Neurocomputing 371 (2020) 51–66

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

A robust recovery algorithm with smoothing strategies

Yuli Sun, Lin Lei, Xiao Li, Ming Li, Gangyao Kuang*

College of Electronic science, National University of Defense Technology, Changsha 410073, China

ARTICLE INFO

Article history:
Received 12 May 2019
Revised 14 August 2019
Accepted 17 August 2019
Available online 27 August 2019

Communicated by Dr. Jie Wang

Keywords:
Robust sparse recovery
Impulsive noise
Smoothing method
Proximal gradient
Non-convex regularization

ABSTRACT

This paper addresses the robust sparse recovery problem in the presence of impulsive measurement noise. In order to overcome the poor performance of ℓ_2-norm loss function with the outliers under the impulsive noise, we employ the ℓ_1-norm as the loss function for the residual error, which is less sensitive to outliers in the measurements than the popular ℓ_2-loss. To rise to the challenges introduced by the non-smooth problem, we first employ two smoothing strategies to approximate the ℓ_1-norm loss function: one introduces a relaxation factor in the ℓ_1-norm and the other uses the infimal convolution smoothing technique to transform it into a smooth counterpart. Both of them can approximate the ℓ_1-norm with arbitrary degree of accuracy and provide a Lipschitz continuous gradient loss function. Then, we employ the accelerated proximal gradient (APG) and monotone APG (mAPG) frameworks for the convex and non-convex regularization functions, respectively. The convergence performance is discussed for generalized regularization penalty. The simulation result demonstrates our conclusions and indicates that the algorithm proposed in this paper can improve the reconstruction quality.

1. Introduction

1.1. Background

In recent years, sparse optimization is a very attractive field which has been found wide applications, for example, compressive sensing (CS), machine learning and medical imaging. In the CS framework, it can sample the sparse or compressible signals below the Nyquist rate, whilst still allowing perfect reconstruction of the signal [1]. Let $\mathbf{x} \in \mathbb{R}^N$ be the unknown signal, which is sparse, or can be sparsely represented on an appropriate basis or dictionary. CS samples $\mathbf{x}$ with an $M \times N$ measurement matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$, where M is much smaller than N, yielding the measurement vector $\mathbf{y} \in \mathbb{R}^M$. It can be expressed as:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n} \tag{1}$$

where the sensing matrix $\mathbf{A}$ is usually chosen to be a random matrix, such as Gaussian matrix, Bernoulli matrix, or partial Fourier matrix, and $\mathbf{n} \in \mathbb{R}^M$ denotes the measurement error or noise.

Reconstructing the sparse signal $\mathbf{x}$ is an underdetermined problem, which can be formulated as the following minimization problem

$$\min_{\mathbf{x}} F(\mathbf{x}) \equiv f(\mathbf{A}\mathbf{x} - \mathbf{y}) + g(\mathbf{x}) \tag{2}$$

where f is the loss function related to (1), and g is the regularization function to penalize the sparsity of $\mathbf{x}$. Intuitively, $g(\mathbf{x})$ should be the ℓ_0-norm $\|\mathbf{x}\|_0$, representing the number of nonzero elements of $\mathbf{x}$. Unfortunately, minimizing the ℓ_0-norm is equivalent to finding the sparsest solution, which is known to be an NP-hard problem. A favorite and common approach is using the ℓ_1-norm convex approximation, i.e., $g(\mathbf{x}) = \|\mathbf{x}\|_1 = \sum_{i=1}^{N} |x_i|$ instead of the ℓ_0-norm. This convex relaxation model has been widely used in many different fields, such as synthetic aperture radar (SAR) images processing [2], direction of arrival (DOA) estimation [3] and magnetic resonance imaging (MRI) [4]. It has been proved that the sparse signal $\mathbf{x}$ can be recovered by the ℓ_1-norm minimization under some assumptions of the sensing matrix $\mathbf{A}$, such as the restricted isometry property (RIP) [1]. However, the ℓ_1-norm regularization sometimes tends to underestimate high-amplitude components of $\mathbf{x}$ as it uniformly penalizes the amplitude, unlike that all nonzero entries have equal contributions in the ℓ_0-norm. This may lead to failure recovery in some cases [5], such as the undesirable blocky images in the CT [6,7]. To address this issue, many non-convex regularizations, which are interpolated between the ℓ_0-norm and the ℓ_1-norm, have been proposed to approximate the ℓ_0-norm more accurately and bring better reconstructions, recently. This can be illustrated by the ℓ_p (quasi)-norm with $p \in (0, 1)$ [8–10], capped ℓ_1-norm [11,12], reweighted ℓ_1-norm [5], the difference of the ℓ_1 and ℓ_2-norms (ℓ_{1-2}) [13,14], log-sum penalty (LSP) [7], smoothly clipped absolute deviation (SCAD) [15],

* Corresponding author.
E-mail address: kuanggangyao@nudt.edu.cn (G. Kuang).

https://doi.org/10.1016/j.neucom.2019.08.035
0925-2312/

minimax-concave penalty (MCP) [16–18], correntropy induced metric (CIM) penalty [19,20].

On the other hand, $f(\mathbf{Ax}-\mathbf{y})=\|\mathbf{Ax}-\mathbf{y}\|_2^2$, which is the ℓ_2-norm of the residuals, is often used as the loss function to measure the data fidelity. This is because that the measurement noise is usually Gaussian distributed or approximately Gaussian, and the ℓ_2-norm makes the reconstruction problem convex and simplifies the derivation of the recovery algorithms. However, the noise sometimes exhibits non-Gaussian properties in practical applications, such as the impulsive noise, i.e., including salt-and-pepper noise and random-valued noise, which are often found in image processing [18,21]. Under the impulsive noise, due to the fact that the least-squares (LS) based algorithms perform poorly with the outliers, the normal CS recovery algorithm with the ℓ_2-norm loss function is rather inefficient [22]. In order to obtain the robust recovery under the condition of impulsive measurement noise, various sparse optimization algorithms have been proposed recently based on different loss functions. For example, Huber penalty function [22,23], ℓ_p-norm loss with $p\in[0,\ 2)$ in [20,24–27], and maximum correntropy criterion (MCC) based function [19,28] have achieved better performance than the ℓ_2-norm loss function. Among them, the algorithms that combining the robust loss function with the nonconvex regularization function have achieved impressive performance under non-Gaussian environments, such as the CIMMCC [19] using MCC and CIM, and CIMLMP [20] using least mean p-power (LMP) and CIM. Meanwhile, one particular interest is the ℓ_1-norm loss function as $f(\mathbf{Ax}-\mathbf{y})=\|\mathbf{Ax}-\mathbf{y}\|_1$, which is optimal when the impulsive noise is modeled as a Cauchy distribution [29].

In this paper, we consider the following $g(\cdot)$-regularized least-absolute (LA) sparse recovery problem

$$\min_{\mathbf{x}} F(\mathbf{x})\equiv f(\mathbf{Ax}-\mathbf{y})+g(\mathbf{x})=\|\mathbf{Ax}-\mathbf{y}\|_1+g(\mathbf{x}) \tag{3}$$

where $g(\mathbf{x})$ is the convex or generalized non-convex penalty for sparsity inducing, such as hard-thresholding, ℓ_p-norm penalty, ℓ_{1-2} penalty, LSP, SCAD, or MCP.

Generally, the problem (3) is difficult to solve, which is due to that the loss function f is non-smooth and the penalty function maybe nonconvex. To address the non-smooth ℓ_1-norm, many researchers using the alternating direction method of multipliers (ADMM) [24,26], in which the loss term and the penalty term are naturally separated. Using an auxiliary vector, the problem can be reformulated as

$$\min_{\mathbf{x},\mathbf{u}}\{\|\mathbf{u}\|_1+g(\mathbf{x})\}\quad\text{subject to}\quad\mathbf{Ax}-\mathbf{y}=\mathbf{u} \tag{4}$$

Some researchers transform the non-smooth ℓ_1-norm into a smooth counterpart and employ the ADMM [30] or difference of convex algorithm (DCA) [18] to solve the recovery problem. As will be shown later, there is a link between the auxiliary vector method and the proposed infimal convolution smoothing method.

1.2. Contributions

The main contributions of this work are summarized as follows.

First, we propose two smoothing strategies for the non-smooth loss function, one introduces a relaxation factor to approximate the ℓ_1-norm, the other uses the infimal convolution smoothing technique to transform the non-differentiable $\|\mathbf{Ax}-\mathbf{y}\|_1$ into a smooth counterpart. Both of them can obtain a continuously differentiable loss function, whose gradient is Lipschitz continuous. As it will be shown in the later section, this property is crucial for the convergence of the new algorithm.

Second, the convergence performance of the algorithm is discussed under the accelerated proximal gradient (APG) [31] and monotone APG (mAPG) [32] frameworks for the convex and non-convex regularize functions, respectively.

Finally, we discuss some properties of these smoothing strategies which can be easily extended to other sparse recovery problems. We also evaluate the effectiveness of the proposed algorithm via numerical experiments.

1.3. Outline and notation

The rest of this paper is structured as follows. In Section 2, we introduce two smoothing strategies. In Section 3, we employ the APG and mAPG frameworks for the minimization problem and provide some theorems to demonstrate the convergence of the proposed algorithm. In Section 4, we extend the smoothing strategies to other sparse recovery problems. Section 5 presents the numerical results. In the end, we provide our conclusion in Section 6.

Here, we define our notation. We define the ℓ_p-norm of the vector $\mathbf{x}\in\mathbb{R}^N$ as $\|\mathbf{x}\|_p=\left(\sum_n|x_n|^p\right)^{\frac{1}{p}}$. Especially, we define ℓ_1, ℓ_2 and ℓ_∞-norms of $\mathbf{x}$ as $\|\mathbf{x}\|_1=\sum_n|x_n|$, $\|\mathbf{x}\|_2=\left(\sum_n|x_n|^2\right)^{\frac{1}{2}}$ and $\|\mathbf{x}\|_\infty=\max_n|x_n|$, respectively. Given a matrix $\mathbf{A}\in\mathbb{R}^{M\times N}$, $\mathbf{A}_m$ is defined as the m-th column of $\mathbf{A}$, $\mathbf{A}^T$ is defined as the transpose of $\mathbf{A}$, $\|\mathbf{A}\|_2^2$ is defined as the maximum eigenvalue of $\mathbf{A}^T\mathbf{A}$, denoted by $\lambda_{\max}(\mathbf{A}^T\mathbf{A})$, and $[\mathbf{Ax}]_m$ is defined as the component m of $\mathbf{Ax}$. $\langle\cdot,\cdot\rangle$ denotes the inner product. $\mathbf{B}\preceq\mathbf{A}$ means that the matrix $\mathbf{A}-\mathbf{B}$ is positive semidefinite. $\mathbf{I}_N$ represents an $N\times N$ identity matrix, and $\mathrm{sign}(\cdot)$ represents the sign of a quantity with $\mathrm{sign}(0)\in[-1,1]$. The set of proper lower semicontinous convex functions from $\mathbb{R}^N$ to $\mathbb{R}\bigcup\{+\infty\}$ is defined as $\Gamma_0\left(\mathbb{R}^N\right)$.

2. Smoothing approximation methods

This section is devoted to construct two functions to smooth the least-absolute loss function. The first one introduces a relaxation factor and approximate the ℓ_1-norm as the sum of ℓ_p-norm. The second one uses the infimal convolution with the convex function $\frac{1}{2}\|\mathbf{B}(\cdot)\|_2^p$.

2.1. $\ell_{\boldsymbol{\varepsilon},p}$-norm smoothing approximation method

Definition 1. Let $\mathbf{x}\in\mathbb{R}^M$, the $\ell_{\boldsymbol{\varepsilon},p}$-norm $(p>1)$ function $\|\mathbf{x}\|_{\varepsilon,p}:\mathbb{R}^M\to\mathbb{R}$ is defined as $\|\mathbf{x}\|_{\varepsilon,p}:=\sum_m\left(|x_m|^p+\varepsilon_m^p\right)^{\frac{1}{p}}$, where $\varepsilon=\mathrm{diag}(\varepsilon_1,\varepsilon_2,\ldots,\varepsilon_M)$ with every $\varepsilon_m>0$.

From Definition 1, we can find that the $\ell_{\boldsymbol{\varepsilon},p}$ degenerates into the '$\ell_{1,\varepsilon}$' in [30] if we choose $p=2$ and $\varepsilon_1=\varepsilon_2=\cdots=\varepsilon_M$. Fig. 1 plot the ℓ_1, '$\ell_{1,\varepsilon}$' and $\ell_{\boldsymbol{\varepsilon},p}$ for comparison. From this, we can find that the $\ell_{\boldsymbol{\varepsilon},p}$-norm have more flexible strategies in smoothing approximation with different choices of $\boldsymbol{\varepsilon}$ and p.

Easily, we have that $\|\mathbf{x}\|_{\boldsymbol{\varepsilon},p}$ is smooth and differentiable. As $\max_m\varepsilon_m\to0$, $\|\mathbf{x}\|_{\boldsymbol{\varepsilon},p}\to\|\mathbf{x}\|_1$, $\|\mathbf{x}\|_{\boldsymbol{\varepsilon},p}$ can approximate $\|\mathbf{x}\|_1$ when each element in $\boldsymbol{\varepsilon}$ is sufficiently small. By using the $\ell_{\boldsymbol{\varepsilon},p}$-norm, we have a smoothing strategy for the non-smooth loss function $f(\mathbf{Ax}-\mathbf{y})=\|\mathbf{Ax}-\mathbf{y}\|_1$, defined as

$$f_{\varepsilon,p}(\mathbf{Ax}-\mathbf{y})=\|\mathbf{Ax}-\mathbf{y}\|_{\varepsilon,p}=\sum_m\left(|e_m|^p+\varepsilon_m^p\right)^{\frac{1}{p}} \tag{5}$$

where $e_m=[\mathbf{Ax}-\mathbf{y}]_m=(\sum_n a_{mn}x_n)-y_m$, a_{mn} is the $(m,\ n)$-th element of matrix $\mathbf{A}$. From (5), we have

$$0\le f_{\varepsilon,p}(\mathbf{Ax}-\mathbf{y})-f(\mathbf{Ax}-\mathbf{y})\le\sum\nolimits_m\varepsilon_m\le M\varepsilon_{\max} \tag{6}$$

where $\varepsilon_{\max}=\max_m\varepsilon_m$. The gradient of $f_{\varepsilon,p}(\mathbf{Ax}-\mathbf{y})$ is given by

$$\nabla f_{\varepsilon,p}(\mathbf{Ax}-\mathbf{y})=\mathbf{A}^T\mathrm{T} \tag{7}$$

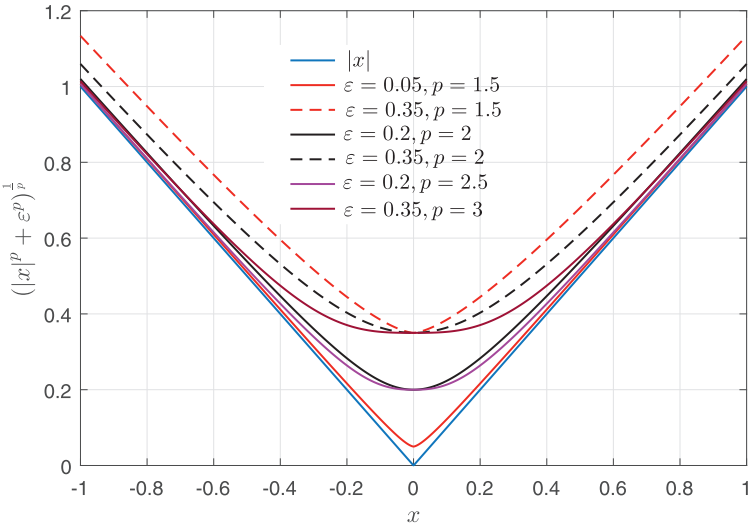

Fig. 1. The scalar $\ell_{\varepsilon,p}$ with different ε and p.

where $\mathrm{T} = [t_1, \ldots, t_M]^T \in \mathbb{R}^M$ with $t_m = \frac{|e_m|^{p-1}\mathrm{sign}(e_m)}{\left(|e_m|^p+\varepsilon_m^p\right)^{\frac{p-1}{p}}}$. And the Hessian Matrix of $f_{\varepsilon,p}(\mathbf{Ax}-\mathbf{y})$ is

$$\nabla^2 f_{\varepsilon,p}(\mathbf{Ax}-\mathbf{y})_{ij} = \frac{\partial^2 f_{\varepsilon,p}(\mathbf{Ax}-\mathbf{y})}{\partial x_i \partial x_j} = \sum_m a_{mi}a_{mj}\frac{(p-1)|e_m|^{p-2}\varepsilon_m^p}{\left(|e_m|^p+\varepsilon_m^p\right)^{2-\frac{1}{p}}},$$
$$i = 1, \ldots, N \quad j = 1, \ldots, N \tag{8}$$

By substituting $\frac{|e_m|^{p-2}\varepsilon_m^p}{\left(|e_m|^p+\varepsilon_m^p\right)^{2-\frac{1}{p}}} \le \min\left\{\frac{\varepsilon_m^p}{|e_m|^{p+1}}, \frac{|e_m|^{p-2}}{\varepsilon_m^{p-1}}\right\}$ into (8), we have

$$\nabla^2 f_{\varepsilon,p}(\mathbf{Ax}-\mathbf{y})_{ij} \le \sum_m a_{mi}a_{mj}\frac{(p-1)}{\varepsilon_m} \tag{9}$$

Then we have

$$\nabla^2 f_{\varepsilon,p}(\mathbf{Ax}-\mathbf{y}) \preceq (p-1)\mathbf{A}^T\varepsilon^{-1}\mathbf{A} \tag{10}$$

where $\varepsilon^{-1} = \mathrm{diag}(1/\varepsilon_1, 1/\varepsilon_2, \ldots, 1/\varepsilon_M)$. From this we can find that the gradient of smoothing function $f_{\varepsilon,p}(\mathbf{Ax}-\mathbf{y})$ is Lipschitz continuous, which is crucial for the solution of minimization problem (3) as will be shown in Section 3.

