGN}(s_i)\max\{|s_i| - \beta, 0\}$. Note that $\text{shrink}(A^\top y, 1) = A^\top y - \text{Proj}_{[-1,1]^n}(A^\top y)$, i.e., shrinkage is built in $\mathbf{D'}(()\alpha)$.

The above results can be extended to more general problems of the form

$$\min\{J(x) : Ax = b\}, \tag{1.4}$$

where J is a piece-wise linear (e.g., ℓ_1-like) regularization function. Similar results may possibly be obtained for J being the nuclear norm of a matrix x, which is used in an optimization problem [**?**] similar to (**??**) for the matrix completion problem.

1.2. Background.

1.2.1. The original Bregman Method. The original Bregman method (formally called the Bregman iterative regularization method) is introduced in [**?**], not for solving a constrained minimization problem like (**??**), but to improve image reconstruction quality in the context of total variation regularization; it has been extended to wavelet-based denoising [**?**], nonlinear inverse scale space in [**?**, **?**] and other papers, and MR imaging in [**?**]. Recently, its usefulness in compressed sensing for solving constrained ℓ_1 and ℓ_1-related minimization problems is studied in [**?**], where its equivalence to the augmented Lagrangian method (the method of multipliers) [**?**, **?**, **?**] is also established. This equivalence in the context of total variation minimization and the split Bregman method is studied in [**?**, **?**].

Let $J(\cdot)$ stand for a convex function. The Bregman distance [**?**] with respect to J between points u and v is defined as

$$D_J^p(u, v) := J(u) - J(v) - \langle p, u - v\rangle \tag{1.5}$$

where $p \in \partial J(v)$, the subdifferential of J at v. Note that because $D_J^p(u,v) \neq D_J^p(v,u)$ in general, $D_J^p(u,v)$ is not a distance in the usual sense. The original Bregman method solves a sequence of convex problems in the iterative scheme

$$x^{k+1} \leftarrow \min_x D_J^{p^k}(x, x^k) + \frac{1}{2}\|Ax - b\|^2 \tag{1.6}$$

for $k = 0, 1, \ldots$ starting from $x^0 = \mathbf{0}$ and $p^0 = \mathbf{0}$. For nondifferentiable J such as $\mu\|\cdot\|_1$ and $\mu TV(\cdot)$, $\partial J(x^{k+1})$ may contain more than one element, leading to many possible choices of p^{k+1}. In (**??**), each p^{k+1} is chosen based on the optimality conditions: $\mathbf{0} \in \partial J(x^{k+1}) - p^k + A^\top(Ax^{k+1} - b)$, which yields $p^{k+1} := p^k + A^\top(b - Ax^{k+1})$.

In [**?**, **?**] three key results for the sequence $\{x^k\}$ are proved. First, $\|Ax^k - b\|$ converges to 0 monotonically and $\lim x^k$ is a solution of $\min\{J(x) : Ax = b\}$; second, for ℓ_1-like functions J, the convergence is finite; third, assuming that b is a *noisy* observation of $A\bar{x}$, where $\bar{x}$ is the *unknown* noiseless signal, x^k monotonically gets closer to $\bar{x}$ in terms of the Bregman distance $D_J^{p^k}(\bar{x}, x^k)$, at least while $\|Ax^k - b\| \geq \|A\bar{x} - b\|$ (note that $\|Ax^k - b\|$ decreases monotonically in k). The first two results were proven previously in the literature of the augmented Lagrangian method.

Interestingly, (**??**) can be interpreted as iteratively "adding back the residual." Since p^k is in the range space of $A^\top$ (assume that $p^k = A^\top v^k$ for a certain v^k), the linear term $\langle p^k, x\rangle = \langle v^k, Ax\rangle$ can merge into $\frac{1}{2}\|Ax - b\|^2$, yielding the equivalent iterative scheme

$$x^{k+1} \leftarrow \min_x J(x) + \frac{1}{2}\|Ax - b^{k+1}\|^2, \text{ where } b^{k+1} = b^k + (b - Ax^k). \tag{1.7}$$

At each iteration, the residual $b - Ax^k$ is added to, rather than subtracted from, b^k. For $J(\cdot) = \mu\|\cdot\|_1$, each iteration of (**??**) solves the so-called basis pursuit denoising problem. Several recent algorithms based on matrix-vector multiplications involving A and $A^\top$ can efficiently solve this problem with large-scale data and sparse solutions. They include iterative soft thresholding (IST) algorithms [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**], GPSR [**?**], SPGL1 [**?**], $\ell_1\_\ell_s$ [**?**], FPC_AS [**?**], Nesterov–type algorithms [**?**, **?**], and others. For finding a solution of (**??**), the iterative scheme (**??**) is preferred over directly solving a single

$$\min_x \ \mu\|x\|_1 + \frac{1}{2}\|Ax - b\|^2 \tag{1.8}$$

because (**??**) needs a tiny μ, which slows down most of the above algorithms[1]. In (**??**), a relatively large μ can be used so that each iteration is cheap while the total number of iterations remains reasonable. For compressed sensing problems with sparse solutions, [**?**] suggests using a moderately large μ and reports that only 2–6 iterations on average will suffice.

The Bregman iteration has been extended and applied to solving various problems. In addition to compressed sensing applications, extensions can be found in [**?**] for deconvolution and sparse reconstruction, [**?**] for image blind deconvolution, [**?**, **?**] for inverse scale space methods, [**?**] for wavelet-based image denoising, [**?**] for the split Bregman method (the "split" part is from in [**?**, **?**, **?**, **?**]) and its applications in [**?**], [**?**] for denoising and partially parallel imaging, and [**?**, **?**] for matrix rank minimization.

1.2.2. The Linearized Bregman Method. The linearized Bregman method [**?**] is obtained by linearizing the last term in (**??**) into $\langle A^\top(Ax^k - b), x\rangle$ and adding the ℓ_2-proximity term $\frac{1}{2\alpha}\|x - x^k\|^2$, yielding the new iterative scheme:

$$x^{k+1} \leftarrow \arg\min_x D_J^{p^k}(x, x^k) + \langle A^\top(Ax^k - b), x\rangle + \frac{1}{2\alpha}\|x - x^k\|^2. \tag{1.9}$$

[1] except for FPC_AS because of its use of subspace optimization.

The components of x are separable in the last two terms of (**??**). Hence, for componentwise separable J such as $\mu\|x\|_1$, (**??**) is very simple to compute. The update formula of p can be derived from the optimality conditions of (**??**):

$$p^{k+1} \leftarrow p^k - A^\top(Ax^k - b) - \frac{1}{\alpha}(x^{k+1} - x^k), \tag{1.10}$$

where $p^{k+1} \in \partial J(x^{k+1})$. The algorithm based on (**??**) and (**??**) is given in Table **??**, in which the fitting term $\frac{1}{2}\|Ax - b\|^2$ has been substituted by a more general convex function $H(x)$.

