Numerische Mathematik (2023) 153:213–248
https://doi.org/10.1007/s00211-022-01335-7

Numerische Mathematik

# Proximal linearization methods for Schatten $p$-quasi-norm minimization

**Chao Zeng**[1]

Received: 3 January 2022 / Revised: 2 November 2022 / Accepted: 12 November 2022 /
Published online: 2 December 2022


**Abstract**
Schatten $p$-quasi-norm minimization has advantages over nuclear norm minimization in recovering low-rank matrices. However, Schatten $p$-quasi-norm minimization is much more difficult, especially for generic linear matrix equations. In this paper, we first extend the lower bound theory of $\ell_p$ minimization to Schatten $p$-quasi-norm minimization. We prove that the positive singular values of local minimizers are bounded from below by a constant. Motivated by this property, we propose a proximal linearization method, whose subproblems can be solved efficiently by the (linearized) alternating direction method of multipliers. The convergence analysis of the proposed method involves the nonsmooth analysis of singular value functions. We give a necessary and sufficient condition for a singular value function to be a Kurdyka–Łojasiewicz function. The subdifferentials of related singular value functions are computed. The global convergence of the proposed method is established under some assumptions. Experiments on matrix completion, Sylvester equation and image deblurring show the effectiveness of the algorithm.

**Mathematics Subject Classification** 15A15 · 15A24 · 15A60 · 90C26 · 90C30

## 1 Introduction

Consider the following minimization problem

$$\begin{aligned} \min_{X\in\mathbb{R}^{m\times n}} \quad & \operatorname{rank}(X) \\ \text{s.t.} \quad & \mathcal{A}(X) = \boldsymbol{b}, \end{aligned} \tag{1}$$

✉ Chao Zeng
zengchao@nankai.edu.cn

[1] School of Mathematical Sciences and LPMC, Nankai University, Tianjin, China

where $\mathcal{A} : \mathbb{R}^{m\times n} \to \mathbb{R}^d$ is a linear operator and $\boldsymbol{b} \in \mathbb{R}^d$. This is called the affine rank minimization problem [41] and arises in various application areas, such as control applications [18, 19, 37], stochastic PDEs [30], matrix completion [8, 10], and image processing [22, 23]. This problem is NP-hard. Let the singular values of $X$ be $\sigma_1(X), \ldots, \sigma_h(X)$, where $h = \min\{m, n\}$. The nuclear norm of $X$ is $\|X\|_* = \sum_{i=1}^h \sigma_i(X)$. It is proved in [19] that the nuclear norm is the convex envelope of the rank function over the unit ball under the spectral norm. Using the nuclear norm to replace the rank function is a common method [7, 8, 10, 40, 41, 50] for finding low-rank solutions, i.e.,

$$\begin{aligned} \min_{X\in\mathbb{R}^{m\times n}} \quad & \|X\|_* \\ \text{s.t.} \quad & \mathcal{A}(X) = \boldsymbol{b}. \end{aligned} \tag{2}$$

In practice, $\boldsymbol{b}$ is usually contaminated by noise, or we just want to gain a low-rank approximate solution. In such cases, (2) is often relaxed to the nuclear norm regularized least squares problem

$$\min_{X\in\mathbb{R}^{m\times n}} \|X\|_* + \frac{\lambda}{2}\|\mathcal{A}(X) - \boldsymbol{b}\|_2^2, \tag{3}$$

where $\lambda > 0$ is the regularization parameter.

For diagonal matrices, the nuclear norm is just the $\ell_1$ norm of the diagonal elements. In sparse reconstruction and image recovery, using the $\ell_1$ norm to replace the cardinality function (also called the $\ell_0$ quasi-norm) for finding sparse solutions is a well-known technique in the literature [9, 16, 43]. So, (3) can be regarded as a generalization of the $\ell_1$ minimization problem. Compared to the $\ell_1$ norm $\|\boldsymbol{x}\|_1$, the power $p$ of the $\ell_p$ quasi-norm $\|\boldsymbol{x}\|_p^p := \sum_i |x_i|^p$, $0 < p < 1$, makes a closer approximation to the cardinality function $\|\boldsymbol{x}\|_0$. Therefore, the $\ell_p$ quasi-norm is widely used in sparse reconstruction [11, 15, 27] and image recovery [39, 52]. Specifically, the $\ell_p$ minimization for sparse reconstruction is

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \|\boldsymbol{x}\|_p^p + \frac{\lambda}{2}\|A\boldsymbol{x} - \boldsymbol{b}\|_2^2, \tag{4}$$

where $A \in \mathbb{R}^{m\times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\lambda > 0$; and the $\ell_p$ minimization is often used for applications where $\boldsymbol{x}$ is an image:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \sum_{i=1}^n \|D_i\boldsymbol{x}\|_2^p + \frac{\lambda}{2}\|A\boldsymbol{x} - \boldsymbol{b}\|_2^2, \tag{5}$$

where $A \in \mathbb{R}^{m\times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\lambda > 0$, and $D_i\boldsymbol{x}$ is the discrete gradient of $\boldsymbol{x}$ at pixel $i$. The Schatten $p$-quasi-norm of $X$ is defined as

$$\|X\|_{*,p} := \left(\sum_{i=1}^h \sigma_i^p(X)\right)^{\frac{1}{p}}.$$

Similarly, the Schatten $p$-quasi-norm ($0 < p < 1$) is also used for finding low-rank solutions, e.g., [21, 27, 36, 38]. The corresponding Schatten $p$-quasi-norm regularized least squares problem is

$$\min_{X\in\mathbb{R}^{m\times n}} \|X\|_{*,p}^{p} + \frac{\lambda}{2}\|\mathcal{A}(X) - \boldsymbol{b}\|_2^2, \tag{6}$$

where $\lambda > 0$ is the regularization parameter. Recently, a sufficient condition stating when the Schatten $p$-quasi-norm minimization can be used for low-rank matrix recovery was established in [26].

To solve (6), most existing methods follow the framework of the iteratively reweighted least-squares algorithms (IRLS-$p$) [20]. These methods are motivated by the following fact

$$\mathrm{tr}\left((X^\top X)^{p/2}\right) = \|X\|_{*,p}^{p}.$$

By introducing a smoothing parameter $\varepsilon > 0$, (6) is converted into

$$\min_{X\in\mathbb{R}^{m\times n}} \mathrm{tr}\left((X^\top X + \varepsilon I)^{p/2}\right) + \frac{\lambda}{2}\|\mathcal{A}(X) - \boldsymbol{b}\|_2^2.$$

The first-order optimality condition of the above problem reads

$$pX(X^\top X + \varepsilon I)^{p/2-1} + \lambda\mathcal{A}^*(\mathcal{A}(X) - \boldsymbol{b}) = 0,$$

where $\mathcal{A}^*$ is the adjoint operator of $\mathcal{A}$. Given $X_k$ at iteration $k$, $X_{k+1}$ can be solved by

$$pXW_k + \lambda\mathcal{A}^*(\mathcal{A}(X) - \boldsymbol{b}) = 0, \tag{7}$$

where $W_k := (X_k^\top X_k + \varepsilon I)^{p/2-1}$. Let the vectorized result of $X$ be $\boldsymbol{x} := \mathrm{vec}(X) \in \mathbb{R}^{mn}$. For a generic linear operator $\mathcal{A}$, (7) is a linear system in the form of $C\boldsymbol{x} = \boldsymbol{d}$, where $C \in \mathbb{R}^{mn\times mn}$, $\boldsymbol{d} \in \mathbb{R}^{mn}$. Hence, directly solving (7) is too expensive for general cases. For example, if $\mathcal{A}(X) = \boldsymbol{b}$ is a Sylvester equation, then (7) is a general matrix equation, which is difficult to solve. See the discussion in Sect. 5.4. In the case of $X$ being square, Gazzola et al. [21] converts (7) into a Tikhonov-regularized problem

$$X_{k+1} \in \arg\min_X \frac{p}{2}\left\|(X_kX_k^\top + \varepsilon I)^{p/4-1/2}X\right\|_F^2 + \frac{\lambda}{2}\|\mathcal{A}(X) - \boldsymbol{b}\|_2^2,$$

and uses Krylov methods to solve it.

In this paper, we first give some theoretical analysis on (6). Our motivation comes from the *low bound theory* of the $\ell_p$ minimization for image recovery and sparse reconstruction. Suppose $\widetilde{\boldsymbol{x}}$ is a (local) minimizer of (5). It is proved in [39, 52] that there exists a constant $\theta > 0$ such that if $D_i\widetilde{\boldsymbol{x}} \neq 0$, then

$$\|D_i\widetilde{\boldsymbol{x}}\|_2 \geq \theta.$$

Note that $\|D_i\widetilde{\boldsymbol{x}}\|_2$ represents the contrast between neighboring pixels. This result suggests that the image restored by (5) has neat edges. Suppose $\widetilde{\boldsymbol{x}}$ is a (local) minimizer of (4). Similarly, it is proved in [15] that there exists a constant $\theta > 0$ such that if $\widetilde{x}_i \neq 0$, then

$$|\widetilde{x}_i| \geq \theta.$$

This explains the advantage of (4) for sparse reconstruction: Each nonzero $\widetilde{x}_i$ plays a significant role, and we cannot obtain a sparser solution by changing any nonzero $\widetilde{x}_i$ into zero, without affecting the approximation error greatly. We will give a similar result for (6), i.e., there is a lower bound for the nonzero singular values of (local) minimizers of (6).

In solving nonlinear optimization problem, proximal linearization is a common technique; see, e.g., [5, 35, 54–56]. However, this technique cannot be used directly on (6), because $\|X\|_{*,p}^p$ is non-Lipschitz when $X$ is singular. In IRLS-$p$, introducing a smoothing parameter $\varepsilon > 0$ is just to tackle the non-Lipschitzian of $\|X\|_{*,p}^p$. After establishing the low bound theory, we will find that the non-Lipschitzian can be overcome by adding a rank constraint. Then the objective function can be proximally linearized and the converted subproblem is a weighted nuclear norm minimization with rank constraints. The subproblem can be solved efficiently by the (linearized) alternating direction method of multipliers. Unlike the existing methods [20, 21, 27, 36, 38], we do not need to solve the complicated linear system (7).

The Kurdyka–Łojasiewicz (KL) property of objective functions has been widely used in the proof of convergence of first-order methods, but most of these methods are aimed at functions whose arguments are vectors. For this work, the objective function (6) involves a singular value function $\|X\|_{*,p}^p$. We give a necessary and sufficient condition for a singular value function to be a KL function. The subdifferentials of related singular value functions are computed. In previous works on the convergence analysis based on the KL property, e.g., [2, 3, 5], Lipschitz gradient is usually a key assumption. For this work, the function $\|X\|_{*,p}^p$ does not have Lipschitz gradient. To tackle this difficulty, we prove the lower bound property of the iterate sequence $\{X_k\}$, under the assumption that the nonzero singular values of $X_k$ are strictly decreasing. In practical applications, this assumption holds generically. See Sect. 4.3 for details. With all these preparation, we prove the global convergence of the proposed algorithm.

The rest of this paper is organized as follows. In Sect. 2, we establish the low bound theory of the Schatten $p$-quasi-norm minimization. The algorithm is proposed in Sect. 3 and the convergence analysis is given in Sect. 4. Experimental results are given in Sect. 5. Conclusions are presented in Sect. 6.

### Notation

Scalars are denoted by lowercase letters, e.g., $a, \alpha$. Vectors are denoted by boldface lowercase letters, e.g., $\boldsymbol{a}, \boldsymbol{\alpha}$. Matrices are denoted by capital letters, e.g., $A, \Sigma$. The $n$-dimensional nonnegative real vector space is denoted by $\mathbb{R}_+^n$. Given a vector $\boldsymbol{x} \in \mathbb{R}^n$, $|\boldsymbol{x}| = [|x_1|, \ldots, |x_n|]^\top \in \mathbb{R}_+^n$. Given a vector $\boldsymbol{x} \in \mathbb{R}^m$, we use $\mathrm{diag}(\boldsymbol{x})$ to denote a matrix whose main diagonal is $x_1, \ldots, x_m$. Usually, we do not mention the size of

$\mathrm{diag}(\boldsymbol{x})$, which will be clear from the context. The $\ell_2$ norm of $\boldsymbol{x}$ is denoted by $\|\boldsymbol{x}\|_2$. The Frobenius norm of $X$ is denoted by $\|X\|_F$. The spectral norm of $X$ is denoted by $\|X\|_2$. Let $X \in \mathbb{R}^{m\times n}$. The singular value vector of $X$ is denoted by $\boldsymbol{\sigma}(X) := [\sigma_1(X), \ldots, \sigma_h(X)]^\top \in \mathbb{R}^h$, where $h = \min\{m, n\}$ and $\sigma_1(X) \geq \cdots \geq \sigma_h(X)$ are all singular values of $X$. The *Stiefel manifold* is defined as

$$\mathrm{St}_t^m := \{Y \in \mathbb{R}^{m\times t} : Y^\top Y = I\}.$$

The set of all *singular matrices* of $X$ is denoted by

$$\mathrm{Sm}(X) := \left\{(U, V) \in \mathrm{St}_m^m \times \mathrm{St}_n^n : U\,\mathrm{diag}(\boldsymbol{\sigma}(X))V^\top = X\right\}. \tag{8}$$

Let $\mathscr{S}$ be a subset of $\mathbb{R}^n$. The indicator function of $\mathscr{S}$ is defined as

$$\delta_{\mathscr{S}}(\boldsymbol{x}) = \begin{cases} 0, & \text{if } \boldsymbol{x} \in \mathscr{S}, \\ +\infty, & \text{otherwise.} \end{cases}$$

## 2 Lower bound of singular values

**Theorem 2.1** *Let* $\theta = \left(\frac{p(1-p)}{\lambda\|\mathcal{A}\|_2^2}\right)^{\frac{1}{2-p}}$, *where* $\|\mathcal{A}\|_2$ *is the spectral norm of* $\mathcal{A}$. *For each (local) minimizer* $\widetilde{X}$ *of* (6), *we have*

$$\sigma_r(\widetilde{X}) \geq \theta \quad \text{for all } r = 1, \ldots, \mathrm{rank}(\widetilde{X}).$$

***Proof*** Let $h = \min\{m, n\}$ in (6). Each $X \in \mathbb{R}^{m\times n}$ can be written as

$$X = U\Gamma V^\top,$$

where $U = [\boldsymbol{u}_1 \cdots \boldsymbol{u}_h] \in \mathrm{St}_h^m$, $V = [\boldsymbol{v}_1 \cdots \boldsymbol{v}_t] \in \mathrm{St}_h^n$, and $\Gamma = \mathrm{diag}(\gamma_1, \ldots, \gamma_h)$. Then $|\gamma_i|$, $i = 1, \ldots, h$ are all singular values of $X$. Define $\boldsymbol{\gamma} := [\gamma_1, \ldots, \gamma_h]^\top \in \mathbb{R}^h$. Problem (6) is equivalent to

$$\min_{\boldsymbol{\gamma}\in\mathbb{R}^h} \min_{U\in\mathrm{St}_h^m} \min_{V\in\mathrm{St}_h^n} \|\boldsymbol{\gamma}\|_p^p + \frac{\lambda}{2}\|\mathcal{A}(U\Gamma V^\top) - \boldsymbol{b}\|_2^2.$$

Suppose $(\widetilde{\boldsymbol{\gamma}}, \widetilde{U}, \widetilde{V})$ is a local minimizer of the above problem. Then $\widetilde{\boldsymbol{\gamma}}$ is a local minimizer of

$$\min_{\boldsymbol{\gamma}\in\mathbb{R}^h} \|\boldsymbol{\gamma}\|_p^p + \frac{\lambda}{2}\|\mathcal{A}(\widetilde{U}\Gamma\widetilde{V}^\top) - \boldsymbol{b}\|_2^2. \tag{9}$$

Note that $\mathcal{A}$ is a linear operator. We have

$$\mathcal{A}(\widetilde{U}\Gamma\widetilde{V}^\top) = \mathcal{A}\left(\sum_{i=1}^{h}\gamma_i\widetilde{\boldsymbol{u}}_i\widetilde{\boldsymbol{v}}_i^\top\right) = \sum_{i=1}^{h}\gamma_i\mathcal{A}\left(\widetilde{\boldsymbol{u}}_i\widetilde{\boldsymbol{v}}_i^\top\right).$$

Define a matrix $C = [\boldsymbol{c}_1 \cdots \boldsymbol{c}_h] \in \mathbb{R}^{d\times h}$ with $\boldsymbol{c}_i = \mathcal{A}\left(\widetilde{\boldsymbol{u}}_i\widetilde{\boldsymbol{v}}_i^\top\right) \in \mathbb{R}^d, i = 1,\ldots,h$. Then $\mathcal{A}(\widetilde{U}\Gamma\widetilde{V}^\top) = C\boldsymbol{\gamma}$ and (9) is converted into

$$\min_{\boldsymbol{\gamma}\in\mathbb{R}^h} \|\boldsymbol{\gamma}\|_p^p + \frac{\lambda}{2}\|C\boldsymbol{\gamma} - \boldsymbol{b}\|_2^2,$$

which is an $\ell_p - \ell_2$ minimization problem. Note that $\|\mathrm{vec}(\widetilde{\boldsymbol{u}}_i\widetilde{\boldsymbol{v}}_i^\top)\|_2 = \|\widetilde{\boldsymbol{u}}_i\|_2\|\widetilde{\boldsymbol{v}}_i\|_2 = 1$ and $\|\boldsymbol{c}_i\|_2 = \|\mathcal{A}\left(\widetilde{\boldsymbol{u}}_i\widetilde{\boldsymbol{v}}_i^\top\right)\|_2 \le \|\mathcal{A}\|_2$. By Chen et al. [15, Theorem 2.1], if $\widetilde{\gamma}_i \neq 0$, we have

$$|\widetilde{\gamma}_i| \ge \left(\frac{p(1-p)}{\lambda\|\boldsymbol{c}_i\|_2^2}\right)^{\frac{1}{2-p}} \ge \left(\frac{p(1-p)}{\lambda\|\mathcal{A}\|_2^2}\right)^{\frac{1}{2-p}} = \theta.$$

This completes the proof. □

Like the lower bound theory of (4), the preceding theorem explains the advantage of (6) for low-rank matrix recovery: Each nonzero singular value plays a significant role, and we cannot obtain a lower rank solution by changing any nonzero singular value into zero, without affecting the approximation error greatly. On the other hand, if the original low-rank matrix has some singular values with very small magnitudes, using (6) cannot recover the original matrix exactly.