2.2. The infimal convolution smoothing method

In this subsection, we first recall the definition of infimal convolution. For two functions h and φ from $\mathbb{R}^M$ to $\mathbb{R}\bigcup\{+\infty\}$, the infimal convolution [33] is given by

$$(h\square\varphi)(\mathbf{x}) = \inf_{\mathbf{u}\in\mathbb{R}^M}\{h(\mathbf{u}) + \varphi(\mathbf{x}-\mathbf{u})\} \tag{11}$$

In the notation of infimal convolution, the Moreau envelope [34] with a scale parameter $\beta > 0$ of function h is defined as

$$h_\beta^{\mathrm{M}}(\mathbf{x}) = h(\mathbf{x})\square\frac{1}{2\beta}\|\mathbf{x}\|_2^2 = \inf_{\mathbf{u}\in\mathbb{R}^M}\left\{h(\mathbf{u}) + \frac{1}{2\beta}\|\mathbf{u}-\mathbf{x}\|_2^2\right\} \tag{12}$$

Then, we introduce the second smoothing method by using the infimal convolution with the convex function $\frac{1}{2}\|\mathbf{B}(\cdot)\|_2^p$.

Definition 2. Let $\mathbf{x} \in \mathbb{R}^M$, $\mathbf{B} \in \mathbb{R}^{M\times M}$. We define the infimal convolution smoothing function $h_{\mathbf{B},p} : \mathbb{R}^M \to \mathbb{R}$ $(p > 1)$ as

$$h_{\mathbf{B},p}(\mathbf{x}) := \inf_{\mathbf{u}\in\mathbb{R}^M}\left\{h(\mathbf{u}) + \frac{1}{2}\|\mathbf{B}(\mathbf{u}-\mathbf{x})\|_2^p\right\} \tag{13}$$

From Definition 2, we can find that the $h_{\mathbf{B},p}$ degenerates into Moreau envelope if we choose $p = 2$ and set $\mathbf{B}^T\mathbf{B}$ to be the scale identity matrix $\frac{1}{\beta}\mathbf{I}_M$.

Figs. 2 and 3 show the curves of $h_{\mathbf{B},p}$ with different scale matrices $\mathbf{B}$ and p, where $\mathbf{B} = \mathbf{I}_M$ and $p = 2$ correspond to the well-known Huber function. Intuitively, we can find that with a bigger p, the $h_{\mathbf{B},p}$ is smoother, and $h_{\mathbf{B},p}$ is closer to h with larger matrix $\mathbf{B}$.

Proposition 1. *Let $h \in \Gamma_0(\mathbb{R}^M)$ and be coercive, and $\mathbf{x} \in \mathbb{R}^N$, $\mathbf{y} \in \mathbb{R}^M$, $\mathbf{A} \in \mathbb{R}^{M\times N}$, $\mathbf{B} \in \mathbb{R}^{M\times M}$, then $h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$ $(p > 1)$ is a proper lower semicontinuous convex function, and the infimal convolution is exact, i.e.,*

$$h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y}) = \min_{\mathbf{u}\in\mathbb{R}^M}\left\{h(\mathbf{u}) + \frac{1}{2}\|\mathbf{B}(\mathbf{u}-\mathbf{Ax}+\mathbf{y})\|_2^p\right\} \tag{14}$$

Proof. Set $\varphi(\mathbf{x}) = \frac{1}{2}\|\mathbf{B}\cdot\|_2^p$ $(p > 1)$ in (11), then we have $h_{\mathbf{B},p}(\cdot) = (h\square\varphi)(\cdot)$. Since $h, \varphi \in \Gamma_0(\mathbb{R}^M)$ and h is coercive, and φ is bounded below, then we can obtain that $h_{\mathbf{B},p} \in \Gamma_0(\mathbb{R}^M)$ and it is exact at every point of its domain by Proposition 12.14 in [35]. By using the preserving convexity property of affine mapping, we have $h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y}) \in \Gamma_0(\mathbb{R}^M)$. □

Next, we will show that $h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$ can approximate $h(\mathbf{Ax}-\mathbf{y})$ at any given precision when we choose proper scale matrix $\mathbf{B}$. □

Proposition 2. *Let $h \in \Gamma_0(\mathbb{R}^M)$ and be coercive, and the scale matrix $\mathbf{B}$ satisfies $\mathbf{B}^T\mathbf{B} \succeq \beta^2\mathbf{I}_M$, suppose that the subgradients of h over $\mathbb{R}^M$ are bounded by L_h, $\|h'(\mathbf{z})\|_2 \le L_h$ for any $\mathbf{z} \in \mathbb{R}^M$ and $h'(\mathbf{z}) \in \partial h(\mathbf{z})$. Then, it follows that*

$$h(\mathbf{Ax}-\mathbf{y}) - \delta_{\mathbf{B},p}(L_h)^{\frac{p}{p-1}} \le h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y}) \le h(\mathbf{Ax}-\mathbf{y}) \tag{15}$$

where $p > 1$ and $\delta_{\mathbf{B},p} = \frac{p-1}{2}\left(\frac{2}{p\beta}\right)^{\frac{p}{p-1}}$.

See Appendix A for the Proof of Proposition 2.

If we choose $p = 2$, then we have

$$h(\mathbf{Ax}-\mathbf{y}) - \frac{1}{2\beta}L_h^2 \le h_{\mathbf{B},2}(\mathbf{Ax}-\mathbf{y}) \le h(\mathbf{Ax}-\mathbf{y}) \tag{16}$$

Next, we focus on the gradient of $h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$ under the condition of $h(\mathbf{Ax}-\mathbf{y}) = \|\mathbf{Ax}-\mathbf{y}\|_1$ and $\mathbf{B}^T\mathbf{B} = \mathrm{diag}(b_1^2, b_2^2, \ldots, b_M^2)$ is diagonal. If $h(\mathbf{Ax}-\mathbf{y}) = \|\mathbf{Ax}-\mathbf{y}\|_1$, we define a function

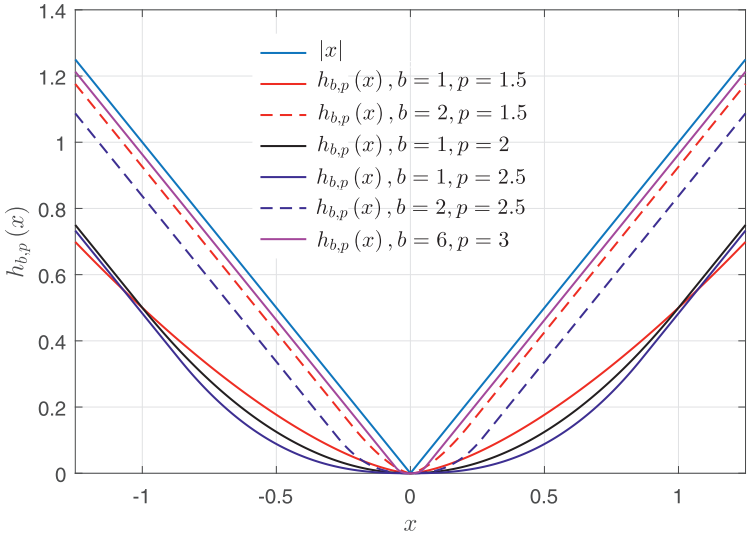

Fig. 2. The scalar $h_{\mathbf{B},p}$ with different b and p ($h(\mathbf{x}) = \|\mathbf{x}\|_1$).

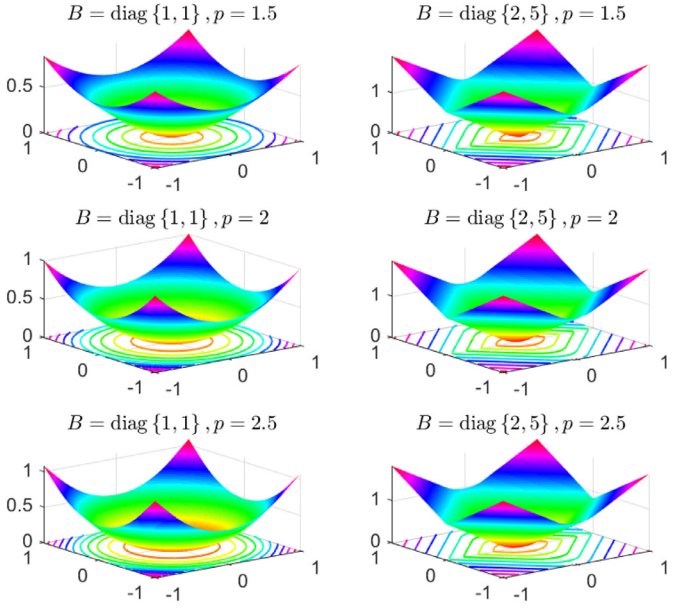

Fig. 3. The level curves of $h_{\mathbf{B},p}$ with different $\mathbf{B}$ and p ($h(\mathbf{x}) = \|\mathbf{x}\|_1$).

$g : \mathbb{R}^M \to \mathbb{R}$ as

$$g(\mathbf{u}) = \|\mathbf{u}\|_1 + \frac{1}{2}\|\mathbf{B}(\mathbf{u} - \mathbf{A}\mathbf{x} + \mathbf{y})\|_2^p \tag{17}$$

Then from (14), we have that $h_{\mathbf{B},p}(\mathbf{A}\mathbf{x} - \mathbf{y})$ is the minimization of $g(\mathbf{u})$, and we suppose that $h_{\mathbf{B},p}(\mathbf{A}\mathbf{x} - \mathbf{y}) = g(\tilde{\mathbf{u}})$. Since $g(\tilde{\mathbf{u}})$ is convex, we have that $\tilde{\mathbf{u}}$ minimizes $g(\mathbf{u})$ if and only if $\mathbf{0} \in g(\tilde{\mathbf{u}})$, where the subdifferential of $g(\tilde{\mathbf{u}})$ is given by

$$\mathbf{0} \in g(\tilde{\mathbf{u}}) = \partial\|\tilde{\mathbf{u}}\|_1 + \frac{p}{2}d^{\frac{p}{2}-1}\mathbf{B}^T\mathbf{B}(\tilde{\mathbf{u}} - \mathbf{A}\mathbf{x} + \mathbf{y}) \tag{18}$$

where $d = \|\mathbf{B}(\tilde{\mathbf{u}} - \mathbf{A}\mathbf{x} + \mathbf{y})\|_2^2$ and $\partial\|\tilde{\mathbf{u}}\|_1 = \mathrm{SIGN}(\tilde{\mathbf{u}}) = \prod_m \mathrm{sign}(\tilde{u}_m) \subset \mathbb{R}^M$ with

$$\mathrm{sign}(\tilde{u}_m) := \begin{cases} \{1\}, & \tilde{u}_m > 0 \\ [-1,1], & \tilde{u}_m = 0 \\ \{-1\}, & \tilde{u}_m < 0 \end{cases} \tag{19}$$

From the definition of $h_{\mathbf{B},p}(\mathbf{A}\mathbf{x} - \mathbf{y})$ in (14), we have

$$\begin{aligned} \partial h_{\mathbf{B},p}(\mathbf{A}\mathbf{x} - \mathbf{y}) &= \partial\tilde{\mathbf{u}}^T(\partial\|\tilde{\mathbf{u}}\|_1) + \frac{p}{2}d^{\frac{p}{2}-1}\left(\mathbf{A}^T - \partial\tilde{\mathbf{u}}^T\right)\mathbf{B}^T\mathbf{B}(\mathbf{A}\mathbf{x} - \mathbf{y} - \tilde{\mathbf{u}}) \\ &= \partial\tilde{\mathbf{u}}^T\left(\partial\|\tilde{\mathbf{u}}\|_1 + \frac{p}{2}d^{\frac{p}{2}-1}\mathbf{B}^T\mathbf{B}(\tilde{\mathbf{u}} - \mathbf{A}\mathbf{x} + \mathbf{y})\right) \\ &\quad + \frac{p}{2}d^{\frac{p}{2}-1}\mathbf{A}^T\mathbf{B}^T\mathbf{B}(\mathbf{A}\mathbf{x} - \mathbf{y} - \tilde{\mathbf{u}}) \end{aligned} \tag{20}$$

We define the support of $\tilde{\mathbf{u}}$ as Γ_*=supp{$\tilde{\mathbf{u}}$}, which is the index set labeling the non-zero elements in $\tilde{\mathbf{u}}$, and define the corresponding zero elements in $\tilde{\mathbf{u}}$ as Γ_0, then we have

$$\begin{aligned} &\partial\tilde{\mathbf{u}}^T\left(\partial\|\tilde{\mathbf{u}}\|_1 + \frac{p}{2}d^{\frac{p}{2}-1}\mathbf{B}^T\mathbf{B}(\tilde{\mathbf{u}} - \mathbf{A}\mathbf{x} + \mathbf{y})\right) \\ &= \left(\partial\tilde{\mathbf{u}}^T\right)_{\Gamma_*}\left(\partial\|\tilde{\mathbf{u}}\|_1 + \frac{p}{2}d^{\frac{p}{2}-1}\mathbf{B}^T\mathbf{B}(\tilde{\mathbf{u}} - \mathbf{A}\mathbf{x} + \mathbf{y})\right)_{\Gamma_*} \\ &\quad + \left(\partial\tilde{\mathbf{u}}^T\right)_{\Gamma_0}\left(\partial\|\tilde{\mathbf{u}}\|_1 + \frac{p}{2}d^{\frac{p}{2}-1}\mathbf{B}^T\mathbf{B}(\tilde{\mathbf{u}} - \mathbf{A}\mathbf{x} + \mathbf{y})\right)_{\Gamma_0} \end{aligned} \tag{21}$$

For $\forall m \in \Gamma_*$, from (18), we have

$$\partial\|\tilde{\mathbf{u}}\|_1 + \frac{p}{2}d^{\frac{p}{2}-1}\mathbf{B}^T\mathbf{B}(\tilde{\mathbf{u}} - \mathbf{A}\mathbf{x} + \mathbf{y}) = \mathbf{0} \tag{22}$$

For $\forall m \in \Gamma_0$, we have

$$\left(\partial\tilde{\mathbf{u}}^T\right)_m = \mathbf{0} \tag{23}$$

Substitute (23) and (22) into (21), we can obtain

$$\partial h_{\mathbf{B}}(\mathbf{A}\mathbf{x} - \mathbf{y}) = \frac{p}{2}d^{\frac{p}{2}-1}\mathbf{A}^T\mathbf{B}^T\mathbf{B}(\mathbf{A}\mathbf{x} - \mathbf{y} - \tilde{\mathbf{u}}) \tag{24}$$

By using (18), we have that

$$\frac{p}{2}d^{\frac{p}{2}-1}\left\|\mathbf{B}^T\mathbf{B}(\mathbf{A}\mathbf{x} - \mathbf{y} - \tilde{\mathbf{u}})\right\|_\infty \le 1 \tag{25}$$

Substitute this into (24), it follows that $\left|\left[\partial h_{\mathbf{B},p}(\mathbf{A}\mathbf{x} - \mathbf{y})\right]_n\right| \le \sum_{m=1}^M |a_{mn}|$.

If $\mathbf{B}^T\mathbf{B} = \mathrm{diag}\left(b_1^2, b_2^2, \ldots, b_M^2\right)$, by substituting it into (18) and considering the convex separable properties of $g(\mathbf{u})$, we have

$$\tilde{\mathbf{u}} = \mathrm{shrink}\left(\mathbf{A}\mathbf{x} - \mathbf{y}, \frac{\mathbf{2}}{\mathbf{p}}\left(\mathbf{B^TB}\right)^{-1}(\mathbf{d})^{\mathbf{1}-\frac{\mathbf{p}}{\mathbf{2}}}\right) \tag{26}$$

Then we can use an iterative framework for the solution of $\tilde{\mathbf{u}}$:

$$\begin{cases} \tilde{\mathbf{u}}^k = \mathrm{shrink}\left(\mathbf{A}\mathbf{x} - \mathbf{y}, \frac{2}{p}\left(\mathbf{B}^T\mathbf{B}\right)^{-1}\left(d^k\right)^{1-\frac{p}{2}}\right) \\ d^{k+1} = \left\|\mathbf{B}\left(\tilde{\mathbf{u}}^k - \mathbf{A}\mathbf{x} + \mathbf{y}\right)\right\|_2^2 \end{cases} \tag{27}$$

where shrink($\mathbf{x}$, $\boldsymbol{\lambda}$) denotes the soft shrinkage operator given by

$$[\text{shrink}(\mathbf{x},\lambda)]_m = \text{sign}(x_m)\max\{|x_m|-\lambda_m,0\} \tag{28}$$

Then, by using (24), we have

$$\partial h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y}) = \mathbf{A}^T\mathbf{v} \tag{29}$$

where $\mathbf{v}=[v_1,v_2,\ldots,v_M]$ with

$$v_m=\begin{cases}1, & [\mathbf{Ax}-\mathbf{y}]_m > \dfrac{2}{pb_m^2}d^{1-\frac{p}{2}}\\ \dfrac{pb_m^2}{2}d^{\frac{p}{2}-1}[\mathbf{Ax}-\mathbf{y}]_m, & |[\mathbf{Ax}-\mathbf{y}]_m| \le \dfrac{2}{pb_m^2}d^{1-\frac{p}{2}}\\ -1, & [\mathbf{Ax}-\mathbf{y}]_m < -\dfrac{2}{pb_m^2}d^{1-\frac{p}{2}}\end{cases} \tag{30}$$

Then, we can calculate the gradient of $h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$ with an approximate d by using the iterative framework of (27) after proper iterations.

