Chin. Ann. Math. Ser. B
43(6), 2022, 1049–1070
DOI: 10.1007/s11401-022-0377-7

Chinese Annals of Mathematics, Series B

Extrapolated Smoothing Descent Algorithm for Constrained Nonconvex and Nonsmooth Composite Problems*

Yunmei CHEN[1] Hongcheng LIU[2] Weina WANG[3]

Abstract In this paper, the authors propose a novel smoothing descent type algorithm with extrapolation for solving a class of constrained nonsmooth and nonconvex problems, where the nonconvex term is possibly nonsmooth. Their algorithm adopts the proximal gradient algorithm with extrapolation and a safe-guarding policy to minimize the smoothed objective function for better practical and theoretical performance. Moreover, the algorithm uses a easily checking rule to update the smoothing parameter to ensure that any accumulation point of the generated sequence is an (affine-scaled) Clarke stationary point of the original nonsmooth and nonconvex problem. Their experimental results indicate the effectiveness of the proposed algorithm.

Keywords Constrained nonconvex and nonsmooth optimization, Smooth approximation, Proximal gradient algorithm with extrapolation, Gradient descent algorithm, Image reconstruction
2000 MR Subject Classification 65F22, 65K05, 94A08, 90C26

1 Introduction

Nonconvex and nonsmooth composite problems have been receiving much attention in modern science and technology, such as signal processing (see [9, 17, 51]), image restoration (see [19, 32, 34, 49]) and image reconstruction (see [2, 15, 33, 39–40]). This is mainly because of their superior ability to produce sparser solutions and recover images with neater edges (see [11, 22, 32–34, 49]). In particular, compared to unconstrained nonconvex models, the corresponding constrained models can achieve reasonable improvements when most pixel intensities of an image are around the boundary of a closed convex set (see [3–6, 10, 12, 24, 41, 51]). In this paper, we focus on the following constrained nonconvex nonsmooth composite minimization

$$\min_{\mathbf{u}\in\Omega} \; F(\mathbf{u}) = r(\mathbf{u}) + h(\mathbf{u}), \tag{1.1}$$

where Ω is a closed convex set, $r(\mathbf{u})$ is a nonconvex possibly non-Lipschitz function and $h(\mathbf{u})$ is a smooth function and possibly nonconvex. In image processing problems, $r(\mathbf{u})$ in (1.1) can be considered as the regularization term dependent on the prior knowledge of images, such as

Manuscript received April 3, 2022.
[1]Department of Mathematics, University of Florida, Gainesville 118105, USA. E-mail: yun@ufl.edu
[2]Industrial and Systems Engineering, University of Florida, Gainesville 118105, USA. E-mail: hliu@ise.ufl.edu
[3]Corresponding author. Department of Mathematics, Hangzhou Dianzi University, Hangzhou 310018, China. E-mail: wnwang@hdu.edu.cn
*This work was supported by the National Natural Science Foundation of China (No. 12001144), Zhejiang Provincial Natural Science Foundation of China (No. LQ20A010007) and NSF/DMS-2152961.

ℓ_p regularization ($0 \leq p < 1$) (see [2, 12, 19, 22, 45, 48–49, 51]), $h(\mathbf{u})$ can be considered as the fidelity term for measuring the deviation of a solution from the observation, such as the least squares data fitting term (see [2, 37, 43]) and Ω is usually a box constrained set.

Algorithms for solving the nonsmooth and nonconvex problems of form (1.1) have been studied extensively, due to their wide range of applications. If h is smooth (possibly nonconvex) and r is simple (i.e., its proximal operator has a closed form solution or the proximal point is easy to compute), the proximal gradient method (also known as the forward-backward splitting) is very effective (see [1, 7, 18, 29]). An abstract convergence result for nonconvex descent methods including proximal gradient and gradient descent algorithms under a sufficient decrease condition and a relative inexact optimality condition has been presented in [1]. For non-simple r, several inexact proximal gradient and gradient descent algorithms have developed to reduce computational cost while still ensuring convergence under certain conditions (see [1, 21, 24–25, 28, 30, 38, 47]). Moreover, a number of works were proposed to integrate the Nesterov's accelerated gradient descent algorithm into the proximal gradient algorithm for improved iteration efficiency while maintaining convergence guarantee for nonconvex programming (see [20, 27–28, 39, 42]). The iPiano algorithm combined proximal gradient method with an inertial force has better performance and nice convergence properties (see [35, 44]). However, most of standard or accelerated and/or inexact proximal gradient algorithms for nonconvex programming require r to be smooth or satisfy the Kurdyka-Łojasiewicz (KL for short) inequality for global convergence (see [1, 27, 44, 46]). In this work we consider more general nonconvex nonsmooth problem composed of gradient operators, which may not satisfy these conditions.

For more general nonconvex and nonsmooth optimization problems, especially for the nonconvex component being also nonsmooth, a natural choice is to use the smoothing strategy (see [4–6, 11–14, 22, 26, 33–34, 36]). Smoothing methods construct a sequence of smooth nonconvex problems to approximate the original nonsmooth problem, and each smooth problem with the fixed smoothing parameter can be solved by efficient algorithms such as the gradient descent method combined with line search (see [11]), the nonlinear conjugate gradient method (see [14]) and the trust region Newton method (see [13]). By updating the smoothing parameter, smoothing algorithms are able to solve the original nonsmooth nonconvex optimizations and any accumulation point of the generated sequence is a Clarke stationary point when the gradient consistency of subdifferential associated with a smoothing function is proved (see [4–6, 11–14]). For instance, [4] discussed the first order necessary optimality condition for local minimizers and defined the generalized stationary point for a class of constrained nonsmooth nonconvex problems where the feasible set is a closed convex set. Recently, to accelerate the smoothing method for nonconvex problems, [39] introduced a convergent smoothing gradient descent type algorithm with extrapolation technique. It can not only guarantee that any accumulation point is an (affine-scaled) Clarke stationary point, but also obtain better experimental results compared to the standard smoothing gradient descent method. Instead of directly converting the nonsmooth function into parameterized smooth function, iterative support shrinking with proximal linearization algorithms (see [19, 30, 40–41, 48]) obtained a nonconvex smooth objective function by putting nondifferentiable points of the nonsmooth function into constraints. These methods were easy to produce piecewise constant regions and thus were not suitable for recovering smooth parts of images.

In this paper, we propose an accelerated smoothing descent algorithm for solving a general class of constrained nonsmooth nonconvex optimization problems, where the nonconvex term is a potential function composed with the L_2 norm of the gradient of the unknown function. Our algorithm adopts the proximal gradient algorithm with extrapolation and a safe-guarding policy to minimize the smoothed objective function to guarantee a better practical and theoretical

performance. The smoothing method is inspired by [11], which is equivalent to Nesterov's smoothing technique for non-smooth optimization (see [31]). Moreover, the algorithm uses a rule that is easy to implement to adoptively reduce the smoothing parameter. We can prove that any accumulation point of the generated sequence is an (affine-scaled) Clarke stationary point of the nonsmooth nonconvex problem (1.1). The main contributions are summarized as follows:

• We propose an extrapolated smoothing descent algorithm for constrained nonconvex nonsmooth minimization problems, where the nonconvex part is also nonsmooth and may not be simple or satisfy KL property. Our algorithm adopts the proximal gradient algorithm with extrapolation and a safe-guarding policy to minimize the smoothed objective function. The algorithm can also adaptively reduce the smoothing level to approach a stationary point of the original problem.

• We prove that the sequence generated by the proposed algorithm has at least an accumulation point, and any accumulation point of the sequence is an (affine-scaled) Clarke stationary point of the nonconvex and nonsmooth problem. Moreover, the total number of iterations required to terminate our algorithm with a given tolerance is also studied.

• We conduct a series of numerical experiments with comparisons to several existing descent type of algorithms with or without box constraints and with or without extrapolation for sparse-view CT reconstruction. The experimental results demonstrate the effectiveness of the proposed algorithm.

The paper is organized as follows. In Section 2, we identify a class of constrained nonconvex and nonsmooth optimization problems, and present an extrapolated smoothing descent algorithm (ESDA for short) for solving the problem. In the meantime the smoothing method and properties of the smoothed objective function are studied. In Section 3, we provide convergence and iteration complexity analyses of the proposed algorithm. Experimental results are given in Section 4. At last, conclusions are summarized in Section 5.

2 The Problem and the Algorithm

2.1 Preliminaries

In this paper, we use $\mathbb{R}^n$ to denote the n-dimensional Euclidean space with inner product $\langle\cdot,\cdot\rangle$ and Euclidean norm $\|\cdot\|_p$ $(p>0)$. For $p=2$, we simply denote it by $\|\cdot\|$. Denote by Ω a compact convex subset of $\mathbb{R}^n$. $\Pi_\Omega(\mathbf{u})$ represents the projection of a vector $\mathbf{u}\in R^n$ to Ω defined by $\Pi_\Omega(\mathbf{u}) = \arg\min_{\mathbf{v}\in\Omega}\|\mathbf{v}-\mathbf{u}\|$. For a real-valued matrix A, $\|A\|_2$ denotes its spectral norm that is the largest singular value of A. For a vectored 2-dimensional image $\mathbf{u}\in\mathbb{R}^n$, $d_i\mathbf{u} = (d_i^x\mathbf{u}; d_i^y\mathbf{u}) \in \mathbb{R}^2$ represents $d\mathbf{u}$ at pixel i, and $d_i = (d_i^x, d_i^y) \in \mathbb{R}^{2\times n}$ is the discrete gradient operator at pixel i. In our notation $\mathbb{R}_+ = [0,\infty)$ and $\mathbb{N}_+$ is the set of non-negative integers.

Using the definition of Clarke generalized directional derivative (see [8, 16]) we give the following definitions.

Definition 2.1 *Assume that $g:\mathbb{R}^n\to(-\infty,+\infty]$ is a locally Lipschitz continuous function. The Clarke subdifferential of g at $\mathbf{x}\in\mathbb{R}^n$ is defined as*

$$\partial^\circ g(\mathbf{x}) = \Big\{\mathbf{w}\in\mathbb{R}^n : \langle\mathbf{w},\mathbf{v}\rangle \le \limsup_{\substack{\mathbf{z}\to\mathbf{x}\\ t\downarrow 0}} \frac{g(\mathbf{z}+t\mathbf{v})-g(\mathbf{z})}{t},\ \forall\mathbf{v}\in\mathbb{R}^n\Big\}.$$

Definition 2.2 (Clarke stationary point) *For a locally Lipschitz function $g : \mathbb{R}^n \to (-\infty, +\infty]$, a point $\mathbf{x}^* \in \Omega$, where Ω is a compact subset of $\mathbb{R}^n$, is said to be a Clarke stationary point, if there exists a $\mathbf{d} \in \partial^\circ g(\mathbf{x})$, such that $\langle \mathbf{d}, \mathbf{x} - \mathbf{x}^* \rangle \geq 0$ for all $\mathbf{x} \in \Omega$.*

Definition 2.3 (see [16]) *A function $g : \mathbb{R}^n \to \mathbb{R}$ is said to be regular at $\mathbf{x}$ provided the following hold*:

(i) *For all $\mathbf{v}$, the usual one-sided directional derivative $g'(\mathbf{x}; \mathbf{v}) = \lim\limits_{t \downarrow 0} \frac{g(\mathbf{x}+t\mathbf{v}) - g(\mathbf{x})}{t}$ exists.*

(ii) *For all $\mathbf{v}$, $g'(\mathbf{x}; \mathbf{v}) = \limsup\limits_{\substack{\mathbf{y} \to \mathbf{x} \\ t \downarrow 0}} \frac{g(\mathbf{y}+t\mathbf{v}) - g(\mathbf{y})}{t}$.*

Remark 2.1 (see [16]) Suppose that $g_i : \mathbb{R}^n \to \mathbb{R}$, $i = 1, 2, \cdots, m$, are Lipschitz continuous and regular near $\mathbf{x}$. Then, their sum $g = \sum\limits_{i=1}^{m} g_i$ is also Lipschitz continuous near $\mathbf{x}$ and

$$\mathring{\partial} g(\mathbf{x}) = \sum_{i=1}^{m} \mathring{\partial} g_i(\mathbf{x}).$$

2.2 The problem and basic assumptions

We consider the following type of regularized inverse problem

$$(\mathcal{P}) \qquad \min_{\mathbf{u} \in \Omega} \; F(\mathbf{u}) = r(\mathbf{u}) + h(\mathbf{u}) := \sum_{i=1}^{n} \varphi(\|d_i \mathbf{u}\|) + \frac{\alpha}{2} \|H\mathbf{u} - \mathbf{f}\|^2, \tag{2.1}$$

where $\|d_i \mathbf{u}\| = \sqrt{(d_i^x \mathbf{u})^2 + (d_i^y \mathbf{u})^2}$; $\varphi : [0, +\infty) \to [0, +\infty)$ is a potential function; $\alpha > 0$ is a model parameter to balance the data fitting term and the regularization term.

