Computational Optimization and Applications (2022) 81:519–538
https://doi.org/10.1007/s10589-021-00337-9

A smoothing proximal gradient algorithm for matrix rank minimization problem

Quan Yu[1] · Xinzhen Zhang[1]

Received: 18 March 2021 / Accepted: 22 November 2021 / Published online: 9 January 2022

Abstract
In this paper, we study the low-rank matrix minimization problem, where the loss function is convex but nonsmooth and the penalty term is defined by the cardinality function. We first introduce an exact continuous relaxation, that is, both problems have the same minimizers and the same optimal value. In particular, we introduce a class of lifted stationary points of the relaxed problem and show that any local minimizer of the relaxed problem must be a lifted stationary point. In addition, we derive lower bound property for the nonzero singular values of the lifted stationary point and hence also of the local minimizers of the relaxed problem. Then the smoothing proximal gradient (SPG) algorithm is proposed to find a lifted stationary point of the continuous relaxation model. Moreover, it is shown that any accumulating point of the sequence generated by SPG algorithm is a lifted stationary point. At last, numerical examples show the efficiency of the SPG algorithm.

Keywords Low-rank approximation · Nonsmooth convex loss function · Smoothing method

Mathematics Subject Classification 15A03 · 15A83 · 90C30 · 65K05

1 Introduction

Over the last decade, finding a low-rank matrix solution to a system or low-rank matrix optimization problem has received more and more attention. Numerous optimization models and methods have been proposed in [9–10, 17, 19, 24]. In this paper, we consider the matrix rank minimization problem of form

Xinzhen Zhang
xzzhang@tju.edu.cn

Quan Yu
quanyu@tju.edu.cn

[1] School of Mathematics, Tianjin University, Tianjin 300350, China

$$\min \mathcal{F}_{l_0}(X) := f(X) + \lambda \cdot \operatorname{rank}(X) = f(X) + \lambda\|\sigma(X)\|_0, \tag{1.1}$$

where $X \in \mathbb{R}^{m\times n}$ $(n \le m)$, $\|z\|_0$ counts the number of nonzero elements of z, and $\sigma(X) := \big(\sigma_1(X), \dots, \sigma_n(X)\big)^T$ with $\sigma_1(X) \ge \dots \ge \sigma_n(X) \ge 0$ being singular values of matrix X. Furthermore, $f : \mathbb{R}^{m\times n} \to [0, +\infty)$ is convex (not necessarily smooth) and λ is a positive parameter.

One application of problem (1.1) is the low-rank matrix recovery problem [12, 13, 16, 18, 23, 25]. To solve such problem, traditional algorithms are always based on l_2 (or Frobenius)-nuclear model, that is, the loss function is a l_2-norm for vector case or Frobenius norm for matrix case, and rank(X) is relaxed as a matrix nuclear norm. However, these models are sensitive to non-Gaussian noise with outliers [11, 28–29]. To overcome this drawback, the l_1 model is considered in the problem with the outlier-resistant loss function. For example, the following loss function is considered in the low-rank matrix recovery problem

$$f(X) = \|\mathscr{A}(X) - b\|_1, \tag{1.2}$$

where the linear map $\mathscr{A} : \mathbb{R}^{m\times n} \to \mathbb{R}^p$, vector $b \in \mathbb{R}^p$ are given, and $\|z\|_1$ denotes the l_1 norm of z. Obviously, f is convex but not smooth. For simplicity, we denote $\|Z\|_1 := \|vec(Z)\|_1$, where $vec(Z)$ is the vectorization operation of a matrix Z. Based on this notation, the low-rank matrix completion problem, a special case of low-rank matrix recovery problem, the corresponding loss function $f(X)$ can be written as

$$f(X) = \|P_\Omega(X - M)\|_1,$$

where $M \in \mathbb{R}^{m\times n}$ is a known matrix, Ω is an index set which locates the observed data, P_Ω is a linear operator that extracts the entries in Ω and fills the entries not in Ω with zeros. In the robust principal component analysis (RPCA) problem [2, 6, 22, 26], the loss function $f(X)$ is adopted as

$$f(X) = \|L - X\|_1, \tag{1.3}$$

where $L \in \mathbb{R}^{m\times n}$ denotes the observed data. The RPCA problem aims to decompose the matrix L as the sum of a low-rank matrix X and a sparse matrix $E = L - X \in \mathbb{R}^{m\times n}$.

It is known that matrix rank function is nonconvex and nonsmooth. In the matrix rank minimization problem, one of the most common used convex relaxations of rank function is matrix nuclear norm. Although the methods based nuclear norm relaxation have strong theoretical guarantees, the obtained approximation solutions under certain incoherence assumptions are usually hard to satisfy in real applications [4, 5]. In other words, the nuclear norm is not a perfect approximation to the rank function.

In [1], the capped l_1 function, a continuous relaxation of l_0 function, was adopted in penalized sparse regression problem with some advantages. Furthermore, a smoothing proximal gradient (SPG) algorithm with global convergence was proposed there. More recently, such technique was applied to group sparse optimization for images recovery in [21]. It is well-known that the matrix norm can be expressed as a vector norm of the singular value vector. Motivated by these, we consider whether such SPG algorithm can be generalized from sparse regression problem to low-rank matrix minimization or not.

For this aim, let $\Phi(X)=\sum_{i=1}^{n}\phi\big(\sigma_i(X)\big)$ be a continuous relaxation of the rank function with the capped ℓ_1 function ϕ given by

$$\phi(t)=\min\{1,t/\nu\},\quad t\geq 0, \tag{1.4}$$

where $\nu>0$ is a parameter. By direct computation, $\Phi(X)\leq \operatorname{rank}(X)$. Based on $\Phi(X)$, we consider the following continuous optimization problem for solving (1.1):

$$\min \mathcal{F}(X) := f(X)+\lambda\Phi(X). \tag{1.5}$$

In this paper, we consider a continuous relaxation of rank function to solve the matrix rank minimization problem. Similar to [1], our method is also based upon the SPG algorithm. From [1], the convexity of those two relaxed functions is essential to show the global convergence and local convergence rate. However, such convexity does not hold for matrix case, shown in Sect. 2. Hence it is not trivial to extend SPG algorithm from vector case to matrix case.

Our contributions are as follows. We first present a continuous relaxation problem (1.5) of problem (1.1), which are shown to have the same global optimizers. Furthermore, the local minimizers of (1.5) is a lifted stationary point of (1.5) with an expected lower bound property of singular values. Then an SPG algorithm is proposed to get a lifted stationary point of (1.5). Moreover, it is shown that any accumulating point of the sequence generated by SPG algorithm is a lifted stationary point.

Notations. We denote $[n]=\{1,2,\dots,n\}$ and $\mathbb{D}^n=\big\{d\in\mathbb{R}^n : d_i\in\{1,2\}, i\in[n]\big\}$. The space of $m\times n$ matrices is denoted by $\mathbb{R}^{m\times n}$. For a given matrix $X\in\mathbb{R}^{m\times n}$, $\mathbb{B}_\delta(X)$ denotes the open ball centered at X with radius δ. In addition, $\mathscr{D}(x)$ denotes a diagonal matrix generated by vector x, whose dimension shall be clear from the context. Denote $\mathbb{Q}^m$ the set of $(m\times m)$-dimension unitary orthogonal matrix. Let $E_i=\mathscr{D}(e_i)$, where e_i is a unit vector whose ith entry is 1.

For any given $X,Y\in\mathbb{R}^{m\times n}$, the standard inner product of X and Y is denoted by $\langle X,Y\rangle$, that is, $\langle X,Y\rangle=\operatorname{tr}\big(XY^T\big)$, where $\operatorname{tr}(\cdot)$ denotes the trace of a matrix. The Frobenius norm of X is denoted by $\|X\|_F$, namely, $\|X\|_F=\sqrt{\operatorname{tr}\big(XX^T\big)}$. Denote $\sigma(X)=\big(\sigma_1(X),\dots,\sigma_n(X)\big)^T$ and

$$\mathcal{M}(X)=\big\{(U,V)\in\mathbb{Q}^m\times\mathbb{Q}^n : X=U\mathscr{D}(\sigma(X))V^T\big\}.$$

2 An exact continuous relaxation for (1.1)

In this section, we present some relationships between (1.1) and (1.5). Without specific explanation, Assumptions 1 and 2 are assumed throughout the paper.

Assumption 1. f is Lipschitz continuous with Lipschitz constant L_f.

Assumption 2. Positive parameter ν in (1.4) satisfies $\nu<\bar{\nu} := \lambda/L_f$.