Remark 1. If we choose $p=2$, then the gradient of $h_{\mathbf{B},2}(\mathbf{Ax}-\mathbf{y})$ is equal to $\partial h_{\mathbf{B},2}(\mathbf{Ax}-\mathbf{y})=\mathbf{A}^T\mathbf{v}$ with

$$v_m=\begin{cases}1, & [\mathbf{Ax}-\mathbf{y}]_m > 1/b_m^2\\ b_m^2[\mathbf{Ax}-\mathbf{y}]_m, & |[\mathbf{Ax}-\mathbf{y}]_m| \le 1/b_m^2\\ -1, & [\mathbf{Ax}-\mathbf{y}]_m < -1/b_m^2\end{cases} \tag{31}$$

which does not need to calculate (27) iteratively.

Proposition 3. *Let* $h(\mathbf{Ax}-\mathbf{y})=\|\mathbf{Ax}-\mathbf{y}\|_1$ *and* $\mathbf{B}^T\mathbf{B}=\text{diag}\left(b_1^2,b_2^2,\ldots,b_M^2\right)$, *the Hessian Matrix of* $h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$ *satisfies*

$$\nabla^2 h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y}) \preceq \begin{cases}\dfrac{p}{2}d^{\frac{p}{2}-1}\mathbf{A}^T\mathbf{B}^T\mathbf{BA}, & 1<p<2\\ \dfrac{p(p-1)}{2}d^{\frac{p}{2}-1}\mathbf{A}^T\mathbf{B}^T\mathbf{BA}, & p\ge 2\end{cases} \tag{32}$$

And if $p\ge 2$, *the gradient* $\nabla h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$ *is* $\rho\|\mathbf{BA}\|_2^2$*-Lipschitz continuous with* $\rho=(p-1)\left(\frac{p}{2}\right)^{\frac{1}{p-1}}\left(\frac{\sqrt{M}}{b_{\min}}\right)^{\frac{p-2}{p-1}}$, *where* $b_{\min}=\min_m\{b_m\}$.

See Appendix B for the Proof of Proposition 3.

Remark 2. If we choose $p=2$, we have that $\nabla h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$ is $\|\mathbf{BA}\|_2^2$-Lipschitz continuous, which can be obtained by using Proposition 3 directly. Meanwhile, this conclusion can be extended to any h that $h\in\Gamma_0\left(\mathbb{R}^M\right)$ and be coercive with normal matrix $\mathbf{B}\in\mathbb{R}^{M\times M}$. See Appendix C for the Proof.

Propositions 1 and 3 help the smoothed loss function $h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$ $(p\ge 2)$ to fit the requirement of the APG and mAPG frameworks, which is quite important for the proposed algorithm as will be shown in the next section.

3. Algorithm for the smoothing based recovery problem

3.1. APG and mAPG frameworks for the convex and nonconvex problems

Rewrite the $g(\cdot)$-regularized least-absolute recovery problem as

$$\min_{\mathbf{x}} F(\mathbf{x}) \equiv f(\mathbf{Ax}-\mathbf{y})+g(\mathbf{x}) = \|\mathbf{Ax}-\mathbf{y}\|_1+g(\mathbf{x}) \tag{33}$$

Firstly, we use the above two smoothing strategies to transform the non-differentiable and non-separable $\|\mathbf{Ax}-\mathbf{y}\|_1$ into a smooth counterpart with Lipschitz continuous gradient and choose the proximal gradient (PG) framework to solve the minimization.

1) The first approach uses the $\ell_{\boldsymbol{\varepsilon},p}$-norm smoothing approximation method to transform the problem (33), and solves the following smoothed problem

$$\min_{\mathbf{x}} F_{\varepsilon,p}(\mathbf{x}) \equiv f_{\varepsilon,p}(\mathbf{Ax}-\mathbf{y})+g(\mathbf{x}) \tag{34}$$

Given a symmetric positive semidefinite matrix $\mathbf{H}\in\mathbb{R}^{N\times N}$, we define the quadratic approximation of $F_{\boldsymbol{\varepsilon},p}(\mathbf{x})$ at a given point $\mathbf{z}$ as

$$Q_{\mathbf{H}}(\mathbf{x},\mathbf{z}) = g(\mathbf{x})+f_{\varepsilon,p}(\mathbf{Az}-\mathbf{y})+\langle\nabla f_{\varepsilon,p}(\mathbf{Az}-\mathbf{y}),\mathbf{x}-\mathbf{z}\rangle + \frac{1}{2}\langle\mathbf{x}-\mathbf{z},\mathbf{H}(\mathbf{x}-\mathbf{z})\rangle \tag{35}$$

For any $\mathbf{x},\mathbf{z}\in\mathbb{R}^N$, if we choose the symmetric positive semidefinite matrix $\mathbf{H}$ that satisfies

$$f_{\varepsilon,p}(\mathbf{Ax}-\mathbf{y})+g(\mathbf{x}) \le Q_{\mathbf{H}}(\mathbf{x},\mathbf{z}) \tag{36}$$

then, for any $\mathbf{x}^0\in\mathbb{R}^N$, the k-th iteration of proximal gradient (PG) [31] for solving (34) is

$$\mathbf{x}^{k+1}=\arg\min_{\mathbf{x}} Q_{\mathbf{H}}\left(\mathbf{x},\mathbf{x}^k\right) \tag{37}$$

Specially, we set $\mathbf{H}=L_\varepsilon\mathbf{I}_N$ with $L_\varepsilon\ge(p-1)\|\mathbf{A}\|_2^2/\varepsilon_{\min}$, or $\mathbf{H}=(p-1)\mathbf{A}^T\varepsilon^{-1}\mathbf{A}$. Then, from (10), easily we have that condition of (36) holds for any $\mathbf{x},\mathbf{y}\in\mathbb{R}^N$.

When $\mathbf{H}=L_\varepsilon\mathbf{I}_N$ with $L_\varepsilon\ge(p-1)\|\mathbf{A}\|_2^2/\varepsilon_{\min}$, we have

$$\begin{aligned}\mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}} Q_{\mathbf{H}}\left(\mathbf{x},\mathbf{x}^k\right)\\ &= \arg\min_{\mathbf{x}}\left\{g(\mathbf{x})+f_{\varepsilon,p}\left(\mathbf{Ax}^k-\mathbf{y}\right)+\left\langle\nabla f_{\varepsilon,p}\left(\mathbf{Ax}^k-\mathbf{y}\right),\mathbf{x}-\mathbf{x}^k\right\rangle\right.\\ &\quad\left.+\frac{L_\varepsilon}{2}\left\langle\mathbf{x}-\mathbf{x}^k,\mathbf{x}-\mathbf{x}^k\right\rangle\right\}\\ &= \arg\min_{\mathbf{x}}\left\{g(\mathbf{x})+\frac{L_\varepsilon}{2}\left\|\mathbf{x}-\mathbf{x}^k+\frac{1}{L_\varepsilon}\mathbf{A}^T\mathrm{T}^k\right\|_2^2\right\}\end{aligned} \tag{38}$$

The last equation comes from the gradient calculation of $f_{\varepsilon,p}(\mathbf{Ax}-\mathbf{y})$ in (7).

Similarly, when $\mathbf{H}=(p-1)\mathbf{A}^T\varepsilon^{-1}\mathbf{A}$, we have

$$\begin{aligned}\mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}}\left\{g(\mathbf{x})+f_{\varepsilon,p}\left(\mathbf{Ax}^k-\mathbf{y}\right)+\left\langle\nabla f_{\varepsilon,p}\left(\mathbf{Ax}^k-\mathbf{y}\right),\mathbf{x}-\mathbf{x}^k\right\rangle\right.\\ &\quad\left.+\frac{p-1}{2}\left\langle\mathbf{x}-\mathbf{x}^k,\mathbf{A}^T\varepsilon^{-1}\mathbf{A}\left(\mathbf{x}-\mathbf{x}^k\right)\right\rangle\right\}\\ &= \arg\min_{\mathbf{x}}\left\{g(\mathbf{x})+\frac{p-1}{2}\left\|\varepsilon^{-1/2}\mathbf{A}\left(\mathbf{x}-\mathbf{x}^k\right)+\frac{1}{p-1}\varepsilon^{1/2}\mathrm{T}^k\right\|_2^2\right\}\end{aligned} \tag{39}$$

where $\varepsilon^{-1/2}=\text{diag}(1/\sqrt{\varepsilon_1},1/\sqrt{\varepsilon_2},\ldots,1/\sqrt{\varepsilon_M})$.

2) The second smoothing strategy is replacing the non-smooth function $f(\mathbf{Ax}-\mathbf{y})$ by its infimal convolution $f_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$, then the recovery problem becomes

$$\min_{\mathbf{x}} F_{\mathbf{B},p}(\mathbf{x}) \equiv f_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})+g(\mathbf{x}) \tag{40}$$

However, in order to calculate the gradient of $f_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$, we only consider a common simple case that $\mathbf{B}^T\mathbf{B}=\text{diag}\left(b_1^2,b_2^2,\ldots,b_M^2\right)$ is diagonal with $b_m>0$ in this paper. From the definition of the $f_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$, one can find that the infimal convolution smoothing function can also be thought as an extension of the generalized Huber function in [17] but with different p. However, in this paper, we mainly consider this generalized Huber function as an infimal convolution smoothing approximation of the ℓ_1-norm loss function, and discuss the relationship between them from the aspect of smoothing approximation.

Similar as the first $\ell_{\boldsymbol{\varepsilon},p}$-norm smoothing strategy, we also define the quadratic approximation of $F_{\mathbf{B},p}(\mathbf{x})$ at a given point $\mathbf{z}$ as:

$$P_{\mathbf{H}}(\mathbf{x},\mathbf{z}) = g(\mathbf{x})+f_{\mathbf{B},p}(\mathbf{Az}-\mathbf{y})+\left\langle\nabla f_{\mathbf{B},p}(\mathbf{Az}-\mathbf{y}),\mathbf{x}-\mathbf{z}\right\rangle + \frac{1}{2}\left\langle\mathbf{x}-\mathbf{z},\mathbf{H}(\mathbf{x}-\mathbf{z})\right\rangle \tag{41}$$

Here we set $\mathbf{H}=L_{\mathbf{B}}\mathbf{I}_N$ with $L_{\mathbf{B}}\ge\rho\|\mathbf{BA}\|_2^2$ or $\mathbf{H}=\rho\mathbf{A}^T\mathbf{B}^T\mathbf{BA}$ with $\rho=(p-1)\left(\frac{p}{2}\right)^{\frac{1}{p-1}}\left(\frac{\sqrt{M}}{b_{\min}}\right)^{\frac{p-2}{p-1}}$ to satisfy the condition $f_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})+g(\mathbf{x})\le P_{\mathbf{H}}(\mathbf{x},\mathbf{z})$ for any $\mathbf{x},\mathbf{z}\in\mathbb{R}^N$ by using Proposition 3.

Table 1
Smoothing based APG and mAPG methods for convex and non-convex regularized LA loss function.

Algorithm 1. Smoothing based APG and mAPG methods

Initialization:
Given **A, y**. Select **H**, ε, β, N_{iter} and ξ_0.
Initialize $\mathbf{x}^1=\mathbf{x}^0=\mathbf{0}$, $t^1=t^0=1$.
Main iteration loop:
for $k=1,2,\ldots,N_{iter}$ do
APG: **x-updating:** $\mathbf{x}^k=\mathbf{x}^k+\frac{t^{k-1}-1}{t^k}\left(\mathbf{x}^k-\mathbf{x}^{k-1}\right)$
$\ell_{\boldsymbol{\varepsilon},p}$-norm smoothing: $\mathbf{x}^{k+1}=\arg\min_{\mathbf{x}} Q_{\mathbf{H}}\left(\mathbf{x},\mathbf{x}^k\right)$ with (38) or (39)
infimal convolution smoothing: $\mathbf{x}^{k+1}=\arg\min_{\mathbf{x}} P_{\mathbf{H}}\left(\mathbf{x},\mathbf{x}^k\right)$ with (42) or (43)
mAPG: **z -updating:** $\mathbf{z}^k=\mathbf{x}^k+\frac{t^{k-1}-1}{t^k}\left(\mathbf{u}^k-\mathbf{x}^k\right)+\frac{t^{k-1}-1}{t^k}\left(\mathbf{x}^k-\mathbf{x}^{k-1}\right)$
u, v -updating: $\ell_{\boldsymbol{\varepsilon},p}$-norm smoothing: $\mathbf{u}^{k+1}=\arg\min_{\mathbf{x}} Q_{\mathbf{H}}\left(\mathbf{x},\mathbf{z}^k\right)$ and $\mathbf{v}^{k+1}=\arg\min_{\mathbf{x}} Q_{\mathbf{H}}\left(\mathbf{x},\mathbf{x}^k\right)$ with (38) or (39)
infimal convolution smoothing: $\mathbf{u}^{k+1}=\arg\min_{\mathbf{x}} P_{\mathbf{H}}\left(\mathbf{x},\mathbf{z}^k\right)$ and $\mathbf{v}^{k+1}=\arg\min_{\mathbf{x}} P_{\mathbf{H}}\left(\mathbf{x},\mathbf{x}^k\right)$ with (42) or (43)
x -updating: $\ell_{\boldsymbol{\varepsilon},p}$-norm smoothing: $\mathbf{x}^{k+1}=\begin{cases}\mathbf{u}^{k+1}, & \text{if } F_{\varepsilon,p}\left(\mathbf{u}^{k+1}\right)\le F_{\varepsilon,p}\left(\mathbf{v}^{k+1}\right)\\ \mathbf{v}^{k+1}, & \text{otherwise}\end{cases}$
infimal convolution smoothing: $\mathbf{x}^{k+1}=\begin{cases}\mathbf{u}^{k+1}, & \text{if } F_{\mathbf{B},p}\left(\mathbf{u}^{k+1}\right)\le F_{\mathbf{B},p}\left(\mathbf{v}^{k+1}\right)\\ \mathbf{v}^{k+1}, & \text{otherwise}\end{cases}$
t**-computation:** $t^{k+1}=\left(1+\sqrt{4\left(t^k\right)^2+1}\right)/2$
Exit criterion: $\xi^{k+1}=\left\|\mathbf{x}^{k+1}-\mathbf{x}^k\right\|_2/\left\|\mathbf{x}^k\right\|_2$
if $\xi^{k+1}<\xi_0$ then
exit
end if
end for

Similar as (38), when $\mathbf{H}=L_{\mathbf{B}}\mathbf{I}_N$ with $L_{\mathbf{B}}\ge\rho\|\mathbf{BA}\|_2^2$, we have

$$\begin{aligned}\mathbf{x}^{k+1}&=\arg\min_{\mathbf{x}} P_{\mathbf{H}}\left(\mathbf{x},\mathbf{x}^k\right)\\&=\arg\min_{\mathbf{x}}\left\{g(\mathbf{x})+\frac{L_{\mathbf{B}}}{2}\left\|\mathbf{x}-\mathbf{x}^k+\frac{1}{L_{\mathbf{B}}}\nabla f_{\mathbf{B},p}\left(\mathbf{Ax}^k-\mathbf{y}\right)\right\|_2^2\right\}\end{aligned}\tag{42}$$

By using (27), (29) and (30), the $\nabla f_{\mathbf{B},p}\left(\mathbf{Ax}^k-\mathbf{y}\right)$ can be calculated with an approximate d when we choose $p>2$, or by using Eq. (31) directly when we choose $p=2$.

When $\mathbf{H}=\rho\mathbf{A}^T\mathbf{B}^T\mathbf{BA}$, we have

$$\begin{aligned}\mathbf{x}^{k+1}&=\arg\min_{\mathbf{x}} P_{\mathbf{H}}\left(\mathbf{x},\mathbf{x}^k\right)\\&=\arg\min_{\mathbf{x}}\left\{g(\mathbf{x})+f_{\mathbf{B},p}\left(\mathbf{Ax}^k-\mathbf{y}\right)+\left\langle\nabla f_{\mathbf{B},p}\left(\mathbf{Ax}^k-\mathbf{y}\right),\mathbf{x}-\mathbf{x}^k\right\rangle\right.\\&\quad\left.+\frac{\rho}{2}\left\langle\mathbf{x}-\mathbf{x}^k,\mathbf{A}^T\mathbf{B}^T\mathbf{BA}\left(\mathbf{x}-\mathbf{x}^k\right)\right\rangle\right\}\\&=\arg\min_{\mathbf{x}}\left\{g(\mathbf{x})+\frac{\rho}{2}\left\|\mathbf{BA}\left(\mathbf{x}-\mathbf{x}^k\right)+\rho^{-1}\mathbf{B}^{-1}\mathbf{v}^k\right\|_2^2\right\}\end{aligned}\tag{43}$$

where $\mathbf{B}^{-1}=\mathrm{diag}(1/b_1,1/b_2,\ldots,1/b_M)$ and $\mathbf{v}^k$ can be calculated by (30) or (31).

Secondly, we apply an acceleration method for the PG framework. The main disadvantage of the PG method is that its convergence rate $\mathcal{O}(1/k)$ is relatively slow, and it requires $g(\mathbf{x})$ being convex. To overcome these, the accelerate versions: accelerate proximal gradient (APG) for convex problem [33] and monotone APG for nonconvex problem [32] are proposed, respectively. Both of them have convergence rates $\mathcal{O}\left(1/k^2\right)$ for convex programs.

By using the APG and mAPG framework, which uses a proximal gradient step as the monitor, we summarize our algorithm in Table 1, where N_{iter} is the max number of reconstruction iterations, ξ^{k+1} is the ℓ_2-distance between two recoveries $\mathbf{x}^{k+1}$ and $\mathbf{x}^k$. The exit criterion $\xi^{k+1}<\xi_0$ means that there is no longer any appreciate changes in the iteration and the algorithm runs into convergence.

3.2. Some supplements of the proposed algorithm

First, we focus on fast solutions for the non-convex regularizations. From Table 1, we can find that the main computation of mAPG is concentrating on the non-convex minimization for **u** and **v**-updating. However, for some frequently-used convex and non-convex regularizes (penalties), problems (38) and (42) have closed-form or fast solutions. In this subsection, we provide some fast solutions for the non-convex problem

$$\mathbf{x}^{k+1}=\mathrm{prox}_{Lg}(\mathbf{d})=\arg\min_{\mathbf{x}}\left\{g(\mathbf{x})+\tfrac{1}{2L}\|\mathbf{x}-\mathbf{d}\|_2^2\right\}\tag{44}$$

where $\mathrm{prox}_{Lg}(\mathbf{d})$ is the proximal operator. Table 2 lists some examples of closed-form expressions of the proximal operators of various functions. In Ref. [38], Table 10.2 lists some other functions in $\Gamma_0\left(\mathbb{R}^N\right)$ with closed-form expressions.