Let

$$\mathscr{M}_{\le r} := \{X \in \mathbb{R}^{m\times n} : \mathrm{rank}(X) \le r\} \tag{10}$$

denote the set of matrices of rank at most $r$. We have the following corollary.

**Corollary 2.2** *Let* $\theta = \left(\frac{p(1-p)}{\lambda\|\mathcal{A}\|_2^2}\right)^{\frac{1}{2-p}}$ *and* $\widetilde{X}$ *be a local minimizer of* (6)*. Then we have*

$$\|X - \widetilde{X}\|_F \ge \theta \quad \forall X \in \mathscr{M}_{\le \mathrm{rank}(\widetilde{X})-1}.$$

***Proof*** Let $h = \min\{m,n\}$. By Horn and Johnson [24, Corollary 7.4.1.3], for any $X \in \mathscr{M}_{\le \mathrm{rank}(\widetilde{X})-1}$, we have

$$\begin{aligned}\|X - \widetilde{X}\|_F^2 &\ge \sum_{i=1}^{h}\left(\sigma_i(X) - \sigma_i(\widetilde{X})\right)^2 \ge \left(\sigma_{\mathrm{rank}(\widetilde{X})}(X) - \sigma_{\mathrm{rank}(\widetilde{X})}(\widetilde{X})\right)^2 \\ &= \sigma_{\mathrm{rank}(\widetilde{X})}^2(\widetilde{X}) \ge \theta^2,\end{aligned}$$

which completes the proof. □

With $\theta$ defined in the preceding corollary, denote the ball of radius $\theta$ around $X$ by

$$\mathscr{B}(X,\theta) := \{Y \in \mathbb{R}^{m\times n} : \|Y - X\|_F \le \theta\}.$$

The preceding corollary is equivalent to say that any minimizer in $\mathscr{B}(X,\theta)$ has rank less than $\mathrm{rank}(X)$.

## 3 Algorithm

Suppose we use an iterative algorithm to find a local minimizer $\widetilde{X}$ of (6) and the iterate sequence is $\{X_k\}$. By Corollary 2.2, if $\lim_{k\to\infty} X_k = \widetilde{X}$, then there is a $\widetilde{k}$ such that $\mathrm{rank}(\widetilde{X}) \le r_k := \mathrm{rank}(X_k)$ for all $k \ge \widetilde{k}$. Motivated by this phenomenon, we consider solving $X_{k+1}$ by

$$\min_{\mathrm{rank}(X)\le r_k} \mathcal{P}(X) := \|X\|_{*,p}^p + \frac{\lambda}{2}\|\mathcal{A}(X) - \boldsymbol{b}\|_2^2. \tag{11}$$

Directly solving (11) is difficult. Usually, the linearization technique, i.e, the first-order Taylor approximation of some term around $X_k$, is employed to obtain an approximate objective function which is easy to solve. In (11), what we need to linearize is the term $\|X\|_{*,p}^p$. Generally, the linearization cannot be used because $\|X\|_{*,p}^p$ is non-Lipschitz when $X_k$ is singular. However, thanks to the constraint $\mathrm{rank}(X) \le r_k$, we can still employ the linearization on (11). Specifically, for $X \in \mathbb{R}^{m\times n}$ with $\mathrm{rank}(X) \le r_k$, we have

$$\|X\|_{*,p}^p = \sum_{j=1}^{r_k} \sigma_j^p(X) \approx \sum_{j=1}^{r_k} \left(\sigma_j^p(X_k) + p\sigma_j^{p-1}(X_k)\cdot\big(\sigma_j(X) - \sigma_j(X_k)\big)\right). \tag{12}$$

For convenience, we define

$$\begin{aligned} \sigma_{k,j} &= \sigma_j(X_k), \quad \boldsymbol{\sigma}_k = [\sigma_{k,1},\ldots,\sigma_{k,r_k}]^\top \in \mathbb{R}^{r_k},\\ w_{k,j} &= p\sigma_{k,j}^{p-1}, \quad \boldsymbol{w}_k = [w_{k,1},\ldots,w_{k,r_k}]^\top \in \mathbb{R}^{r_k}. \end{aligned} \tag{13}$$

Since $\sigma_1(X_k) \ge \cdots \ge \sigma_{r_k}(X_k) > 0$, we have

$$w_{k,1} \le \cdots \le w_{k,r_k}. \tag{14}$$

In addition, since the function $z \mapsto z^p$ is concave on $[0,+\infty)$, it follows that for all $X \in \mathbb{R}^{m\times n}$ with $\mathrm{rank}(X) \le r_k$,

$$\|X\|_{*,p}^p \le \sum_{j=1}^{r_k} w_{k,j}\sigma_j(X) + \sum_{j=1}^{r_k}(\sigma_{k,j}^p - w_{k,j}\sigma_{k,j}). \tag{15}$$

Substituting (12) into (11), deleting some constants and adding a proximal term, we obtain the following proximal linearization version of (11):

$$\min_{\text{rank}(X)\le r_k} \mathcal{H}_k(X) := \sum_{j=1}^{r_k} w_{k,j}\sigma_j(X) + \frac{\lambda}{2}\|\mathcal{A}(X) - \boldsymbol{b}\|_2^2 + \frac{\tau}{2}\|X - X_k\|_F^2. \quad (16)$$

The proximal term $\frac{\tau}{2}\|X - X_k\|_F^2$ is to increase the robustness of the nonconvex algorithm and is helpful in the global convergence. By (15), we can set $\tau$ to be any positive scalar. Let $\delta_{\mathscr{M}_{\le r}}(X)$ be the indicator function of $\mathscr{M}_{\le r}$, where $\mathscr{M}_{\le r}$ is defined in (10). Then (16) is equivalent to

$$\min_{X\in\mathbb{R}^{m\times n}} \mathcal{G}_k(X) := \mathcal{H}_k(X) + \delta_{\mathscr{M}_{\le r_k}}(X). \quad (17)$$

We will use an iterative algorithm to solve (16) or (17). Precisely solving the subproblem (17) needs infinite iteration steps, which cannot be realized in real computation. Therefore, we consider solving (17) inexactly. The whole procedure of the *proximal linearization method* (PLM) is summarized in Algorithm 1. In general, algorithms for nonconvex problems seek only a local solution. Therefore, the initial values are important for nonconvex optimization. For Algorithm 1, $r_0$ and $\boldsymbol{w}_0$ are determined by $X_0$. In addition, PLM has used the scheme of proximal linearization, which approximates the original model linearly. If the starting point $X_0$ is far from the final result, a large number of steps would be required for convergence. Therefore, the initial value $X_0$ is very crucial for PLM. We will elaborate this issue in Sect. 5.1.

**Algorithm 1:** proximal linearization method (PLM) for solving Schatten $p$-quasi-norm minimization (6)

**Input**: $\mathcal{A}$, $\boldsymbol{b}$, $p$, $\lambda$, $\tau$, $\rho > 0$, initial value $X_0$, $r_0$, $\boldsymbol{w}_0$, $k = 0$

**repeat**

  Compute $X_{k+1}$ by approximately solving (17): $X_{k+1} \approx \arg\min_{X\in\mathbb{R}^{m\times n}} \mathcal{G}_k(X)$, such that

$$\mathcal{G}_k(X_{k+1}) \le \mathcal{G}_k(X_k) \quad (18)$$

  and there exists $Y_{k+1} \in \partial\mathcal{G}_k(X_{k+1})$ satisfying

$$\|Y_{k+1}\|_F < \rho\|X_{k+1} - X_k\|_F \quad (19)$$

  Update $r_{k+1}$, $\boldsymbol{w}_{k+1}$ by (13)

  $k \leftarrow k + 1$

**until** termination criteria met

***Remark 3.1*** The problem (17) will be solved by an iterative algorithm, whose starting point is $X_k$. Hence, the condition (18) can be satisfied easily.

***Remark 3.2*** The inexact optimal condition (19) originates from the algorithmic framework proposed in [3]. Similar inexact optimal conditions can be found in [35, 55, 56]. The rationality of this condition will be elaborated in Remark 4.18.

Before solving (16), we discuss how to solve a simpler problem as follows

$$\min_{\mathrm{rank}(X)\le r} \sum_{i=1}^{r} w_i\sigma_i(X) + \frac{1}{2}\|X-Y\|_F^2,$$

which constitutes a core step in solving (16). Let $M \in \mathbb{R}^{m\times n}$, $h = \min\{m,n\}$, and let $\boldsymbol{w} = [w_1,\ldots,w_r]^\top \in \mathbb{R}^r$ with $r \le h$. We define the following set

$$\mathcal{D}_{\boldsymbol{w}}^r(M) := \Big\{ U\Sigma_{\boldsymbol{w}}^r V^\top : (U,V) \in \mathrm{Sm}(M), \\ \Sigma_{\boldsymbol{w}}^r = \mathrm{diag}\left((\sigma_1(M)-w_1)_+,\ldots,(\sigma_r(M)-w_r)_+,0,\ldots,0\right)\Big\},$$

where $\mathrm{Sm}(M)$ is defined in (8) and $(a)_+ = \max\{a,0\}$. We have the following lemma.

**Lemma 3.3** *Let $Y \in \mathbb{R}^{m\times n}$, $h = \min\{m,n\}$, and $\boldsymbol{w} = [w_1,\ldots,w_r]^\top \in \mathbb{R}^r$ with $r \le h$. If $(\sigma_1(Y)-w_1)_+ \ge \cdots \ge (\sigma_r(Y)-w_r)_+$, then we have*

$$\mathcal{D}_{\boldsymbol{w}}^r(Y) = \arg\min_{\mathrm{rank}(X)\le r} \sum_{i=1}^{r} w_i\sigma_i(X) + \frac{1}{2}\|X-Y\|_F^2. \tag{20}$$

***Proof*** For any $X \in \mathbb{R}^{m\times n}$ with $\mathrm{rank}(X) \le r$, by Horn and Johnson [24, Corollary 7.4.1.3], we have

$$\begin{aligned}
&\sum_{i=1}^{r} w_i\sigma_i(X) + \frac{1}{2}\|X-Y\|_F^2 \\
&\ge \sum_{i=1}^{r} w_i\sigma_i(X) + \frac{1}{2}\sum_{i=1}^{r}(\sigma_i(X)-\sigma_i(Y))^2 + \frac{1}{2}\sum_{i=r+1}^{h}\sigma_i^2(Y) \\
&= \sum_{i=1}^{r}\left(\frac{1}{2}\sigma_i^2(X) + (w_i-\sigma_i(Y))\sigma_i(X)\right) + \frac{1}{2}\sum_{i=1}^{h}\sigma_i^2(Y),
\end{aligned}$$

where the equality holds if and only if there are orthogonal matrices $U$ and $V$ such that $X = U\Sigma_X V^\top$ and $Y = U\Sigma_Y V^\top$ are SVD's of $X$ and $Y$, respectively. Therefore, the matrix optimization problem (20) is equivalent to the following vector optimization problem

$$\begin{aligned}
\min_{\sigma_i(X),1\le i\le r}\ & \sum_{i=1}^{r}\frac{1}{2}\sigma_i^2(X) + (w_i-\sigma_i(Y))\sigma_i(X) \\
\text{s.t.}\ & \sigma_1(X) \ge \cdots \ge \sigma_r(X) \ge 0.
\end{aligned} \tag{21}$$

This problem is a strictly convex quadratic programming and has a unique solution. If $(\sigma_1(Y) - w_1)_+ \geq \cdots \geq (\sigma_r(Y) - w_r)_+$, the minimum of (21) is achieved by $\sigma_i(X) = (\sigma_i(Y) - w_i)_+, i = 1, \ldots, r$. This completes the proof. □

The result (20) can be viewed as a combination of the classic Eckart-Young theorem [17] and the singular value shrinkage [7, Theorem 2.1]. In addition, just like the low-rank approximation problem [17], (20) has a unique solution if and only if $\sigma_r(Y) > \sigma_{r+1}(Y)$.

The problem (16) is a weighted nuclear norm minimization with rank constraints, and its computation can follow that of the standard nuclear norm minimization (3). Refer to Yang and Yuan [50, Sect. 5] and references therein for a thorough review on the computation of (3). As shown in [50], in general, the alternating direction method of multipliers (ADMM) or linearized ADMM (LADMM) is more efficient than other methods for solving (3). Therefore, we also use (L)ADMM to solve (16).

### 3.1 Solving (16) via ADMM

By introducing an auxiliary variable $Y \in \mathbb{R}^{m\times n}$, (16) is equivalently transformed into

$$\min_{\mathrm{rank}(X)\leq r_k} \min_{Y\in\mathbb{R}^{m\times n}} \sum_{j=1}^{r_k} w_{k,j}\sigma_j(X) + \frac{\lambda}{2}\|\mathcal{A}(Y) - \boldsymbol{b}\|_2^2 + \frac{\tau}{2}\|Y - X_k\|_F^2$$
$$\text{s.t.} \quad X = Y.$$

The augmented Lagrangian function of the above problem reads

$$\mathcal{L}(X, Y, Z, \beta_k) = \sum_{j=1}^{r_k} w_{k,j}\sigma_j(X) + \langle Z, X - Y\rangle + \frac{\beta_k}{2}\|X - Y\|_F^2 + \frac{\lambda}{2}\|\mathcal{A}(Y) - \boldsymbol{b}\|_2^2 + \frac{\tau}{2}\|Y - X_k\|_F^2,$$

where $Z$ is the Lagrange multiplier and $\beta_k > 0$ is the penalty parameter. The iterative scheme of ADMM reads

$$\begin{cases} X_{t+1} \in \arg\min_{\mathrm{rank}(X)\leq r_k} \mathcal{L}(X, Y_t, Z_t, \beta_k); & (22) \\ Y_{t+1} \in \arg\min_{Y} \mathcal{L}(X_{t+1}, Y, Z_t, \beta_k); & \\ Z_{t+1} = Z_t + \beta_k(X_{t+1} - Y_{t+1}). & (23) \end{cases}$$

The two subproblems in the above algorithm are calculated as follows.

