Fixed point continuation algorithm with extrapolation for Schatten p-quasi-norm regularized matrix optimization problems

Xian Zhang, Dingtao Peng

Abstract. In this paper, we consider a general low-rank matrix optimization problem which is modeled by a general Schatten p-quasi-norm ($0 < p < 1$) regularized matrix optimization. For this nonconvex nonsmooth and non-Lipschitz matrix optimization problem, based on the matrix p-thresholding operator, we first propose a fixed point continuation algorithm with extrapolation (FPCAe) for solving it. Secondly, we prove that any accumulation point of the iterative sequence generated by the proposed algorithm is not only a critical point but also a global stationary point of the problem, where the global stationary point possesses some global optimality which can exclude too many stationary points even some local minimizers of the nonconvex problem. We also prove the rank invariance of the iterative sequence. Thirdly, we prove the global convergence and R-linear convergence rate of the whole iterative sequence generated by the proposed algorithm under some mild conditions. Finally, we conduct a large number of numerical experiments on random square and rectangular matrix completion problem, grayscale image and three-channel image recovery problem. The numerical results illustrate that the proposed FPCAe algorithm is competitive with some state-of-the-art algorithms for low-rank matrix recovery in terms of speed, accuracy, robustness and anti-noise.

Keywords. Low-rank matrix recovery problem, Schatten p-quasi-norm, fixed point continuation algorithm with extrapolation, global whole sequence convergence, R-linear convergence rate

AMS subject classification. 90C26, 90C30, 65K06

1 Introduction

In the last twenty years, the low-rank matrix recovery problem has received great attention, which can be described as the recovery of unknown matrix from limited information. The low-rank matrix recovery problem arises in a wide range of fields, such as computer vision and pattern recognition [3, 21], compressed sensing [6, 8], control [14, 49], machine learning [44], and system identification [30, 35, 50]. The general low-rank matrix recovery problem can be modeled as finding a matrix of minimum rank that satisfies a given linear system, namely,

$$\min_{X\in\mathbb{R}^{m\times n}}\{\operatorname{rank}(X)\ \text{ s.t. } \mathcal{A}(X)=b\},\ \text{ or }\ \min_{X\in\mathbb{R}^{m\times n}}\ \|\mathcal{A}(X)-b\|^2+\lambda\cdot\operatorname{rank}(X), \tag{1.1}$$

Dingtao Peng ✉
School of Mathematics and Statistics, Guizhou University, Guiyang 550025, China. E-mail: dingtaopeng@126.com
Xian Zhang
School of Mathematics and Statistics, Guizhou University, Guiyang 550025, China. E-mail: zhangxian05@163.com

where $\mathcal{A}:\mathbb{R}^{m\times n}\to\mathbb{R}^d$ with $d\ll mn$ is a linear map, and $b\in\mathbb{R}^d$ is a vector. A typical example of the low-rank matrix recovery problem is the matrix completion problem

$$\min_{X\in\mathbb{R}^{m\times n}}\{\text{rank}(X)\ \text{ s.t. }\ X_{i,j}=M_{i,j},\ (i,j)\in\Omega\},\tag{1.2}$$

where Ω is a subset of index set of X. Problem (1.2) is non-convex, non-smooth, non-Lipschitz, even discontinuous. Some researchers [15, 26, 27, 50, 52] have pointed out that (1.1) is an NP-hard problem from different perspectives. Fazel [15] first proposed a convex relaxation model to solve problem (1.1), and proved that the nuclear norm is the tightest convex envelope of rank function. The convex relaxation model can be expressed as

$$\min_{X\in\mathbb{R}^{m\times n}}\{\|X\|_*\ \text{ s.t. }\mathcal{A}(X)=b\},\quad\text{or}\quad\min_{X\in\mathbb{R}^{m\times n}}\lambda\|X\|_*+\|\mathcal{A}(X)-b\|^2,\tag{1.3}$$