TABLE 1.1
The linearized Bregman method

Input: $J(\cdot)$, $H(\cdot)$, $\alpha > 0$; optional: x^0 and p^0

1. **Initialize**: $k = 0$, let $x^0 = \mathbf{0}$ and $p^0 = \mathbf{0}$.
2. **while** stopping conditions not satisfied **do**
3. $x^{k+1} \leftarrow \arg\min_x D_J^{p^k}(x, x^k) + \langle \nabla H(x^k), x\rangle + \frac{1}{2\alpha}\|x - x^k\|^2$
4. $p^{k+1} \leftarrow p^k - \nabla H(x^k) - \frac{1}{\alpha}(x^{k+1} - x^k)$

   (If possible, replace Steps 3 and 4 by simpler updates. For solving (**??**), use (**??**) and (**??**).)
5. Optional: apply *kicking* if $x^{k+1} = x^k$
6. $k \leftarrow k + 1$
7. **end while**

For $H(x) = \frac{1}{2}\|Ax - b\|^2$, Steps 3 and 4 in Table **??** can be significantly simplified. First, from (**??**) or Step 4, we get

$$p^{k+1} = p^k - A^\top(Ax^k - b) - \frac{1}{\alpha}(x^{k+1} - x^k) = \cdots = \sum_{i=0}^{k} A^\top(b - Ax^i) - \frac{x^{k+1}}{\alpha}.$$

Then, introduce

$$v^k = p^{k+1} + \frac{x^{k+1}}{\alpha} = p^k - A^\top(Ax^k - b) + \frac{x^k}{\alpha} = \sum_{i=0}^{k} A^\top(b - Ax^i),\ \forall k, \tag{1.11}$$

and simplify (**??**) or Step 3 to

$$\begin{aligned} x^{k+1} &\leftarrow \arg\min_x J(x) - \langle p^k, x\rangle + \langle A^\top(Ax^k - b), x\rangle + \frac{1}{2\alpha}\left\|x - x^k\right\|^2 \\ &\leftarrow \arg\min_x J(x) + \frac{1}{2\alpha}\left\|x - \alpha\left(p^k - A^\top(Ax^k - b) + \frac{x^k}{\alpha}\right)\right\|^2 \\ &\leftarrow \arg\min_x J(x) + \frac{1}{2\alpha}\left\|x - \alpha v^k\right\|^2. \end{aligned} \tag{1.12}$$

Therefore, Steps 3 and 4 are rewritten as

$$x^{k+1} \leftarrow \arg\min_x J(x) + \frac{1}{2\alpha}\left\|x - \alpha v^k\right\|^2, \tag{1.13a}$$

$$v^{k+1} \leftarrow v^k + A^\top(b - Ax^{k+1}), \tag{1.13b}$$

Problem (**??**) can be quickly solved for various choices of $J(x)$ such as $\mu\|x\|_1$, $\mu TV(x)$, $\mu\|\Phi x\|_1$ with a fast transform Φ (an orthonormal basis or tight frame), and more generally, for component-separable regularization terms in the form of $\sum_i \phi(x_i)$; see paragraph 2 of subsection **??**. For $J(\cdot) = \mu\|\cdot\|_1$, the solution of (**??**)

is $\alpha\,\mathrm{shrink}(v^k,\mu)$, so we obtain the simplified iterative scheme [**?**]

$$x^{k+1} \leftarrow \alpha\,\mathrm{shrink}(v^k,\mu), \tag{1.14a}$$

$$v^{k+1} \leftarrow v^k + A^\top(b - Ax^{k+1}). \tag{1.14b}$$

Sometimes (**??**) can stagnate, but the stagnation is easily removed by a technique called *kicking* [**?**]. It can happen that over a sequence of consecutive iterations, the components v_i satisfying $|v_i| > \mu$ stay constant while the remaining components v_i, which satisfy $|v_i| \le \mu$, are (slowly) updated. Until one of the latter components finally violates $|v_i| \le \mu$, x remains unchanged. Kicking detects this stagnation by testing $x^k = x^{k+1}$ and breaks the stagnation by consolidating all the remaining iterations over which x is unchanged.

It is proved in [**?**, **?**] that the linearized Bregman method converges to the solution of

$$\min\left\{\mu\|x\|_1 + \frac{1}{2\alpha}\|x\|^2 : Ax = b\right\}. \tag{1.15}$$

By scaling the objective function, (**??**) can be simplified to $\mathbf{P}(\alpha)$, i.e., μ is removed. The convergence was initially established for convex, continuously differentiable convex functions $J(x)$ in [**?**] (note that both ℓ_1-norm and total variation must be smoothed to qualify). However, the same paper shows that if the convergence for $J(x) = \|x\|_1$ occurs, then the limit is the solution of $\mathbf{P}(\alpha)$. The convergence assumption was later removed in the authors' follow-up paper [**?**], which was drafted around the same time when the first version of this report was written. In addition, it was shown that as $\alpha \to \infty$, the solution of $\mathbf{P}(\alpha)$ converges to one of (**??**). This paper reduces the requirement to $\alpha > \alpha_\infty$ for a certain finite α_∞.

Before ending this subsection, we list some applications of the linearized Bregman method that have appeared in the literature: compressed sensing [**?**, **?**, **?**], the matrix completion problem [**?**], and image deblurring [**?**, **?**, **?**]. Good numerical performance is reported in these papers.

1.3. Organization. The remaining of this paper is organized as follows. In section **??**, the linearized Bregman iteration is shown equivalent to a unit-step gradient descent iteration, from which a global convergence result follows directly. In section **??**, the dual solution set is analyzed, and the exact regularization property is proved. Section **??** presents simulation results. Conclusions and discussions are given in section **??**.

2. Linearized Bregman as Dual Gradient Descent. Let us introduce a smoothed version of J:

$$g_\alpha(x) := J(x) + \frac{1}{2\alpha}\|x\|^2,$$

and let $g_\alpha^*(\cdot)$ denote the Fenchel dual (or convex conjugate) of $g_\alpha(\cdot)$. The Lagrangian dual of $\min_x\{g_\alpha(x) : Ax = b\}$ is

$$\min_y \; -b^\top y + g_\alpha^*(A^\top y). \tag{2.1}$$

Since $g_\alpha(\cdot)$ is strictly convex, $g_\alpha^*(\cdot)$ is differentiable [**?**].

THEOREM 2.1. *The linearized Bregman iteration* (**??**) *is equivalent to the gradient descent iteration applied to problem* (**??**) *with a unit step size.*

Proof. We shall relate the dual variable y^k in (**??**) to the variable v^k in (**??**), and then show that (**??**) is a gradient descent iteration. From (**??**), we have $v^k \in \mathcal{R}(A^\top)$ for all k, so we introduce y^k such that $v^k = A^\top y^k$, and thus (**??**) yields the iteration $y^{k+1} = y^k - (Ax^k - b)$. Next, we show that $(Ax^k - b)$ is a

gradient of the objective function of (**??**) at y^k. Because p^k is a subgradient of $J(\cdot)$ at x^k, we have

$$\begin{aligned} p^k \in \partial_x J(x^k) &\Longleftrightarrow A^\top y^k = p^k + \frac{1}{\alpha} x^k \in \partial_x g_\alpha(x^k) \\ &\Longleftrightarrow x^k \in \nabla g_\alpha^*(A^\top y^k) \\ &\Longrightarrow Ax^k = A\nabla g_\alpha^*(A^\top y^k) = \nabla_y g_\alpha^*(A^\top y^k) \\ &\Longleftrightarrow Ax^k - b = \nabla_y(-b^\top y^k + g_\alpha^*(A^\top y^k)), \end{aligned}$$

where the second line is a well-known property of Fenchel duality (cf. [**?**]). □

Comments: Dual gradient descent is equivalent to a multiplier method[2]. Define the Lagrangian $L(x, y) := g_\alpha(x) + \langle y, b - Ax\rangle$. Then, the linearized Bregman iteration (**??**)–(**??**) can be exactly obtained from the multiplier method

1. $x^{k+1} \leftarrow \min_x L(x, y^k)$,
2. $y^{k+1} \leftarrow y^k + \nabla_y L(x^{k+1}, y^k)$,

and by letting $v^k = A^\top y^k$.

Theorem **??** means that one can apply the general convergence results of gradient descent on the linearized Bregman method. Let us take $J(x) = \|x\|_1$ as an example (the result for $J(x) = \mu\|x\|_1$ is the same), and show that iterative scheme (**??**) generates a sequence $\{x^k\}$ that converges to the solution of $\mathbf{P}(\alpha)$ if $\|A\|^2 < 2/\alpha$.

