# An inexact successive quadratic approximation method for a class of difference-of-convex optimization problems

Tianxiang Liu* Akiko Takeda†

January 29, 2022

**Abstract**

In this paper, we propose a new method for a class of difference-of-convex (DC) optimization problems, whose objective is the sum of a smooth function and a possibly non-prox-friendly DC function. The method sequentially solves subproblems constructed from a quadratic approximation of the smooth function and a linear majorization of the concave part of the DC function. We allow the subproblem to be solved inexactly, and propose a new inexact rule to characterize the inexactness of the approximate solution. For several classical algorithms applied to the subproblem, we derive practical termination criteria so as to obtain solutions satisfying the inexact rule. We also present some convergence results for our method, including the global subsequential convergence and a non-asymptotic complexity analysis. Finally, numerical experiments are conducted to illustrate the efficiency of our method.

## 1 Introduction

In this paper, we consider the following optimization problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \quad F(\boldsymbol{x}) = f(\boldsymbol{x}) + h(\boldsymbol{x}) - g(\boldsymbol{x}), \tag{1.1}$$

where $f$ is a smooth function whose gradient $\nabla f$ is Lipschitz continuous with modulus $L > 0$, $h$ is a proper closed convex function which is *continuous* on its domain and $g$ is a real-valued *continuous* convex function. We assume that $F$ is level-bounded. Unlike many existing models arising from compressed sensing, our focus is on the case in which $h - g$ is *not* necessarily prox-friendly, *i.e.*, one might not always expect that the proximal operator of $h - g$ is easy to compute. On the other hand, $h$ is assumed to be prox-friendly.

Problems of the form (1.1) could arise in applications such as statistics and machine learning, in which $f$ is a data-fidelity loss function and $h - g$ is a difference-of-convex (DC) regularizer for inducing sparsity in the solution; see, for example, [1, 10, 43]. Generally, many commonly used DC regularizers in these application problems, such as $\ell_{1-2}$ [43], smoothly clipped absolute deviation (SCAD) [10] and minimax concave penalty (MCP) [45], are prox-friendly; see [13, 23, 27]. In this case, variants of proximal gradient methods, such as the non-monotone proximal gradient (NPG) method proposed in [41], can be directly applied for solving (1.1). However, when the DC regularizer is taken as, for example, the truncated $\ell_{1-2}$ regularizer proposed in [28], the proximal

*School of Computing, Tokyo Institute of Technology, Tokyo, Japan. Email: (liu.t.af@m.titech.ac.jp).

†Department of Creative Informatics, Graduate School of Information Science and Technology, the University of Tokyo, Tokyo, Japan. Email: (takeda@mist.i.u-tokyo.ac.jp). Center for Advanced Intelligence Project, RIKEN, 1-4-1, Nihonbashi, Chuo-ku, Tokyo 103-0027, Japan. Email: (akiko.takeda@riken.jp).

operator of $h - g$ does not have a closed-form expression. The above formulation can represent a wide class of machine learning problems with regularization.

Another variety of problems of the form (1.1) come from the subproblem of the so-called SDCAM method proposed in [25, Algorithm 1]. SDCAM is an iterative algorithm for a general class of nonconvex nonsmooth optimization problems whose objective is the sum of a smooth function and a finite number of prox-friendly regularization terms. The regularizer $h - g$ in the SDCAM subproblem solved in each iteration is in general not prox-friendly, since $h$ corresponds to some regularization term while $g$ arises from the DC decomposition of the Moreau envelopes of other regularization terms. To handle (1.1) with such a structure, in [25, Algorithm 2] the so-called $\mathrm{NPG}_{\mathrm{major}}$ algorithm was proposed, which combines the NPG method with a linear majorization technique applied to concave term $-g$. Prior to $\mathrm{NPG}_{\mathrm{major}}$, in [12], a proximal DC decomposition algorithm (the so-called proximal DCA) was proposed for solving (1.1). It is equivalent to applying the classical DC algorithm [37] to a specific DC decomposition of $F$. Based on the proximal DCA, in [39] an algorithm named $\mathrm{pDCA}_e$ was proposed to accelerate the proximal DCA under further assumption on the convexity of $f$ and some smoothness of $g$. Later in [24], a refined analysis allows $\mathrm{pDCA}_e$ to include the case when $g$ is possibly nonsmooth, which is the general case for the SDCAM subproblem in [25].

While most existing methods for (1.1) are variants of proximal gradient methods combined with a majorization technique, there are few methods using a second-order approximation of the smooth part $f$. However, for problem (1.1) with $g = 0$, a large quantity of so-called successive quadratic approximation (SQA) methods, in which second-order information of $f$ is used, has been proposed recently; see [5, 6, 9, 14, 16, 29, 31, 36] and see [3, 7, 11, 15, 18, 20, 21, 33, 35, 44] when $f$ is in addition convex. In the $k$th iteration of such an SQA method for (1.1) with $g = 0$, one first computes an (approximate) solution $\hat{\boldsymbol{z}}$ of the problem whose objective approximates the smooth part $f$ quadratically:

$$\min_{\boldsymbol{z}\in\mathbb{R}^n} F_k(\boldsymbol{z}) := \left\langle \nabla f(\boldsymbol{x}^k),\, \boldsymbol{z} - \boldsymbol{x}^k \right\rangle + \frac{1}{2}(\boldsymbol{z} - \boldsymbol{x}^k)^\top \boldsymbol{B}_k(\boldsymbol{z} - \boldsymbol{x}^k) + h(\boldsymbol{z}), \tag{1.2}$$

where $\boldsymbol{B}_k$ is an approximate matrix of the Hessian $\nabla^2 f(\boldsymbol{x}^k)$. Then the next iterate $\boldsymbol{x}^{k+1}$ is obtained after performing a backtracking line search along the direction $\hat{\boldsymbol{z}} - \boldsymbol{x}^k$. Different from the subproblem of proximal gradient methods (in which $\boldsymbol{B}_k$ is a positive multiple of the identity), the minimizer of (1.2) generally does not admit a closed-form solution. Consequently, an iterative method is needed to find an inexact solution of (1.2), and some termination criterion should also be specified to evaluate the quality of the inexact solution. Basically, a typical SQA method has three main ingredients: the choice of the approximate Hessian matrix $\boldsymbol{B}_k$, the solver for the subproblem (1.2) and the inexact rule for the inexact solution of (1.2). For example, in [3], the focus was put on proposing efficient algorithms for solving subproblem (1.2) when $\boldsymbol{B}_k$ has a structure of a positive definite diagonal matrix plus or minus a low-rank term, and a practically efficient inexact rule was proposed in [6] for $L_1$ regularized problems. With proper choices on the three ingredients, the numerical advantage of SQA methods over most first-order methods has been well noticed.

In view of the above, we are motivated to develop a new method for problem (1.1) that makes use of second-order information of $f$. In this paper, we propose an inexact successive quadratic approximation method with majorization ($\mathrm{iSQA}_{\mathrm{major}}$) for solving (1.1), which incorporates a majorization technique in a generic SQA method. Specifically, given $\boldsymbol{x}^k \in \operatorname{dom} h$, we pick $\boldsymbol{\xi}^{k+1} \in \partial g(\boldsymbol{x}^k)$ and an approximate Hessian matrix $\boldsymbol{B}_k$, and then approximately solve the following subproblem:

$$\boldsymbol{u}^k \approx \arg\min_{\boldsymbol{z}} G_k(\boldsymbol{z}) := \left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{z} - \boldsymbol{x}^k \right\rangle + \frac{1}{2}(\boldsymbol{z} - \boldsymbol{x}^k)^\top \boldsymbol{B}_k(\boldsymbol{z} - \boldsymbol{x}^k) + h(\boldsymbol{z}). \tag{1.3}$$

To guarantee sufficient decrease in the objective value, we perform a non-monotone line search along the direction $\boldsymbol{d}_k := \boldsymbol{u}^k - \boldsymbol{x}^k$. Notice that our method is not just a simple extension of existing SQA methods to problems with DC regularizers. In fact, in iSQA$_{\text{major}}$, we propose a new inexact rule for the inexact solution of (1.3), which together with the line search guarantees global subsequential convergence. On the other hand, for several efficient algorithms applied to (1.3), we derive corresponding very cheaply implementable termination criteria each of which outputs an inexact solution satisfying the inexact rule. We list the contributions of this paper as follows:

- We propose a new method (Algorithm 1) for DC problem (1.1), which allows for using second-order information and solving the subproblem inexactly. In Algorithm 1, we give a new inexact rule to measure the inexactness of solutions for the subproblem.
- For Algorithm 1, we provide a global convergence result (Theorem 3.3) and a non-asymptotic complexity analysis (Theorem 3.10), in which the influence of inexact tolerance is quantified.
- We derive practical/cheap termination criteria for several classic algorithms applied to the subproblem so as to obtain inexact solutions satisfying the inexact rule (see Lemma 4.1 and Theorem 4.5). We also conduct numerical experiments to illustrate the efficiency of our method.

The rest of paper is organized as follows. Notation and some preliminaries are given in Section 2. In Section 3, we outline the inexact successive quadratic approximation method (iSQA$_{\text{major}}$), including a new inexact rule for the subproblem, and analyze some convergence properties of the method. In Section 4, we consider practical implementation of the inexact rule for several classic methods. In Section 5, we present two practical choices of the approximate Hessian matrix. Finally, preliminary numerical results are provided in Section 6.

## 2 Notation and preliminaries

Throughout this paper, matrices and vectors are written in bold uppercase letters and lowercase letters, respectively. Let $\mathbb{R}^n$ denote the $n$-dimensional Euclidean space equipped with norm $\|\cdot\|$ induced by inner product $\langle\cdot,\cdot\rangle$. For $\boldsymbol{x}\in\mathbb{R}^n$, we let $\|\boldsymbol{x}\|_0$ and $\|\boldsymbol{x}\|_1$ denote the $\ell_0$ norm (number of nonzero entries) and $\ell_1$ norm, respectively. Let $\boldsymbol{I}$ denote the $n$-dimensional identity matrix. We use $\mathcal{S}^n$ to denote the set of all $n$-dimensional symmetric matrices and $\mathcal{S}^n_+$ ($\mathcal{S}^n_{++}$) to denote the set of all $n$-dimensional positive semidefinite (definite) matrices. For $\boldsymbol{A},\boldsymbol{B}\in\mathcal{S}^n$, we let $\boldsymbol{A}\succeq\boldsymbol{B}$ ($\boldsymbol{A}\succ\boldsymbol{B}$) denote $\boldsymbol{A}-\boldsymbol{B}\in\mathcal{S}^n_+(\mathcal{S}^n_{++})$. Given $\boldsymbol{A}\in\mathcal{S}^n$, we let $\lambda_{\min}(\boldsymbol{A})$ and $\lambda_{\max}(\boldsymbol{A})$ denote the smallest eigenvalue and the largest eigenvalue of matrix $\boldsymbol{A}$, respectively.

Given a nonempty closed set $C$, we let $\operatorname{dist}(\boldsymbol{x}, C) := \inf_{\boldsymbol{y}\in C}\|\boldsymbol{x}-\boldsymbol{y}\|$ and define the indicator function $\delta_C(\cdot)$ as

$$\delta_C(\boldsymbol{x}) := \begin{cases} 0 & \text{if } \boldsymbol{x}\in C,\\ \infty & \text{otherwise.}\end{cases}$$

An extended real-valued function $h:\mathbb{R}^n\to\mathbb{R}\cup\{\infty\}$ is said to be proper if $\operatorname{dom} h := \{\boldsymbol{x} : h(\boldsymbol{x}) < \infty\} \neq \varnothing$, and closed if it is lower semicontinuous. For a proper closed convex function $h$, we let $h^*$ denote the conjugate function of $h$. Following [32, Definition 1.22], for a proper closed function $h$ and parameter $\lambda>0$, we define the proximal mapping of $\lambda h$ at $\boldsymbol{x}$ as

$$\operatorname{prox}_{\lambda h}(\boldsymbol{x}) := \underset{\boldsymbol{u}\in\mathbb{R}^n}{\arg\min}\left\{\frac{1}{2\lambda}\|\boldsymbol{x}-\boldsymbol{u}\|^2 + h(\boldsymbol{u})\right\}.$$

We call $h$ "prox-friendly" if $\operatorname{prox}_{\lambda h}(\boldsymbol{x})$ is easy to compute for any $\lambda>0$; for example, $h(\boldsymbol{x}) = \|\boldsymbol{x}\|_1$. Following [32, Definition 8.3], for a proper function $h:\mathbb{R}^n\to\mathbb{R}\cup\{\infty\}$, we let $\partial h(\boldsymbol{x})$ denote the limiting subdifferential at $\boldsymbol{x} \in \operatorname{dom} h$, which is defined as

$$\partial h(\boldsymbol{x}) = \left\{ \boldsymbol{v} : \exists\, \boldsymbol{v}^k \to \boldsymbol{v}, \boldsymbol{x}^k \xrightarrow{h} \boldsymbol{x} \text{ with } \liminf_{\substack{\boldsymbol{y} \to \boldsymbol{x}^k \\ \boldsymbol{y} \neq \boldsymbol{x}^k}} \frac{h(\boldsymbol{y}) - h(\boldsymbol{x}^k) - \langle \boldsymbol{v}^k, \boldsymbol{y} - \boldsymbol{x}^k \rangle}{\|\boldsymbol{y} - \boldsymbol{x}^k\|} \geqslant 0 \ \ \forall\, k \right\},$$

where $\boldsymbol{x}^k \xrightarrow{h} \boldsymbol{x}$ means $\boldsymbol{x}^k \to \boldsymbol{x}$ and $h(\boldsymbol{x}^k) \to h(\boldsymbol{x})$.

To end this section, we give the following definition of stationary points; see, for example, [13, Remark 1] and [39, Definition 4.1].

**Definition 2.1.** (Stationarity) *We say that $\bar{\boldsymbol{x}}$ is a stationary point of* (1.1) *if*

$$\mathbf{0} \in \nabla f(\bar{\boldsymbol{x}}) + \partial h(\bar{\boldsymbol{x}}) - \partial g(\bar{\boldsymbol{x}}).$$

# 3 Inexact successive quadratic approximation method with majorization

In this section, we propose a new method for solving (1.1) and establish corresponding convergence analysis. The idea of the new method is to combine a quadratic approximation of the smooth part $f$ with a linear majorization technique applied to the concave term $-g$. Specifically, given $\boldsymbol{x}^k \in \operatorname{dom} h$, we take $\boldsymbol{\xi}^{k+1} \in \partial g(\boldsymbol{x}^k)$, choose an approximate Hessian matrix $\boldsymbol{B}_k \in \mathcal{S}^n$ and approximately solve the following problem:

$$\boldsymbol{u}^k \approx \operatorname*{arg\,min}_{\boldsymbol{z} \in \mathbb{R}^n} G_k(\boldsymbol{z}) = \left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{z} - \boldsymbol{x}^k \right\rangle + \frac{1}{2}(\boldsymbol{z} - \boldsymbol{x}^k)^\top \boldsymbol{B}_k (\boldsymbol{z} - \boldsymbol{x}^k) + h(\boldsymbol{z}). \tag{3.1}$$

Notice that problem (3.1) reduces to the subproblem of $\text{NPG}_{\text{major}}$ proposed in [25, Algorithm 2] when $\boldsymbol{B}_k = \frac{1}{\gamma_k}\boldsymbol{I}$ for some stepsize $\gamma_k > 0$. Nevertheless, our method is not a direct extension of $\text{NPG}_{\text{major}}$. Indeed, in each iteration of $\text{NPG}_{\text{major}}$, one needs to solve the problem in (3.1) multiple times for searching a final stepsize $\gamma_k$, while in our method the problem in (3.1) is solved only once and inexactly, due to the high computational cost for solving (3.1) with general $\boldsymbol{B}_k$.

Next, we consider the criterion (inexact rule) for accepting an inexact solution $\boldsymbol{u}^k$ in (3.1). In case of $g = 0$ (thus $\boldsymbol{\xi}^{k+1} = \mathbf{0}$) and $\boldsymbol{B}_k \succ \mathbf{0}$, various inexact rules for solving the problem in (3.1) have been proposed; see, for example, [16, 18, 21, 29, 34, 44]. Among them, typical inexact rules include

$$\begin{aligned}
&\left\|\boldsymbol{u}^k - \operatorname{prox}_{\alpha h}\left(\boldsymbol{u}^k - \alpha\left(\nabla f(\boldsymbol{x}^k) + \boldsymbol{B}_k(\boldsymbol{u}^k - \boldsymbol{x}^k)\right)\right)\right\| \leqslant \theta_k \left\|\boldsymbol{x}^k - \operatorname{prox}_{\alpha h}(\boldsymbol{x}^k - \alpha \nabla f(\boldsymbol{x}^k))\right\|, \\
&G_k(\boldsymbol{u}^k) - G_k^* \leqslant \tau_k \left(G_k(\boldsymbol{x}^k) - G_k^*\right), \\
&G_k(\boldsymbol{u}^k) - G_k^* \leqslant \nu_k,
\end{aligned} \tag{3.2}$$

where $G_k^*$ denotes the optimal value of the problem in (3.1), $\alpha > 0$, and $\theta_k$, $\tau_k$ and $\nu_k$ are some non-negative parameters. All the inexact rules in (3.2) basically correspond to some approximate optimality conditions of $\boldsymbol{u}^k$ with tolerance controlled by these parameters. Practical implementation of these rules is generally expensive or even unavailable, due to the extra computation of proximal operators or the difficulty of finding a global minimum $G_k^*$.

In view of the above, we propose the following inexact rule to characterize the inexactness of $\boldsymbol{u}^k$ in (3.1) with tolerance $\epsilon_k > 0$:

$$\operatorname{dist}\left(\mathbf{0},\, \partial G_k(\boldsymbol{u}^k)\right) \leqslant \epsilon_k \, \|\boldsymbol{u}^k - \boldsymbol{x}^k\|. \tag{3.3}$$

As we can see later in Section 4, inexact rule (3.3) can be efficiently implemented when (3.1) is solved by several classical methods. On the other hand, due to the existence of term $\|\boldsymbol{u}^k - \boldsymbol{x}^k\|$ in (3.3), tolerance $\epsilon_k$ is *not* required to diminish to zero, which will be shown later.