where $\|X\|_*:=\sum_{i=1}^{min\{m,n\}}\sigma_i(X)$, $\sigma_i(X)$ is the i-th singular value of X. Many scholars [7, 9, 48] have studied the feasibility of nuclear norm relaxation for this problem, including the exact recovery conditions and effective algorithms. The study of exact recovery conditions include null space property [39, 46, 47], restricted isometry property [40, 45], etc. Effective algorithms for solving problem (1.3) include semi-definite programming SDPT3 method [47], singular value thresholding algorithm [5], interior-point methods [30], fixed point and Bregman iterative methods [34], iteratively reweighted algorithm [16], proximal point algorithm [29], alternating direction methods [10], linearized augmented Lagrangian [56], alternating direction methods [56] and so on. In order to accelerate the convergence of the iterations, some accelerate technique have been adopted for such convex optimization problems, dating back to Polyak's heavy ball method [43], Nesterov's extrapolation techniques [36, 37, 51]. A representative algorithm that combines these techniques is the accelerated proximal gradient with linesearch-like acceleration strategy (APGL) algorithm [51]. We know that the function values generated by APGL converges at a rate of $\mathcal{O}(\frac{1}{k^2})$, which is faster than the $\mathcal{O}(\frac{1}{k})$ convergence rate of the proximal gradient algorithm.

However, some strict conditions are required to successfully and accurately recover low-rank matrix through nuclear norm. In addition, the nuclear norm convex relaxation model may generate the matrix with a rank much higher than the real rank, and cannot recover the low-rank targets from the minimum measurements. After all, the rank function may not be approximated very well by the nuclear norm since the former is nonconvex but the latter is convex. Hence, many researchers began using some non-convex relaxations of the rank function [23, 28, 31, 32, 38, 41, 50, 60, 62], among which the matrix Schatten p-quasi-norm with $0<p<1$ has been studied by many researchers [22, 23, 28, 31, 32, 38, 41, 61]. The results show that the matrix Schatten p-quasi-norm provides better results from theory and practice than the standard nuclear norm relaxation. The matrix Schatten p-quasi-norm relaxation model for problem (1.1) is given as

$$\min_{X\in\mathbb{R}^{m\times n}}\{\|X\|_p^p\ \text{ s.t. }\ \mathcal{A}(X)=b\},\ \text{or}\quad\min_{X\in\mathbb{R}^{m\times n}}\lambda\|X\|_p^p+\|\mathcal{A}(X)-b\|^2,\tag{1.4}$$

where $\|X\|_p:=\Big(\sum_{i=1}^{\min\{m,n\}}\sigma_i^p(X)\Big)^{1/p}$ is the Schatten p-quasi-norm of matrix X.

In this paper, we consider a more general Schatten p-quasi-norm regularized matrix optimization problem:

$$\min_{X\in\mathbb{R}^{m\times n}}\ f_\lambda(X):=\theta(X)+\lambda\|X\|_p^p,\tag{1.5}$$

where $\lambda > 0$ is a regularization parameter, $\theta : \mathbb{R}^{m\times n} \to \mathbb{R}$ is bounded from below and is a smooth convex function with L-Lipschitz continuous gradient in $\mathbb{R}^{m\times n}$, i.e.,

$$\|\nabla\theta(X) - \nabla\theta(Y)\| \leq L\|X - Y\|_F.$$

Many numerical results [28,31,32,41,61] have shown that the solutions of nonconvex problem (1.4) are of lower rank than the solutions of convex problem (1.3), and the same low-rank solution can be obtained at a lower sampling rate by the former. Therefore, problem (1.5) has been studied extensively in recent years. Let's briefly review some of them. Reference [11] showed that even if problem (1.5) degenerates to the vector case, it is still strongly NP-hard. The optimality conditions for problem (1.5) have been analyzed in [32,41,61], and several different lower bound estimations for nonzero singular values of the solution matrices of probelm (1.5) have been obtained. The exact recovery condition has been analyzed in [22,23] and the estimation of error bounds for the recovery matrices has been established. In addition, many algorithms for solving problem (1.5) or its variants have been proposed, including truncated iteratively reweighted unconstrained ℓ_p minimization algorithm [16,23, 32], majorization minimization method [33], smoothing majorization method [31], fixed point continuation algorithms [41,42], and singular value p-shrinkage thresholding algorithm [28], inexact accelerated proximal gradient algorithm [50], proximal linearization method [61], etc. However, almost all of these works [16,23,28,31,32,41,42,50,55,60] only obtained the subsequential convergence of the algorithms to some stationary/critical points. Although the whole sequence convergence of the algorithms were discussed in [58,61], the convergence rate is not investigated.