Here, we consider other two special penalty functions.

(i) Generalized MCP, $g(\mathbf{x})=\lambda(\|\mathbf{x}\|_1-S_{\mathbf{Z}}(\mathbf{x}))$, where $S_{\mathbf{Z}}(\mathbf{x})$ is the generalized Huber function that defined as $S_{\mathbf{Z}}(\mathbf{x})=\inf_{\mathbf{v}\in\mathbb{R}^N}\left\{\|\mathbf{v}\|_1+\frac{1}{2}\|\mathbf{Z}(\mathbf{x}-\mathbf{v})\|_2^2\right\}$ in [17]. When $\mathbf{Z}^T\mathbf{Z}$ is diagonal, i.e., $\mathbf{Z}^T\mathbf{Z}=\mathrm{diag}\left(z_1^2,z_2^2,\ldots,z_N^2\right)$, problem (44) has a closed-form solution:

$$x_i^{k+1}=\arg\min_{w_i\in\Omega}\left\{\tfrac{1}{2L}(w_i-d_i)^2+\lambda(|w_i|-s_{z_i}(w_i))\right\}\tag{45}$$

where $s_{z_i}(w_i)$ is the scalar Huber function, defined as:

$$s_{z_i}(x_i)=\begin{cases}z_i^2x_i^2/2, & |x_i|\le 1/z_i^2, z_i\ne 0\\ |x_i|-z_i^2/2, & |x_i|>1/z_i^2, z_i\ne 0\end{cases}\tag{46}$$

and $s_0(x_i):=0$ when $z=0$; and Ω is a set composed of 6 elements $\Omega=\left\{0,z_i,\frac{1}{z_i^2},-\frac{1}{z_i^2},\frac{d_i-\lambda L}{1-\lambda Lz_i^2},\frac{d_i+\lambda L}{1-\lambda Lz_i^2}\right\}$.

(ii) s-difference penalty. the penalty is $g(\mathbf{x})=R(\mathbf{x})-R(\mathbf{x}^s)$, where $\mathbf{x}^s$ is the best s term approximation to $\mathbf{x}$. In Ref. [39], we give the closed-form solutions for some commonly used $R(\mathbf{x})$, such as ℓ_1, ℓ_2, ℓ_{1-2}, MCP, LSP. For example, if $g(\mathbf{x})=\lambda(\|\mathbf{x}\|_1-\|\mathbf{x}^s\|_1)$, then the solution $\mathbf{x}^{k+1}$ is

$$x_i^{k+1}=\begin{cases}d_i, & i\in\Gamma_{\mathbf{d}}^s\\ \mathrm{shrink}(d_i,\lambda L), & else\end{cases}\tag{47}$$

where $\Gamma_{\mathbf{d}}^s$ denotes the index set of top-s elements of vector $\mathbf{d}$ in absolute value.

Second, we analyze the complexity of the proposed smoothing based recovery methods in Table 1. We assume that arithmetic with individual elements has complexity $\mathcal{O}(1)$. Take the APG for example, the main computational complexity comes from the calculation of $\mathbf{x}^{k+1}$. For the $\ell_{\boldsymbol{\varepsilon},p}$-norm smoothing strategy, calculating

Table 2
Some proximal operators with closed-form expressions.

Function type	$g(\mathbf{x})$	$\text{prox}_{Lg}(\mathbf{d})$
ℓ_1-norm penalty	$g(\mathbf{x}) = \lambda\|\mathbf{x}\|_1$	$x_i^{k+1} = \text{shrink}(d_i, \lambda L)$
Hard-thresholding	$g(\mathbf{x}) = \lambda\|\mathbf{x}\|_0$	$x_i^{k+1} = \begin{cases} 0, & \|d_i\| \le \sqrt{2\lambda L} \\ d_i, & \text{else} \end{cases}$
ℓ_p-norm penalty	$g(\mathbf{x}) = \lambda\|\mathbf{x}\|_p^p,\ 0 < p < 1$	$p = 1/2$ or $p = 2/3$ in [8]; Otherwise, it can be solved as in [36]
ℓ_{1-2} penalty	$g(\mathbf{x}) = \lambda(\|\mathbf{x}\|_1 - \alpha\|\mathbf{x}\|_2)$	Lou and Yan [14, Lemma 1]
SCAD	$g(\mathbf{x}) = \sum_i g_i(x_i) g_i(x_i) = \begin{cases} \lambda\|x_i\|, & \|x_i\| < \lambda \\ \frac{2\alpha\lambda\|x_i\| - x_i^2 - \lambda^2}{2(\alpha-1)}, & \lambda \le \|x_i\| < \alpha\lambda \text{ and } \alpha > 2 \\ (\alpha+1)\lambda^2/2, & \|x_i\| \ge \alpha\lambda \end{cases}$	The corresponding solution can be found in [15]
LSP	$g(\mathbf{x}) = \lambda \sum_i \log(1 + \|x_i\|/\theta);\ \theta > 0$	The corresponding solution can be found in [37]
MCP	$g(\mathbf{x}) = \lambda(\|\mathbf{x}\|_1 - S(\mathbf{x}))\ S(\mathbf{x}) = \inf_{\mathbf{v}\in\mathbb{R}^N} \left\{\|\mathbf{v}\|_1 + \frac{1}{2}\|\mathbf{x} - \mathbf{v}\|_2^2\right\}$	The corresponding solution can be found in [37]

$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} Q_{\mathbf{H}}(\mathbf{x}, \mathbf{x}^k)$ by (38) includes: the proximal operator by using (44) and the calculation of gradient $\nabla f_{\varepsilon,p}(\mathbf{Ax} - \mathbf{y})$ by using (7). The former depends on the choice of penalty function $g(\mathbf{x})$. For example, if we choose $g(\mathbf{x}) = \lambda\|\mathbf{x}\|_1$, then the proximal operator costs $\mathcal{O}(N)$ as shown in Table 2. The latter needs to compute the vector $\mathbf{T}$ and the matrix multiplication of $\mathbf{A}^T\mathbf{T}$, which both need to cost $\mathcal{O}(MN)$. For the infimal convolution smoothing strategy, calculating $\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} P_{\mathbf{H}}(\mathbf{x}, \mathbf{x}^k)$ by (42) also includes the proximal operator (44) and the gradient $\nabla f_{\mathbf{B},p}(\mathbf{Ax} - \mathbf{y})$ calculation. The latter needs to compute a matrix multiplication of $\mathbf{A}^T\mathbf{v}$ that it costs $\mathcal{O}(MN)$, and calculate the vector $\mathbf{v}$. If we choose $p = 2$ for the $f_{\mathbf{B},p}$, calculating $\mathbf{v}$ requires $\mathcal{O}(MN)$ by using (31). Otherwise, we need to use the iterative framework (27) and (30) to obtain the approximate $\mathbf{v}$ for $p \neq 2$, which requires $\mathcal{O}(MN + 2i_f M)$ with i_f standing for the iteration number of the framework (27). For the mAPG, the computational complexity is twice that of the APG as there is an additional monitor in the mAPG. From the above analysis, we can find that the proposed two smoothing strategies do not increase the computational complexity too much.

3.3. Convergence analysis

The purpose of this subsection is to analyze the convergence performance of the proposed smoothing based algorithm.

We first look at the convergence performance under the convex condition. Let the sequence $\{\mathbf{x}_{\varepsilon,p}^k\}$ and $\{\mathbf{x}_{\mathbf{B},p}^k\}$ being generated by the algorithm 1 based on the $\ell_{\boldsymbol{\varepsilon},p}$-norm and infimal convolution smoothing strategies, and $\mathbf{x}_{\varepsilon,p}^*$ and $\mathbf{x}_{\mathbf{B},p}^*$ are optimal solutions of the smoothed problems of (34) and (40), respectively.

Proposition 4 *(Ref. [31], Theorem 4.4; Ref. [40], Theorem 2.1). For convex $g(\mathbf{x})$, the $\{\mathbf{x}_{\varepsilon,p}^k\}$ or $\{\mathbf{x}_{\mathbf{B},p}^k\}$ generated by algorithm 1 satisfies*

$$F_{\varepsilon,p}(\mathbf{x}_{\varepsilon,p}^k) - F_{\varepsilon,p}(\mathbf{x}_{\varepsilon,p}^*) \le \frac{2\|\mathbf{x}^0 - \mathbf{x}_{\varepsilon,p}^*\|_{\mathbf{H}}^2}{(k+1)^2} \text{ or}$$
$$F_{\mathbf{B},p}(\mathbf{x}_{\mathbf{B},p}^k) - F_{\mathbf{B},p}(\mathbf{x}_{\mathbf{B},p}^*) \le \frac{2\|\mathbf{x}^0 - \mathbf{x}_{\mathbf{B},p}^*\|_{\mathbf{H}}^2}{(k+1)^2} \tag{48}$$

where $\|\mathbf{x}^0 - \mathbf{x}_{\varepsilon,p}^*\|_{\mathbf{H}}^2$ is defined as $\|\mathbf{x}^0 - \mathbf{x}_{\varepsilon,p}^*\|_{\mathbf{H}}^2 = \langle \mathbf{x}^0 - \mathbf{x}_{\varepsilon,p}^*, \mathbf{H}(\mathbf{x}^0 - \mathbf{x}_{\varepsilon,p}^*)\rangle$, and $\|\mathbf{x}^0 - \mathbf{x}_{\mathbf{B},p}^*\|_{\mathbf{H}}^2$ has a similar meaning.

Proposition 4 means that the proposed algorithm can ensure to obtain an $\mathcal{O}(1/k^2)$ convergence rate for the convex problem. However, one may be more interested in the original problem (33) rather than the smoothed problem (34) or (40), which means that we need to consider the expressions of $F(\mathbf{x}_{\varepsilon,p}^k) - F(\mathbf{x}^*)$ and $F(\mathbf{x}_{\mathbf{B},p}^k) - F(\mathbf{x}^*)$. Here $\mathbf{x}^*$ stands for the optimal solution for the original non-smooth problem (33).

Lemma 1. *For convex $g(\mathbf{x})$, the difference $F(\mathbf{x}_{\varepsilon,p}^k) - F(\mathbf{x}^*)$ or $F(\mathbf{x}_{\mathbf{B},p}^k) - F(\mathbf{x}^*)$ is bounded by a term that depends on $\boldsymbol{\varepsilon}$ or $\mathbf{B}$ and p, respectively.*

$$F(\mathbf{x}_{\varepsilon,p}^k) - F(\mathbf{x}^*) \le \frac{2\|\mathbf{x}^0 - \mathbf{x}_{\varepsilon,p}^*\|_{\mathbf{H}}^2}{(k+1)^2} + \sum_m \varepsilon_m \quad \text{or}$$
$$F(\mathbf{x}_{\mathbf{B},p}^k) - F(\mathbf{x}^*) \le \frac{2\|\mathbf{x}^0 - \mathbf{x}_{\mathbf{B},p}^*\|_{\mathbf{H}}^2}{(k+1)^2} + \delta_{\mathbf{B},p} M^{\frac{p}{2(p-1)}} \tag{49}$$

where $\delta_{\mathbf{B},p}$ is defined as

$$\delta_{\mathbf{B},p} = \frac{p-1}{2}\left(\frac{2}{pb_{\min}}\right)^{\frac{p}{p-1}} \tag{50}$$

Proof. By using the inequality (6), we have

$$F(\mathbf{x}) \le F_{\varepsilon,p}(\mathbf{x}) \le F(\mathbf{x}) + \sum_m \varepsilon_m, \quad \forall \mathbf{x} \in \mathbb{R}^N \tag{51}$$

Note that $\mathbf{x}_{\varepsilon,p}^*$ is an optimal solution of $F_{\boldsymbol{\varepsilon},p}(\mathbf{x})$, by using (48), we have

$$\begin{aligned} F(\mathbf{x}_{\varepsilon,p}^k) - F(\mathbf{x}^*) &\le F_{\varepsilon,p}(\mathbf{x}_{\varepsilon,p}^k) - F_{\varepsilon,p}(\mathbf{x}^*) + \sum_m \varepsilon_m \\ &\le F_{\varepsilon,p}(\mathbf{x}_{\varepsilon,p}^k) - F_{\varepsilon,p}(\mathbf{x}_{\varepsilon,p}^*) + \sum_m \varepsilon_m \\ &\le \frac{2\|\mathbf{x}^0 - \mathbf{x}_{\varepsilon,p}^*\|_{\mathbf{H}}^2}{(k+1)^2} + \sum_m \varepsilon_m \end{aligned} \tag{52}$$

Similarly, for the infimal convolution smoothing method, by using Proposition 2, the following inequality holds

$$f(\mathbf{Ax} - \mathbf{y}) - \delta_{\mathbf{B},p}(L_f)^{\frac{p}{p-1}} \le f_{\mathbf{B},p}(\mathbf{Ax} - \mathbf{y}) \le f(\mathbf{Ax} - \mathbf{y}) \tag{53}$$

Here we use the property of the convex function $f(\mathbf{z}) = \|\mathbf{z}\|_1, \mathbf{z} \in \mathbb{R}^M$, that the subgradients of f are bounded by $L_f = \sqrt{M}$. Contrasting with (52), we have

$$\begin{aligned} F(\mathbf{x}_{\mathbf{B},p}^k) - F(\mathbf{x}^*) &\le F_{\mathbf{B},p}(\mathbf{x}_{\mathbf{B},p}^k) - F_{\mathbf{B}}(\mathbf{x}^*) + \delta_{\mathbf{B},p} M^{\frac{p}{2(p-1)}} \\ &\le F_{\mathbf{B},p}(\mathbf{x}_{\mathbf{B},p}^k) - F_{\mathbf{B},p}(\mathbf{x}_{\mathbf{B},p}^*) + \delta_{\mathbf{B},p} M^{\frac{p}{2(p-1)}} \\ &\le \frac{2\|\mathbf{x}^0 - \mathbf{x}_{\mathbf{B},p}^*\|_{\mathbf{H}}^2}{(k+1)^2} + \delta_{\mathbf{B},p} M^{\frac{p}{2(p-1)}} \end{aligned} \tag{54}$$

This completes the proof. □

Theorem 1. *Let $\varsigma > 0$, if we choose the symmetric positive semidefinite matrix $\mathbf{H}$ as $\mathbf{H} = \rho\mathbf{A}^T\mathbf{B}^T\mathbf{BA}$ with $\rho = (p-1)\left(\frac{p}{2}\right)^{\frac{1}{p-1}}\left(\frac{\sqrt{M}}{b_{\min}}\right)^{\frac{p-2}{p-1}}$ and select $b_{\max} = 2b_{\min} = \frac{4\sqrt{M}}{p}\left(\frac{p-1}{\varsigma}\right)^{\frac{p-1}{p}}$, then an ς-optimal solution of (40), i.e., $F(\mathbf{x}_{\mathbf{B}}^k) - F(\mathbf{x}^*) \le \varsigma$, can be obtained by using the*

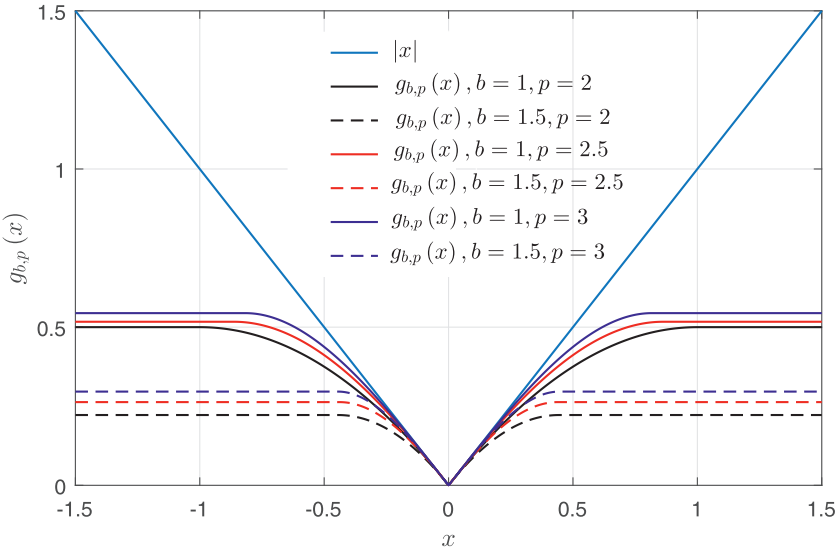

Fig. 4. The scalar $g_{\mathbf{B},p}(\mathbf{x})$ with different b and p.

proposed infimal convolution smoothing based APG method after at most iterations of

$$k = \max\left\{\left(4(p-1)\sqrt{2M/p}\left\|\mathbf{A}\left(\mathbf{x}^0 - \mathbf{x}^*_{\varepsilon,p}\right)\right\|_2/\varsigma\right) - 1, 1\right\} \tag{55}$$

Proof. Taking $b_{\max} = 2b_{\min} = \frac{4\sqrt{M}}{p}\left(\frac{p-1}{\varsigma}\right)^{\frac{p-1}{p}}$ and $\mathbf{H}=\rho\mathbf{A}^T\mathbf{B}^T\mathbf{B}\mathbf{A}$, and using Lemma 1, we have

$$\begin{aligned} F\left(\mathbf{x}^k_{\mathbf{B},p}\right) - F(\mathbf{x}^*) &\le \frac{2\rho b^2_{\max}\left\|\mathbf{A}\left(\mathbf{x}^0 - \mathbf{x}^*_{\mathbf{B},p}\right)\right\|_2^2}{(k+1)^2} + \delta_{\mathbf{B},p}M^{\frac{p}{2(p-1)}} \\ &= 2(p-1)\left(\frac{p}{2}\right)^{\frac{1}{p-1}}\left(\frac{\sqrt{M}}{b_{\min}}\right)^{\frac{p-2}{p-1}} b^2_{\max}\frac{\left\|\mathbf{A}\left(\mathbf{x}^0 - \mathbf{x}^*_{\mathbf{B},p}\right)\right\|_2^2}{(k+1)^2} \\ &\quad + \frac{p-1}{2}\left(\frac{2}{pb_{\min}}\right)^{\frac{p}{p-1}} M^{\frac{p}{2(p-1)}} \\ &= \frac{16M(p-1)^2\left\|\mathbf{A}\left(\mathbf{x}^0 - \mathbf{x}^*_{\varepsilon,p}\right)\right\|_2^2}{p\varsigma(k+1)^2} + \frac{\varsigma}{2} \end{aligned} \tag{56}$$

Therefore, to guarantee the inequality $F\left(\mathbf{x}^k_{\mathbf{B},p}\right) - F(\mathbf{x}^*) \le \varsigma$, the iteration number k needs to satisfy that

$$\frac{4(p-1)\sqrt{2M}\left\|\mathbf{A}\left(\mathbf{x}^0 - \mathbf{x}^*_{1,\varepsilon}\right)\right\|_2}{\sqrt{p}(k+1)} \le \varsigma \tag{57}$$

This means that after at most $k = \max\left\{\left(4(p-1)\sqrt{2M/p}\left\|\mathbf{A}\left(\mathbf{x}^0 - \mathbf{x}^*_{\varepsilon,p}\right)\right\|_2/\varsigma\right) - 1, 1\right\}$ iterations, we can obtain an ς-optimal solution of the original problem. This completes the proof. □

For the $\ell_{\boldsymbol{\varepsilon},p}$-norm smoothing method, we have a similar result without proof.