Assumption 2.1 We assume that

(a) $\varphi(t)$ is C^2 on $(0, +\infty)$, and $\varphi(0) = 0$. Specifically, for any fixed η, φ and φ' are Lipschitz continuous on $\left[\frac{\eta}{2}, \infty\right)$ with constants $L_{0,\varphi} \eta^{-b_{0,\varphi}}$ and $L_{1,\varphi} \eta^{-b_{1,\varphi}}$, respectively, where $L_{0,\varphi}$, $L_{1,\varphi}$, $b_{0,\varphi}$, $b_{1,\varphi} \geq 0$ and independent of η.

(b) $\varphi'(t)|_{(0,+\infty)} \geq 0$ and $\lim\limits_{t \to 0^+} \varphi'(t) = \varphi'(0^+) > 0$.

(c) $\varphi''(t)$ is increasing on $(0, +\infty)$ with $\varphi''(t)|_{(0,+\infty)} \leq 0$.

Many widely used regularizations in image deblurring and reconstruction problems meet these assumptions. In Table 1, we list some nonconvex nonsmooth potential functions (see [11, 14, 32–33]).

Remark 2.2 (1) If $\varphi'(0^+)$ is finite, $\varphi(\|d_i \mathbf{u}\|)$ is Lipschitz at $\|d_i \mathbf{u}\| = 0$. For example, $\varphi_1(t)$ and $\varphi_2(t)$ in Table 1.

(2) If $\varphi'(0^+) = +\infty$, $\varphi(\|d_i \mathbf{u}\|)$ is non-Lipschitz at $\|d_i \mathbf{u}\| = 0$. For example, $\varphi_3(t)$ and $\varphi_4(t)$ in Table 1.

(3) Assumption 2.1(a) and (b) show that 0 is the strict minimizer of $\varphi(t)$.

(4) Assumption 2.1(c) implies that $\varphi(t)$ is a concave function.

2.3 The algorithm

Before we present the proposed algorithm, we first define a smooth approximation problem for the nonsmooth and nonconvex problem (2.1).

The function $\varphi(\|d_i \mathbf{u}\|)$ is nonsmooth and possibly also non-Lipschitz at $\|d_i \mathbf{u}\| = 0$ when $\varphi'(0^+) = +\infty$. Inspired by the approximation technique for $|t|$ in [11], we approximate $\varphi(\|d_i \mathbf{u}\|)$

Table 1 Nonconvex nonsmooth potential functions with a parameter $\beta > 0$.

$\varphi(t)$	$\varphi'(t)\|_{(0,+\infty)}$	$\varphi''(t)\|_{(0,+\infty)}$
$\varphi_1(t) = \frac{\beta t}{1+\beta t}$	$\frac{\beta}{(1+\beta t)^2}$	$\frac{-2\beta^2}{(1+\beta t)^3}$
$\varphi_2(t) = \ln(1+\beta t)$	$\frac{\beta}{(1+\beta t)}$	$\frac{-\beta^2}{(1+\beta t)^2}$
$\varphi_3(t) = t^p, 0 < p < 1$	pt^{p-1}	$p(p-1)t^{p-2}$
$\varphi_4(t) = \ln(1+\beta t^p), 0 < p < 1$	$\frac{\beta p t^{p-1}}{1+\beta t^p}$	$\frac{-\beta p t^{p-2}(\beta t^p+1-p)}{(1+\beta t^p)^2}$

by

$$\varphi_\eta(\|d_i\mathbf{u}\|) = \begin{cases} \varphi(\|d_i\mathbf{u}\|), & \text{if } \|d_i\mathbf{u}\| > \eta, \\ \varphi\Big(\dfrac{\|d_i\mathbf{u}\|^2}{2\eta} + \dfrac{\eta}{2}\Big), & \text{if } \|d_i\mathbf{u}\| \le \eta, \end{cases} \tag{2.2}$$

where $\eta > 0$ is a smoothing parameter. For this smoothed $\varphi_\eta(\cdot)$, we have the following proposition.

Proposition 2.1 *If $\varphi_\eta(\|d_i\mathbf{u}\|)$ is a smoothing approximation function of $\varphi(\|d_i\mathbf{u}\|)$ in (2.2), the following statements hold*:

(i) *For any fixed $\eta > 0$, we have $\varphi(\|d_i\mathbf{u}\|) \le \varphi_\eta(\|d_i\mathbf{u}\|) \le \varphi(\|d_i\mathbf{u}\|) + \varphi(\frac{\eta}{2})$.*

(ii) $\lim\limits_{\eta\to 0} \varphi_\eta(\|d_i\mathbf{u}\|) = \varphi(\|d_i\mathbf{u}\|)$.

(iii) *For any fixed $\eta \in (0, 1]$, $\nabla\big(\sum\limits_{i=1}^{n} \varphi_\eta(\|d_i\mathbf{u}\|)\big)$ is L_η-Lipschitz continuous, where*

$$L_\eta = C_\varphi \eta^{-b_\varphi} \tag{2.3}$$

for some constants $C_\varphi > 0$ and $b_\varphi > 0$ that depend on φ, but independent of η.

Proof From the definition of $\varphi_\eta(\|d_i\mathbf{u}\|)$ in (2.2), it is easy to get Parts (i)–(ii). To show Part (iii), we let

$$\varkappa_i(\mathbf{u}) := \begin{cases} \|d_i\mathbf{u}\|, & \text{if } \|d_i\mathbf{u}\| > \eta; \\ \dfrac{\|d_i\mathbf{u}\|^2}{2\eta} + \dfrac{\eta}{2}, & \text{if } \|d_i\mathbf{u}\| \le \eta. \end{cases}$$

Apparently, $\varkappa_i(\mathbf{u}) \ge \frac{\eta}{2}$. In the next, we first to show that $\varkappa_i(\mathbf{u})$ admits Lipschitz continuous gradient with constant

$$L_{\varkappa,i} := \eta^{-1}\|d_i\|_2^2, \tag{2.4}$$

where $\|d_i\|_2$ is the spectral norm of the matrix $d_i \in \mathbb{R}^{2\times n}$.

Notice that $\varkappa_i(\mathbf{u})$ can be rewritten as

$$\varkappa_i(\mathbf{u}) = \frac{\eta}{2} + \arg\max_{\mathbf{v}\in V} \Big\{\langle d_i\mathbf{u}, \mathbf{v}\rangle - \frac{\eta}{2}\|\mathbf{v}\|^2\Big\}, \tag{2.5}$$

where $V = \{\mathbf{v} \in \mathbb{R}^2 \mid \|\mathbf{v}\| \le 1\}$.

For any $\mathbf{u}_1, \mathbf{u}_2 \in \Omega$, define $\mathbf{v}_1$ and $\mathbf{v}_2$ as follows,

$$\mathbf{v}_1 = \frac{\eta}{2} + \arg\max_{\mathbf{v}\in V} \Big\{\langle d_i\mathbf{u}_1, \mathbf{v}\rangle - \frac{\eta}{2}\|\mathbf{v}\|^2\Big\}, \tag{2.6}$$

$$\mathbf{v}_2 = \frac{\eta}{2} + \arg\max_{\mathbf{v}\in V} \Big\{\langle d_i\mathbf{u}_2,\, \mathbf{v}\rangle - \frac{\eta}{2}\|\mathbf{v}\|^2\Big\}, \tag{2.7}$$

which are well defined since the maximization problems have unique solutions. Also using the concavity of the set (V) constrained problems above in $\mathbf{v}$, the optimality conditions of $\mathbf{v}_1$ and $\mathbf{v}_2$ lead to

$$\begin{aligned}\langle d_i\mathbf{u}_1 - \eta\mathbf{v}_1,\, \mathbf{v}_2 - \mathbf{v}_1\rangle \le 0,\\ \langle d_i\mathbf{u}_2 - \eta\mathbf{v}_2,\, \mathbf{v}_1 - \mathbf{v}_2\rangle \le 0.\end{aligned}$$

Adding the two inequalities above yields

$$\langle d_i\mathbf{u}_1 - d_i\mathbf{u}_2 - \eta(\mathbf{v}_1 - \mathbf{v}_2),\, \mathbf{v}_2 - \mathbf{v}_1\rangle \le 0,$$

which, together with the Cauchy-Schwarz inequality, implies

$$\|d_i\mathbf{u}_1 - d_i\mathbf{u}_2\| \ge \eta\|\mathbf{v}_1 - \mathbf{v}_2\|. \tag{2.8}$$

From (2.5)–(2.6), it is easy to see that $\nabla\varkappa_i(\mathbf{u}_j) = \nabla\langle\mathbf{u}_j, d_i^T\mathbf{v}_j\rangle = d_i^T\mathbf{v}_j$ for $j = 1, 2$, where $d_i^T \in \mathbb{R}^{n\times 2}$. Therefore,

$$\begin{aligned}\|\nabla\varkappa_i(\mathbf{u}_1) - \nabla\varkappa_i(\mathbf{u}_2)\| &= \|d_i^T(\mathbf{v}_1 - \mathbf{v}_2)\|\\ &\le \|d_i\|_2\|(\mathbf{v}_1 - \mathbf{v}_2)\|\\ &\le \eta^{-1}\|d_i\|_2\|d_i\mathbf{u}_1 - d_i\mathbf{u}_2\|\\ &\le \eta^{-1}\|d_i\|_2^2\|\mathbf{u}_1 - \mathbf{u}_2\|,\end{aligned}$$

in the last inequality we used (2.8). The claim (2.4) is proved.

Observe that $\nabla\varphi(\varkappa_i(\mathbf{u})) = \frac{\mathrm{d}\varphi(x)}{\mathrm{d}x}\Big|_{x=\varkappa_i(\mathbf{u})} \cdot \frac{\mathrm{d}\varkappa_i(\mathbf{u})}{d\mathbf{u}} = \varphi'(\varkappa_i(\mathbf{u}))\nabla\varkappa_i(\mathbf{u})$ and thus,

$$\begin{aligned}&\|\nabla\varphi(\varkappa_i(\mathbf{u}_1)) - \nabla\varphi(\varkappa_i(\mathbf{u}_2))\|\\ &= \|\varphi'(\varkappa_i(\mathbf{u}_1))\nabla\varkappa_i(\mathbf{u}_1) - \varphi'(\varkappa_i(\mathbf{u}_2))\nabla\varkappa_i(\mathbf{u}_2)\|\\ &\le \|\varphi'(\varkappa_i(\mathbf{u}_1))\nabla\varkappa_i(\mathbf{u}_1) - \varphi'(\varkappa_i(\mathbf{u}_1))\nabla\varkappa_i(\mathbf{u}_2)\|\\ &\quad + \|\varphi'(\varkappa_i(\mathbf{u}_1))\nabla\varkappa_i(\mathbf{u}_2) - \varphi'(\varkappa_i(\mathbf{u}_2))\nabla\varkappa_i(\mathbf{u}_2)\|\\ &\le \sup_{\mathbf{u}\in\Omega}\|\varphi'(\varkappa_i(\mathbf{u}))\| \cdot \|\nabla\varkappa_i(\mathbf{u}_1) - \nabla\varkappa_i(\mathbf{u}_2)\|\\ &\quad + \|\varphi'(\varkappa_i(\mathbf{u}_1)) - \varphi'(\varkappa_i(\mathbf{u}_2))\| \cdot \sup_{\mathbf{u}\in\Omega}\|\nabla\varkappa_i(\mathbf{u})\|\\ &\le L_{0,\varphi}\cdot\eta^{-b_{0,\varphi}}\cdot L_{\varkappa,i}\|\mathbf{u}_1 - \mathbf{u}_2\| + L_{1,\varphi}\eta^{-b_{1,\varphi}}\cdot\|\varkappa_i(\mathbf{u}_1) - \varkappa_i(\mathbf{u}_2)\|\cdot\max\{1,\eta\}\\ &\le (L_{0,\varphi}\eta^{-b_{0,\varphi}} + L_{1,\varphi}\eta^{-b_{1,\varphi}})\cdot L_{\varkappa,i}\cdot\|\mathbf{u}_1 - \mathbf{u}_2\|.\end{aligned}$$

Here, the last second inequality is due to (a) the Lipschitz continuity of φ with constant $L_{0,\varphi}$, (b) the Lipschitz continuity of the gradient of φ with constant $L_{1,\varphi}$ from Assumption 2.1, (c) the facts that $\eta \le \eta_0 = 1$ from algorithm ESDA and hence $\|\nabla\varkappa_i(\mathbf{u})\| \le \max\{1,\eta\} = 1$, (d) (2.4). Meanwhile, the last inequality is due to again $\sup_{\mathbf{u}}\|\nabla\varkappa_i(\mathbf{u})\| \le 1$ and $\|\varkappa_i(\mathbf{u}_1) - \varkappa_i(\mathbf{u}_2)\| \le \sup_{\mathbf{u}\in\Omega}\|\nabla\varkappa_i(\mathbf{u})\| \cdot \|\mathbf{u}_1 - \mathbf{u}_2\|$. Combining the above with (2.4), we immediately have the desired result in Theorem 3.2 with $C_\varphi = (L_{0,\varphi} + L_{1,\varphi})\cdot|\Omega|\sum_{i=1}^{n}\|d_i\|_2^2$, where $|\Omega|$ represents the diameter of Ω, and $b_\varphi = 1 + \max\{b_{0,\varphi},\, b_{1,\varphi}\}$.