1. The $X$-subproblem (22). We can simplify (22) as

$$X_{t+1} \in \arg\min_{\mathrm{rank}(X)\leq r_k} \sum_{j=1}^{r_k} w_{k,j}\sigma_j(X) + \frac{\beta_k}{2}\left\|X - \left(Y_t - \frac{1}{\beta_k}Z_t\right)\right\|_F^2. \quad (24)$$

Combining (20) and (14) gives one solution of (24):

$$X_{t+1} \in \mathcal{D}^{r_k}_{\boldsymbol{w}_k/\beta_k}\left(Y_t - \frac{1}{\beta_k} Z_t\right). \tag{25}$$

2. The $Y$-subproblem (23). We can simplify (23) as

$$Y_{t+1} \in \arg\min_{Y\in\mathbb{R}^{m\times n}} \left(\langle Z_t, X_t - Y\rangle + \frac{\beta_k}{2}\|X_t - Y\|_F^2 + \frac{\lambda}{2}\|\mathcal{A}(Y) - \boldsymbol{b}\|_2^2 + \frac{\tau}{2}\|Y - X_k\|_F^2\right).$$

This is a least squares problem and the corresponding normal equation is

$$\left(\mathcal{I} + \frac{\lambda}{\beta_k + \tau}\mathcal{A}^*\mathcal{A}\right) Y = \frac{1}{\beta_k + \tau}\left(Z_t + \beta_k X_t + \lambda\mathcal{A}^*(\boldsymbol{b}) + \tau X_k\right), \tag{26}$$

where $\mathcal{I}$ represents the identity operator.

We summarize the whole procedure of ADMM in Algorithm 2.

**Algorithm 2:** ADMM for solving (16)

**Input**: $\mathcal{A}, \boldsymbol{b}, p, \lambda, \tau, \boldsymbol{w}_k, r_k, \beta_k$, initial value $Y_0 = X_k, Z_0 = 0$

**repeat**
- $X_{t+1} \leftarrow \mathcal{D}^{r_k}_{\boldsymbol{w_k}/\beta_k}\left(Y_t - \frac{1}{\beta_k} Z_t\right)$
- Solve $Y_{t+1}$ by (26)
- $Z_{t+1} \leftarrow Z_t + \beta_k(X_{t+1} - Y_{t+1})$
- $t \leftarrow t + 1$

**until** termination criteria met

The computation of (25) is just a partial SVD and has a time cost $O(mnr_k)$. The efficiency of Algorithm 2 depends heavily on whether we can solve (26) efficiently and this is related to the feature of $\mathcal{A} : \mathbb{R}^{m\times n} \to \mathbb{R}^d$. For a generic linear operator $\mathcal{A}$, the time cost of (26) is as high as $O(m^3n^3)$. However, if $\mathcal{A}$ has special structures or features, (26) can be solved much faster with specifical methods. For example, if there exists $A \in \mathbb{R}^{s\times m}$ such that $\mathcal{A}(X) = \mathrm{vec}(AX)$, then (26) is just a matrix equation in the form of $CY = B$, where $C \in \mathbb{R}^{m\times m}$, $B \in \mathbb{R}^{m\times n}$. More generally, if the matrix representation of $\mathcal{A}$ is block diagonal, (26) can be solved in a low cost. In the following, we will give two special cases where (26) can be solved efficiently.

### 3.1.1 The case $\mathcal{A}\mathcal{A}^* = \mathcal{I}$

In the matrix completion problem, $\mathcal{A}$ is a sampling operator collecting a fraction of entries of a matrix, and $\mathcal{A}\mathcal{A}^* = \mathcal{I}$. When $\mathcal{A}\mathcal{A}^* = \mathcal{I}$, by the Sherman-Morrison-Woodbury formula, we have

$$\left(\mathcal{I} + \frac{\lambda}{\beta_k + \tau}\mathcal{A}^*\mathcal{A}\right)^{-1} = \mathcal{I} - \frac{\lambda}{\lambda + \beta_k + \tau}\mathcal{A}^*\mathcal{A}. \tag{27}$$

By noting (27), (26) has a closed-form solution:

$$Y_{t+1} = \frac{1}{\beta_k + \tau}\left(\mathcal{I} - \frac{\lambda}{\lambda + \beta_k + \tau}\mathcal{A}^*\mathcal{A}\right)\left(Z_t + \beta_k X_t + \lambda\mathcal{A}^*(\boldsymbol{b}) + \tau X_k\right).$$

### 3.1.2 The case where $\mathcal{A}$ is a convolution operator

In signal processing, image processing and many other engineering applications, convolution has a wide applications. If $\mathcal{A}$ is a convolution operator, then (26) is a circulant linear system and can be solved efficiently by the discrete Fourier transform (DFT). If $\mathcal{A}$ is a 2D convolution operator, then the matrix representation of $\mathcal{A}^*\mathcal{A}$ is a block circulant matrix with circulant blocks and can be diagonalized by the two-dimensional DFT.

Specifically, if $\mathcal{A}$ is a convolution operator, let $\boldsymbol{y} := \mathrm{vec}(Y)$. Suppose the matrix representation of $\mathcal{A}$ is a circulant matrix $A$ whose first column is $\boldsymbol{a}_1 \in \mathbb{R}^{mn}$. Let $\boldsymbol{e}_1$ be the first standard basis. Then (26) is transformed into

$$\left((\beta_k + \tau)I + \lambda A^\top A\right)\boldsymbol{y} = \mathrm{vec}\left(Z_t + \beta_k X_t + \tau X_k\right) + \lambda A^\top \boldsymbol{b}. \tag{28}$$

Let $\mathcal{F}$ be the DFT operator, $\bar{\boldsymbol{c}}$ be the conjugate of $\boldsymbol{c}$ and "$\circ$" be the Hadamard product. Applying the DFT on both sides of (28), we obtain

$$\begin{aligned} &\left((\beta_k + \tau)\mathcal{F}(\boldsymbol{e}_1) + \lambda\overline{\mathcal{F}(\boldsymbol{a}_1)} \circ \mathcal{F}(\boldsymbol{a}_1)\right) \circ \mathcal{F}(\boldsymbol{y}) \\ &\quad = \mathcal{F}\left(\mathrm{vec}\left(Z_t + \beta_k X_t + \tau X_k\right)\right) + \lambda\overline{\mathcal{F}(\boldsymbol{a}_1)} \circ \mathcal{F}(\boldsymbol{b}) \\ \Rightarrow \quad \boldsymbol{y} &= \mathcal{F}^{-1}\left(\frac{\mathcal{F}\left(\mathrm{vec}\left(Z_t + \beta_k X_t + \tau X_k\right)\right) + \lambda\overline{\mathcal{F}(\boldsymbol{a}_1)} \circ \mathcal{F}(\boldsymbol{b})}{(\beta_k + \tau)\mathcal{F}(\boldsymbol{e}_1) + \lambda\overline{\mathcal{F}(\boldsymbol{a}_1)} \circ \mathcal{F}(\boldsymbol{a}_1)}\right). \end{aligned} \tag{29}$$

Note that $\mathcal{F}(\boldsymbol{e}_1)$ is a vector of all ones and all quantities but $Z_t + \beta_k X_t + \tau X_k$ are constant. Hence, computing $\boldsymbol{y}$ from (29) involves one DFT and one inverse DFT, once the constant quantities are computed.

The case where $\mathcal{A}$ is a 2D convolution operator is similar. Refer to [47, Sect. 2] for details.

## 3.2 Solving (16) via LADMM

If (26) is too expensive to solve, we consider using LADMM. By introducing an auxiliary variable $\boldsymbol{y} \in \mathbb{R}^d$, (16) is equivalently transformed into

$$\begin{aligned} \min_{\mathrm{rank}(X) \le r_k} \min_{\boldsymbol{y} \in \mathbb{R}^d} \quad & \sum_{j=1}^{r_k} w_{k,j}\sigma_j(X) + \frac{\lambda}{2}\|\boldsymbol{y}\|_2^2 + \frac{\tau}{2}\|X - X_k\|_F^2 \\ \text{s.t.} \quad & \mathcal{A}(X) - \boldsymbol{b} = \boldsymbol{y}. \end{aligned}$$

The augmented Lagrangian function of the above problem reads

$$\mathcal{L}(X, \boldsymbol{y}, \boldsymbol{z}, \beta_k) = \sum_{j=1}^{r_k} w_{k,j}\sigma_j(X) + \langle \boldsymbol{z}, \mathcal{A}(X) - \boldsymbol{b} - \boldsymbol{y}\rangle + \frac{\beta_k}{2}\|\mathcal{A}(X) - \boldsymbol{b} - \boldsymbol{y}\|_2^2 + \frac{\lambda}{2}\|\boldsymbol{y}\|_2^2 + \frac{\tau}{2}\|X - X_k\|_F^2,$$

where $\boldsymbol{z}$ is the Lagrange multiplier and $\beta_k > 0$ is the penalty parameter. The iterative scheme of ADMM reads

$$\begin{cases} \boldsymbol{y}_{t+1} \in \arg\min_{\boldsymbol{y}} \mathscr{L}(X_t, \boldsymbol{y}, \boldsymbol{z}_t, \beta_k); & (30) \\ X_{t+1} \in \arg\min_{\mathrm{rank}(X)\le r_k} \mathscr{L}(X, \boldsymbol{y}_{t+1}, \boldsymbol{z}_t, \beta_k); & \\ \boldsymbol{z}_{t+1} = \boldsymbol{z}_t + \beta_k(\mathcal{A}(X_{t+1}) - \boldsymbol{b} - \boldsymbol{y}_{t+1}). & (31) \end{cases}$$

The $\boldsymbol{y}$-subproblem (30) can be simplified as

$$\boldsymbol{y}_{t+1} \in \arg\min_{\boldsymbol{y}} \frac{\lambda}{2}\|\boldsymbol{y}\|_2^2 + \frac{\beta}{2}\left\|\boldsymbol{y} - \left(\mathcal{A}(X_t) - \boldsymbol{b} + \frac{1}{\beta_k}\boldsymbol{z}_t\right)\right\|_2^2,$$

which has a closed-form solution given by

$$\boldsymbol{y}_{t+1} = \frac{\beta_k}{\lambda + \beta_k}\left(\mathcal{A}(X_t) - \boldsymbol{b} + \frac{1}{\beta_k}\boldsymbol{z}_t\right).$$

The $X$-subproblem (31) can be simplified as

$$X_{t+1} \in \arg\min_{\mathrm{rank}(X)\le r_k} \sum_{j=1}^{r_k} w_{k,j}\sigma_j(X) + \frac{\beta_k}{2}\|\mathcal{A}(X) - \boldsymbol{c}_t\|_2^2 + \frac{\tau}{2}\|X - X_k\|_F^2, \quad (32)$$

where $\boldsymbol{c}_t = \boldsymbol{b} + \boldsymbol{y}_{t+1} - \frac{1}{\beta_k}\boldsymbol{z}_t$. Like (12), we consider the proximal linearization:

$$\frac{1}{2}\|\mathcal{A}(X) - \boldsymbol{c}_t\|_2^2 \approx \frac{1}{2}\|\mathcal{A}(X_t) - \boldsymbol{c}_t\|_2^2 + \langle X - X_t, G_t\rangle + \frac{\eta}{2}\|X - X_t\|_F^2, \quad (33)$$

where

$$G_t = \mathcal{A}^*(\mathcal{A}(X_t) - \boldsymbol{c}_t) = \mathcal{A}^*\left(\mathcal{A}(X_t) - \boldsymbol{b} - \boldsymbol{y}_{t+1} + \frac{1}{\beta_k}\boldsymbol{z}_t\right) \quad (34)$$

is the gradient of $\frac{1}{2}\|\mathcal{A}(X) - \boldsymbol{c}_t\|_2^2$ at $X_t$ and $\eta > 0$ is the proximal parameter. We will choose

$$\eta \ge \|\mathcal{A}^*\mathcal{A}\|_2 \quad (35)$$

to guarantee the right hand of (33) is an upper bound of $\frac{1}{2}\|\mathcal{A}(X)-\boldsymbol{c}_t\|_2^2$ near $X_t$. Substituting (33) into (32) and deleting some constants, we obtain the following proximal linearization version of (32)

$$X_{t+1}\in\arg\min_{\mathrm{rank}(X)\le r_k}\sum_{j=1}^{r_k}w_{k,j}\sigma_j(X)+\frac{\beta_k\eta+\tau}{2}\left\|X-\frac{\beta_k\eta X_t-\beta_k G_t+\tau X_k}{\beta_k\eta+\tau}\right\|_F^2,$$

whose solution is

$$X_{t+1}=\mathcal{D}^{r_k}_{\boldsymbol{w}_k/(\beta_k\eta+\tau)}\left(\frac{\beta_k\eta X_t-\beta_k G_t+\tau X_k}{\beta_k\eta+\tau}\right),\tag{36}$$

where $G_t$ and $\eta$ are given in (34) and (35).

We summarize the whole procedure of LADMM in Algorithm 3.

**Algorithm 3:** LADMM for solving (16)

**Input**: $\mathcal{A}, \boldsymbol{b}, p, \lambda, \tau, \eta, \boldsymbol{w}_k, r_k, \beta_k$, initial value $X_0=X_k, z_0=0$

**repeat**

- $\boldsymbol{y}_{t+1}\leftarrow\frac{\beta_k}{\lambda+\beta_k}\left(\mathcal{A}(X_t)-\boldsymbol{b}+\frac{1}{\beta_k}\boldsymbol{z}_t\right)$
- $X_{t+1}\leftarrow D^{r_k}_{\boldsymbol{w}_k/(\beta_k\eta+\tau)}\left(\frac{\beta_k\eta X_t-\beta_k G_t+\tau X_k}{\beta_k\eta+\tau}\right)$; see (36)
- $\boldsymbol{z}_{t+1}\leftarrow\boldsymbol{z}_t+\beta_k(\mathcal{A}(X_{t+1})-\boldsymbol{b}-\boldsymbol{y}_{t+1})$
- $t\leftarrow t+1$

**until** termination criteria met

## 4 Convergence analysis

We discuss the convergence properties of PLM in Algorithm 1. In Sect. 4.1, we will establish the convergence properties of $\{r_k\}$ and $\{\mathcal{P}(X_k)\}$, where $\mathcal{P}(X)$ is the objective function defined in (11). The global convergence of $\{X_k\}$ will be discussed in Sect. 4.3.

### 4.1 Basic convergence properties of PLM

**Proposition 4.1** *Let $\{X_k\}$ be generated by Algorithm 1. Then,*

1. *The objective function sequence $\{\mathcal{P}(X_k)\}$ satisfies*

$$\frac{\tau}{2}\|X_{k+1}-X_k\|_F^2\le\mathcal{P}(X_k)-\mathcal{P}(X_{k+1})\tag{37}$$

   *and converges;*
2. *The iterate sequence $\{X_k\}$ is bounded and satisfies*

$$\lim_{k\to\infty}\|X_k-X_{k+1}\|_F=0.\tag{38}$$

***Proof*** For $X \in \mathbb{R}^{m\times n}$ with $\mathrm{rank}(X) \le r_k$, it follows from (15) that

$$\mathcal{P}(X) + \frac{\tau}{2}\|X - X_k\|_F^2 \le \mathcal{H}_k(X) + \sum_{j=1}^{r_k}(\sigma_{k,j}^p - w_{k,j}\sigma_{k,j}). \tag{39}$$

Note that $\delta_{\mathscr{M}_{\le r_k}}(X_k) = \delta_{\mathscr{M}_{\le r_k}}(X_{k+1}) = 0$. We have

$$\mathcal{H}_k(X_{k+1}) = \mathcal{H}_k(X_{k+1}) + \delta_{\mathscr{M}_{\le r_k}}(X_{k+1}) \overset{(18)}{\le} \mathcal{H}_k(X_k) + \delta_{\mathscr{M}_{\le r_k}}(X_k) = \mathcal{H}_k(X_k). \tag{40}$$

Combining the above results, we have

$$\begin{aligned}\mathcal{P}(X_{k+1}) + \frac{\tau}{2}\|X_{k+1} - X_k\|_F^2 &\overset{(39)}{\le} \mathcal{H}_k(X_{k+1}) + \sum_{j=1}^{r_k}(\sigma_{k,j}^p - w_{k,j}\sigma_{k,j}) \\ &\overset{(40)}{\le} \mathcal{H}_k(X_k) + \sum_{j=1}^{r_k}(\sigma_{k,j}^p - w_{k,j}\sigma_{k,j}) \\ &\overset{(16)}{=} \sum_{j=1}^{r_k}\sigma_{k,j}^p + \frac{\lambda}{2}\|\mathcal{A}(X_k) - \boldsymbol{b}\|_2^2 = \mathcal{P}(X_k),\end{aligned}$$

which proves (37).

