Numer Algor (2015) 69:819–838
DOI 10.1007/s11075-014-9927-8

ORIGINAL PAPER

A Barzilai-Borwein type method for minimizing composite functions

Yakui Huang · Hongwei Liu

Received: 27 May 2014 / Accepted: 2 October 2014 / Published online: 17 October 2014
© Springer Science+Business Media New York 2014

Abstract In this paper, we propose a Barzilai-Borwein (BB) type method for minimizing the sum of a smooth function and a convex but possibly nonsmooth function. At each iteration, our method minimizes an approximation function of the objective and takes the difference between the minimizer and the current iteration as the search direction. A nonmonotone strategy is employed for determining the step length to accelerate the convergence process. We show convergence of our method to a stationary point for nonconvex functions. Consequently, when the objective function is convex, the proposed method converges to a global solution. We establish sublinear convergence of our method when the objective function is convex. Moreover, when the objective function is strongly convex the convergence rate is R-linear. Preliminary numerical experiments show that the proposed method is promising.

Keywords Barzilai-Borwein method · Linear convergence · Nonmonotone · ℓ_2-ℓ_1 minimization

1 Introduction

In this paper we consider the following minimization problem

$$\min_{x\in\mathbb{R}^n} \ \phi(x) := f(x) + h(x), \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a smooth function with a Lipschitz constant $L > 0$ for the gradient:

$$\|\nabla f(x) - \nabla f(y)\| \le L\|x - y\|, \ \forall\, x, y \in \mathbb{R}^n,$$

Y. Huang (✉) · H. Liu
School of Mathematics and Statistics, Xidian University, Xi'an 710126, People's Republic of China
e-mail: huangyakui2006@gmail.com

and $h : \mathbb{R}^n \to \mathbb{R}$ usually called the regularizer function is continuous convex but may be nonsmooth.

Problems of the form (1) can be found in many important applications. For example, the ℓ_2-ℓ_1 problem [15] arising in sparse reconstruction:

$$\min_{x \in \mathbb{R}^n} \frac{1}{2}\|Ax - b\|_2^2 + \tau \|x\|_1, \tag{2}$$

where $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, and $\tau > 0$. It is well known that the sparsest solution of underdetermined linear system $Ax = b$ can be obtained by (2) under suitable conditions [10–12, 22].

Algorithms for solving problems of the form (1) have been studied extensively in recent literature. To name a few of them, the authors of [34] proposed the interior-point algorithm ℓ_1-ℓ_s for (2). Hale et al. [31] developed the fixed point continuation (FPC) method to solve (2) and showed their method converges linearly under proper conditions on the step length. Nesterov [37], and Beck and Teboulle [2] proposed the iterative shrinkage thresholding algorithm (ISTA) and fast iterative shrinkage thresholding algorithm (FISTA) independently. They proved that the number of iterations required by ISTA and FISTA to get an ϵ-optimal solution to problem (1) are respectively $O(1/\epsilon)$ and $O(1/\epsilon^{1/2})$. Bredies and Lorenz [9] established the linear convergence of the ISTA for solving linear operator equations in infinite dimensional Hilbert spaces. However, the aforementioned methods suffer from one or two of the following problems: require a given Lipschitz constant L or adaptively estimate it at the cost of extra gradient computations; limit the step length to be smaller than a value associated with L during the iterative process. Gonzaga and Karas [26] proposed an algorithm that eliminates the usage of L by an inexact line search and designed an adaptive procedure to estimate a strong convexity constant for the function. Wright et al. [41] developed the sparse reconstruction by separable approximation (SpaRSA) algorithm which uses the Barzilai-Borwein (BB) stepsize [1] with safeguards combined with a nonmonotone line search strategy. The combination of the BB stepsize with safeguards and a nonmonotone line search strategy was originally introduced by Raydan in [38] for unconstrained optimization, and by Birgin et al. in [6] for convex constrained optimization; see also [7] for a complete convergence analysis. Recently, Hager et al. [30] showed that SpaRSA converges sublinearly for general convex functions and the rate is R-linear when ϕ is strongly convex. For more methods, see [3, 16, 23, 35, 40, 42] and references therein.

In this paper, we propose a new method independent of the Lipschitz constant L for solving problems of the form (1). Particularly, at each iteration, our method minimizes an approximation function of ϕ and takes the difference between the minimizer and the current iteration as the search direction. The nonmonotone strategy in [27] is employed for determining the step length to accelerate the convergence process. We prove that all accumulation points of the sequence generated by our method are stationary points, and therefore global solutions of (1) when the objective function is convex. Moreover, if ϕ is convex, then the proposed method converges sublinearly; if ϕ is strongly convex, then the rate of convergence is R-linear.

The rest of the paper is organized as follows. In Section 2, we present our method for solving (1) formally. In Section 3, we prove global convergence result as well as the rate of convergence of our method. Finally, we present some preliminary numerical results for problems of the form (1) in Section 4.

Notations. Throughout this paper $\langle x, y\rangle = x^T y$ denotes the inner product of two vectors $x, y \in \mathbb{R}^n$. $\|\cdot\|_p$ denotes the standard ℓ_p norm. $\|\cdot\|$ denotes the Euclidean norm. $\nabla f(x)$ denotes the gradient of $f(x)$.

2 Algorithm

In this section, we present our method for solving problems of the form (1).

Our approach updates the iterate by

$$x^{k+1} = x^k + \lambda_k d^k, \tag{3}$$

where $\lambda_k \in (0, 1]$ is the step length decided by some search scheme and d^k is the search direction. Since ϕ is possibly nonsmooth, we cannot take the gradient direction as the search direction. Motivated by the SpaRSA, we calculate the search direction by making use of the minimizer of an approximation function of $\phi(x)$. More precisely, the search direction d^k is given by

$$d^k = p_{\alpha_k}\left(x^k\right) - x^k, \tag{4}$$

where $\alpha_k > 0$ and $p_{\alpha_k}(x^k)$ is a minimizer of

$$\min_z \; Q_{\alpha_k}\left(z, x^k\right) := f\left(x^k\right) + \left\langle z - x^k, \nabla f(x^k)\right\rangle + \frac{\alpha_k}{2}\|z - x^k\|^2 + h(z), \tag{5}$$

In the next section, we will see that d^k is a decent direction.

Notice that h is convex, then $Q_{\alpha_k}(\cdot, x^k)$ is strongly convex. Therefore, problem (5) has a unique minimizer. In addition, by the definition of Q in (5), we have

$$\begin{aligned}
& \min_z \; Q_{\alpha_k}\left(z, x^k\right) \\
\Leftrightarrow \quad & \min_z \; \frac{1}{2\alpha_k}\left\|\nabla f\left(x^k\right)\right\|^2 + \left\langle z - x^k, \nabla f(x^k)\right\rangle + \frac{\alpha_k}{2}\|z - x^k\|^2 + h(z) \\
\Leftrightarrow \quad & \min_z \; \frac{\alpha_k}{2}\left\|z - \left(x^k - \frac{1}{\alpha_k}\nabla f(x^k)\right)\right\|^2 + h(z) \\
\Leftrightarrow \quad & \min_z \; \frac{1}{2}\|z - u^k\|^2 + \frac{1}{\alpha_k}h(z),
\end{aligned} \tag{6}$$

where

$$u^k = x^k - \frac{1}{\alpha_k}\nabla f\left(x^k\right).$$

We are especially interested in the case that (5) or (6) can be easily solved. For instance, when $h(x) = \tau\|x\|_1$ for some $\tau > 0$, the unique minimizer $p_{\alpha_k}(x^k)$ of (6) can be obtained by solving the following n problems:

$$\min_{t\in\mathbb{R}} \; \frac{1}{2}|t - u_i^k| + \frac{\tau}{\alpha_k}|t|, \; i = 1, 2, \ldots, n,$$

whose exact minimizer is given by

$$t^* = \begin{cases} u_i^k - \frac{\tau}{\alpha_k}, & \text{if } u_i^k > \frac{\tau}{\alpha_k}, \\ 0, & \text{if } |u_i^k| < \frac{\tau}{\alpha_k}, \\ u_i^k + \frac{\tau}{\alpha_k}, & \text{if } u_i^k < -\frac{\tau}{\alpha_k}, \end{cases} \quad i = 1, 2, \ldots, n.$$