The function $\frac{\alpha}{2}\|H\mathbf{u}-\mathbf{f}\|^2$ also has Lipschitz continuous gradient, we denote its Lipschitz constant by L_h. Define a smoothing approximation problem of (2.1) as

$$\min_{\mathbf{u}\in\Omega} F_\eta(\mathbf{u}) := r_\eta(\mathbf{u}) + h(\mathbf{u}) = \sum_{i=1}^{n} \varphi_\eta(\|d_i\mathbf{u}\|) + \frac{\alpha}{2}\|H\mathbf{u}-\mathbf{f}\|^2. \tag{2.9}$$

Clearly, $F_\eta(\mathbf{u})$ has Lipschitz continuous gradient with the Lipschitz continuous gradient $L_\eta + L_h$.

Now we propose the following extrapolated smoothing decent algorithm (ESDA for short) for (2.1).

Algorithm 1: Extrapolated Smoothing Decent Algorithm (ESDA for short) for (2.1)

Step 0: Input $(\rho,\delta,\tau_1)\in(0,1)$, $\tau>0$, Maximum number of iterations K or tolerance $\varepsilon_{\text{tol}}>0$; Initialize $\mathbf{u}_{-1}=\mathbf{u}_0\in\Omega$, $\theta_0=1$, and $\eta_{-1}=\eta_0>0$.

Step 1: For $k=0,1,2,\cdots$,

Step 1.1: Set $\theta_{k+1}=\frac{1+\sqrt{1+4\theta_k^2}}{2}$.

Step 1.2: Let $\mathbf{w}_{k+1}=\mathbf{u}_k+\big(\frac{\theta_k-1}{\theta_{k+1}}\big)(\mathbf{u}_k-\mathbf{u}_{k-1})$.

Step 1.3: Define $\mathbf{q}_{k+1}$:

$$\mathbf{q}_{k+1}=\begin{cases}\mathbf{w}_{k+1}, & \text{if } F_{\eta_k}(\mathbf{w}_{k+1})\le F_{\eta_k}(\mathbf{u}_k) \text{ and } \mathbf{w}_{k+1}\in\Omega,\\ \mathbf{u}_k, & \text{otherwise.}\end{cases} \tag{2.10}$$

Step 1.4: Compute $\widehat{\mathbf{u}}_{\mathbf{k+1}}$:

$$\mathbf{z}_{k+1}=\mathbf{q}_{k+1}-s_0\nabla h(\mathbf{q}_{k+1}), \tag{2.11}$$

$$\widehat{\mathbf{u}}_{\mathbf{k+1}}=\Pi_\Omega(\mathbf{z}_{k+1}-s_{k+1}\nabla r_{\eta_k}(\mathbf{z}_{k+1})), \tag{2.12}$$

Step 1.5: Compute $\overline{\mathbf{u}}_{k+1}$:

$$\overline{\mathbf{u}}_{k+1}=\Pi_\Omega(\mathbf{u}_k-\alpha_{k+1}\nabla F_{\eta_k}(\mathbf{u}_k)),\ \text{if} \tag{2.13}$$

$$F_{\eta_k}(\overline{\mathbf{u}}_{k+1})-F_{\eta_k}(\mathbf{u}_k)\le-\delta\|\overline{\mathbf{u}}_{k+1}-\mathbf{u}_k\|^2. \tag{2.14}$$

$$\text{Otherwise, } \alpha_{k+1}\leftarrow\rho\alpha_{k+1} \text{ and go back to (2.13).} \tag{2.15}$$

Step 1.6: Choose $\mathbf{u}_{k+1}$:

$$\mathbf{u}_{k+1}=\begin{cases}\widehat{\mathbf{u}}_{k+1}, & \text{if } F_{\eta_k}(\widehat{\mathbf{u}}_{k+1})\le F_{\eta_k}(\overline{\mathbf{u}}_{k+1}),\\ \overline{\mathbf{u}}_{k+1}, & \text{otherwise.}\end{cases} \tag{2.16}$$

Step 2: Update η_{k+1}

$$\eta_{k+1}=\begin{cases}\tau_1\eta_k, & \text{if } \|\overline{\mathbf{u}}_{k+1}-\mathbf{u}_k\|<\tau\eta_k\alpha_{k+1},\\ \eta_k, & \text{otherwise.}\end{cases} \tag{2.17}$$

Step 3: If $\tau\eta_k\alpha_{k+1}<\varepsilon_{\text{tol}}$, terminate and output u_{k+1}.

Note that in ESDA the generation of $\widehat{\mathbf{u}}_{k+1}$ in (2.11) can be viewed as using the proximal gradient algorithm with extrapolation, where s_0 and s_{k+1} are stepsizes determined by user. The $\widehat{\mathbf{u}}_{k+1}$ plays a role in ESDA to attain better efficiency than the standard gradient descent

method. Our experimental results confirmed this. However, due to the nonconvexity and nonsmoothness of problem (2.1), the sequence $\{\widehat{\mathbf{u}}_{k+1}\}$ may not converge. The $\overline{\mathbf{u}}_{k+1}$ in (2.13) is obtained by the standard gradient descent to safeguard the convergence of the ESDA. The stepsize α_{k+1} is determined by a simple line search strategy in (2.14). We set $\mathbf{u}_{k+1}$ being $\widehat{\mathbf{u}}_{k+1}$ or $\overline{\mathbf{u}}_{k+1}$ whichever has lower value of F_{η_k} to encourage reduction of the objective function.

3 Convergence and Complexity Analysis

In this section, we will discuss the convergence of ESDA and the bound for the number of the iterations required to terminate the algorithm with the prescribed accuracy ε_{tol} for $\|\overline{\mathbf{u}}_{k+1}-\mathbf{u}_k\|$.

First, we give the following lemma that has been proved in [39].

Lemma 3.1 *For $\varphi_{\eta_k}(\cdot)$ defined in* (2.2)*, we have $\varphi_{\eta_{k+1}}(\|d_i\mathbf{u}\|) \le \varphi_{\eta_k}(\|d_i\mathbf{u}\|)$ for any $i \in \{1,2,\cdots,n\}$, if $\eta_{k+1} \le \eta_k$. Consequently, $F_{\eta_{k+1}}(\mathbf{u}) \le F_{\eta_k}(\mathbf{u})$, where $F_\eta(\mathbf{u})$ is defined in* (2.9).

Theorem 3.1 *Let $\{\mathbf{u}_k\}$ be the sequence generated by* ESDA *with any fixed $\eta = \eta_k > 0$ (that is by Step* 1 *of the algorithm). Then for any $\mathbf{u}_0 \in \Omega$ and $\delta > 0$, we have*

1. *the condition* (2.14) *in Step* 1.5 *can be met by finitely many times of line search.*
2. *$\|\overline{\mathbf{u}}_{k+1} - \mathbf{u}_k\| \to 0$ as $k \to \infty$.*
3. *For any $\varepsilon > 0$, let $k_\varepsilon := \min\{k \in \mathbb{N}^+ : \|\overline{\mathbf{u}}_{k+1} - \mathbf{u}_k\| \le \varepsilon\}$. Then*

$$k_\varepsilon \le F_\eta(\mathbf{u}_0)\delta^{-1}\varepsilon^{-2} \le \Big(F(\mathbf{u}_0) + \varphi\Big(\frac{\eta_0}{2}\Big)\Big)\cdot\delta^{-1}\varepsilon^{-2}. \tag{3.1}$$

Proof To prove Part 1. By the optimality conditions for $\overline{\mathbf{u}}_{k+1}$, we have

$$\langle \overline{\mathbf{u}}_{k+1} - \mathbf{u}_k + \alpha_{k+1}\nabla F_{\eta_k}(\mathbf{u}_k),\, \mathbf{u} - \overline{\mathbf{u}}_{k+1}\rangle \ge 0, \quad \forall u \in \Omega. \tag{3.2}$$

And thus

$$\langle \nabla F_{\eta_k}(\mathbf{u}_k),\, \overline{\mathbf{u}}_{k+1} - \mathbf{u}\rangle \le -\frac{1}{\alpha_{k+1}}\|\overline{\mathbf{u}}_{k+1} - \mathbf{u}\|\cdot\|\overline{\mathbf{u}}_{k+1} - \mathbf{u}_k\|, \quad \forall \mathbf{u} \in \Omega. \tag{3.3}$$

If we let $\mathbf{u} = \mathbf{u}_k$, we then obtain

$$\langle \nabla F_\eta(\mathbf{u}_k),\, \overline{\mathbf{u}}_{k+1} - \mathbf{u}_k\rangle \le -\frac{1}{\alpha_{k+1}}\|\overline{\mathbf{u}}_{k+1} - \mathbf{u}_k\|^2. \tag{3.4}$$

By the last statement of Proposition 2.1, ∇F_η is $(L_\eta + L_h)$-Lipschitz, where L_η is given in (2.3) with $\eta = \eta_k$, and L_h is the Lipschitz continuous for $\frac{\alpha}{2}\nabla(\|H\mathbf{u} - \mathbf{f}\|^2) = \frac{\alpha}{2}\sigma(H)$, where $\sigma(H)$ is the largest singular value of H. Hence, we have

$$F_\eta(\overline{\mathbf{u}}_{k+1}) \le F_\eta(\mathbf{u}_k) + \langle \nabla F_\eta(\mathbf{u}_k),\, \overline{\mathbf{u}}_{k+1} - \mathbf{u}_k\rangle + \frac{L_\eta + L_h}{2}\cdot\|\overline{\mathbf{u}}_{k+1} - \mathbf{u}_k\|^2. \tag{3.5}$$

Combining (3.4) and (3.5) we get

$$F_\eta(\overline{\mathbf{u}}_{k+1}) - F_\eta(\mathbf{u}_k) \le \Big(-\frac{1}{\alpha_{k+1}} + \frac{L_\eta + L_h}{2}\Big)\cdot\|\overline{\mathbf{u}}_{k+1} - \mathbf{u}_k\|^2 \le -\delta\|\overline{\mathbf{u}}_{k+1} - \mathbf{u}_k\|^2, \tag{3.6}$$

if $\alpha_{k+1} \le \big(\delta + \frac{L_\eta + L_h}{2}\big)^{-1}$. Hence, the condition (2.14) in Step 1.5 can be met after finitely many line search steps. This proves Part 1. Notice that the smallest α_{k+1} for having (2.14) can be chosen as $\alpha_{k+1} = \big(\delta + \frac{L_\eta + L_h}{2}\big)^{-1}$. The purpose of the line search is to search a better stepsize α_{k+1}, which makes the condition (2.14) met and

$$\alpha_{k+1} \ge \Big(\delta + \frac{L_\eta + L_h}{2}\Big)^{-1}. \tag{3.7}$$

Moreover, even the α_{k+1} satisfying (3.7), it is able to find a $s \in \mathbb{N}_+$, such that

$$\rho^s \alpha_{k+1} \le \Big(\delta + \frac{L_\eta + L_h}{2}\Big)^{-1}. \tag{3.8}$$

From (3.6) and the choice for $\mathbf{u}_{k+1}$ in Step 1.6, we have that for any $\eta = \eta_k$,

$$F_\eta(\mathbf{u}_{k+1}) \le F_\eta(\overline{\mathbf{u}}_{k+1}) \le F_\eta(\mathbf{u}_k) \le \cdots \le F_\eta(\mathbf{u}_0), \quad k = 0, 1, \cdots \tag{3.9}$$

For any positive integer K, summing up over k from $k = 0$ to $k = K$ on the both sides of (3.6). By using (3.9) and the fact that $F_\eta(\mathbf{u}) \ge 0$ due to Assumption 2.1, we obtain

$$\begin{aligned}\sum_{k=0}^{K} \|\overline{\mathbf{u}}_{k+1} - \mathbf{u}_k\|^2 &\le \delta^{-1} \sum_{k=0}^{K} [F_\eta(\mathbf{u}_k) - F_\eta(\overline{\mathbf{u}}_{k+1})] \\ &= \delta^{-1}(F_\eta(\mathbf{u}_0) - F_\eta(\overline{\mathbf{u}}_{K+1})) \le \delta^{-1} F_\eta(\mathbf{u}_0).\end{aligned} \tag{3.10}$$

Because K is arbitrary, we have $\|\overline{\mathbf{u}}_{k+1} - \mathbf{u}_k\| \to 0$ as $k \to \infty$. This then proves Part 2.