Combining (37) and the fact that $\mathcal{P}(X_k) \ge 0$ yields the convergence of $\{\mathcal{P}(X_k)\}$. Since $\mathcal{P}(X)$ is coercive, it follows that $\{X_k\}$ is bounded. Let $c > 1$ be an integer. Summing (37) from $k = 0$ to $c - 1$ we obtain

$$\sum_{k=0}^{c-1}\|X_{k+1} - X_k\|_F^2 \le \frac{2}{\tau}\left(\mathcal{P}(X_0) - \mathcal{P}(X_c)\right) \le \frac{2}{\tau}\mathcal{P}(X_0).$$

Taking the limit as $c \to \infty$ gives $\sum_{k=0}^{\infty}\|X_{k+1} - X_k\|_F^2 < +\infty$, and hence yields (38). □

For the convergence of $\{r_k\}$, we have the following result.

**Lemma 4.2** *The sequence $\{r_k\}$ satisfies $r_{k+1} \le r_k$ and converges within finite steps.*

***Proof*** From Algorithm 1 we know that $r_{k+1} \le r_k$. Since $r_k \in \{0, 1, \ldots, r_0\}$, it follows that $\{r_k\}$ will converge within finite steps. □

The preceding lemma implies that

$$\exists \widetilde{r}, \widetilde{k} \text{ such that } r_k = \widetilde{r} \quad \forall k \ge \widetilde{k}. \tag{41}$$

Let

$$\mathscr{M}_r := \{X \in \mathbb{R}^{m\times n} : \operatorname{rank}(X) = r\}$$

denote the manifold of fixed-rank matrices and let $\delta_{\mathscr{M}_{\widetilde{r}}}(X)$ be the indicator function of $\mathscr{M}_{\widetilde{r}}$. When $k \geq \widetilde{k}$, $\mathcal{G}_k(X)$ defined in (17) has the following form:

$$\mathcal{G}_k(X) = \sum_{j=1}^{\widetilde{r}} w_{k,j}\sigma_j(X) + \frac{\lambda}{2}\|\mathcal{A}(X) - \boldsymbol{b}\|_2^2 + \frac{\tau}{2}\|X - X_k\|_F^2 + \delta_{\mathscr{M}_{\widetilde{r}}}(X). \quad (42)$$

### 4.2 Singular value functions and KL functions

To prove the global convergence of $\{X_k\}$, we need some knowledge on singular value functions and KL functions.

**Definition 4.3** A function $f : \mathbb{R}^n \to [-\infty, +\infty]$ is an *absolutely symmetric function* if $f(\boldsymbol{x}) = f(P|\boldsymbol{x}|)$ for every $\boldsymbol{x} \in \mathbb{R}^n$ and every permutation matrix $P \in \mathbb{R}^{n\times n}$.

Absolutely symmetric functions are extensions of symmetric gauge functions, which play a special role in unitarily invariant norms. Refer to Horn and Johnson [24, Chapter 7.4] and Lewis and Sendov [31, Sect. 5] for details.

**Definition 4.4** A function $\mathcal{F} : \mathbb{R}^{m\times n} \to [-\infty, +\infty]$ is a *singular value function* if there is an absolutely symmetric function $f$ such that $\mathcal{F}(X) = f(\boldsymbol{\sigma}(X))$.

The subdifferential of a singular value function has the following formula.

**Theorem 4.5** (See [31, Theorem 7.1]) *The subdifferential of a singular value function $\mathcal{F}(X) = f(\boldsymbol{\sigma}(X))$ is given by the formula*

$$\partial\mathcal{F}(X) = \left\{U\operatorname{diag}(\boldsymbol{y})V^\top : (U, V) \in \operatorname{Sm}(X), \boldsymbol{y} \in \partial f(\boldsymbol{\sigma}(X))\right\}. \quad (43)$$

The following lemma gives the subdifferential of the Schatten $p$-quasi-norm.

**Lemma 4.6** *Define $\mathcal{S}(X) : \mathbb{R}^{m\times n} \to (-\infty, +\infty)$ by $\mathcal{S}(X) = \|X\|_{*,p}^p (0 < p < 1)$ and let $h = \min\{m, n\}$. For a matrix $\hat{X} \in \mathbb{R}^{m\times n}$ with $\operatorname{rank}(\hat{X}) = r$, we have*

$$\begin{aligned}\partial\mathcal{S}(\hat{X}) = \Bigg\{U\begin{bmatrix} D & 0 \\ 0 & \operatorname{diag}(\boldsymbol{a})\end{bmatrix}V^\top : (U, V) \in \operatorname{Sm}(\hat{X}), \boldsymbol{a} \in \mathbb{R}^{h-r},\\ D = \operatorname{diag}(p\sigma_1^{p-1}(\hat{X}), \ldots, p\sigma_r^{p-1}(\hat{X})) \in \mathbb{R}^{r\times r}\Bigg\}.\end{aligned} \quad (44)$$

***Proof*** Define $f(\boldsymbol{x}) : \mathbb{R}^h \to \mathbb{R}$ by $f(\boldsymbol{x}) = \sum_{i=1}^h |x_i|^p$. Then $f$ is an absolutely symmetric function and $\mathcal{S}(X) = f(\boldsymbol{\sigma}(X))$. Note that

$$\partial|x|^p = \begin{cases}\{p|x|^{p-1}\}, & \text{if } x \neq 0\\ (-\infty, +\infty), & \text{otherwise.}\end{cases}$$

Then the subdifferential of $\mathcal{S}(X)$ at $\hat{X}$ is obtained by (43). □

In (16) or (17), there is one term $\sum_{j=1}^{r_k} w_{k,j}\sigma_j(X)$. For convenience, we define $\mathcal{W}(X) : \mathbb{R}^{m\times n} \to (-\infty, +\infty)$ by

$$\mathcal{W}(X) = \sum_{i=1}^{r} b_i\sigma_i(X), \tag{45}$$

where $r \le \min\{m, n\}$ and $b_i > 0, i = 1, \ldots, r$. The function $\mathcal{W}(X)$ can be regarded as a generalization of Ky Fan $r$-norms; see [24, Chapter 7.4.8] for details. Generally, it is difficult to give an explicit formula of $\partial\mathcal{W}(X)$. However, when the first $r + 1$ singular values of $X$ are strictly decreasing, we have the following result.

**Lemma 4.7** *Let $\mathcal{W}(X)$ be defined in (45) and $h = \min\{m, n\}$. For a matrix $\hat{X} \in \mathbb{R}^{m\times n}$ with $\sigma_1(\hat{X}) > \cdots > \sigma_r(\hat{X}) > \sigma_{r+1}(\hat{X})$, where $\sigma_{h+1}(\hat{X}) := 0$, we have*

$$\partial\mathcal{W}(\hat{X}) = \left\{ U\begin{bmatrix} D & 0\\ 0 & 0\end{bmatrix}V^\top : (U, V) \in \mathrm{Sm}(\hat{X}),\ D = \mathrm{diag}(b_1, \ldots, b_r) \in \mathbb{R}^{r\times r}\right\}. \tag{46}$$

***Proof*** Given $\boldsymbol{x} \in \mathbb{R}^h$, we write $\underline{\boldsymbol{x}}$ for the vector in $\mathbb{R}^h_+$ with the same entries as $|\boldsymbol{x}|$ arranged in nonincreasing order. Define

$$f(\boldsymbol{x}) = \sum_{i=1}^{r} b_i\underline{x}_i. \tag{47}$$

Then $f$ is an absolutely symmetric function and $\mathcal{W}(X) = f(\boldsymbol{\sigma}(X))$.

Given $\hat{\boldsymbol{x}} \in \mathbb{R}^h_+$, suppose $\hat{x}_1 > \cdots > \hat{x}_r > \hat{x}_d, d = r + 1, \ldots, h$, where $\hat{x}_{h+1} := 0$. Then, $f(\hat{\boldsymbol{x}}) = \sum_{i=1}^{r} b_i\hat{x}_i$. There exists $\varepsilon > 0$ such that for each $\boldsymbol{y} \in \mathbb{R}^h$ with $\|\boldsymbol{y}\|_2 < \varepsilon$, $\boldsymbol{z} := \hat{\boldsymbol{x}} + \boldsymbol{y}$ satisfies $\underline{z}_i = z_i, i = 1, \ldots, r$. Hence, $f(\boldsymbol{z}) = \sum_{i=1}^{r} b_i z_i$. It follows that $f$ is differentiable at $\hat{\boldsymbol{x}}$ and

$$\nabla f(\hat{\boldsymbol{x}}) = [b_1, \ldots, b_r, 0, \ldots, 0]^\top.$$

Then the subdifferential of $\mathcal{W}(X)$ at $\hat{X}$ follows from (43). □

With the same condition of the preceding lemma, we can give an explicit formula of the subdifferential of $\mathcal{W}(X) + \delta_{\mathscr{M}_r}(X)$.

**Lemma 4.8** *Define $\mathcal{R}(X) : \mathbb{R}^{m\times n} \to (-\infty, +\infty]$ by $\mathcal{R}(X) := \mathcal{W}(X) + \delta_{\mathscr{M}_r}(X)$, where $\mathcal{W}(X)$ is defined in (45) and let $h = \min\{m, n\}$. For a matrix $\hat{X} \in \mathbb{R}^{m\times n}$ with $\mathrm{rank}(\hat{X}) = r$ and $\sigma_1(\hat{X}) > \cdots > \sigma_r(\hat{X})$, we have*

$$\begin{aligned}\partial\mathcal{R}(\hat{X}) = \Bigg\{ U\begin{bmatrix} D & 0\\ 0 & \mathrm{diag}(\boldsymbol{a})\end{bmatrix}V^\top : (U, V) \in \mathrm{Sm}(\hat{X}), \boldsymbol{a} \in \mathbb{R}^{h-r}\\ D = \mathrm{diag}(b_1, \ldots, b_r) \in \mathbb{R}^{r\times r}\Bigg\}.\end{aligned} \tag{48}$$

***Proof*** First, it is well know that $\mathscr{M}_r$ is a smooth manifold; see, e.g., [46, Proposition 2.1] and [25, Theorem 2.1] for the formulae of tangent cone and the normal cone of $\mathscr{M}_r$, in terms of SVD. Denote by $\mathscr{N}_{\mathscr{M}_r}(X)$ the normal cone of $\mathscr{M}_r$ at $\hat{X}$. By [42, Exercise 8.14], we have

$$\partial\delta_{\mathscr{M}_r}(\hat{X}) = \mathscr{N}_{\mathscr{M}_r}(\hat{X}) = \left\{ U \begin{bmatrix} 0_{r\times r} & 0 \\ 0 & \mathrm{diag}(\boldsymbol{a}) \end{bmatrix} V^\top : (U, V) \in \mathrm{Sm}(\hat{X}), \boldsymbol{a} \in \mathbb{R}^{h-r} \right\}. \tag{49}$$

Clearly, $\delta_{\mathscr{M}_r}(X)$ is regular at $\hat{X}$. From the proof of (46), we know that $f(\boldsymbol{x})$ defined in (47) is differentiable at $\boldsymbol{\sigma}(\hat{X})$ and of course regular at $\boldsymbol{\sigma}(\hat{X})$. By Lewis and Sendov [31, Corollary 7.5], $\mathcal{W}(X)$ is regular at $\hat{X}$. Consequently, by Rockafellar and Wets [42, Corollary 10.9], $\mathcal{R}(X)$ is regular at $\hat{X}$ and

$$\partial\mathcal{R}(\hat{X}) = \partial\mathcal{W}(\hat{X}) + \partial\delta_{\mathscr{M}_r}(\hat{X}). \tag{50}$$

Combining (46), (49) and (50) yields (48). □

We next recall the Kurdyka–Łojasiewicz (KL) property. This property has been widely used in establishing convergence and analyzing the convergence rate of first-order methods; see, for example, [1–3, 5, 33, 51] First we introduce some notation and definitions. We use $\mathbb{E}$ to denote a Euclidean space and $\|\cdot\|$ to denote the Euclidean norm. For any subset $\mathscr{S} \subseteq \mathbb{E}$ and any point $u \in \mathbb{E}$, the distance from $u$ to $\mathscr{S}$ is defined by $\mathrm{dist}(u, \mathscr{S}) := \inf\{\|v - u\| : v \in \mathscr{S}\}$. An extended-real-valued function $f : \mathbb{E} \to (-\infty, +\infty]$ is said to be proper if $\{x \in \mathbb{E} : f(x) < +\infty\} \neq \emptyset$. Let $\eta \in (0, \infty]$. We denote by $\Phi_\eta$ the class of all concave and continuous functions $\varphi : [0, \eta) \to (0, +\infty]$ which satisfy the following conditions

(i) $\varphi(0) = 0$;
(ii) $\varphi$ is continuous differentiable on $(0, \eta)$;
(iii) $\varphi' > 0$ on $(0, \eta)$.

**Definition 4.9** (*KL property and KL functions*) A proper and lower semicontinuous function $f : \mathbb{E} \to (-\infty, +\infty]$ is said to have the KL property at $\hat{x} \in \mathrm{dom}\,\partial f := \{x \in \mathbb{E} : \partial f(x) \neq \emptyset\}$ if there exist $\eta \in (0, +\infty]$, a neighborhood $\mathscr{V}$ of $\hat{x}$, and a function $\varphi \in \Phi_\eta$ such that for all $x \in \mathscr{V} \cap \{y \in \mathbb{E} : f(\hat{x}) < f(y) < f(\hat{x}) + \eta\}$, the KL inequality holds:

$$\varphi'(f(x) - f(\hat{x}))\mathrm{dist}(0, \partial f(x)) \geq 1.$$

If $f$ satisfies the KL property at each point of $\mathrm{dom}\,\partial f$, $f$ is called a KL function.

A rich class of KL functions used widely in practice belongs to a special structure called o-minimal structure. Related facts can be found in [2, 45]. The KL properties of some related functions are proved in the Appendix.

**Proposition 4.10** *Let $\mathcal{F} : \mathbb{R}^{m\times n} \to (-\infty, +\infty]$, $h = \min\{m, n\}$ and $f : \mathbb{R}^h \to (-\infty, +\infty]$ be an absolutely symmetric function. Suppose $\mathcal{F}(X) = f(\boldsymbol{\sigma}(X))$. Then $\mathcal{F}$ is a KL function if and only if $f$ is a KL function.*

***Proof*** First, we assume that $f$ is a KL function. For any $\hat{X} \in \operatorname{dom} \partial \mathcal{F}$, by the subdifferential formula (43), $\boldsymbol{\sigma}(\hat{X}) \in \operatorname{dom} \partial f$. Since $f$ is a KL function, there exist $\eta \in (0, +\infty]$, a neighborhood $\mathscr{V}$ of $\boldsymbol{\sigma}(\hat{X})$, and a function $\varphi \in \Phi_\eta$ such that for all

$$\boldsymbol{x} \in \mathscr{V} \cap \{\boldsymbol{y} \in \mathbb{R}^h : \mathcal{F}(\hat{X}) < f(\boldsymbol{y}) < \mathcal{F}(\hat{X}) + \eta\},$$

the KL inequality holds:

$$\varphi'(f(\boldsymbol{x}) - \mathcal{F}(\hat{X}))\operatorname{dist}(0, \partial f(\boldsymbol{x})) \geq 1,$$

where $\mathcal{F}(\hat{X}) = f(\boldsymbol{\sigma}(\hat{X}))$ has been used. By the continuity of singular values (see [24, Corollary 7.4.1.3]), there exists a neighborhood $\mathscr{U}$ of $\hat{X}$ such that $\boldsymbol{\sigma}(X) \in \mathscr{V}$ for all $X \in \mathscr{U}$. Hence, for any $X \in \mathscr{U} \cap \{Y \in \mathbb{R}^{m\times n} : \mathcal{F}(\hat{X}) < \mathcal{F}(Y) < \mathcal{F}(\hat{X}) + \eta\}$, we have $\boldsymbol{\sigma}(X) \in \mathscr{V} \cap \{\boldsymbol{y} \in \mathbb{R}^h : \mathcal{F}(\hat{X}) < f(\boldsymbol{y}) < \mathcal{F}(\hat{X}) + \eta\}$ and

$$\begin{aligned}&\varphi'(\mathcal{F}(X) - \mathcal{F}(\hat{X}))\operatorname{dist}(0, \partial\mathcal{F}(X))\\ &\quad = \varphi'(f(\boldsymbol{\sigma}(X)) - \mathcal{F}(\hat{X}))\operatorname{dist}(0, \partial f(\boldsymbol{\sigma}(X))) \geq 1,\end{aligned}$$

where $\operatorname{dist}(0, \partial\mathcal{F}(X)) = \operatorname{dist}(0, \partial f(\boldsymbol{\sigma}(X)))$ follows from (43). By definition, $\mathcal{F}$ is a KL function.