That is,

$$t^* = \text{soft}\left(u_i^k, \frac{\tau}{\alpha_k}\right), \; i = 1, 2, \ldots, n,$$

where for $u \in \mathbb{R}$ and $a \in \mathbb{R}$, $\text{soft}(u, a) = \text{sign}(u) \max\{|u| - a, 0\}$, which is often referred to as the soft-threshold [21] or wavelet shrinkage (see [13], for example) operator. When $h(x) = \tau \|x\|_p^p$, the closed form solution of (6) is known for $p = 4/3, 3/2, 2$ [14, 16].

The setting of α_k will affect the performance of the algorithm significantly. Barzilai and Borwein [1] suggested the following stepsize for gradient methods to solve unconstrained problems:

$$\alpha_+ = \arg\min_{\alpha} \|\alpha s^{k-1} - y^{k-1}\|^2 = \frac{\langle s^{k-1}, y^{k-1}\rangle}{\langle s^{k-1}, s^{k-1}\rangle}, \tag{7}$$

where $s^{k-1} = x^k - x^{k-1}$ and $y^{k-1} = \nabla f(x^k) - \nabla f(x^{k-1})$. Due to its simplicity and efficiency, the BB approach and its variants have received considerable attention, see [6–8, 17, 18, 20, 24, 33] and references therein. Recently, Hale et al. [32] used the BB stepsize to improve the performance of the FPC approach. Figueiredo et al. [23] formulated problem (2) as a quadratic program and employed the spectral projected gradient method introduced in [6], which combines the BB stepsize with the gradient projection strategy and nonmonotone scheme [27], to solve it. Motivated by these successful applications, we use the formula (7) with safeguards to ensure convergence, that is,

$$\alpha_k = \min\{\alpha_{\max}, \max\{\alpha_{\min}, \alpha_+\}\}, \tag{8}$$

where $\alpha_{\max} > \alpha_{\min} > 0$.

As we know, the performance of BB-type methods benefits from the nonmonotone line search techniques [19, 28, 38] in which the objective function is required to be slightly smaller than the largest objective function value in some recent past iterates. Similar as [41], we use the following acceptance criterion to determine the step length λ_k:

$$\phi\left(x^k + \lambda d^k\right) \le \max_{0 \le j \le \min\{k, M-1\}} \phi\left(x^{k-j}\right) - \frac{\gamma}{2}\lambda\alpha_k \|d^k\|^2, \tag{9}$$

where M is a fixed integer and $\gamma \in (0, 1)$ is a constant.

Our method for solving (1) is summarized in Algorithm 1.

Remark 1 Algorithm 1 is closely related to SpaRSA but has two differences in updating iterates: (i) Algorithm 1 solves the subproblem (6) only once while SpaRSA has to solve it for each trial point on each line search; (ii) Algorithm 1 decreases the step length λ_k to meet the acceptance criterion (9) while SpaRSA increases α_k until the solution to the subproblem (6) satisfies an acceptance criterion similar as (9).

Algorithm 1 BB type method for solving (1)

Choose step factor $\rho \in (0, 1)$ and constants $\alpha_{\max} > \alpha_{\min} > 0$;
Initialize iteration counter $k = 1$, $\alpha_1 = 1$; choose initial guess x^1;
repeat
  $\lambda \leftarrow 1$;
  while
    λ does not satisfies (9) **do**
    $\lambda \leftarrow \rho\lambda$;
  end
  $\lambda_k \leftarrow \lambda$;
  Compute x^{k+1} by (3);
  Compute α_{k+1} by (8);
until *stopping criterion is satisfied.*
$k \leftarrow k + 1$;

Let $\partial h(x)$ be the subdifferential at x, the set of vectors $s \in \mathbb{R}^n$ satisfying

$$h(z) \geq h(x) + \langle s, z - x\rangle, \ \forall\, z \in \mathbb{R}^n.$$

By the first-order optimality conditions for (1), we know that a point x^* is a stationary point of (1) if

$$0 \in \nabla f(x^*) + \partial h(x^*).$$

It is not difficult to show that for any given $\alpha > 0$, if $\|p_\alpha(x^*) - x^*\| = 0$, then x^* is a stationary point of (1). Therefore, a direct and simple termination criteria for Algorithm 1 is

$$\|d^k\|_\infty \leq \epsilon, \tag{10}$$

where ϵ the error tolerance. We can also make use of the relative change in objective value at the last step, that is,

$$\left|\phi\left(x^{k+1}\right) - \phi\left(x^k\right)\right| \leq \epsilon\phi\left(x^k\right). \tag{11}$$

For other stopping criteria, see [41].

3 Convergence analysis

3.1 Global convergence analysis

Using the same argument as the one in Lemma 2 of [41], we have the following result.

Lemma 1 *Suppose $\{x^k\}$ is generated by Algorithm 1 and x^* is not a stationary point of (1). Then for any subsequence $\{x^{k_t}\}_{t=1,2,\ldots}$ with $\lim_{t\to\infty} x^{k_t} = x^*$, there is $\epsilon(\alpha_{\max}) > 0$ such that $\|d^{k_t}\| \geq \epsilon(\alpha_{\max})$ for all t sufficiently large.*

Let $l(k)$ be an integer such that $k - \min\{k, M-1\} \leq l(k) \leq k$ and

$$\phi\left(x^{l(k)}\right) = \max_{0 \leq j \leq \min\{k, M-1\}} \phi\left(x^{k-j}\right). \tag{12}$$

Next lemma shows that Algorithm 1 is well defined. In addition, the step length λ_k is bounded away from 0 for $k \geq 1$.