First, we derive (**??**), which gives x_α. The Lagrangian dual problem of $\mathbf{P}(\alpha)$ is $\mathbf{D'}(\alpha)$. Specifically, corresponding to $J(x) = \|x\|_1$, we get $g_\alpha(x) = \|x\|_1 + \frac{1}{2\alpha}\|x\|^2$ and its Fenchel dual:

$$g_\alpha^*(z) = \sum_{i=1}^n \frac{\alpha}{2}(z_i - \mathrm{Proj}_{[-1,1]}(z_i))^2 = \frac{\alpha}{2}\|z - \mathrm{Proj}_{[-1,1]^n}(z)\|^2.$$

Plugging g_α^* into (**??**) gives the objective function of $\mathbf{D'}(\alpha)$, denoted by

$$F_\alpha(y) := -b^\top y + \frac{\alpha}{2}\|A^\top y - \mathrm{Proj}_{[-1,1]^n}(A^\top y)\|^2. \tag{2.2}$$

Because $\nabla g_\alpha^*(z) = \alpha(z - \mathrm{Proj}_{[-1,1]^n}(z))$, we have

$$\nabla F_\alpha(y) = -b + \alpha A\left(A^\top y - \mathrm{Proj}_{[-1,1]^n}(A^\top y)\right). \tag{2.3}$$

Hence, the first-order optimality conditions of $\mathbf{D'}(\alpha)$ are $\nabla F_\alpha(y) = 0$ or $\alpha A\left(A^\top y - \mathrm{Proj}_{[-1,1]^n}(A^\top y)\right) = b$. Since $\alpha\left(A^\top y - \mathrm{Proj}_{[-1,1]^n}(A^\top y)\right) = \alpha\mathrm{shrink}(A^\top y_\alpha, 1)$, it is easy to observe that x_α defined in (**??**) satisfies $Ax = b$, i.e, is a feasible solution of $\mathbf{P}(\alpha)$. The optimality of x_α for $\mathbf{P}(\alpha)$ are proved in Theorem **??** below by matching the primal objective value given by x_α to that of the dual given by y_α.

To establish that $\{x^k\}$ generated by (**??**) converges to x_α, all we need to show is that the $\psi(\cdot) := \nabla F_\alpha(\cdot)$ is Lipschitz continuous with the constant $L \le \alpha\|A\|^2$. Then, according to the classical result of gradient descent, $\{x^k\}$ converges under the condition that the step size (which is 1 in our case) is no more than $2/L$, or equivalently, $\|A\|^2 < 2/\alpha$. To show that $\psi(\cdot)$ is Lipschitz continuous, we derive

$$\begin{aligned} \|\psi(y^1) - \psi(y^2)\| &= \|A\left(\nabla g(A^\top y^1) - \nabla g(A^\top y^1)\right)\| \\ &\le \|A\| \cdot \alpha\|(A^\top y^1 - \mathrm{Proj}_{[-1,1]^n}(A^\top y^1)) - (A^\top y^2 - \mathrm{Proj}_{[-1,1]^n}(A^\top y^2))\| \\ &\le \alpha\|A\|\|A^\top(y^1 - y^2)\| \\ &\le \alpha\|A\|^2\|y^1 - y^2\|, \end{aligned}$$

where the second inequality holds because $|(s - \mathrm{Proj}_{[-1,1]}s) - (t - \mathrm{Proj}_{[-1,1]}t)| \le |s - t|$ for any $s, t \in \mathbb{R}$.

[2]The authors of [**?**] pointed out that the linearized Bregman method can be derived from Uzawa's method [**?**], which is a multiplier method motivated by economical equilibria.

2.1. Generalizations of the Linearized Bregman Method. It is natural to improve unit-step gradient descent by methods such as line search, quasi-Newton methods, L-BFGS [**?**], Nesterov's recent algorithm [**?**], and nonlinear conjugate gradient methods, all of which need only gradient computations.

Our purpose is not to detail the above enhancements one by one but to argue that with any of these enhancements, the main computation remains almost as simple as (**??**) and (**??**), or (**??**) and (**??**) for[3] $J(x) = \|x\|_1$, because the gradient of the objective function in (**??**) is simple to compute. At $y = y^k$, the gradient is given by $Ax^k - b$, where x^k is further given by (**??**) in which $v^k = A^\top y^k$. For many choices of $J(x)$, computing gradients remains simple. For $J(x) = \|x\|_1$, we have shown that $x^k = \alpha\left(A^\top y^k - \text{Proj}_{[-1,1]^n}(A^\top y^k)\right)$. For $J(x) = \|\Phi x\|_1$ where Φ is a non-singular transform, one can introduce $\bar{x} := \Phi x$ and thus solve $\min\{\|\bar{x}\|_1 : A\Phi^{-1}\bar{x} = b\}$. Furthermore if Φ is orthonormal, then $\Phi^{-1} = \Phi^\top$ and thus $\bar{x}^k = \alpha\left(\Phi A^\top y^k - \text{Proj}_{[-1,1]^n}(\Phi A^\top y^k)\right)$. For $J(x) = TV(x)$, problem (**??**) is the ROF model, which can quickly solved by many algorithms including the latest graph-cut/max-flow algorithms [**?**, **?**, **?**]. The list of functions $J(x)$ permitting fast solutions is not short.

In section **??** below, we will compare three different implementations of the linearized Bregman method for $J(x) = \|x\|_1$. The first implementation is given in Table **??** with kicking enabled. The other two implementations are based on the algorithm in Table **??** but have additional parts. We add a technique combining kicking and the Barzilai-Borwein step size accompanied by non-monotone line search and obtain the *kicking+BB_line_search* approach. We refer to [**?**, **?**] for details on the Barzilai-Borwein method with non-monotone line search. Recent uses of this method on ℓ_1-minimization can be found in [**?**, **?**, **?**]. Let τ^k denote the step size at iteration k. The iterative scheme of kicking+BB_line_search is based on

$$x^{k+1} \leftarrow \alpha\,\text{shrink}(v^k, \mu), \tag{2.4a}$$

$$v^{k+1} \leftarrow v^k + \tau^k A^\top(b - Ax^{k+1}), \tag{2.4b}$$

where τ^k is a step size, instead of (**??**). The third implementation uses limited memory BFGS (L-BFGS) [**?**], a well-known implementation of quasi-Newton optimization. It is based on the Broyden-Fletcher-Goldfarb-Shanno (BFGS) approximate Hessian update but does not explicitly store either the approximate Hessian or its inverse. Instead, it implicitly applies the approximate Hessian or its inverse that are generated from the last `m` updates of x and $\nabla f(x)$ on the fly, where `m` is generally as small as between 5 and 20. Hence, L-BFGS is particularly suited for large-scale optimization problems. Let the inverse of the approximate Hessian at iteration k be denoted by H^k and its step size by $\bar{\tau}^k$. The corresponding iterative scheme is based on

$$x^{k+1} \leftarrow \alpha\,\text{shrink}(v^k, \mu), \tag{2.5a}$$

$$v^{k+1} \leftarrow v^k + \bar{\tau}^k H^k A^\top(b - Ax^{k+1}). \tag{2.5b}$$

A non-monotone line search can be used to select $\bar{\tau}^k$. Simulation results are reported in section (**??**) below.

3. Exact Regularization. In this section we prove the exact regularization property: there exists a finite scalar $\alpha_\infty > 0$ such that whenever $\alpha > \alpha_\infty$, the solution x_α of $\mathbf{P}(\alpha)$ is also a solution of the basis pursuit problem (**??**). The approach presented below is not concise as it could be (compared to the one in [**?**]), but the steps in the approach help us develop insights and ideas for checking $\alpha > \alpha_\infty$, leading us to the results in subsections **??** and **??**.

First, we introduce necessary definitions and then go over the sketch of the proof. Let Y_α denote the set of solutions of $\mathbf{D'}(\alpha)$ and $y_\alpha \in Y_\alpha$. For the convenience of subsequent analysis, we partition the index set

[3] We work with $\|x\|_1$ instead of $\mu\|x\|_1$ because of the scaling redundancy with both μ and α.