As long as some $\boldsymbol{u}^k$ satisfying (3.3) is obtained from (3.1), to guarantee sufficient decrease in the objective value of (1.1), we take a backtracking line search along direction $\boldsymbol{d}_k = \boldsymbol{u}^k - \boldsymbol{x}^k$. Specifically, we consider three variants of line-search methods, which naturally extend existing line-search schemes to our setting by taking the DC regularizer into account and incorporating a non-monotone line-search technique. Namely, when applied to $g = 0$ in (1.1) and taking a monotone scheme (*i.e.*, setting $M = 0$ below), our line-search methods reduce to corresponding existing schemes.

Let $\sigma \in (0,\, 1)$, $\beta \in (0,\, 1)$ and integer $M \geqslant 0$. Our first line-search method is an extension of the one given, for instance, in [16] and [38]. It finds the largest element $\alpha_k$ of $\{\alpha = \beta^i : i = 0, 1, \ldots\}$ satisfying

$$F(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) \leqslant \max_{[k-M]_+ \leqslant j \leqslant k} F(\boldsymbol{x}^j) + \sigma\alpha \triangle_{k,1}, \tag{LS$_1$}$$

where $\triangle_{k,1} := \left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{d}_k \right\rangle + h(\boldsymbol{u}^k) - h(\boldsymbol{x}^k)$.[1] Our second line-search method follows as an extension of the one used, for instance, in [6] and [44]. It finds the largest element $\alpha_k$ of $\{\alpha = \beta^i : i = 0, 1, \ldots\}$ satisfying

$$F(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) \leqslant \max_{[k-M]_+ \leqslant j \leqslant k} F(\boldsymbol{x}^j) + \sigma \triangle_k(\alpha), \tag{LS$_2$}$$

where $\triangle_k(\alpha) := \alpha \left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{d}_k \right\rangle + h(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - h(\boldsymbol{x}^k)$.[2] Our third line-search method extends the one in [41]. It finds the largest element $\alpha_k$ of $\{\alpha = \beta^i : i = 0, 1, \ldots\}$ satisfying

$$F(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) \leqslant \max_{[k-M]_+ \leqslant j \leqslant k} F(\boldsymbol{x}^j) - \sigma\alpha \|\boldsymbol{d}_k\|^2. \tag{LS$_3$}$$

The types of line search (LS$_1$) and (LS$_2$) are commonly used in SQA methods, while the type of line search (LS$_3$) is generally used in first-order methods. Our method allows a free choice of these three line-search methods, and a unifying convergence analysis will be given.

Now we are ready to present our method for solving (1.1) as follows. We call it an inexact successive quadratic approximation method with majorization (iSQA$_{\text{major}}$) .

**Algorithm 1** iSQA$_{\text{major}}$ for solving (1.1)

**Step 0.** Let $\boldsymbol{x}^0 \in \operatorname{dom} h$, $\beta \in (0, 1)$, $\sigma \in (0, 1)$, integer $M \geqslant 0$ and positive sequence $\{\epsilon_k\}$. Set $k = 0$.

**Step 1.** Choose $\boldsymbol{B}_k \in \mathcal{S}^n$ and $\boldsymbol{\xi}^{k+1} \in \partial g(\boldsymbol{x}^k)$. Approximately minimize $G_k$ in (3.1), starting at $\boldsymbol{x}^k$ and terminating at $\boldsymbol{u}^k$ when $\boldsymbol{u}^k$ satisfies (3.3).

**Step 2.** Let $\boldsymbol{d}_k = \boldsymbol{u}^k - \boldsymbol{x}^k$. Find the largest $\alpha_k \in \{\beta^i : i = 0, 1, \ldots\}$ such that (LS$_1$) or (LS$_2$) or (LS$_3$) holds with $\alpha = \alpha_k$.

**Step 3.** Set $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \alpha_k \boldsymbol{d}_k$ and $k \leftarrow k + 1$. Go to **Step 1**.

Before analyzing Algorithm 1, throughout this paper, we make the following blanket assumption on the choice of $\boldsymbol{B}_k$.

**Assumption 3.1.** *For each $k \geqslant 0$, we choose $\boldsymbol{B}_k \succ \boldsymbol{0}$.*

[1] As shown later in (A.4), we have $\triangle_{k,1} \leqslant 0$ under some condition given in Lemma 3.2.

[2] As shown in (A.2), we have $\triangle_k(\alpha) \leqslant \alpha \triangle_{k,1} \leqslant 0$ under some condition given in Lemma 3.2.

### 3.1 Well-definedness of line search and subsequential convergence

In this subsection, we analyze the convergence properties of Algorithm 1. First, we show that under proper choices of $\boldsymbol{B}_k$, each of the three variants of line search in Step 2 of Algorithm 1 is well-defined, *i.e.*, each line-search criterion, (LS$_1$) or (LS$_2$) or (LS$_3$), will be satisfied after finitely many number of backtracking line-search steps. The proof can be found in Appendix A.

**Lemma 3.2.** (Well-definedness of line search in Algorithm 1) *If $\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k > 0$, then* (LS$_1$) *and* (LS$_2$) *are satisfied with $\alpha = \alpha_k$ after finite steps of line search; if $\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k > \sigma$, then* (LS$_3$) *is satisfied with $\alpha = \alpha_k$ after finite steps of line search. Furthermore,*

$$\alpha_k \geqslant \begin{cases} \min\{1,\, 2\beta(1-\sigma)(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k)/L\} & \textit{for } (\mathrm{LS}_1),\, (\mathrm{LS}_2) \textit{ when } \lambda_{\min}(\boldsymbol{B}_k) > \epsilon_k, \\ \min\{1,\, 2\beta(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k - \sigma)/L\} & \textit{for } (\mathrm{LS}_3) \textit{ when } \lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k > \sigma. \end{cases} \tag{3.4}$$

Now, based on Lemma 3.2, we establish the global subsequential convergence of Algorithm 1. The proof can be found in Appendix B.

**Theorem 3.3.** (Subsequential convergence) *Let sequence $\{\boldsymbol{x}^k\}$ be generated by* Algorithm 1*. Suppose that $\{\boldsymbol{B}_k\}$ is bounded. Let $\delta := \inf_k(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k)$. If $\delta > 0$ in case of* (LS$_1$) *or* (LS$_2$)*, or $\delta > \sigma$ in case of* (LS$_3$)*, then the following statements hold.*

(i) *The sequence $\{\boldsymbol{x}^k\}$ is bounded.*

(ii) $\lim_{k\to\infty} \|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\| = 0$.

(iii) *Any accumulation point of $\{\boldsymbol{x}^k\}$ is a stationary point of* (1.1)*.*

**Remark 3.4.** (Assumption on $\delta$) *In* Theorem 3.3*, the assumptions on $\delta$ require that $\epsilon_k$ is uniformly smaller or uniformly $\sigma$ smaller than $\lambda_{\min}(\boldsymbol{B}_k)$. Since there is much freedom on the choices of sequences $\{\boldsymbol{B}_k\}$ and $\{\epsilon_k\}$, the assumptions on $\delta$ can be achieved by properly choosing each $\boldsymbol{B}_k$ first and setting $\sigma$ and $\epsilon_k$ subsequently. For example, when $\{\boldsymbol{B}_k\}$ is constructed as in* [3, 29]*, it has been shown that there exists some $\kappa > 0$ such that $\inf_k(\lambda_{\min}(\boldsymbol{B}_k)) > \kappa$. Therefore, we can set $\sigma \in (0, \min\{1, \kappa\})$ and $\sup_k \epsilon_k < \kappa - \sigma$.*

**Remark 3.5.** (On Assumption 3.1) *Though we assume $\boldsymbol{B}_k \succ \boldsymbol{0}$ throughout the paper, we can obtain similar convergence results as in* Section 3.1 *by allowing $\boldsymbol{B}_k$ to have negative eigenvalues when $h$ is strongly convex. Indeed, in case of $\lambda_{\min}(\boldsymbol{B}_k) < 0$ but $\lambda_{\min}(\boldsymbol{B}_k)$ is not too negative compared with the modulus of strong convexity of $h$ and the inexact tolerance $\epsilon_k$, the well-definedness of line search and subsequential convergence still hold. Specific example of strongly convex $h$ is the elastic net regularizer, i.e., $h(\boldsymbol{x}) = \lambda_1\|\boldsymbol{x}\|_1 + \lambda_2\|\boldsymbol{x}\|^2$ with $\lambda_1 > 0$ and $\lambda_2 > 0$.*

### 3.2 Non-asymptotic complexity analysis

In this subsection, we provide a non-asymptotic analysis of the sequence $\{\boldsymbol{x}^k\}$ generated by Algorithm 1. For this purpose, we first let $\boldsymbol{x}^k_{\boldsymbol{B}_k}$ denote the exact solution of subproblem (3.1) for $\boldsymbol{B}_k \succ \boldsymbol{0}$:

$$\boldsymbol{x}^k_{\boldsymbol{B}_k} := \operatorname*{arg\,min}_{\boldsymbol{z}\in\mathbb{R}^n} G_k(\boldsymbol{z}) = \left\{ \left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{z} - \boldsymbol{x}^k \right\rangle + \frac{1}{2}(\boldsymbol{z} - \boldsymbol{x}^k)^\top \boldsymbol{B}_k(\boldsymbol{z} - \boldsymbol{x}^k) + h(\boldsymbol{z}) \right\}. \tag{3.5}$$

Especially, when $\boldsymbol{B}_k = \boldsymbol{I}$, we let

$$\boldsymbol{x}^k_{\boldsymbol{I}} := \operatorname*{arg\,min}_{\boldsymbol{z}\in\mathbb{R}^n} \left\{ \left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{z} - \boldsymbol{x}^k \right\rangle + \frac{1}{2}\|\boldsymbol{z} - \boldsymbol{x}^k\|^2 + h(\boldsymbol{z}) \right\}. \tag{3.6}$$

Now, we define the following indicator of closeness to the first-order optimality condition for sequence $\{\boldsymbol{x}^k\}$:

$$R_k := \|\boldsymbol{x}_{\boldsymbol{I}}^k - \boldsymbol{x}^k\|. \tag{3.7}$$

The indicator $R_k$ can be regarded as an extension of the so-called residual function proposed in [26]. Indeed, we can further write $R_k$ as

$$R_k = \left\|\boldsymbol{x}^k - \operatorname{prox}_h\left(\boldsymbol{x}^k - \left(\nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1}\right)\right)\right\|,$$

which reduces to be the residual function given in [26] when $g = 0$ (thus $\boldsymbol{\xi}^{k+1} = \mathbf{0}$). The indicator $R_k$ can also be viewed as a natural extension of the indicator in [16], in which problem (1.1) with $g = 0$ was considered. We use $R_k$ as a measure of closeness to the first-order optimality condition for $\boldsymbol{x}^k$ by the reason of the following lemma, which is similar to [16, Lemma 7]. The proof of the lemma can be found in Appendix C.

**Lemma 3.6.** *Let $\boldsymbol{x}_{\boldsymbol{I}}^k$ be defined as in (3.6). If $\boldsymbol{x}_{\boldsymbol{I}}^k = \boldsymbol{x}^k$, then $\boldsymbol{x}^k$ is a stationary point of (1.1). If $\boldsymbol{x}^k$ is a stationary point of (1.1) and $\partial g(\boldsymbol{x}^k)$ is a singleton, then we have $\boldsymbol{x}_{\boldsymbol{I}}^k = \boldsymbol{x}^k$.*

Next, based on $R_k$, we define the following approximate $\varepsilon$-stationary point.

**Definition 3.7.** (Approximate $\varepsilon$-stationary point) *Given $\varepsilon > 0$, we say that an approximate $\varepsilon$-stationary point of (1.1) is obtained from* Algorithm 1 *if there exists some $N \geqslant 0$ such that*

$$\min_{0 \leqslant k \leqslant N} R_k^2 \leqslant \varepsilon. \tag{3.8}$$

In the remaining part of this subsection, we derive the iteration number $N$ needed to obtain an approximate $\varepsilon$-stationary point. We start by bounding $R_k$ in the lemma below, which follows directly from [38, Lemma 3].

**Lemma 3.8.** *Let $\boldsymbol{x}_{\boldsymbol{B}_k}^k$ be defined as in (3.5) with $\boldsymbol{B}_k > \mathbf{0}$ and $R_k$ be defined as in (3.7). Then we have*

$$R_k \leqslant \frac{1 + 1/\lambda_{\min}(\boldsymbol{B}_k) + \sqrt{1 - 2/\lambda_{\max}(\boldsymbol{B}_k) + 1/\lambda_{\min}^2(\boldsymbol{B}_k)}}{2} \lambda_{\max}(\boldsymbol{B}_k) \left\|\boldsymbol{x}_{\boldsymbol{B}_k}^k - \boldsymbol{x}^k\right\|. \tag{3.9}$$

In the following lemma, we can further bound the right-hand side of (3.9) by the searching direction $\boldsymbol{d}_k = \boldsymbol{u}^k - \boldsymbol{x}^k$.

**Lemma 3.9.** *Suppose that $\lambda_{\min}(\boldsymbol{B}_k) \geqslant \epsilon_k$. Let $\boldsymbol{x}_{\boldsymbol{B}_k}^k$ be defined as in (3.5). Then we have*

$$\left\|\boldsymbol{x}_{\boldsymbol{B}_k}^k - \boldsymbol{x}^k\right\| \leqslant 2\left\|\boldsymbol{d}_k\right\|. \tag{3.10}$$

*Proof.* First, we see from (3.3) that there exists some $\boldsymbol{w}_k$ satisfying $\|\boldsymbol{w}_k\| \leqslant \epsilon_k \|\boldsymbol{u}^k - \boldsymbol{x}^k\|$ and

$$\boldsymbol{w}_k \in \partial G_k(\boldsymbol{u}^k) = \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1} + \boldsymbol{B}_k(\boldsymbol{u}^k - \boldsymbol{x}^k) + \partial h(\boldsymbol{u}^k). \tag{3.11}$$

On the other hand, we know from the optimality of $\boldsymbol{x}_{\boldsymbol{B}_k}^k$ in (3.5) that

$$\mathbf{0} \in \partial G_k(\boldsymbol{x}_{\boldsymbol{B}_k}^k) = \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1} + \boldsymbol{B}_k\left(\boldsymbol{x}_{\boldsymbol{B}_k}^k - \boldsymbol{x}^k\right) + \partial h(\boldsymbol{x}_{\boldsymbol{B}_k}^k). \tag{3.12}$$

We further rewrite (3.11) and (3.12) as, respectively,

$$\begin{aligned} \boldsymbol{w}_k - \nabla f(\boldsymbol{x}^k) + \boldsymbol{\xi}^{k+1} - \boldsymbol{B}_k(\boldsymbol{u}^k - \boldsymbol{x}^k) &\in \partial h(\boldsymbol{u}^k), \\ -\nabla f(\boldsymbol{x}^k) + \boldsymbol{\xi}^{k+1} - \boldsymbol{B}_k\left(\boldsymbol{x}_{\boldsymbol{B}_k}^k - \boldsymbol{x}^k\right) &\in \partial h(\boldsymbol{x}_{\boldsymbol{B}_k}^k). \end{aligned}$$

Applying the monotonicity of operator $\partial h$ to above inclusions, we further obtain

$$\begin{aligned}
0 &\leqslant \langle \boldsymbol{w}_k - \boldsymbol{B}_k(\boldsymbol{u}^k - \boldsymbol{x}^k_{\boldsymbol{B}_k}),\, \boldsymbol{u}^k - \boldsymbol{x}^k_{\boldsymbol{B}_k}\rangle \\
&= \langle \boldsymbol{w}_k,\, \boldsymbol{u}^k - \boldsymbol{x}^k_{\boldsymbol{B}_k}\rangle - \left(\boldsymbol{u}^k - \boldsymbol{x}^k_{\boldsymbol{B}_k}\right)^\top \boldsymbol{B}_k \left(\boldsymbol{u}^k - \boldsymbol{x}^k_{\boldsymbol{B}_k}\right) \\
&\leqslant \epsilon_k \left\|\boldsymbol{u}^k - \boldsymbol{x}^k\right\| \left\|\boldsymbol{u}^k - \boldsymbol{x}^k_{\boldsymbol{B}_k}\right\| - \lambda_{\min}(\boldsymbol{B}_k) \left\|\boldsymbol{u}^k - \boldsymbol{x}^k_{\boldsymbol{B}_k}\right\|^2 \\
&\leqslant \lambda_{\min}(\boldsymbol{B}_k) \left\|\boldsymbol{u}^k - \boldsymbol{x}^k\right\| \left\|\boldsymbol{u}^k - \boldsymbol{x}^k_{\boldsymbol{B}_k}\right\| - \lambda_{\min}(\boldsymbol{B}_k) \left\|\boldsymbol{u}^k - \boldsymbol{x}^k_{\boldsymbol{B}_k}\right\|^2,
\end{aligned}$$

where the second inequality follows from $\|\boldsymbol{w}_k\| \leqslant \epsilon_k \|\boldsymbol{u}^k - \boldsymbol{x}^k\|$ and the last inequality follows from $\epsilon_k \leqslant \lambda_{\min}(\boldsymbol{B}_k)$. This implies that

$$\left\|\boldsymbol{u}^k - \boldsymbol{x}^k_{\boldsymbol{B}_k}\right\| \leqslant \left\|\boldsymbol{u}^k - \boldsymbol{x}^k\right\|,$$

which further gives

$$\left\|\boldsymbol{x}^k_{\boldsymbol{B}_k} - \boldsymbol{x}^k\right\| \leqslant \left\|\boldsymbol{u}^k - \boldsymbol{x}^k_{\boldsymbol{B}_k}\right\| + \left\|\boldsymbol{u}^k - \boldsymbol{x}^k\right\| \leqslant 2\left\|\boldsymbol{u}^k - \boldsymbol{x}^k\right\| = 2\|\boldsymbol{d}_k\|.$$

This proves (3.10) and completes the proof. □

Now we are ready to present the iteration complexity of obtaining an approximate $\varepsilon$-stationary point as defined in (3.8). In the following theorem, we first bound $R_k$ by a quantity related to several parameters including the inexact tolerance $\epsilon_k$, and then establish the iteration complexity.