As for Part 3, we observe that, for any $k < k_\varepsilon$, $\|\overline{\mathbf{u}}_{k+1} - \mathbf{u}_k\| > \varepsilon$. From (3.10) with $K = k_\varepsilon - 1$, it must hold that

$$k_\varepsilon \cdot \varepsilon^2 \le \delta^{-1} F_\eta(\mathbf{u}_0) \le \delta^{-1}\Big(F(\mathbf{u}_0) + \varphi\Big(\frac{\eta_0}{2}\Big)\Big).$$

Here, the last inequality is due to Part (i) of Proposition 2.1. Then, (3.1) follows immediately and Part 3 is proved.

Now we are ready to discuss the iteration complexity for the ESDA for any $\varepsilon_{\text{tol}} > 0$.

Note that Part 3 of the Theorem 3.1 implies that the reduction criterion in Step 2 of ESDA can be met within finitely many iterations of Step 1 (Steps 1.1–1.6). Let k_l be the counter of iteration when the criterion for reduction of η_k in (2.17) is met for the l-th time (we set $k_0 = -1$), then we can partition the iteration counters $k = 0, 1, 2, \cdots$, into segments accordingly, such that in the l-th segment $k = k_l + 1, \cdots, k_{l+1}$ and $\eta_k = \eta_{k_l} = \eta_0 \tau_1^l$. The following theorem will provide the bound for the length of each segment, from which we can get the total iteration number required to terminate the algorithm with ε_{tol} tolerance.

Theorem 3.2 *Let* $\{\mathbf{u}_k\}$ *be the sequence generated by* ESDA *with any* $\mathbf{u}_0 \in \Omega$ *and* $\delta > 0$. *Then we have*

1. *the number of iterations required for the* l*-th segment*

$$k_{l+1} - k_l \le C_1 \tau_1^{-2l} + C_2 \tau_1^{-2l(1+b_\varphi)}, \tag{3.11}$$

where

$$C_1 = 2\delta^{-1}\Big(F(\mathbf{u}_0) + \varphi\Big(\frac{\eta_0}{2}\Big)\Big) \cdot \tau^{-2} \eta_0^{-2} \Big(\delta + \frac{L_h}{2}\Big)^2$$

and

$$C_2 = 2\delta^{-1}\Big(F(\mathbf{u}_0) + \varphi\Big(\frac{\eta_0}{2}\Big)\Big) \cdot \tau^{-2} \eta_0^{-2(1+b_\varphi)} C_\varphi^2,$$

and the constants C_φ *and* b_φ *are given in* (2.3).

2. *The total number of iterations* L *for* ESDA *to terminate with the tolerance* $\varepsilon_{\text{tol}} > 0$ *is bounded by*

$$\sum_{l=0}^{L-1} (k_{l+1} - k_l) \le C_1 \frac{\tau_1^{-2(L-1)} - \tau_1^2}{1 - \tau_1^2} \; + \; C_2 \frac{\tau_1^{-2(L-1)(1+b_\varphi)} - \tau_1^{2(1+b_\varphi)}}{1 - \tau_1^{2(1+b_\varphi)}} = O(\varepsilon_{\text{tol}}^{-2}). \tag{3.12}$$

Proof Applying Part 3 of Theorem 3.1 to the l-th segment with the initial $u_0 = u_{k_l+1}$ and hyper-parameter $\varepsilon = \tau\eta_{k_l}\alpha_{k+1}$, where α_{k+1} is the stepsize used in the l-th segment to minimize $F_{\eta_{k_l}}(\mathbf{u})$. From (3.1) we obtain

$$k_{l+1} - k_l \le \delta^{-1} \cdot F_{\eta_{k_l}}(\mathbf{u}_{k_l+1}) \cdot (\tau\eta_{k_l}\alpha_{k+1})^{-2}. \tag{3.13}$$

Next we estimate each term in the RHS of (3.13) in terms of the order of τ_1.

By the combination of Lemma 3.1 and (3.9), for all $k \ge 0$, it holds that

$$F_{\eta_{k+1}}(\mathbf{u}_{k+1}) \le F_{\eta_k}(\mathbf{u}_{k+1}) \le F_{\eta_k}(\mathbf{u}_k) \le \cdots \le F_{\eta_0}(\mathbf{u}_0) \le F(\mathbf{u}_0) + \varphi\Big(\frac{\eta_0}{2}\Big). \tag{3.14}$$

From (3.7) with $\eta = \eta_{k_l}$ and (2.3), we have

$$\alpha_{k+1}^{-2} \le \Big[\Big(\delta + \frac{L_{\eta_{k_l}} + L_h}{2}\Big)\Big]^2 = \Big[\Big(\delta + \frac{L_h}{2}\Big) + \frac{1}{2}C_\varphi \eta_{k_l}^{-b_\varphi}\Big]^2. \tag{3.15}$$

Combining (3.13)–(3.15), also noticing that $\eta_{k_l} = \eta_0\tau_1^l$, we obtain

$$\begin{aligned} k_{l+1} - k_l &\le \delta^{-1}\Big(F(\mathbf{u}_0) + \varphi\Big(\frac{\eta_0}{2}\Big)\Big)\tau^{-2}\eta_{k_l}^{-2}\Big[\Big(\delta + \frac{L_h}{2}\Big) + \frac{1}{2}C_\varphi\eta_{k_l}^{-b_\varphi}\Big]^2 \\ &\le C_1\tau_1^{-2l} + C_2\tau_1^{-2l(1+b_\varphi)}, \end{aligned} \tag{3.16}$$

where

$$C_1 = 2\delta^{-1}\Big(F(\mathbf{u}_0) + \varphi\Big(\frac{\eta_0}{2}\Big)\Big) \cdot \tau^{-2}\eta_0^{-2}\Big(\delta + \frac{L_h}{2}\Big)^2$$

and

$$C_2 = 2\delta^{-1}\Big(F(\mathbf{u}_0) + \varphi\Big(\frac{\eta_0}{2}\Big)\Big) \cdot \tau^{-2}\eta_0^{-2(1+b_\varphi)}C_\varphi^2.$$

To show Part 2, let L be the number of times the reduction of η is satisfied before the algorithm is terminated in Step 3 with the tolarence ε_{tol}. Then $\tau\eta_{k_{L-1}}\alpha_{k+1} \ge \varepsilon_{\text{tol}}$, where α_{k+1} is the stepsize used in the $(L-1)$-th segment to minimize $F_{\eta_{k_{L-1}}}(\mathbf{u})$. Hence,

$$\tau\eta_{k_{L-1}}\alpha_{k+1} \ge \varepsilon_{\text{tol}}. \tag{3.17}$$

From (3.8) and (2.3) with $\eta = \eta_{k_{L-1}}$ it holds that

$$\alpha_{k+1} \le \rho^{-s}\Big(\delta + \frac{L_{\eta_{k_{L-1}}} + L_h}{2}\Big)^{-1} \le \rho^{-s}\Big(\frac{L_{\eta_{k_{L-1}}}}{2}\Big)^{-1} \le 2\rho^{-s}C_\varphi^{-1}\eta_{k_{L-1}}^{b_\varphi}. \tag{3.18}$$

Noticing that $\eta_{k_{L-1}} = \eta_0\tau_1^{L-1}$, then the combination of (3.17) and (3.18) yields that

$$2\rho^{-s}\tau(\eta_0\tau_1^{L-1})C_\varphi^{-1}(\eta_0\tau_1^{L-1})^{b_\varphi} \ge \varepsilon_{\text{tol}}.$$

Rearranging the above inequality, we have

$$\tau_1^{(L-1)(1+b_\varphi)} \ge \rho^s\frac{C_\varphi\varepsilon_{\text{tol}}}{2\tau\eta_0^{(1+b_\varphi)}} =: C_3\varepsilon_{\text{tol}}, \tag{3.19}$$

where $C_3 = \rho^s\frac{C_\varphi}{2\tau\eta_0^{(1+b_\varphi)}}$.

To terminate with tolerant ε_{tol} after L times reduction of η from η_0, from (3.16), we then have

$$\sum_{l=0}^{L-1}(k_{l+1} - k_l) \le \sum_{l=0}^{L-1}(C_1\tau_1^{-2l} + C_2\tau_1^{-2l(1+b_\varphi)})$$

$$= C_1 \frac{\tau_1^{-2(L-1)} - \tau_1^2}{1-\tau_1^2} \ + \ C_2 \frac{\tau_1^{-2(L-1)(1+b_\varphi)} - \tau_1^{2(1+b_\varphi)}}{1-\tau_1^{2(1+b_\varphi)}}. \tag{3.20}$$

From this estimate and (3.19), we can see that for fixed $\tau_1 \in (0,1)$,

$$\sum_{l=0}^{L-1}(k_{l+1}-k_l) = O(\varepsilon_{\rm tol}^{-2}).$$

The theorem is proved.

If we set $\varepsilon_{\rm tol}=0$ and $K=\infty$ in the ESDA, the algorithm will never terminate and hence can generate an infinite sequence $\{\mathbf{u}_k\}$. We focus on the subsequence $\{\mathbf{u}_{k_l+1}\}$ as discussed in Theorem 3.2. That is the reduction criterion in Step 2 being satisfied for $k=k_l$ and η_k being reduced. The next theorem below will show that every accumulation point of this subsequence is a Clarke stationary point.

We need the following lemma before to prove the convergence result. This lemma has been proved in [39]. But here we provide more simple proof.

Lemma 3.2 *Let $\varphi_\eta(\|d_i\mathbf{u}\|)$ be a smooth approximation of $\varphi(\|d_i\mathbf{u}\|)$ defined in* (2.2) *for any $i=1,\cdots,n$. Suppose that $\varphi(t)$ is continuously differentiable in $[0,+\infty)$. If $\{\mathbf{u}_j\}\subset\Omega$ is a sequence that converges to a point $\mathbf{u}^*\in\Omega$, then*

$$\lim_{\substack{\mathbf{u}_j\to\mathbf{u}^*\\ \eta_j\downarrow 0}} \nabla\varphi_{\eta_j}(\|d_i\mathbf{u}_j\|) \in \mathring{\partial}\varphi(\|d_i\mathbf{u}^*\|) \tag{3.21}$$

and

$$\lim_{\substack{\mathbf{u}_j\to\mathbf{u}^*\\ \eta_j\downarrow 0}} \nabla F_{\eta_j}(\mathbf{u}_j) \in \mathring{\partial}F(\mathbf{u}^*). \tag{3.22}$$

Proof Firstly, we will show that the Clarke subdifferential $\mathring{\partial}\varphi(\|d_i\mathbf{u}\|)$ for any $\mathbf{u}\in\Omega$ and $i=1,\cdots,n$ is the following:

$$\mathring{\partial}\varphi(\|d_i\mathbf{u}\|) = \begin{cases} \varphi'(\|d_i\mathbf{u}\|)\frac{d_i^T d_i\mathbf{u}}{\|d_i\mathbf{u}\|}, & \text{if } \|d_i\mathbf{u}\|\neq 0,\\ \{\varphi'(0^+)d_i^T\boldsymbol{\xi}:\forall\boldsymbol{\xi}\in\mathbb{R}^2,\ \|\mathrm{P}_i\boldsymbol{\xi}\|\le 1\}, & \text{if } \|d_i\mathbf{u}\|=0 \end{cases} \tag{3.23}$$

can be obtained by using Definition 2.1. Consider following two cases:

Case 1: If $\|d_i\mathbf{u}\|\neq 0$, then there is a small neighborhood of $\mathbf{u}$ such that for any $\mathbf{z}$ in the neighborhood it holds $\|d_i\mathbf{z}\|\neq 0$. Then, for any $\mathbf{v}\in\mathbb{R}^n$, we have

$$\limsup_{\substack{\mathbf{z}\to\mathbf{u}\\ t\downarrow 0}} \frac{\varphi(\|d_i(\mathbf{z}+t\mathbf{v})\|)-\varphi(\|d_i\mathbf{z}\|)}{t} = \lim_{\mathbf{z}\to\mathbf{u}}\varphi'(\|d_i\mathbf{z}\|)\frac{\langle d_i\mathbf{z}, d_i\mathbf{v}\rangle}{\|d_i\mathbf{z}\|} = \varphi'(\|d_i\mathbf{u}\|)\frac{\langle d_i^T d_i\mathbf{u},\mathbf{v}\rangle}{\|d_i\mathbf{u}\|}. \tag{3.24}$$

Case 2: If $\|d_i\mathbf{u}\|=0$, then $d_i\mathbf{u}=0$. Moreover by Assumption 2.1(a), $\varphi(0)=0$. Then, for any $\mathbf{v}\in\mathbb{R}^n$, we have

$$\begin{aligned} &\limsup_{\substack{\mathbf{z}\to\mathbf{u}\\ t\downarrow 0}} \frac{\varphi(\|d_i(\mathbf{z}+t\mathbf{v})\|)-\varphi(\|d_i\mathbf{z}\|)}{t} = \lim_{t\downarrow 0}\frac{\varphi(\|td_i\mathbf{v}\|)-\varphi(0)}{t}\\ &= \varphi'(0^+)\|d_i\mathbf{v}\| \ge \varphi'(0^+)\langle\boldsymbol{\xi}, d_i\mathbf{v}\rangle = \varphi'(0^+)\langle \mathrm{P}_i\boldsymbol{\xi}, d_i\mathbf{v}\rangle \end{aligned} \tag{3.25}$$

for any $\boldsymbol{\xi} \in \mathbb{R}^2$ and $\|\mathrm{P}_i\boldsymbol{\xi}\| \leq 1$, where $\mathrm{P}_i\boldsymbol{\xi}$ is the projection of $\boldsymbol{\xi}$ onto the column space of d, which is perpendicular to the null space of d^T. The combination of (3.24) and (3.25) gives (3.23).