$\{1,\dots,n\}$ into three subsets according to the values of $(A^\top y)_i$, $i=1,\dots,n$. Define

$$q_i^1(y) := \begin{cases} 1, & (A^\top y)_i < -1, \\ 0, & o.w., \end{cases} \quad q_i^2(y) := \begin{cases} 1, & -1 \le (A^\top y)_i \le 1, \\ 0, & o.w., \end{cases} \quad q_i^3(y) := \begin{cases} 1, & (A^\top y)_i > 1, \\ 0, & o.w. \end{cases}$$

for $i=1,\dots,n$. Let $Q^j(y) := \mathrm{Diag}(q^j)$ for $j=1,2,3$, which act as "partition" or "selection" matrices. For any i and y, exactly one of $Q^1_{ii}(y)$, $Q^2_{ii}(y)$, and $Q^3_{ii}(y)$ equals 1. The following example illustrates the above definitions:

$$A^\top y = \begin{bmatrix} 2 \\ -4 \\ -1 \\ \frac{1}{2} \end{bmatrix} \begin{matrix} > 1 \\ < -1 \\ \in [-1,1] \\ \in [-1,1] \end{matrix} \implies Q^1(y) = \begin{bmatrix} 0 & & & \\ & 1 & & \\ & & 0 & \\ & & & 0 \end{bmatrix},\ Q^2(y) = \begin{bmatrix} 0 & & & \\ & 0 & & \\ & & 1 & \\ & & & 1 \end{bmatrix},\ Q^3(y) = \begin{bmatrix} 1 & & & \\ & 0 & & \\ & & 0 & \\ & & & 0 \end{bmatrix}.$$

Furthermore, we let $Q(y) := (Q^1(y), Q^2(y), Q^3(y))$.

Proof sketch: The proof analyzes the partitions $Q(y_\alpha)$, which are shown to be uniquely determined by α. For each feasible partition Q, there exist a set of α values such that $Q(y_\alpha) = Q$. Such set is either a singleton or an interval. There are finitely many partitions and thus finitely many corresponding intervals, the union of which covers $(0,\infty)$, so the right most interval is unbounded toward $+\infty$. This right most interval is denoted by I_J and its lower bound is defined as α_∞. All $\alpha \in I_J$ not only give the same partition $Q(y_\alpha)$ but also the same x_α, denoted by x^*. x^* is shown to be optimal for both $\mathbf{P}(\alpha)$, $\alpha \in I_J$, and (**??**) through constructing corresponding dual solutions y_α and y_∞, respectively. Specifically, given any y_α, $\alpha \in I_J$, there exists a vector Δy which gives $y_\beta := y_\alpha + (\beta^{-1} - \alpha^{-1})\Delta y \in Y_\beta$, $\beta > \alpha$, and $y_\infty := y_\alpha - \alpha^{-1}\Delta y \in Y_\infty$. This proof is detailed in subsection **??** below, which is based on the results in next subsection.

3.1. Solutions of P(α) and D'(α). Given a partition Q, A can be correspondingly divided column-wise into three submatrices. For $j=1,2,3$ each, let A^j be submatrix of A formed by the columns i of A that correspond to $(Q^j)_{ii} = 1$, and let $e^j = [1\ 1\ \dots\ 1]^\top$ with the dimension equal to the number of columns of A^j.

DEFINITION 3.1. *We say that y is* consistent *with a given Q if $Q(y) = Q$.*

The following theorem states that it is straightforward to obtain a primal solution from a dual solution.

THEOREM 3.2. *Let $\alpha > 0$. For any solution $y_\alpha \in Y_\alpha$ of **D'**(α), x_α given by (**??**) is the unique solution of **P**(α). In particular, $Ax_\alpha = b$.*

Proof. The proof uses classical convex duality. According to (**??**) and combining (**??**) with the dual optimality conditions $\nabla_y F_\alpha(y_\alpha) = \mathbf{0}$, we obtain $Ax_\alpha = b$, i.e., x_α is feasible. It remains to show that the duality gap vanishes, namely, $b^\top y_\alpha - g_\alpha^*(A^\top y_\alpha) = \|x_\alpha\|_1 + \frac{1}{2\alpha}\|x_\alpha\|_2^2$. Let $p := \mathrm{Proj}_{[-1,1]^n}(A^\top y_\alpha)$. Since $(x_\alpha)_i$ is strictly positive (strictly negative) if and only if p_i equals 1 (-1, respectively), we have $p_i \cdot (x_\alpha)_i = |x_i|$ and thus $p^\top x_\alpha = \|x_\alpha\|_1$. Furthermore, $x_\alpha = \alpha\mathrm{shrink}(A^\top y_\alpha, 1) = \alpha(A^\top y_\alpha - p)$. Hence,

$$\begin{aligned} b^\top y_\alpha - g_\alpha^*(A^\top y_\alpha) &= (Ax_\alpha)^\top y_\alpha - \frac{\alpha}{2}\|A^\top y_\alpha - p\|_2^2 \\ &= (A^\top y_\alpha)^\top x_\alpha - \frac{1}{2\alpha}\|\alpha(A^\top y_\alpha - p)\|_2^2 \\ &= \frac{1}{\alpha}(\alpha(A^\top y_\alpha - p))^\top x_\alpha + p^\top x_\alpha - \frac{1}{2\alpha}\|\alpha(A^\top y_\alpha - p)\|_2^2 \\ &= \frac{1}{\alpha}\|x_\alpha\|_2^2 + \|x_\alpha\|_1 - \frac{1}{2\alpha}\|x_\alpha\|_2^2 \\ &= \|x_\alpha\|_1 + \frac{1}{2\alpha}\|x_\alpha\|_2^2. \end{aligned}$$

Finally, because the objective function of $\mathbf{P}(\alpha)$ is strictly convex, its solution x_α is unique. ∎

Comments: This theorem lets one recover the primal solution x_α from any dual solution $y_\alpha \in Y_\alpha$. In case that y_α is not an exact but *approximate* solution, x_α is not an exact solution of $\mathbf{P}(\alpha)$ either, and the primal feasibility measure $\|Ax_\alpha - b\|_2$ is equal to the first-order dual optimality measure $\|\nabla F_\alpha(y_\alpha)\|$ while, on the other hand, the duality gap given by this pair of x_α and y_α is always zero.

Whether Y_α is a singleton or not, $x_\alpha = \alpha(A^\top y_\alpha - \mathrm{Proj}_{[-1,1]^n}(A^\top y_\alpha))$ is unique, so we have

COROLLARY 3.3. *Let $\alpha > 0$. Both $A^\top y_\alpha - \mathrm{Proj}_{[-1,1]^n}(A^\top y_\alpha)$ and $Q(y_\alpha)$ are constant over $y_\alpha \in Y_\alpha$.*

Hence, $Q(y_\alpha)$ only depends on α, so we introduce the y-independent notation $Q^j_\alpha := Q^j(y_\alpha)$, $j = 1, 2, 3$, $y_\alpha \in Y_\alpha$. We similarly define A^j_α, $j = 1, 2, 3$, as the submatrices of A corresponding to Q^j_α. The corollary below characterizes the dual solution set Y_α and exhibits the roles played by A^j_α, $j = 1, 2, 3$.

COROLLARY 3.4. *Let $y_\alpha \in Y_\alpha$. Then,*

$$Y_\alpha = \left(\{y_\alpha\} + \mathrm{Null}\left(A(Q^1_\alpha + Q^3_\alpha)A^\top\right)\right) \cap \{y : Q(y) = Q_\alpha\},$$

where $\mathrm{Null}\left(A(Q^1_\alpha + Q^3_\alpha)A^\top\right) = \mathrm{Null}\left(A^1_\alpha(A^1_\alpha)^\top + A^3_\alpha(A^3_\alpha)^\top\right)$.

This corollary is easy to prove by noticing

$$\begin{aligned} &\min_y F_\alpha(y) = \\ &\min_y \ -b^\top y + \frac{\alpha}{2}\left\|(A^1_\alpha)^\top y + e^1_\alpha\right\|_2^2 + \frac{\alpha}{2}\left\|(A^3_\alpha)^\top y - e^3_\alpha\right\|_2^2, \end{aligned} \tag{3.1}$$

where e^1_α and e^3_α are vectors of all ones of appropriate dimensions. The above equation only means the two problems have the same optimal objective value but not necessarily the same solution set. A^1_α and A^3_α together determine the optimal value but, generally, not enough to determine Y_α because $y_\alpha \in Y_\alpha$ must be consistent with the partition Q_α, in particular, satisfying $-e^2_\alpha \le A^2_\alpha y_\alpha \le e^2_\alpha$. An exception arises when $A(Q^1_\alpha + Q^3_\alpha)A^\top$ has a full rank because then, the normal equations of (**??**) have a unique solution y_α, which must lie in Y_α and thus satisfy $Q(y) = Q_\alpha$.