2.1 Lifted stationary points of (1.5)

Clearly, ϕ in (1.4) can be rewritten as a DC function, i.e.,

$$\phi(t) = \frac{t}{\nu} - \max\left\{\theta_1(t), \theta_2(t)\right\}$$

with $\theta_1(t) = 0$ and $\theta_2(t) = t/\nu - 1$. Denote

$$\mathcal{D}(t) = \left\{i \in \{1,2\} : \theta_i(t) = \max\left\{\theta_1(t), \theta_2(t)\right\}\right\}. \tag{2.1}$$

Following Theorem 3.7 in [14], the Clarke subdifferential of Φ at X is given by

$$\partial\Phi(X) = \left\{ U\mathscr{D}(x)V^T : x \in \partial \sum_{i=1}^{n} \phi\big(\sigma_i(X)\big), \, (U,V) \in \mathcal{M}(X) \right\},$$

where $\partial\phi(x)$ is the Clarke subdifferential [7] of $\phi(x)$. Then we have the following definition.

Definition 2.1 We say that X is a lifted stationary point of (1.5) if there exist $d_i \in \mathcal{D}\big(\sigma_i(X)\big)$ for all $i \in [n]$ such that

$$\lambda \sum_{i=1}^{n} \theta'_{d_i}\big(\sigma_i(X)\big)E_i \in \left\{ U^T\partial f(X)V + \frac{\lambda}{\nu}\mathscr{D}\big(\partial\|\sigma(X)\|_1\big) : (U,V) \in \mathcal{M}(X)\right\}, \tag{2.2}$$

where $\sigma_i(X)$ is the ith largest singular value of X.

If (2.2) holds for all $d_i \in \mathcal{D}\big(\sigma_i(X)\big)$, $i \in [n]$, then we call X a d-stationary point.

2.2 Characterizations of lifted stationary points of (1.5)

We first show that $\mathcal{D}\big(\sigma_i(\bar{X})\big)$ for a lifted stationary point $\bar{X}$ is unique.

Proposition 2.2 *If $\bar{X}$ is a lifted stationary point of* (1.5), *then the vector $d^{\bar{X}} = \left(d_1^{\bar{X}}, \dots, d_n^{\bar{X}}\right)^T \in \prod_{i=1}^{n} \mathcal{D}\big(\sigma_i(\bar{X})\big)$ satisfying* (2.2) *is unique. In particular, for $i \in [n]$,*

$$d_i^{\bar{X}} = \begin{cases} 1 & \text{if } \sigma_i(\bar{X}) < \nu, \\ 2 & \text{if } \sigma_i(\bar{X}) \geq \nu. \end{cases} \tag{2.3}$$

Proof For case of $\sigma_i(\bar{X}) \neq \nu$, the statement in this proposition follows. Hence, it suffices to consider the index i satisfying $\sigma_i(\bar{X}) = \nu$.

Now we assume that $d_i^{\bar{X}} = 1$ by contradiction when $\sigma_i(\bar{X}) = \nu$. By (2.2), there exists $\xi(\bar{X}) \in \partial f(\bar{X})$ such that $0 = \big(\bar{U}^T\xi(\bar{X})\bar{V}\big)_{ii} + \lambda/\nu$, where $(\bar{U}, \bar{V}) \in \mathcal{M}\big(\bar{X}\big)$. Then,

$\lambda/\nu = \left|\left(\bar{U}^T\xi(\bar{X})\bar{V}\right)_{ii}\right| \leq \|\bar{U}^T\xi(\bar{X})\bar{V}\|_F = \|\xi(\bar{X})\|_F \leq L_f$. This leads to a contradiction to $\nu < \lambda/L_f$. Then, we can assert that $d_i^{\bar{X}} = 2$, and hence (2.3) holds for $\sigma_i(\bar{X}) = \nu$.

For a given $d = \left(d_1, \dots, d_n\right)^T \in \mathbb{D}^n$, we define

$$\Phi^d(X) := \sum_{i=1}^{n} \sigma_i(X)/\nu - \sum_{i=1}^{n} \theta_{d_i}\left(\sigma_i(X)\right). \tag{2.4}$$

It is easy to see that

$$\Phi(X) = \min_{d\in\mathbb{D}^n} \Phi^d(X),\ \forall X \in \mathbb{R}^{m\times n}.$$

Furthermore, for a fixed $\bar{X}$, $\Phi(\bar{X}) = \Phi^{d^{\bar{X}}}(\bar{X})$ with $d^{\bar{X}}$ defined in (2.3). We next show that any local minimizer of (1.5) is a lifted stationary point of the problem.

Theorem 2.3 *Suppose that $\bar{X}$ is a local minimizer of problem* (1.5). *Then $\bar{X}$ is a lifted stationary point of* (1.5), *that is*, (2.2) *holds at $\bar{X}$.*

Proof Since $\bar{X}$ is a local minimizer of (1.5), it gives

$$\begin{aligned} f(\bar{X}) + \lambda\Phi^{d^{\bar{X}}}\left(\bar{X}\right) &= f(\bar{X}) + \lambda\Phi\left(\bar{X}\right) \\ &\leq f(X) + \lambda\Phi(X) \leq f(X) + \lambda\Phi^{d^{\bar{X}}}(X), \quad \forall X \in \mathbb{B}_\varrho(\bar{X}), \end{aligned} \tag{2.5}$$

where the first equality comes from $\Phi^{d^{\bar{X}}}(\bar{X}) = \Phi(\bar{X})$, $d^{\bar{X}}$ is defined as in (2.3) and the last inequality is due to $\Phi^{d^{\bar{X}}}(X) \geq \Phi(X)$, $\forall X \in \mathbb{R}^{m\times n}$. Then $\bar{X}$ is a local minimizer of the problem

$$\min_X f(X) + \lambda\Phi^{d^{\bar{X}}}(X). \tag{2.6}$$

Hence, there exists some $(\bar{U}, \bar{V}) \in \mathcal{M}\left(\bar{X}\right)$ such that

$$0 \in \partial f\left(\bar{X}\right) + \lambda\bar{U}\left(\frac{1}{\nu}\mathscr{D}\left(\partial\|\sigma(\bar{X})\|_1\right) - \sum_{i=1}^{n}\theta'_{d_i^{\bar{X}}}\left(\sigma_i(\bar{X})\right)E_i\right)\bar{V}^T, \tag{2.7}$$

which implies (2.2) at $\bar{X}$.

To end this subsection, we present a lower bound property of the lifted stationary points of (1.5), which is similar to Lemma 2.3 in [1]. For the ease of the reader, we present the proof here.

Lemma 2.4 *If $\bar{X}$ is a lifted stationary point of* (1.5), *then it holds that*

$$\sigma_i(\bar{X}) \in [0, \nu) \Rightarrow \sigma_i(\bar{X}) = 0, \quad i \in [n]. \tag{2.8}$$

Proof Suppose $\bar{X}$ is a lifted stationary point of (1.5). Assume that $\sigma_i(\bar{X}) \in (0, \nu)$ for some $i \in [n]$. Then, $d_i^{\bar{X}} = 1$. By Definition 2.1, there exists $\xi(\bar{X}) \in \partial f(\bar{X})$ such that

$0=\left(\bar{U}^T\xi(\bar{X})\bar{V}\right)_{ii}+\lambda/\nu$, where $(\bar{U},\bar{V})\in\mathcal{M}\left(\bar{X}\right)$. Then, $\lambda/\nu=\left|\left(\bar{U}^T\xi(\bar{X})\bar{V}\right)_{ii}\right|\le\|\bar{U}^T\xi(\bar{X})\bar{V}\|_F=\|\xi(\bar{X})\|_F\le L_f$, which leads to a contradiction to $\nu<\lambda/L_f$. Thus, for any $i\in[n]$, $\sigma_i(\bar{X})\in[0,\nu)$ implies that $\sigma_i(\bar{X})=0$.

2.3 Relationships between (1.1) and (1.5)

This subsection presents some relationships between problem (1.1) and its continuous relaxation (1.5). According to the lower bound property of the lifted stationary points of (1.5) obtained in Lemma 2.4, we are ready to link (1.1) and (1.5) in the following two results. The first result deals with the global minimizers of two problems and the second one with their local minimizers.