Lemma 2 *Let $\{x^k\}$ be a sequence generated by Algorithm 1. If x^k is not a stationary point of (1), then there exists a constant $\lambda' \in (0, 1]$ such that for any $\lambda \in (0, \lambda']$,*

$$\phi\left(x^k + \lambda d^k\right) \leq \phi\left(x^{l(k)}\right) - \frac{\gamma}{2}\lambda\alpha_k \|d^k\|^2. \tag{13}$$

Moreover, if λ_k satisfies the inequality (13), then

$$\lambda_k \geq \min\left\{1, \frac{\rho(1-\gamma)\alpha_{\min}}{L}\right\} := \bar{\lambda}. \tag{14}$$

Proof By Lipschitz continuity of ∇f and convexity of h, we have

$$\begin{aligned} \phi\left(x^k + \lambda d^k\right) - \phi\left(x^{l(k)}\right) &\leq \phi\left(x^k + \lambda d^k\right) - \phi\left(x^k\right) \\ &\leq f\left(x^k + \lambda d^k\right) + h\left(x^k + \lambda d^k\right) - f\left(x^k\right) - h\left(x^k\right) \\ &\leq \left\langle \nabla f(x^k), \lambda d^k \right\rangle + \frac{\lambda^2 L}{2}\|d^k\|^2 + \lambda\left(h(x^k + d^k) - h(x^k)\right). \end{aligned} \tag{15}$$

Since $p_{\alpha_k}\left(x^k\right) = x^k + d^k$ minimizes $Q_{\alpha_k}\left(z, x^k\right)$, we obtain

$$\begin{aligned} Q_{\alpha_k}\left(x^k + d^k, x^k\right) &= df\left(x^k\right) + \left\langle d^k, \nabla f(x^k)\right\rangle + \frac{\alpha_k}{2}\|d^k\|^2 + h\left(x^k + d^k\right) \\ &\leq Q_{\alpha_k}\left(x^k, x^k\right) = f\left(x^k\right) + h\left(x^k\right). \end{aligned} \tag{16}$$

It follows from (16) that

$$h\left(x^k + d^k\right) - h\left(x^k\right) \leq -\left\langle d^k, \nabla f(x^k)\right\rangle - \frac{\alpha_k}{2}\|d^k\|^2. \tag{17}$$

Combining (15) with (17) gives

$$\phi\left(x^k + \lambda d^k\right) - \phi\left(x^{l(k)}\right) \leq \frac{\lambda(\lambda L - \alpha_k)}{2}\|d^k\|^2. \tag{18}$$

The inequality (13) then follows provided that

$$\frac{\lambda(\lambda L - \alpha_k)}{2}\|d^k\|^2 \leq -\frac{\gamma\lambda\alpha_k}{2}\|d^k\|^2,$$

which by Lemma 1 is satisfied when $\lambda \leq \lambda'$, with $\lambda' = \min\left\{1, \dfrac{(1-\gamma)\alpha_{\min}}{L}\right\}$.

Now we prove the lower bound for λ_k. As we know that either $\lambda_k = 1$ or the inequality (9) will be failed at least once. Therefore,

$$\phi\left(x^k + \frac{\lambda_k}{\rho} d^k\right) > \phi\left(x^{l(k)}\right) - \frac{\gamma}{2}\frac{\lambda_k}{\rho}\alpha_k \|d^k\|^2.$$

Combining this with (18) yields

$$-\frac{\gamma}{2}\alpha_k \|d^k\|^2 < \frac{\frac{\lambda_k}{\rho}L - \alpha_k}{2}\|d^k\|^2.$$

Rearranging terms and using the fact that $\alpha_k \geq \alpha_{\min}$ for all k, we get

$$\frac{\lambda_k}{\rho}L - (1-\gamma)\alpha_{\min} > 0.$$

That is

$$\lambda_k > \frac{\rho(1-\gamma)\alpha_{\min}}{L}.$$

This completes the proof. □

The following theorem implies that Algorithm 1 converges to a global solution of (1) when the objective function is convex.

Theorem 1 *Any accumulation point of the sequence* $\{x^k\}$ *generated by Algorithm 1 is a stationary point of* (1).

Proof By the definition of $\phi(x^{l(k)})$ in (12), we have

$$\begin{aligned}
\phi\left(x^{l(k+1)}\right) &= \max_{0\leq j<\min\{k+1,M\}} \phi\left(x^{k+1-j}\right) \\
&= \max\left\{\max_{1\leq j<\min\{k+1,M\}} \phi(x^{k+1-j}), \phi(x^{k+1})\right\} \\
&\leq \max\left\{\phi(x^{l(k)}), \phi(x^{l(k)}) - \frac{\gamma}{2}\lambda_k\alpha_k\|d^k\|^2\right\} \\
&= \phi\left(x^{l(k)}\right),
\end{aligned} \tag{19}$$

which means that the sequence $\{\phi(x^{l(k)})\}$ is nonincreasing. By applying (19) with k replaced by $l(k)-1$, we get

$$\phi\left(x^{l(k)}\right) \leq \phi\left(x^{l(l(k)-1)}\right) - \frac{\gamma}{2}\lambda_{l(k)-1}\alpha_{l(k)-1}\|d^{l(k)-1}\|^2. \tag{20}$$

Since ϕ is bounded below, taking limits in both sides of (20) we deduce

$$\lim_{k\to\infty} \lambda_{l(k)-1}\alpha_{l(k)-1}\|d^{l(k)-1}\|^2 = 0.$$

Note that $\alpha_k \geq \alpha_{\min}$ for all k, we have

$$\lim_{k\to\infty} \lambda_{l(k)-1} d^{l(k)-1} = 0. \tag{21}$$

Suppose for contradiction that $\bar{x}$ is an accumulation point that is not stationary. Let $\{x^{k_t}\}$ be the subsequence such that

$$\lim_{t\to\infty} x^{k_t} = \bar{x}.$$

By Lemma 1 we have $\|d^{l(k_t)-1}\| \geq \epsilon(\alpha_{\max})$ for some $\epsilon(\alpha_{\max}) > 0$ and all t sufficiently large. Then by (21) we must have

$$\lim_{t\to\infty} \lambda_{l(k_t)-1} = 0.$$

This contradicts Lemma 2. □

3.2 Rate of convergence

By the proof of Theorem 1 we know that the sequence $\{\phi(x^{l(k)})\}$ is nonincreasing. Hence all the iterates generated by Algorithm 1 are contained in the level set

$$\mathcal{L} = \left\{x \in \mathbb{R}^n : \phi(x) \leq \phi(x^1)\right\}.$$

From now on, we assume that the level set $\mathcal{L}$ is bounded, ϕ attains a minimum on $\mathcal{L}$ at point x^* and the associated objective function value $\phi^* = \phi(x^*)$.

We can show sublinear convergence of Algorithm 1 in a similar way as Theorem 3.2 in [30].

Theorem 2 *Let $\{x^k\}$ be a sequence generated by Algorithm 1. If f is convex, then there exists a constant c such that*

$$\phi\left(x^k\right) - \phi^* \leq \frac{c}{k}$$

for all k, where ϕ^* is the optimal objective function value for (1).

Proof By convexity of ϕ and the fact that $0 \leq \lambda_k \leq 1$, we have

$$\phi\left(x^{k+1}\right) = \phi\left(x^k + \lambda_k d^k\right) \leq (1-\lambda_k)\phi\left(x^k\right) + \lambda_k \phi\left(x^k + d^k\right). \tag{22}$$

Using the Lipschitz continuity of f and the definition of Q in (5), it follows that

$$\phi\left(x^k + d^k\right) \leq Q_{\alpha_k}\left(x^k + d^k, x^k\right) + \frac{L}{2}\|d^k\|^2. \tag{23}$$

Since $x^k + d^k$ minimizes $Q_{\alpha_k}(x, x^k)$ and f is convex, we have

$$\begin{aligned} Q_{\alpha_k}\left(x^k + d^k, z^k\right) &= \min_x \left\{ f\left(x^k\right) + \left\langle x - x^k, \nabla f\left(x^k\right)\right\rangle + \frac{\alpha_k}{2}\|x - x^k\|^2 + h(x)\right\} \\ &\leq \min_x \left\{ f(x) + h(x) + \frac{\alpha_k}{2}\|x - x^k\|^2 \right\} \\ &\leq \min_x \left\{ \phi(x) + \frac{\alpha_{\max}}{2}\|x - x^k\|^2 \right\}. \end{aligned} \tag{24}$$