Given x and α, it can be costly to test whether x solves $\mathbf{P}(w)$ by computing a dual solution $y_\alpha \in Y_\alpha$ since y_α must satisfy both equality and inequality equations. This is the case for sparse optimization where one computes x_α and then is interested in knowing whether this x_α is optimal to (**??**). Fortunately, alternative means exist for highly sparse solutions; see the discussions in subsection **??** below.

3.2. The Exact Regularization Proof. In this subsection, through analyzing the point set

$$I(\alpha) = \{\beta > 0 : Q_\beta = Q_\alpha\} \subset \mathbb{R}, \tag{3.2}$$

we show that for α sufficiently large, the solution x_α of $\mathbf{P}(\alpha)$ is also a solution of (**??**).

LEMMA 3.5. *Let $\alpha > 0$. $I(\alpha)$ is nonempty and connected, so it is either a singleton or an interval (possibly unbounded).*

Proof. Since $\alpha \in I(\alpha)$, $I(\alpha)$ is nonempty. It remains to show that for $\alpha_1, \alpha_2 \in I(\alpha)$ and $\gamma \in (0, 1)$, $\beta := \gamma\alpha^1 + (1-\gamma)\alpha^2 \in I(\alpha)$. From $\alpha_1, \alpha_2 \in I(\alpha)$, it follows that $A^1_\alpha = A^1_{\alpha_1} = A^1_{\alpha_2}$, $A^3_\alpha = A^3_{\alpha_1} = A^3_{\alpha_2}$, and there exist $y_{\alpha_1} \in Y_{\alpha_1}$ and $y_{\alpha_2} \in Y_{\alpha_2}$, which satisfy the optimality conditions of (**??**) in the following form: for $\nu = \alpha_1, \alpha_2$ each

$$\left(A^1_\alpha(A^1_\alpha)^\top + A^3_\alpha(A^3_\alpha)^\top\right) y_\nu = (A^1_\alpha e^1_\alpha - A^3_\alpha e^3_\alpha) + \nu^{-1} b. \tag{3.3}$$

Define $\Delta y := (y_{\alpha_1} - y_{\alpha_2})/(\alpha_1^{-1} - \alpha_2^{-1})$ and $y_\beta := y_{\alpha_2} + (\beta^{-1} - \alpha_2^{-1})\Delta y$. Since y_β is on the line segment connecting y_{α_1} and y_{α_2}, we have $Q(y_\beta) = Q_\alpha$ following from the definition of Q and the assumption $Q_{\alpha_1} = Q_{\alpha_2} = Q_\alpha$. From $Q(y_\beta) = Q_\alpha$ and the fact that y_β satisfies (**??**) for $\nu = \beta$, it is easy to derive that

$\nabla F_\beta(y_\beta) = 0$ and, therefore, both $y_\beta \in Y_\beta$ and $Q(y_\beta) = Q_\beta$. So, we get $Q_\beta = Q_\alpha$ and thus, $\beta \in I(\alpha)$. □
The key of the above proof is the looking for a direction Δy that linearly connects y_{α_1} and y_{α_2} and generating dual solutions y_β for $\beta \in (\alpha_1, \alpha_2)$. The proof of Lemma **??** uses the same technique.

Let $\mathcal{I} = \{I(\alpha) : \alpha > 0\}$. Since there are finitely many distinct Q_a's, $\mathcal{I}$ is a finite set. Since $I \cap I' = \emptyset$ for any two *distinct* $I, I' \in \mathcal{I}$, Lemma **??** lets us order the elements of $\mathcal{I}$ *increasingly* as $I_1, I_2, \ldots, I_j, \ldots, I_J$, where $J < \infty$. Since Q_α does not depend on the choice of $\alpha \in I_j$, we introduce the α–independent notation

$$Q_j := Q_\alpha, \quad \alpha \in I_j,\ j = 1, \ldots, J.$$

Similarly, we define $A^1_j := A^1_\alpha$, $A^2_j := A^2_\alpha$, and $A^3_j := A^3_\alpha$, where $\alpha \in I_j$, for $j = 1, \ldots, J$.

Because $\cup\mathcal{I} = \{\alpha : \alpha > 0\}$ and $\mathcal{I}$ is a finite set, we have

LEMMA 3.6. *$J = |\mathcal{I}|$ is finite and $\sup I_J = +\infty$.*

To proceed we need the following assumption, which leads to the boundedness of $\cup_{\beta \geq \alpha} Y_\beta$ for any $\alpha > 0$ (otherwise, it is easy to show that (**??**) is unbounded, violating the fact that (**??**) is feasible and finite.).

ASSUMPTION 1. *A has full row rank, and $Ax = b$ is consistent.*

When this assumption does not hold, there exists at least one redundant constraint in the system $Ax = b$.

Next, we prove that x_α, $\forall \alpha \in I_J$, is unique and solves problem (**??**) by identifying a corresponding dual solution y_∞ for (**??**). The following lemma proves that given y_α, $\alpha \in I_J$, a set of key equations have a joint solution Δy, from which we construct

$$y_\infty := y_\alpha - \alpha^{-1}\Delta y. \tag{3.4}$$

LEMMA 3.7. *Let $\alpha \in I_J$ and $y_\alpha \in Y_\alpha$. Under Assumption **??**, the following system has a solution Δy:*

$$\left(A^1_\alpha (A^1_\alpha)^\top + A^3_\alpha (A^3_\alpha)^\top\right) \Delta y = b, \tag{3.5a}$$

$$\|A^\top (y_\alpha - \alpha^{-1}\Delta y)\|_\infty \leq 1, \tag{3.5b}$$

$$(A^1_\alpha)^\top (y_\alpha - \alpha^{-1}\Delta y) = -e^1_\alpha, \tag{3.5c}$$

$$(A^3_\alpha)^\top (y_\alpha - \alpha^{-1}\Delta y) = e^3_\alpha. \tag{3.5d}$$

Before proving the lemma, let us describe where (**??**) and (**??**) come from. (**??**) is obtained by taking the limit $\beta \to \infty$ in (**??**) below. Equation (**??**) is a result of (**??**) after varying ν. (**??**) is the feasibility condition. The remaining equations (**??**) and (**??**) follow from (**??**) and (**??**) when $\alpha \in I_J$ and $y_\alpha \in Y_\alpha$, as shown in Theorem **??** below. They are explicitly given in the lemma because when $\alpha \in I_j$ and $\alpha < \alpha_\infty$, they sometimes still hold while (**??**) does not; we can show that for a given j, if (**??**), (**??**), and (**??**) are consistent, then x_α is constant over $\alpha \in I_j$.