**Theorem 3.10.** *Let sequence $\{\boldsymbol{x}^k\}$ be generated by* Algorithm 1 *with $M = 0$. Suppose that for all $k$ it holds that*

$$\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k > \begin{cases} 0 & \textit{for } (\mathrm{LS}_1) \textit{ and } (\mathrm{LS}_2), \\ \sigma & \textit{for } (\mathrm{LS}_3). \end{cases} \tag{3.13}$$

*Let $F^*$ denote the minimum of $F$ and*

$$c_k := \frac{\sigma(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k)\min\{1,\, 2\beta(1-\sigma)\left(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k\right)/L\}}{\lambda^2_{\max}(\boldsymbol{B}_k)\left(1 + 1/\lambda_{\min}(\boldsymbol{B}_k) + \sqrt{1 - 2/\lambda_{\max}(\boldsymbol{B}_k) + 1/\lambda^2_{\min}(\boldsymbol{B}_k)}\right)^2}, \tag{3.14a}$$

$$\theta_k := \frac{\sigma\min\{1,\, 2\beta(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k - \sigma)/L\}}{\lambda^2_{\max}(\boldsymbol{B}_k)\left(1 + 1/\lambda_{\min}(\boldsymbol{B}_k) + \sqrt{1 - 2/\lambda_{\max}(\boldsymbol{B}_k) + 1/\lambda^2_{\min}(\boldsymbol{B}_k)}\right)^2}. \tag{3.14b}$$

*Then we have for any $N \geqslant 0$ that*

$$\min_{0\leqslant k\leqslant N} R_k^2 \leqslant \begin{cases} \left(F(\boldsymbol{x}^0) - F^*\right)/\sum_{k=0}^N c_k & \textit{for } (\mathrm{LS}_1) \textit{ and } (\mathrm{LS}_2), \\ \left(F(\boldsymbol{x}^0) - F^*\right)/\sum_{k=0}^N \theta_k & \textit{for } (\mathrm{LS}_3). \end{cases} \tag{3.15}$$

*In particular, if (3.13) holds with $\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k$ replaced by $\inf_k\left(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k\right)$ and $\{\boldsymbol{B}_k\}$ is bounded, then the iteration complexity of obtaining an approximate $\varepsilon$-stationary point defined as in (3.8) is $N = O(\varepsilon^{-1})$.*

*Proof.* First, we see from assumption (3.13) that (3.4) holds:

$$\alpha_k \geqslant \begin{cases} \min\{1,\, 2\beta(1-\sigma)\left(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k\right)/L\} & \textit{for } (\mathrm{LS}_1) \textit{ and } (\mathrm{LS}_2), \\ \min\{1,\, 2\beta(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k - \sigma)/L\} & \textit{for } (\mathrm{LS}_3). \end{cases} \tag{3.16}$$

Now we define two parameters based on (3.16):

$$\begin{aligned}
\hat{\alpha}_1^k &:= \left(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k\right)\min\{1,\, 2\beta(1-\sigma)\left(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k\right)/L\}, \\
\hat{\alpha}_2^k &:= \min\{1,\, 2\beta(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k - \sigma)/L\}.
\end{aligned} \tag{3.17}$$

Using (3.16) and (3.17), we further obtain

$$F(\boldsymbol{x}^{k+1}) = F(\boldsymbol{x}^k + \alpha_k \boldsymbol{d}_k) \overset{\text{(a)}}{\leqslant} F(\boldsymbol{x}^k) + \begin{cases} \sigma\alpha_k \triangle_{k,1} \overset{\text{(b)}}{\leqslant} -\sigma\widehat{\alpha}_1^k \|\boldsymbol{d}_k\|^2 & \textit{for } (\mathrm{LS}_1), \\ \sigma\triangle_k(\alpha_k) \overset{\text{(c)}}{\leqslant} \sigma\alpha_k\triangle_{k,1} \leqslant -\sigma\widehat{\alpha}_1^k\|\boldsymbol{d}_k\|^2 & \textit{for } (\mathrm{LS}_2), \\ -\sigma\alpha_k\|\boldsymbol{d}_k\|^2 \overset{\text{(d)}}{\leqslant} -\sigma\widehat{\alpha}_2^k\|\boldsymbol{d}_k\|^2 & \textit{for } (\mathrm{LS}_3). \end{cases} \tag{3.18}$$

where (a) follows from $M = 0$, (b) follows from $\triangle_{k,1} \leqslant -(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k)\|\boldsymbol{d}_k\|^2$ (see (A.4)), the definition of $\widehat{\alpha}_1^k$ in (3.17) and the first inequality in (3.16), (c) follows from $\triangle_k(\alpha) \leqslant \alpha\,\triangle_{k,1}$ (due to (A.2) for $\alpha \in (0, 1]$), and (d) follows from (3.16) and (3.17). Now we define another parameter

$$\beta_k := \lambda_{\max}(\boldsymbol{B}_k)\Big(1 + 1/\lambda_{\min}(\boldsymbol{B}_k) + \sqrt{1 - 2/\lambda_{\max}(\boldsymbol{B}_k) + 1/\lambda_{\min}^2(\boldsymbol{B}_k)}\Big). \tag{3.19}$$

Furthermore, we have

$$R_k^2 \leqslant \frac{\beta_k^2}{4}\left\|\boldsymbol{x}_{\boldsymbol{B}_k}^k - \boldsymbol{x}^k\right\|^2 \leqslant \beta_k^2\left\|\boldsymbol{d}_k\right\|^2 \leqslant \begin{cases} \beta_k^2/(\sigma\widehat{\alpha}_1^k)\left(F(\boldsymbol{x}^k) - F(\boldsymbol{x}^{k+1})\right) & \textit{for } (\mathrm{LS}_1) \textit{ and } (\mathrm{LS}_2), \\ \beta_k^2/(\sigma\widehat{\alpha}_2^k)\left(F(\boldsymbol{x}^k) - F(\boldsymbol{x}^{k+1})\right) & \textit{for } (\mathrm{LS}_3), \end{cases} \tag{3.20}$$

where the first inequality follows from (3.9) and (3.19), the second inequality follows from (3.10), and the last inequalities follow from (3.18). We further see from (3.14a), (3.14b), (3.17), (3.19) and (3.20) that

$$\left.\begin{aligned} \textit{for } (\mathrm{LS}_1) \textit{ and } (\mathrm{LS}_2),\ \ c_k R_k^2 &= (\sigma\widehat{\alpha}_1^k/\beta_k^2)R_k^2 \\ \textit{for } (\mathrm{LS}_3),\ \ \theta_k R_k^2 &= (\sigma\widehat{\alpha}_2^k/\beta_k^2)R_k^2 \end{aligned}\right\} \leqslant F(\boldsymbol{x}^k) - F(\boldsymbol{x}^{k+1}). \tag{3.21}$$

Now for any $N \geqslant 0$, in case of (LS$_1$) or (LS$_2$), we sum the first inequality in (3.21) for $k = 0, 1, \ldots, N$ and obtain

$$\Big(\min_{0\leqslant k\leqslant N} R_k^2\Big)\Big(\sum_{k=0}^N c_k\Big) \leqslant \sum_{k=0}^N c_k\, R_k^2 \leqslant \sum_{k=0}^N \left(F(\boldsymbol{x}^k) - F(\boldsymbol{x}^{k+1})\right) = F(\boldsymbol{x}^0) - F(\boldsymbol{x}^{N+1}) \leqslant F(\boldsymbol{x}^0) - F^*.$$

We rearrange this and obtain the first inequality in (3.15). Similarly, in case of (LS$_3$), we can obtain another inequality in (3.15).

Moreover, if $\inf_k \lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k > 0$ in case of (LS$_1$) and (LS$_2$), and $\inf_k \lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k > \sigma$ in case of (LS$_3$), we have $\inf_k c_k > 0$ and $\inf_k \theta_k > 0$, respectively. This together with (3.15) gives

$$\min_{0\leqslant k\leqslant N} R_k^2 \leqslant \begin{cases} \left(F(\boldsymbol{x}^0) - F^*\right)/\left((N+1)\inf_k c_k\right) & \textit{for } (\mathrm{LS}_1) \textit{ and } (\mathrm{LS}_2), \\ \left(F(\boldsymbol{x}^0) - F^*\right)/\left((N+1)\inf_k \theta_k\right) & \textit{for } (\mathrm{LS}_3), \end{cases}$$

which further implies that (3.8) is guaranteed when $N = O(\varepsilon^{-1})$. This completes the proof. □

**Remark 3.11.** *As we can see from* Theorem 3.10*, the iteration complexity $O(\varepsilon^{-1})$ is guaranteed when the inexact tolerance $\epsilon_k$ in comparison with $\lambda_{\min}(\boldsymbol{B}_k)$ is uniformly smaller than some threshold, but does not need to diminish to zero.*

# 4 Towards implementation of methods for subproblem

In this section, we consider the implementation of subproblem (3.1) for obtaining an inexact solution $\boldsymbol{u}^k$ satisfying (3.3). Our emphasis in this section is *not* put on the guidance of choosing solvers for subproblem (3.1). Instead, we focus on achieving an inexact solution $\boldsymbol{u}^k$ satisfying (3.3) by seeking a practical/cheap termination criterion for each solver.

First, we rewrite the $k$th subproblem (3.1) as follows:

$$\min_{\boldsymbol{z}\in\mathbb{R}^n} G_k(\boldsymbol{z}) = \underbrace{\left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{z}-\boldsymbol{x}^k\right\rangle + \frac{1}{2}(\boldsymbol{z}-\boldsymbol{x}^k)^\top \boldsymbol{B}_k(\boldsymbol{z}-\boldsymbol{x}^k)}_{\phi(\boldsymbol{z})} + h(\boldsymbol{z}). \tag{4.1}$$

Here, since $\boldsymbol{B}_k$ is positive definite, due to the structure of (4.1), various optimization methods can be applied. In the following, we consider two typical classes of methods among them, and propose corresponding proper termination criterion for each method.

## 4.1 Accelerated proximal gradient methods

Since $\phi$ is strongly convex and smooth and $h$ is prox-friendly, variants of accelerated proximal gradient methods can be directly applied to (4.1). One famous such algorithm is the so-called FISTA proposed in [8]. We apply FISTA to (4.1) with initialization at $\boldsymbol{x}^k$: let $\boldsymbol{y}^1 = \boldsymbol{z}^0 = \boldsymbol{x}^k$, set $\theta_1 = 1$ and for $\ell = 1, 2, \ldots$, let

$$\begin{cases} \boldsymbol{z}^\ell = \operatorname{prox}_{\frac{1}{L_\phi}h}\left(\boldsymbol{y}^\ell - \nabla\phi(\boldsymbol{y}^\ell)/L_\phi\right), \\ \theta_{\ell+1} = \dfrac{1+\sqrt{1+4\theta_\ell^2}}{2}, \\ \boldsymbol{y}^{\ell+1} = \boldsymbol{z}^\ell + \dfrac{\theta_\ell - 1}{\theta_{\ell+1}}(\boldsymbol{z}^\ell - \boldsymbol{z}^{\ell-1}), \end{cases} \tag{FISTA}$$

where $L_\phi = \lambda_{\max}(\boldsymbol{B}_k)$ is a Lipschitz constant of $\nabla\phi$. It has been known that (FISTA) achieves the sublinear convergence rate; see [8, Theorem 4.4]. Another variant of accelerated proximal gradient method given in [2, page 302], called V-FISTA, exhibits linear convergence rate; see [2, Theorem 10.42]. Given $\boldsymbol{y}^1 = \boldsymbol{z}^0 = \boldsymbol{x}^k$, the iterates of V-FISTA applied to (4.1) are as follows:

$$\begin{cases} \boldsymbol{z}^\ell = \operatorname{prox}_{\frac{1}{L_\phi}h}\left(\boldsymbol{y}^\ell - \nabla\phi(\boldsymbol{y}^\ell)/L_\phi\right), \\ \boldsymbol{y}^{\ell+1} = \boldsymbol{z}^\ell + \dfrac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}(\boldsymbol{z}^\ell - \boldsymbol{z}^{\ell-1}), \end{cases} \tag{V-FISTA}$$

where $\kappa = \frac{L_\phi}{\sigma_\phi}$ with $L_\phi = \lambda_{\max}(\boldsymbol{B}_k)$ and $\sigma_\phi = \lambda_{\min}(\boldsymbol{B}_k)$.

As mentioned at the beginning of this section, we aim for achieving an inexact solution $\boldsymbol{u}^k$ satisfying (3.3). To this purpose, we propose the following termination criterion for both (FISTA) and (V-FISTA). Specifically, we terminate (FISTA) and (V-FISTA) and set $\boldsymbol{u}^k = \boldsymbol{z}^\ell$ if $\boldsymbol{z}^\ell$ satisfies

$$\left\|\boldsymbol{z}^\ell - \boldsymbol{y}^\ell\right\| \leqslant \frac{\epsilon_k}{2L_\phi}\left\|\boldsymbol{z}^\ell - \boldsymbol{z}^0\right\|, \tag{4.2}$$

where $\epsilon_k$ is the inexact tolerance in (3.3). Before showing that the output $\boldsymbol{u}^k$ satisfies inexact rule (3.3), we take a look at termination criterion (4.2). The left-hand side of (4.2) can also be written as $\frac{1}{L_\phi}\|M_{\phi,h}^{L_\phi}(\boldsymbol{y}^\ell)\|$, where $M_{\phi,h}^{L_\phi}(\boldsymbol{y}^\ell)$ is the gradient mapping at $\boldsymbol{y}^\ell$; see the definition of gradient mapping in [2, Definition 10.5]. Consequently, termination criterion (4.2) can be rewritten as $\|M_{\phi,h}^{L_\phi}(\boldsymbol{y}^\ell)\| \leqslant \frac{\epsilon_k}{2}\left\|\boldsymbol{z}^\ell - \boldsymbol{z}^0\right\|$. On the other hand, checking stopping criterion (4.2) does *not* need further computational cost. Indeed, $\boldsymbol{y}^\ell$ and $\boldsymbol{z}^\ell$ are the quantities already obtained in the iterates.

Now, we show that termination criterion (4.2) outputs a solution $\boldsymbol{u}^k = \boldsymbol{z}^\ell$ satisfying inexact rule (3.3).

**Lemma 4.1.** (Guarantee of inexact rule) *Suppose that $\boldsymbol{B}_k \succ \boldsymbol{0}$. Let $\boldsymbol{z}^{\bar{\ell}}$ be the output of* (FISTA) *and* (V-FISTA) *with termination criterion* (4.2)*. Then inexact rule* (3.3) *holds with $\boldsymbol{u}^k = \boldsymbol{z}^{\bar{\ell}}$.*

*Proof.* For both (FISTA) and (V-FISTA), we see from the update of $\boldsymbol{z}^{\ell}$ and termination criterion (4.2) that

$$\boldsymbol{z}^{\bar{\ell}} = \operatorname{prox}_{\frac{1}{L_\phi} h}\left(\boldsymbol{y}^{\bar{\ell}} - \nabla\phi(\boldsymbol{y}^{\bar{\ell}})/L_\phi\right), \tag{4.3}$$

$$\|\boldsymbol{z}^{\bar{\ell}} - \boldsymbol{y}^{\bar{\ell}}\| \leqslant \frac{\epsilon_k}{2L_\phi}\|\boldsymbol{z}^{\bar{\ell}} - \boldsymbol{z}^0\| = \frac{\epsilon_k}{2L_\phi}\|\boldsymbol{u}^k - \boldsymbol{x}^k\|, \tag{4.4}$$

where the last equality follows from $\boldsymbol{z}^0 = \boldsymbol{x}^k$ and $\boldsymbol{u}^k = \boldsymbol{z}^{\bar{\ell}}$. Let $\boldsymbol{w}_k = \boldsymbol{y}^{\bar{\ell}} - \boldsymbol{z}^{\bar{\ell}}$. We then have

$$L_\phi \boldsymbol{w}_k - \nabla\phi(\boldsymbol{z}^{\bar{\ell}} + \boldsymbol{w}_k) = L_\phi(\boldsymbol{y}^{\bar{\ell}} - \boldsymbol{z}^{\bar{\ell}}) - \nabla\phi(\boldsymbol{y}^{\bar{\ell}}) = L_\phi\left(\boldsymbol{y}^{\bar{\ell}} - \nabla\phi(\boldsymbol{y}^{\bar{\ell}})/L_\phi - \boldsymbol{z}^{\bar{\ell}}\right) \in \partial h(\boldsymbol{z}^{\bar{\ell}}) = \partial h(\boldsymbol{u}^k),$$

where the inclusion follows from the optimality condition of the corresponding optimization problem in (4.3). Furthermore, by using this and $\partial G_k = \nabla\phi + \partial h$, we have

$$\begin{aligned}
&\operatorname{dist}\left(\mathbf{0},\, \partial G_k(\boldsymbol{u}^k)\right) \leqslant \|\nabla\phi(\boldsymbol{u}^k) + L_\phi\boldsymbol{w}_k - \nabla\phi(\boldsymbol{z}^{\bar{\ell}} + \boldsymbol{w}_k)\| = \|\nabla\phi(\boldsymbol{u}^k) + L_\phi\boldsymbol{w}_k - \nabla\phi(\boldsymbol{u}^k + \boldsymbol{w}_k)\| \\
&\leqslant \|\nabla\phi(\boldsymbol{u}^k) - \nabla\phi(\boldsymbol{u}^k + \boldsymbol{w}_k)\| + L_\phi\|\boldsymbol{w}_k\| \leqslant 2L_\phi\|\boldsymbol{w}_k\| = 2L_\phi\|\boldsymbol{z}^{\bar{\ell}} - \boldsymbol{y}^{\bar{\ell}}\| \leqslant \epsilon_k\,\|\boldsymbol{u}^k - \boldsymbol{x}^k\|,
\end{aligned}$$

where the third inequality follows from the $L_\phi$-smoothness of $\phi$ and the last inequality follows from (4.4). This completes the proof. □

Still, we need to show that (4.2) is a well-defined termination criterion, *i.e.*, it can be achieved after finitely many number of iterations. In the following, specifically, we establish the iteration complexity of (FISTA) and (V-FISTA) for obtaining a solution satisfying termination criterion (4.2). The proof can be found in Appendix D.