Theorem 2. *For convex $g(\mathbf{x})$, let $\varsigma > 0$, if we choose the smoothing matrix $\boldsymbol{\varepsilon}$ as $\sum_m \varepsilon_m = 2M\varepsilon_{\min} = \varsigma/2$ and the symmetric positive semidefinite matrix $\mathbf{H}$ as $\mathbf{H}=(p-1)\mathbf{A}^T\varepsilon^{-1}\mathbf{A}$, then an ς-optimal solution of (34), i.e., $F\left(\mathbf{x}^k_{\varepsilon,p}\right) - F(\mathbf{x}^*) \le \varsigma$, can be obtained by using the proposed $\ell_{\boldsymbol{\varepsilon},p}$-norm smoothing based APG method after at most iterations of*

$$k = \max\left\{\left(4\sqrt{M(p-1)}\left\|\mathbf{A}\left(\mathbf{x}^0 - \mathbf{x}^*_{\varepsilon,p}\right)\right\|_2/\varsigma\right) - 1, 1\right\} \tag{58}$$

From (55) and (58), we can find that the most iteration numbers have a same expression if we choose $p = 2$. Meanwhile, if we choose the symmetric positive semidefinite matrix $\mathbf{H}$ as $\mathbf{H} = (p-1)\|\mathbf{A}\|_2^2\mathbf{I}_N/\varepsilon_{\min}$ and $\mathbf{H} = \rho\|\mathbf{B}\mathbf{A}\|_2^2\mathbf{I}_N$ for $\ell_{\boldsymbol{\varepsilon},p}$-norm and infimal convolution smoothing methods, we have similar conclusions as Theorems 1 and 2, respectively.

For the non-convex penalty function, since we employ the mAPG framework to solve the optimization problem and the smoothed functions $f_{\varepsilon,p}(\mathbf{A}\mathbf{x}-\mathbf{y})$ and $f_{\mathbf{B},p}(\mathbf{A}\mathbf{x}-\mathbf{y})$ have Lipschitz continuous gradients, same as the analysis of Theorem 1 in [32], we can obtain the convergence performance under the nonconvex condition.

Theorem 3 *(Ref. [32], Theorem 1). Let $g(\mathbf{x})$ be a proper and lower semicontinuous, for non-convex and non-smooth $g(\mathbf{x})$, assume that $F_{\boldsymbol{\varepsilon},p}(\mathbf{x})$ and $F_{\mathbf{B},p}(\mathbf{x})$ are coercive. Let $\mathbf{x}^*_{\varepsilon,p}$ and $\mathbf{x}^*_{\mathbf{B},p}$ be any accumulation points of $\left\{\mathbf{x}^k_{\varepsilon,p}\right\}$ and $\left\{\mathbf{x}^k_{\mathbf{B},p}\right\}$, we have $0 \in \partial F_{\varepsilon,p}(\mathbf{x}^*)$ and $0 \in \partial F_{\mathbf{B},p}(\mathbf{x}^*)$, respectively, i.e., $\mathbf{x}^*_{\varepsilon,p}$ and $\mathbf{x}^*_{\mathbf{B},p}$ are critical points.*

4. Extensions

In this section, we discuss some related algorithms for solving problem (3), show a link between the smoothing strategy with other methods, and simply extend the infimal convolution smoothing method to construct a non-convex penalty function.

4.1. Related algorithms

Here, we discuss some related algorithms. First, we look at the Moreau envelope. From the definition of $h_{\mathbf{B},p}$, we can find that the Moreau's proximal $h^{\mathrm{M}}_{\beta}(\mathbf{A}\mathbf{x}-\mathbf{y})$ is a special case of the infimal convolution function $h_{\mathbf{B},p}(\mathbf{A}\mathbf{x}-\mathbf{y})$ under the condition of $p = 2$ and $\mathbf{B}^T\mathbf{B}$ is the scale identity matrix $\frac{1}{\beta}\mathbf{I}_M$. Then, by using Propositions 1–3, we can obtain three important properties of the Moreau's proximal operator, which can also be find in [41–43].

Let $h : \mathbb{R}^M \to \mathbb{R}\bigcup\{\infty\}$ be a closed proper convex function and let $\mathbf{A} \in \mathbb{R}^{M\times N}$ be a given matrix. For any $\mathbf{x} \in \mathbb{R}^N$, $\mathbf{y} \in \mathbb{R}^M$ and $\beta > 0$, the following results hold:

(1) $h^{\mathrm{M}}_{\beta}(\mathbf{x}) = h\left(\mathrm{prox}_{\beta h}(\mathbf{x})\right) + \frac{1}{2\beta}\left\|\mathrm{prox}_{\beta h}(\mathbf{x}) - \mathbf{x}\right\|_2^2$

(2) $h^{\mathrm{M}}_{\beta}(\mathbf{A}\mathbf{x}-\mathbf{y})$ is continuously differentiable and its gradient is Lipschitz continuous with constant $\|\mathbf{A}\|_2^2/\beta$. The gradient of $h^{\mathrm{M}}_{\beta}(\mathbf{A}\mathbf{x}-\mathbf{y})$ is given by

$$\nabla h^{\mathrm{M}}_{\beta}(\mathbf{A}\mathbf{x}-\mathbf{y}) = \frac{1}{\beta}\mathbf{A}^T\left(\mathbf{A}\mathbf{x}-\mathbf{y}-\mathrm{prox}_{\beta h}(\mathbf{A}\mathbf{x}-\mathbf{y})\right) \tag{59}$$

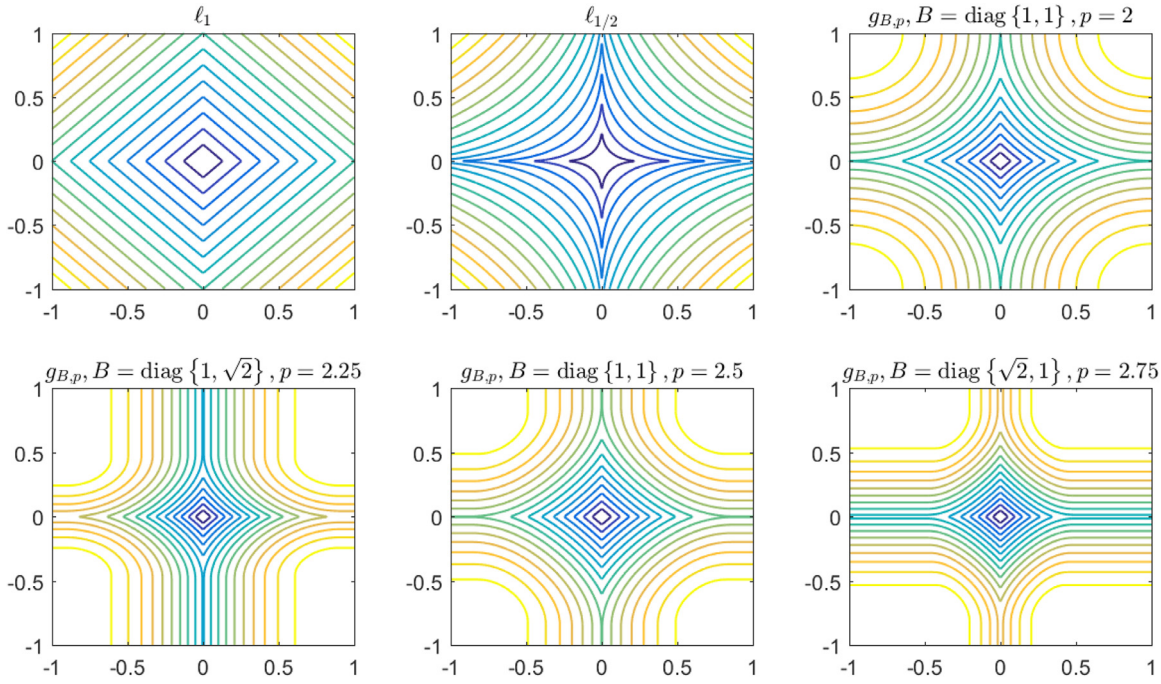

Fig. 5. Level curves of ℓ_1, $\ell_{1/2}$ and $g_{\mathbf{B},p}(\mathbf{x})$ with different $\mathbf{B}$ and p.

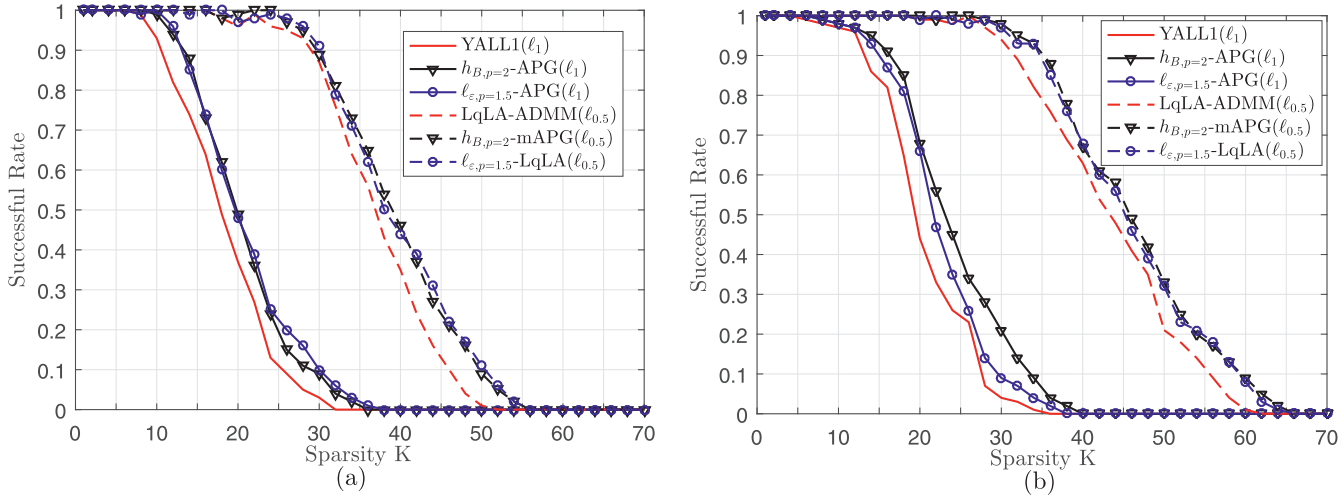

Fig. 6. Recovery performance versus sparsity for the compared methods: (a) Gaussian mixture noise, (b) Cauchy distribution noise.

(3) Suppose that the subgradients of h over $\mathbb{R}^M$ are bounded by L_h, then

$$h(\mathbf{Ax}-\mathbf{y}) - \frac{\beta L_h^2}{2} \le h_\beta^{\mathrm{M}}(\mathbf{Ax}-\mathbf{y}) \le h(\mathbf{Ax}-\mathbf{y}) \tag{60}$$

Second, for the non-smooth optimization problem (3), one common alternative is to transform the non-differentiable problem into a smooth counterpart, for example, the Nesterovs smoothing-based method [41,44]. Another commonly used method is the decomposition based algorithm [45] or alternating minimization (AM) method [46], which introduce an auxiliary vector $\mathbf{u}$ in (3) and consider the following optimization problem

$$\min_{\mathbf{x},\mathbf{u}} \left\{ \|\mathbf{u}\|_1 + \frac{\beta}{2}\|\mathbf{u}-\mathbf{Ax}+\mathbf{y}\|_2^2 + g(\mathbf{x}) \right\} \tag{61}$$

In Ref. [47] and [48], the authors have shown that there is a close relationship between the Moreau's proximal smoothing model and the decomposition model. Indeed, by fixing $\mathbf{x}$ and minimizing the objective function of (61) with respect to $\mathbf{u}$, we can obtain

$$\begin{aligned} &\min_{\mathbf{x},\mathbf{u}} \left\{ \|\mathbf{u}\|_1 + \frac{\beta}{2}\|\mathbf{u}-\mathbf{Ax}+\mathbf{y}\|_2^2 + g(\mathbf{x}) \right\} \\ &= \min_{\mathbf{x}} \left\{ h_{1/\beta}^{\mathrm{M}}(\mathbf{Ax}-\mathbf{y}) + g(\mathbf{x}) \right\} \end{aligned} \tag{62}$$

Here, we can also find a connection between the proposed infimal convolution smoothing technique and the AM algorithm. The infimal convolution smoothing technique can be thought as a development of the AM with a more flexible punishment of $\mathbf{Ax}+\mathbf{y}=\mathbf{z}$:

$$\min_{\mathbf{x},\mathbf{z}} \left\{ F_{\mathbf{B},p}(\mathbf{x}) = \|\mathbf{u}\|_1 + \frac{1}{2}\|\mathbf{B}(\mathbf{u}-\mathbf{Ax}+\mathbf{y})\|_2^p + g(\mathbf{x}) \right\} \tag{63}$$

Comparing (63) with (61), the infimal convolution smoothing is equal to the AM algorithm if we choose $\mathbf{B}^T\mathbf{B}=\beta\mathbf{I}_M$ and $p=2$.

From the convergence analysis in Section 3, we can find that the recovery performance is significantly dependent on the choice of the scale matrix $\mathbf{B}$ and the parameter p. With $b_{\min} \to \infty$, (63) is

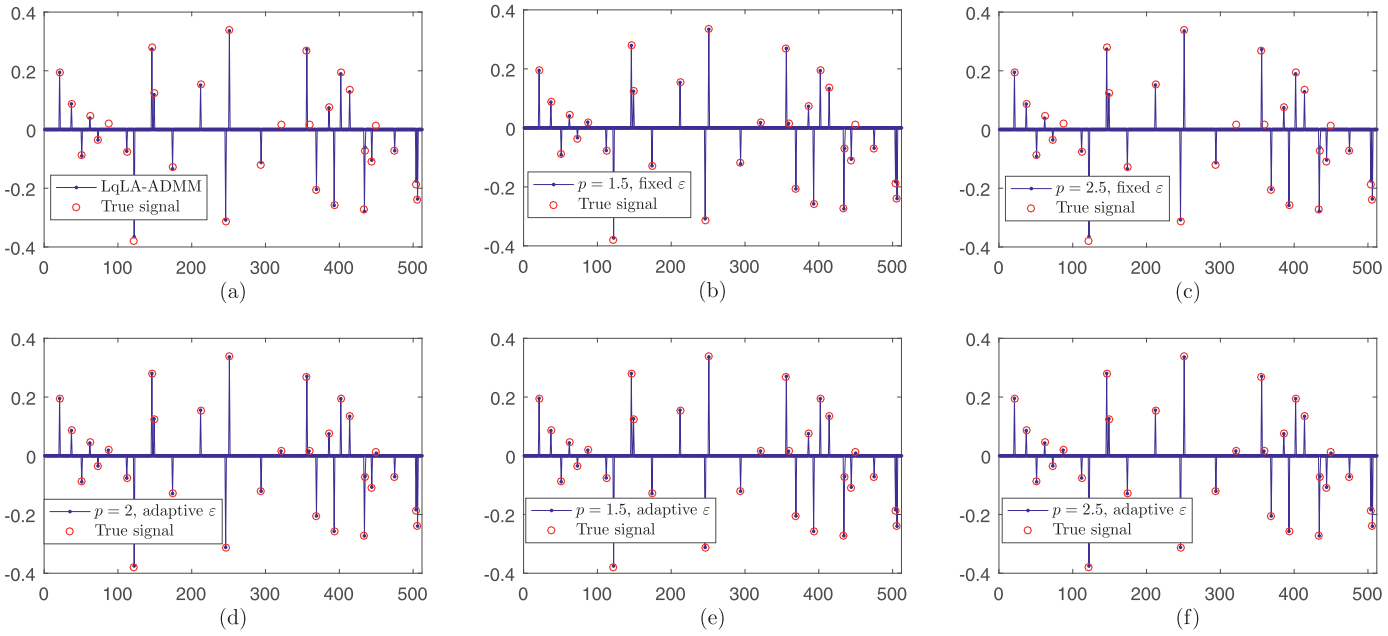

Fig. 7. Recovery performance of the compared methods under Gaussian mixture noise: (a) LqLA-ADMM, Rel.Err =4.37%, (b) $\ell_{\varepsilon,p=1.5}$-LqLA with fixed $\boldsymbol{\varepsilon}$, Rel.Err =1.74%, (c) $\ell_{\varepsilon,p=2.5}$-LqLA with fixed $\boldsymbol{\varepsilon}$, Rel.Err =4.41%, (d) $\ell_{\varepsilon,p=2}$-LqLA with adaptive $\boldsymbol{\varepsilon}$, Rel.Err =0.82%, (e) $\ell_{\varepsilon,p=1.5}$-LqLA with adaptive $\boldsymbol{\varepsilon}$, Rel.Err =0.79% and (f) $\ell_{\varepsilon,p=2.5}$-LqLA with adaptive $\boldsymbol{\varepsilon}$, Rel.Err =0.91%.