Theorem 2.5 *$\bar{X}$ is a global minimizer of (1.1) if and only if it is a global minimizer of (1.5). Moreover, problems (1.1) and (1.5) have the same optimal value.*

Proof Let $\bar{X}$ be a global minimizer of (1.5), then $\bar{X}$ is a lifted stationary point of (1.5) from Theorem 2.3. By (2.8), it follows $\Phi(\bar{X})=\|\sigma(\bar{X})\|_0$. Then,

$$\begin{aligned}f(\bar{X})+\lambda\|\sigma(\bar{X})\|_0 &= f(\bar{X})+\lambda\Phi(\bar{X})\\ &\le f(X)+\lambda\Phi(X)\\ &\le f(X)+\lambda\|\sigma(X)\|_0,\end{aligned}$$

where the last inequality comes from $\Phi(X)\le\|\sigma(X)\|_0$, $\forall X\in\mathbb{R}^{m\times n}$. Thus, $\bar{X}$ is a global minimizer of (1.1).

Next, suppose $\bar{X}$ is a global minimizer of (1.1) but not a global minimizer of (1.5). Assume that $\hat{X}$ is a global minimizer of (1.5) satisfying

$$f(\hat{X})+\lambda\Phi(\hat{X})<f(\bar{X})+\lambda\Phi(\bar{X}).$$

As shown earlier, $\Phi(\hat{X})=\|\sigma(\hat{X})\|_0$. Together with $\Phi(\bar{X})\le\|\sigma(\bar{X})\|_0$, we have $f(\hat{X})+\lambda\|\sigma(\hat{X})\|_0<f(\bar{X})+\lambda\|\sigma(\bar{X})\|_0$, which leads to a contradiction. Thus, any global minimizer of (1.1) must be a global minimizer of (1.5).

The same optimal value is clear from above.

Theorem 2.5 ensures that problems (1.1) and (1.5) have the same optimal value. The following result will show the relationship between the local minimizers of problems (1.1) and those of (1.5).

Proposition 2.6 *If $\bar{X}$ is a local minimizer of (1.5), then it is a local minimizer of (1.1), and the objective functions of (1.1) and (1.5) have the same value at $\bar{X}$, i.e., $\mathcal{F}_{\ell_0}(\bar{X})=\mathcal{F}(\bar{X})$.*

The proof is similar to the first part of Theorem 2.5 and hence we omit it here.

Note that the convexity of $\Phi^d(\cdot)$ for any given d is essential for further relationships between the stationary point (2.2) and (1.5) as in Fig. 1 in [1]. However, the

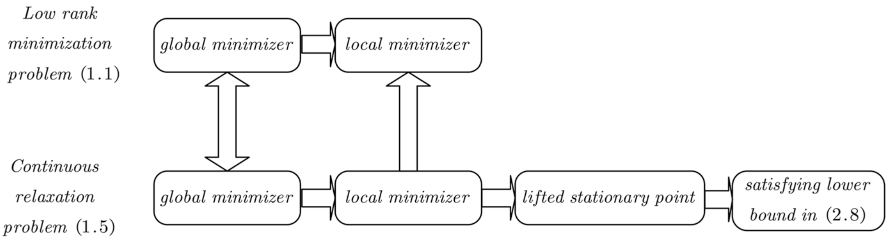
Low rank minimization problem (1.1)
global minimizer
local minimizer
Continuous relaxation problem (1.5)
global minimizer
local minimizer
lifted stationary point
satisfying lower bound in (2.8)

Fig. 1 Relationships between problems (1.1) and (1.5)

following counterexample illustrates the non-convexity of $\Phi^d(X)$, and hence no further relationships in [1] can be established.

Example 2.7 Let $d = (2, 1)^T$ and $X = [1, 0;0, 0]$, $Y = [0, 0;0, 2]$. Then for any $0 < \lambda < 1$, it follows

$$\Phi^d(\lambda X + (1 - \lambda)Y) = 1 + \frac{1}{\nu} \min\{\lambda, 2(1 - \lambda)\} > \lambda\Phi^d(X) + (1 - \lambda)\Phi^d(Y) = 1.$$

Hence, it is asserted that $\Phi^d(X)$ for such d is not convex.

To end this subsection, we present Fig. 1 to summarize the relationships between problems (1.1) and (1.5).

3 Numerical algorithm and its convergence analysis

In this section, we are ready to propose a numerical algorithm to find a lifted stationary point of (1.5). For this aim, we first introduce some preliminary results on smoothing methods and the proximal gradient algorithm, then we propose a proximal gradient algorithm based on the smoothing method. After that we establish the convergence of the proposed algorithm.

3.1 Smoothing approximation method and proximal gradient method

Smoothing approximation method is a common used numerical method for nonsmooth optimization problems. For more details, see [20] and references therein. We first recall a class of smoothing functions for $f(X)$ in (1.5).

Definition 3.1 We call $\tilde{f} : \mathbb{R}^{m\times n} \times [0, \bar{\mu}] \rightarrow \mathbb{R}$ with $\bar{\mu} > 0$ a smoothing function of the convex function f in (1.5), if $\tilde{f}(\cdot, \mu)$ is continuously differentiable in $\mathbb{R}^{m\times n}$ for any fixed $\mu > 0$ and satisfies the following conditions:

(i) $\lim_{X\rightarrow\bar{X},\mu\downarrow 0} \tilde{f}(X, \mu) = f(\bar{X}),\ \forall \bar{X} \in \mathbb{R}^{m\times n}$;

(ii) (convexity) $\tilde{f}(X,\mu)$ is convex with respect to X for any fixed $\mu > 0$;

(iii) (gradient consistency) $\left\{ \lim_{Z\to X,\mu\downarrow 0} \nabla_Z \tilde{f}(Z,\mu) \right\} \subseteq \partial f(X),\ \forall X \in \mathbb{R}^{m\times n}$;

(iv) ($\tilde{f}(X,\cdot)$ Lipschitz continuity with respect to μ) there exists a positive constant η such that

$$\left|\tilde{f}\big(X,\mu_2\big) - \tilde{f}\big(X,\mu_1\big)\right| \le \eta\left|\mu_1 - \mu_2\right|,\ \forall X \in \mathbb{R}^{m\times n},\ \mu_1,\mu_2 \in [0,\bar{\mu}];$$

(v) ($\nabla_X \tilde{f}(\cdot,\mu)$ Lipschitz continuity with respect to X) there exists a constant $L > 0$ such that for any $\mu \in (0,\bar{\mu}]$, $\nabla_X \tilde{f}(\cdot,\mu)$ is Lipschitz continuous with Lipschitz constant $L\mu^{-1}$.

Let $\tilde{f}(X,\mu)$ be a smoothing function of $f(X)$ in (1.5). For convenience of notation, the gradient of $\tilde{f}(X,\mu)$ with respect to X is denoted as $\nabla\tilde{f}(X,\mu)$. Furthermore, Definition 3.1-(iv) indicates that

$$|\tilde{f}(X,\mu) - f(X)| \le \eta\mu, \quad \forall X \in \mathbb{R}^{m\times n},\ 0 < \mu \le \bar{\mu}. \tag{3.9}$$

Some smoothing functions of the l_1 loss function in (1.2) can be found in Example 3.1 of [1] and we omit it here for simplicity.

Some notations are listed as follows

$$\tilde{\mathcal{F}}^d(X,\mu) \triangleq \tilde{f}(X,\mu) + \lambda\Phi^d(X) \quad \text{and} \quad \tilde{\mathcal{F}}(X,\mu) \triangleq \tilde{f}(X,\mu) + \lambda\Phi(X),$$

where $\tilde{f}$ is a smoothing function of f, $\mu > 0$ and $d \in \mathbb{D}^n$. For any fixed $\mu > 0$ and $d \in \mathbb{D}^n$, both $\tilde{\mathcal{F}}^d(X,\mu)$ and $\tilde{\mathcal{F}}(X,\mu)$ are nonconvex, which are different from the vector case considered in [1]. Moreover,

$$\tilde{\mathcal{F}}^d(X,\mu) \ge \tilde{\mathcal{F}}(X,\mu), \quad \forall d \in \mathbb{D}^n,\ X \in \mathbb{R}^{m\times n},\ \mu \in (0,\bar{\mu}]. \tag{3.10}$$

Next we are ready to recall some preliminaries on proximal gradient method.

Similar to the analysis in Subsection 3.2 of [1], we have a closed-form solution to proximal operator of $\tau\Phi^d$ as follows.