Next we compute $\lim\limits_{\substack{\mathbf{u}_j \to \mathbf{u}^* \\ \eta_j \downarrow 0}} \nabla\varphi_{\eta_j}(\|d_i\mathbf{u}_j\|)$.

It is easy to compute the maximizer $\mathbf{v}^*$ in (2.5) that yields

$$\mathbf{v}^* = \begin{cases} \dfrac{d_i\mathbf{u}}{\|d_i\mathbf{u}\|}, & \text{if } \|d_i\mathbf{u}\| > \eta; \\ \dfrac{d_i\mathbf{u}}{\eta}, & \text{if } \|d_i\mathbf{u}\| \leq \eta \end{cases} \tag{3.26}$$

and $\nabla\varkappa_i(\mathbf{u}) = d_i^T\mathbf{v}$. Hence we have

$$\nabla\varphi_{\eta_j}(\|d_i\mathbf{u}_j\|) = \varphi'_{\eta_j}(\varkappa_i(\mathbf{u}_j))\nabla\varkappa_i(\mathbf{u}_j) = \begin{cases} \varphi'_{\eta_j}(\varkappa_i(\mathbf{u}_j))d_i^T\dfrac{d_i\mathbf{u}_j}{\|d_i\mathbf{u}_j\|}, & \text{if } \|d_i\mathbf{u}_j\| > \eta_j; \\ \varphi'_{\eta_j}(\varkappa_i(\mathbf{u}_j))d_i^T\dfrac{d_i\mathbf{u}_j}{\eta_j}, & \text{if } \|d_i\mathbf{u}_j\| \leq \eta_j. \end{cases} \tag{3.27}$$

Let $\mathbf{u}_j \to \mathbf{u}^*$ and $\eta_j \downarrow 0$ on both sides of (3.27). From (3.23) and the fact $\left\|\frac{d_i\mathbf{u}_j}{\eta_j}\right\| \leq 1$, (3.21) follows immediately.

Next we prove (3.22). From (3.21), we have

$$\begin{aligned} \lim_{\substack{\mathbf{u}_j \to \mathbf{u}^* \\ \eta_j \downarrow 0}} \nabla F_{\eta_j}(\mathbf{u}_j) &= \sum_{i=1}^n \lim_{\substack{\mathbf{u}_j \to \mathbf{u}^* \\ \eta_j \downarrow 0}} \nabla\varphi_{\eta_j}(\|d_i\mathbf{u}_j\|) + \lim_{\mathbf{u}_j \to \mathbf{u}^*} \frac{\alpha}{2}\nabla(\|H\mathbf{u}_j - \mathbf{f}\|^2) \\ &\in \sum_{i=1}^n \mathring{\partial}\varphi(\|d_i\mathbf{u}^*\|) + \frac{\alpha}{2}\nabla(\|H\mathbf{u}^* - \mathbf{f}\|^2). \end{aligned} \tag{3.28}$$

Furthermore, from (3.24)–(3.25) and Definition 2.3, $\varphi(\|d_i\mathbf{u}\|)$ is regular at any $\mathbf{u} \in \Omega$, in particular $\mathbf{u} = \mathbf{u}^*$. Then, by Remark 2.1 it holds that

$$\begin{aligned} &\sum_{i=1}^n \mathring{\partial}\varphi(\|d_i\mathbf{u}^*\|) + \frac{\alpha}{2}\nabla(\|H\mathbf{u}^* - \mathbf{f}\|^2) \\ &= \mathring{\partial}\Big(\sum_{i=1}^n \varphi(\|d_i\mathbf{u}^*\|) + \frac{\alpha}{2}\|H\mathbf{u}^* - \mathbf{f}\|^2\Big) = \mathring{\partial}F(\mathbf{u}^*). \end{aligned} \tag{3.29}$$

The combination of (3.28) and (3.29) gives (3.22).

Now we are ready to present the convergence result for ESDA.

Theorem 3.3 *Let $\{\mathbf{u}_k\}$ is the sequence generated by the* ESDA *with any $\mathbf{u}_0 \in \Omega$, $\delta > 0$, $\varepsilon_{\mathrm{tol}} = 0$, and the maximum number of iterations $K = \infty$. Let $\{\mathbf{u}_{k_l+1}\}$ be the subsequence of $\{u_k\}$, where the reduction criterion in Step 2 is satisfied for $k = k_l$ and $l = 1, 2, \cdots$ Then the following statements hold*:

1. *$\{\mathbf{u}_{k_l+1}\}$ has at least one accumulation point on Ω.*

2. *If $\varphi(t)$ is continuously differentiable on $[0, +\infty)$, every accumulation point of $\{\mathbf{u}_{k_l+1}\}$ is a Clarke stationary point of problem* (2.1).

3. *If $\varphi(t)$ is continuously differentiable only on $(0, +\infty)$, then every accumulation point of $\{\mathbf{u}_{k_l+1}\}$ is an affine-scaled Clarke stationary point of model* (2.1)*, i.e., if $\mathbf{u}^*$ is an accumulation*

point of $\{\mathbf{u}_{k_l+1}\}$*, then there is a*

$$\mathbf{d} \in Z_{\mathbf{u}^*}^T \mathring{\partial}\Big(\sum_{i\in I_{\mathbf{u}^*}^c} \varphi(\|d_i\mathbf{u}^*\|) + \frac{\alpha}{2}\|H\mathbf{u}^* - \mathbf{f}\|^2\Big)$$

satisfying

$$\langle \mathbf{d},\, Z_{\mathbf{u}^*}^T(\mathbf{u}-\mathbf{u}^*)\rangle \geq 0 \quad \textit{for all } \mathbf{u}\in\Omega\cap\{\mathbf{u}:\, \mathbf{u}=Z_{\mathbf{u}^*}\mathbf{v} \textit{ for some } \mathbf{v}\}, \tag{3.30}$$

where $Z_{\mathbf{u}_*}$ *is an* $n\times r$ *matrix whose columns are an orthonormal basis for the null space of* $\{d_i^x, d_i^y : i\in I_{\mathbf{u}^*}\}$ *with* $r>0$ *being its dimension,* $I_{\mathbf{u}^*}=\{i=1,\cdots,n : \|d_i\mathbf{u}^*\|=0\}$*, and* $I_{\mathbf{u}^*}^c=\{i=1,\cdots,n : \|d_i\mathbf{u}^*\|\neq 0\}$*.*

Proof The first statement is evident due to the boundedness of Ω and $\{\mathbf{u}_k\}\subset\Omega$.

To prove Part 2, denote by $\{\mathbf{u}_{k_m+1}\}$ the convergent subsequence of $\{\mathbf{u}_{k_l+1}\}$ to an accumulation point $\widehat{\mathbf{u}}\in\Omega$ as $m\to\infty$. Denote also by η_{k_m} the corresponding η_{k_l} used in the iteration to generate $\overline{\mathbf{u}}_{k_m+1}$ and $\mathbf{u}_{k_m+1}$. Since the reduction criterion in Step 2 is satisfied for $k=k_l$ and $l=1,2,\cdots$, we have

$$\|\overline{\mathbf{u}}_{k_m+1}-\mathbf{u}_{k_m}\| < \tau\eta_{k_m}\alpha_{k+1}, \tag{3.31}$$

where α_{k+1} is the stepsize used for the m-th segment to minimize $F_{\eta_{k_m}}(\mathbf{u})$.

By the optimality condition for generating $\overline{\mathbf{u}}_{k_m+1}$, it holds that

$$\langle \overline{\mathbf{u}}_{k_m+1}-\mathbf{u}_{k_m}+\alpha_{k+1}\nabla F_{\eta_{k_m}}(\mathbf{u}_{k_m}),\, \mathbf{u}-\overline{\mathbf{u}}_{k_m+1}\rangle \geq 0, \quad \forall \mathbf{u}\in\Omega, \tag{3.32}$$

and thus for any $\mathbf{u}\in\Omega$,

$$\begin{aligned}\langle \nabla F_{\eta_{k_m}}(\mathbf{u}_{k_m}),\, \mathbf{u}-\overline{\mathbf{u}}_{k_m+1}\rangle &\geq -\frac{1}{\alpha_{k+1}}\langle \overline{\mathbf{u}}_{k_m+1}-\mathbf{u}_{k_m}, \mathbf{u}-\overline{\mathbf{u}}_{k_m+1}\rangle \\ &\geq -\frac{1}{\alpha_{k+1}}\|\overline{\mathbf{u}}_{k_m+1}-\mathbf{u}\|\cdot\|\overline{\mathbf{u}}_{k_m+1}-\mathbf{u}_{k_m}\| \geq -\tau\eta_{k_m}\mathrm{diam}(\Omega),\end{aligned} \tag{3.33}$$

where $\mathrm{diam}(\Omega)$ is the diameter of Ω, and the last inequality is from (3.31).

Recall that as $m\to\infty$,

$$\eta_{k_m}=\eta_0\tau_1^m\downarrow 0, \quad \overline{\mathbf{u}}_{k_m+1}\to\widehat{\mathbf{u}}.$$

Denote $\mathbf{d}:=\lim_{m\to\infty}\nabla F_{\eta_{k_m}}(\mathbf{u}_{k_m})$. Now letting $m\to\infty$ on both sides of (3.33), we get

$$\langle \mathbf{d},\, \mathbf{u}-\widehat{\mathbf{u}}\rangle \geq 0. \tag{3.34}$$

By Lemma 3.2, $\mathbf{d}\in\partial^\circ F(\widehat{\mathbf{u}})$. Hence, by Definion 2.2 $\widehat{u}$ is a Clarke stationary point of problem (2.1).

To prove the last statement, let $\mathbf{u}^*$ be an accumulation point of $\{\mathbf{u}_{k_l+1}\}$ and $\{\mathbf{u}_{k_m+1}\}$ is the subsequence of $\{\mathbf{u}_{k_l+1}\}$ converging to $\mathbf{u}^*\in\Omega$ as $m\to\infty$, and η_{k_m} the corresponding smoothing parameter to generate $\overline{\mathbf{u}}_{k_m+1}$ and $\mathbf{u}_{k_m+1}$. When $\varphi(t)$ is continuously differentiable in $(0,+\infty)$ rather than $[0,+\infty)$, from the proof of Lemma 3.2 we have

$$\lim_{\substack{\mathbf{u}_{k_m}\to\mathbf{u}^*\\ \eta_{k_m}\downarrow 0}} \nabla\varphi_{\eta_{k_m}}(\|d_i\mathbf{u}_{k_m}\|)\in\mathring{\partial}\varphi(\|d_i\mathbf{u}^*\|) \quad \text{if } \|d_i\mathbf{u}^*\|\neq 0.$$

Now let

$$\mathbf{d}=\lim_{\substack{\mathbf{u}_{k_m}\to\mathbf{u}^*\\ \eta_{k_m}\downarrow 0}} Z_{\mathbf{u}^*}^T\nabla F_{\eta_{k_m}}(\mathbf{u}_{k_m}).$$

Then

$$\begin{aligned}\mathbf{d} &= \lim_{\substack{\mathbf{u}_{k_m}\to\mathbf{u}^*\\ \eta_{k_m}\downarrow 0}} \sum_{i=1}^{n} Z_{\mathbf{u}^*}^T \nabla\varphi_{\eta_{k_m}}(\|d_i\mathbf{u}_{k_m}\|) + \lim_{\mathbf{u}_{k_m}\to\mathbf{u}^*} \frac{\alpha}{2} Z_{\mathbf{u}^*}^T \nabla(\|H\mathbf{u}_{k_m}-\mathbf{f}\|^2)\\ &= Z_{\mathbf{u}^*}^T \lim_{\substack{\mathbf{u}_{k_m}\to\mathbf{u}^*\\ \eta_{k_m}\downarrow 0}} \sum_{i\in I_{\mathbf{u}^*}^c} \nabla\varphi_{\eta_{k_m}}(\|d_i\mathbf{u}_{k_m}\|) + \frac{\alpha}{2} Z_{\mathbf{u}^*}^T \nabla(\|H\mathbf{u}^*-\mathbf{f}\|^2)\\ &\in Z_{\mathbf{u}^*}^T \sum_{i\in I_{\mathbf{u}^*}^c} \mathring{\partial}\varphi(\|d_i\mathbf{u}^*\|) + \frac{\alpha}{2} Z_{\mathbf{u}^*}^T \mathring{\partial}(\|H\mathbf{u}^*-\mathbf{f}\|^2)\\ &= Z_{\mathbf{u}^*}^T \mathring{\partial}\Big(\sum_{i\in I_{\mathbf{u}^*}^c} \varphi(\|d_i\mathbf{u}^*\|) + \frac{\alpha}{2}\|H\mathbf{u}^*-\mathbf{f}\|^2\Big), \end{aligned} \tag{3.35}$$

where the second equality uses (3.27) and $Z_{\mathbf{u}^*}^T d_i^T = 0$ for all $i \in I_{\mathbf{u}^*}$, and the last equality can be obtained by a discussion similar to get (3.29).