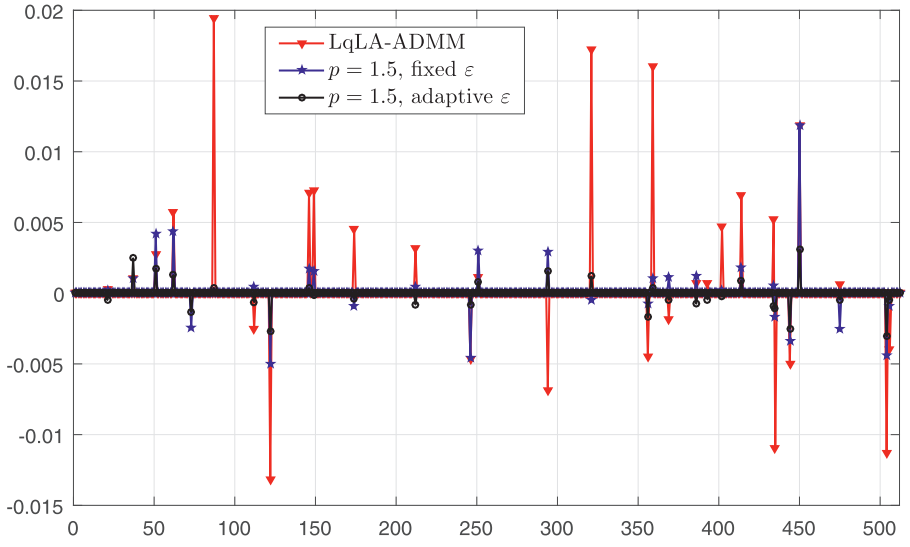

Fig. 8. Reconstruction errors $\hat{\mathbf{x}} - \mathbf{x}^*$ of the LqLA-ADMM, $\ell_{\varepsilon,p=1.5}$-LqLA with fixed $\boldsymbol{\varepsilon}$ and $\ell_{\varepsilon,p=1.5}$-LqLA with adaptive $\boldsymbol{\varepsilon}$.

equivalent to the constrained problem (4). However, with a very large $\mathbf{B}$, the algorithm would be very slow and impractical. Specifically, we can use a self-adjustment strategy for b_m with a properly small starting value and gradually increase it until reaching the target value based on the convergence speed. A simple strategy is that we compare $\left|\left[\mathbf{A}\mathbf{x}^k - \mathbf{y}\right]_m / \left[\mathbf{A}\mathbf{x}^{k-1} - \mathbf{y}\right]_m\right|$ with a constant $\gamma \in (0,\ 1)$ after the $\mathbf{x}$ updating: if its value is bigger than γ, then $b_m^{k+1} = \min\left\{b_{\text{target}}, b_m^k/\gamma\right\}$, else $b_m^{k+1} = b_m^k$. For the choice of matrix $\boldsymbol{\varepsilon}$ of $\ell_{\boldsymbol{\varepsilon},p}$-norm smoothing algorithm, we have a similar self-adjustment strategy as $\mathbf{B}$. This adjustment strategy based on the convergence speed leads to some improvements in our experiment as shown in Section 5.

4.2. Extend to construct non-convex penalty function

Inspired by the generalized MCP in [17] and the integral convolution based penalty function in [49], we can construct a new penalty function that may improve the sparsity

$$g_{\mathbf{B},p}(\mathbf{x}) = \|\mathbf{x}\|_1 - h_{\mathbf{B},p}(\mathbf{x}) \tag{64}$$

or $g_{\mathbf{B},p}(\mathbf{D}\mathbf{x}) = \|\mathbf{D}\mathbf{x}\|_1 - h_{\mathbf{B},p}(\mathbf{D}\mathbf{x})$ when $\mathbf{D}\mathbf{x}$ is sparse.

Fig. 4 plots the scalar $g(x)$ with different parameters and Fig. 5 shows the contours of various regularizations. From Figs. 4 and 5, we can find that the $g(\mathbf{x})$ approaches the ℓ_0-norm closer than the normal ℓ_1-norm $\|\mathbf{x}\|_1$, hence promoting sparsity.

By using this infimal convolution based penalty function, the sparse recovery problem (2) turns into

$$\min_{\mathbf{x}} F(\mathbf{x}) = f(\mathbf{x}) + \|\mathbf{x}\|_1 - h_{\mathbf{B},p}(\mathbf{x}) \tag{65}$$

If $f(\mathbf{x})$ is convex or $\frac{L_f}{2}\|\mathbf{x}\|_2^2 - f(\mathbf{x})$ is convex, we can use the DCA [50] or the proximal DCA (PDCA) [51] to solve this minimization problem. The DCA solves (65) by decomposing the objective function as the difference of $f(\mathbf{x}) + \|\mathbf{x}\|_1$ and $h_{\mathbf{B},p}(\mathbf{x})$. Then the

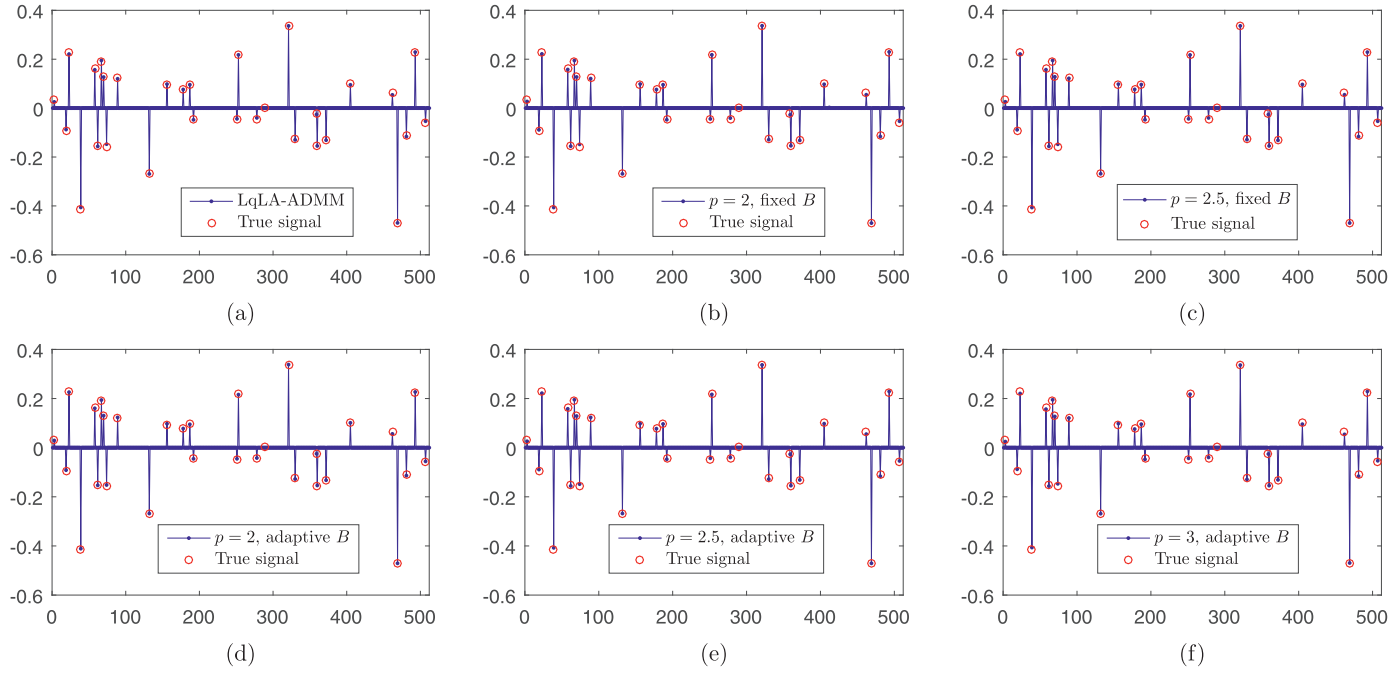

Fig. 9. Recovery performance of the compared methods under Cauchy distribution noise: (a) LqLA-ADMM, Rel.Err =2.62%, (b) $h_{\mathbf{B},p=2}$-mAPG with fixed $\mathbf{B}$, Rel.Err =2.38%, (c) $h_{\mathbf{B},p=2.5}$-mAPG with fixed $\mathbf{B}$, Rel.Err =2.45%, (d) $h_{\mathbf{B},p=2}$-mAPG with adaptive $\mathbf{B}$, Rel.Err =1.58%, (e) $h_{\mathbf{B},p=2.5}$-mAPG with adaptive $\mathbf{B}$, Rel.Err =2.29% and (f) $h_{\mathbf{B},p=3}$-mAPG with adaptive $\mathbf{B}$, Rel.Err =2.37%.

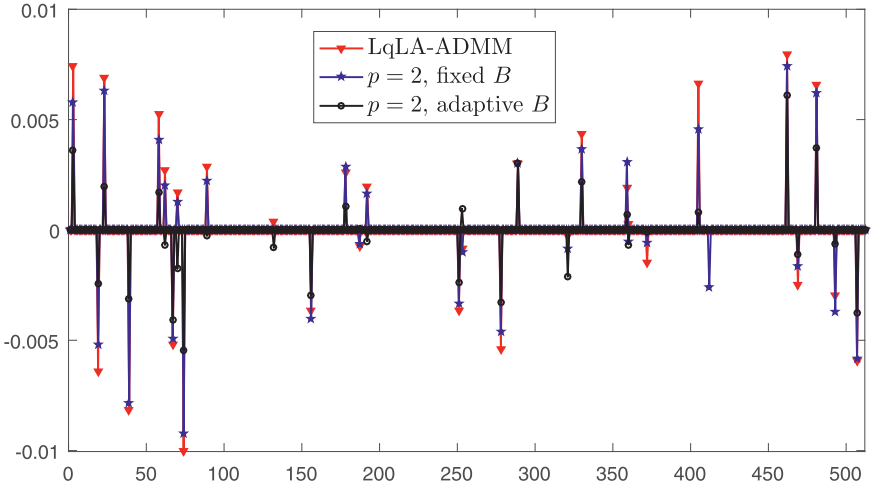

Fig. 10. Reconstruction errors $\hat{\mathbf{x}} - \mathbf{x}^*$ of the LqLA-ADMM, $h_{\mathbf{B},p=2}$-mAPG with fixed $\mathbf{B}$ and $h_{\mathbf{B},p=2}$-mAPG with adaptive $\mathbf{B}$.

subproblem of the corresponding DCA takes the following form:

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \left\{ f(\mathbf{x}) + \|\mathbf{x}\|_1 - \left(h_{\mathbf{B},p}(\mathbf{x}^k) + \left\langle \partial h_{\mathbf{B},p}(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \right\rangle \right) \right\} \tag{66}$$

where $\partial h_{\mathbf{B},p}(\mathbf{x}^k)$ can be calculated by using (29).

Another DC decomposition of (65) is the difference of $\frac{L_f}{2}\|\mathbf{x}\|_2^2 + \|\mathbf{x}\|_1$ and $\frac{L_f}{2}\|\mathbf{x}\|_2^2 - f(\mathbf{x}) + h_{\mathbf{B},p}(\mathbf{x})$, and the corresponding DCA subproblem is

$$\begin{cases} \mathbf{w}^k \in \partial\left(\dfrac{L_f}{2}\left\|\mathbf{x}^k\right\|_2^2 - f\left(\mathbf{x}^k\right) + h_{\mathbf{B},p}\left(\mathbf{x}^k\right) \right) \\ \mathbf{x}^{k+1} = \arg\min\limits_{\mathbf{x}} \left\{ \|\mathbf{x}\|_1 + \dfrac{L_f}{2}\left\|\mathbf{x} - \mathbf{w}^k/L_f\right\|_2^2 \right\} \end{cases} \tag{67}$$

As it can be solved by using the proximal operator (44), this method can also be called as PDCA. Since the infimal convolution based penalty $\|\mathbf{x}\|_1 - h_{\mathbf{B},p}(\mathbf{x})$ can approximate the original ℓ_0-norm better and the corresponding solution process is also simple, we believe that this new penalty function is expected to have potential value on other sparse reconstruction problems, which is also our next work.

5. Numerical experiments

In this section, simulations are performed to demonstrate the proposed conclusions and evaluate the performance of the proposed smoothing based APG and mAPG algorithms. We call the $\ell_{\boldsymbol{\varepsilon},p}$-norm and the infimal convolution smoothing approximation methods as the $\ell_{\boldsymbol{\varepsilon},p}$-APG (mAPG) and $h_{\mathbf{B},p}$-APG (mAPG) for short, respectively. All experiments are performed in MATLAB 2015b running on ASUS laptop with Intel(R) Core(TM) i7-8550U CPU, 8 GB of RAM and 64 bit Windows 10 operating system.

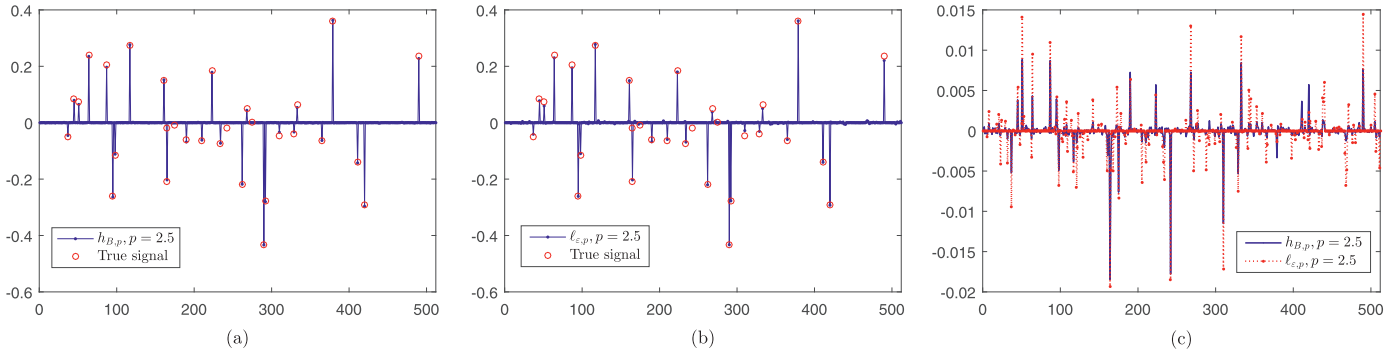

Fig. 11. Recovery performance of the compared methods with ℓ_{1-2}-norm penalty under Gaussian mixture noise: (a) $h_{\mathbf{B},p=2.5}$-mAPG, Rel.Err=3.9%, (b) $\ell_{\varepsilon,p=2.5}$-mAPG, Rel.Err =5.6% and (c) errors comparison.

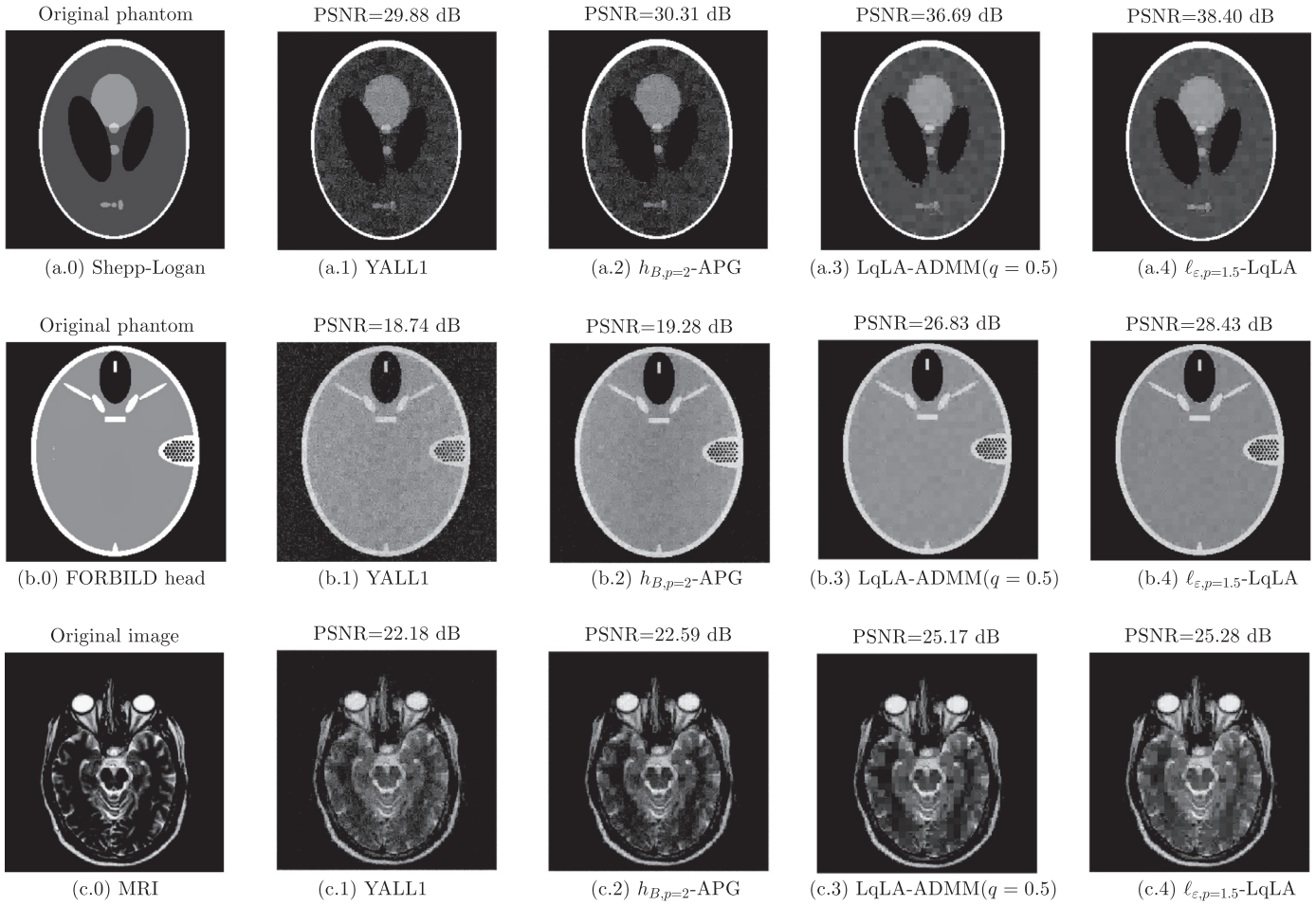

Fig. 12. Recovery performance the compared methods on different images under Gaussian mixture noise: (a) the first row corresponds to Shepp-Logan, (b) the second row corresponds to FORBILD head and (c) the third row corresponds to MRI image.