Choose $x = (1-\delta)x^k + \delta x^*$ with $\delta \in [0, 1]$ and x^* be a solution of (1). Again by the convexity of ϕ, we have

$$\begin{aligned}\min_x \left\{\phi(x) + \frac{\alpha_{\max}}{2}\|x - x^k\|^2\right\} &\leq \phi\left((1-\delta)x^k + \delta x^*\right) + \frac{\alpha_{\max}}{2}\delta^2\|x^k - x^*\|^2 \\ &\leq (1-\delta)\phi\left(x^k\right) + \delta\phi^* + \beta_k\delta^2,\end{aligned}$$

where $\beta_k = \dfrac{\alpha_{\max}}{2}\|x^k - x^*\|^2$. Combining this with (22), (23), and (24) yields

$$\begin{aligned}\phi\left(x^{k+1}\right) &\leq (1-\lambda_k)\phi\left(x^k\right) + \lambda_k\Big[(1-\delta)\phi\left(x^k\right) + \delta\phi^* + \beta_k\delta^2 + \frac{L}{2}\|d^k\|^2\Big] \\ &\leq (1-\lambda_k\delta)\phi\left(x^k\right) + \lambda_k(\delta\phi^* + \beta_k\delta^2) + \frac{\lambda_k L}{2}\|d^k\|^2. \qquad (25)\end{aligned}$$

Since both x^k and x^* lie in the level set $\mathcal{L}$, which is assumed to be bounded, we deduce

$$\beta_k = \frac{\alpha_{\max}}{2}\|x^k - x^*\|^2 \leq \frac{\alpha_{\max}}{2}(\text{diameter of } \mathcal{L})^2 := c_1 < +\infty. \qquad (26)$$

The acceptance test (9) implies that

$$\frac{\lambda_k}{2}\|d^k\|^2 \leq \frac{\phi\left(x^{l(k)}\right) - \phi\left(x^{k+1}\right)}{c_2}, \qquad (27)$$

where $c_2 = \gamma\alpha_{\min}$. Submitting (26) and (27) into (25) and using the fact that $\phi(x^k) \leq \phi(x^{l(k)})$, we obtain for $\delta \in [0, 1]$,

$$\phi\left(x^{k+1}\right) \leq \phi\left(x^{l(k)}\right) + \lambda_k\left(\delta\phi^* - \delta\phi\left(x^{l(k)}\right) + c_1\delta^2\right) + c_3\left(\phi\left(x^{l(k)}\right) - \phi\left(x^{k+1}\right)\right), \qquad (28)$$

where $c_3 = \dfrac{L}{c_2}$. The minimum on the right-hand side of (28) is attained at

$$\delta_{\min} = \min\left\{1, \frac{\phi(x^{l(k)}) - \phi^*}{2c_1}\right\}. \qquad (29)$$

Since $\mathcal{L}$ is bounded and $\phi(x^{l(k)})$ is nonincreasing, by Theorem 1 we know that $\phi(x^{l(k)})$ converges to ϕ^*. Thus there exists an integer k_0 such that $\delta_{\min} < 1$ for all $k > k_0$. Consequently, for $k > k_0$,

$$\begin{aligned}\phi\left(x^{k+1}\right) &\leq \phi\left(x^{l(k)}\right) - \frac{\lambda_k\left(\phi(x^{l(k)}) - \phi^*\right)^2}{4c_1} + c_3\left(\phi(x^{l(k)}) - \phi(x^{k+1})\right) \\ &\leq \phi\left(x^{l(k)}\right) - c_4\Big(\phi(x^{l(k)}) - \phi^*\Big)^2 + c_3\left(\phi(x^{l(k)}) - \phi(x^{k+1})\right), \qquad (30)\end{aligned}$$

where $c_4 = \dfrac{\bar{\lambda}}{4c_1}$ with $\bar{\lambda}$ given by (14). Let $r_k = \phi(x^k) - \phi^*$. Subtracting ϕ^* from both sides of (30) gives

$$r_{k+1} \leq r_{l(k)} - c_4 r_{l(k)}^2 + c_3\left(r_{l(k)} - r_{k+1}\right).$$

It follows that

$$r_{k+1} \leq r_{l(k)} - c_5 r_{l(k)}^2,\ k > k_0,$$

where $c_5 = \dfrac{c_4}{1+c_3}$. Using the fact that $r_{k+1} \leq r_{l(k)}$, we have for all $k > k_0$,

$$\frac{1}{r_{l(k)}} \leq \frac{1}{r_{k+1}} - c_5 \frac{r_{l(k)}}{r_{k+1}} \leq \frac{1}{r_{k+1}} - c_5,$$

that is

$$\frac{1}{r_{k+1}} \geq \frac{1}{r_{l(k)}} + c_5. \tag{31}$$

We can find an integer i_0 such that $k_0 \in ((i_0-1)M, i_0 M]$. For all $k \in ((i-1)M, iM]$, $i > i_0$, by the definition of $l(k)$ in (12), we have $k - M + 1 \leq l(k) \leq k$. It follows from (31) that

$$\frac{1}{r_k} \geq \frac{1}{r_{l(k)}} \geq \frac{1}{r_{l(k-M)}} + c_5 \geq \frac{1}{r_{l(k-(i-i_0)M)}} + (i-i_0)c_5,\ i > i_0,$$

that is,

$$r_k \leq \frac{r_{l(k-(i-i_0)M)}}{1+(i-i_0)c_5 r_{l(k-(i-i_0)M)}} \leq \frac{1}{(i-i_0)c_5},\ i > i_0.$$

Now consider these k such that $i > 2i_0$, we have

$$r_k \leq \frac{2}{ic_5} \leq \frac{2M}{c_5 k}.$$

Notice that there are a finite number of $k \in [1, 2i_0 M]$, then we can find a finite $c_6 > \dfrac{2}{c_5}$ for all $k \in [1, 2i_0 M]$. We complete the proof by taking $c = c_6 M$. □

Now we show the R-linear convergence of Algorithm 1 when ϕ is a strongly convex function.

Theorem 3 *Suppose that f is convex and ϕ satisfies*

$$\phi(z) \geq \phi(x^*) + \eta\|z - x^*\|^2, \tag{32}$$

for all $z \in \mathbb{R}^n$, where $\eta > 0$. Let $\{x^k\}$ be a sequence generated by Algorithm 1, then there exist constants $\theta \in (0,1)$ and μ such that

$$\phi\left(x^k\right) - \phi^* \leq \mu\theta^k\left(\phi(x^1) - \phi^*\right) \tag{33}$$

for all k.

Proof We will show that there exists $\nu \in (0,1)$ such that

$$\phi\left(x^{k+1}\right) - \phi^* \leq \nu\left(\phi(x^{l(k)}) - \phi^*\right). \tag{34}$$

Let ω satisfies that

$$0 < \omega < \min\left\{\frac{2}{c_2\bar{\lambda}}, \frac{1}{L}, \frac{\eta}{\alpha_{\max}L}\right\},$$

where $c_2 = \gamma\alpha_{\min}$ is defined in Theorem 2 and $\bar{\lambda}$ is given by (14). We consider two cases.