Let $Z_{u^*}\overline{\mathbf{v}}_{k_m+1}$ be the Euclidean projection of $\overline{\mathbf{u}}_{k_m+1}$ onto the intersection between Ω and the null space $\{\mathbf{u}:\ \mathbf{u} = Z_{u^*}\mathbf{v} \text{ for some } \mathbf{v}\}$. Observe that (3.33) and the fact that $I = Z_{\mathbf{u}^*}^\top Z_{\mathbf{u}^*}$ imply that, for any $\mathbf{u} \in \Omega \cap \{\mathbf{u}:\ \mathbf{u} = Z_{\mathbf{u}^*}\mathbf{v} \text{ for some } \mathbf{v}\}$,

$$\begin{aligned} &\langle Z_{\mathbf{u}^*}^\top \nabla F_{\eta_{k_m}}(\mathbf{u}_{k_m}),\, \mathbf{v}-\overline{\mathbf{v}}_{k_m+1}\rangle = \langle \nabla F_{\eta_{k_m}}(\mathbf{u}_{k_m}),\, Z_{\mathbf{u}^*}(\mathbf{v}-\overline{\mathbf{v}}_{k_m+1})\rangle\\ &= \langle \nabla F_{\eta_{k_m}}(\mathbf{u}_{k_m}),\, \mathbf{u}-\overline{\mathbf{u}}_{k_m+1}\rangle - \langle \nabla F_{\eta_{k_m}}(\mathbf{u}_{k_m}),\, Z_{\mathbf{u}^*}\overline{\mathbf{v}}_{k_m+1}-\overline{\mathbf{u}}_{k_m+1}\rangle\\ &\geq -\frac{1}{\alpha_{k+1}}\|\overline{\mathbf{u}}_{k_m+1}-\mathbf{u}\|\cdot\|\overline{\mathbf{u}}_{k_m+1}-\mathbf{u}_{k_m}\| - \langle \nabla F_{\eta_{k_m}}(\mathbf{u}_{k_m}),\, Z_{\mathbf{u}^*}\overline{\mathbf{v}}_{k_m+1}-\overline{\mathbf{u}}_{k_m+1}\rangle\\ &\geq -\tau\eta_{k_m}\mathrm{diam}(\Omega) - \|\nabla F_{\eta_{k_m}}\mathbf{u}_{k_m})\|\cdot\|Z_{\mathbf{u}^*}\overline{\mathbf{v}}_{k_m+1}-\overline{\mathbf{u}}_{k_m+1}\|. \end{aligned} \tag{3.36}$$

Let $m \to \infty$, which implies $\mathbf{u}_{k_m} \to \mathbf{u}^* \in \{\mathbf{u}:\ \mathbf{u} = Z_{\mathbf{u}^*}\mathbf{v} \text{ for some } \mathbf{v}\}$, $Z_{\mathbf{u}^*}\overline{\mathbf{v}}_{k_m+1} \to \mathbf{u}^*$ and $\eta_{k_m} \downarrow 0$. Thus, we have $\|Z_{\mathbf{u}^*}\overline{\mathbf{v}}_{k_m+1} - \overline{\mathbf{u}}_{k_m+1}\| \to 0$. We therefore obtain (3.30) immediately. This combined with (3.35) leads to the desired.

4 Numerical Experiments

In this section, we consider a class of box constrained problem (2.1), where $\Omega = \{\mathbf{u} \in \mathbb{R}^n : l_1\mathbf{e} \leq \mathbf{u} \leq l_2\mathbf{e}\}$ and $\mathbf{e} = (1, 1, \cdots, 1)^T \in \mathbb{R}^n$ in the application of sparse view CT reconstruction. To exam the performance of the proposed algorithm, we compare it to the standard smoothing gradient descent method to minimize the same objective function with and without box constraints, named as (BSGD for short) and (SGD for short) respectively. The BSGD is the same as the proposed algorithm without Steps 1.1–1.4. We also compare the proposed algorithm with accelerated smoothing algorithm (ESA for short) in [39] for corresponding unconstrained problem. All numerical experiments are conducted in MATLAB R2016a running on a PC with Intel Core i5 CPU at 1.6GHz and 8G of memory. Besides visual evaluation we also use peak signal-to-noise ratio (PSNR for short) to evaluate the quality of reconstruction. The PSNR is defined by

$$\mathrm{PSNR}(\mathbf{u}, \underline{\mathbf{u}}) = 10\log_{10}\frac{\underline{\mathbf{u}}_{\max}^2 \cdot N_1N_2}{\|\mathbf{u}-\underline{\mathbf{u}}\|}\mathrm{dB},$$

where $\mathbf{u}$ and $\underline{\mathbf{u}}$ are restored and original images, N_1N_2 is the total number of pixels of an image with the same rows and columns, and $\underline{\mathbf{u}}_{\max}$ represents the maximum pixel value of the image.

CT reconstruction problem can be modeled as an inverse problem $\mathbf{f} = H\mathbf{u} + \boldsymbol{\upsilon}$, where $\mathbf{u}$ is the image to be reconstructed, H is the system matrix for CT scanner depending on the beam geometer, $\mathbf{f}$ is the noisy sinogram measurements and $\boldsymbol{\upsilon}$ is the noise with normal distribution.

Here, we consider 2D parallel-beam CT with an $N \times N$ domain, using $\widehat{p}$ parallel rays for each angle as in [23]. The regular view CT has angles $0, 1, \cdots, 179$, whereas the sparse view CT that we deal with has angles $0, 5, 10, \cdots, 175$ (i.e., $N_{\widehat{p}}$= 36 rotated projection views). Number of parallel rays for each angle and the distance from the first ray to the last ray are set to be the nearest integer to $\sqrt{2}N$ and $\sqrt{2}N$, respectively. H is implemented by Radon transform. The images used in this experiment are the "Shepp-Logan" phantom (128×128), "NCAT" phantom (256×256) and the cerebral phantom (512×512) (see [50]) shown in Figure 1. The corresponding noisy sinograms for parallel-beam scanning with $N_{\widehat{p}} = 36$ are also presented in this figure.

Figure 1 Test images and the corresponding sinogram observations when $N_{\widehat{p}} = 36$.

Figure 2 presents the reconstruction results after 100 iterations by using aforementioned four different algorithms with three potential functions $\varphi(t) = t^{0.8}$, $\varphi(t) = \ln(1 + 0.5t)$ and $\varphi(t) = \frac{0.5t}{1+0.5t}$ when the Gaussian noise level is $0.005\|f\|_\infty$. Although the convergence of SGD and ESA might fail for $\varphi(t) = \frac{0.5t}{1+0.5t}$ due to the lack of coercivity but they can still work experimentally. In both SGD and ESA, we fix the parameters $\eta_0 = 0.01$, $\delta = 10^{-3}$, $\rho = 0.25$ and $\tau_1 = 0.5$ as that in [39], while tune the model parameter α, s_0 and τ. In BSGD, we fix $\eta_0 = 0.01$, $\rho = 0.25$, $\delta = 10^{-5}$, $\tau_1 = 0.5$ and tune α and τ. In the proposed algorithm, we also fix $\eta_0 = 0.01$, $\rho = 0.25$, $\delta = 10^{-5}$, $\tau_1 = 0.5$, and tune α, s_0, s_{k+1} and τ. Moreover, we set $l_1 = -5$ and $l_2 = 5$ in both BSGD and our algorithm. For a fair comparison, each algorithm is tuned to get the highest PSNR values. From Figure 2, under all three potential functions, one can see that BSGD yields higher PSNR values than SGD, while the proposed algorithm always performs better than BSGD and obtains comparable results with ESA. Similar phenomena can be found from Figure 3 visually and quantitatively, where all parameters in these four algorithms are tuned as that in Figure 2.

In Figure 4, we present reconstruction results on "cerebral" after 100 iterations with $N_{\widehat{p}} = 36$. In this experiment, we adopt same rules as above to tune parameters in all compared algorithms. The PSNR of reconstructed images from SGD, ESA, BSGD and proposed algorithms for three different regularization functions are shown under the image in this figure. The improvement of PSNR by the proposed algorithm is about 1.01dB, 0.20dB, 0.90dB increase on average for those regularization functions compared to SGD, ESA and BSGD, respectively. To better visualize the results, the zoomed regions are shown in Figure 5.

Figure 6 gives the PSNR values of reconstruction versus number of iterations on "Shepp-Logan", "NCAT" and "cerebral" images obtained by BSGD and the proposed algorithm. One can observe that for potential functions $\varphi(t) = \ln(1 + 0.5t)$ and $\varphi(t) = \frac{0.5t}{1+0.5t}$ the PSNR values resulted from BSGD are similar for all of three images, while the PSNR values produced by the proposed algorithm increase faster than BSGD after 40 iterations in all experiments.

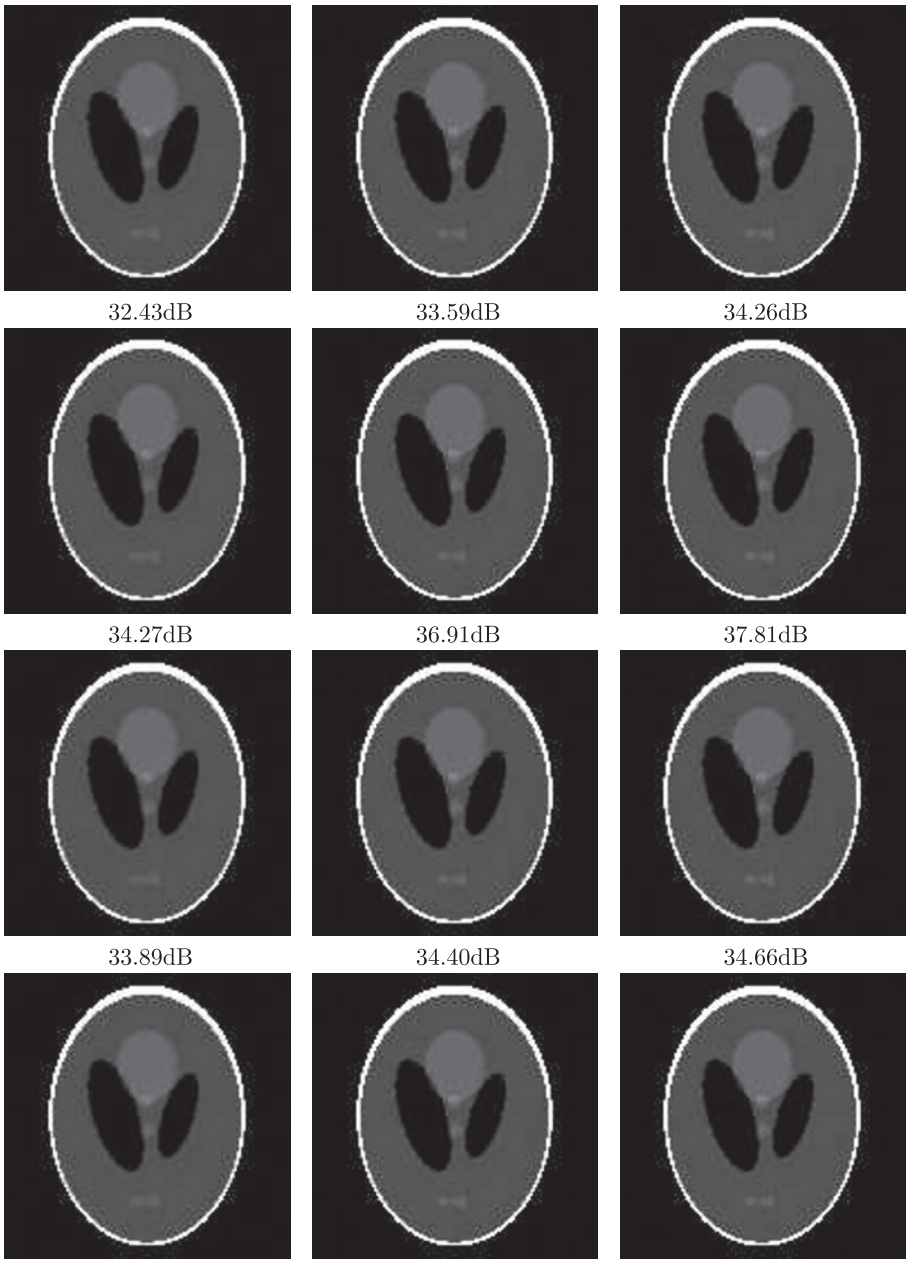

Figure 2 Results after 100 iterations on "Shepp-Logan". From the first column to the third column: Reconstructions by different potential functions. From the first row to the fourth row: Reconstructions by SGD, ESA, BSGD and the proposed algorithm. PSNR values are listed.