The main purpose of this paper is to exploit the extrapolation technique to accelerate the fixed point continuation algorithm for problem (1.5), and more importantly, to provide a deep analysis for the global whole sequence convergence and fast convergence rate of the proposed algorithm under some mild conditions.

The main contributions of this paper are as follows.

(i) For the nonconvex nonsmooth and non-Lipschitz matrix optimization problem (1.5), based on the matrix p-thresholding operator, we first propose a fixed point continuation algorithm with extrapolation (FPCAe) for solving it.

(iii) Prove that any accumulation point of the iterative sequence generated by the proposed algorithm is not only a critical point but also a global stationary point of the problem, where the global stationary point possesses some global optimality which can exclude too many stationary points even some local minimizers of the nonconvex problem. The rank invariance of the iterative sequence is also proved.

(iii) Analyze the global convergence and R-linear convergence rate of the whole iterative sequence generated by the proposed algorithm under the assumption of Kurdyka-Łojasiewicz (KŁ) property.

(iv) Conduct a large number of numerical experiments on random square and rectangular matrix completion problem, grayscale image and three-channel image recovery problem to demonstrate the efficiency of the proposed FPCAe algorithm.

The structure of the remaining part of this paper is as follows. Section 2 mainly introduces some notations and preliminaries. Section 3 gives the scheme of the proposed algorithm and its convergence analysis including the subsequential convergence, the whole sequence convergence and the convergence rate. Section 4 exhibits the numerical results on simulated data and real data for low matrix recovery problem. The last section is a brief conclusion.

2 Notations and preliminaries

For an extended-real-valued function $h : \mathbb{R}^{m\times n} \to [-\infty, \infty]$, and denote its domain by $\mathrm{dom}h = \{X \in \mathbb{R}^{m\times n} : h(X) < \infty\}$. The function h is said to be proper if its value is not equal to $-\infty$ and $\mathrm{dom}h \neq \varnothing$, and is said to be closed if it is lower semicontinuous.

For any proper closed function $h : \mathbb{R}^{m\times n} \to \mathbb{R}\bigcup\{\infty\}$, the (limiting) subdifferential [24, 48] of h at $X \in \mathrm{dom}h$ is written as

$$\partial h(X) := \left\{\nu \in \mathbb{R}^{m\times n} : \exists X^k \xrightarrow{h} X, \nu^k \to \nu \text{ with } \nu^k \in \widehat{\partial} h(X^k) \text{ for all k}\right\},$$

where $\widehat{\partial} h(Z)$ denotes the Fréchet subdifferential of h at $Z \in \mathrm{dom}h$, which is defined as

$$\widehat{\partial} h(Z) := \left\{\nu \in \mathbb{R}^{m\times n} : \liminf_{Y\neq Z, Y\to Z} \frac{h(Y) - h(Z) - \langle \nu, Y - Z\rangle}{\|Y - Z\|} \geq 0\right\},$$

and $X^k \xrightarrow{h} X$ means $X^k \to X$ and $h(X^k) \to h(X)$.

It is known from [48] that if h is convex, then the subdifferential of h can be expressed as

$$\partial h(X) = \{\nu \in \mathbb{R}^{m\times n} : h(Y) - h(X) - \langle \nu, Y - X\rangle \geq 0 \text{ for each } Y \in \mathbb{R}^{m\times n}\}.$$

If h is continuously differentiable, then $\partial h(X) = \{\nabla h(X)\}$, which is a singleton of the gradient of h, see [48]. Denote $\mathrm{dom}\partial h := \{X \in \mathbb{R}^{m\times n} : \partial h(X) \neq \varnothing\}$.