Lemma 3.2 *For any given vectors $d \in \mathbb{D}^n$, $w \in \mathbb{R}^n_+$, and a positive number $\tau > 0$, the proximal operator of $\boldsymbol{prox}_{\tau\Phi^d}(w)$ has a closed-form solution, i.e.,*

$$\hat{x} = \boldsymbol{prox}_{\tau\Phi^d}(w) := \arg\min_{x\in\mathbb{R}^n_+} \left\{ \tau\Phi^d(\mathscr{D}(x)) + \frac{1}{2}\|x - w\|_F^2 \right\} \tag{3.11}$$

can be calculated by

$$\hat{x}_i = \max\{\bar{w}_i - \tau/\nu, 0\}, \quad i \in [n], \tag{3.12}$$

where

$$\bar{w}_i = \begin{cases} w_i & \text{if } d_i = 1, \\ w_i + \frac{\tau}{\nu} & \text{if } d_i = 2. \end{cases}$$

Theorem 3.3 *For given $W \in \mathbb{R}^{m\times n}$ and $\tau > 0$, let $U\mathscr{D}(w)V^T$ be the singular value decomposition of W and $\hat{x} = \boldsymbol{prox}_{\tau\Phi^d}(w)$. Here $d := d^W$ is defined as in (2.3). Then $\hat{x}_1 \geq \hat{x}_2 \geq \ldots \geq \hat{x}_n \geq 0$ and $\hat{X} = U\mathscr{D}(\hat{x})V^T$ is an optimal solution of the problem*

$$\min_X \left\{ \tau\Phi^d(X) + \frac{1}{2}\|X - W\|_F^2 \right\}. \tag{3.13}$$

Proof From $w_1 \geq w_2 \geq \ldots \geq w_n \geq 0$, it is clear that $d_1 \geq d_2 \geq \ldots \geq d_n$. Next, we will prove that $\hat{x}_1 \geq \hat{x}_2 \geq \ldots \geq \hat{x}_n \geq 0$. We split the proof into three cases.
Case 1. $d_i = d_{i+1} = 2$. By (3.12), it holds

$$\hat{x}_i = w_i \geq w_{i+1} = \hat{x}_{i+1} \geq 0.$$

Case 2. $d_i = 2$ and $d_{i+1} = 1$. By (3.12), it holds

$$\hat{x}_i = w_i \geq \max\left\{w_{i+1} - \tau/\nu, 0\right\} = \hat{x}_{i+1} \geq 0.$$

Case 3. $d_i = d_{i+1} = 1$. By (3.12), it holds

$$\hat{x}_i = \max\left\{w_i - \tau/\nu, 0\right\} \geq \max\left\{w_{i+1} - \tau/\nu, 0\right\} = \hat{x}_{i+1} \geq 0.$$

Combining with all cases, the non-increasing of $\hat{x}_i$ is asserted.

Invoking by [15, Proposition 2.1] with $F(X) = \tau\Phi^d(X)$, $\phi(t) = t^2/2, \|\cdot\| = \|\cdot\|_F$ and using the fact that $\hat{x}$ is an optimal solution of $\boldsymbol{prox}_{\tau\Phi^d}(w)$, it is concluded that $\hat{X} = U\mathscr{D}(\hat{x})V^T$ is an optimal solution of (3.13).

To end this subsection, we introduce the first order approximation of $\tilde{\mathcal{F}}^d(\cdot, \mu)$ on a given matrix Z

$$Q_{d,\gamma}(X, Z, \mu) = \tilde{f}(Z, \mu) + \langle X - Z, \nabla\tilde{f}(Z, \mu)\rangle + \frac{1}{2}\gamma\mu^{-1}\|X - Z\|_F^2 + \lambda\Phi^d(X) \tag{3.14}$$

with a constant $\gamma > 0$. Then, the optimization problem $\min_X Q_{d,\gamma}(X, Z, \mu)$ has a closed form, denoted by $\hat{X}$, which can be calculated by using Theorem 3.3 with $\tau = \lambda\gamma^{-1}\mu$ and $W = Z - \gamma^{-1}\mu\nabla\tilde{f}(Z, \mu)$.

3.2 SPG algorithm

In this subsection, a proximal gradient algorithm based on the smoothing method, denoted by SPG for simplicity, will be proposed to get a lifted stationary point of (1.5).

The following assumptions are needed to analyze the convergence of the proposed SPG algorithm:

- (A1) Assumption 1 and Assumption 2 hold;
- (A2) $\tilde{f}$ is a smoothing function of f defined in Definition 3.1;
- (A3) the global minimum point of $\mathcal{F}$ in (1.5) (or $\mathcal{F}_{\ell_0}$ in (1.1)) is bounded.

Borrowing from L_f in Assumption 1, ν can be defined such that problems (1.1) is consistent with (1.5) in Theorem 2.5 and Proposition 2.6. Based upon the above assumptions, the SPG algorithm for solving (1.5) is outlined as Algorithm 1 here.

```
**Algorithm 1** (SPG algorithm).
**Input:** Let $X^0 \in \mathbb{R}^{m\times n}$ and $\mu_{-1} = \mu_0 \in (0, \bar{\mu}]$. Choose $\rho > 1$, $\sigma \in (0.5, 1)$, $\alpha > 0$,
    and $0 < \underline{\gamma} \le \bar{\gamma}$. Set $k := 0$ and $l := 0$.
    **while** not converge **do**
        *Step* 1. Choose $\gamma_k \in [\underline{\gamma}, \bar{\gamma}]$ and let $d^k := d^{X^k}$, where $d^{X^k}$ is defined in (2.3).
        *Step* 2. 2a) Compute
(3.15)            $\hat{X}^{k+1} = \arg\min_X Q_{d^k,\gamma_k}\left(X, X^k, \mu_k\right)$.
                  2b) If $\hat{X}^{k+1}$ satisfies
(3.16)            $\tilde{\mathcal{F}}^{d^k}\left(\hat{X}^{k+1}, \mu_k\right) \le Q_{d^k,\gamma_k}\left(\hat{X}^{k+1}, X^k, \mu_k\right) - \frac{1}{4}\gamma_k\mu_k^{-1}\left\|\hat{X}^{k+1} - X^k\right\|_F^2$,
        let
(3.17)            $X^{k+1} := \hat{X}^{k+1}$
        and go to *Step* 3. Otherwise, $\gamma_k := \rho\gamma_k$, and return to Step 2a).
        *Step* 3. If
(3.18)            $\tilde{\mathcal{F}}\left(X^{k+1}, \mu_k\right) + \eta\mu_k - \tilde{\mathcal{F}}\left(X^k, \mu_{k-1}\right) - \eta\mu_{k-1} \le -\alpha\mu_k^2$,
        set $\mu_{k+1} = \mu_k$; otherwise, set
(3.19)            $\mu_{k+1} := \frac{\mu_0}{(k+1)^\sigma}$,
        and
(3.20)            $Y^{l+1} = X^{k+1}, \quad l := l + 1$.
        Let $k := k + 1$ and go to *Step* 1.
    **end while**
**Output:** $Y^l$.
```

At each iteration, we adopt the proximal gradient algorithm to solve $\min_X Q_{d^k,\gamma_k}(X, X^k, \mu_k)$ for fixed μ_k, γ_k, and d^k. At each iteration in Step 1, we choose γ_k independently. The smoothing parameter μ_k is updated in Step 3 by (3.18), where $\tilde{\mathcal{F}}\big(X^{k+1}, \mu_k\big) + \eta\mu_k$ can be seen as loss function with monotone nonincreasing property, which can be seen from Lemma 3.5. If the loss function decreases more than a given scale, then the smoothing parameter μ_k is still acceptable; otherwise, it is updated by (3.19).

For convenience, let

$$\mathcal{N}^s = \left\{k \in \mathbb{N} : \mu_{k+1} \neq \mu_k\right\},$$

and n_r^s be the rth smallest number in $\mathcal{N}^s$. Clearly, $Y^l \in \left\{X^k\right\}_{k\in\mathcal{N}^s}$ for all l. Then, we can update $\left\{\mu_k\right\}$ by

$$\mu_k = \mu_{n_r^s+1} = \frac{\mu_0}{\left(n_r^s + 1\right)^{\sigma}}, \quad \forall n_r^s + 1 \le k \le n_{r+1}^s, \tag{3.21}$$

which is helpful to show Lemma 3.6.

3.3 Convergence analysis

In this subsection, we will present the convergence analysis for the SPG algorithm.

Let $\left\{Y^l\right\}$ be the sequences generated by SPG algorithm. We first show some basic properties of the iterates $\left\{X^k\right\}$, $\left\{Y^l\right\}$, $\left\{\gamma_k\right\}$ and $\left\{\mu_k\right\}$ in Lemmas 3.4–3.6. Next, we show that any accumulating point of $\left\{Y^l\right\}$ is a lifted stationary point of (1.5), as stated in Theorem 3.7.