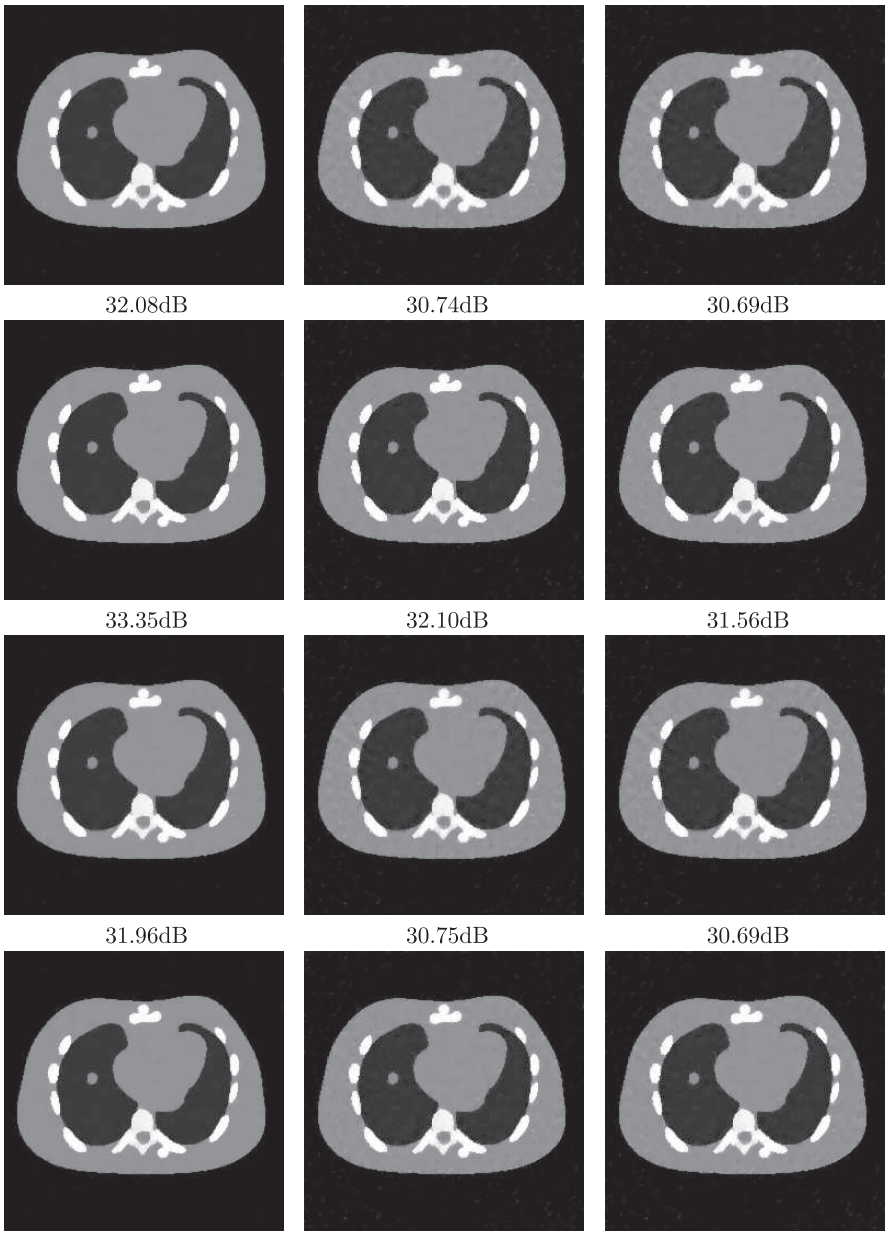

Figure 3 Results after 100 iterations on "NCAT". From the first column to the third column: Reconstructions by different potential functions. From the first row to the fourth row: Reconstructions by SGD, ESA, BSGD and the proposed algorithm. PSNR values are listed.

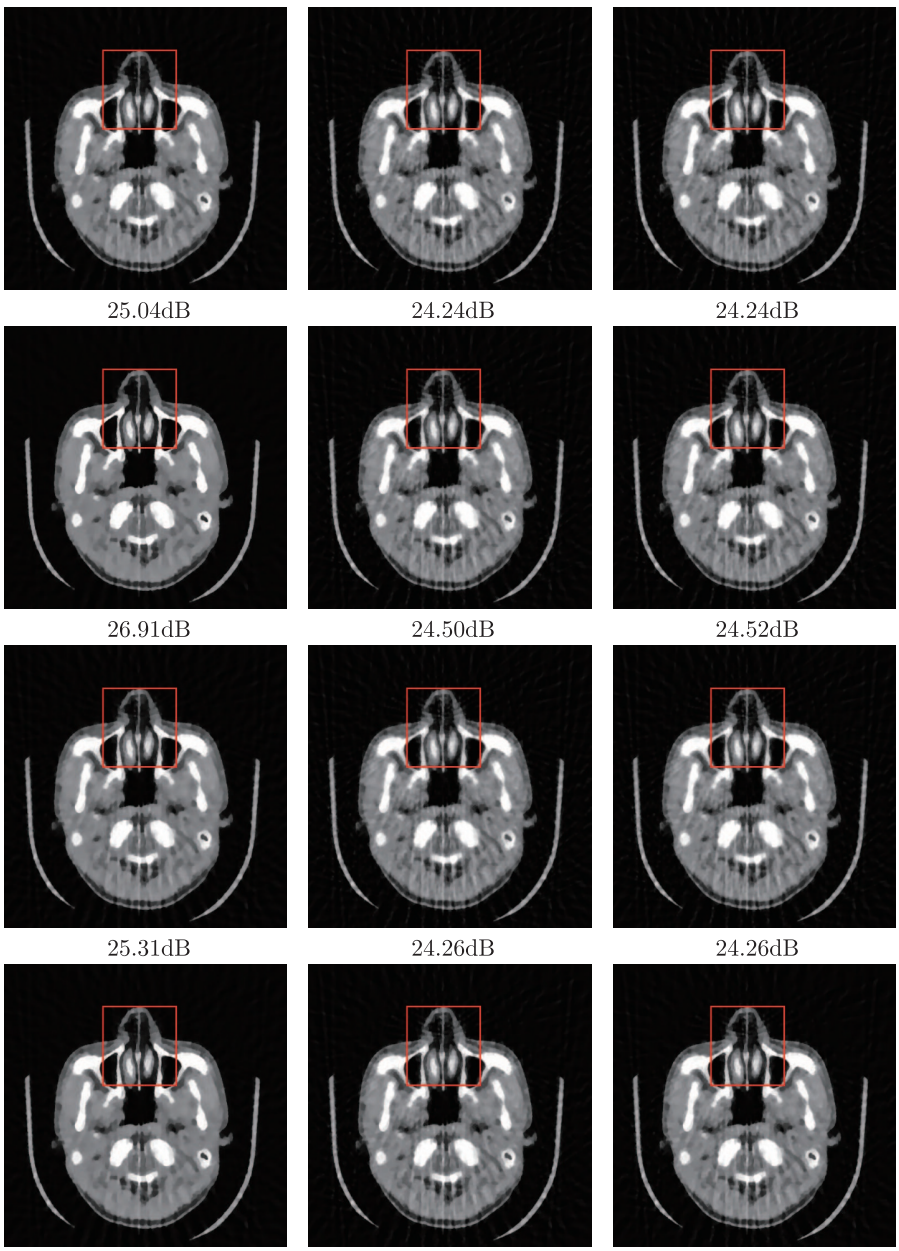

Figure 4 Results after 100 iterations on "cerebral". From the first column to the third column: Reconstructions by different potential functions. From the first row to the fourth row: Reconstructions by SGD, ESA, BSGD and the proposed algorithm. PSNR values are listed.

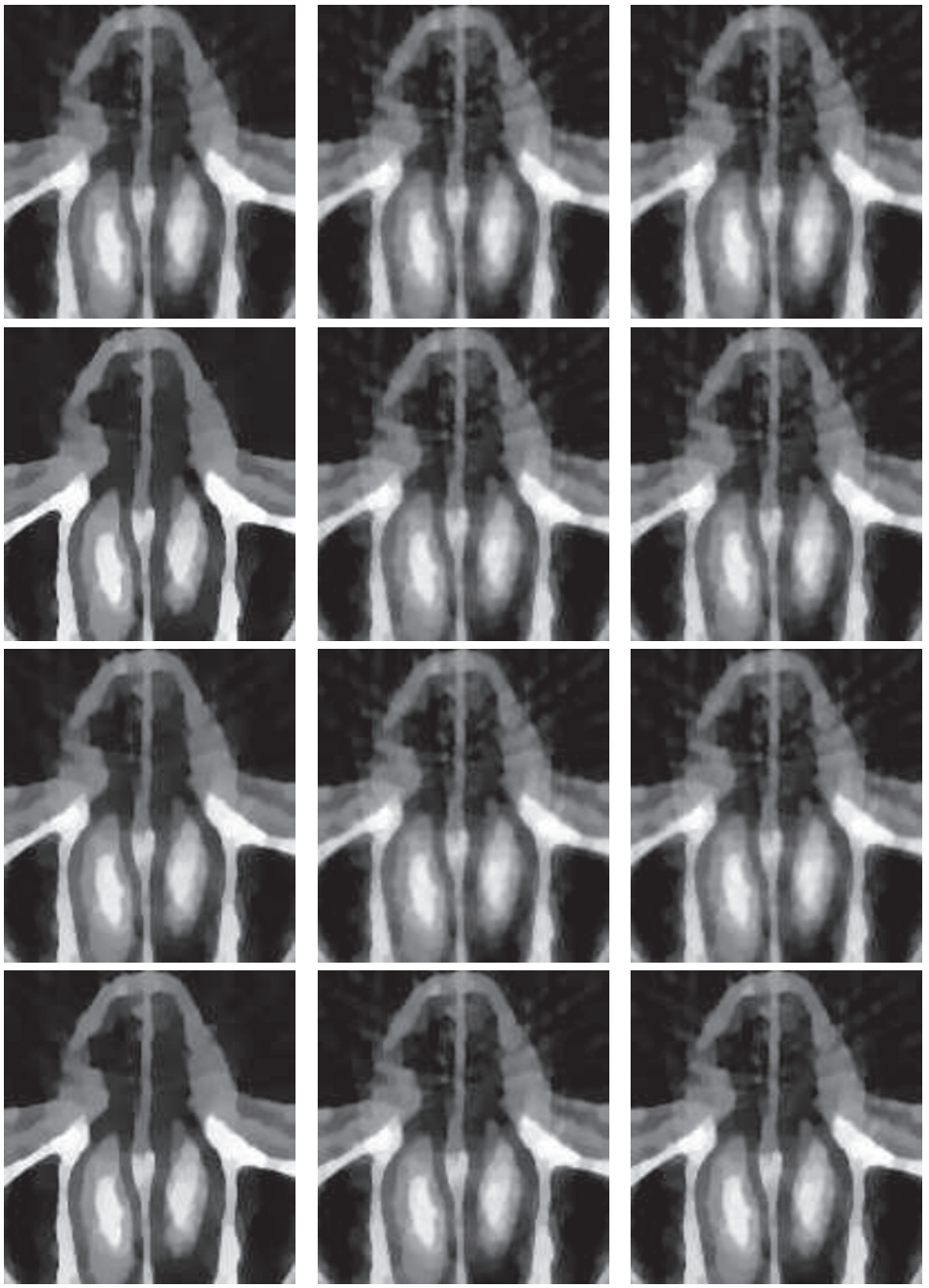

Figure 5 The zoomed regions corresponding to results in Figure 4.