Case 1 $\|d^k\|^2 \geq \omega\left(\phi\left(x^{l(k)}\right) - \phi^*\right)$. From (25), one has

$$\frac{\phi\left(x^{l(k)}\right) - \phi\left(x^{k+1}\right)}{c_2} \geq \frac{\bar{\lambda}}{2}\|d^k\|^2 \geq \frac{\bar{\lambda}\omega}{2}\left(\phi(x^{l(k)}) - \phi^*\right), \tag{35}$$

Rearranging terms of (35) yields

$$\phi\left(x^{k+1}\right) - \phi^* \leq \left(1 - \frac{c_2\bar{\lambda}\omega}{2}\right)\left(\phi(x^{l(k)}) - \phi^*\right).$$

Case 2 $\|d^k\|^2 < \omega\left(\phi(x^{l(k)}) - \phi^*\right)$. By (32), we have

$$\beta_k = \frac{\alpha_{\max}}{2}\|x^k - x^*\|^2 \leq \frac{\alpha_{\max}}{2\eta}\left(\phi(x^k) - \phi^*\right) \leq c_7\left(\phi(x^{l(k)}) - \phi^*\right),$$

where $c_7 = \dfrac{\alpha_{\max}}{2\eta}$. It follows from (23) that

$$\begin{aligned}\phi\left(x^{k+1}\right) &\leq (1 - \lambda_k\delta)\phi\left(x^{l(k)}\right) + \lambda_k\delta\phi^* + \left(c_7\delta^2 + \frac{\omega L}{2}\right)\lambda_k\left(\phi(x^{l(k)}) - \phi^*\right) \\ &\leq \phi\left(x^{l(k)}\right) + \lambda_k\left(c_7\delta^2 - \delta + \frac{\omega L}{2}\right)\left(\phi(x^{l(k)}) - \phi^*\right),\end{aligned} \tag{36}$$

Subtracting ϕ^* from each side of (36) to obtain

$$\phi\left(x^{k+1}\right) - \phi^* \leq \left[1 + \lambda_k\left(c_7\delta^2 - \delta + \tfrac{\omega L}{2}\right)\right]\left(\phi(x^{l(k)}) - \phi^*\right),$$

for all $\delta \in [0, 1]$. The minimum on the right-hand side is attained at

$$\delta_{\min} = \left\{1, \frac{1}{2c_7}\right\}.$$

If $\delta_{\min} = 1$, then $c_7 \leq \dfrac{1}{2}$. Since $\omega \leq \dfrac{1}{L}$, we have

$$\nu = 1 + \lambda_k\left(c_7 - 1 + \frac{\omega L}{2}\right) \leq 1 - \frac{1 - \omega L}{2}\lambda_k < 1.$$

If $\delta_{\min} < 1$, since $\omega < \dfrac{\eta}{\alpha_{\max} L}$, then

$$\nu = 1 + \lambda_k\left(\frac{1}{4c_7} - \frac{1}{2c_7} + \frac{\omega L}{2}\right) = 1 - \lambda_k\left(\frac{1}{4c_7} - \frac{\omega L}{2}\right) < 1.$$

Thus (34) holds for all $k \geq 1$. Replacing k with $l(k) - 1$ in (34) and using the monotonicity of $\phi(x^{l(k)})$, we obtain

$$\phi\left(x^{l(k)}\right) - \phi^* \leq \nu\left(\phi(x^{l(l(k)-1)}) - \phi^*\right) \leq \nu\left(\phi(x^{l(k-M)}) - \phi^*\right).$$

For any $k \geq 1$, there exists $i \geq 1$ such that $k \in ((i-1)M, iM]$. Applying the above inequality recursively gives

$$\phi\left(x^k\right) - \phi^* \leq \phi\left(x^{l(k)}\right) - \phi^* \leq \nu^{i-1}\left(\phi(x^{l(k-(i-1)M)}) - \phi^*\right).$$

Recalling that $\phi(x^{k+1}) \le \phi(x^{l(k)})$, using $l(k-(i-1)M) \in (0, M]$, we have

$$\phi\left(x^{l(k-(i-1)M)}\right) \le \max\left\{\phi(x^1), \phi(x^{l(1)}), \phi(x^{l(2)}), \ldots, \phi(x^{l(M-1)})\right\}$$
$$\le \max\left\{\phi(x^1), \phi(x^{l(1)})\right\} = \phi(x^1).$$

Therefore,

$$\phi\left(x^k\right) - \phi^* \le \nu^{i-1}\left(\phi(x^1) - \phi^*\right) \le \frac{1}{\nu}\left(\nu^{1/M}\right)^k\left(\phi(x^1) - \phi^*\right),$$

where the last inequality is due to $i \ge \frac{k}{M}$ and the function a^x is decreasing for $0 < a < 1$. We get the expected inequality (33) by taking $\mu = \frac{1}{\nu}$ and $\theta = (\nu^{1/M})^k$. □

4 Computational experiments

In this section, we present numerical experiments comparing several algorithms for solving problems of the form (1). We test these algorithms on ℓ_2-ℓ_1 problems, image deblurring problems, group-separable regularizers, and total variation (TV) regularization problems. All the experiments were carried out on a laptop with an Intel dual Core 2 GHz processor and 2 GB of RAM running Windows 7. Our method was written in MATLAB with the following parameter values:

$$\rho = 0.25,\ M = 5,\ \gamma = 0.01,\ \alpha_{\max} = 10^{30},\ \alpha_{\min} = 10^{-30}.$$

As SpaRSA, we also test the monotone variant of Algorithm 1 by setting

$$M = 0,\ \gamma = 0.0001.$$

The other algorithms were run with default parameters.

In the following tables, "-mono" means we use the monotone variant of the algorithm, "Ax" denotes the number of times that a vector is multiplied by A or A^T, "iter" denotes the number of iterations, "cpu" denotes the CPU time in seconds, and "Obj" is the objective function value.

4.1 ℓ_2-ℓ_1 problems

We compare the performance of Algorithm 1 with that of other recently proposed algorithms for ℓ_2-ℓ_1 problems (2) using the randomly generated data introduced in [23, 34, 41]. We consider the following algorithms: SpaRSA[1] [41], FPC-BB[2] [32], GPSR[3] [23], TwIST[4] [4].

---

[1] Available at http://www.lx.it.pt/~mtf/SpaRSA

[2] Available at http://www.caam.rice.edu/~optimization/L1/fpc

[3] Available at http://www.lx.it.pt/~mtf/GPSR

[4] Available at http://www.lx.it.pt/~bioucas/TwIST/TwIST.htm

Table 1 Average over 10 runs for ℓ_2-ℓ_1 problems without continuation, where $\tau = 0.1\|A^T b\|_\infty$. Value of the objective function is 3.5665 for all methods

Algorithm	Ax	iter	cpu	MSE
FPC-BB	-	37.6	4.34	3.96e-03
TwIST-mono	-	46.2	1.13	3.94e-03
GPRS-Basic	-	35.0	1.71	3.96e-03
GPRS-BB	-	45.2	1.08	3.96e-03
SpaRSA	60.2	29.8	0.74	3.95e-03
SpaRSA-mono	66.0	28.1	0.71	3.95e-03
Algorithm 1	58.8	29.4	0.70	3.95e-03
Algorithm 1-mono	58.8	29.4	0.71	3.95e-03

The matrix A is a random $k \times n$ matrix, with $k = 2^{10}$ and $n = 2^{12}$, the elements of which are chosen from a Gaussian independent and identically distributed with mean zero and variance $1/(2n)$. The observed vector is $b = Ax_{\text{true}} + \mathbf{n}$, where $\mathbf{n}$ is a Gaussian white vector with variance 10^{-4}, and x_{true} is the original signal contains 160 randomly placed ± 1 spikes, with zeros in the other components. The regularization parameter is set to $\tau = 0.1\|A^T b\|_\infty$; notice that for $\tau \geq \|A^T b\|_\infty$ the unique minimum of (2) is the zero vector [25].