To show the opposite, we can fix $U \in \mathrm{St}_m^m, V \in \mathrm{St}_n^n$. Then $f(\boldsymbol{x}) = \mathcal{F}(U \operatorname{diag}(\boldsymbol{x}) V^\top)$ for all $\boldsymbol{x} \in \mathbb{R}^h$ and the KL property of $f$ can be verified by definition. □

By Proposition 4.10 and Examples 7.5–7.7, $\mathcal{P}(X)$ defined in (11) and $\mathcal{G}_k(X)$ defined in (17) are both KL functions.

### 4.3 Global convergence of PLM and remarks on nonconvex ADMM

Now we start to prove the global convergence of $\{X_k\}$. We need the following assumption.

**Assumption 4.11** The sequence $\{X_k\}$ generated by Algorithm 1 satisfies

$$\sigma_{k,1} > \cdots > \sigma_{k,\widetilde{r}} \quad \forall k \geq \widetilde{k},$$

where $\widetilde{r}, \widetilde{k}$ are defined in (41).

***Remark 4.12*** In the convergence analysis based on the KL property for proximal linearization methods, the function that is linearized is usually assumed to have Lipschitz gradient. See, for example, [5, Assumption 2] and [55, Assumption 2.2]. In (12), the function that is linearized is $f(z) = z^p$, which does not have Lipschitz gradient on $(0, +\infty)$. Our strategy is to prove the lower bound property of the nonzero singular values of $\{X_k\}$; see Lemma 4.13. Then, $f(z)$ also has Lipschitz gradient in the domain where the nonzero singular values lie; see (55). To prove Lemma 4.13 and obtain a subdifferential lower bound for the iterates gap (Lemma 4.14), we need the explicit formula of $\partial\mathcal{G}_k(X)$, which is ensured by Assumption 4.11.

In Theorem 2.1, we obtain a lower bound property for the original problem (6). Since $\{X_k\}$ is generated by approximately solving the proximal linearization (17), Theorem 2.1 does not hold for $\{X_k\}$. In the following, we give a similar result with respect to $\{X_k\}$ .

**Lemma 4.13** *Suppose that Assumption 4.11 holds. Let $\{X_k\}$ be generated by Algorithm 1 and $k \geq \widetilde{k}$. Then, there is a constant $\vartheta > 0$ such that*

$$\sigma_r(X_k) \geq \vartheta \quad \text{for all } r = 1, \ldots, \widetilde{r}.$$

***Proof*** Define $D_k := \mathrm{diag}(w_{k,1}, \ldots, w_{k,\widetilde{r}})$. By noting (42) and (48), the subdifferential of $\mathcal{G}_k(X)$ at $X_{k+1}$ is

$$\partial \mathcal{G}_k(X_{k+1}) = \left\{ U \begin{bmatrix} D_k & 0 \\ 0 & \mathrm{diag}(\boldsymbol{a}) \end{bmatrix} V^\top + \lambda \mathcal{A}^*(\mathcal{A}(X_{k+1}) - \boldsymbol{b}) + \tau(X_{k+1} - X_k) : (U, V) \in \mathrm{Sm}(X_{k+1}), \boldsymbol{a} \in \mathbb{R}^{h-\widetilde{r}} \right\}. \tag{51}$$

There exist $Y_{k+1} \in \partial \mathcal{G}_k(X_{k+1})$ and $(U, V) \in \mathrm{Sm}(X_{k+1})$ such that

$$U \begin{bmatrix} D_k & 0 \\ 0 & \mathrm{diag}(\boldsymbol{a}) \end{bmatrix} V^\top = Y_{k+1} - \lambda \mathcal{A}^*(\mathcal{A}(X_{k+1}) - \boldsymbol{b}) - \tau(X_{k+1} - X_k).$$

By noting the condition (19), we have

$$\begin{aligned} p\sigma_{\widetilde{r}}^{p-1}(X_k) = w_{k,\widetilde{r}} \leq \|\boldsymbol{w}_k\|_2 = \|D_k\|_F \leq \left\| \begin{bmatrix} D_k & 0 \\ 0 & \mathrm{diag}(\boldsymbol{a}) \end{bmatrix} \right\|_F \\ \leq \|Y_{k+1}\|_F + \lambda \|\mathcal{A}^*(\mathcal{A}(X_{k+1}) - \boldsymbol{b})\|_F + \tau \|X_{k+1} - X_k\|_F \\ \leq (\rho + \tau)\|X_{k+1} - X_k\|_F + \lambda \|\mathcal{A}^*(\mathcal{A}(X_{k+1}) - \boldsymbol{b})\|_F. \end{aligned}$$

Since $\{X_k\}$ is bounded (see Proposition 4.1), there is a constant $\alpha$ such that

$$p\sigma_{\widetilde{r}}^{p-1}(X_k) \leq \alpha \Rightarrow \sigma_{\widetilde{r}}(X_k) \geq \left(\frac{p}{\alpha}\right)^{\frac{1}{1-p}} =: \vartheta,$$

which completes the proof. □

**Lemma 4.14** *Suppose that Assumption 4.11 holds. Let $\{X_k\}$ be generated by Algorithm 1 and $k \geq \widetilde{k}$. Define $D_k := \mathrm{diag}(w_{k,1}, \ldots, w_{k,\widetilde{r}})$. Then there exist $Y_{k+1} \in \partial \mathcal{G}_k(X_{k+1})$, $(U, V) \in \mathrm{Sm}(X_{k+1})$ and $\mu > 0$ such that*

$$B_{k+1} := Y_{k+1} + U \begin{bmatrix} D_{k+1} - D_k & 0 \\ 0 & 0 \end{bmatrix} V^\top + \tau(X_k - X_{k+1}) \in \partial \mathcal{P}(X_{k+1}) \tag{52}$$

*and*

$$\|B_{k+1}\|_F \leq \mu \|X_{k+1} - X_k\|_F.$$

***Proof*** Note that $\operatorname{rank}(X_k) = \widetilde{r}$ for all $k \geq \widetilde{k}$. By using (44), we have

$$\partial\mathcal{P}(X_{k+1}) = \left\{ U \begin{bmatrix} D_{k+1} & 0 \\ 0 & \operatorname{diag}(\boldsymbol{a}) \end{bmatrix} V^\top + \lambda\mathcal{A}^*(\mathcal{A}(X_{k+1}) - \boldsymbol{b}) : (U, V) \in \operatorname{Sm}(X_{k+1}), \boldsymbol{a} \in \mathbb{R}^{h-\widetilde{r}} \right\}. \tag{53}$$

By noting (51), there exist $Y_{k+1} \in \partial\mathcal{G}_k(X_{k+1})$ and $(U, V) \in \operatorname{Sm}(X_{k+1})$ such that

$$\lambda\mathcal{A}^*(\mathcal{A}(X_{k+1}) - \boldsymbol{b}) = Y_{k+1} + \tau(X_k - X_{k+1}) - U \begin{bmatrix} D_k & 0 \\ 0 & 0 \end{bmatrix} V^\top. \tag{54}$$

Combining (53) and (54) yields (52).

Now we estimate the norm of $B_{k+1}$. First, by the condition (19), we have

$$\|B_{k+1}\|_F \leq \|D_{k+1} - D_k\|_F + (\rho + \tau)\|X_{k+1} - X_k\|_F.$$

By Lemma 4.13, $\sigma_{k,r} \geq \vartheta$ for all $k \geq \widetilde{k}$ and $r = 1, \ldots, \widetilde{r}$. Consider the function $[\vartheta, +\infty) \to \mathbb{R}$ defined by $z \mapsto z^{p-1}$. For any $z_2 > z_1 \geq \vartheta$, by the mean value theorem, there exists $z_0 \in (z_1, z_2)$ such that

$$|z_1^{p-1} - z_2^{p-1}| = (1-p)z_0^{p-2}|z_1 - z_2| \leq (1-p)\vartheta^{p-2}|z_1 - z_2|. \tag{55}$$

With this fact, we have

$$\begin{aligned} \|D_{k+1} - D_k\|_F^2 = \|\boldsymbol{w}_{k+1} - \boldsymbol{w}_k\|_2^2 &= p^2 \sum_{r=1}^{\widetilde{r}} \left| \sigma_{k,r}^{p-1} - \sigma_{k+1,r}^{p-1} \right|^2 \\ &\leq p^2(1-p)^2\vartheta^{2p-4} \sum_{r=1}^{\widetilde{r}} \left| \sigma_{k,r} - \sigma_{k+1,r} \right|^2 \leq p^2(1-p)^2\vartheta^{2p-4}\|X_{k+1} - X_k\|_F^2, \end{aligned}$$

where the last inequality follows from Horn and Johnson [24, Corollary 7.4.1.3]. Therefore, we have

$$\begin{aligned} \|B_{k+1}\|_F &\leq p(1-p)\vartheta^{p-2}\|X_{k+1} - X_k\|_F \\ &\quad + (\rho + \tau)\|X_{k+1} - X_k\|_F =: \mu\|X_{k+1} - X_k\|_F, \end{aligned}$$

which completes the proof. □

Now we can use the framework in [3] to establish the global convergence. To match the theme of this paper, we present the framework of Attouch et al. [3] in the form matrix functions as follows.

**Lemma 4.15** (See [3, Theorem 2.9]) *Suppose $\mathcal{F}(X) : \mathbb{R}^{m\times n} \to (-\infty, +\infty]$ is a KL function. Let $\{X_k\}$ be a sequence of $\mathbb{R}^{m\times n}$. Suppose there exist $a > 0, b > 0$ such that*

**H**1. $\mathcal{F}(X_k) - \mathcal{F}(X_{k+1}) \geq a\|X_k - X_{k+1}\|^2$ *for all $k$.*

**H**2. *There exists* $Y_{k+1} \in \partial \mathcal{F}(X_{k+1})$ *such that*

$$\|Y_{k+1}\|_F \leq b\|X_k - X_{k+1}\|_F.$$

**H**3. *There exists a subsequence* $\{X_{k_j}\}$ *and* $\widetilde{X}$ *such that*

$$X_{k_j} \to \widetilde{X} \quad \text{and} \quad \mathcal{F}(X_{k_j}) \to \mathcal{F}(\widetilde{X}), \quad \text{as } j \to \infty.$$

*Then* $\lim_{k\to\infty} X_k = \widetilde{X}$, *and* $\widetilde{X}$ *is a critical point of* $\mathcal{F}(X)$.

**Theorem 4.16** *Suppose that Assumption 4.11 holds. Let* $\{X_k\}$ *be a sequence generated by PLM. Then* $\{X_k\}$ *converges to a critical point of* $\mathcal{P}(X)$.

***Proof*** Proposition 4.1 and Lemma 4.14 show that conditions **H**1 and **H**2 are satisfied. Combining the fact that $\{X_k\}$ is bounded (see Proposition 4.1) and the fact that $\mathcal{P}(X)$ is continuous gives that **H**3 is satisfied. In addition, $\mathcal{P}(X)$ is a KL function. These together with Lemma 4.15 demonstrate the assertion. □

The convergence analysis of nonconvex (L)ADMM can also be established by using the KL property; see [6, 32, 48]. Note that $\mathcal{G}_k(X)$ is a KL function. Algorithm 2 is in the framework of Li and Pong [32], Wang et al. [48] and Boţ and Nguyen [6], while Algorithm 3 is only in the framework of Wang et al. [48]. We summarize the results in the following proposition.

**Proposition 4.17** *Let* $\{X_t\}$ *be the sequence in Algorithm 2 or Algorithm 3. Then for any sufficiently large* $\beta_k$, $\{X_t\}$ *converges globally to a critical point of* $\mathcal{G}_k(X)$ *defined in* (17).

The lower bounds of $\beta_k$ have been given in all these three references. Note that these bounds are only for sufficient conditions of convergence, and computing them is not very easy. In practice, setting $\beta_k$ empirically is more efficient.

***Remark 4.18*** Proposition 4.17 implies that, after a sufficiently large number of iterations of (L)ADMM, the condition (19) will be satisfied. However, checking (19) is not easy. Like [35, 55, 56], in practical computation, we adopt the relative change as stopping criterion in the inner loop; see the next section. It turns out numerically that PLM shows good convergence behaviour with this stopping criterion.

## 5 Experiments

We evaluate the performance of PLM in this section. All experiments are performed on MATLAB R2016a on a laptop (Intel Core i5-6300HQ CPU @ 2.30GHz, 8.00G RAM). We measure the quality of a solution $X$ by its relative error (RErr) to the true low-rank matrix $\hat{X}$:

$$\text{RErr} := \frac{\|X - \hat{X}\|_F}{\|\hat{X}\|_F}.$$

It turns out that the performance of a low-rank matrix recovery algorithm is related to the distribution of singular values; see [28, 49]. We will test low-rank matrices with singular values in different distributions. To obtain a matrix $Z \in \mathbb{R}^{m\times n}$ with $\mathrm{rank}(Z) = r_{\mathrm{true}}$, firstly we generate $M_L \in \mathrm{St}^m_{r_{\mathrm{true}}}$ and $M_R \in \mathrm{St}^n_{r_{\mathrm{true}}}$. Then we generate a vector $\boldsymbol{s} \in \mathbb{R}^{r_{\mathrm{true}}}$ with a given distribution, which will be specified later. The yielded low-rank matrix is

$$Z = cM_L\,\mathrm{diag}(\boldsymbol{s})M_R^\top,$$

where $c$ is a random scalar. We consider two types of distributions of singular values:

1. Uniform distribution in $[a, b]$: $\boldsymbol{s}$ is generated by the Matlab command "$a + (b - a)*\mathtt{rand}(r_{\mathrm{true}},1)$".
2. Power-law distribution with power $a$: $\boldsymbol{s}$ has entries $s_i = i^a$, $i = 1, \ldots, r_{\mathrm{true}}$.

### 5.1 Implementation details

As mentioned before, the initial value $X_0$ is crucial for PLM. The ideal choice would be a matrix not far from the solution of (6). Note that (3) and (6) are both approximate models of the original problem (1), meaning that the solution of (3) is not very far from that of (6). Therefore, we use the solution of the convex problem (3), which can be solved efficiently by (L)ADMM [50], as the initialization $X_0$. We call the algorithm [50] nuclear-(L)ADMM. An important advantage of nuclear-(L)ADMM is that the yielded solutions have low-rank. Because the subproblem (25) or (36) is solved by partial SVD, a smaller $r_0$ results in a lower time cost. We will use this initialization in all tests.

The magnitudes of different datasets may differ greatly, which brings difficulties in the settings of parameters. This issue can be tackled by scaling. Specifically, we set the scaling value $c = \|X_0\|_2$. The scaled model corresponding to (11) is

$$\min_{\mathrm{rank}(X)\le r_k} \|X\|^p_{*,p} + \frac{\lambda}{2}\|\mathcal{A}(X) - \boldsymbol{b}/c\|_2^2, \tag{56}$$

and the scaled initialization is $X_0/c$. After obtaining the solution $\widetilde{X}$ of (56) by PLM, the solution of the original problem is $c\widetilde{X}$. Similar scaling can be used in nuclear-(L)ADMM: we set $c = \|\boldsymbol{b}\|_2$ and the scaled model corresponding to (3) is

$$\min_X \ \|X\|_* + \frac{\lambda}{2}\|\mathcal{A}(X) - \boldsymbol{b}/c\|_2^2. \tag{57}$$

In the remainder of this section, PLM is discussed based on the scaled model (56), and $\mathcal{P}(X)$ and $\{X_k\}$ are the scaled objective function and the scaled sequence, respectively.