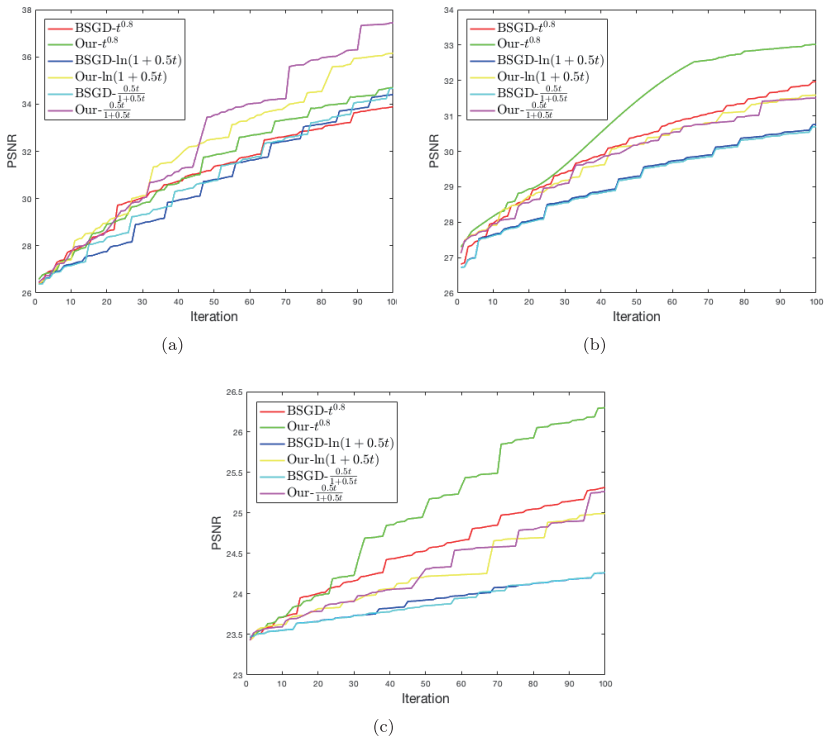

Figure 6 From left to right: The PSNR value of the recovered images by BSGD and the proposed algorithm versus iteration with three potential functions for "Shepp-Logan", "NCAT" and "cerebral".

5 Conclusion

In this paper, we proposed a smoothing inexact projected gradient descent with extrapolation to solve a class of constrained nonsmooth nonconvex minimization problems. The inexact projected gradient descent with extrapolation is applied to improve the performance of minimizing the corresponding smoothed nonconvex problem. Combined with a safe-guarding policy and adaptively updating the smoothing parameter, the proposed algorithm guarantees that any accumulation point of the sequence generated by this algorithm is an (affine-scaled) Clarke stationary point of the original nonsmooth and nonconvex problem. Numerical experiments and comparisons indicated that the proposed algorithm performed better visually and quantitatively than nonaccelerated gradient descent algorithms for the same model with or without box constraints for CT reconstruction problem.

References

[1] Attouch, H., Bolte, J. and Svaiter, B. F., Convergence of descent methods for semi-algebraic and tame problems: Proximal algorithms, forward-backward splitting, and regularized Gauss-Seidel methods, *Math. Program.*, **137**, 2013, 91–129.

[2] Bao, C. L., Dong, B., Hou, L. K., et al., Image restoration by minimizing zero norm of wavelet frame coefficients, *Inverse Probl.*, **32**, 2016, 115004.

[3] Bian, W. and Chen, X. J., Linearly constrained non-Lipschitz optimization for image restoration, *SIAM J. Imaging Sci.,* **8**, 2015, 2294–2322.

[4] Bian, W. and Chen, X. J., Optimality and complexity for constrained optimization problems with nonconvex regularization, *Math. Oper. Res.,* **42**, 2017, 1063–1084.

[5] Bian, W. and Chen, X. J., A smoothing proximal gradient algorithm for nonsmooth convex regression with cardinality penalty, *SIAM J. Numer. Anal.,* **58**, 2020, 858–883.

[6] Bian, W., Chen, X. J. and Ye, Y. Y., Complexity analysis of interior point algorithms for non-Lipschitz and nonconvex minimization, *Math. Program.*, **149**, 2015, 301–327.

[7] Bonettini, S., Loris, I., Porta, F., et al., On the convergence of a linesearch based proximal-gradient method for nonconvex optimization, *Inverse Probl.,* **33**, 2017, 055005.

[8] Burke, J. V., Ferris, M. C. and Qian, M. J., On the Clarke subdifferential of the distance function of a closed set, *J. Math. Anal. Appl.,* **166**, 1992, 199–213.

[9] Candes, E. J., Wakin, M. B. and Boyd, S. P., Enhancing sparsity by reweighted ℓ_1 minimization, *J. Fourier Anal. Appl.,* **14**, 2008, 877–905.

[10] Chan, R. H., Tao, M. and Yuan, X. M., Constrained total variation deblurring models and fast algorithms based on alternating direction method of multipliers, *SIAM J. Imaging Sci.,* **6**, 2013, 680–697.

[11] Chen, X. J., Smoothing methods for nonsmooth, nonconvex minimization, *Math. Program.,* **134**, 2012, 71–99.

[12] Chen, X. J., Ng, M. K. and Zhang, C., Non-Lipschitz ℓ_p-regularization and box constrained model for image restoration, *IEEE Trans. Image Process.,* **21**, 2012, 4709–4721.

[13] Chen, X. J., Niu, L. F. and Yuan, Y. X., Optimality conditions and a smoothing trust region Newton method for nonLipschitz optimization, *SIAM J. Optim.,* **23**, 2013, 1528–1552.

[14] Chen, X. J. and Zhou, W. J., Smoothing nonlinear conjugate gradient method for image restoration using nonsmooth nonconvex minimization, *SIAM J. Imaging Sci.,* **3**, 2010, 765–790.

[15] Chen, Y. M., Liu, H. C., Ye, X. J. and Zhang, Q. C., Learnable descent algorithm for nonsmooth nonconvex image reconstruction, *SIAM J. Imaging Sci.,* **14**, 2021, 1532–1564.

[16] Clarke, F. H., Optimization and Nonsmooth Analysis, John Wiley and Sons, Philadelphia, 1990.

[17] Foucart, S. and Lai, M. J., Sparsest solutions of underdetermined linear systems via ℓ_q-minimization for $0 < q < 1$, *Appl. Comput. Harmon. Anal.,* **26**, 2009, 395–407.

[18] Fukushima, M. and Mine, H., A generalized proximal point algorithm for certain non-convex minimization problems, *International Journal of Systems Science,* **12**, 1981, 989–1000.

[19] Gao, Y. M. and Wu, C. L., On a general smoothly truncated regularization for variational piecewise constant image restoration: construction and convergent algorithms, *Inverse Probl.,* **36**, 2020, 045007.

[20] Ghadimi, S. and Lan, G. H., Accelerated gradient methods for nonconvex nonlinear and stochastic programming, *Math. Program.,* **156**, 2016, 59–99.

[21] Gu, B., Wang, D., Huo, Z. Y. and Huang, H., Inexact proximal gradient methods for non-convex and non-smooth optimization, in AAAI, **32**, 2018.

[22] Hintermüller, M. and Wu, T., Nonconvex TV^q-models in image restoration: Analysis and a trust-region regularization based superlinearly convergent solver, *SIAM J. Imaging Sci.,* **6**, 2013, 1385–1415.

[23] Kak, A. C. and Slaney, M., Principles of Computerized Tomographic Imaging, Philadelphia, PA, USA: SIAM, 2001.

[24] Kong, W. W., Melo, J. G. and Monteiro, R. D., Complexity of a quadratic penalty accelerated inexact proximal point method for solving linearly constrained nonconvex composite programs, *SIAM J. Optim.,* **29**, 2019, 2566–2593.

[25] Kong, W. W., Melo, J. G. and Monteiro, R. D., An efficient adaptive accelerated inexact proximal point method for solving linearly constrained nonconvex composite problems, *Comput. Optim. Appl.,* **76**, 2020, 305–346.

[26] Lai, M. J. and Xu, Y. Y. and Yin, W. T., Improved iteratively reweighted least squares for unconstrained smoothed ℓ_q minimization, *SIAM J. Numer. Anal.,* **51**, 2013, 927–957.

[27] Li, H. and Lin, Z. C., Accelerated proximal gradient methods for nonconvex programming, in NIPS, 2015, 379–387.

[28] Li, Q. W., Zhou, Y., Liang, Y. B. and Varshney, P. K., Convergence analysis of proximal gradient with momentum for nonconvex optimization, in ICML, PMLR, 2017, 2111–2119.

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

[30] Liu, Z. F., Wu, C. L. and Zhao, Y, N., A new globally convergent algorithm for non-Lipschitz $l_p - l_q$ minimization, *Adv. Comput. Math.,* **45**, 2019, 1369–1399.

[31] Nesterov, Y., Smooth minimization of non-smooth functions, *Math. Program.,* **103**, 2005, 127–152.

[32] Nikolova, M., Analysis of the recovery of edges in images and signals by minimizing nonconvex regularized least-squares, *SIAM J. Multiscale Model. Simul.,* **4**, 2005, 960–991.

[33] Nikolova, M., Ng, M. K. and Tam, C. P., Fast nonconvex nonsmooth minimization methods for image restoration and reconstruction, *IEEE Trans. Image Process.,* **19**, 2010, 3073–3088.

[34] Nikolova, M., Ng, M. K., Zhang, S. Q. and Ching, W. K., Efficient reconstruction of piecewise constant images using nonsmooth nonconvex minimization, *SIAM J. Imaging Sci.,* **1**, 2008, 2–25.

[35] Ochs, P., Chen, Y. J., Brox, T. and Pock, T., iPiano: Inertial proximal algorithm for nonconvex optimization, *SIAM J. Imaging Sci.,* **7**, 2014, 1388–1419.

[36] Ochs, P., Dosovitskiy, A., Brox, T. and Pock, T., On iteratively reweighted algorithms for nonsmooth nonconvex optimization in computer vision, *SIAM J. Imaging Sci.,* **8**, 2015, 331–372.

[37] Rudin, L. I., Osher, S. and Fatemi, E., Nonlinear total variation based noise removal algorithms, *Phys. D, Nonlinear Phenomena,* **60**, 1992, 259–268.

[38] Villa, S., Salzo, S., Baldassarre, L. and Verri, A., Accelerated and inexact forward-backward algorithms, *SIAM J. Optim,* **23**, 2013, 1607–1633.

[39] Wang, W. and Chen, Y. M., An accelerated smoothing gradient method for nonconvex nonsmooth minimization in image processing, *J. Sci. Comput.,* **90**, 2022, 1–28.

[40] Wang, W., Wu, C. L. and Gao, Y. M., A nonconvex truncated regularization and box-constrained model for CT reconstruction, *Inverse Probl. Imag.,* **14**, 2020, 867–890.

[41] Wang, W., Wu, C. L. and Tai, X. C., A globally convergent algorithm for a constrained non-Lipschitz image restoration model, *J. Sci. Comput.,* **83**, 2020, 1–29.

[42] Wen, B., Chen, X. J. and Pong, T. K., Linear convergence of proximal gradient algorithm with extrapolation for a class of nonconvex nonsmooth minimization problems, *SIAM J. Optim.,* **27**, 2017, 124–145.

[43] Wu, C. L. and Tai, X. C., Augmented Lagrangian method, dual methods, and split Bregman iteration for ROF, vectorial TV, and high order models, *SIAM J. Imaging Sci.,* **3**, 2010, 300–339.

[44] Wu, Z. and Li, M., General inertial proximal gradient method for a class of nonconvex nonsmooth optimization problems, *Comput. Optim. Appl.,* **73**, 2019, 129–158.

[45] Xu, Z. B., Chang, X. Y., Xu, F. M. and Zhang, H., $\ell_{\frac{1}{2}}$ regularization: A thresholding representation theory and a fast solver, *IEEE Trans. Neural Netw. Learn. Syst.,* **23**, 2012, 1013–1027.

[46] Yang, L., Proximal gradient method with extrapolation and line search for a class of nonconvex and nonsmooth problems, 2017, arXiv:1711.06831.

[47] Yao, Q. M., Kwok, J. T., Gao, F., et al., Efficient inexact proximal gradient algorithm for nonconvex problems, 2016, arXiv:1612.09069.

[48] Zeng, C. and Jia, R. and Wu, C. L., An iterative support shrinking algorithm for non-Lipschitz optimization in image restoration, *J. Math. Imaging Vis.,* **61**, 2019, 122–139.

[49] Zeng, C. and Wu, C. L., On the edge recovery property of noncovex nonsmooth regularization in image restoration, *SIAM J. Numer. Anal.,* **56**, 2018, 1168–1182.

[50] Zhang, H. M., Dong, B. and Liu, B. D., A reweighted joint spatial-radon domain CT image reconstruction model for metal artifact reduction, *SIAM J. Imaging Sci.,* **11**, 2018, 707–733.

[51] Zhang, X. and Zhang, X. Q., A new proximal iterative hard thresholding method with extrapolation for ℓ_0 minimization, *J. Sci. Comput.,* **79**, 2019, 809–826.