In our experiments, we consider two types of impulsive noise. (1) Gaussian mixture noise, which is a two-component Gaussian mixture model with probability density function given by

$$\mathbf{n}=\eta\mathcal{N}\left(\mathbf{0},\sigma^{\mathbf{2}}\right)+(\mathbf{1}-\eta)\mathcal{N}\left(\mathbf{0},\kappa\sigma^{\mathbf{2}}\right) \tag{68}$$

The first term of this model is the Gaussian thermal noise, which stands for the normal background noise, while the second one stands for the impulsive behavior of the noise. The ratio and strength of the outliers in the noise are controlled by the parameters $\eta \in (0,\ 1)$ and $\kappa > 1$, respectively. We use the signal-to-noise ratio (SNR) as $\mathrm{SNR}=20\log_{10}\left(\left\|\mathbf{A}\hat{\mathbf{x}}-E\left\{\mathbf{A}\hat{\mathbf{x}}\right\}\right\|_2/\|\mathbf{n}\|_2\right)$ to quantify the strength of noise, where $\hat{\mathbf{x}}$ denotes the true signal. (2) Cauchy distribution noise, which is a special case of both the stable distribution and the t-distribution. The characteristic function of Cauchy distribution with scale γ and location δ is given by $\varphi(t)=\exp(j\delta t-|\gamma t|)$. Then we measure the different Cauchy distribution noise levels with the scale parameter γ. We test two types of matrices $\mathbf{A}$: the random Gaussian matrix with i.i.d. standard Gaussian entries and being normalized that each column has unit norm, and the random partial DCT matrix which is formed by randomly selecting rows from the full DCT matrix.

We apply two methods in comparison with the proposed algorithm: the ℓ_1-norm loss and ℓ_1-norm penalty based YALL1 as in [24] by using the alternating direction algorithm, and the

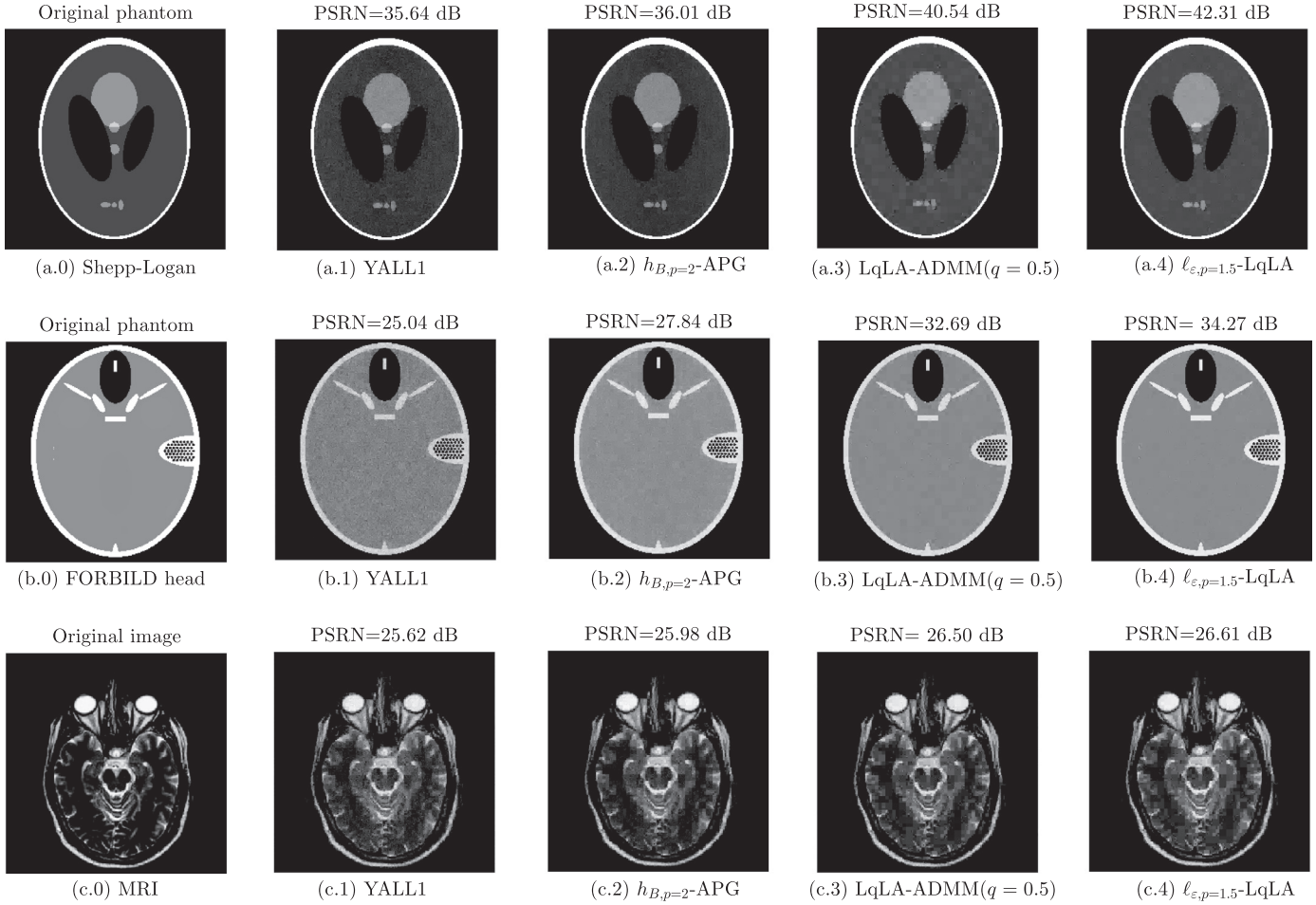

Fig. 13. Recovery performance the compared methods on different images under Cauchy distribution noise: (a) the first row corresponds to Shepp–Logan, (b) the second row corresponds to FORBILD head, (c) the third row corresponds to MRI image.

LqLA-ADMM as in [30] by using the ℓ_q-norm penalty and the smooth strategy on the ℓ_1-loss function. We select three penalty functions: (1) ℓ_1-norm penalty, $g(\mathbf{x})=\lambda\|\mathbf{x}\|_1$; (2) $\ell_{0.5}$-norm penalty, $g(\mathbf{x})=\lambda\|\mathbf{x}\|_{1/2}^{1/2}$; (3) ℓ_{1-2} penalty, $g(\mathbf{x})=\lambda(\|\mathbf{x}\|_1-\alpha\|\mathbf{x}\|_2)$ with α=1. We also apply the $\ell_{\boldsymbol{\varepsilon},p}$-norm smoothing strategy for LqLA-ADMM to replace the original '$\ell_{1,\varepsilon}$' smoothing strategy, which is a special case of $\ell_{\boldsymbol{\varepsilon},p}$ with $\varepsilon=\varepsilon\mathbf{I}$ and $p=2$. We can find that the $\ell_{\boldsymbol{\varepsilon},p}$ also meets the convergence condition of [30], and we denote this $\ell_{\boldsymbol{\varepsilon},p}$ smoothing LqLA-ADMM as $\ell_{\boldsymbol{\varepsilon},p}$-LqLA for short. The initial value for all the methods is an approximated solution of the ℓ_1 minimization using ADMM after N iterations. The max iteration for all these methods is $5N$, and the stopping condition is set to be $\frac{\|\mathbf{x}^{[k]}-\mathbf{x}^{[k-1]}\|_2}{\max\{\|\mathbf{x}^{[k]}\|_2,1\}}<10^{-5}$. Meanwhile, we initialize the parameters for the matrix $\boldsymbol{\varepsilon}$ and $\mathbf{B}$ as $\varepsilon_m\propto|[\mathbf{Ax}-\mathbf{y}]_m|$ and $b_m\propto 1/|[\mathbf{Ax}-\mathbf{y}]_m|$, respectively. This can be illustrated by Fig. 2, when the amplitude of $[\mathbf{Ax}-\mathbf{y}]_m$ is very large, which means that y_m may be contaminated by the impulsive noise, then we need to reduce the value of b_m to reduce the impact of noise. The weighting parameter λ is selected to balance the regularization and data fitting. On the one hand, λ should be big enough to weaken the influence of fitting the corrupted data. On the other hand, if λ is too big, the reconstruction is mostly over regularized. We vary the regularization parameter λ from 10^{-4} to 10 (with 30 logarithmically equally spaced) for each method and noise condition, and then select the best one as the result.

In the first study, we look at the success rates with 100 random instances under different noise conditions: Gaussian mixture noise with $\eta=0.9$, $\kappa=10^3$, SNR=30 dB and Cauchy distribution noise with γ=10^{-4}. For the original K-sparse vector $\hat{\mathbf{x}}$, we generate it with random index set and draw non-zero elements with standard normal distribution. We set the size of random Gaussian matrix $\mathbf{A}$ as 100×256, and consider a recovery $\mathbf{x}^*$ as successful if the relative error of recovery (Rel.Err) satisfies $\|\mathbf{x}^*-\hat{\mathbf{x}}\|_2/\|\hat{\mathbf{x}}\|_2\le10^{-2}$. Fig. 6 shows the success rates of the comparing methods for both Gaussian mixture noise and Cauchy distribution noise. From Fig. 6, we can find that the $\ell_{\boldsymbol{\varepsilon},p}$-APG and $h_{\mathbf{B},p}$-APG outperform the YALL1 under ℓ_1-norm penalty, and the $h_{\mathbf{B},p}$-mAPG and $\ell_{\boldsymbol{\varepsilon},p}$-LqLA outperform the LqLA-ADMM under $\ell_{0.5}$-norm penalty. Meanwhile, by comparing recovery performance of ℓ_1 and $\ell_{0.5}$-norm penalties, we can find that the non-convex $\ell_{0.5}$-norm penalty function bring better performance than the convex ℓ_1-norm penalty function. This is mainly due to that the non-convex penalties can approximate the ℓ_0-norm more closely than the convex ℓ_1-norm.

In the second study, we focus on the recovery quantities of these methods. We set the size of random Gaussian matrix $\mathbf{A}$ as 100×512 and the sparsity of vector $\hat{\mathbf{x}}$ is $K=30$. Fig. 7 presents the recovery signals of different methods with $\ell_{0.5}$-norm penalty under Gaussian mixture noise with $\eta=0.9$, $\kappa=10^3$, SNR=20 dB. Fig. 8 presents the reconstruction errors $\hat{\mathbf{x}}-\mathbf{x}^*$ of the LqLA-ADMM, $\ell_{\varepsilon,p=1.5}$-LqLA with fixed $\boldsymbol{\varepsilon}$ and $\ell_{\varepsilon,p=1.5}$-LqLA with adaptive $\boldsymbol{\varepsilon}$. Fig. 9 presents the recovery signals of different methods with $\ell_{0.5}$-norm penalty under Cauchy distribution noise with γ=5×10^{-4}. Fig. 10 presents the reconstruction errors $\hat{\mathbf{x}}-\mathbf{x}^*$ of the LqLA-ADMM, $h_{\mathbf{B},p=2}$-mAPG with fixed $\mathbf{B}$ and $h_{\mathbf{B},p=2}$-mAPG with adaptive $\mathbf{B}$. From Figs. 7 to 10, it is clear that all these

compared methods work well on these impulsive noise conditions and can obtain nice reconstructions with a few minor mistakes, which once again demonstrates the effectiveness of the proposed smoothing strategies based reconstruction algorithm. Meanwhile, we can also find that the smoothing strategies based mAPG can obtain a better reconstruction, and when the smoothing scale p becomes larger (from 1.5 to 2.5), the recovery performance decreases. Moreover, the smoothing $\ell_{\boldsymbol{\varepsilon},p}$ and $h_{\mathbf{B},p}$ based methods with adaptive matrix $\boldsymbol{\varepsilon}$ and $\mathbf{B}$ can gain better reconstructions than fixed $\boldsymbol{\varepsilon}$ and $\mathbf{B}$, respectively.

Fig. 11 (a) and (b) shows the reconstructions of $h_{\mathbf{B},p=2.5}$-mAPG and $\ell_{\varepsilon,p=2.5}$-mAPG with ℓ_{1-2}-norm penalty under Gaussian mixture noise with $\eta = 0.9$, $\kappa = 10^3$, SNR=20 dB. Fig. 11 (c) shows the errors comparison. The Rel.Err of $h_{\mathbf{B},p=2.5}$-mAPG and $\ell_{\varepsilon,p=2.5}$-mAPG are 3.9% and 5.6%, respectively. It can be observed that the $h_{\mathbf{B},p}$ smoothing strategy is better than the $\ell_{\boldsymbol{\varepsilon},p}$ smoothing strategy with the same smoothing scale $p = 2.5$.

Finally, we evaluate the performance of the methods on image reconstruction. We test three images, the Shepp–Logan phantom, the 2D FORBILD head phantom [52], and an MRI image. Each image has a size 256×256 ($N = 65,536$). We take $M = round(0.4N)$ measurements, and employ a random partial DCT matrix as the sensing matrix $\mathbf{A}$. We use the Haar wavelets as the sparsity representation basis and consider two noise conditions, the Gaussian mixture noise with $\eta = 0.9$, $\kappa = 10^3$, SNR=20 dB and Cauchy distribution noise with γ=10^{-4}. The quality of reconstructed image is measured by the peak signal to noise ratio (PSNR) refer to the original phantom. Figs. 12 and 13 show the original truth images and reconstructions of the compared YALL1 and $h_{\mathbf{B},p=2}$-APG with ℓ_1-norm penalty, LqLA-ADMM and $\ell_{\varepsilon,p=1.5}$-LqLA with $\ell_{0.5}$-norm penalty under Gaussian mixture noise and Cauchy distribution noise, respectively. It can be found that the smoothing strategies also work well on image reconstruction under these noise conditions. Here, we observe that the magnitudes of improvement by both the smoothing strategies and the nonconvex penalty are weakened as the sparsity increases from simple Shepp–Logan phantom to complicated FORBILD head phantom and then to the real MRI image.

6. Conclusion

In this paper, we mainly considered the ℓ_1-norm loss function for the residual error to deal with the sparse recovery problem under the impulsive noise condition. To solve the non-smooth problem, we proposed two smoothing strategies to transform the ℓ_1-norm loss function into a smooth counterpart with Lipschitz continuous gradient, and then adopted the APG and mAPG frameworks for the convex and non-convex regularization functions, respectively. We proved the convergence of the proposed algorithm by the theoretical proof and demonstrated its effectiveness by the numerical experiments, respectively. Moreover, the proposed algorithm is flexible and can be extended to more general recovery regularizers, such as wavelets basis, total variation and sparse dictionary, and can be expended to practical image reconstruction, such as CT and MRI. We believe that the proposed reconstruction algorithm is expected to have potential practical merits.

Declaration of Competing Interest

We declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work, there is no professional or other personal interest of any nature or kind in any product, service and/or company that could be construed as influencing the position presented in, or the review of, the manuscript entitled “A Robust Recovery Algorithm with Smoothing Strategies”.

Acknowledgment

The work was partially supported by the National Natural Science Foundation of China (61701508). The authors would like to thank the editors and anonymous reviewers for their careful reading of an earlier version of this article and constructive suggestions that improved the presentation of this work.