Proof. [Lemma **??**] Let $\alpha' > \alpha \in I_J$, and according the proof of Lemma **??**, one can pick $y_{\alpha'} \in Y_{\alpha'}$ satisfying (**??**) with $\nu = \alpha'$. So does $y_\alpha \in Y_\alpha$ with $\nu = \alpha$. By taking the differences between two copies of (**??**) with $\nu = \alpha$ and $\nu = \alpha'$, we get

$$\Delta y_{\alpha'} := \frac{y_{\alpha'} - y_\alpha}{\alpha'^{-1} - \alpha^{-1}}, \tag{3.6}$$

which satisfies (**??**). Hence, the equations in (**??**) are consistent. In addition, we have $y_{\alpha'} = y_\alpha + (\alpha'^{-1} - \alpha^{-1})\Delta y_{\alpha'} \in Y_{\alpha'}$ and $Q(y_{\alpha'}) = Q_J$. Therefore, the set

$$S_{\alpha'} := \{\Delta y : \Delta y \text{ satisfies } (\mathbf{??})\} \cap \{\Delta y : Q(y_\alpha + (\alpha'^{-1} - \alpha^{-1})\Delta y) = Q_J\}$$

contains $\Delta y_{\alpha'}$ and thus is nonempty. Following the argument in the proof of Lemma **??**, one can show that $Q(y_\alpha + (\beta^{-1} - \alpha^{-1}))\Delta y = Q_J$ and thus $\Delta y \in S_\beta$ hold for any $\Delta y \in S_{\alpha'}$ and $\beta \in [\alpha, \alpha']$. This

means $S_{\alpha'}$ is monotonically non-increasing in α'. From Assumption **??**, $T_\alpha := \cup_{\beta\geq\alpha} Y_\beta$ is bounded, so $S_{\alpha'} \subset \{u - v : u, v \in T_\alpha\}$ is bounded. From the theorem of nested sets, there exists $\Delta y \in \cap_{\alpha'>\alpha}\mathrm{cl}(S_{\alpha'})$ satisfying (**??**) and using this Δy, we have

$$y_\beta := y_\alpha + (\beta^{-1} - \alpha^{-1})\Delta y \in Y_\beta, \tag{3.7}$$

for all $\beta \geq \alpha$.

It is a classical result of the quadratic penalty method that in **D'**(α), the penalized terms vanish as the penalty parameter goes to infinity, i.e.,

$$\lim_{\beta\to\infty} \|(A^1_\alpha)^\top y_\beta + e^1_\alpha\| = 0, \ \lim_{\beta\to\infty} \|(A^3_\alpha)^\top y_\beta - e^3_\alpha\| = 0, \quad \forall y_\beta \in Y_\beta. \tag{3.8}$$

Combining (**??**) and (**??**) and letting $\beta \to \infty$, we get (**??**) and (**??**).

Finally, y_β defined in (**??**) is optimal and thus consistent with Q_J for all $\beta \geq \alpha$ and, in particular, satisfy $-e^2_\alpha \leq (A^2_\alpha)^\top y_\beta \leq e^2_\alpha$. Letting $\beta \to \infty$ gives $-e^2_\alpha \leq (A^2_\alpha)^\top(y_\alpha - \alpha^{-1}\Delta y) \leq e^2_\alpha$. From this result, as well as (**??**) and (**??**), (**??**) follows. ∎

It is worth noting that (**??**) can have multiple solutions, not all satisfying (**??**)–(**??**). The whole system of (**??**)–(**??**) can have multiple solutions as well. However, it can be referred from the above Lemma that if (**??**) has a unique solution Δy, Δy must satisfy (**??**)–(**??**) . Next, we prove the main result of this section.

THEOREM 3.8. *x_α is constant for $\alpha \in I_J$, and it is a solution of problem* (**??**).

Proof. Let $\alpha \in I_J$, $\beta \geq \alpha$, and $y_\alpha \in Y_\alpha$, Δy be a solution of (**??**)–(**??**). Define y_∞ and y_β as in (**??**) and (**??**), respectively. We have $y_\infty = y_\beta - \beta^{-1}\Delta y$ and, from (**??**) and (**??**), $Q^1_J(A^\top y_\infty + e) = 0$ and $Q^3_J(A^\top y_\infty - e) = 0$. Therefore,

$$\begin{aligned} x_\beta &= \beta(A^\top y_\beta - \mathrm{Proj}_{[-1,1]^n}(A^\top y_\beta)) \\ &= \beta Q^1_J(A^\top y_\beta + e) + \beta Q^3_J(A^\top y_\beta - e) \\ &= \beta Q^1_J(A^\top y_\infty + e) + \beta Q^3_J(A^\top y_\infty - e) + (Q^1_J + Q^3_J)A^\top \Delta y \\ &= (Q^1_J + Q^3_J)A^\top \Delta y. \end{aligned}$$

Since both $\alpha \in I_J$ and $\beta \geq \alpha$ are arbitrary and x_β is independent of β, the first half of the theorem is proved.

The second half following from the strong duality theorem, which holds given that x_α is primal feasible (i.e., $Ax_\alpha = b$), y_∞ is dual feasible (from (**??**)), and the primal and dual have equal objectives:

$$\|x_\alpha\|_1 = x^\top_\alpha Q^3_J e - x^\top_\alpha Q^1_J e = x^\top_\alpha Q^3_J(A^\top y_\infty) + x^\top_\alpha Q^1_J(A^\top y_\infty) = x^\top_\alpha (Q^1_J + Q^3_J)A^\top y_\infty = x^\top_\alpha A^\top y_\infty = b^\top y_\infty.$$

∎

3.3. An Pathological Example. For a given α, it is generally tricky to test $\alpha \in I_J$ based only on a primal-dual solution pair x_α and $y_\alpha \in Y_\alpha$. One needs to solve (**??**)–(**??**) (in fact, only (**??**) and (**??**) will suffice as is shown below), which include inequality constraints implicitly in (**??**). Is there a simple alternative to avoid the inequalities?

Theorem **??** states that x_α is constant over $\alpha \in I_J$, so one may wonder the sufficiency of this property, namely, if $x_\alpha = x_\beta$ for $\alpha \neq \beta$, then x_α solves (**??**)? This does not hold in the following example.

Let

$$A = \begin{bmatrix} 1 & 1 & 2 \\ 1 & 0 & -2 \end{bmatrix}, \quad b = \begin{bmatrix} 4 \\ 3 \end{bmatrix}.$$

Then, for $\alpha = 1, 2, 3, 4, 8$, as well as ∞ (for problem (**??**)), the primal and dual solutions x_α and y_α of $\mathbf{P}(\alpha)$ are given in the following table:

α	x_α	y_α	$\|x_\alpha\|_1$
1	$[3\ 1\ 0]^\top$	$[2\ 2]^\top$	4
2	$[3\ 1\ 0]^\top$	$[\frac{3}{2}\ 1]^\top$	4
3	$[\frac{65}{21}\ \frac{17}{21}\ \frac{1}{21}]^\top$	$[\frac{80}{63}\ \frac{16}{21}]^\top$	$\frac{83}{21}$
4	$[\frac{67}{21}\ \frac{13}{21}\ \frac{2}{21}]^\top$	$[\frac{97}{84}\ \frac{9}{14}]^\top$	$\frac{82}{21}$
8	$[\frac{7}{2}\ 0\ \frac{1}{4}]^\top$	$[\frac{125}{128}\ \frac{59}{128}]^\top$	$\frac{15}{4}$
∞	$[\frac{7}{2}\ 0\ \frac{1}{4}]^\top$	$[\frac{3}{4}\ \frac{1}{4}]^\top$	$\frac{15}{4}$

For $\alpha = 1, 2$, the primal solution x_α of $\mathbf{P}(\alpha)$ remain the same but is not optimal to (**??**). Therefore, one cannot conclude the optimality of x_α only because it is constant over an interval of α. Suppose $\alpha, \alpha' \in I_j \in \mathcal{I}$ and $\alpha \neq \alpha'$. From the proof of Theorem **??**, it is easy to see when equations (**??**), (**??**) and (**??**) hold for $\Delta y := \frac{y_{\alpha'} - y_\alpha}{\alpha'^{-1} - \alpha^{-1}}$ for $y_\alpha \in Y_\alpha$ and $y_{\alpha'} \in Y_{\alpha'}$, then $x_\alpha = x_{\alpha'}$ and it is unique over $\alpha \in I_j$. Therefore, condition (**??**) is indispensable for $\alpha \in I_J$.

The above example also demonstrates that x_α can vary over α lying in the same interval I_j. x_α for $\alpha = 3$ and $\alpha = 4$ have the same signs, so 3 and 4 belong to the same interval I_j. However, $x_3 \neq x_4$.