**Theorem 4.2.** (Iteration complexity of (FISTA) and (V-FISTA)) *Suppose that $\boldsymbol{B}_k \succ \mathbf{0}$. Let $\boldsymbol{z}^*$ be the optimal solution of* (4.1) *and*

$$\kappa := \frac{\lambda_{\max}(\boldsymbol{B}_k)}{\lambda_{\min}(\boldsymbol{B}_k)},\ \tau := \left(1 - \frac{1}{\sqrt{\kappa}}\right)^{-\frac{1}{2}},\ c_1 := 2\sqrt{\kappa},\ c_2 := \sqrt{\frac{2\left(G_k(\boldsymbol{x}^k) - G_k(\boldsymbol{z}^*)\right)}{\lambda_{\min}(\boldsymbol{B}_k)} + \|\boldsymbol{x}^k - \boldsymbol{z}^*\|^2}. \tag{4.5}$$

*Suppose that $\boldsymbol{x}^k \neq \boldsymbol{z}^*$. Then the termination criterion* (4.2) *for* (FISTA) *is satisfied whenever*

$$\ell \geqslant \max\left\{2,\, c_1,\, \frac{c_1}{2} + \frac{4c_1L_\phi}{\epsilon_k} + \sqrt{\frac{(c_1-2)^2}{4} + \frac{4c_1L_\phi(c_1+2)}{\epsilon_k} + \frac{16c_1^2L_\phi^2}{\epsilon_k^2}}\right\}, \tag{4.6}$$

*and the termination criterion* (4.2) *for* (V-FISTA) *is satisfied whenever*

$$\ell \geqslant \max\left\{2,\, 1 + \log_\tau \frac{c_2}{\|\boldsymbol{z}^0 - \boldsymbol{z}^*\|},\, \log_\tau \frac{c_2\,\left(\epsilon_k + 8L_\phi\tau^2\right)}{\epsilon_k\|\boldsymbol{z}^0 - \boldsymbol{z}^*\|}\right\}. \tag{4.7}$$

**Remark 4.3.** *It has been known from* [8, Theorem 4.4] *and* [2, Theorem 10.42] *that in case of $\boldsymbol{B}_k \succ 0$ the iteration complexity of* (FISTA) *and* (V-FISTA) *is sub-linear and linear, respectively, which has the same complexity order as that in* Theorem 4.2*. However, in* [8, Theorem 4.4] *and* [2, Theorem 10.42]*, the complexity is measured by the distance to the optimal function value, while in* Theorem 4.2 *our complexity is measured by using the termination criterion* (4.2)*. Moreover, when $\boldsymbol{x}^k = \boldsymbol{z}^*$, we have $\boldsymbol{z}^1 = \boldsymbol{y}^1 = \boldsymbol{x}^k$ for both* (FISTA) *and* (V-FISTA)*, and termination criterion* (4.2) *is satisfied after one iteration.*

**Remark 4.4.** (Inexact tolerance and outer/inner iteration complexity) *On one hand, we know from* (3.14a) *and the first inequality of* (3.15) *in* Theorem 3.10 *that in case of $\lambda_{\min}(\boldsymbol{B}_k) > \epsilon_k$ and line search* ($\text{LS}_1$) *or* ($\text{LS}_2$) *in* Algorithm 1*, it holds that*

$$\min_{0 \leqslant k \leqslant N} R_k^2 \leqslant \frac{F(\boldsymbol{x}^0) - F^*}{\sum_{k=0}^N c_k} \text{ with } c_k := \frac{\sigma(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k)\min\left\{1,\, 2\beta(1-\sigma)\left(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k\right)/L\right\}}{\lambda_{\max}^2(\boldsymbol{B}_k)\left(1 + 1/\lambda_{\min}(\boldsymbol{B}_k) + \sqrt{1 - 2/\lambda_{\max}(\boldsymbol{B}_k) + 1/\lambda_{\min}^2(\boldsymbol{B}_k)}\right)^2}.$$

*This implies that an approximate $\varepsilon$-stationary point defined as in* (3.8), i.e., $\min_{0\leqslant k\leqslant N} R_k^2 \leqslant \varepsilon$, *is obtained whenever*

$$\sum_{k=0}^{N} c_k \geqslant \frac{F(\boldsymbol{x}^0)-F^*}{\varepsilon}. \tag{4.8}$$

*Note that smaller $\epsilon_k$ implies larger $c_k$. This together with* (4.8) *further implies that to obtain an approximate $\varepsilon$-stationary point, smaller inexact tolerance $\epsilon_k$ potentially leads to fewer outer iteration number $N$.*

*On the other hand, we see from* (4.6) *and* (4.7) *in* Theorem 4.2 *that for both* (FISTA) *and* (V-FISTA), *smaller inexact tolerance $\epsilon_k$ potentially leads to more inner iteration number $\ell$. Nevertheless, it is difficult to properly characterize optimal choices of $\epsilon_k$. Indeed, as we can see from* (4.7), *the inner iteration number in $k$th outer iterate also depends on $\boldsymbol{x}^k$ ($\boldsymbol{z}^0 = \boldsymbol{x}^k$ and $c_2$ also depends on $\boldsymbol{x}^k$) which is implicitly influenced by previous $\epsilon_{k-1}, \epsilon_{k-2}, \ldots$. Consequently, all optimal $\epsilon_k$s are not independent.*

## 4.2 A semismooth Newton augmented Lagrangian method

Due to the positive definiteness of $\boldsymbol{B}_k$, we can decompose $\boldsymbol{B}_k = \boldsymbol{A}^\top\boldsymbol{A}$ and rewrite (4.1) as

$$\min_{\boldsymbol{v}\in\mathbb{R}^n} G_k(\boldsymbol{v}) = \left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{v}\right\rangle + \frac{1}{2}\left\|\boldsymbol{A}(\boldsymbol{v}-\boldsymbol{x}^k)\right\|^2 + h(\boldsymbol{v}). \tag{4.9}$$

For simplicity of notation, we define

$$\boldsymbol{c} := -\nabla f(\boldsymbol{x}^k) + \boldsymbol{\xi}^{k+1}, \quad \psi(\boldsymbol{y}) := \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}^k\|^2.$$

Then problem (4.9) can be equivalently written as

$$\max_{\boldsymbol{v}\in\mathbb{R}^n} -\left\{G_k(\boldsymbol{v}) = \psi(\boldsymbol{A}\boldsymbol{v}) - \langle \boldsymbol{c},\, \boldsymbol{v}\rangle + h(\boldsymbol{v})\right\}, \tag{4.10}$$

whose dual problem is

$$\min_{\boldsymbol{y},\,\boldsymbol{z}} \quad \psi^*(\boldsymbol{y}) + h^*(\boldsymbol{z}) \ \text{ s.t. } \ \boldsymbol{A}^\top\boldsymbol{y} + \boldsymbol{z} = \boldsymbol{c}. \tag{4.11}$$

Consequently, a semismooth Newton augmented Lagrangian (Ssnal) method in [22, Section 3] can be applied for solving (4.10) and (4.11). In this subsection, we first briefly revisit the Ssnal method and then propose a practical/implementable termination criterion for the method to achieve an inexact solution $\boldsymbol{u}^k$ satisfying (3.3).

First, given $\sigma > 0$, the augmented Lagrangian function associated with (4.11) is defined as

$$\mathcal{L}_\sigma(\boldsymbol{y},\, \boldsymbol{z};\, \boldsymbol{w}) := \psi^*(\boldsymbol{y}) + h^*(\boldsymbol{z}) - \langle \boldsymbol{w},\, \boldsymbol{A}^\top\boldsymbol{y} + \boldsymbol{z} - \boldsymbol{c}\rangle + \frac{\sigma}{2}\left\|\boldsymbol{A}^\top\boldsymbol{y} + \boldsymbol{z} - \boldsymbol{c}\right\|^2.$$

The Ssnal method in [22, Section 3] contains the outer algorithm [22, Algorithm Ssnal] and the inner algorithm [22, Algorithm Ssn]. Let $\boldsymbol{w}^0 = \boldsymbol{x}^k$. Given sequence $\{\sigma_t\} \uparrow \infty$. The iterates of outer algorithm [22, Algorithm Ssnal] applied to (4.11) are as follows:

$$\begin{cases} (\boldsymbol{y}^{t+1},\, \boldsymbol{z}^{t+1}) \approx \arg\min\left\{\Phi_t(\boldsymbol{y},\, \boldsymbol{z}) := \mathcal{L}_{\sigma_t}(\boldsymbol{y},\, \boldsymbol{z};\, \boldsymbol{w}^t)\right\}, \\ \qquad\quad \boldsymbol{w}^{t+1} = \boldsymbol{w}^t - \sigma_t(\boldsymbol{A}^\top\boldsymbol{y}^{t+1} + \boldsymbol{z}^{t+1} - c), \end{cases} \tag{Ssnal}$$

where "$\approx$" means that for some summable non-negative sequence $\{\mu_t\}$, *i.e.*, $\sum_{t=0}^\infty \mu_t < \infty$, approximate solution $(\boldsymbol{y}^{t+1},\, \boldsymbol{z}^{t+1})$ satisfies

$$\Phi_t(\boldsymbol{y}^{t+1},\, \boldsymbol{z}^{t+1}) - \inf \Phi_t \leqslant \mu_t^2/2\sigma_t. \tag{4.12}$$

To obtain such an approximate solution, we first define

$$\varphi_t(\boldsymbol{y}) := \inf_{\boldsymbol{z}} \Phi_t(\boldsymbol{y},\, \boldsymbol{z}) = \psi^*(\boldsymbol{y}) - \frac{\|\boldsymbol{w}^t\|^2}{2\sigma_t} + \inf_{\boldsymbol{z}}\left\{h^*(\boldsymbol{z}) + \frac{\sigma_t}{2}\left\|\boldsymbol{A}^\top\boldsymbol{y} + \boldsymbol{z} - \boldsymbol{c} - \frac{\boldsymbol{w}^t}{\sigma_t}\right\|^2\right\}. \tag{4.13}$$

By [32, Theorem 2.26], it holds that

$$\begin{aligned}\nabla\varphi_t(\boldsymbol{y}) &= \nabla\psi^*(\boldsymbol{y}) + \sigma_t\boldsymbol{A}\left(\boldsymbol{A}^\top\boldsymbol{y} + \mathrm{prox}_{h*/\sigma_t}(\boldsymbol{c} + \boldsymbol{w}^t/\sigma_t - \boldsymbol{A}^\top\boldsymbol{y}) - \boldsymbol{c} - \boldsymbol{w}^t/\sigma_t\right)\\ &= \boldsymbol{y} + \boldsymbol{A}\boldsymbol{x}^k - \boldsymbol{A}\mathrm{prox}_{\sigma_t h}\left(\boldsymbol{w}^t - \sigma_t(\boldsymbol{A}^\top\boldsymbol{y} - \boldsymbol{c})\right),\end{aligned}$$

where the last equation follows from the generalized Moreau decomposition. Next, the inner algorithm [22, Algorithm SSN] is applied for solving equation $\nabla\varphi_t(\boldsymbol{y}) = 0$, and terminates at some $\boldsymbol{y}^{t+1}$ satisfying

$$\|\nabla\varphi_t(\boldsymbol{y}^{t+1})\| \leqslant \mu_t/\sqrt{\sigma_t}. \tag{4.14}$$

We know from [22, Theorem 3.6] that if $\mathrm{prox}_{\sigma h}(\cdot)$ is strongly semismooth, criterion (4.14) is well-defined. Furthermore, $\boldsymbol{z}^{t+1}$ is obtained by

$$\begin{aligned}\boldsymbol{z}^{t+1} &= \arg\min_{\boldsymbol{z}} \Phi_t(\boldsymbol{y}^{t+1}, \boldsymbol{z}) = \arg\min_{\boldsymbol{z}} \left\{h^*(\boldsymbol{z}) + \frac{\sigma_t}{2}\left\|\boldsymbol{A}^\top\boldsymbol{y}^{t+1} + \boldsymbol{z} - \boldsymbol{c} - \frac{\boldsymbol{w}^t}{\sigma_t}\right\|^2\right\}\\ &= \mathrm{prox}_{h*/\sigma_t}(\boldsymbol{c} + \boldsymbol{w}^t/\sigma_t - \boldsymbol{A}^\top\boldsymbol{y}^{t+1}).\end{aligned} \tag{4.15}$$

Indeed, it has been indicated in [22, Section 3.2] that $(\boldsymbol{y}^{t+1}, \boldsymbol{z}^{t+1})$ obtained from above (*i.e.*, $\boldsymbol{y}^{t+1}$ satisfies (4.14) and $\boldsymbol{z}^{t+1}$ from (4.15)) satisfies criterion (4.12).

As mentioned at the beginning of this section, we focus on achieving an inexact solution $\boldsymbol{u}^k$ satisfying (3.3). To this end, we propose an implementable termination criterion for (SSNAL). Specifically, we terminate (SSNAL) and set $\boldsymbol{u}^k = \boldsymbol{w}^{t+1}$ if $\boldsymbol{w}^{t+1}$ satisfies

$$\left\|\frac{\boldsymbol{w}^t - \boldsymbol{w}^{t+1}}{\sigma_t} - \boldsymbol{A}^\top\nabla\varphi_t(\boldsymbol{y}^{t+1})\right\| \leqslant \epsilon_k\|\boldsymbol{w}^{t+1} - \boldsymbol{x}^k\|, \tag{4.16}$$

where $\epsilon_k$ is the inexact tolerance in (3.3). As one can see, checking termination criterion (4.16) does *not* involve further computational cost. Indeed, $\nabla\varphi_t(\boldsymbol{y}^{t+1})$ is the quantity already computed in the termination criterion (4.14) of the inner algorithm.

Now, we show that termination criterion (4.16) is well-defined, and it outputs a solution $\boldsymbol{u}^k = \boldsymbol{w}^{t+1}$ satisfying inexact rule (3.3).

**Theorem 4.5.** *Let sequence $\{(\boldsymbol{y}^t, \boldsymbol{z}^t, \boldsymbol{w}^t)\}$ be generated by* (SSNAL) *with the inner algorithm terminated by* (4.14). *Suppose that $\boldsymbol{w}^0$ is not the optimal solution of* (4.1). *Then termination criterion* (4.16) *is well-defined. Let $\boldsymbol{w}^{\bar{t}+1}$ be the output of* (SSNAL) *with termination criterion* (4.16). *Then inexact rule* (3.3) *holds for $\boldsymbol{u}^k = \boldsymbol{w}^{\bar{t}+1}$.*

*Proof.* First, we prove the well-definedness of termination criterion (4.16). Since $h$ is convex, $\psi$ is strongly convex and $\nabla\psi$ is Lipschitz continuous with Lipschitz constant 1, by [22, Theorem 3.2], we have that $\{\boldsymbol{w}^t\}$ converges to the optimal solution of (4.1), say $\boldsymbol{z}^*$. This together with (4.14) and $\sigma_t \to \infty$ gives

$$\left\|\frac{\boldsymbol{w}^t - \boldsymbol{w}^{t+1}}{\sigma_t} - \boldsymbol{A}^\top\nabla\varphi_t(\boldsymbol{y}^{t+1})\right\| \leqslant \frac{\|\boldsymbol{w}^t - \boldsymbol{w}^{t+1}\|}{\sigma_t} + \|\boldsymbol{A}\|\|\nabla\varphi_t(\boldsymbol{y}^{t+1})\| \to 0. \tag{4.17}$$

Since $\boldsymbol{x}^k = \boldsymbol{w}^0 \neq \boldsymbol{z}^*$, we then know from $\boldsymbol{w}^t \to \boldsymbol{z}^*$ that there exists some $\tau > 0$ such that when $t$ is large enough, we have $\|\boldsymbol{w}^{t+1} - \boldsymbol{x}^k\| > \tau$. This together with (4.17) implies that (4.16) is satisfied when $t$ is large enough. This proves the well-definedness of (4.16).