Appendix A. Proof of Proposition 2

Proof. The right-side inequality of (15) is obvious. By using (14), we have

$$\begin{aligned} h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y}) &= \min_{\mathbf{u}\in\mathbb{R}^M}\left\{h(\mathbf{u})+\frac{1}{2}\|\mathbf{B}(\mathbf{u}-\mathbf{Ax}+\mathbf{y})\|_2^p\right\} \\ &\le \left[h(\mathbf{u})+\frac{1}{2}\|\mathbf{B}(\mathbf{u}-\mathbf{Ax}+\mathbf{y})\|_2^p\right]_{\mathbf{u}=\mathbf{Ax}-\mathbf{y}} \\ &= h(\mathbf{Ax}-\mathbf{y}) \end{aligned} \tag{A.1}$$

For the opposite inequality, we can use the subgradient inequality for h to obtain that for every $\mathbf{z}\in\mathbb{R}^M$,

$$\begin{aligned} h_{\mathbf{B},p}(\mathbf{z}) - h(\mathbf{z}) &= \min_{\mathbf{u}\in\mathbb{R}^M}\left\{h(\mathbf{u})-h(\mathbf{z})+\frac{1}{2}\|\mathbf{B}(\mathbf{u}-\mathbf{z})\|_2^p\right\} \\ &\ge \min_{\mathbf{u}\in\mathbb{R}^M}\left\{\langle h'(\mathbf{z}),\mathbf{u}-\mathbf{z}\rangle+\frac{1}{2}\|\mathbf{B}(\mathbf{u}-\mathbf{z})\|_2^p\right\} \\ &\ge \min_{\mathbf{u}\in\mathbb{R}^M}\left\{\frac{\beta^p}{2}\|\mathbf{u}-\mathbf{z}\|_2^p+\langle h'(\mathbf{z}),\mathbf{u}-\mathbf{z}\rangle\right\} \\ &\ge \min_{\mathbf{u}\in\mathbb{R}^M}\left\{\frac{\beta^p}{2}\|\mathbf{u}-\mathbf{z}\|_2^p-L_h\|\mathbf{u}-\mathbf{z}\|_2\right\} \\ &\ge \frac{1-p}{2}\left(\frac{2}{p\beta}\right)^{\frac{p}{p-1}}(L_h)^{\frac{p}{p-1}} \end{aligned} \tag{A.2}$$

Substitute $\mathbf{z} = \mathbf{Ax}-\mathbf{y}$ in (A.2), we can obtain the left-side inequality of (15). This completes the proof. □

Appendix B. Proof of Proposition 3

Proof. By using (29), the gradient of $h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$ is

$$\partial h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})_i = \sum_m a_{mi}v_m \tag{B.1}$$

Then we have the Hessian Matrix of $h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$ is given by

$$\begin{aligned} &\nabla^2 h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})_{ij} \\ &= \frac{\partial^2 h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})}{\partial x_i\partial x_j} = \sum_m a_{mi}\frac{\partial v_m}{\partial x_j} \\ &= \frac{p}{2}d^{\frac{p}{2}-1}\left(\sum_m a_{mi}b_m^2 w_m a_{mj} + (p-2)\right. \\ &\quad\left.\times\frac{\sum_m a_{mi}b_m^2 w_m[\mathbf{Ax}-\mathbf{y}-\tilde{\mathbf{u}}]_m\sum_k a_{kj}b_k^2 w_k[\mathbf{Ax}-\mathbf{y}-\tilde{\mathbf{u}}]_k}{\|\mathbf{B}(\tilde{\mathbf{u}}-\mathbf{Ax}+\mathbf{y})\|_2^2}\right) \end{aligned} \tag{B.2}$$

where $w_m = 0$ if $\tilde{u}_m \neq 0$ and $w_m = 1$ if $\tilde{u}_m = 0$. The last equation comes from that $\frac{\partial\tilde{u}_k}{\partial x_j} = a_{kj}$ if $\tilde{u}_k \neq 0$. Then, we have

$$\nabla^2 h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y}) \preceq \begin{cases} \frac{p}{2}d^{\frac{p}{2}-1}\mathbf{A}^T\mathbf{B}^T\mathbf{BA}, & 1<p<2 \\ \frac{p(p-1)}{2}d^{\frac{p}{2}-1}\mathbf{A}^T\mathbf{B}^T\mathbf{BA}, & p\ge 2 \end{cases} \tag{B.3}$$

Substitute (26) into $d = \|\mathbf{B}(\tilde{\mathbf{u}}-\mathbf{Ax}+\mathbf{y})\|_2^2$, then we can obtain

$$d \le \sum_m b_m^2\left(\frac{2}{pb_m^2}d^{1-\frac{p}{2}}\right)^2 \le \sum_m \frac{4d^{2-p}}{p^2 b_m^2} \tag{B.4}$$

We have an upper bound for d

$$d \le \left(\sum_m \frac{4}{p^2 b_m^2}\right)^{\frac{1}{p-1}} \le \left(\frac{4M}{p^2 b_{\min}^2}\right)^{\frac{1}{p-1}} \tag{B.5}$$

If we choose $p \ge 2$ and substitute (B.5) into (B.3), we have

$$\nabla^2 h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y}) \preceq (p-1)\left(\frac{p}{2}\right)^{\frac{1}{p-1}}\left(\frac{\sqrt{M}}{b_{\min}}\right)^{\frac{p-2}{p-1}}\mathbf{A}^T\mathbf{B}^T\mathbf{BA} \tag{B.6}$$

This means that the gradient $\nabla h_{\mathbf{B},p}(\mathbf{Ax}-\mathbf{y})$ is $\rho\|\mathbf{BA}\|_2^2$-Lipschitz continuous with $\rho = (p-1)\left(\frac{p}{2}\right)^{\frac{1}{p-1}}\left(\frac{\sqrt{M}}{b_{\min}}\right)^{\frac{p-2}{p-1}}$. This completes the proof. □

Appendix C. Proof of Remark 2

Proof. Let $\alpha = \|\mathbf{BA}\|_2^2$, we first prove that $(\alpha/2)\|\mathbf{x}\|_2^2 - h_{\mathbf{B},2}(\mathbf{Ax}-\mathbf{y})$ is convex. Rewrite $(\alpha/2)\|\mathbf{x}\|_2^2 - h_{\mathbf{B}}(\mathbf{Ax}-\mathbf{y})$ as

$$\begin{aligned}
&\frac{\alpha}{2}\|\mathbf{x}\|_2^2 - h_{\mathbf{B},2}(\mathbf{Ax}-\mathbf{y}) \\
&= \frac{\alpha}{2}\|\mathbf{x}\|_2^2 - \min_{\mathbf{u}\in\mathbb{R}^M}\left\{h(\mathbf{u}) + \frac{1}{2}\|\mathbf{B}(\mathbf{u}-\mathbf{Ax}+\mathbf{y})\|_2^2\right\} \\
&= \max_{\mathbf{u}\in\mathbb{R}^M}\left\{\frac{\alpha}{2}\|\mathbf{x}\|_2^2 - h(\mathbf{u}) - \frac{1}{2}\|\mathbf{B}(\mathbf{u}-\mathbf{Ax}+\mathbf{y})\|_2^2\right\} \\
&= \frac{1}{2}\mathbf{x}^T\left(\alpha\mathbf{I} - \mathbf{A}^T\mathbf{B}^T\mathbf{BA}\right)\mathbf{x} \\
&\quad + \max_{\mathbf{u}\in\mathbb{R}^M}\left\{(\mathbf{y}+\mathbf{u})^T\mathbf{B}^T\mathbf{BAx} - \frac{1}{2}\|\mathbf{B}(\mathbf{u}+\mathbf{y})\|_2^2 - h(\mathbf{u})\right\}
\end{aligned} \tag{C.1}$$

The last term is affine in $\mathbf{x}$, and it is convex as it is the pointwise maximum of a set of convex functions. The first term is also convex when $\alpha = \|\mathbf{BA}\|_2^2$. Hence, $(\alpha/2)\|\mathbf{x}\|_2^2 - h_{\mathbf{B}}(\mathbf{Ax}-\mathbf{y})$ is convex.

Then, we use the Theorem 18.15 in [35], that is if $f \in \Gamma_0(\mathbb{R}^M)$, f is Frchet differential and ∇f is β-Lipschitz continuous if and only if $(\beta/2)\|\cdot\|_2^2 - f$ is convex. Here, let $h \in \Gamma_0(\mathbb{R}^M)$ and be coercive, then $h_{\mathbf{B},2}(\mathbf{Ax}-\mathbf{y}) \in \Gamma_0(\mathbb{R}^M)$ by using Proposition 1. Then, it follows that $\nabla h_{\mathbf{B},2}(\mathbf{Ax}-\mathbf{y})$ is $\|\mathbf{BA}\|_2^2$-Lipschitz continuous. □

References

[1] E.J. Cands, J. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information, IEEE Trans. Inf. Theory 52 (2) (2006) 489C509.
[2] G. Dong, G. Kuang, N. Wang, et al., Classification via sparse representation of steerable wavelet frames on Grassmann manifold: application to target recognition in SAR image, IEEE Trans. Image Process. 26 (6) (2017) 2892–2904.
[3] F. Chen, J. Dai, N. Hu, et al., Sparse Bayesian learning for off-grid DOA estimation with nested arrays, Digit. Signal Process. 82 (2018) 187–193.
[4] M. Lustig, D.L. Donoho, J.M. Santos, et al., Compressed sensing MRI, IEEE Signal Process. Mag. 25 (2) (2008) 72.
[5] E.J. Candes, M.B. Wakin, S.P. Boyd, Enhancing sparsity by reweighted l1 minimization, J. Fourier Anal. Appl. 14 (5–6) (2008) 877–905.
[6] Y. Sun, J. Tao, Few views image reconstruction using alternating direction method via l0-norm minimization, Int. J. Imag. Syst. Technol. 24 (3) (2014) 215–223.
[7] Y. Sun, H. Chen, J. Tao, et al., Computed tomography image reconstruction from few views via log-norm total variation minimization, Digit. Signal Process. 88 (2019) 172–181.
[8] Z. Xu, X. Chang, F. Xu, H. Zhang, L1/2 regularization: a thresholding representation theory and a fast solver, IEEE Trans. Neur. Net. Learn. Syst. 23 (7) (2012) 1013–1027.
[9] T. Li, X. Dong, H. Chen, Single image super-resolution incorporating example-based gradient profile estimation and weighted adaptive p-norm, Neurocomputing (2019), doi:10.1016/j.neucom.2019.04.051.
[10] J. Cao, S. Liu, H. Liu, et al., Sparse representation of classified patches for CS-MRI reconstruction, Neurocomputing 339 (2019) 255–269.
[11] T. Zhang, Analysis of multi-stage convex relaxation for sparse regularization, J. Mach. Learn. Res 11 (2010) 1081–1107.
[12] T. Zhang, Multi-stage convex relaxation for feature selection, Bernoulli 19 (5B) (2013) 2277–2293.
[13] Y. Lou, P. Yin, Q. He, J. Xin, Computing sparse representation in a highly coherent dictionary based on difference of l1 and l2, J. Sci. Comput. 64 (1) (2015) 178–196.
[14] Y. Lou, M. Yan, Fast l1CL2 minimization via a proximal operator, J. Sci. Comput. 74 (2) (2018) 767–785.
[15] J. Fan, R. Li, Variable selection via nonconcave penalized likelihood and its oracle properties, J. Am. Stat. Assoc. 96 (456) (2001) 1348–1360.
[16] C.H. Zhang, Nearly unbiased variable selection under minimax concave penalty, Ann. Stat. 38 (2) (2010) 894–942.
[17] I. Selesnick, Sparse regularization via convex analysis, IEEE Trans. Signal Process. 65 (17) (2017) 4481–4494.
[18] Y. Sun, H. Chen, J. Tao, Sparse signal recovery via minimax-concave penalty and l1-norm loss function, IET Signal Process. 12 (9) (2018) 1091–1098.
[19] W. Ma, H. Qu, G. Gui, et al., Maximum correntropy criterion based sparse adaptive filtering algorithms for robust channel estimation under non-gaussian environments, J. Frankl. Inst. 352 (7) (2015) 2708–2727.
[20] W. Ma, B. Chen, H. Qu, et al., Sparse least mean p-power algorithms for channel estimation in the presence of impulsive noise, Signal Image Video Process. 10 (3) (2016) 503–510.
[21] L. Bar, A. Brook, N. Sochen, N. Kiryati, Deblurring of color images corrupted by impulsive noise, IEEE Trans. Image. Process. 16 (4) (2007) 1101–1111.
[22] D.S. Pham, S. Venkatesh, Improved image recovery from compressed data contaminated with impulsive noise, IEEE Trans. Image. Process. 21 (1) (2012) 397–405.
[23] D.S. Pham, S. Venkatesh, Efficient algorithms for robust recovery of images from compressed data, IEEE Trans. Image. Process. 22 (12) (2013) 4724–4737.
[24] J.F. Yang, Y. Zhang, Alternating direction algorithms for l1-problems in compressive sensing, SIAM J. Sci. Comput. 33 (1) (2011) 250–278.
[25] F. Wen, P. Liu, Y. Liu, R.C. Qiu, W. Yu, Robust sparse recovery for compressive sensing in impulsive noise using lp-norm model fitting, Proceedings of IEEE international Conference on Acoustics Speech and Signal Processing, Shang-Hai(2016) 4643–4647.
[26] F. Wen, P. Liu, Y. Liu, R.C. Qiu, W. Yu, Robust sparse recovery in impulsive noise via lp-l1 optimization, IEEE Trans. Signal Process. 65 (1) (2017) 105–118.
[27] W. Ma, J. Duan, W. Man, et al., Robust kernel adaptive filters based on mean p-power error for noisy chaotic time series prediction, Eng. Appl. Artif. Intell. 58 (2017) 101–110.
[28] W. Ma, D. Zheng, Z. Zhang, et al., Robust proportionate adaptive filter based on maximum correntropy criterion for sparse system identification in impulsive noise environments, Signal Image Video Process. 12 (1) (2018) 117–124.
[29] P.J. Huber, Robust Statistics, Wiley, New York, 1981.
[30] F. Wen, L. Pei, Y. Yang, et al., Efficient and robust recovery of sparse signal and image using generalized nonconvex regularization, IEEE Trans. Comput. Imaging 3 (4) (2017) 566–579.
[31] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, SIAM J. Imaging Sci. 2 (1) (2009) 183–202.
[32] H. Li, Z. Lin, Accelerated proximal gradient methods for nonconvex programming, in: Proceedings of Advances in Neural Information Processing Systems, Montral, 2015, pp. 379–387.
[33] H.H. Bauschke, R. Burachik, P.L. Combettes, V. Elser, D.R. Luke, H. Wolkowicz, Fixed-point Algorithms for Inverse Problems in Science and Engineering, 49, Springer Science Business Media, 2011.
[34] J.J. Moreau, Proximit et dualit dans un espace hilbertien, Bull. Soc. Math. France 93 (1965) 273–299.
[35] H.H. Bauschke, P.L. Combettes, Convex Analysis and Monotone Operator Theory in Hilbert spaces, Springer, New York, 2011.
[36] G. Marjanovic, V. Solo, On lq optimization and matrix completion, IEEE Trans. Signal Process. 60 (11) (2012) 5714–5724.
[37] P. Gong, C. Zhang, Z. Lu, et al., A general iterative shrinkage and thresholding, Algorithm Non-convex Regul. Optim. Probl. 28 (2) (2013) 37–45.
[38] P.L. Combettes, J.C. Pesquet, Proximal Splitting Methods in Signal Processing, Springer, New York, NY, 2011, pp. 185–212.
[39] Y. Sun, X. Tan, X. Li, et al., Sparse Optimization Problem with s-difference Regularization, arXiv preprint, 2019, arXiv:1905.04474.
[40] K. Jiang, D. Sun, K.C. Toh, An inexact accelerated proximal gradient method for large scale linearly constrained convex SDP, SIAM J. Optim. 22 (3) (2011) 1042–1064.
[41] Y. Nesterov, Smooth minimization of non-smooth functions, Math. Prog. 103 (1) (2005) 127–152.
[42] A. Beck, M. Teboulle, Smoothing and first order methods: a unified framework, SIAM J. Optim. 22 (2) (2012) 557–580.
[43] D.F. Sun, K.C. Toh, Y. Yang, An efficient inexact ABCD method for least squares semidefinite programming, SIAM J. Optim. 26 (2) (2016) 1072C1100.
[44] S. Becker, J. Bobin, E.J. Cands, NESTA: A fast and accurate first-order method for sparse recovery, SIAM J. Imaging Sci. 4 (1) (2011) 1–39.
[45] R. Courant, Variational Methods for the Solution of Problems of Equilibrium and Vibrations[, Verlag nicht ermittelbar, 1943.
[46] Y. Wang, J. Yang, W. Yin, et al., A new alternating minimization algorithm for total variation image reconstruction, SIAM J. Imaging Sci. 1 (3) (2008) 248–272.
[47] Z. Tan, Y.C. Eldar, A. Beck, et al., Smoothing and decomposition for analysis sparse recovery, IEEE Trans. Signal Process. 62 (7) (2014) 1762–1774.
[48] J. Xie, A. Liao, Y. Lei, A new accelerated alternating minimization method for analysis sparse recovery, Signal Process. 145 (2018) 167–174.
[49] J. Wang, F. Zhang, J. Huang, et al., A nonconvex penalty function with integral convolution approximation for compressed sensing, Signal Process. 158 (2019) 116–128.

[50] H.A. Le Thi, T.P. Dinh, H.M. Le, X.T. Vo, DC approximation approaches for sparse optimization, Eur. J. Oper. Res. 244 (1) (2015) 26–46.
[51] B. Wen, X. Chen, T.K. Pong, A proximal difference-of-convex algorithm with extrapolation, Comput. Optim. Appl. 69 (2) (2018) 297–324.
[52] Z. Yu, F. Noo, F. Dennerlein, W. Adam, L. Gunter, H. Joachim, Simulation tools for two-dimensional experiments in x-ray computed tomography using the FORBILD head phantom, Phys. Med. Biol. 57 (13) (2012) N237.

Yuli Sun received the M.S. degree from University of Science and Technology of China, China, in 2014; Currently, he is a Ph.D. candidate in College of Electronic Science, National University of Defense Technology since 2019. His research interests cover computer vision, signal processing and remote sensing image processing.

Lin Lei received the Ph.D. degree in Information and Communication Engineering from National University of Defense Technology, Changsha, China, in 2008. She is currently an Associate Professor with the school of Electronic Science, National University of Defense Technology. Her research interests include computer vision, remote sensing image interpretation and data fusion.

Xiao Li received the B.S. degree in the electrical engineering and automation from the University of Jinan, Jinan, China, in 2015, and the M.S. degrees in control science and engineering at Xiangtan University, Xiangtan, China, in 2018. He is currently pursuing the Ph.D. degree in information and communication engineering from the National University of Defense Technology, Changsha, China. His research interests include image processing and pattern recognition, representation and dictionary learning, computational pathology applications.

Ming Li received the Bachelor degree in 2013; the M.S. degree from Central South University, Changsha, China, in 2017; Currently, he is a Ph.D. candidate in College of Electronic Science, National University of Defense Technology since 2017. His research interests cover remote sensing object detection, image retrieval and change detection.

Gangyao Kuang received the B.S. and M.S. degrees in geophysics from the Central South University of Technology, Changsha, China, in 1988 and 1991, respectively, and the Ph.D. degree in communication and information from the National University of Defense Technology, Changsha, in 1995. He is currently a Professor with the School of Electronic Science, National University of Defense Technology. His research interests include computer vision, remote sensing, synthetic aperture radar (SAR) image processing, and classification with polarimetric SAR images.