Since minimizing $\|x\|_1$ and $\|x\|_2$ tend to yield sparse and non-sparse solutions, respectively, it is natural to conjecture that the solution x_α of $\mathbf{P}(0)$ becomes monotonically sparser as α increases. However, in the above example x_α has more nonzero entries for $\alpha = 4$ than $\alpha = 1$ or 2, so the number of nonzero entries in x_α are generally not monotonic in α.

Finally, exact regularization holds for $\alpha = 8$ since $x_8 = x_\infty$.

3.4. Recognize $\alpha \in I_J$. As stated in the following theorem, in order to verify $\alpha \in I_J$, one generally needs to solve (**??**) and (**??**).

THEOREM 3.9. *$\alpha \in I_J$ if and only if* (**??**) *and* (**??**) *have a solution.*

The proof of this theorem is given in Appendix **??**.

COROLLARY 3.10. *Equations* (**??**) *and* (**??**) *are implied by* (**??**) *and* (**??**).

Next we study the special cases in which $\alpha \in I_J$ or $\alpha \notin I_J$ can be determined without fully solving (**??**) or (**??**).

Case 1. If (**??**) has a unique solution or, equivalent, the matrix $[A^1_\alpha\ A^3_\alpha]$ has full row rank, then one can solve (**??**) and test its solution against (**??**). If (**??**) is satisfied, then $\alpha \in I_J$ and x_α is optimal to (**??**); otherwise, $\alpha \notin I_J$ and x_α is not optimal.

Case 2. For sparse optimization, there are results stating that there exists a number M depending on A such that any x satisfying $Ax = b$ and $\|x\|_0 \leq M$ is the sparsest representation. Similar results exist for compressed sensing problems with sparse solutions. See papers [**?**, **?**, **?**, **?**, **?**, **?**, **?**] and references therein. For compressed sensing, cross validation [**?**, **?**] can also be applied.

Case 3. If two solutions x_α and $x_{\alpha'}$, $\alpha \neq \alpha'$, have the same signs but different values, then we can conclude neither α nor α' is in I_J.

Case 4. Assume that (**??**) has a unique solution. Then, the solution has no more than m nonzeros. Consequently, if x_α has more than m nonzeros, then $\alpha \notin I_J$.

In a compressed sensing problem where the entries of A independently are drawn randomly, the expected solution is almost always either highly sparse or has exactly m nonzero elements. In the former situation, Case 3 applies. In the latter situation, $[A^1_\alpha\ A^3_\alpha]$ often has full rank so Case 1 applies. Therefore, it is often straightforward to test the optimality for a compressed sensing problem.

(d)
Test
12

Test	1	2	3
Dim.	1024	1024	1024
#.Meas.	512	307	307
Sparsity	102	31	61
Meas.Mtrx.	Gassian/QR	Gaussian	Bernoulli
Nonzero	± 1	Gaussian	Gaussian
α	1	5	5

(c)
Test
3

(d) Summary

FIG. 4.1. *Kicking v.s. kicking+BB_line_search v.s. L-BFGS: absolute errors in 2-norm v.s. iterations .*

4. Numerical Simulation. In this section, we report numerical results to demonstrate the effectiveness of two implementations of the linearized Bregman algorithm: one using Barzilai-Borwein steps with non-monotone line search and the other using L-BFGS. The results also illustrate that exact regularization is easily satisfied for a moderate α, at least for the tested problems. We refer the reader to [**?**] for a series of numerical simulations that study the efficiency and robustness of the linearized Bregman algorithm (using the basic implementation with kicking; see subsection **??**) on various sparse optimization problems.

In Figure **??**, we present comparison results of three different implementations of the linearized Bregman method applied to $J(x) = \|x\|_1$: (i) kicking–only, (ii) kicking+BB_line_search, and (iii) L-BFGS[4]. The kicking–only implementation is described in Table **??**. Kicking+BB_line_search and L-BFGS implementations are based on the iterative schemes (**??**) and (**??**), respectively.

Figure **??** depicts absolute errors in 2-norm versus the number of iterations, corresponding to the three tests described in subfigure (d). For kicking and kicking+BB_line_search, exactly two matrix–vector multiplications, one involving A and the other involving $A^\top$, are performed at each iteration; however, the L-BFGS implementation can perform more than one pair of such multiplications at each iteration. For fairness of comparison, we count each pair of A and $A^\top$ multiplications as one iteration in all of the three implementations. The comparison results clearly show that standard optimization enhancements can significantly accelerate the linearized Bregman method. It is worth noting that because of the use of non-monotone line search, the objective values of kicking+BB_line_search and L-BFGS do not always decrease.

Table **??** reports the performance of three implementations on a larger set of sparse optimization tests. All test sets used the same type of sensing matrix: orthogonalized Gaussian matrices whose elements were generated from i.i.d. normal distributions and whose rows were orthogonalized by QR decompositions. Although different matrix types lead to varying performance, the relative speed and robustness of the three implementations remain roughly the same across different matrix types. We chose the matrix type above for our test since it is the one used in the recent report [**?**], which compares the kicking-only implementation

[4]Courtesy of Zaiwen Wen for a pure MATLAB implementation of L-BFGS.

TABLE 4.1

Simulation results three implementations using 20 *random instances for each configuration of* $(n, m, \|\bar{u}\|_0)$.

Results of (**KO**)kicking-only, (**KB**)kicking+BB_line_search, (**LB**)L-BFGS										
Stopping tol.		$\|Au^k - b\|/\|b\| < 10^{-5}$; up to 6000 total $A/A^\top$ multiplications								
Signal type		sparse, i.i.d. standard Gaussian								
		# A and $A^\top$ mult's			relative error $\|u^k - \bar{u}\|/\|\bar{u}\|$			time (sec.)		
		KO	**KB**	**LB**	**KO**	**KB**	**LB**	**KO**	**KB**	**LB**
#Dim.	#Meas.	$\|\bar{u}\|_0 = 50$								
1000	300	2410	324	193	2.54e-004	7.71e-006	6.62e-006	0.5	0.1†	0.1†
2000	600	3309	1297	286	3.76e-004	1.58e-005	7.16e-006	7.1	2.9	0.7†
4000	1200	3850	1018	339	7.66e-004	6.11e-006	6.94e-006	31.3	8.4	2.8
#Dim.	#Meas.	$\|\bar{u}\|_0 = 20$								
1000	300	629	102	128	1.12e-005	5.14e-006	3.75e-006	0.1	0.0†	0.0†
2000	600	863	145	152	1.08e-005	5.67e-006	5.46e-006	1.9	0.3†	0.4†
4000	1200	1313	275	215	1.08e-005	5.39e-006	6.43e-006	10.7	2.3	1.8
Signal type		sparse, uniformly random $[-1, 1]$								
		# A and $A^\top$ mult's			relative error $\|u^k - \bar{u}\|/\|\bar{u}\|$			time (sec.)		
		KO	**KB**	**LB**	**KO**	**KB**	**LB**	**KO**	**KB**	**LB**
#Dim.	#Meas.	$\|\bar{u}\|_0 = 50$								
1000	300	2287	343	214	4.33e-004	7.17e-006	6.60e-006	0.5	0.1†	0.1†
2000	600	3346	968	282	8.12e-004	8.40e-006	5.87e-006	7.2	2.2	0.7†
4000	1200	3851	1183	370	9.52e-004	9.74e-006	5.94e-006	31.3	9.9	3.1
#Dim.	#Meas.	$\|\bar{u}\|_0 = 20$								
1000	300	753	119	141	1.06e-005	4.77e-006	4.42e-006	0.2	0.0†	0.0†
2000	600	903	269	167	1.10e-005	6.33e-006	4.82e-006	2.0	0.6†	0.4†
4000	1200	1395	435	257	1.09e-005	5.83e-006	5.42e-006	11.4	3.7	2.2

†: A sub-second timing result may be inaccurate. A more reliable indicator of the computing cost is the number of A and $A^\top$ multiplications.

to various other ℓ_1 codes. Therefore, the reader can easily infer how efficient the two novel implementations are compared to those ℓ_1 codes tested in [**?**]. The tested sparse signals $\bar{u}$ had numbers of nonzeros equal to either 20 on 50 depending on test sets. The positions of the nonzero entries of $\bar{u}$ were selected uniformly at random, and each nonzero value was sampled either from standard Gaussian (`randn` in MATLAB) or from $[-1, 1]$ uniformly at random (`2*rand-1` in MATLAB) depending on test sets. No noise was added to either $\bar{u}$ or the measurements $A\bar{u}$. We set $\alpha = 5$ uniformly for all tests for the kicking+BB_line_search and L-BFGS implementations, i.e., they solve $\mathbf{P}(5)$. To ensure convergence, we had to use $\alpha = 1$ and thus $\mu = 5$ (see (**??**)) for the kicking-only implementation so that it also solves $\mathbf{P}(5)$.