Lemma 3.4 *In the SPG algorithm,* $\{X^k\}$ *is well defined and* $\left\{\gamma_k\right\} \subseteq [\underline{\gamma}, \max\{\bar{\gamma}, 2\rho L\}]$.

Proof Clearly, (3.16) holds if and only if

$$\tilde{f}\left(\hat{X}^{k+1}, \mu_k\right) \le \tilde{f}\left(X^k, \mu_k\right) + \left\langle \nabla\tilde{f}\left(X^k, \mu_k\right), \hat{X}^{k+1} - X^k \right\rangle + \frac{1}{4}\gamma_k\mu_k^{-1}\left\|\hat{X}^{k+1} - X^k\right\|_F^2.$$

From Definition 3.1-(v), (3.16) holds if $\gamma_k \ge 2L$. Thus, γ_k in Step 2 is attainable by at most $\log_\rho(2L/\underline{\gamma}) + 1$ times at each iteration. Hence, it is asserted that the SPG algorithm is well-defined, and hence $\gamma_k \le \max\{\bar{\gamma}, 2\rho L\}$, $\forall k \in \mathbb{N}$.

Lemma 3.5 *For any* $k \in \mathbb{N}$, *we have*

$$\tilde{\mathcal{F}}\left(X^{k+1}, \mu_k\right) \le \tilde{\mathcal{F}}\left(X^k, \mu_k\right) - \frac{1}{4}\gamma_k\mu_k^{-1}\left\|X^{k+1} - X^k\right\|_F^2. \tag{3.22}$$

Furthermore, $\left\{\tilde{\mathcal{F}}\left(X^{k+1}, \mu_k\right) + \eta\mu_k\right\}$ *is nonincreasing.*

Proof By (3.15), it follows

$$Q_{d^k,\gamma_k}\left(X^{k+1}, X^k, \mu_k\right) \le Q_{d^k,\gamma_k}\left(X^k, X^k, \mu_k\right) = \tilde{f}\left(X^k, \mu_k\right) + \lambda\Phi^{d^k}\left(X^k\right).$$

From the definition of $Q_{d^k,\gamma_k}\left(X^{k+1}, X^k, \mu_k\right)$ in (3.14), it can be rewritten as

$$\lambda\Phi^{d^k}\left(X^{k+1}\right) \le \lambda\Phi^{d^k}(X) + \left\langle X^k - X^{k+1}, \nabla\tilde{f}\left(X^k, \mu_k\right)\right\rangle - \frac{1}{2}\gamma_k\mu_k^{-1}\left\|X^{k+1} - X^k\right\|_F^2. \tag{3.23}$$

Furthermore, we can write (3.16) as

$$\begin{aligned}\tilde{\mathcal{F}}^{d^k}\left(X^{k+1}, \mu_k\right) \le& \tilde{f}\left(X^k, \mu_k\right) + \left\langle X^{k+1} - X^k, \nabla\tilde{f}\left(X^k, \mu_k\right)\right\rangle \\ &+ \frac{1}{4}\gamma_k\mu_k^{-1}\left\|X^{k+1} - X^k\right\|_F^2 + \lambda\Phi^{d^k}\left(X^{k+1}\right).\end{aligned} \tag{3.24}$$

Summing up (3.23) and (3.24), there holds

$$\begin{aligned}\tilde{\mathcal{F}}^{d^k}\left(X^{k+1},\mu_k\right) \leq & \tilde{f}\left(X^k,\mu_k\right)+\lambda\Phi^{d^k}(X)-\frac{1}{4}\gamma_k\mu_k^{-1}\left\|X^{k+1}-X^k\right\|_F^2 \\ = & \tilde{\mathcal{F}}^{d^k}\left(X^k,\mu_k\right)-\frac{1}{4}\gamma_k\mu_k^{-1}\left\|X^{k+1}-X^k\right\|_F^2 \\ = & \tilde{\mathcal{F}}\left(X^k,\mu_k\right)-\frac{1}{4}\gamma_k\mu_k^{-1}\left\|X^{k+1}-X^k\right\|_F^2,\end{aligned} \tag{3.25}$$

where the first "=" comes from the definition of $\tilde{\mathcal{F}}^{d^k}$ and the second "=" is due to $\Phi^{d^k}\left(X^k\right)=\Phi(X^k)$. Furthermore, $\tilde{\mathcal{F}}^{d^k}\left(X^{k+1},\mu_k\right)\geq\tilde{\mathcal{F}}\left(X^{k+1},\mu_k\right)$ from (3.10). Therefore, (3.22) is arrived.

Since $\tilde{\mathcal{F}}(X,\mu)=\tilde{f}(X,\mu)+\lambda\Phi(X)$, it is clear that

$$\tilde{\mathcal{F}}(X,\mu_k)-\tilde{\mathcal{F}}(X,\mu_{k-1})=\tilde{f}(X,\mu_k)-\tilde{f}(X,\mu_{k-1})\leq\eta\left(\mu_{k-1}-\mu_k\right),$$

where the last inequality comes from Definition 3.1 (iv). Together with (3.22), there holds

$$\begin{aligned}\tilde{\mathcal{F}}\left(X^{k+1},\mu_k\right)+\eta\mu_k\leq & \tilde{\mathcal{F}}\left(X^k,\mu_k\right)+\eta\mu_k-\frac{1}{4}\gamma_k\mu_k^{-1}\left\|X^{k+1}-X^k\right\|_F^2 \\ \leq & \tilde{\mathcal{F}}\left(X^k,\mu_{k-1}\right)+\eta\mu_{k-1}-\frac{1}{4}\gamma_k\mu_k^{-1}\left\|X^{k+1}-X^k\right\|_F^2,\end{aligned} \tag{3.26}$$

which implies the nonincreasing property of $\left\{\tilde{\mathcal{F}}\left(X^{k+1},\mu_k\right)+\eta\mu_k\right\}$.

Lemma 3.6 *Suppose that $\{Y^l\}$, together with $\{X^k,\mu_k,\gamma_k\}$, is generated by the SPG algorithm. Then*

(i) $\sum\limits_{k=0}^{\infty}\mu_k^2\leq\Lambda$ *with* $\Lambda=\frac{1}{\alpha}\left(\tilde{\mathcal{F}}\left(X^0,\mu_{-1}\right)+\eta\mu_{-1}-\min\mathcal{F}(X)\right)+\frac{2\mu_0^2\sigma}{2\sigma-1}<\infty;$

(ii) $\lim\limits_{k\to\infty}\mu_k=0;$

(iii) $\{Y^l\}$ *is an infinite sequence.*

Proof (i) From (3.21), one has

$$\sum_{k\in\mathcal{N}^s}\mu_k^2=\sum_{r=1}^{\infty}\frac{\mu_0^2}{\left(n_r^s+1\right)^{2\sigma}}\leq\sum_{k=1}^{\infty}\frac{\mu_0^2}{k^{2\sigma}}\leq\frac{2\mu_0^2\sigma}{2\sigma-1}, \tag{3.27}$$

where n_r^s is the rth smallest element in $\mathcal{N}^s$.

By (A3) and (3.9), we see that

$$\tilde{\mathcal{F}}\left(X^{k+1},\mu_k\right)+\eta\mu_k\geq\mathcal{F}\left(X^{k+1}\right)\geq\min\mathcal{F}(X)=\min\mathcal{F}_{\ell_0}(X)>-\infty, \tag{3.28}$$

where the equality follows from Theorem 2.5. When $k\notin\mathcal{N}^s$, (3.18) can be rewritten as

$$\alpha\mu_k^2\leq\tilde{\mathcal{F}}\left(X^k,\mu_{k-1}\right)+\eta\mu_{k-1}-\tilde{\mathcal{F}}\left(X^{k+1},\mu_k\right)-\eta\mu_k.$$

Together with the nonincreasing property of $\left\{\tilde{\mathcal{F}}\left(X^{k+1},\mu_k\right)+\eta\mu_k\right\}$ and (3.28), we have that

$$\sum_{k\notin\mathcal{N}^s}\mu_k^2\leq\frac{1}{\alpha}\left(\tilde{\mathcal{F}}\left(X^0,\mu_{-1}\right)+\eta\mu_{-1}-\min\mathcal{F}(X)\right). \tag{3.29}$$

Summing up (3.27) and (3.29), (i) is arrived here.

(ii) From (i), (ii) is obvious.

(iii) (iii) is clear from (ii).

Now we present the convergence of SPG algorithm, stated as follows.