In our implementation, the partial SVD is realized by the optPROPACK package[1] [34]. By our test, this package outperforms the influential PROPACK package [29].

[1] The code is available at https://zhouchenlin.github.io/.

As discussed in [27] and based our own test, $p = 0.5$ performs better than other choice of $p$ generally. For simplicity, we fix $p = 0.5$ for all tests. As mentioned before, $\tau$ can be any positive scalar. To approximate the original model well, we set $\tau$ as small as $10^{-5}$. As with Yang and Yuan [50], we set $\eta = \|\mathcal{A}^*\mathcal{A}\|_2$ for LADMM. The performance of PLM depends heavily on whether the (L)ADMM in the inner loops converges well. Hence, the stopping tolerance for the inner loops cannot be too loose. In our tests, as suggested by Remark 4.18, the stopping tolerance of the inner loops is set as

$$\frac{\|X_{t+1} - X_t\|_F}{\max(\|X_t\|_F, 1)} \leq 10^{-6};$$

and the stopping tolerance of the outer loops is set as

$$\frac{\|X_{k+1} - X_k\|_F}{\max(\|X_k\|_F, 1)} \leq 10^{-5}.$$

### 5.2 Convergence behaviour of PLM

We show the convergence behaviour of PLM on the matrix completion problem. First, we generate a low-rank matrix $Z \in \mathbb{R}^{500\times 500}$ with $\operatorname{rank}(Z) = 10$ and singular values in power-law distribution, where the power is set to be $-3$. Then, we select 30% entries of $Z$ randomly (i.e, the sample ratio is 30%) and use $\boldsymbol{b}'$ to denote the vector formed by these entries. At last, we impose some Gaussian noise on $\boldsymbol{b}'$ and obtain the observed vector $\boldsymbol{b}$ by

$$\boldsymbol{b} = \boldsymbol{b}' + \varepsilon \frac{\|\boldsymbol{b}'\|_2}{\|\boldsymbol{n}\|_2}\boldsymbol{n},$$

where the entries of $\boldsymbol{n}$ are drawn from a standard normal distribution and $\varepsilon$ is the noise level. We set $\varepsilon = 10^{-3}$ in this test. Besides using the solution of (3) as the initialization, we use two other initializations to show the convergence behaviour of PLM:

1. the initialization $X_0^{(1)}$ is the solution of (3) solved by nuclear-ADMM;
2. the initialization $X_0^{(2)}$ is a random generated matrix with $\operatorname{rank}(X_0^{(2)}) = 10$;
3. the initialization $X_0^{(3)}$ is the best rank-7 approximation of $X_0^{(1)}$, obtained by the truncated SVD of $X_0^{(1)}$.

We run PLM to 8 outer loops, ignoring the stopping tolerance of outer loops, with different initializations.

As shown in Fig. 1, the sequence $\{\mathcal{P}(X_k)\}$ is monotonically decreasing. For the initializations $X_0^{(1)}$ and $X_0^{(3)}$, $\operatorname{rank}(X_k)$ is unchanged for all $k$, and for the initialization $X_0^{(2)}$, $\operatorname{rank}(X_k)$ is unchanged when $k \geq 3$. PLM converges very fast: the relative change between $X_3$ and $X_4$ is less than $10^{-5}$ for all initializations. For the initialization $X_0^{(2)}$, the change between $\mathcal{P}(X_0)$ and $\mathcal{P}(X_1)$ and the relative change between $X_0$ and $X_1$ are both very large, implying that $X_0^{(2)}$ is far from a critical point.

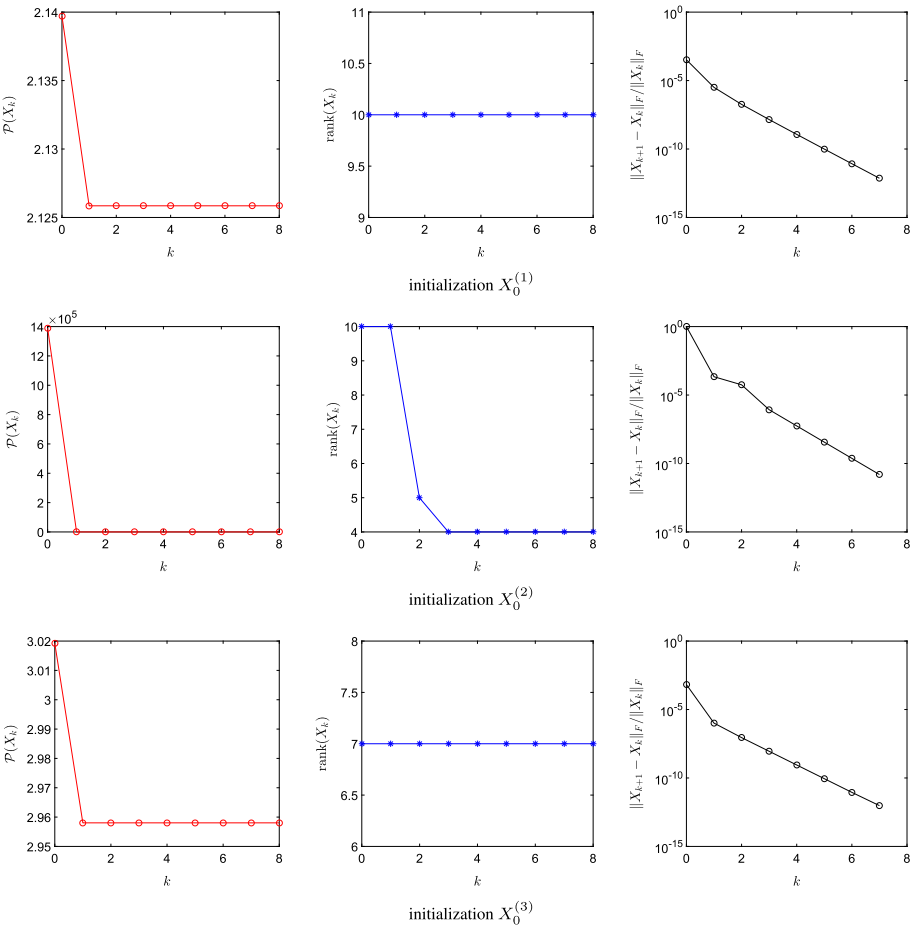


**Fig. 1** The values $\mathcal{P}(X_k)$, $\mathrm{rank}(X_k)$ and $\|X_{k+1} - X_k\|_F / \|X_k\|_F$ versus the outer iteration number in PLM, where all values are with respect to the scaled model (56). The first row is with the initialization $X_0^{(1)}$, the second row is with the initialization $X_0^{(2)}$, and the third row is with the initialization $X_0^{(3)}$

The iteration number of nuclear-ADMM for generating $X_0^{(1)}$ is 81, and we show the iteration numbers of inner loops of PLM in Table 1. We can find that these numbers are relatively small for $X_0^{(1)}$ and $X_0^{(3)}$, while for $X_0^{(2)}$ the iteration number of inner loops for the first step is as high as 5448. The iteration numbers are consistent with the results shown in Fig. 1.

### 5.3 Matrix completion

In this subsection, we compare PLM with nuclear-ADMM and t-IRucLq-M [27] on the matrix completion problem. We choose these two methods to compare due to their significant superiorities to many other existing methods. See the numerical results in

**Table 1** Iteration numbers of inner loops of each step, with different initializations

| Initialization | Iteration numbers of inner loops |
|---|---|
| $X_0^{(1)}$ | 35, 3, 1, 1, 1, 1, 1, 1 |
| $X_0^{(2)}$ | 5448, 15, 7, 1, 1, 1, 1, 1 |
| $X_0^{(3)}$ | 24, 2, 1, 1, 1, 1, 1, 1 |

For example, the iteration numbers corresponding to $X_0^{(1)}$ is 35, 3, . . ., meaning that when the initialization is $X_0^{(1)}$, the iteration numbers of ADMM for generating $X_1, X_2, \ldots$ are 35, 3, . . .

[13, 27, 50]. We test matrices with singular values uniformly distributed in [0.1, 1] and matrices with singular values in power-law distribution, where the power is set to be $-3$. We consider three levels of noise $\varepsilon = 10^{-3}, 10^{-4}, 10^{-5}$.

nuclear-ADMM is run based on the scaled model (57), whose parameters are relatively stable. We set $\lambda = 10^3/\varepsilon$ and the stopping tolerance to be $10^{-5}$. For PLM (56), we set $\lambda = 10^3/\varepsilon$ and $\beta_k = 10^5$ for all $k$. t-IRucLq-M solves (6) by (7). Its model parameters $p$ and $\lambda$ are set to be the same as those of PLM for the original model (6). The rank estimate of t-IRucLq-M is set to be $r_0$ and the stopping tolerance is set to be $10^{-3}$. The comparison results are shown in Tables 2 and 3, where the running time of PLM has included the time for generating the initialization. For matrices with singular values in power-law distribution, we only test $r_{\text{true}} = 10$, because $\sigma_{r_{\text{true}}}/\sigma_1$ is too small for greater $r_{\text{true}}$, resulting in difficulties in recovering the original matrices. All results are averaged over 10 trials for each case.

We consider the relative error. For matrices with uniformly distributed singular values, as shown in Table 2, when the noise level is as high as $10^{-3}$, the results of different methods are close; as the noise level decreases, the advantage of Schatten $p$-quasi-norm minimizations becomes obvious: both t-IRucLq-M and PLM outperform nuclear-ADMM and PLM has the best performance. For matrices with singular values in power-law distribution, as shown in Table 3, nuclear-ADMM performs worse than it does on the former type of matrices; both t-IRucLq-M and PLM outperform nuclear-ADMM and PLM has the best performance.

Now, we focus on the running time. We can see that the time cost of nuclear-ADMM is very cheap, which has also been demonstrated in previous works. PLM converges very fast: the numbers of outer loops of PLM are at most 2 for all cases. The gap between the running time of nuclear-ADMM and PLM is not very big, and PLM is much faster than t-IRucLq-M.

### 5.4 Sylvester equation

In this subsection, we consider solving a low-rank solution of a Sylvester equation

$$AX + XB \approx C,$$

**Table 2** Comparison results of different methods on matrix completion: matrices with uniformly distributed singular values

| Noise level | $(m, n, r_{\text{true}})$ | SR (%) | Nuclear-ADMM | | t-IRucLq-M | | PLM | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | Time | RErr | Time | RErr | Time | RErr | Iter |
| $10^{-3}$ | (500, 500, 10) | 20 | 5.0 | 4.95e−4 | 12.8 | 4.99e−4 | 5.3 | 4.94e−4 | 2 |
| | | 40 | 1.5 | 3.43e−4 | 8.7 | 3.45e−4 | 1.7 | 3.26e−4 | 2 |
| | | 60 | 0.8 | 2.63e−4 | 3.9 | 2.63e−4 | 1.0 | 2.62e−4 | 1 |
| | (800, 1200, 20) | 20 | 15.8 | 5.03e−4 | 33.1 | 5.13e−4 | 17.4 | 5.02e−4 | 2 |
| | | 40 | 6.6 | 3.33e−4 | 14.0 | 3.35e−4 | 7.7 | 3.32e−4 | 2 |
| | | 60 | 2.9 | 2.66e−4 | 5.8 | 2.71e−4 | 3.7 | 2.66e−4 | 1 |
| | (1500, 1500, 30) | 20 | 26.2 | 4.89e−4 | 53.7 | 4.92e−4 | 31.4 | 4.88e−4 | 2 |
| | | 40 | 13.9 | 3.27e−4 | 30.3 | 3.35e−4 | 17.7 | 3.26e−4 | 2 |
| | | 60 | 9.4 | 2.61e−4 | 23.8 | 2.67e−4 | 12.3 | 2.60e−4 | 1 |
| $10^{-4}$ | (500, 500, 10) | 20 | 4.9 | 5.70e−5 | 13.1 | 5.33e−5 | 5.7 | 4.93e−5 | 2 |
| | | 40 | 1.6 | 3.54e−5 | 8.0 | 3.45e−5 | 1.8 | 3.27e−5 | 1.6 |
| | | 60 | 0.9 | 2.72e−5 | 3.8 | 2.71e−5 | 1.1 | 2.61e−5 | 1 |
| | (800, 1200, 20) | 20 | 15.8 | 5.61e−5 | 32.2 | 5.48e−5 | 18.7 | 5.02e−5 | 2 |
| | | 40 | 5.9 | 3.56e−5 | 15.4 | 3.55e−5 | 7.2 | 3.34e−5 | 1.8 |
| | | 60 | 3.0 | 2.75e−5 | 6.1 | 2.76e−5 | 3.9 | 2.66e−5 | 1 |
| | (1500, 1500, 30) | 20 | 25.8 | 5.37e−5 | 61.1 | 5.30e−5 | 34.5 | 4.89e−5 | 2 |
| | | 40 | 13.2 | 3.45e−5 | 29.6 | 3.40e−5 | 17.5 | 3.25e−5 | 1.9 |
| | | 60 | 9.1 | 2.74e−5 | 22.9 | 2.71e−5 | 12.1 | 2.61e−5 | 1 |
| $10^{-5}$ | (500, 500, 10) | 20 | 5.4 | 3.01e−5 | 13.7 | 2.03e−5 | 6.0 | 6.94e−6 | 2 |
| | | 40 | 1.6 | 1.48e−5 | 8.3 | 1.11e−5 | 1.8 | 4.04e−6 | 1.9 |
| | | 60 | 0.9 | 8.39e−6 | 3.9 | 6.85e−6 | 1.1 | 2.90e−6 | 1 |
| | (800, 1200, 20) | 20 | 15.1 | 2.67e−5 | 35.5 | 1.12e−5 | 18.2 | 6.79e−6 | 2 |
| | | 40 | 6.2 | 1.24e−5 | 14.7 | 8.35e−6 | 7.6 | 3.63e−6 | 2 |
| | | 60 | 2.8 | 7.18e−6 | 5.9 | 5.71e−6 | 3.7 | 2.86e−6 | 1 |
| | (1500, 1500, 30) | 20 | 24.5 | 2.38e−5 | 57.7 | 1.83e−5 | 33.8 | 6.00e−6 | 2 |
| | | 40 | 14.0 | 1.25e−5 | 28.3 | 9.11e−6 | 18.5 | 3.63e−6 | 2 |
| | | 60 | 8.7 | 8.44e−6 | 21.6 | 5.85e−6 | 11.8 | 2.87e−6 | 1 |

Here "SR" is sample ratio, "iter" is the number of outer loops, and the running time is measured in seconds

where $A \in \mathbb{R}^{m\times m}$, $B \in \mathbb{R}^{n\times n}$, $C \in \mathbb{R}^{m\times n}$. This problem is mentioned in [44]. In this case,

$$\mathcal{A}(X) = \text{vec}(AX + XB) = (I_m \otimes A + B^\top \otimes I_n)\text{vec}(X), \tag{58}$$

where $\otimes$ is the Kronecker product, and $I_s \in \mathbb{R}^{s\times s}$ is an identity matrix, $s = m, n$. We define $\mathcal{B} : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$ by $\mathcal{B}(X) = AX + XB$. Equation (58) implies that the representing matrix of $\mathcal{B}$ is $(I_m \otimes A + B^\top \otimes I_n)$ and hence $\mathcal{B}^*(Y) = A^\top Y + YB^\top$.