In the following part of this section, we provide some useful lemmas needed in the later analysis.

Lemma 2.1 *[19, 20] For any two matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{m\times n}$, it holds*

$$\|\sigma(A) - \sigma(B)\|_2^2 \leq \|A - B\|_F^2,$$

where $\sigma(A) = (\sigma_1(A), \cdots, \sigma_{\min(m,n)}(A))^\top$ is the singular value vectors of the matrix A arranged in non-increasing order, that is, $\sigma_1(A) \geq \cdots \geq \sigma_{\min(m,n)}(A)$.

The following lemma can be found in [41, Lemma 2.1], which provides the p-thresholding function (or proximal operator) for the function $|x|^p$. One can refer to [12, 50, 63] for more properties of the p-thresholhing function.

Lemma 2.2 *[41] Let $\tau > 0$, $0 < p < 1$. Define*

$$h_\tau(t) := \mathrm{Arg}\min_{x\geq 0} x^p + \frac{1}{2\tau}(x - t)^2, \quad t \in \mathbb{R}. \tag{2.1}$$

Then

$$h_\tau(t) := \begin{cases} h_{\tau,p}(t), & if \quad t > t^*, \\ \{[2\tau(1-p)]^{1/(2-p)}, 0\}, & if \quad t = t^*, \\ 0, & if \quad t < t^*, \end{cases} \tag{2.2}$$

where $t^* = \frac{2-p}{2(1-p)}[2\tau(1-p)]^{1/(2-p)}$, and $h_{\tau,p}(t)$ is the unique root in $(\bar{x}, +\infty)$ of the following equation:

$$p\tau x^{p-1} + x - t = 0 \tag{2.3}$$

where $\bar{x} = [\tau(1-p)]^{1/(2-p)} > 0$, and $h_{\tau,p}(t)$ is differentiable and strictly increasing on $[t^*, +\infty)$.

As mentioned in [41], equation (2.3) has a unique root in $(\bar{x}, +\infty)$, so $h_{\tau,p}(t)$ can be simply calculated by the Newton method

$$x_{k+1} = x_k - \frac{p\tau x_k^{p-1} + x_k - t}{p(p-1)\tau x_k^{p-2} + 1}$$

with the initial point $x_0 = 1.5\bar{x}$.

Since $h_\tau(t)$ has two values when $t = t^*$, then $h_\tau : \mathbb{R} \rightrightarrows \mathbb{R}_+$ is essentially a set-valued mapping, which is called p-thresholding function.

Proposition 2.3 ($h_\tau(\cdot)$ is monotone) *The p-thresholding function $h_\tau(\cdot)$ is monotone, i.e., for any $y_i^* \in h_\tau(\omega_i)$, $i = 1, 2$, it holds $y_1^* \geq y_2^*$ whenever $\omega_1 > \omega_2$.*

Proof Since x^p is lower bounded and $\lim\limits_{x\to+\infty} x^p = +\infty$, the solution $h_\tau(t)$ of problem (2.1) is finite. By the optimality of problem (2.1) and $y_i^* \in h_\tau(\omega_i), i = 1, 2$, we have

$$(y_2^*)^p + \frac{1}{2\tau}(y_2^* - \omega_1)^2 \geq (y_1^*)^p + \frac{1}{2\tau}(y_1^* - \omega_1)^2, \tag{2.4}$$

$$(y_1^*)^p + \frac{1}{2\tau}(y_1^* - \omega_2)^2 \geq (y_2^*)^p + \frac{1}{2\tau}(y_2^* - \omega_2)^2. \tag{2.5}$$

Summing (2.4) and (2.5), we obtain $(y_1^* - y_2^*) \cdot (\omega_1 - \omega_2) \geq 0$. □

The following lemma provides the proximal operator of the matrix Schatten p-quasi-norm $\|X\|_p^p$, which is called matrix p-thresholding operator in [41].