Now we prove that inexact rule (3.3) holds for $\boldsymbol{u}^k = \boldsymbol{w}^{\bar{t}+1}$. First, we see from the $\boldsymbol{w}$-update in (SSNAL) and (4.15) that

$$\boldsymbol{z}^{t+1} = \mathrm{prox}_{h*/\sigma_t}(\boldsymbol{c} + \boldsymbol{w}^t/\sigma_t - \boldsymbol{A}^\top\boldsymbol{y}^{t+1}) = \mathrm{prox}_{h*/\sigma_t}(\boldsymbol{z}^{t+1} + \boldsymbol{w}^{t+1}/\sigma_t).$$

Using the optimality condition of the optimization problem involved in the second proximal term above, we further have $\boldsymbol{w}^{t+1} \in \partial h^*(\boldsymbol{z}^{t+1})$, which together with the convexity of $h$ implies

$$\boldsymbol{z}^{t+1} \in \partial h(\boldsymbol{w}^{t+1}). \tag{4.18}$$

On the other hand, we see from (4.13) that

$$\begin{aligned} \nabla\varphi_t(\boldsymbol{y}^{t+1}) &= \nabla\psi^*(\boldsymbol{y}^{t+1}) + \sigma_t \boldsymbol{A}\left(\boldsymbol{A}^\top \boldsymbol{y}^{t+1} + \operatorname{prox}_{h^*/\sigma_t}(\boldsymbol{c} + \boldsymbol{w}^t/\sigma_t - \boldsymbol{A}^\top \boldsymbol{y}^{t+1}) - \boldsymbol{c} - \boldsymbol{w}^t/\sigma_t\right) \\ &= \boldsymbol{y}^{t+1} + \boldsymbol{A}\boldsymbol{x}^k + \sigma_t \boldsymbol{A}\left(\boldsymbol{A}^\top \boldsymbol{y}^{t+1} + \boldsymbol{z}^{t+1} - \boldsymbol{c} - \boldsymbol{w}^t/\sigma_t\right) \\ &= \boldsymbol{y}^{t+1} + \boldsymbol{A}\boldsymbol{x}^k - \boldsymbol{A}\boldsymbol{w}^{t+1}, \end{aligned} \tag{4.19}$$

where the second equality follows from (4.15) and the last equality follows from $\boldsymbol{w}$-update in (Ssnal). We combine (4.18) with (4.19) to obtain

$$\begin{aligned} \partial G_k(\boldsymbol{w}^{t+1}) \ni \boldsymbol{z}^{t+1} + \boldsymbol{A}^\top \boldsymbol{A}(\boldsymbol{w}^{t+1} - \boldsymbol{x}^k) - \boldsymbol{c} &= \boldsymbol{z}^{t+1} + \boldsymbol{A}^\top(\boldsymbol{y}^{t+1} - \nabla\varphi_t(\boldsymbol{y}^{t+1})) - \boldsymbol{c} \\ = \boldsymbol{A}^\top \boldsymbol{y}^{t+1} + \boldsymbol{z}^{t+1} - \boldsymbol{c} - \boldsymbol{A}^\top \nabla\varphi_t(\boldsymbol{y}^{t+1}) &= \frac{\boldsymbol{w}^t - \boldsymbol{w}^{t+1}}{\sigma_t} - \boldsymbol{A}^\top \nabla\varphi_t(\boldsymbol{y}^{t+1}), \end{aligned} \tag{4.20}$$

where the last equality follows from $\boldsymbol{w}$-update in (Ssnal). We then see from (4.20) that

$$\operatorname{dist}\left(\boldsymbol{0},\, \partial G_k(\boldsymbol{w}^{t+1})\right) \leqslant \left\|\frac{\boldsymbol{w}^t - \boldsymbol{w}^{t+1}}{\sigma_t} - \boldsymbol{A}^\top \nabla\varphi_t(\boldsymbol{y}^{t+1})\right\|. \tag{4.21}$$

By assumption, we know that (4.16) holds with $t = \bar{t}$, which together with (4.21) implies that

$$\operatorname{dist}\left(\boldsymbol{0},\, \partial G_k(\boldsymbol{w}^{\bar{t}+1})\right) \leqslant \epsilon_k \|\boldsymbol{w}^{\bar{t}+1} - \boldsymbol{x}^k\|.$$

This completes the proof. □

**Remark 4.6.** *In* Theorem 4.5, *if $\boldsymbol{x}^k = \boldsymbol{w}^0$ is the optimal solution of (4.1), we know from the optimality condition of (4.1) that $\boldsymbol{0} \in \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1} + \partial h(\boldsymbol{x}^k)$. This together with $\boldsymbol{\xi}^{k+1} \in \partial g(\boldsymbol{x}^k)$ implies that $\boldsymbol{x}^k$ is a stationary point of (1.1). In this case, we should terminate* Algorithm 1.

## 5 Choices of $B_k$

As we can see, the main computational cost of inner algorithms for solving (4.1), such as (FISTA) and (V-FISTA), lies in matrix multiplication $\boldsymbol{B}_k\boldsymbol{v}$ for any $\boldsymbol{v}$. This implies that it is practical to construct $\boldsymbol{B}_k$ with simple structures, for example, low-rank structure. On the other hand, to ensure the applicability of our convergence theory, $\boldsymbol{B}_k$ is expected to be positive definite, especially uniformly positive definite; see, Theorem 3.10. Based on such consideration, we present two variants of construction of approximate Hessian matrix $\boldsymbol{B}_k$ proposed in [3] as follows. Note that in the following, we consider the case when $f$ is strongly convex.

The first variant was proposed in [3, Section 4.1], called *modified SR1*, in which the inverse $\boldsymbol{H}_k = \boldsymbol{B}_k^{-1}$ was constructed as follows. Define

$$\boldsymbol{s}_k = \boldsymbol{x}^k - \boldsymbol{x}^{k-1}, \quad \boldsymbol{y}_k = \nabla f(\boldsymbol{x}^k) - \nabla f(\boldsymbol{x}^{k-1}), \quad \tau_k = \boldsymbol{s}_k^\top \boldsymbol{y}_k / \|\boldsymbol{y}_k\|^2. \tag{5.1}$$

Choose $\gamma \in (0, 1)$. For $k \geqslant 1$, let

$$\boldsymbol{H}_k = \gamma\tau_k \boldsymbol{I} + \boldsymbol{u}_k\boldsymbol{u}_k^\top \quad \text{with} \quad \boldsymbol{u}_k = (\boldsymbol{s}_k - \gamma\tau_k\boldsymbol{y}_k)/\sqrt{\langle \boldsymbol{s}_k - \gamma\tau_k\boldsymbol{y}_k,\, \boldsymbol{y}_k\rangle}.$$

Using Sherman-Morrison formula, we derive the corresponding update of $\boldsymbol{B}_k$ in *modified SR1* as

$$\boldsymbol{B}_k = \boldsymbol{H}_k^{-1} = \frac{1}{\gamma\tau_k}\left(\boldsymbol{I} - \frac{\boldsymbol{u}_k\boldsymbol{u}_k^\top}{\boldsymbol{u}_k^\top\boldsymbol{u}_k + \gamma\tau_k}\right). \tag{5.2}$$

Notice that this $\boldsymbol{B}_k$ is the type of "identity minus rank one". Then matrix multiplication cost $\boldsymbol{B}_k\boldsymbol{v}$ is cheap for any $\boldsymbol{v}$. Moreover, it has been shown in [3] that further assumption on the strong convexity of $f$ guarantees the uniformly positive definiteness of $\boldsymbol{B}_k$ in (5.2).

**Lemma 5.1.** [3, Lemma 4.1] *Suppose that $f$ is $\mu$-strongly convex and $L$-smooth. Let $\boldsymbol{B}_k$ be given in* (5.2)*. Then*

$$\frac{1-\gamma}{(1+\gamma)\mu^{-1} - 2\gamma L^{-1}}\boldsymbol{I} \preceq \boldsymbol{B}_k \preceq \frac{L}{\gamma}\boldsymbol{I} \qquad \forall\ k.$$

The second variant of construction of $\boldsymbol{B}_k$, called *sophisticated L-BFGS*, follows from [3, Section 5.1]. Let $\gamma \in (0, 1)$. Let $\boldsymbol{s}_k$, $\boldsymbol{y}_k$ and $\tau_k$ be defined as in (5.1). It is the type of "identity plus rank two": for $k \geqslant 1$,

$$\boldsymbol{B}_k = \frac{1}{\gamma\tau_k}\left(I - \frac{\boldsymbol{s}_k\boldsymbol{s}_k^\top}{\|\boldsymbol{s}_k\|^2} + \gamma\frac{\boldsymbol{y}_k\boldsymbol{y}_k^\top}{\|\boldsymbol{y}_k\|^2}\right). \tag{5.3}$$

Similarly, it has been shown in [3] that uniformly positive definiteness of $\boldsymbol{B}_k$ in (5.3) is guaranteed under the strong convexity of $f$.

**Lemma 5.2.** [3, Lemma 5.1] *Suppose that $f$ is $\mu$-strongly convex and $L$-smooth. Let $\boldsymbol{B}_k$ be given in* (5.3)*. Then*

$$\frac{1+\gamma}{(1+\gamma)(1+2\gamma)\mu^{-1} - (2+\gamma)\gamma L^{-1}}\boldsymbol{I} \preceq \boldsymbol{B}_k \preceq \frac{(1+\gamma)L}{\gamma}\boldsymbol{I} \qquad \forall\ k.$$

# 6 Numerical experiements

In this section, we conduct numerical experiments to test our iSQA$_{\text{major}}$, *i.e.*, Algorithm 1 for solving the $\ell_{1-2}$ regularized least squares problem and the truncated $\ell_1$ regularized least trimmed squares problem. All experiments are performed in Matlab R2019a on a 64-bit PC with 3.8 GHz Intel Core i5 Quad-Core and 8GB of DDR4 RAM.

## 6.1 The $\ell_{1-2}$ regularized least squares problem

We consider the $\ell_{1-2}$ regularized least squares problem [43]:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \quad F_{1\text{-}2}(\boldsymbol{x}) := \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2 + \lambda\left(\|\boldsymbol{x}\|_1 - \|\boldsymbol{x}\|\right), \tag{6.1}$$

where $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $\boldsymbol{b} \in \mathbb{R}^m$ and $\lambda > 0$ is the regularization parameter. We apply Algorithm 1 with $\boldsymbol{B}_k$ updated by (5.2). By Lemma 5.1, the positive definiteness of $\boldsymbol{B}_k$ is then guaranteed when $f$ is strongly convex. Consequently, we first reformulate (6.1) as

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \quad F_{1\text{-}2}(\boldsymbol{x}) := \underbrace{\frac{1}{2}\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2 + \frac{\tau}{2}\|\boldsymbol{x}\|^2}_{f(\boldsymbol{x})} + \underbrace{\lambda\|\boldsymbol{x}\|_1}_{h(\boldsymbol{x})} - \underbrace{\left(\lambda\|\boldsymbol{x}\| + \frac{\tau}{2}\|\boldsymbol{x}\|^2\right)}_{g(\boldsymbol{x})}, \tag{6.2}$$

where $\tau = 0.01$. We solve (6.2) by our iSQA$_{\text{major}}$ (Algorithm 1) with subproblem (3.1) solved by (V-FISTA)(**SQA**$_{\text{SR1}}$) and [3, Theorem 3.8 and Algorithm 5] (**SQA**$_{\text{bis}}$)[3], respectively. We compare

[3] This is applicable since our choice of $\boldsymbol{B}_k$ in (5.2) is the sum of a diagonal matrix and a rank-one matrix.

both methods with three first-order methods applied to (6.1): non-monotone APG (**nmAPG**) in [19], proximal difference-of-convex algorithm with extrapolation (**pDCA**$_\mathrm{e}$) in [39] and nonmonotone proximal gradient method (**NPG**) in [41].[4]

**Setting for SQA$_{\mathrm{SR1}}$ and SQA$_{\mathrm{bis}}$.** In Algorithm 1, we let $\boldsymbol{x}^0 = \mathbf{0}$, $\beta = 0.5$, $\sigma = 10^{-4}$, $M = 4$ and choose $\boldsymbol{B}_k$ as in (5.2) with $\gamma = 0.87$ for $k > 0$ and set $\boldsymbol{B}_0 = (\|\boldsymbol{A}\|^2 + \tau)\boldsymbol{I}$. We choose line search (LS$_1$) in Algorithm 1. We terminate Algorithm 1 when the running time exceeds some fixed time *maxtime* (second). In **SQA**$_{\mathrm{SR1}}$, we solve subproblem (3.1) by (V-FISTA), and terminate it when (4.2) holds with $\epsilon_k = 2\max\left\{1/(0.1\cdot k)^{1.2},\, 10^{-4}\right\} L_\phi$. In **SQA**$_{\mathrm{bis}}$, we solve (3.1) by calling the solver [5] in [3] with default setting.

**Setting for nmAPG, pDCA$_\mathrm{e}$ and NPG.** For these three methods, we take the same initial point $\boldsymbol{x}^0 = \mathbf{0}$ and termination criterion as in our methods **SQA**$_{\mathrm{SR1}}$ and **SQA**$_{\mathrm{bis}}$. Moreover, in **pDCA**$_\mathrm{e}$, we use a restart technique in [4, 30] and a restart frequency $T = 2000$ is used [6]. In **NPG**, the initial stepsize for each iteration is given by $\mu_0^0 = 1$ and for $k \geqslant 1$,

$$\mu_k^0 = \min\left\{\|\boldsymbol{A}\|^2 + 10^{-4},\, \max\left\{\max\left\{\frac{\langle \boldsymbol{x}^k - \boldsymbol{x}^{k-1},\, \boldsymbol{A}^\top\boldsymbol{A}(\boldsymbol{x}^k - \boldsymbol{x}^{k-1})\rangle}{\|\boldsymbol{x}^k - \boldsymbol{x}^{k-1}\|^2},\, 0.5\,\bar{\mu}_{k-1}\right\},\, 10^{-6}\right\}\right\},$$

where $\mu_k^i$ denotes the $k$th initial stepsize and $\bar{\mu}_k$ denotes the final stepsize of $k$th iteration after non-monotone line search.

**Data generation.** We first randomly generate a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ with i.i.d. standard Gaussian entries. Next, we uniformly at random generate an $s$-sparsity vector $\widehat{\boldsymbol{x}} \in \mathbb{R}^n$ and let $\boldsymbol{x} = \mathrm{sign}(\widehat{\boldsymbol{x}})$. Finally, we generate $\boldsymbol{b} = \boldsymbol{A}\,\boldsymbol{x} + 0.01 * \boldsymbol{\eta}$, where $\boldsymbol{\eta} \in \mathbb{R}^m$ is a random vector with i.i.d. standard Gaussian entries.

In the following, we consider two triples: $(n, m, s) = (3000, 900, 180)$ with $\lambda \in \{0.1,\, 0.01\}$ and $(n, m, s) = (5000, 1500, 300)$ with $\lambda \in \{0.01,\, 0.001\}$. For each group $(n, m, s, \lambda)$, we generate 10 independent trials and compare all five methods in terms of the average performance over the 10 trials. Specifically, we first follow the notation in [42] and define a normalized measure of the $k$th iteration as

$$e(k) := \frac{F_{1-2}(\boldsymbol{x}^k) - F_{1-2}^{\min}}{F_{1-2}(\boldsymbol{x}^0) - F_{1-2}^{\min}},$$

where $F_{1-2}^{\min}$ denotes the minimum function value obtained among all methods in one trial. Then, we let $T(k)$ denote the total computational time when $\boldsymbol{x}^k$ is obtained and further define

$$E(t) := \min\left\{e(k) : k \in \{i : T(i) \leqslant t\}\right\}. \tag{6.3}$$

We compare all the methods in terms of $E(t)$ averaged over 10 trials.

Figure 1 shows the performance of all five methods for triple $(n, m, s) = (3000, 900, 180)$ with $\lambda = 0.1$ and *maxtime* $= 10$, and $\lambda = 0.01$ and *maxtime* $= 25$. Figure 2 shows the performance of all five methods for triple $(n, m, s) = (5000, 1500, 300)$ with $\lambda = 0.01$ and *maxtime* $= 80$, and $\lambda = 0.001$ and *maxtime* $= 120$. One can see that our method **SQA**$_{\mathrm{SR1}}$ outperforms other methods in terms of CPU time needed for obtaining a solution with the same function value.

[4] We do not compare our methods with nonmonotone proximal gradient method with majorization (**NPG**$_{\mathrm{major}}$) in [25, Algorithm 2], because the performance of **NPG**$_{\mathrm{major}}$ is very similar to that of **NPG**; see [23, Section 5].

[5] The code can be downloaded in https://github.com/stephenbeckr/zeroSR1/tree/master/paperExperiments/Lasso.

[6] In [4], the restart frequency is 200. We replace the frequency by 2000 thus to improve the performance of **pDCA**$_\mathrm{e}$ in numerical experiments.

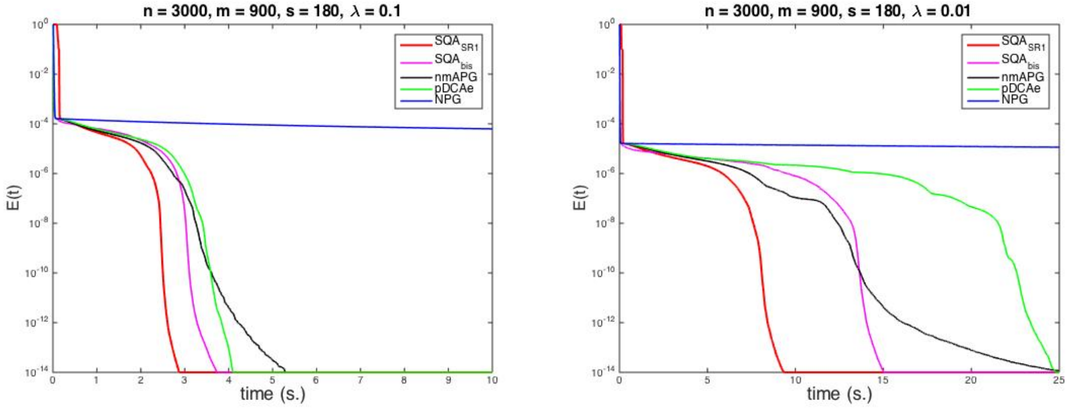

Figure 1: Comparison of all methods for $(n, m, s) = (3000, 900, 180)$.

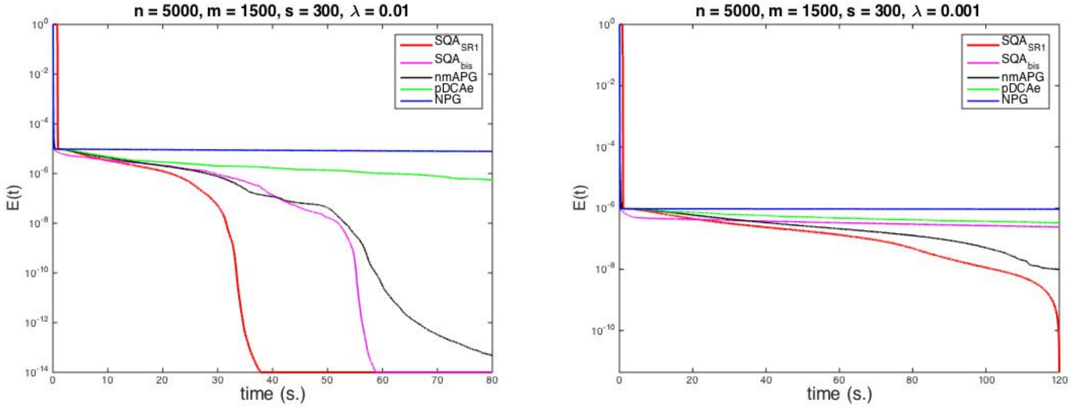

Figure 2: Comparison of all methods for $(n, m, s) = (5000, 1500, 300)$.