**Table 3** Comparison results of different methods on matrix completion: matrices with singular values in power-law distribution

| Noise level | $(m, n, r_{\text{true}})$ | SR (%) | Nuclear-ADMM | | t-IRucLq-M | | PLM | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | Time | RErr | Time | RErr | Time | RErr | Iter |
| $10^{-3}$ | (500, 500, 10) | 20 | 4.1 | 7.80e−4 | 15.6 | 6.99e−4 | 4.7 | 5.18e−4 | 2 |
| | | 40 | 1.8 | 4.89e−4 | 9.4 | 4.63e−4 | 2.2 | 3.32e−4 | 2 |
| | | 60 | 1.0 | 3.75e−4 | 4.9 | 3.12e−4 | 1.3 | 2.64e−4 | 2 |
| | (800, 1200, 10) | 20 | 9.3 | 5.95e−4 | 25.1 | 5.88e−4 | 12.3 | 3.71e−4 | 2 |
| | | 40 | 4.5 | 3.90e−4 | 12.4 | 3.16e−4 | 5.8 | 2.41e−4 | 2 |
| | | 60 | 3.4 | 2.83e−4 | 5.8 | 2.62e−4 | 4.3 | 1.92e−4 | 2 |
| | (1500, 1500, 10) | 20 | 20.3 | 4.75e−4 | 42.4 | 4.37e−4 | 25.3 | 3.04e−4 | 2 |
| | | 40 | 12.4 | 3.01e−4 | 22.3 | 2.75e−4 | 16.1 | 1.97e−4 | 2 |
| | | 60 | 9.1 | 2.30e−4 | 18.9 | 2.01e−4 | 12.3 | 1.56e−4 | 2 |
| $10^{-4}$ | (500, 500, 10) | 20 | 4.6 | 4.64e−4 | 14.2 | 8.73e−5 | 5.6 | 5.33e−5 | 2 |
| | | 40 | 2.3 | 2.92e−4 | 8.6 | 6.46e−5 | 2.7 | 3.32e−5 | 2 |
| | | 60 | 1.1 | 2.15e−4 | 3.9 | 4.77e−5 | 1.4 | 2.64e−5 | 2 |
| | (800, 1200, 10) | 20 | 10.9 | 3.35e−4 | 26.3 | 7.58e−5 | 14.1 | 3.80e−5 | 2 |
| | | 40 | 5.3 | 2.54e−4 | 13.2 | 5.15e−5 | 6.9 | 2.39e−5 | 2 |
| | | 60 | 3.4 | 1.92e−4 | 6.2 | 3.81e−5 | 4.5 | 1.93e−5 | 2 |
| | (1500, 1500, 10) | 20 | 27.6 | 2.93e−4 | 45.1 | 6.32e−5 | 34.2 | 2.97e−5 | 2 |
| | | 40 | 12.7 | 2.28e−4 | 21.6 | 3.96e−5 | 16.8 | 1.93e−5 | 2 |
| | | 60 | 9.0 | 1.77e−4 | 17.9 | 2.88e−5 | 12.1 | 1.56e−5 | 2 |
| $10^{-5}$ | (500, 500, 10) | 20 | 4.6 | 4.62e−4 | 14.1 | 2.01e−5 | 5.8 | 6.55e−6 | 2 |
| | | 40 | 2.2 | 2.93e−4 | 8.4 | 1.35e−5 | 2.7 | 4.32e−6 | 2 |
| | | 60 | 1.2 | 2.13e−4 | 3.8 | 1.01e−5 | 1.5 | 2.83e−6 | 1 |
| | (800, 1200, 10) | 20 | 11.2 | 3.36e−4 | 25.7 | 1.65e−5 | 14.8 | 5.21e−6 | 2 |
| | | 40 | 5.5 | 2.54e−4 | 12.7 | 9.83e−6 | 7.2 | 3.30e−6 | 2 |
| | | 60 | 3.7 | 1.90e−4 | 6.6 | 6.77e−6 | 4.9 | 2.04e−6 | 2 |
| | (1500, 1500, 10) | 20 | 28.5 | 2.81e−4 | 62.6 | 1.43e−5 | 36.2 | 4.12e−6 | 2 |
| | | 40 | 13.0 | 2.31e−4 | 29.2 | 8.22e−6 | 17.3 | 2.61e−6 | 2 |
| | | 60 | 9.6 | 1.74e−4 | 18.6 | 5.72e−6 | 12.9 | 1.64e−6 | 2 |

Here "SR" is sample ratio, "iter" is the number of outer loops, and the running time is measured in seconds

Therefore,

$$\mathcal{A}^*\mathcal{A}(X) = A^\top AX + A^\top XB + AXB^\top + XBB^\top,$$

and (7) is a difficult linear matrix equation; see the discussion for Simoncini [44, Eq. (2)]. We compare PLM with nuclear-LADMM and the Matlab function "`sylvester`".

We test matrices with singular values uniformly distributed in [0.1, 1] and matrices with singular values in power-law distribution, where the power is set to be −2. We generate $A$ and $B$ by the Matlab commands "`randn`($m$)" and "`randn`($n$)", respectively.

**Table 4** Comparison results of different methods on solving Sylvester equations: matrices with uniformly distributed singular values

| $(m, n, r_{\text{true}})$ | Noise level | `sylvester` | | Nuclear-LADMM | | PLM | | |
|---|---|---|---|---|---|---|---|---|
| | | Time | RErr | Time | RErr | Time | RErr | Iter |
| (300, 300, 5) | $10^{-3}$ | 0.2 | 1.16e+0 | 1.4 | 2.30e−4 | 1.7 | 2.08e−4 | 2 |
| | $10^{-4}$ | 0.2 | 2.70e−1 | 1.5 | 1.00e−4 | 1.8 | 2.12e−5 | 2 |
| | $10^{-5}$ | 0.2 | 2.92e−2 | 1.4 | 1.06e−4 | 2.2 | 7.07e−6 | 2 |
| (500, 600, 10) | $10^{-3}$ | 0.7 | 1.91e+0 | 4.9 | 2.52e−4 | 6.6 | 2.18e−4 | 2 |
| | $10^{-4}$ | 0.7 | 3.70e−1 | 5.0 | 1.25e−4 | 6.7 | 2.23e−5 | 2 |
| | $10^{-5}$ | 0.7 | 2.17e−2 | 5.0 | 1.26e−4 | 9.6 | 6.88e−6 | 2 |
| (1000, 1000, 20) | $10^{-3}$ | 4.2 | 2.89e+0 | 25.4 | 2.71e−4 | 35.5 | 2.27e−4 | 2 |
| | $10^{-4}$ | 4.2 | 2.75e−1 | 25.9 | 1.58e−4 | 36.1 | 2.33e−5 | 2 |
| | $10^{-5}$ | 4.2 | 3.83e−2 | 25.2 | 1.57e−4 | 51.2 | 6.60e−6 | 2 |

Here "iter" is the number of outer loops, and the running time is measured in seconds

**Table 5** Comparison results of different methods on solving Sylvester equations: matrices with singular values in power-law distribution

| $(m, n, r_{\text{true}})$ | Noise level | `sylvester` | | Nuclear-LADMM | | PLM | | |
|---|---|---|---|---|---|---|---|---|
| | | Time | RErr | Time | RErr | Time | RErr | Iter |
| (300, 300, 5) | $10^{-3}$ | 0.2 | 1.90e+0 | 1.9 | 2.49e−4 | 2.3 | 2.14e−4 | 2 |
| | $10^{-4}$ | 0.2 | 1.65e−1 | 1.8 | 1.56e−4 | 2.2 | 2.16e−5 | 2 |
| | $10^{-5}$ | 0.2 | 1.16e−2 | 1.8 | 1.48e−4 | 2.7 | 6.92e−6 | 2 |
| (500, 600, 10) | $10^{-3}$ | 0.7 | 2.34e+0 | 7.0 | 6.40e−4 | 10.4 | 2.23e−4 | 2 |
| | $10^{-4}$ | 0.7 | 2.31e−1 | 7.1 | 5.62e−4 | 9.6 | 2.29e−5 | 2 |
| | $10^{-5}$ | 0.7 | 3.08e−2 | 7.1 | 5.24e−4 | 9.6 | 6.76e−6 | 2 |
| (1000, 1000, 10) | $10^{-3}$ | 4.2 | 2.60e+0 | 35.1 | 4.64e−4 | 48.8 | 1.58e−4 | 2 |
| | $10^{-4}$ | 4.2 | 4.63e−1 | 33.9 | 4.71e−4 | 46.1 | 1.69e−5 | 2 |
| | $10^{-5}$ | 4.2 | 2.63e−2 | 34.5 | 3.80e−4 | 64.3 | 5.51e−6 | 2 |

Here "iter" is the number of outer loops, and the running time is measured in seconds

Then we generate a low-rank matrix $X$ with singular values in specified distributions. After getting $C$ by $C = AX + XB$, we impose some Gaussian noise on $C$. We consider three levels of noise $\varepsilon = 10^{-3}, 10^{-4}, 10^{-5}$.

For nuclear-LADMM (57), we set $\lambda = 10^3/\varepsilon$ and the stopping tolerance to be $10^{-6}$. For PLM (56), we set $\lambda = 10^3/\varepsilon$ and $\beta_k = 10^6$ for all $k$. The comparison results are shown in Tables 4 and 5, where the running time of PLM has included the time for generating the initialization. All results are averaged over 10 trials for each case.

We consider the relative error. `sylvester` performs badly. For all cases with noise level $10^{-3}$, its RErr is greater than 1, meaning the failure in recovering $X$. nuclear-LADMM performs well, but its performance is affected by the distribution of singular values. Like the results for matrix completion, nuclear-LADMM performs worse on

matrices with singular values in power-law distribution than it does on matrices with uniformly distributed singular values. PLM outperforms nuclear-LADMM in all cases and its performance is independent of the distribution of singular values. The advantage of PLM over nuclear-LADMM is very obvious in the cases where the noise level is lower or singular values are with power-law distributions.

Now we look at the running time. `sylvester` is a direct method and consumes the least running time. The running time of nuclear-LADMM depends on the size of the matrix and the distribution of singular values. PLM converges fast: only two outer loops are needed in all cases. PLM consumes much longer time than nuclear-LADMM, but less than twice the consumed time of nuclear-LADMM in almost all cases.

### 5.5 Image deblurring

In this subsection, we consider a grayscale image deblurring problem. The original image is the Greek flag shown in Fig. 2, whose size is $256 \times 384$ and numerical rank[2] is 4. The original image is corrupted by a $9 \times 9$ Gaussian blurring kernel with standard deviation 5 (`fspecial ('gaussian',[9,9],5)`) and Gaussian noise. We consider three levels of noise $\varepsilon = 10^{-3}, 10^{-4}, 10^{-5}$.

Suppose the original image is $X$, the blurring kernel is $\mathcal{A}$ (a convolution operator), and the noise is $N$. Then the observed image is

$$B = \mathcal{A}(X) + N.$$

Image deblurring is estimating $X$ from the observed image $B$, where $\mathcal{A}$ is known and $N$ is unknown. This problem is ill-conditioned. Deblurring by solving the linear system $\mathcal{A}(X) = B$ directly cannot obtain a good image restoration. To overcome this difficulty, regularization methods are required to stabilize the solution. Besides (3) and (6), we also consider the Tikhonov regularization method:

$$\min_{X \in \mathbb{R}^{m \times n}} \|X\|_F^2 + \frac{\lambda}{2} \|\mathcal{A}(X) - B\|_F^2.$$

We compare PLM with the Tikhonov method and nuclear-ADMM, where the feature of $\mathcal{A}$ will be considered (see Sect. 3.1.2). For nuclear-ADMM (57), we set $\lambda = 10^3/\varepsilon$; for PLM (56), we set $\lambda = 10^4/\varepsilon$.

The comparison results are shown in Table 6, where the running time of PLM has included the time for generating the initialization. All results are averaged over 10 trials for each case. nuclear-ADMM and PLM obtain the same numerical rank as that of the original image, while the Tikhonov method yields a much higher numerical rank. This illustrates the advantage of nuclear-ADMM and PLM in recovering low-rank matrix. In terms of relative error, the Tikhonov method is the worst, nuclear-ADMM yields an acceptable result, and PLM outperforms nuclear-ADMM. In terms of the

[2] Given a matrix $X$, we set the numerical rank as the number of singular values $\sigma_r(X)$ satisfying $\sigma_r(X)/\|X\|_F \geq 10^{-4}$.

**Table 6** Comparison results of different methods on image deblurring

| Noise level | Tikhonov method | | | Nuclear-ADMM | | | PLM | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Rank | RErr | Time | Rank | RErr | Time | Rank | RErr | Time | Iter |
| $10^{-3}$ | 254 | 5.24e−2 | 7.2e−3 | 4 | 1.17e−2 | 1.7 | 4 | 1.10e−2 | 9.0 | 2 |
| $10^{-4}$ | 230 | 2.28e−2 | 6.9e−3 | 4 | 3.00e−3 | 0.6 | 4 | 2.65e−3 | 2.5 | 2 |
| $10^{-5}$ | 121 | 7.13e−3 | 5.6e−3 | 4 | 7.40e−4 | 0.5 | 4 | 4.87e−4 | 0.8 | 2 |

Here "rank" is the numerical rank, "iter" is the number of outer loops, and the running time is measured in seconds

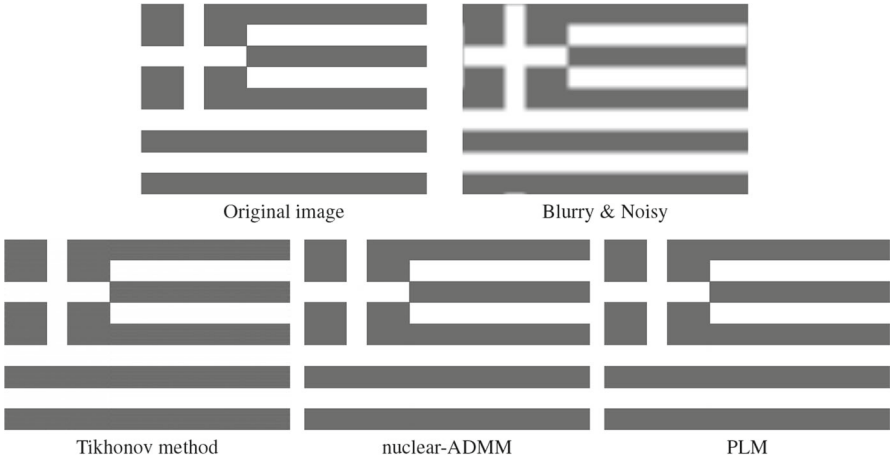


**Fig. 2** Restorations of different methods, where the noise level is $10^{-5}$

running time, the Tikhonov method is very fast; the running time of nuclear-ADMM depends on the noise level; PLM converges very fast and consumes much longer time than nuclear-ADMM; the gap between PLM time and nuclear-ADMM time becomes smaller as the noise level decreases. The visual comparison of some restorations is shown in Fig. 2.

## 6 Conclusion

We have proved the lower bound theory for Schatten $p$-quasi-norm minimization: the positive singular values of local minimizers are bounded from below by a constant. This property explains the advantage of the Schatten $p$-quasi-norm minimization in recovering low-rank matrices. Using this property, we have proposed a proximal linearization method to solve the Schatten $p$-quasi-norm minimization problem. After studying singular value functions, we proved the global convergence of the proposed algorithm by using the KL property of the objective function. In the experiments, we considered matrix completion, Sylvester equation and image deblurring. The pro-

posed method outperforms existing methods in terms of approximation error, and has acceptable running time.

Experiments show that PLM converges very fast. The convergence rate can be analysed based on the use of the KL exponent. See [33, 51] and references therein. The convergence rate of PLM will be discussed in the future.

In many applications, the entries of $X$ lie in a certain range: $l \le x_{ij} \le u$, where $-\infty \le l \le u \le +\infty$. For example, in imaging applications, we usually have non-negativity constraints, i.e., $l = 0, u = +\infty$. To fit this situation, we can consider the following model:

$$\min_{X \in \mathbb{R}^{m \times n}} \|X\|_{*,p}^p + \frac{\lambda}{2}\|\mathcal{A}(X) - \boldsymbol{b}\|_2^2 + \delta_{\mathscr{D}}(X), \tag{59}$$

where $\mathscr{D}$ is a convex closed set. Models with similar constraints corresponding to (5) have been studied in many papers, e.g., [4, 12, 14, 53]. Theories and algorithms related to (59) will be considered in the future.

**Acknowledgements** The author is extremely grateful to the editor and the two anonymous referees for their valuable feedback, which improved this paper significantly. The author is also grateful to Dr. Xianshun Nian and Dr. Guomin Liu for helpful discussions. This work was partially supported by the National Natural Science Foundation of China (12201319).

## 7 Appendix

This appendix summarizes some important results on KL theory and gives some examples. The following definition is adopted from Attouch et al. [2, Definition 4.1].