To make the comparison independent of the stopping rule for each approach, we first run FPC-BB to set a benchmark objective value, then run the other algorithms until they each reach this benchmark. Table 1 reports the average CPU times (seconds), the number of iterations, the number of matrix-vector multiplications, and the final mean squared error (MSE) of the reconstructions with respect to x_{true} over 10 runs for the algorithms tested. From these results, we can see that all methods give

Table 2 Average over 10 runs for ℓ_2-ℓ_1 problems, where $\tau = 0.001\|A^T b\|_\infty$. Value of the objective function is 0.0439 for all methods

	Without continuation				With continuation			
Algorithm	Ax	iter	cpu	MSE	Ax	iter	cpu	MSE
FPC-BB	-	-	-	-	-	93.2	5.24	6.19e-04
TwIST-mono	-	360.3	7.81	8.27e-04	-	-	-	-
GPRS-Basic	-	2956.5	134.63	8.36e-04	-	267.0	11.30	6.46e-04
GPRS-BB	-	2299.3	54.22	8.36e-04	-	178.4	4.33	5.19e-04
SpaRSA	2503.6	983.9	26.32	8.32e-04	239.1	111.7	2.55	5.09e-04
SpaRSA-mono	3010.0	1020.2	33.75	8.31e-04	277.3	110.4	3.04	5.18e-04
Algorithm 1	1905.4	952.7	20.62	8.31e-04	223.6	107.8	2.40	5.11e-04
Algorithm 1-mono	1994.6	997.3	20.35	8.24e-04	240.8	116.4	2.55	5.17e-04

a same objective function value and a similar value of MSE. Algorithm 1 is slightly faster than SpaRSA and much faster than other methods.

As the practical performance of SpaRSA, GPSR, and other approaches degrades for small values of τ, Hale, Yin, and Zhang [31] introduced the "continuation" technique and integrated it into their fixed-point iteration scheme. We found it helpful for our algorithm to adopt the continuation strategy. We use the same continuation scheme as SpaRSA. For completeness we present the continuation scheme in Algorithm 2. We set $\delta = 0.2$ for our test.

Algorithm 2 Algorithm 1 adaptive continuation [41]

Initialize iteration counter $k = 1$, and choose initial guess x^1.
Set $y^k = y$;
repeat
  $\tau_k = \max\left\{\delta \|A^T y^k\|_\infty, \tau\right\}$, where $\delta < 1$;
  Calculate x^{k+1} by Algorithm 1;
  $y^{k+1} = y - Ax^{k+1}$;
  $k = k + 1$;
until $\tau_k = \tau$;

We compared these approaches with and without continuation on problems generated in a similar way to the former ones with $\tau = 0.001\|A^T b\|_\infty$. From Table 2, we can see that GPSR, SpaRSA, and Algorithm 1 without continuation become slower for this small τ and that continuation yields a significant speed improvement. Algorithm 1 with continuation is slightly faster than SpaRSA, and clearly faster than GPSR, TwIST, and FPC-BB. It takes Algorithm 1 less matrix-vector multiplications and less iterations to reach the same objective function value than SpaRSA.

4.2 Image deblurring problems

In this subsection, we consider three standard benchmark problems summarized in Table 3, all based on the well-known Cameraman image, with 256×256 pixels. These problems have the form (2), where b represents the (vectorized) observed image, and $A = RW$, where R is the matrix representing the blur operator and W represents the inverse orthogonal wavelet transform, with Haar wavelets. The regularization parameter τ set to 0.01, 0.25, and 0.5 for these three problems, respectively. We use the condition (11) with $\epsilon = 10^{-5}$ as the termination criteria.

Table 3 Image deblurring experiments

Experiments	blur kernel	σ^2
1	9×9 uniform	0.56^2
2	$h_{ij} = 1/\left(i^2 + j^2\right)$	2
3	$h_{ij} = 1/\left(i^2 + j^2\right)$	8

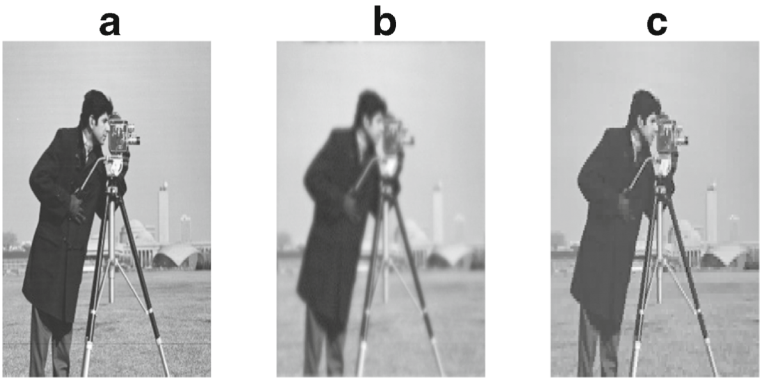

Fig. 1 Deblurring the Cameraman image for problem 2 in Table 3: **a** original image, **b** observed image, **c** deblurred image. The displayed reconstructions were obtained by using Algorithm 1

Since the continuation approaches are no faster, all the methods were implemented without continuation. Table 4 presents numerical results for problems in Table 3. For all three experiments, Algorithm 1 is the fastest method. The original, observed, and deblurred images for problem 2 in Table 3 are presented in Fig. 1.

4.3 Group-separable regularizers

In this subsection, we compare the performance of Algorithm 1 with that of SpaRSA for group-separable regularizers [41] of the form

$$\min_{x\in\mathbb{R}^n} \ \phi(x) = \frac{1}{2}\|Ax-b\|_2^2 + \tau\sum_{i=1}^{m}\|x_{[i]}\|_p,$$

where $p = \{2, \infty\}$, $x_{[1]}, x_{[2]}, \ldots, x_{[m]}$ are m disjoint subvectors of x. The matrix $A \in \mathbb{R}^{1024\times 4096}$ is generated as subsection 4.1. The vector x_{true} has 4096 components

Table 4 Deblurring images without continuation

	1			2			3		
Algorithm	Ax	iter	cpu	Ax	iter	cpu	Ax	iter	cpu
TwIST-mono	-	349	16.64	-	79	4.13	-	72	3.54
GPRS-Basic	-	661	71.52	-	176	19.47	-	121	13.36
GPRS-BB	-	307	16.99	-	154	8.58	-	107	5.78
SpaRSA	562	215	13.18	156	68	3.84	113	51	2.93
SpaRSA-mono	764	242	18.87	205	71	6.03	98	36	2.40
Algorithm 1	396	198	9.40	108	54	2.63	82	41	2.01
Algorithm 1-mono	396	198	9.60	108	54	2.68	82	41	1.97

Table 5 Average over 10 runs for group-separable ℓ_2 and ℓ_∞ Regularizers

	ℓ_2 Regularizer					ℓ_∞ Regularizer				
Algorithm	Ax	iter	cpu	Obj	MSE	Ax	iter	cpu	Obj	MSE
SpaRSA	141.2	63.0	1.69	6.4697	1.52e-03	100.9	47.8	1.87	2.7982	7.95e-05
Algorithm 1	128.0	64.0	1.56	6.4697	1.52e-03	98.8	49.4	1.81	2.7982	7.95e-05

divided into $m = 64$ groups of length $l_i = 64$. When $p = 2$, x_{true} is generated by randomly choosing 8 groups and filling them with zero-mean Gaussian random samples of unit variance, while all other groups are filled with zeros. When $p = \infty$, x_{true} is generated in a similar way to the former case, but filled the chosen 8 groups with ones. The vector $b = Ax_{\text{true}} + \mathbf{n}$, where $\mathbf{n}$ is a Gaussian white vector with mean zero and variance 10^{-4}. Both algorithms are implemented without continuation. We set $\tau = 0.1\|A^T b\|_\infty$ when $p = 2$ and $\tau = 0.5\|A^T b\|_\infty$ when $p = \infty$. We stop the algorithms if the new iterate satisfies (10). We ran 10 test problems with error tolerance $\epsilon = 10^{-5}$ and computed the average results. Table 5 shows that Algorithm 1 solved the test problem in less CPU time and less matrix-vector multiplications than SpaRSA. We present the reconstructions for ℓ_2 and ℓ_∞ regularizers in Figs. 2 and 3, respectively.