The three implementations were written in MATLAB 2009b, and simulations were run on a Lenovo T400s laptop running Windows 7 32-bit with a P9600 CPU and 3GB of memory.

From Table **??** it is easy to see that the L-BFGS implementation was the fastest among the three, and the kicking+BB_line_search implementation was faster than the kicking-only implementation. Under the same stopping rule, L-BFGS required significantly fewer total numbers of matrix-vector multiplications than the

other two while it returned more accurate solutions. Kicking+BB_line_search was not as good but not too far off either. Kicking-only was the slowest and also returned solutions with the worst mean relative errors. The large mean relative errors were caused by at least a couple tests in each set of 20 independent tests that reached the 6000-multiplication limit and thus were terminated before achieving $\|Au^k - b\|/\|b\| < 10^{-5}$. We have compared the three tested implementations on other types of sensing matrices and sparse signals and arrived at the same conclusion.

For $\alpha = 5$, exact regularization holds for all the tested problems. This can be seen from the low relative errors of $O(10^{-6})$ achieved by the L-BFGS implementation. Increasing α will maintain exact regularization but make the three implementation to take more iterations.

5. Conclusions and Discussions. One of the main results of this paper is the exact regularization property, which implies that to solve the basis pursuit problem (**??**), one can choose to solve the simpler unconstrained problem $\mathbf{P}(\alpha)$ with α greater than a certain threshold using, for example, the fast implementations of the linearized Bregman method. In many applications, a moderate α such as 10 is large enough. Generally, however, it is tricky to choose α because too large an α will slow down the linearized Bregman method. This leaves us the following questions: how to choose α and how to check if it is large enough.

In papers [**?**, **?**, **?**, **?**], the authors have demonstrated good numerical results with relatively small α values in their tested compressed sensing and matrix completion problems. Their empirical choices of α worked fine. For problems with sparse solutions (or low-rank solutions in the matrix completion problem), simple posterior optimality tests are available. For non-degenerate, non-sparse solutions, solving the linear system (**??**) seems unavoidable. In the worst case with degenerate yet non-sparse solutions, both (**??**) and (**??**) need to be solved. A forthcoming report will introduce an alternative algorithm, related to but not the same as $\mathbf{P}(\alpha)$, which works for any $\alpha > 0$ and returns a solution of (**??**).

Acknowledgements. The author wants to thank Donald Goldfarb (Columbia), Wenye Ma (UCLA), Yangyang Xu (Chinese Academy of Sciences), and Zaiwen Wen (UCLA and Rice) for proofreading this paper and Michael Friedlander and Defeng Sun for contributing important references. The comments from the associate editor and two anonymous referees helped the author improve this paper significantly.

Appendix A. Proof of Theorem ??.

Proof. The "only if" part is shown in Lemma **??**.

We show the "if" part by contradiction. Let $\alpha \in I_j$. Suppose (**??**) and (**??**) hold for Δy but $j \neq J$.

First, we show that some equation in (**??**) or (**??**) must be violated by contradiction (to the assumption $j \neq J$). Suppose that all equations in (**??**) and (**??**) hold for Δy. Then, we know $y_\beta = y_\alpha + (\beta^{-1} - \alpha^{-1})\Delta y \in Y_\beta$ from (**??**) in the proof of Lemma **??**. From (**??**), (**??**), and the fact $(A^1_\alpha)^\top y_\alpha < -e^1_\alpha$ and $(A^3_\alpha)^\top y_\alpha > e^3_\alpha$, we have

$$(A^1_\alpha)^\top(y_\alpha + (\beta^{-1} - \alpha^{-1})\Delta y) < -e^1_\alpha \text{ and } (A^3_\alpha)^\top(y_\alpha + (\beta^{-1} - \alpha^{-1})\Delta y) > e^3_\alpha, \quad \forall \beta \geq \alpha. \tag{A.1}$$

Recalling the definition of A^2_α, we have $-e^2_\alpha \leq (A^2_\alpha)^\top y_\alpha \leq e^2_\alpha$, and from (**??**), we also have

$$-e^2_\alpha \leq (A^2_\alpha)^\top(y_\alpha + (\beta^{-1} - \alpha^{-1})\Delta y) \leq e^2_\alpha, \quad \forall \beta \geq \alpha. \tag{A.2}$$

From (**??**) and (**??**), we conclude that $y_\beta = y_\alpha + (\beta^{-1} - \alpha^{-1})\Delta y$ is compatible with Q_α for all $\beta > \alpha$. From this and $\alpha \in I_j$, it follows that $I_j = I_J$, which contradicts $j \neq J$.

Let V^1 and V^3 denote the index sets of violated equations in (**??**) and (**??**), respectively. Therefore, we

have shown $V^1 \cup V^3 \neq \emptyset$. Together with (**??**), we have

$$-1 \leq A_i^\top (y_\alpha - \alpha^{-1}\Delta y) < 1, i \in V^1, \tag{A.3}$$

$$-1 < A_i^\top (y_\alpha - \alpha^{-1}\Delta y) \leq 1, i \in V^3. \tag{A.4}$$

Second, we show that there exists a vector $z \neq \mathbf{0}$ such that $Az = \mathbf{0}$ and

$$z_i \begin{cases} < 0, & i \in V^1, \\ = 0, & i \in (V^1 \cup V^3)^C, \\ > 0, & i \in V^3. \end{cases} \tag{A.5}$$

From (**??**) and (**??**), it is easy to derive

$$\left(A_\alpha^1 (A_\alpha^1)^\top + A_\alpha^3 (A_\alpha^3)^\top\right)(y_\alpha - \alpha^{-1}\Delta y) = A_\alpha^3 e_\alpha^3 - A_\alpha^1 e_\alpha^1,$$

or using the convention $A = [A_\alpha^1 \ A_\alpha^2 \ A_\alpha^3]$,

$$A \begin{bmatrix} -e_\alpha^1 - (A_\alpha^1)^\top (y_\alpha - \alpha^{-1}\Delta y) \\ \mathbf{0} \\ e_\alpha^3 - (A_\alpha^3)^\top (y_\alpha - \alpha^{-1}\Delta y) \end{bmatrix} = \mathbf{0}.$$

Let z be the vector in the brackets. We know that the entries in $(A_\alpha^1)^\top (y_\alpha - \alpha^{-1}\Delta y)$ equal -1 except those in V^1 and the entries in $(A_\alpha^3)^\top (y_\alpha - \alpha^{-1}\Delta y)$ equal 1 except those in V^3. From (**??**) and (**??**), we obtain (**??**).

Finally, we show that x_α defined in Theorem **??**, which assumed to be optimal to $\mathbf{P}(\alpha)$ as $\alpha \in J$, is however not optimal. According to the definition of V^1 and V^3, we have $(x_\alpha)_i < 0$, $i \in V^1$, and $(x_\alpha)_i > 0$, $i \in V^3$. From (**??**), there exists a small scalar $\rho > 0$ such that $x_\alpha - \rho z$ yields a strictly smaller objective of $\mathbf{P}(\alpha)$. Moreover, since $Ax_\alpha = b$ and $Az = 0$, we have $A(x_\alpha - \rho z) = b$. Hence, $x_\alpha - \rho z$ is a better solution than x_α, meaning that x_α is not optimal. This contradicts to the optimality of x_α. □