Lemma 2.4 *[41] Let $\tau > 0$, $0 < p < 1$. Suppose $Y \in \mathbb{R}^{m\times n}$ of rank r admits a singular value decomposition (SVD) as*

$$Y = U\mathrm{Diag}(\sigma)V^\top,$$

where U and V are, respectively $m \times r$ and $n \times r$ matrices with orthonormal columns, and the vector $\sigma = (\sigma_1, \cdots, \sigma_r)^\top$ consists of positive singular values of Y arranged in non-increasing order (unless specified otherwise, we will always suppose the SVD of a matrix is given in this reduced form). Then $H_\tau(Y) := U\mathrm{Diag}(h_\tau(\sigma_1), \cdots, h_\tau(\sigma_r))V^\top$ is the proximal operator of the matrix Schatten p-quasi-norm $\|X\|_p^p$, that is,

$$H_\tau(Y) = \mathrm{Arg} \min_{X\in\mathbb{R}^{m\times n}} \|X\|_p^p + \frac{1}{2\tau}\|X - Y\|_F^2.$$

$H_\tau(Y)$ is called matrix p-thresholding operator.

From [48, Theorem 10.1], the necessary optimality condition of problem (1.5) at the local minimizer X^* is given by $0 \in \partial f_\lambda(X^*) = \nabla\theta(X^*) + \partial(\lambda\|X^*\|_p^p)$.

Definition 2.5 *$X^* \in \mathbb{R}^{m\times n}$ is a critical point of problem (1.5) if*

$$0 \in \partial f_\lambda(X^*) = \nabla\theta(X^*) + \partial(\lambda\|X^*\|_p^p). \tag{2.6}$$

From [41, Theorem 2.5], the necessary optimality condition of problem (1.5) at the global minimizer X^* is described as $X^* \in H_{\lambda\mu}(X^* - \mu\nabla\theta(X^*))$ for some $0 < \mu < \frac{1}{L}$.

Definition 2.6 *$X^* \in \mathbb{R}^{m\times n}$ is called a global stationary point of problem (1.5) if there exists $0 < \mu < \frac{1}{L}$ such that*

$$X^* \in H_{\lambda\mu}(X^* - \mu\nabla\theta(X^*)), \tag{2.7}$$

where L is the Lipschitz constant of $\nabla\theta(\cdot)$.

We now recall the Kurdyka-Łojasiewicz (KŁ) property, which can be satisfied by many functions, such as proper closed semialgebraic functions, and which plays an important role in global convergence analysis and convergence rate of some first-order methods, see, for example[1,2,4,54,57,59,61].

Definition 2.7 *(KŁ property) A proper closed function h is said to satisfy the KŁ property at $\bar{X} \in \mathrm{dom}\partial h$, if there exist $a \in (0, \infty]$, a neighborhood $\mathcal{U}$ of $\bar{X}$, and a continuous and concave function $\psi : [0, a) \to \mathbb{R}_+$ such that*

(i) ψ is continuously differentiable in $(0, a)$ with $\psi(0) = 0$ and $\psi'(t) > 0$ for all $t \in (0, a)$;

(ii) For any $X \in \mathcal{U}$ with $h(\bar{X}) < h(X) < h(\bar{X}) + a$, it holds

$$\psi'(h(X) - h(\bar{X}))\mathrm{dist}(0, \partial h(X)) \geq 1.$$

A proper closed function h is called a KŁ function if it satisfies the KŁ property at all points in $\mathrm{dom}\partial h$.

Lemma 2.8 *[4] (Uniformized KŁ property) Suppose that h is a proper closed function and let Γ be a compact set. If h is a constant on Γ and satisfies the KŁ property at each point of Γ, then there exist $\epsilon > 0, a > 0$ and a continuous and concave function $\psi : [0, a) \to \mathbb{R}_+$ such that*

(i) ψ is continuously differentiable in $(0, a)$ with $\psi(0) = 0$ and $\psi'(t) > 0$ for all $t \in (0, a)$;