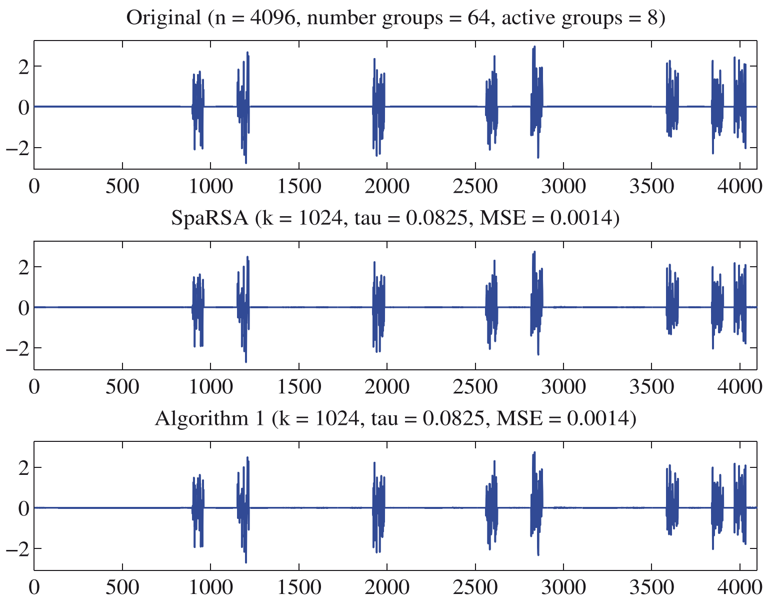

Fig. 2 Comparison of SpaRSA and Algorithm 1 on group-separable reconstruction using ℓ_2 regularizer without continuation

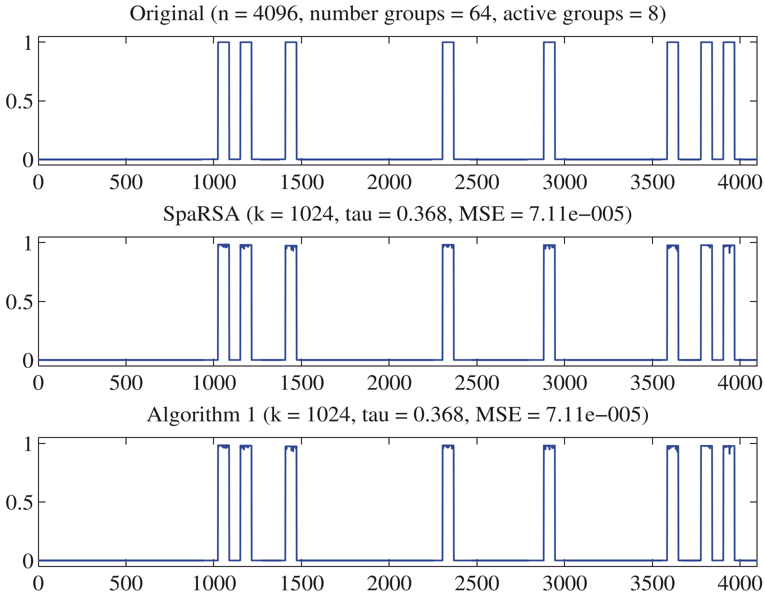

Fig. 3 Comparison of SpaRSA and Algorithm 1 on group-separable reconstruction using ℓ_∞ regularizer without continuation

4.4 Total variation phantom reconstruction

In this subsection, we compare the performance of Algorithm 1 with that of SpaRSA for the 256 × 256 Shepp-Logan phantom image using the total variation (TV) regularization model:

$$\min_{x \in \mathbb{R}^n} \phi(x) = \frac{1}{2}\|Ax - b\|_2^2 + \tau \mathrm{TV}(x),$$

where the definition of TV is given by

$$\mathrm{TV}(x) = \sum_i \sqrt{(\Delta_i^h x)^2 + (\Delta_i^v x)^2},$$

where $\Delta_i^h = x_i = x_{j_i}$ with j_i be the first order neighbor to the left of i and $\Delta_i^v = x_i = x_{k_i}$ with k_i be the first order neighbor above i.

The blur is uniform of size 9×9 and the signal-to-noise ratio of the blurred image ($\mathrm{BSNR} = \mathrm{var}(Ax)/\sigma^2$) is set to 40dB, corresponding to a noise standard deviation of $\sigma = 0.4$ (see [5]). We set $\tau = 0.001$ for this experiment. We first run the monotone version of SpaRSA to set a benchmark objective value, where the condition (11) is used as the termination criteria. Then we run Algorithm 1 and SpaRSA until they each reach the benchmark. For each ϵ, we run 10 trails and report the average results in Table 6. We can see that Algorithm 1 is much faster than SpaRSA, especially for a tight tolerance.

Table 6 Average over 10 runs for TV phantom reconstruction

	1e-3				1e-4				1e-5			
Algorithm	Ax	iter	cpu	Obj	Ax	iter	cpu	Obj	Ax	iter	cpu	Obj
SpaRSA	22.0	9.2	1.23	5.81e3	202.4	79.4	10.71	4.52e3	447.3	165.9	27.72	3.89e3
Algorithm 1	18.4	9.2	0.94	5.81e3	143.6	71.8	6.85	4.54e3	340.2	170.1	17.65	3.89e3

5 Conclusion

We have presented a new Barzilai and Borwein type method to minimize the sum of a smooth function and a convex regularizer. Global convergence result is proved under mild conditions. We established the sublinear and R-linear convergence of our method when the objective function is convex and strongly convex, respectively. In a series of numerical experiments, it is shown that our approach often being faster than SpaRSA, GPSR, FPC-BB, and TwIST. Ongoing work includes incorporate the method with different line search strategy and a more thorough experiments involving wider classes of regularizers. It is also interesting to investigate other schemes concerning the BB stepsize and nonmonotone globalization strategies (see e.g. [29, 36, 39]) for solving problems of the form (1).

Acknowledgments The authors would like to thank the associate editor Professor José Mario Martínez and the anonymous referees for numerous useful suggestions and for having drawn the authors' attention to additional relevant references. This work was supported by the National Natural Science Foundation of China (NNSFC) under Grant No. 61072144 and No. 61179040 and the Fundamental Research Funds for the Central Universities No. K50513100007.