Theorem 3.7 *Any accumulation point of* $\left\{Y^l\right\}$ *generated by SPG algorithm is a lifted stationary point of* (1.5).

Proof Suppose that $\bar{Y}$ is an accumulation point of any convergence subsequent $\left\{Y^{l_i}\right\}_{l_i\in\mathcal{N}^s}$. For convenience, denote $X^{k_i}=Y^{l_i}$. Since (3.18) fails for k_i, by rearranging (3.26), one has

$$4\alpha\mu_{k_i-1}^2\geq\gamma_{k_i-1}\mu_{k_i-1}^{-1}\left\|X^{k_i}-X^{k_i-1}\right\|_F^2. \tag{3.30}$$

Combining $\lim\limits_{i\to\infty}\mu_{k_i}=0$ and $\left\{\gamma_{k_i}\right\}\subseteq[\underline{\gamma},\max\{\bar{\gamma},2\rho L\}]$, it follows

$$\lim_{i\to\infty}\mu_{k_i-1}^{-1}\left\|X^{k_i}-X^{k_i-1}\right\|_F=0\text{ and }\lim_{i\to\infty}X^{k_i-1}=\lim_{i\to\infty}X^{k_i}=\bar{Y}. \tag{3.31}$$

Considering $X^{k_i}=\hat{X}^{k_i}$ defined as in (3.15), the first-order optimality condition of (3.15) gives

$$\nabla\tilde{f}\left(X^{k_i-1},\mu_{k_i-1}\right)+\gamma_{k_i-1}\mu_{k_i-1}^{-1}\left(X^{k_i}-X^{k_i-1}\right)+\lambda\zeta^{k_i-1}=0,\forall\zeta^{k_i-1}\in\partial\Phi^{d^{k_i-1}}\left(X^{k_i}\right). \tag{3.32}$$

From the fact that $\left\{d^{k_i}:i\in\mathbb{N}\right\}$ is finite and $\lim\limits_{i\to\infty}X^{k_i}=\bar{Y}$, there exists a subsequence of $\left\{k_i\right\}$, denoted as $\left\{k_{i_j}\right\}$, and $\bar{d}\in\mathcal{D}(\sigma(\bar{X}))$ such that $d^{k_{i_j}}=\bar{d}$, $\forall j\in\mathbb{N}$. By the definition of $\partial\Phi^{\bar{d}}$ and $\lim\limits_{j\to\infty}X^{k_{i_j}}=\bar{Y}$, it gives

$$\left\{\lim_{j\to\infty}\zeta^{k_{i_j}}:\zeta^{k_{i_j}}\in\partial\Phi^{d^{k_{i_j}}}\left(X^{k_{i_j}+1}\right)\right\}\subseteq\partial\Phi^{\bar{d}}(\bar{Y}). \tag{3.33}$$

Letting $k_{i_j}\to\infty$ in (3.32), from Definition 3.1-(iii), (3.31) and (3.33), then there exist $\bar{\xi}\in\partial f(\bar{Y})$ and $\bar{\zeta}^{\bar{d}}\in\partial\Phi^{\bar{d}}(\bar{Y})$ such that

$$\bar{\xi}+\lambda\bar{\zeta}^{\bar{d}}=0. \tag{3.34}$$

By $\bar{d}\in\prod_{i=1}^n\mathcal{D}\left(\sigma_i(\bar{Y})\right)$ and the definition of $\Phi^{\bar{d}}$ in (2.4), (3.34) indicates that $\bar{Y}$ is a lifted stationary point of (1.5).

Remark 3.8 Let $X = diag(x)$ and $Z = Diag(z)$ be diagonal matrices generated by x and z, respectively. Then $\Phi^d(X)$ and $Q_{d,\gamma}(X, Z, \mu)$ equal to $\Phi^d(x)$ and $Q_{d,\gamma}(x, z, \mu)$, as defined in [1], which are convex on x for fixed d, z, γ, μ, respectively. The convexity of $\Phi^d(x)$ and $Q_{d,\gamma}(x, z, \mu)$ is essential to establish the global convergence and local convergence rate. However, $\Phi^d(X)$ and $Q_{d,\gamma}(X, Z, \mu)$ on X are not convex, see Example 2.7. So the global convergence and local convergence rate can not be established in a similar way in [1].

4 Numerical experiments

In this section we conduct numerical experiments to test the performance of the SPG method. In particular, we apply it to solve the problem (1.1) with $f(X) = \|P_\Omega(X - M)\|_1$, that is,

$$\min_{X \in \mathbb{R}^{m \times n}} \mathcal{F}_{l_0}(X) := \|P_\Omega(X - M)\|_1 + \lambda \cdot \operatorname{rank}(X). \tag{4.35}$$

We conduct extensive experiments to evaluate our method and then compare it with some existing methods, including FPCA [17], SVT [3] and VBMFL1 [28]. The platform is Matlab R2014a under Windows 10 on a desktop of a 3.2GHz CPU and 8GB memory. We adopt the root-mean-square error (RMSE) as an evaluation metric

$$RMSE := \sqrt{\frac{\|X^* - M\|_F^2}{mn}},$$

and the final performance of each simulation is evaluated by obtaining an ensemble average of the relative error with T independent Monte Carlo runs.

In the simulation, a typical two-component Gaussian mixture model (GMM) is used as the non-Gaussian noise model. The probability density function (PDF) of GMM is defined as

$$p_v(i) = (1 - c)N\left(0, \sigma_A^2\right) + cN\left(0, \sigma_B^2\right),$$

where $N\left(0, \sigma_A^2\right)$ represents the noise disturbance with variance σ_A^2, and $N\left(0, \sigma_B^2\right)$ stands for outliers that occur occasionally with a large variance σ_B^2. The parameter c controls the occurrence probability of outliers.

4.1 Random matrix completion

In this subsection, we aim to recover a random matrix $M \in \mathbb{R}^{m \times n}$ of rank r from known entries $\left\{M_{ij}\right\}_{(i,j)\in\Omega}$. In detail, we first generate random matrices $M_L = unifrnd(-0.1, 0.3, m, r) \in \mathbb{R}^{m \times r}$ and $M_R = unifrnd(-0.1, 0.3, n, r) \in \mathbb{R}^{n \times r}$, then let $M = M_L M_R^T$. We sample a subset with sampling ratio SR uniformly at random, where $SR = |\Omega|/(mn)$. In our experiment, we set $m = n$. The GMM noise are set as $\sigma_A^2 = 0.0001, \sigma_B^2 = 0.1$ and $c = 0.2$. The rank r is set to be 30 and the

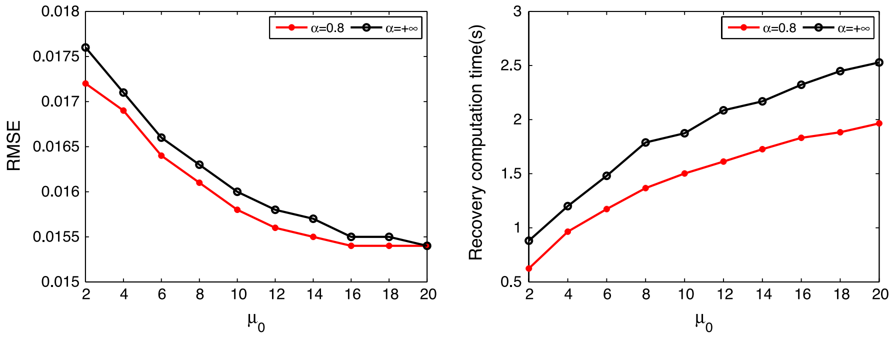

Fig. 2 Curves of RMSE and average running times with different μ_0

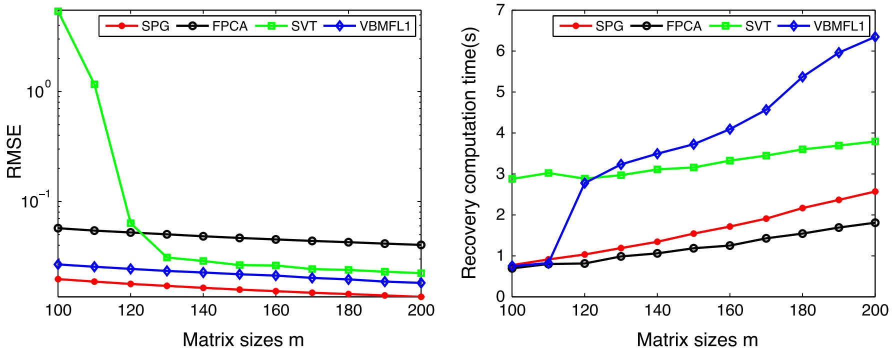

Fig. 3 Curves of RMSE and average running times with different matrix size m

sampling ratio SR is set to be 0.8. For each simulation, an average relative error is obtained via 100 Monte Carlo runs with different realizations of M, Ω and noise.