(ii) For any $\bar{X} \in \Gamma$ and any $X \in \mathcal{U}$ with $\mathrm{dist}(X, \Gamma) < \epsilon$ and $h(\bar{X}) < h(X) < h(\bar{X}) + a$, it holds

$$\psi'(h(X) - h(\bar{X}))\mathrm{dist}(0, \partial h(X)) \geq 1. \tag{2.8}$$

3 FPCAe algorithm for problem (1.5) and its convergence analysis

In this section, we first present a fixed point algorithm with extrapolation (FPe algorithm) for solving problem (1.5). Secondly, we analyze the global subsequential convergence of the FPe algorithm. Thirdly, we analyze the global whole sequence convergence and convergence rate of the FPe algorithm under KŁ property. Finally, two acceleration technologies, i.e. the continuation of the parameter and the approximate SVD, are employed to the FPe algorithm to get the FPCAe algorithm.

3.1 Scheme of FPe algorithm for problem (1.5)

In order to analyze and solve problem (1.5), we need an auxiliary function. Define

$$Q_{\lambda,\mu}(X, Y) := \theta(Y) + \langle \nabla\theta(Y), X - Y \rangle + \frac{1}{2\mu}\|X - Y\|_F^2 + \lambda\|X\|_p^p.$$

Note that it holds $f_\lambda(X) = Q_{\lambda,\mu}(X, X)$ for all $X \in \mathbb{R}^{m\times n}$.

The main iteration of FPe for problem (1.5) is as below. Suppose X^{k-1} and X^k have been obtained, let $Y^k = X^k + \beta_k(X^k - X^{k-1})$, where $\beta_k > 0$ is an extrapolation coefficient, then X^{k+1} is generated by solving problem $\min_X Q_{\lambda,\mu}(X, Y^k)$, that is,

$$\begin{aligned} X^{k+1} &\in \mathrm{Arg}\min_{X\in\mathbb{R}^{m\times n}} Q_{\lambda,\mu}(X, Y^k) \\ &= \mathrm{Arg}\min_{X\in\mathbb{R}^{m\times n}} \frac{1}{2\mu}\|X - (Y^k - \mu\nabla\theta(Y^k)\|_F^2 + \lambda\|X\|_p^p \\ &= H_{\lambda\mu}\left(Y^k - \mu\nabla\theta(Y^k)\right). \end{aligned} \tag{3.1}$$

The detailed scheme of FPe algorithm for problem (1.5) is in Algorithm 1.

Algorithm 1 FPe algorithm for problem (1.5)

For the given parameter $\lambda > 0$, select a nonnegative sequence $\{\beta_k\}$;
Initialize: Take $X^{-1} = X^0 \in \mathbb{R}^{m\times n}$, and set $k = 0$;
Step1. Compute Y^k by

$$Y^k = X^k + \beta_k(X^k - X^{k-1}); \tag{3.2}$$

Step2. Compute G^k by $G^k = Y^k - \mu\nabla\theta(Y^k)$ and its SVD:

$$G_k = U^k \mathrm{Diag}(\sigma^k)(V^k)^T; \tag{3.3}$$

Step3. Compute X^{k+1} by

$$\begin{aligned} X^{k+1} &= U^k \mathrm{Diag}(h_{\lambda\mu}(\sigma_1^k), ..., h_{\lambda\mu}(\sigma^k_{\min\{m,n\}}))(V^k)^T \\ &\in H_{\lambda\mu}\left(Y^k - \mu\nabla\theta(Y^k)\right); \end{aligned} \tag{3.4}$$

Step4. If some stop criteria is attained, let $X^* = X^{k+1}$;
Otherwise, let $k := k + 1$ and return to **Step 1**.
Output: X^*, $r^* = \mathrm{rank}(X^*)$.

3.2 Global subsequential convergence of Algorithm 1

In this subsection, we give the general convergence results of Algorithm 1 for problem (1.5). For convenience, we first define the following auxiliary function and quantity

$$\Phi_\rho(X, Y) := f_\lambda(X) + \rho\|X - Y\|_F^2, \tag{3.5}$$

$$\Phi_{k,\rho} := \Phi_\rho(X^k, X^{k-1}), \tag{3.6}$$

where $\rho > 0$ is a parameter.