**Definition 7.1** (o-minimal structure on $\mathbb{R}$) Let $\mathscr{O} = \{\mathscr{O}_n\}_{n \in \mathbb{N}}$ such that each $\mathscr{O}_n$ is a collection of subsets of $\mathbb{R}^n$. The family $\mathscr{O}$ is an o-minimal structure on $\mathbb{R}$, if it satisfies the following axioms:

(i) Each $\mathscr{O}_n$ is a boolean algebra. Namely $\emptyset \in \mathscr{O}_n$ and for each $\mathscr{A}$, $\mathscr{B}$ in $\mathscr{O}_n$, $\mathscr{A} \cup \mathscr{B}$, $\mathscr{A} \cap \mathscr{B}$, and $\mathbb{R}^n \setminus \mathscr{A}$ belong to $\mathscr{O}_n$.
(ii) For all $\mathscr{A}$ in $\mathscr{O}_n$, $\mathscr{A} \times \mathbb{R}$ and $\mathbb{R} \times \mathscr{A}$ belong to $\mathscr{O}_{n+1}$.
(iii) For all $\mathscr{A}$ in $\mathscr{O}_{n+1}$, $\{(x_1, \ldots, x_n) \in \mathbb{R}^n : (x_1, \ldots, x_n, x_{n+1}) \in \mathscr{A}\}$ belongs to $\mathscr{O}_n$.
(iv) For all $i \ne j$ in $\{1, 2, \ldots, n\}$, $\{(x_1, \ldots, x_n) \in \mathbb{R}^n : x_i = x_j\}$ belongs to $\mathscr{O}_n$.
(v) The set $\{(x_1, x_2) \in \mathbb{R}^2 : x_1 < x_2\}$ belongs to $\mathscr{O}_2$.
(vi) The elements of $\mathscr{O}_1$ are exactly finite unions of intervals.

Let $\mathscr{O}$ be an o-minimal structure on $\mathbb{R}$. We say that a set $\mathscr{A} \subseteq \mathbb{R}^n$ is *definable* (on $\mathscr{O}$) if $\mathscr{A} \in \mathscr{O}_n$. A function $f : \mathbb{R}^n \to (-\infty, +\infty]$ is definable if its graph $\{(\boldsymbol{x}, y) \in \mathbb{R}^n \times (-\infty, +\infty] : y \in f(\boldsymbol{x})\}$ is definable on $\mathscr{O}$. We list some known elementary properties of definable functions below.

**Property 7.2** (See [2]) *Finite sums of definable functions are definable; indicator functions of definable sets are definable; compositions of definable functions or mappings are definable.*

It is known that any proper lower semicontinuous function that is definable is a KL function; see [2, Theorem 4.1].

***Example 7.3*** A class of o-minimal structure is the log-exp structure [45, Example 2.5]. By this structure, the following functions are all definable:

1. semi-algebraic functions; see Definition 7.4 below.
2. the function $\mathbb{R} \to \mathbb{R}$ given by

$$x \mapsto \begin{cases} x^r, & x > 0 \\ 0, & x \leq 0, \end{cases}$$

   where $r \in \mathbb{R}$.
3. the exponential function: $\mathbb{R} \to \mathbb{R}$ given by $x \mapsto e^x$ and the logarithm function: $(0, \infty) \to \mathbb{R}$ given by $x \mapsto \log(x)$.

**Definition 7.4** (See [5, Definition 5]) A subset $\mathscr{S}$ of $\mathbb{R}^d$ is a real semi-algebraic set if there exists a finite number of real polynomial functions $f_{ij}, g_{ij} : \mathbb{R}^d \to \mathbb{R}$ such that

$$\mathscr{S} = \bigcup_{j=1}^{s} \bigcap_{i=1}^{t} \left\{ \boldsymbol{x} \in \mathbb{R}^d : f_{ij}(\boldsymbol{x}) = 0 \text{ and } g_{ij}(\boldsymbol{x}) < 0 \right\}$$

A function $f : \mathbb{R}^d \to (-\infty, +\infty]$ is called semi-algebraic if its graph

$$\left\{ (\boldsymbol{x}, y) \in \mathbb{R}^{d+1} : f(\boldsymbol{x}) = y \right\}$$

is a semi-algebraic subset of $\mathbb{R}^{d+1}$.

The class of semi-algebraic sets is stable under the following operations: finite unions, finite intersections, complementation and Cartesian products.

***Example 7.5*** [5, Example 2] There is broad class of semi-algebraic sets and functions arising in optimization.

1. Real polynomial functions.
2. Indicator functions of semi-algebraic sets.
3. Finite sums and product of semi-algebraic functions.
4. Composition of semi-algebraic functions.
5. In matrix theory, all the following are semi-algebraic sets: cone of positive semidefinite matrices, Stiefel manifolds and constant rank matrices.

***Example 7.6*** Define $f(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R}$ by $f(\boldsymbol{x}) = \sum_{i=1}^{n} |x_i|^p$. We prove that $f$ is definable. First, consider the function $g(t) = |t|$. The graph of $g(t)$ is

$$\{(t, y) \in \mathbb{R}^2 : y = t, t > 0\} \cup \{(t, y) \in \mathbb{R}^2 : y = -t, t > 0\}$$
$$\cup \{(t, y) \in \mathbb{R}^2 : y = t, t = 0\}.$$

Hence, $g(t)$ is a semi-algebraic function. From Property 7.2 and Example 7.3, we know that $f(\boldsymbol{x})$ is definable.

***Example 7.7*** We prove that the function $f(\boldsymbol{x})$ defined in (47) is a semi-algebraic function. From Example 7.5, we see that it suffices to prove that the function $f_i(\boldsymbol{x}) := b_i \underline{x}_i, i = 1, \ldots, r$ is semi-algebraic. We only prove that $f_1(\boldsymbol{x}) = b_1 \underline{x}_1$ is semi-algebraic, and the other cases are similar. Define

$$\mathscr{T}_j := \{\boldsymbol{x} \in \mathbb{R}^h : x_j = \underline{x}_1\}, \quad j = 1, \ldots, h.$$

By the definition of $\underline{\boldsymbol{x}}$, we have

$$\mathscr{T}_j = \bigcap_{k=1}^{h} \{\boldsymbol{x} \in \mathbb{R}^h : |x_j| \geq |x_k|\}$$

and $\{\boldsymbol{x} \in \mathbb{R}^h : |x_j| \geq |x_k|\}$ can be written as a union of some semi-algebraic sets. Hence, $\mathscr{T}_j$ is semi-algebraic. The graph of $f_1(\boldsymbol{x})$ is

$$\bigcup_{j=1}^{h} \left( \left\{ (\boldsymbol{x}, y) \in \mathbb{R}^{h+1} : y = b_1 |x_j| \right\} \cap (\mathscr{T}_j \times \mathbb{R}) \right),$$

which is a semi-algebraic set.

## References


1. Attouch, H., Bolte, J.: On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. Math. Program. **116**(1), 5–16 (2009)
2. Attouch, H., Bolte, J., Redont, P., Soubeyran, A.: Proximal alternating minimization and projection methods for nonconvex problems: an approach based on the Kurdyka–Łojasiewicz inequality. Math. Oper. Res. **35**(2), 438–457 (2010)
3. Attouch, H., Bolte, J., Svaiter, B.F.: Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward-backward splitting, and regularized Gauss–Seidel methods. Math. Program. **137**(1–2), 91–129 (2013)
4. Beck, A., Teboulle, M.: Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. IEEE Trans. Image Process. **18**(11), 2419–2434 (2009)
5. Bolte, J., Sabach, S., Teboulle, M.: Proximal alternating linearized minimization for nonconvex and nonsmooth problems. Math. Program. **146**(1–2), 459–494 (2014)
6. Boţ, R.I., Nguyen, D.-K.: The proximal alternating direction method of multipliers in the nonconvex setting: convergence analysis and rates. Math. Oper. Res. **45**(2), 682–712 (2020)
7. Cai, J.-F., Candès, E.J., Shen, Z.: A singular value thresholding algorithm for matrix completion. SIAM J. Optim. **20**(4), 1956–1982 (2010)
8. Candès, E.J., Recht, B.: Exact matrix completion via convex optimization. Found. Comput. Math. **9**(6), 717 (2009)
9. Candès, E.J., Romberg, J., Tao, T.: Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Trans. Inf. Theory **52**(2), 489–509 (2006)
10. Candès, E.J., Tao, T.: The power of convex relaxation: near-optimal matrix completion. IEEE Trans. Inf. Theory **56**(5), 2053–2080 (2010)
11. Candès, E.J., Wakin, M.B., Boyd, S.P.: Enhancing sparsity by reweighted $\ell_1$ minimization. J. Four. Anal. Appl. **14**(5), 877–905 (2008)

12. Chan, R.H., Tao, M., Yuan, X.: Constrained total variation deblurring models and fast algorithms based on alternating direction method of multipliers. SIAM J. Imag. Sci. **6**(1), 680–697 (2013)
13. Chen, C., He, B., Yuan, X.: Matrix completion via an alternating direction method. IMA J. Numer. Anal. **32**(1), 227–245 (2012)
14. Chen, X., Ng, M.K., Zhang, C.: Non-Lipschitz-regularization and box constrained model for image restoration. IEEE Trans. Image Process. **21**(12), 4709–4721 (2012)
15. Chen, X., Xu, F., Ye, Y.: Lower bound theory of nonzero entries in solutions of $\ell_2$-$\ell_p$ minimization. SIAM J. Sci. Comput. **32**(5), 2832–2852 (2011)
16. Donoho, D.L.: For most large underdetermined systems of linear equations the minimal $\ell_1$-norm solution is also the sparsest solution. Commun. Pure Appl. Math. **59**(6), 797–829 (2006)
17. Eckart, C., Young, G.: The approximation of one matrix by another of lower rank. Psychometrika **1**(3), 211–218 (1936)
18. El Ghaoui, L., Gahinet, P.: Rank minimization under LMI constraints: a framework for output feedback problems. In: European Control Conf., pp. 1176–1179 (1993)
19. Fazel, M., Hindi, H., Boyd, S.P.: A rank minimization heuristic with application to minimum order system approximation. In: Proceedings of the 2001 American Control Conference (Cat. No. 01CH37148), vol. 6, pp. 4734–4739. IEEE (2001)
20. Fornasier, M., Rauhut, H., Ward, R.: Low-rank matrix recovery via iteratively reweighted least squares minimization. SIAM J. Optim. **21**(4), 1614–1640 (2011)
21. Gazzola, S., Meng, C., Nagy, J.G.: Krylov methods for low-rank regularization. SIAM J. Matrix Anal. Appl. **41**(4), 1477–1504 (2020)
22. Gu, S., Xie, Q., Meng, D., Zuo, W., Feng, X., Zhang, L.: Weighted nuclear norm minimization and its applications to low level vision. Int. J. Comput. Vis. **121**(2), 183–208 (2017)
23. Gu, S., Zhang, L., Zuo, W., Feng, X.: Weighted nuclear norm minimization with application to image denoising. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2862–2869 (2014)
24. Horn, R.A., Johnson, C.R.: Matrix Analysis. Cambridge University Press, Cambridge (2012)
25. Hosseini, S., Luke, D.R., Uschmajew, A.: Tangent and normal cones for low-rank matrices. In: Nonsmooth Optimization and its Applications, pp. 45–53 (2019)
26. Lai, M.-J., Liu, Y., Li, S., Wang, H.: On the Schatten $p$-quasi-norm minimization for low-rank matrix recovery. Appl. Comput. Harmon. Anal. **51**, 157–170 (2021)
27. Lai, M.-J., Xu, Y., Yin, W.: Improved iteratively reweighted least squares for unconstrained smoothed $\ell_q$ minimization. SIAM J. Numer. Anal. **51**(2), 927–957 (2013)
28. Lai, M.-J., Yin, W.: Augmented $\ell_1$ and nuclear-norm models with a globally linearly convergent algorithm. SIAM J. Imag. Sci. **6**(2), 1059–1091 (2013)
29. Larsen, R.M.: PROPACK-software for large and sparse SVD calculations. http://sun.stanford.edu/~rmunk/PROPACK/
30. Lee, K., Elman, H.C.: A preconditioned low-rank projection method with a rank-reduction scheme for stochastic partial differential equations. SIAM J. Sci. Comput. **39**(5), S828–S850 (2017)
31. Lewis, A.S., Sendov, H.S.: Nonsmooth analysis of singular values. Part I: theory. Set-Valued Anal. **13**(3), 213–241 (2005)
32. Li, G., Pong, T.K.: Global convergence of splitting methods for nonconvex composite optimization. SIAM J. Optim. **25**(4), 2434–2460 (2015)
33. Li, G., Pong, T.K.: Calculus of the exponent of Kurdyka–Łojasiewicz inequality and its applications to linear convergence of first-order methods. Found. Comput. Math. **18**(5), 1199–1232 (2018)
34. Lin, Z.: Some software packages for partial SVD computation. arXiv preprint arXiv:1108.1548 (2011)
35. Liu, Z., Wu, C., Zhao, Y.: A new globally convergent algorithm for non-Lipschitz $\ell_p - \ell_q$ minimization. Adv. Comput. Math. **45**(3), 1369–1399 (2019)
36. Lu, C., Lin, Z., Yan, S.: Smoothed low rank and sparse matrix recovery by iteratively reweighted least squares minimization. IEEE Trans. Image Process. **24**(2), 646–654 (2014)
37. Markovsky, I.: Structured low-rank approximation and its applications. Automatica **44**(4), 891–909 (2008)
38. Mohan, K., Fazel, M.: Iterative reweighted algorithms for matrix rank minimization. J. Mach. Learn. Res. **13**(1), 3441–3473 (2012)
39. Nikolova, M.: Analysis of the recovery of edges in images and signals by minimizing nonconvex regularized least-squares. Multiscale Model. Simul. **4**(3), 960–991 (2005)

40. Pong, T.K., Tseng, P., Ji, S., Ye, J.: Trace norm regularization: reformulations, algorithms, and multi-task learning. SIAM J. Optim. **20**(6), 3465–3489 (2010)
41. Recht, B., Fazel, M., Parrilo, P.A.: Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. SIAM Rev. **52**(3), 471–501 (2010)
42. Rockafellar, R.T., Wets, R.J.-B.: Variational Analysis, vol. 317. Springer (2009)
43. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D **60**(1–4), 259–268 (1992)
44. Simoncini, V.: Computational methods for linear matrix equations. SIAM Rev. **58**(3), 377–441 (2016)
45. Van den Dries, L., Miller, C., et al.: Geometric categories and o-minimal structures. Duke Math. J. **84**(2), 497–540 (1996)
46. Vandereycken, B.: Low-rank matrix completion by Riemannian optimization. SIAM J. Optim. **23**(2), 1214–1236 (2013)
47. Wang, Y., Yang, J., Yin, W., Zhang, Y.: A new alternating minimization algorithm for total variation image reconstruction. SIAM J. Imag. Sci. **1**(3), 248–272 (2008)
48. Wang, Y., Yin, W., Zeng, J.: Global convergence of ADMM in nonconvex nonsmooth optimization. J. Sci. Comput. **78**(1), 29–63 (2019)
49. 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**(4), 333–361 (2012)
50. Yang, J., Yuan, X.: Linearized augmented Lagrangian and alternating direction methods for nuclear norm minimization. Math. Comput. **82**(281), 301–329 (2013)
51. Yu, P., Li, G., Pong, T.K.: Kurdyka–Łojasiewicz exponent via INF-projection. Found. Comput. Math. **22**, 1–47 (2021)
52. Zeng, C., Wu, C.: On the edge recovery property of noncovex nonsmooth regularization in image restoration. SIAM J. Numer. Anal. **56**(2), 1168–1182 (2018)
53. Zeng, C., Wu, C.: On the discontinuity of images recovered by noncovex nonsmooth regularized isotropic models with box constraints. Adv. Comput. Math. **45**(2), 589–610 (2019)
54. Zeng, C., Wu, C., Jia, R.: Non-Lipschitz models for image restoration with impulse noise removal. SIAM J. Imag. Sci. **12**(1), 420–458 (2019)
55. Zhang, X., Bai, M., Ng, M.K.: Nonconvex-TV based image restoration with impulse noise removal. SIAM J. Imag. Sci. **10**(3), 1627–1667 (2017)
56. Zheng, Z., Ng, M., Wu, C.: A globally convergent algorithm for a class of gradient compounded non-Lipschitz models applied to non-additive noise removal. Inverse Prob. **36**(12), 125017 (2020)