We also conduct numerical experiments on real data. Specifically, we consider problem (6.1) with $\lambda = 0.005$, and $\boldsymbol{A}$ and $\boldsymbol{b}$ from four sets of real data: leukemia data with 3051 genes and 72 samples ($m = 72$, $n = 3051$), lymph node status data with 4514 genes and 148 samples ($m = 148$, $n = 4514$), breast cancer prognosis data with 4919 genes and 76 samples ($m = 76$, $n = 4919$) and colon tumor gene expression data with 2000 genes and 62 samples ($m = 62$, $n = 2000$). We use these four data sets in the same way as in [17, Section 4.1]. Figure 3 shows the performance of all five methods on each data set. One can see that for leukemia data, lymph node status data and colon tumor gene expression data, our proposed methods (**SQA**$_{\text{SR1}}$ or **SQA**$_{\text{bis}}$) outperforms other methods, while method **nmAPG** outperforms all other methods for breast cancer prognosis data.

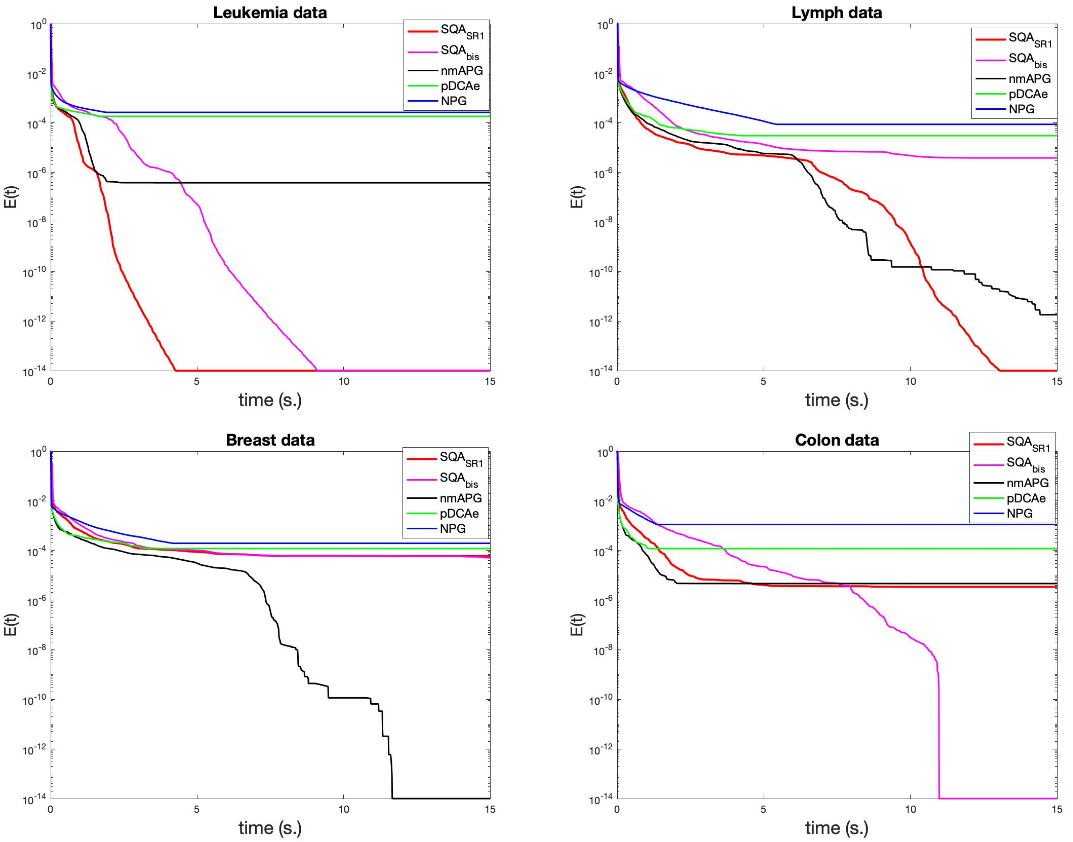

Figure 3: Performance comparion on four sets of real data.

## 6.2 Truncated $\ell_1$ regularized least trimmed squares problem

We consider the following truncated $\ell_1$ regularized least trimmed squares problem given in [24] for simultaneous sparse recovery and outlier detection:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n,\, \boldsymbol{z}\in\mathbb{R}^m} \quad F_{\mathrm{trc}}(\boldsymbol{x},\, \boldsymbol{z}) := \frac{1}{2}\left\|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{z}-\boldsymbol{b}\right\|^2 + \delta_\Omega(\boldsymbol{z}) + \lambda\left\|\boldsymbol{x}\right\|_1 - \lambda\mu\sum_{i=1}^{k}\left|\boldsymbol{x}_{[i]}\right|, \tag{6.4}$$

where $\boldsymbol{A}\in\mathbb{R}^{m\times n}$, $\boldsymbol{b}\in\mathbb{R}^m$, $\Omega = \{\boldsymbol{z}\in\mathbb{R}^m : \|\boldsymbol{z}\|_0 \leqslant r\}$, $\mu\in(0,1)$, $\lambda>0$, $k<n$ and $\boldsymbol{x}_{[i]}$ denotes the $i$th largest element of $\boldsymbol{x}$ in magnitude. In order to apply our Algorithm 1, we first reformulate (6.4) as formulation (1.1); see [24, equation 37]:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n,\, \boldsymbol{z}\in\mathbb{R}^m} \quad F_{\mathrm{trc}}(\boldsymbol{x},\, \boldsymbol{z}) = \frac{1}{2}\left\|\boldsymbol{A}\boldsymbol{x}\right\|^2 + \lambda\left\|\boldsymbol{x}\right\|_1 - \left(\lambda\mu\sum_{i=1}^{k}\left|\boldsymbol{x}_{[i]}\right| + Q(\boldsymbol{x})\right), \tag{6.5}$$

where $Q$ is continuous and convex with $\boldsymbol{A}^\top(\bar{\boldsymbol{z}}+\boldsymbol{b})\in\partial Q(\boldsymbol{x})$ for any $\bar{\boldsymbol{z}}\in P_\Omega(\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b})$. Similarly, to guarantee the positive definiteness of $\boldsymbol{B}_k$ in Algorithm 1 when $\boldsymbol{B}_k$ is updated as in (5.2) or (5.3), we reformulate (6.5) as

$$\min_{\boldsymbol{x}\in\mathbb{R}^n,\, \boldsymbol{z}\in\mathbb{R}^m} \quad F_{\mathrm{trc}}(\boldsymbol{x},\, \boldsymbol{z}) = \underbrace{\frac{1}{2}\left\|\boldsymbol{A}\boldsymbol{x}\right\|^2 + \frac{\tau}{2}\|\boldsymbol{x}\|^2}_{f(\boldsymbol{x})} + \underbrace{\lambda\left\|\boldsymbol{x}\right\|_1}_{h(\boldsymbol{x})} - \underbrace{\left(\frac{\tau}{2}\|\boldsymbol{x}\|^2 + \lambda\mu\sum_{i=1}^{k}\left|\boldsymbol{x}_{[i]}\right| + Q(\boldsymbol{x})\right)}_{g(\boldsymbol{x})}, \tag{6.6}$$

where $\tau = 0.01$. Since the proximal mapping of either the DC regularizer in (6.5) or $h - g$ in (6.6) is no longer easy to obtain, neither **nmAPG** nor **NPG** is applicable. On the other hand, a subgradient of $g$ in (6.6) can be obtained easily [7]. In the following, we will therefore compare three variants of our method applied to (6.6) with $\mathbf{pDCA}_{\mathrm{e}}$[8] applied to (6.5): $\mathbf{SQA}_{\mathrm{SR1}}$, $\mathbf{SQA}_{\mathrm{bis}}$ and $\mathbf{SQA}_{\text{L-BFGS}}$ ($\boldsymbol{B}_k$ is taken as in (5.3) in Algorithm 1 with subproblem solved by (V-FISTA)). We take the same setting for the three variants as in Section 6.1.

**Data generation**. We first randomly generate a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ with i.i.d. standard Gaussian entries. Next, we uniformly at random generate an $s$-sparsity vector $\boldsymbol{x} \in \mathbb{R}^n$ and let $\boldsymbol{z} \in \mathbb{R}^m$ be the vector with the last $r$ entries being 8 and others being 0. Finally, we generate $\boldsymbol{b} = \boldsymbol{A}\,\boldsymbol{x} - \boldsymbol{z} + 0.01 * \boldsymbol{\eta}$, where $\boldsymbol{\eta} \in \mathbb{R}^m$ is a random vector with i.i.d. standard Gaussian entries. In (6.4), we let $\mu = 0.99$ and $k = 0.8s$.

In the following, we consider two groups $(n, m, s, r) = (3000, 600, 150, 30)$ and $(n, m, s, r) = (9000, 1800, 450, 90)$ with $\lambda = 5 \times 10^{-4}$. For each data $(n, m, s, r, \lambda)$, we generate 10 independent trials and compare our three methods with $\mathbf{pDCA}_{\mathrm{e}}$ in terms of the average $E(t)$ defined as in (6.3) over the 10 trials.

Figure 4 and Figure 5 show the performance on $E(t)$ of all methods and the recovery of $\mathbf{SQA}_{\mathrm{SR1}}$ in one random trial (solution of $\mathbf{SQA}_{\mathrm{SR1}}$ when it terminates is marked by asterisks and true signal $\boldsymbol{x}$ is marked by circle) for group $(n, m, s, r) = (3000, 600, 150, 30)$ and $(n, m, s, r) = (9000, 1800, 450, 90)$, respectively. One can see from the performance that $\mathbf{SQA}_{\mathrm{SR1}}$ generally outperforms $\mathbf{SQA}_{\mathrm{bis}}$ and $\mathbf{SQA}_{\text{L-BFGS}}$ and is comparable with $\mathbf{pDCA}_{\mathrm{e}}$ in terms of CPU time needed for obtaining a solution with the same function value. In Figure 6, we also report the numbers of outer iterations for all methods (when they terminate) over the 10 independent trials. One can see that within the same CPU time, due to the simplicity of subproblem, $\mathbf{pDCA}_{\mathrm{e}}$ can run more iterations than $\mathbf{SQA}_{\mathrm{SR1}}$, $\mathbf{SQA}_{\mathrm{bis}}$ and $\mathbf{SQA}_{\text{L-BFGS}}$.

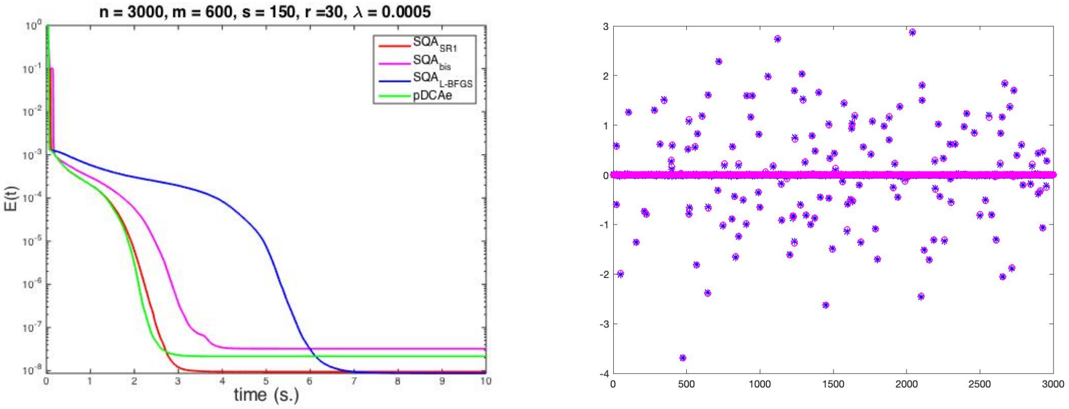

Figure 4: Comparison of all methods for $(n, m, s, r) = (3000, 600, 150, 30)$ and signal recovery of $\mathbf{SQA}_{\mathrm{SR1}}$.

[7] Indeed, for any $\boldsymbol{x} \in \mathbb{R}^n$, a subgradient in $\partial g(\boldsymbol{x})$ is $\boldsymbol{\xi} = \tau\boldsymbol{x} + \lambda\mu\,\boldsymbol{u} + \boldsymbol{A}^\top(\boldsymbol{v} + \boldsymbol{b})$, where $\boldsymbol{u}$ and $\boldsymbol{v}$ are given by

$$\boldsymbol{u}_i = \begin{cases} \operatorname{sign}(\boldsymbol{x}_i), & \text{if } i \in C_u \\ 0. & \text{else.} \end{cases} \qquad \boldsymbol{v}_i = \begin{cases} (\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})_i, & \text{if } i \in C_v \\ 0. & \text{else.} \end{cases}$$

Here, $C_u$ is an arbitrary index set corresponding to the largest $k$ elements of $\boldsymbol{x}$ in magnitude, and $C_v$ is an arbitrary index set corresponding to the largest $r$ elements of $\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}$ in magnitude.

[8] It has been shown in [24] that sequence generated by $\mathbf{pDCA}_{\mathrm{e}}$ converges locally linearly to a stationary point of (6.5). Moreover, it has been shown in [24] that $\mathbf{pDCA}_{\mathrm{e}}$ outperforms $\mathbf{NPG}_{\mathrm{major}}$ proposed in [25, Algorithm 2]. Consequently, we do not compare our method with $\mathbf{NPG}_{\mathrm{major}}$ here.

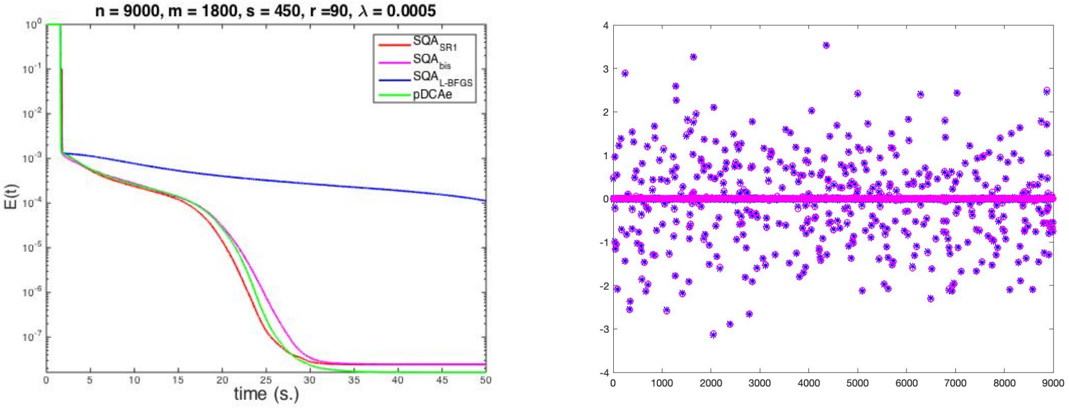

Figure 5: Comparison of all methods for $(n, m, s, r) = (9000, 1800, 450, 90)$ and signal recovery of $\mathbf{SQA}_{\mathrm{SR1}}$.

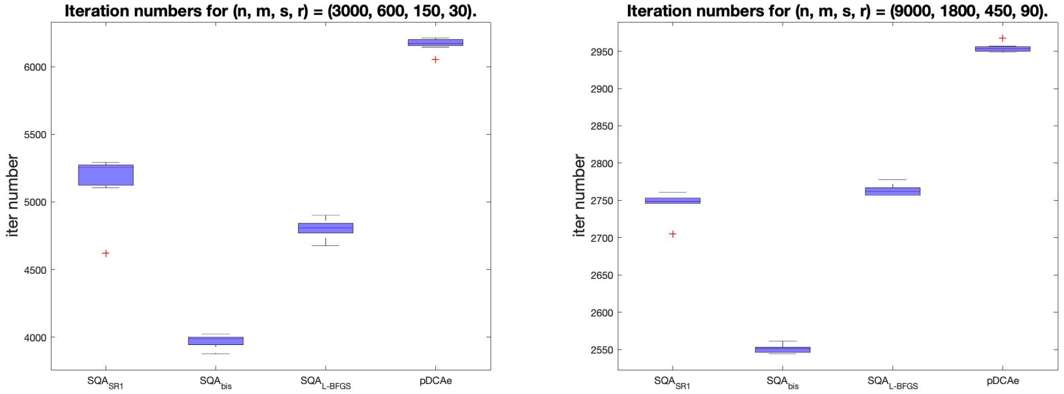

Figure 6: Iteration numbers for all methods.

# Acknowledgements

The authors would like to thank Lei Yang for his constructive suggestions during the writing of this paper. The research of the first author is supported in part by JSPS KAKENHI Grants No.19H04069. The research of the second author is supported in part by JSPS KAKENHI Grants No. 17H01699 and 19H04069.