References

1. Barzilai, J., Borwein, J.M.: Two-point step size gradient methods. IMA J. Numer. Anal. **8**(1), 141–148 (1988)
2. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. **2**(1), 183–202 (2009)
3. Becker, S., Fadili, M.J.: A quasi-Newton proximal splitting method. In: Advances in Neural Information Processing Systems, pp. 1–9 (2012)
4. Bioucas-Dias, J., Figueiredo, M.: A new TwIST: two-step iterative shrinkage/thresholding algorithms for image restoration. IEEE Trans. Image Process. **16**(12), 2992–3004 (2007)
5. Bioucas-Dias, J., Figueiredo, M., Oliveira, J.P.: Total variation-based image deconvolution: a majorization-minimization approach. In: IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 861–864 (2006)
6. Birgin, E.G., Martínez, J.M., Raydan, M.: Nonmonotone spectral projected gradient methods on convex sets. SIAM J. Optim. **10**(4), 1196–1211 (2000)
7. Birgin, E.G., Martínez, J.M., Raydan, M.: Inexact spectral projected gradient methods on convex sets. IMA J. Numer. Anal. **23**(4), 539–559 (2003)
8. Birgin, E.G., Martínez, J.M., Raydan, M.: Spectral projected gradient methods: Review and perspectives. http://www.ime.usp.br/~egbirgin/ (2013)
9. Bredies, K., Lorenz, D.A.: Linear convergence of iterative soft-thresholding. J. Fourier Anal. Appl. **14**(5–6), 813–837 (2008)

10. Candès, E., Romberg, J., Tao, T.: Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. IEEE Trans. Inf. Theory **52**(2), 489–509 (2006)
11. Candès, E.J., Romberg, J.: Quantitative robust uncertainty principles and optimally sparse decompositions. Found. Comput. Math. **6**(2), 227–254 (2006)
12. Candès, E.J., Tao, T.: Near-optimal signal recovery from random projections: Universal encoding strategies. IEEE Trans. Inf. Theory **52**(12), 5406–5425 (2006)
13. Chambolle, A., DeVore, R.A., Lee, N.Y., Lucier, B.J.: Nonlinear wavelet image processing: Variational problems, compression, and noise removal through wavelet shrinkage. IEEE Trans. Image Process. **7**(3), 319–335 (1998)
14. Chaux, C., Combettes, P., Pesquet, J.C., Wajs, V.: A variational formulation for frame-based inverse problems. Inverse Probl. **23**(4), 1495 (2007)
15. Chen, S.S., Donoho, D.L., Saunders, M.A.: Atomic decomposition by basis pursuit. SIAM J. Sci. Comput. **20**(1), 33–61 (1998)
16. Combettes, P.L., Wajs, V.R.: Signal recovery by proximal forward-backward splitting. SIAM Multiscale Model. Simul. **4**(4), 1168–1200 (2005)
17. Cores, D., Escalante, R., González-Lima, M., Jimenez, O.: On the use of the spectral projected gradient method for support vector machines. Comput. Appl. Math. **28**(3), 327–364 (2009)
18. Dai, Y.H., Fletcher, R.: Projected Barzilai-Borwein methods for large-scale box-constrained quadratic programming. Numer. Math. **100**(1), 21–47 (2005)
19. Dai, Y.H., Liao, L.Z.: R-linear convergence of the Barzilai and Borwein gradient method. IMA J. Numer. Anal. **22**(1), 1–10 (2002)
20. Dai, Y.H., Zhang, H.: Adaptive two-point stepsize gradient algorithm. Numer. Algor. **27**(4), 377–385 (2001)
21. Donoho, D.L.: De-noising by soft-thresholding. IEEE Trans. Inf. Theory **41**(3), 613–627 (1995)
22. Donoho, D.L.: Compressed sensing. IEEE Trans. Inf. Theory **52**(4), 1289–1306 (2006)
23. Figueiredo, M., Nowak, R., Wright, S.: Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. IEEE J. Sel. Top. Signal Process. **1**(4), 586–597 (2007)
24. Fletcher, R.: On the Barzilai-Borwein method. In: Qi, L.Q., Teo, K.L., Yang, X.Q., Pardalos, P.M., Hearn, D.W., (eds.) Optimization and Control with Applications. Applied Optimization **96**, 235–256 (2005)
25. Fuchs, J.J.: On sparse representations in arbitrary redundant bases. IEEE Trans. Inf. Theory **50**(6), 1341–1344 (2004)
26. Gonzaga, C.C., Karas, E.W.: Fine tuning Nesterov's steepest descent algorithm for differentiable convex programming. Math. Program. **138**(1–2), 141–166 (2013)
27. Grippo, L., Lampariello, F., Lucidi, S.: A nonmonotone line search technique for Newton's method. SIAM J. Numer. Anal. **23**(4), 707–716 (1986)
28. Grippo, L., Sciandrone, M.: Nonmonotone globalization techniques for the Barzilai-Borwein gradient method. Comput. Optim. Appl. **23**(2), 143–169 (2002)
29. Guerrero, J., Raydan, M., Rojas, M.: A hybrid-optimization method for large-scale non-negative full regularization in image restoration. Inverse Probl. Sci. Eng. **21**(5), 741–766 (2013)
30. Hager, W.W., Phan, D.T., Zhang, H.: Gradient-based methods for sparse recovery. SIAM J. Imaging Sci. **4**(1), 146–165 (2011)
31. Hale, E.T., Yin, W., Zhang, Y.: Fixed-point continuation for ℓ_1-minimization: methodology and convergence. SIAM J. Optim. **19**(3), 1107–1130 (2008)
32. Hale, E.T., Yin, W., Zhang, Y.: Fixed-point continuation applied to compressed sensing: implementation and numerical experiments. J. Comput. Math. **28**(2), 170–194 (2010)
33. Huang, Y.K., Liu, H.W., Zhou, S.: A Barzilai-Borwein type method for stochastic linear complementarity problems. Numer. Algor. (2013). doi:10.1007/s11075-013-9803-y
34. Kim, S., Koh, K., Lustig, M., Boyd, S., Gorinevsky, D.: An interior-point method for large-scale ℓ_1-regularized least squares. IEEE J. Sel. Top. Signal Process. **1**(4), 606–617 (2007)
35. Lee, J., Sun, Y., Saunders, M.: Proximal Newton-type methods for convex optimization. In: Advances in Neural Information Processing Systems, pp. 836–844 (2012)
36. Loris, I., Bertero, M., De Mol, C., Zanella, R., Zanni, L.: Accelerating gradient projection methods for ℓ_1-constrained signal recovery by steplength selection rules. Appl. Comput. Harm. Anal. **27**(2), 247–254 (2009)
37. Nesterov, Y.: Gradient methods for minimizing composite functions. Math. Program. **140**(1), 125–161 (2013)

38. Raydan, M.: The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem. SIAM J. Optim. **7**(1), 26–33 (1997)
39. Wang, Y., Ma, S.: Projected Barzilai-Borwein method for large-scale nonnegative image restoration. Inverse Probl. Sci. Eng. **15**(6), 559–583 (2007)
40. Wen, Z., Yin, W., Goldfarb, D., Zhang, Y.: A fast algorithm for sparse reconstruction based on shrinkage, subspace optimization, and continuation. SIAM J. Sci. Comput. **32**(4), 1832–1857 (2010)
41. Wright, S., Nowak, R., Figueiredo, M.: Sparse reconstruction by separable approximation. IEEE Trans. Signal Process. **57**(7), 2479–2493 (2009)
42. Yin, W., Osher, S., Goldfarb, D., Darbon, J.: Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing. SIAM J. Imaging Sci. **1**(1), 143–168 (2008)