The performance is firstly compared for different μ_0 under different μ_k in step 3 of SPG algorithm. For $m = 150$, $\alpha = 0.8$ and $\alpha = +\infty$, the results for different μ_0 are reported in Fig. 2, where μ_0 increases from 2 to 20 with increment 2. Figure 2 shows that the larger μ_0 becomes, the smaller the value of RMSE with the more running time. It can also be observed that SPG algorithm of $\alpha = 0.8$ performs better than that of $\alpha = +\infty$. For $\alpha = 0.8$, it is necessary to reduce μ_k when (3.18) is not satisfied, while μ_k reduces for $\alpha = +\infty$. We can assert that the strategy of μ_k in step 3 of SPG algorithm works well.

Secondly, we report the performance of the algorithms for different sizes of completion problems in Fig. 3. The size of the square matrix increases from 100 to 200 with increment 10. Figure 3 shows the curves of the average RMSE and running times in terms of different matrix sizes m. As can be seen from Fig. 3, the SPG algorithm achieves comparably lower average RMSE than the other algorithms, while FPCA and SVT algorithms based on l_2 norm have higher RMSE values. Moreover, as the size of the matrix increases, the average RMSE values decrease for all algorithms. With the increase of matrix size, the average running time of all algorithms increases gradually, while the average running time of SPG algorithm is the least except FPCA. The average running time of VBMFL1 algorithm based on l_1 norm increases much faster than another three algorithms. In summary, SPG algorithm performs best among the four algorithms.

4.2 Image inpainting

In this subsection, the performance of the algorithms is compared for some image inpainting tasks with non-Gaussian noise. Note that grayscale images can be expressed as matrix. When the matrix data is of low-rank, or numerical low-rank, the image inpainting problem can be modeled as matrix completion problem. To evaluate the algorithm performances under non-Gaussian noise, a mixture of Gaussian is selected for the noise model. We adopt the peak signal-to-noise ratio (PSNR) as an evaluation metric, which is defined by

$$\mathrm{PSNR} := 10\log_{10}\left(\frac{mn}{\|X^* - M\|_F^2}\right).$$

A higher PSNR represents better recovery performance.

We use the USC-SIPI image database[1] to evaluate our method for image inpainting. In the test, we randomly select 6 images from this database for testing and the images are normalized in the range $[0, 1]$. In Fig. 4, we consider the case where entries are missing at random by sampling ratio $SR = 0.9$. The GMM noise is set at $\sigma_A^2 = 0.001, \sigma_B^2 = 0.1, c = 0.1$. From Fig. 4, we can see that the image restored by FPCA and SVT algorithm with l_2 norm is very blurred, while the image restored

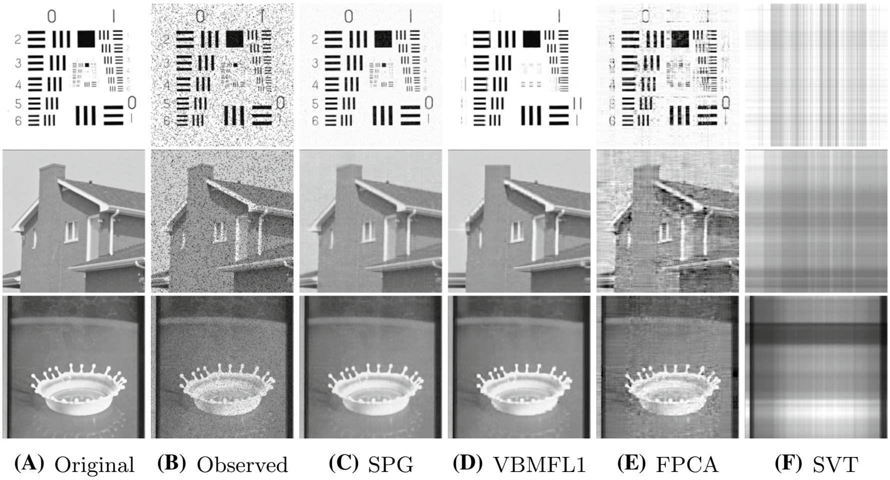

Fig. 4 Image inpainting sample of image under a mixture of Gaussian noise

[1] http://sipi.usc.edu/database/.

Table 1 Image inpainting performance comparison under a mixture of Gaussian noise: PSNR and running times

Method	Image					
	Chart		House		Splash	
	PSNR	Time	PSNR	Time	PSNR	Time
SPG	**28.07**	**1.53**	**30.03**	**2.02**	**31.69**	**6.61**
VBMFL1	20.70	30.22	28.09	27.82	30.88	96.98
FPCA	16.74	2.50	20.62	2.63	23.81	12.08
SVT	10.95	5.40	17.21	4.52	13.80	9.52

Bold values indicate the results obtained by SPG

by SPG and VBMFL1 algorithm with l_1 norm is relatively clear, indicating that the recovery effect of l_1 norm for non-Gaussian noise is better than that of l_2 norm. In the image restored by VBMFL1 algorithm, the recovery effect is not good for those isolated small pixels, especially in “Chart". It may be that these abnormal small pixels are treated as outliers. The image restored by SPG algorithm performs well in all aspects. At the same time, in order to compare the recovery effect of the four algorithms more clearly, we give the PSNR and running time of the four algorithms in Table 1. It can be seen from the table that the VBMFL1 algorithm based on l_1 norm has higher PSNR values than the FPCA and SVT algorithms based on l_2 norm, while the running time is much longer. SPG algorithm has the highest PSNR value and the least running time. We can assert that SPG is the best of the four algorithms.

In Fig. 5 and Table 2, we consider the case where entries are missing at random by sampling ratio $SR = 0.7$, and Gaussian noise with variance 0.0001 is added to the

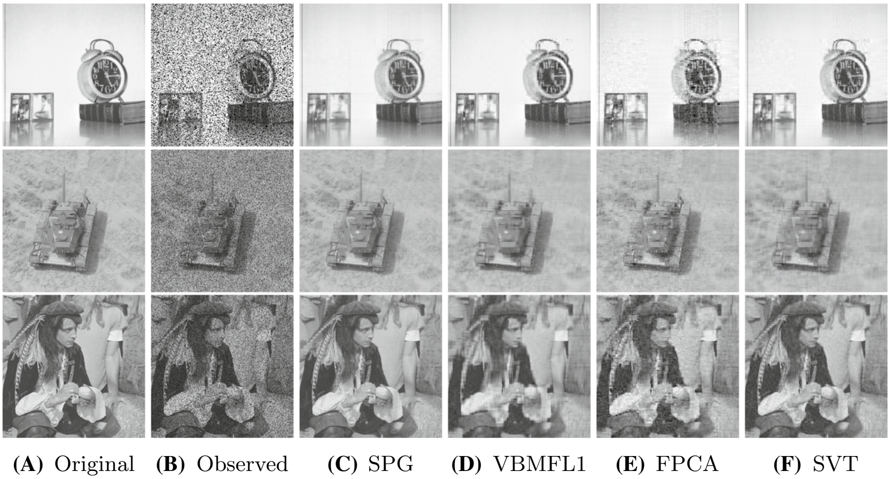

Fig. 5 Image inpainting sample of image under Gaussian noise

Table 2 Image inpainting performance comparison under Gaussian noise: PSNR and running times

Method	Image					
	Clock		Tank		Man	
	PSNR	Time	PSNR	Time	PSNR	Time
SPG	**30.15**	**1.32**	**32.07**	**7.38**	**30.15**	**7.18**
VBMFL1	28.11	10.48	27.40	84.71	23.82	70.09
FPCA	24.06	2.25	26.09	12.36	22.79	11.39
SVT	28.80	14.18	26.93	23.54	25.85	35.79

Bold values indicate the results obtained by SPG

observed pixels. It can be seen from Fig. 5 and Table 2 that the recovery effect of the algorithm based on l_1 norm is similar to that of the algorithm based on l_2 norm in Gaussian noise, and the running time is relatively long. However, SPG algorithm still has the highest PSNR value and the least running time. Therefore, no matter whether Gaussian noise or not, SPG algorithm performs best for image restoration.