# A Proof of Lemma 3.2

*Proof.* First, for any $\alpha \in (0, 1]$, we see from the $L$-smoothness of $f$ and the convexity of $g$ with $\boldsymbol{\xi}^{k+1} \in \partial g(\boldsymbol{x}^k)$ that

$$\begin{aligned} F(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - F(\boldsymbol{x}^k) &= f(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - f(\boldsymbol{x}^k) + h(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - h(\boldsymbol{x}^k) - \big(g(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - g(\boldsymbol{x}^k)\big) \\ &\leqslant \alpha\, \nabla f(\boldsymbol{x}^k)^\top \boldsymbol{d}_k + \frac{\alpha^2 L}{2}\, \|\boldsymbol{d}_k\|^2 + h(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - h(\boldsymbol{x}^k) - \alpha\, {\boldsymbol{\xi}^{k+1}}^\top \boldsymbol{d}_k \\ &= \frac{\alpha^2 L}{2}\, \|\boldsymbol{d}_k\|^2 + \alpha \left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{d}_k \right\rangle + h(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - h(\boldsymbol{x}^k) \\ &= \frac{\alpha^2 L}{2}\, \|\boldsymbol{d}_k\|^2 + \triangle_k(\alpha). \end{aligned} \tag{A.1}$$

On the other hand, the convexity of $h$ and $\alpha \in (0, 1]$ yield

$$\begin{aligned} \triangle_k(\alpha) &= \alpha \left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{d}_k \right\rangle + h(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - h(\boldsymbol{x}^k) \\ &= \alpha \left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{d}_k \right\rangle + h\left(\alpha(\boldsymbol{x}^k + \boldsymbol{d}_k) + (1-\alpha)\boldsymbol{x}^k\right) - h(\boldsymbol{x}^k) \\ &\leqslant \alpha \left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{d}_k \right\rangle + \alpha\, h(\boldsymbol{x}^k + \boldsymbol{d}_k) + (1-\alpha) h(\boldsymbol{x}^k) - h(\boldsymbol{x}^k) \\ &= \alpha \left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{d}_k \right\rangle + \alpha\, h(\boldsymbol{u}^k) - \alpha\, h(\boldsymbol{x}^k) = \alpha\, \triangle_{k,1}. \end{aligned} \tag{A.2}$$

To show the well-definedness of three termination criteria (LS$_1$), (LS$_2$) and (LS$_3$), we see from (A.1) and (A.2) that it suffices to show that $\triangle_{k,1}$ could be bounded by a proper multiple of $\|\boldsymbol{d}_k\|^2$. To proceed, we first see from the convexity of $h$ and $\boldsymbol{B}_k \succ \boldsymbol{0}$ that $G_k$ is strongly convex with modulus $\lambda_{\min}(\boldsymbol{B}_k)$. On the other hand, we know from (3.3) and $\boldsymbol{d}_k = \boldsymbol{u}^k - \boldsymbol{x}^k$ that there exists some $\boldsymbol{w}_k \in \partial G_k(\boldsymbol{u}^k)$ such that $\|\boldsymbol{w}_k\| \leqslant \epsilon_k \|\boldsymbol{d}_k\|$. This together with the strong convexity of $G_k$ with modulus $\lambda_{\min}(\boldsymbol{B}_k)$ implies that

$$G_k(\boldsymbol{u}^k) - G_k(\boldsymbol{x}^k) \leqslant -\left\langle \boldsymbol{w}_k,\, \boldsymbol{x}^k - \boldsymbol{u}^k \right\rangle - \frac{\lambda_{\min}(\boldsymbol{B}_k)}{2} \|\boldsymbol{x}^k - \boldsymbol{u}^k\|^2 \leqslant \left(\epsilon_k - \frac{\lambda_{\min}(\boldsymbol{B}_k)}{2}\right) \|\boldsymbol{d}_k\|^2. \tag{A.3}$$

Moreover, we know from (A.3) and $\boldsymbol{d}_k = \boldsymbol{u}^k - \boldsymbol{x}^k$ that

$$\begin{aligned} &\left(\epsilon_k - \frac{\lambda_{\min}(\boldsymbol{B}_k)}{2}\right) \|\boldsymbol{d}_k\|^2 \geqslant G_k(\boldsymbol{u}^k) - G_k(\boldsymbol{x}^k) \\ &= \left\langle \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1},\, \boldsymbol{d}_k \right\rangle + \frac{1}{2}(\boldsymbol{u}^k - \boldsymbol{x}^k)^\top \boldsymbol{B}_k (\boldsymbol{u}^k - \boldsymbol{x}^k) + h(\boldsymbol{u}^k) - h(\boldsymbol{x}^k) \\ &= \triangle_{k,1} + \frac{1}{2}\boldsymbol{d}_k^\top \boldsymbol{B}_k \boldsymbol{d}_k \geqslant \triangle_{k,1} + \frac{\lambda_{\min}(\boldsymbol{B}_k)}{2} \|\boldsymbol{d}_k\|^2, \end{aligned}$$

which implies that

$$\triangle_{k,1} \leqslant (\epsilon_k - \lambda_{\min}(\boldsymbol{B}_k))\, \|\boldsymbol{d}_k\|^2. \tag{A.4}$$

Now we are ready to prove the well-definedness of the three termination criteria. We first consider the termination criteria (LS$_1$) and (LS$_2$) when $\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k > 0$. For any $\alpha \in (0, 1]$, we have

$$\begin{aligned} &F(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - \max_{[k-M]_+ \leqslant j \leqslant k} F(\boldsymbol{x}^j) - \sigma\alpha\, \triangle_{k,1} \leqslant F(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - F(\boldsymbol{x}^k) - \sigma\alpha\, \triangle_{k,1} \\ &\leqslant \frac{\alpha^2 L}{2}\, \|\boldsymbol{d}_k\|^2 + \alpha\, \triangle_{k,1} - \sigma\alpha\, \triangle_{k,1} \leqslant \frac{L}{2}\alpha\left(\alpha - 2(1-\sigma)\left(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k\right)/L\right) \|\boldsymbol{d}_k\|^2, \end{aligned} \tag{A.5}$$

where the second inequality follows from (A.1) and (A.2), and the last inequality follows from (A.4). Notice that $\sigma \in (0, 1)$ and $\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k > 0$. We see from (A.5) that its left-hand side is nonpositive when $\alpha \in (0,\, 2(1-\sigma)\left(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k\right)/L]$. This proves the well-definedness of (LS$_1$). Similarly, we consider termination criterion (LS$_2$). By (A.1), (A.2) and (A.4), we have

$$\begin{aligned} &F(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - \max_{[k-M]_+ \leqslant j \leqslant k} F(\boldsymbol{x}^j) - \sigma \triangle_k(\alpha) \leqslant F(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - F(\boldsymbol{x}^k) - \sigma \triangle_k(\alpha) \\ &\leqslant \frac{\alpha^2 L}{2} \left\|\boldsymbol{d}_k\right\|^2 + \triangle_k(\alpha) - \sigma \triangle_k(\alpha) \leqslant \frac{\alpha^2 L}{2} \left\|\boldsymbol{d}_k\right\|^2 + \alpha(1-\sigma)\triangle_{k,1} \\ &\leqslant \frac{L}{2}\alpha \left(\alpha - 2(1-\sigma)\left(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k\right)/L\right) \|\boldsymbol{d}_k\|^2. \end{aligned} \tag{A.6}$$

This proves the well-definedness of (LS$_2$). Finally, we see from (A.1), (A.2) and (A.4) that

$$\begin{aligned} &F(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - \max_{[k-M]_+ \leqslant j \leqslant k} F(\boldsymbol{x}^j) + \sigma\alpha\|\boldsymbol{d}_k\|^2 \leqslant F(\boldsymbol{x}^k + \alpha\, \boldsymbol{d}_k) - F(\boldsymbol{x}^k) + \sigma\alpha\|\boldsymbol{d}_k\|^2 \\ &\leqslant \frac{\alpha^2 L}{2} \left\|\boldsymbol{d}_k\right\|^2 + \triangle_k(\alpha) + \sigma\alpha\|\boldsymbol{d}_k\|^2 \leqslant \frac{\alpha^2 L}{2} \left\|\boldsymbol{d}_k\right\|^2 + \alpha\left(\epsilon_k - \lambda_{\min}(\boldsymbol{B}_k)\right)\left\|\boldsymbol{d}_k\right\|^2 + \sigma\alpha\|\boldsymbol{d}_k\|^2 \\ &= \frac{L}{2}\alpha \left(\alpha - 2(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k - \sigma)/L\right) \|\boldsymbol{d}_k\|^2. \end{aligned} \tag{A.7}$$

This proves the well-definedness of (LS$_3$) when $\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k > \sigma$. Furthermore, we have (3.4) by noticing (A.5), (A.6), (A.7) and the line-search rule $\alpha_k \in \left\{\beta^i : i = 0, 1, \ldots\right\}$. This completes the proof. □

# B Proof of Theorem 3.3

*Proof.* For simplicity of notation, we define

$$t(k) := \operatorname*{arg\,max}_{[k-M]_+ \leqslant j \leqslant k} F(\boldsymbol{x}^j). \tag{B.1}$$

We know from (3.4) that in case of (LS$_1$) or (LS$_2$) with $\delta > 0$ we have $\alpha_k \geqslant c_1 > 0$ with $c_1 := \min\{1,\, 2\beta(1-\sigma)\delta/L\}$, and in case of (LS$_3$) with $\delta > \sigma$, we have $\alpha_k \geqslant c_2 > 0$ with $c_2 := \min\{1,\, 2\beta(\delta - \sigma)/L\}$. Furthermore, for the three line-search criteria we have

$$F(\boldsymbol{x}^{k+1}) = F(\boldsymbol{x}^k + \alpha_k \boldsymbol{d}_k) \leqslant F(\boldsymbol{x}^{t(k)}) + \begin{cases} \sigma\alpha_k\triangle_{k,1} \overset{\text{(a)}}{\leqslant} -\sigma c_1 \delta \|\boldsymbol{d}_k\|^2 & for\ (\text{LS}_1), \\ \sigma\triangle_k(\alpha_k) \overset{\text{(b)}}{\leqslant} \sigma\alpha_k\triangle_{k,1} \leqslant -\sigma c_1\delta\|\boldsymbol{d}_k\|^2 & for\ (\text{LS}_2), \\ -\sigma\alpha_k\|\boldsymbol{d}_k\|^2 \overset{\text{(c)}}{\leqslant} -\sigma c_2\|\boldsymbol{d}_k\|^2 & for\ (\text{LS}_3). \end{cases}$$

where (a) follows from (A.4), $\delta = \inf_k \left(\lambda_{\min}(\boldsymbol{B}_k) - \epsilon_k\right)$ and $\alpha_k \geqslant c_1$ in case of (LS$_1$) with $\delta > 0$, (b) follows from (A.2) and (c) follows from $\alpha_k \geqslant c_2$ in case of (LS$_3$) with $\delta > \sigma$. Consequently, for each line search there exists some $c > 0$ such that

$$F(\boldsymbol{x}^{k+1}) \leqslant F(\boldsymbol{x}^{t(k)}) - c\|\boldsymbol{d}_k\|^2. \tag{B.2}$$

Next, we prove the three statements based on (B.2). First, we know from (B.2) that

$$F(\boldsymbol{x}^k) \leqslant F(\boldsymbol{x}^0) < \infty,$$

which together with the level-boundedness of $F$ gives the boundedness of $\{\boldsymbol{x}^k\}$. This proves (i).

Next, we prove (ii). We see from (B.2) that sequence $\{F(\boldsymbol{x}^{t(k)})\}$ is non-increasing:

$$\begin{aligned} &F\left(\boldsymbol{x}^{t(k+1)}\right) = \max_{[k+1-M]_+ \leqslant j \leqslant k+1} F(\boldsymbol{x}^j) = \max\left\{\max_{[k+1-M]_+ \leqslant j \leqslant k} F(\boldsymbol{x}^j),\, F\left(\boldsymbol{x}^{k+1}\right)\right\} \\ &\leqslant \max\left\{F\left(\boldsymbol{x}^{t(k)}\right),\, F\left(\boldsymbol{x}^{t(k)}\right) - c\left\|\boldsymbol{d}_k\right\|^2\right\} \leqslant F\left(\boldsymbol{x}^{t(k)}\right). \end{aligned}$$

This together with the level-boundedness of $F$ implies that there exists some $\bar{F}$ such that

$$\lim_{k\to\infty} F\big(\boldsymbol{x}^{t(k)}\big) = \bar{F}. \tag{B.3}$$

Next, we prove that the following relationships hold for all $j \geqslant 1$ by induction:

$$\lim_{k\to\infty} \boldsymbol{d}_{t(k)-j} = \boldsymbol{0}, \tag{B.4a}$$

$$\lim_{k\to\infty} F\big(\boldsymbol{x}^{t(k)-j}\big) = \bar{F}. \tag{B.4b}$$

We first prove that (B.4a) and (B.4b) hold for $j = 1$. Replacing $k$ in (B.2) by $t(k) - 1$, we obtain

$$F\big(\boldsymbol{x}^{t(k)}\big) \leqslant F\big(\boldsymbol{x}^{t(t(k)-1)}\big) - c\left\|\boldsymbol{d}_{t(k)-1}\right\|^2. \tag{B.5}$$

Rearranging (B.5) and letting $k \to \infty$, using (B.3) and $t(k) \to \infty$ while $k \to \infty$ (when $k \geqslant M$ we have $t(k) \in [k - M, k]$), we see that $\lim_{k\to\infty} \boldsymbol{d}_{t(k)-1} = \boldsymbol{0}$. This proves that (B.4a) holds for $j = 1$. Furthermore, we see from (B.3) that

$$\bar{F} = \lim_{k\to\infty} F\big(\boldsymbol{x}^{t(k)}\big) = \lim_{k\to\infty} F\big(\boldsymbol{x}^{t(k)-1} + \alpha_{t(k)-1}\boldsymbol{d}_{t(k)-1}\big) = \lim_{k\to\infty} F\big(\boldsymbol{x}^{t(k)-1}\big),$$

where the last equality follows from $\alpha_k \leqslant 1$, $\lim_{k\to\infty} \boldsymbol{d}_{t(k)-1} = \boldsymbol{0}$ and the uniform continuity of $F$ on the closure of the sequence $\{\boldsymbol{x}^k\}$ (This is because $h$ is continuous on its domain, $\boldsymbol{x}^k \in \operatorname{dom} h$ and sequence $\{\boldsymbol{x}^k\}$ is bounded). This proves that (B.4b) holds for $j = 1$.

Now we assume that (B.4a) and (B.4b) hold for some $J \geqslant 1$, i.e., $\lim_{k\to\infty} \boldsymbol{d}_{t(k)-J} = \boldsymbol{0}$ and $\lim_{k\to\infty} F\big(\boldsymbol{x}^{t(k)-J}\big) = \bar{F}$. Replacing $k$ in (B.2) by $t(k) - J - 1$, we obtain

$$F\big(\boldsymbol{x}^{t(k)-J}\big) \leqslant F\big(\boldsymbol{x}^{t(t(k)-J-1)}\big) - c\left\|\boldsymbol{d}_{t(k)-J-1}\right\|^2. \tag{B.6}$$

Rearranging (B.6) and letting $k \to \infty$, using assumption $\lim_{k\to\infty} F\big(\boldsymbol{x}^{t(k)-J}\big) = \bar{F}$ and (B.3) with $t(k) - J - 1 \to \infty$ while $k \to \infty$ (when $k \geqslant M$ we have $t(k) \in [k - M, k]$), we see that $\lim_{k\to\infty} \boldsymbol{d}_{t(k)-J-1} = \boldsymbol{0}$. This proves that (B.4a) holds for $j = J + 1$. Similarly, we have

$$\bar{F} = \lim_{k\to\infty} F\big(\boldsymbol{x}^{t(k)-J}\big) = \lim_{k\to\infty} F\big(\boldsymbol{x}^{t(k)-J-1} + \alpha_{t(k)-J-1}\boldsymbol{d}_{t(k)-J-1}\big) = \lim_{k\to\infty} F\big(\boldsymbol{x}^{t(k)-J-1}\big),$$

which proves that (B.4b) holds for $J + 1$. This completes the induction.

Now we are ready to prove (ii). Note from (B.1) that when $k \geqslant M$, we have $k - M \leqslant t(k) \leqslant k$. Thus, for any $k$, we have $k - M - 1 = t(k) - j_k$ for some $j_k \in [1, M + 1]$. Therefore, it follows from (B.4a) that

$$\boldsymbol{0} = \lim_{k\to\infty} \boldsymbol{d}_{t(k)-j_k} = \lim_{k\to\infty} \boldsymbol{d}_{k-M-1} = \lim_{k\to\infty} \boldsymbol{d}_k,$$

which together with $\boldsymbol{x}^{k+1} - \boldsymbol{x}^k = \alpha_k \boldsymbol{d}_k$ and $\alpha_k \leqslant 1$ proves (ii).