In the following discussion, we always suppose the parameters $\{\alpha, \beta_k, L, \mu, \rho\}$ satisfy the following **conditions**:

- $0 < \alpha < 1, \quad 0 < \mu L < 1, \quad 0 < \beta_k \le \gamma := \sqrt{\dfrac{\alpha(1-\alpha)(1-\mu L)}{1-(1-\alpha)\mu L}};$ (3.7)
- $\dfrac{\gamma^2}{2}\Big(L + \dfrac{1}{\alpha}(\dfrac{1}{\mu} - L)\Big) < \rho \le \dfrac{1}{2}\Big(\dfrac{1}{\mu} - L\Big)(1-\alpha).$ (3.8)

Lemma 3.1 *Let $\{X^k\}$ be a sequence generated by Algorithm 1, then*

$$\begin{aligned} \Phi_{k+1,\rho} - \Phi_{k,\rho} \le & \left[\frac{\beta_k^2}{2}\left(L + \frac{1}{\alpha}(\frac{1}{\mu} - L)\right) - \rho\right]\|X^k - X^{k-1}\|_F^2 \\ & - \left[\frac{1}{2}\Big(\frac{1}{\mu} - L\Big)(1-\alpha) - \rho\right]\|X^{k+1} - X^k\|_F^2. \end{aligned} \tag{3.9}$$

Proof Since $\nabla\theta$ is Lipschitz continuous with modulus $L > 0$ as well as (3.1), we have

$$\begin{aligned}
f_\lambda(X^{k+1}) &= \lambda\|X^{k+1}\|_p^p + \theta(X^{k+1}) \\
&\le \lambda\|X^{k+1}\|_p^p + \theta(Y^k) + \langle\nabla\theta(Y^k), X^{k+1} - Y^k\rangle + \frac{L}{2}\|X^{k+1} - Y^k\|_F^2 \\
&= \lambda\|X^{k+1}\|_p^p + \theta(Y^k) + \langle\nabla\theta(Y^k), X^{k+1} - Y^k\rangle + \frac{1}{2\mu}\|X^{k+1} - Y^k\|_F^2 \\
&\quad + \frac{1}{2}\Big(L - \frac{1}{\mu}\Big)\|X^{k+1} - Y^k\|_F^2 \\
&\le \lambda\|X^k\|_p^p + \theta(Y^k) + \langle\nabla\theta(Y^k), X^k - Y^k\rangle + \frac{1}{2\mu}\|X^k - Y^k\|_F^2 \\
&\quad + \frac{1}{2}\Big(L - \frac{1}{\mu}\Big)\|X^{k+1} - Y^k\|_F^2. \qquad (3.10)
\end{aligned}$$

Due to the convexity of θ, we have

$$\theta(Y^k) + \langle\nabla\theta(Y^k), X^k - Y^k\rangle \le \theta(X^k). \qquad (3.11)$$

From (3.10), (3.11) and the definition of Y^k, we obtain immediately that

$$\begin{aligned}
&f_\lambda(X^{k+1}) - f_\lambda(X^k) \\
&\le \frac{1}{2\mu}\|X^k - Y^k\|_F^2 - \frac{1}{2}\left(\frac{1}{\mu} - L\right)\|X^{k+1} - Y^k\|_F^2 \\
&= \frac{1}{2\mu}\beta_k^2\|X^k - X^{k-1}\|_F^2 - \frac{1}{2}\left(\frac{1}{\mu} - L\right)\|X^{k+1} - X^k - \beta_k(X^k - X^{k-1})\|_F^2 \\
&= \frac{L}{2}\beta_k^2\|X^k - X^{k-1}\|_F^2 - \frac{1}{2}\left(\frac{1}{\mu} - L\right)\|X^{k+1} - X^k\|_F^2 \\
&\quad + \left(\frac{1}{\mu} - L\right)\langle X^{k+1} - X^k, \beta_k(X^k - X^{k-1})\rangle. \qquad (3.12)
\end{aligned}$$