4.3 MRI volume dataset

The resolution of the MRI volume dataset[2] is of size 217 × 181 with 181 slices and we selected the 38th slice and the 88th slice for the experiment. We consider the case where entries are missing at random by sampling ratio $SR = 0.9$. The GMM noise is set at $\sigma_A^2 = 0.0001, \sigma_B^2 = 0.1, c = 0.01$.

From Figs. 6 and 7, we can see that the effect of FPCA and SVT to restore images is very poor. Although the effect of VBMFL1 algorithm to restore images is better, the running time is relatively longer. Furthermore, the effect of restored image by SPG algorithm is the best one with the least running time. In summary, the SPG algorithm has the best recovery effect.

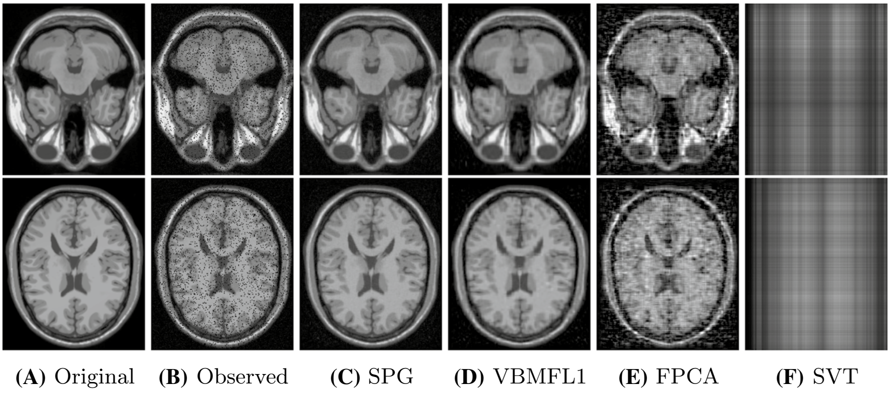

Fig. 6 Completion results of the MRI volume dataset

[2] http://graphics.stanford.edu/data/voldata/.

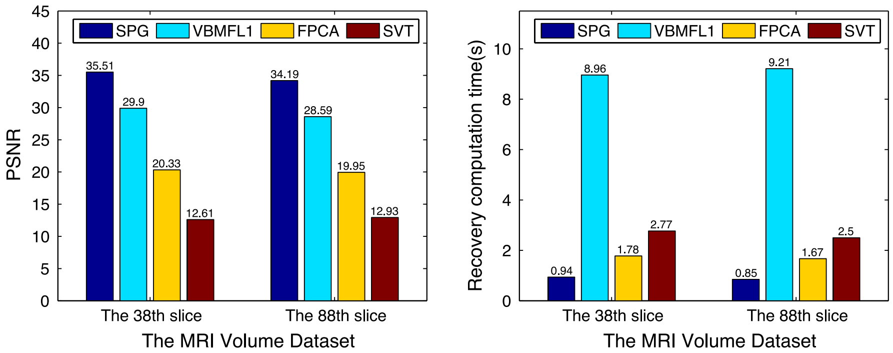

Fig. 7 Histogram of representation results for the MRI volume dataset

Acknowledgements Xinzhen Zhang was partly supported by the National Natural Science Foundation of China (Grant No. 11871369). Quan Yu was partly supported by Tianjin Research Innovation Project for Postgraduate Students (Grant No. 2020YJSS140).

Data availibility The datasets generated by the study during and/or analyzed during the current study are available in the Dataverse repository: https://github.com/Spring-527/SPG.

References

1. Bian, W., Chen, X.: A smoothing proximal gradient algorithm for nonsmooth convex regression with cardinality penalty. SIAM J. Numer. Anal. **58**, 858–883 (2020)
2. Bouwmans, T., Zahzah, E.: Robust PCA via principal component pursuit: a review for a comparative evaluation in video surveillance. Comput. Vis. Image Underst. **122**, 22–34 (2014)
3. Cai, J., Candes, E., Shen, Z.: A singular value thresholding algorithm for matrix completion. SIAM J. Optim. **20**, 1956–1982 (2008)
4. Candès, E., Recht, B.: Exact matrix completion via convex optimization. Found. Comput. Math. **9**, 717–772 (2009)
5. Candès, E., Tao, T.: The power of convex relaxation: near-optimal matrix completion. IEEE Trans. Inf. Theory **56**, 2053–2080 (2010)
6. Cao, F., Chen, J., Ye, H., Zhao, J., Zhou, Z.: Recovering low-rank and sparse matrix based on the truncated nuclear norm. Neural Netw. **85**, 10–20 (2017)
7. Clarke, F.: Optimization and nonsmooth analysis. Society for Industrial and Applied Mathematics (1990)
8. Fazel, M., Hindi, H., Boyd, S.: A rank minimization heuristic with application to minimum order system approximation. Proc. Am. Control Conf. **6**, 4734–4739 (2001)
9. Fornasier, M., Rauhut, H., Ward, R.: Low-rank matrix recovery via iteratively reweighted least squares minimization. SIAM J. Optim. **21**, 1614–1640 (2011)
10. Haldar, J., Hernando, D.: Rank-constrained solutions to linear matrix equations using powerfactorization. IEEE Signal Process. Lett. **16**, 584–587 (2009)
11. He, Y., Wang, F., Li, Y., Qin, J., Chen, B.: Robust matrix completion via maximum correntropy criterion and half-quadratic optimization. IEEE Trans. Signal Process. **68**, 181–195 (2020)
12. Lai, M., Xu, Y., Yin, W.: Improved iteratively reweighted least squares for unconstrained smoothed ℓ_q minimization. SIAM J. Numer. Anal. **51**, 927–957 (2013)
13. Lee, C., Lam, E.: Computationally efficient truncated nuclear norm minimization for high dynamic range imaging. IEEE Trans. Image Process. **25**, 4145–4157 (2016)
14. Lewis, A., Sendov, H.: Nonsmooth analysis of singular values. Part II: applications. Set-Valued Anal. **13**, 243–264 (2005)

15. Lu, Z., Zhang, Y., Li, X.: Penalty decomposition methods for rank minimization. Optim. Methods Softw. **30**, 531–558 (2015)
16. Lu, Z., Zhang, Y., Lu, J.: l_p Regularized low-rank approximation via iterative reweighted singular value minimization. Comput. Optim. Appl. **68**, 619–642 (2017)
17. Ma, S., Goldfarb, D., Chen, L.: Fixed point and Bregman iterative methods for matrix rank minimization. Math. Program. **128**, 321–353 (2011)
18. Ma, T., Lou, Y., Huang, T.: Truncated l_{1-2} models for sparse recovery and rank minimization. SIAM J. Imag. Sci. **10**, 1346–1380 (2017)
19. Meng, D., Xu, Z., Zhang, L., and Zhao, J.: A cyclic weighted median method for L1 low-rank matrix factorization with missing entries. In: Seventh AAAI Conference on Artificial Intelligence (2013), pp. 704–710
20. Nocedal, J., Wright, S.: Numerical Optimization. Springer, Berlin (2006)
21. Pan, L., Chen, X.: Group sparse optimization for images recovery using capped folded concave functions. SIAM J. Imag. Sci. **14**, 1–25 (2021)
22. Peng, C., Kang, Z., Cheng, Q.: A fast factorization-based approach to robust PCA. In: 2016 IEEE 16th International Conference on Data Mining, (2016) pp. 1137–1142
23. Su, X., Wang, Y., Kang, X., Tao, R.: Nonconvex truncated nuclear norm minimization based on adaptive bisection method. IEEE Trans. Circuits Syst. Video Technol. **29**, 3159–3172 (2019)
24. Wang, N., Yao, T., Wang, J., Yeung, D.: A probabilistic approach to robust matrix factorization. Comput. Vis. ECCV **2012**, 126–139 (2012)
25. Wen, Z., Yin, W., Zhang, Y.: Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm. Math. Program. Comput. **4**, 333–361 (2012)
26. Xu, H., Caramanis, C., Sanghavi, S.: Robust PCA via outlier pursuit. IEEE Trans. Inf. Theory **58**, 3047–3064 (2012)
27. Zeng, W., So, H.: Outlier-robust matrix completion via l_p-minimization. IEEE Trans. Signal Process. **66**, 1125–1140 (2018)
28. Zhao, Q., Meng, D., Xu, Z., Zuo, W., Yan, Y.: L_1-norm low-rank matrix factorization by variational bayesian method. IEEE Trans. Neural Netw. Learn.Syst. **26**, 825–839 (2015)
29. Zheng, Y., Liu, G., Sugimoto, S., Yan, S., Okutomi, M.: Practical low-rank matrix approximation under robust L1-norm. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1410–1417 (2012)