Finally, we prove (iii). Since $\{\boldsymbol{x}^k\}$ is bounded, there exists some convergence subsequence, say $\{\boldsymbol{x}^{k_j}\}$, which satisfies $\lim_{j\to\infty} \boldsymbol{x}^{k_j} = \boldsymbol{x}^*$. On the other hand, since the set $\partial G_k(\boldsymbol{u}^k)$ is closed, we see from (3.3) that there exists some $\boldsymbol{w}_k \in \operatorname{dom} h$ satisfying $\|\boldsymbol{w}_k\| \leqslant \epsilon_k \|\boldsymbol{u}^k - \boldsymbol{x}^k\|$ and

$$\boldsymbol{w}_k \in \partial G_k(\boldsymbol{u}^k) = \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1} + \boldsymbol{B}_k(\boldsymbol{u}^k - \boldsymbol{x}^k) + \partial h(\boldsymbol{u}^k).$$

This combined with $\boldsymbol{d}_k = \boldsymbol{u}^k - \boldsymbol{x}^k$ further implies that

$$\boldsymbol{w}_{k_j} - \boldsymbol{B}_{k_j}\boldsymbol{d}_{k_j} \in \nabla f(\boldsymbol{x}^{k_j}) - \boldsymbol{\xi}^{k_j+1} + \partial h(\boldsymbol{x}^{k_j} + \boldsymbol{d}_{k_j}). \tag{B.7}$$

Due to $\boldsymbol{\xi}^{k+1} \in \partial g(\boldsymbol{x}^k)$, the boundedness of $\{\boldsymbol{x}^k\}$ and the convexity and continuity of $g$, we see that $\{\boldsymbol{\xi}^k\}$ is bounded. Thus, by passing to a further subsequence if necessary, without loss of generality, we assume that $\boldsymbol{\xi}^* := \lim_{j\to\infty} \boldsymbol{\xi}^{k_j+1}$ exists and thus $\boldsymbol{\xi}^* \in \partial g(\boldsymbol{x}^*)$ due to $\boldsymbol{\xi}^{k_j+1} \in \partial g(\boldsymbol{x}^{k_j})$ and the closedness of $\partial g$. On the other hand, we see from the boundedness of $\{\boldsymbol{B}_k\}$ and the assumption $\delta > 0$ that $\{\epsilon_k\}$ is bounded, which further gives $\|\boldsymbol{w}_k\| \leqslant \epsilon_k \|\boldsymbol{u}^k - \boldsymbol{x}^k\| = \epsilon_k \|\boldsymbol{d}_k\| \to 0$. Now passing to the limit in (B.7) and using $\|\boldsymbol{w}_k\| \to 0$, $\|\boldsymbol{d}_k\| \to 0$, the boundedness of $\{\boldsymbol{B}_k\}$, the $L$-smoothness of $f$ and the closedness of $\partial h$, we see that

$$\mathbf{0} \in \nabla f(\boldsymbol{x}^*) + \partial h(\boldsymbol{x}^*) - \partial g(\boldsymbol{x}^*).$$

This proves (iii) and completes the proof. □

## C Proof of Lemma 3.6

*Proof.* Since $\boldsymbol{x}_{\boldsymbol{I}}^k$ is a global minimizer of the optimization problem in (3.6), we have

$$\mathbf{0} \in \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1} + \boldsymbol{x}_{\boldsymbol{I}}^k - \boldsymbol{x}^k + \partial h(\boldsymbol{x}_{\boldsymbol{I}}^k). \tag{C.1}$$

If $\boldsymbol{x}_{\boldsymbol{I}}^k = \boldsymbol{x}^k$, we see from (C.1) that $\mathbf{0} \in \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1} + \partial h(\boldsymbol{x}^k)$, which together with $\boldsymbol{\xi}^{k+1} \in \partial g(\boldsymbol{x}^k)$ proves that $\boldsymbol{x}^k$ is a stationary point of (1.1). On the other hand, if $\boldsymbol{x}^k$ is a stationary point of (1.1) and $\partial g(\boldsymbol{x}^k)$ is a singleton, these together with $\boldsymbol{\xi}^{k+1} \in \partial g(\boldsymbol{x}^k)$ give

$$\mathbf{0} \in \nabla f(\boldsymbol{x}^k) - \partial g(\boldsymbol{x}^k) + \partial h(\boldsymbol{x}^k) = \nabla f(\boldsymbol{x}^k) - \boldsymbol{\xi}^{k+1} + \partial h(\boldsymbol{x}^k). \tag{C.2}$$

Now, using the monotonicity of operator $\partial h$ with (C.1) and (C.2), we further have

$$\langle \boldsymbol{x}^k - \boldsymbol{x}_{\boldsymbol{I}}^k, \boldsymbol{x}_{\boldsymbol{I}}^k - \boldsymbol{x}^k \rangle \geqslant 0,$$

which implies that $\boldsymbol{x}_{\boldsymbol{I}}^k = \boldsymbol{x}^k$. This completes the proof. □

## D Proof of Theorem 4.2

*Proof.* First, we consider (FISTA). Since $\boldsymbol{B}_k \succ \mathbf{0}$, we know from the convexity of $h$ that $G_k(\cdot)$ is strongly convex with modulus $\lambda_{\min}(\boldsymbol{B}_k)$. We then further have

$$\frac{\lambda_{\min}(\boldsymbol{B}_k)}{2}\|\boldsymbol{z}^\ell - \boldsymbol{z}^*\|^2 \leqslant G_k(\boldsymbol{z}^\ell) - G_k(\boldsymbol{z}^*) \leqslant \frac{2L_\phi}{(\ell+1)^2}\|\boldsymbol{z}^0 - \boldsymbol{z}^*\|^2, \tag{D.1}$$

where the last inequality follows from [8, Theorem 4.4]. Furthermore, inequality (D.1) together with the definition of $c_1$ in (4.5) implies that

$$\|\boldsymbol{z}^\ell - \boldsymbol{z}^*\| \leqslant \frac{2\sqrt{L_\phi}}{(\ell+1)\sqrt{\lambda_{\min}(\boldsymbol{B}_k)}}\|\boldsymbol{z}^0 - \boldsymbol{z}^*\| = \frac{c_1}{\ell+1}\|\boldsymbol{z}^0 - \boldsymbol{z}^*\|. \tag{D.2}$$

Furthermore, we have that for $\ell \geqslant 2$,

$$\begin{aligned}
&\|\boldsymbol{z}^\ell - \boldsymbol{y}^\ell\| = \left\|\boldsymbol{z}^\ell - \boldsymbol{z}^{\ell-1} - \frac{\theta_{\ell-1}-1}{\theta_\ell}(\boldsymbol{z}^{\ell-1} - \boldsymbol{z}^{\ell-2})\right\| \leqslant \|\boldsymbol{z}^\ell - \boldsymbol{z}^{\ell-1}\| + \|\boldsymbol{z}^{\ell-1} - \boldsymbol{z}^{\ell-2}\| \\
&\leqslant \|\boldsymbol{z}^\ell - \boldsymbol{z}^*\| + 2\|\boldsymbol{z}^{\ell-1} - \boldsymbol{z}^*\| + \|\boldsymbol{z}^{\ell-2} - \boldsymbol{z}^*\| \leqslant \frac{4c_1}{\ell-1}\|\boldsymbol{z}^0 - \boldsymbol{z}^*\|,
\end{aligned} \tag{D.3}$$

where the first equality follows from the $\boldsymbol{y}$-update in (FISTA) and the last inequality follows from (D.2). Notice that $\boldsymbol{z}^0 = \boldsymbol{x}^k \neq \boldsymbol{z}^*$. Using (D.2) and (D.3), we further have for $\ell \geqslant \max\{2, c_1\}$ that

$$\frac{\|\boldsymbol{z}^\ell - \boldsymbol{y}^\ell\|}{\|\boldsymbol{z}^\ell - \boldsymbol{z}^0\|} \leqslant \frac{4c_1}{\ell-1}\frac{\|\boldsymbol{z}^0 - \boldsymbol{z}^*\|}{\|\boldsymbol{z}^0 - \boldsymbol{z}^*\| - \|\boldsymbol{z}^\ell - \boldsymbol{z}^*\|} \leqslant \frac{4c_1(\ell+1)}{(\ell-1)(\ell+1-c_1)}. \tag{D.4}$$

Then the termination criterion (4.2) is satisfied whenever the right-hand side of (D.4) is upper bounded by $\frac{\epsilon_k}{2L_\phi}$, which by calculus further gives (4.6).

Now we consider (V-FISTA). Similarly, the strong convexity of $G_k(\cdot)$ implies that

$$\begin{aligned}&\lambda_{\min}(\boldsymbol{B}_k)\left\|\boldsymbol{z}^\ell-\boldsymbol{z}^*\right\|^2/2 \leqslant G_k(\boldsymbol{z}^\ell)-G_k(\boldsymbol{z}^*)\\&\leqslant\left(1-\frac{1}{\sqrt{\kappa}}\right)^\ell\left(G_k(\boldsymbol{z}^0)-G_k(\boldsymbol{z}^*)+\frac{\lambda_{\min}(\boldsymbol{B}_k)}{2}\left\|\boldsymbol{z}^0-\boldsymbol{z}^*\right\|^2\right)=c_2^2\Big(\frac{1}{\tau}\Big)^{2\ell}\lambda_{\min}(\boldsymbol{B}_k)/2,\end{aligned}\tag{D.5}$$

where the second inequality follows from [2, Theorem 10.42] and the last equality follows from the definition of $\tau$ and $c_2$ in (4.5). We then see from (D.5) that

$$\left\|\boldsymbol{z}^\ell-\boldsymbol{z}^*\right\|\leqslant\frac{c_2}{\tau^\ell}.\tag{D.6}$$

This together with the $\boldsymbol{y}$-update in (V-FISTA) that for $\ell\geqslant 2$,

$$\begin{aligned}\left\|\boldsymbol{z}^\ell-\boldsymbol{y}^\ell\right\|&=\left\|\boldsymbol{z}^\ell-\boldsymbol{z}^{\ell-1}-\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}(\boldsymbol{z}^{\ell-1}-\boldsymbol{z}^{\ell-2})\right\|\leqslant\left\|\boldsymbol{z}^\ell-\boldsymbol{z}^{\ell-1}\right\|+\left\|\boldsymbol{z}^{\ell-1}-\boldsymbol{z}^{\ell-2}\right\|\\&\leqslant\left\|\boldsymbol{z}^\ell-\boldsymbol{z}^*\right\|+2\left\|\boldsymbol{z}^{\ell-1}-\boldsymbol{z}^*\right\|+\left\|\boldsymbol{z}^{\ell-2}-\boldsymbol{z}^*\right\|\leqslant c_2\left(\frac{1}{\tau^\ell}+\frac{2}{\tau^{\ell-1}}+\frac{1}{\tau^{\ell-2}}\right)\leqslant\frac{4c_2}{\tau^{\ell-2}}.\end{aligned}\tag{D.7}$$

Since $\boldsymbol{z}^0\neq\boldsymbol{z}^*$, we use (D.6) and have that for $\ell\geqslant 1+\log_\tau\frac{c_2}{\|\boldsymbol{z}^0-\boldsymbol{z}^*\|}$,

$$\left\|\boldsymbol{z}^\ell-\boldsymbol{z}^0\right\|\geqslant\left\|\boldsymbol{z}^0-\boldsymbol{z}^*\right\|-\left\|\boldsymbol{z}^\ell-\boldsymbol{z}^*\right\|\geqslant\left\|\boldsymbol{z}^0-\boldsymbol{z}^*\right\|-\frac{c_2}{\tau^\ell}>0.$$

Using this and (D.7), we have for any $\ell\geqslant\max\{2,\,1+\log_\tau\frac{c_2}{\|\boldsymbol{z}^0-\boldsymbol{z}^*\|}\}$ that

$$\frac{\left\|\boldsymbol{z}^\ell-\boldsymbol{y}^\ell\right\|}{\left\|\boldsymbol{z}^\ell-\boldsymbol{z}^0\right\|}\leqslant\frac{4c_2/\tau^{\ell-2}}{\|\boldsymbol{z}^0-\boldsymbol{z}^*\|-c_2/\tau^\ell}=\frac{4c_2\tau^2}{\tau^\ell\|\boldsymbol{z}^0-\boldsymbol{z}^*\|-c_2}.\tag{D.8}$$

Then the termination criterion (4.2) is satisfied whenever the right-hand side of (D.8) is upper bounded by $\frac{\epsilon_k}{2L_\phi}$, which by calculus further gives (4.7). This completes the proof. □

# References

[1] M. Ahn, J. S. Pang and J. Xin. Difference-of-convex learning: directional stationarity, optimality, and sparsity. *SIAM Journal on Optimization*, 27, 1637–1665, 2017.

[2] A. Beck. *First-Order Methods in Optimization*, SIAM 2017.

[3] S. Becker, J. Fadili and P. Ochs. On quasi-Newton forward–backward splitting: proximal calculus and convergence. *SIAM Journal on Optimization*, 29, 2445–2482, 2019.

[4] S. Becker, E. J. Candès and M. C. Grant. Templates for convex cone problems with applications to sparse signal recovery. *Mathematical Programming Computation*, 3, 165–218, 2011.

[5] S. Bonettini, I. Loris, F. Porta and M. Prato. Variable metric inexact line-search-based methods for nonsmooth optimization. *SIAM Journal on Optimization*, 26, 891–921, 2016.

[6] R. H. Byrd, J. Nocedal and F. Oztoprak. An inexact successive quadratic approximation method for $L$-1 regularized optimization. *Mathematical Programming*, 157, 375–396, 2016.

[7] S. Bonettini, F. Porta and V. Ruggiero. A variable metric forward-backward method with extrapolation. *SIAM Journal on Scientific Computing*, 38, A2558–A2584, 2016.

[8] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2, 183–202, 2009.

[9] E. Chouzenoux, J. C. Pesquet, and A. Repetti. Variable metric forward–backward algorithm for minimizing the sum of a differentiable function and a convex function. *Journal of Optimization Theory and Applications*, 162, 107–132, 2014.

[10] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 456, 1348–1360, 2001.

[11] H. Ghanbari and K. Scheinberg. Proximal quasi-Newton methods for regularized convex optimization with linear and accelerated sublinear convergence rates. *Computational Optimization and Applications*, 69, 597–627, 2018.

[12] J. Y. Gotoh, A. Takeda and K. Tono. DC formulations and algorithms for sparse optimization problems. *Mathematical Programming*, 169, 141–176, 2018.

[13] P. Gong, C. Zhang, Z. Lu, J. Huang and J. Ye. A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems. *International Conference on Machine Learning*, 37–45, 2013.

[14] C. Kanzow and T. Lechner. Globalized inexact proximal Newton-type methods for nonconvex composite functions. *Available in https://www.mathematik.uni-wuerzburg.de/fileadmin/10040700/paper/ProxNewton.pdf*, 2020.

[15] S. Karimi and S. Vavasis. IMRO: A proximal quasi-Newton method for solving $\ell_1$-regularized least squares problems. *SIAM Journal on Optimization*, 27, 583–615, 2017.

[16] C. P. Lee and S. J. Wright. Inexact successive quadratic approximation for regularized optimization. *Computational Optimization and Applications*, 72, 641–674, 2019.

[17] G. Li, T. Liu and T. P. Pong. Peaceman–Rachford splitting for a class of nonconvex optimization problems. *Computational Optimization and Applications*, 68, 407–436, 2017.

[18] J. Li, M. S. Andersen and L. Vandenberghe. Inexact proximal Newton methods for self-concordant functions. *Mathematical Methods of Operations Research*, 85, 19–41, 2017.

[19] H. Li and Z. Lin. Accelerated proximal gradient methods for nonconvex programming. *In Advances in Neural Information Processing Systems*, 379–387, 2015.

[20] H. Lin, J. Mairal and Z. Harchaoui. An inexact variable metric proximal point algorithm for generic quasi-Newton acceleration. *SIAM Journal on Optimization*, 29, 1408–1443, 2019.

[21] J. D. Lee, Y. Sun and M. A. Saunders. Proximal Newton-type methods for minimizing composite functions. *SIAM Journal on Optimization*, 24, 1420–1443, 2014.

[22] X. Li, D. Sun and K. C. Toh. A highly efficient semismooth Newton augmented Lagrangian method for solving Lasso problems. *SIAM Journal on Optimization*, 28, 433–458, 2018.

[23] T. Liu and T. K. Pong. Further properties of the forward–backward envelope with applications to difference-of-convex programming. *Computational Optimization and Applications*, 67, 489–520, 2017.

[24] T. Liu, T. K. Pong and A. Takeda. A refined convergence analysis of pDCA$_e$ with applications to simultaneous sparse recovery and outlier detection. *Computational Optimization and Applications*, 73, 69–100, 2019.

[25] T. Liu, T. K. Pong and A. Takeda. A successive difference-of-convex approximation method for a class of nonconvex nonsmooth optimization problems. *Mathematical Programming*, 176, 339–367, 2019.

[26] Z. Q. Luo and P. Tseng. Error bound and convergence analysis of matrix splitting algorithms for the affine variational inequality problem. *SIAM Journal on Optimization*, 2, 43–54, 1992.

[27] Y. Lou and M. Yan. Fast L$_1$–L$_2$ minimization via a proximal operator. *Journal of Scientific Computing*, 74, 767–785, 2018.

[28] T. H. Ma, Y. Lou and T. Z. Huang. Truncated $\ell_{1-2}$ models for sparse recovery and rank minimization. *SIAM Journal on Imaging Sciences*, 10, 1346–1380, 2017.

[29] S. Nakayama, Y. Narushima and H. Yabe. Inexact proximal memoryless quasi-Newton methods based on the Broyden family for minimizing composite functions. *Computational Optimization and Applications*, 1–28, 2021.

[30] B. O'donoghue and E. J. Candès. Adaptive restart for accelerated gradient schemes. *Foundations of Computational Mathematics*, 15, 715–732, 2015.

[31] W. Peng, H. Zhang, X. Zhang and L. Cheng. Global complexity analysis of inexact successive quadratic approximation methods for regularized optimization under mild assumptions. *Journal of Global Optimization*, 1–21, 2020.

[32] R. T. Rockafellar and R. J-B. Wets. *Variational Analysis*. Springer 1998.

[33] S. Salzo. The variable metric forward-backward splitting algorithm under mild differentiability assumptions. *SIAM Journal on Optimization*, 27, 2153–2181, 2017.

[34] M, Schmidt, N. L. Roux and F. Bach. Convergence rates of inexact proximal-gradient methods for convex optimization. *In Advances in Neural Information Processing Systems*, 1458–1466, 2011.

[35] K. Scheinberg and X. Tang. Practical inexact proximal quasi-Newton method with global complexity analysis. *Mathematical Programming*, 160, 495–529, 2016.

[36] L. Stella, A. Themelis and P. Patrinos. Forward–backward quasi-Newton methods for nonsmooth optimization problems. *Computational Optimization and Applications*, 67, 443–487. 2017.

[37] P. D. Tao and L. T. H. An. Convex analysis approach to DC programming: theory, algorithms and applications. *Acta Mathematica Vietnamica*, 22, 289–355, 1997.

[38] P. Tseng and S. Yun. A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming*, 117, 387–423, 2009.

[39] B. Wen, X. Chen and T. K. Pong. A proximal difference-of-convex algorithm with extrapolation. *Computational Optimization and Applications*, 69, 297–324, 2018.

[40] Y. Wang, Z. Luo and X. Zhang. New improved penalty methods for sparse reconstruction based on difference of two norms. Available at researchgate. DOI: 10.13140/RG.2.1.3256.3369.

[41] S. J. Wright, R. D. Nowak and M. A. T. Figueiredo. Sparse reconstruction by separable approximation. *IEEE Transactions on Signal Processing*, 57, 2479–2493, 2009.

[42] L. Yang. Proximal gradient method with extrapolation and line search for a class of nonconvex and nonsmooth problems. Available online from https://arxiv.org/abs/1711.06831.

[43] P. Yin, Y. Lou, Q. He and J. Xin. Minimization of $\ell_{1-2}$ for compressed sensing. *SIAM Journal on Scientific Computing*, 37, A536–A563, 2015.

[44] M. C. Yue, Z. Zhou and A. M. C. So. A family of inexact SQA methods for non-smooth convex minimization with provable convergence guarantees based on the Luo–Tseng error bound property. *Mathematical Programming*, 174, 327–358, 2019.

[45] C. H. Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38, 894–942, 2010.