Moreover, for any $\alpha > 0$, it holds from the inequality $ab \le \frac{\alpha}{2}a^2 + \frac{1}{2\alpha}b^2$ that

$$\langle X^{k+1} - X^k, \beta_k(X^k - X^{k-1})\rangle \le \frac{\alpha}{2}\|X^{k+1} - X^k\|_F^2 + \frac{1}{2\alpha}\beta_k^2\|X^k - X^{k-1}\|_F^2. \qquad (3.13)$$

Summing (3.12) and (3.13), we get that

$$\begin{aligned}
&f_\lambda(X^{k+1}) - f_\lambda(X^k) \\
&\le \frac{L}{2}\beta_k^2\|X^k - X^{k-1}\|_F^2 - \frac{1}{2}\left(\frac{1}{\mu} - L\right)\|X^{k+1} - X^k\|_F^2 \\
&\quad + \frac{\alpha}{2}\left(\frac{1}{\mu} - L\right)\|X^{k+1} - X^k\|_F^2 + \frac{1}{2\alpha}\left(\frac{1}{\mu} - L\right)\beta_k^2\|X^k - X^{k-1}\|_F^2 \\
&= \frac{\beta_k^2}{2}\left(L + \frac{1}{\alpha}(\frac{1}{\mu} - L)\right)\|X^k - X^{k-1}\|_F^2 - \frac{1}{2}\Big(\frac{1}{\mu} - L\Big)(1-\alpha)\|X^{k+1} - X^k\|_F^2. \qquad (3.14)
\end{aligned}$$

By (3.14) and the definition of $\Phi_{k,\rho}$ in (3.6), we obtain

$$\begin{aligned}
&\Phi_{k+1,\rho} - \Phi_{k,\rho} \\
&= f_\lambda(X^{k+1}) + \rho\|X^{k+1} - X^k\|_F^2 - f_\lambda(X^k) - \rho\|X^k - X^{k-1}\|_F^2 \\
&\leq \frac{\beta_k^2}{2}\left(L + \frac{1}{\alpha}(\frac{1}{\mu} - L)\right)\|X^k - X^{k-1}\|_F^2 - \frac{1}{2}\Big(\frac{1}{\mu} - L\Big)(1-\alpha)\|X^{k+1} - X^k\|_F^2 \\
&\quad +\rho\|X^{k+1} - X^k\|_F^2 - \rho\|X^k - X^{k-1}\|_F^2 \\
&= \left[\frac{\beta_k^2}{2}\left(L + \frac{1}{\alpha}(\frac{1}{\mu} - L)\right) - \rho\right]\|X^k - X^{k-1}\|_F^2 - \left[\frac{1}{2}\Big(\frac{1}{\mu} - L\Big)(1-\alpha) - \rho\right]\|X^{k+1} - X^k\|_F^2.
\end{aligned}$$

This completes the proof. □

Lemma 3.2 *Let $\{X^k\}$ be a sequence generated by Algorithm 1, then the following statements hold.*

(i) The sequence $\{\Phi_{k,\rho}\}$ is non-increasing and convergent. Moreover,

$$\Phi_{k+1,\rho} - \Phi_{k,\rho} \leq \left[\frac{\beta_k^2}{2}\left(L + \frac{1}{\alpha}(\frac{1}{\mu} - L)\right) - \rho\right]\|X^k - X^{k-1}\|_F^2. \tag{3.15}$$

(ii) $\sum_{k=0}^{\infty}\|X^{k+1} - X^k\|_F^2 < \infty$.

Proof (i) From conditions (3.7) and (3.8), we have

$$\left[\frac{\beta_k^2}{2}\left(L + \frac{1}{\alpha}(\frac{1}{\mu} - L)\right) - \rho\right] < 0 \quad \text{and} \quad \left[\frac{1}{2}\Big(\frac{1}{\mu} - L\Big)(1-\alpha) - \rho\right] \geq 0. \tag{3.16}$$

Then from Lemma 3.1, we get

$$\Phi_{k+1,\rho} - \Phi_{k,\rho} \leq 0,$$

which means that $\{\Phi_{k,\rho}\}$ is non-increasing. Since $\theta(\cdot)$ and $\|\cdot\|_p^p$ are both bounded from below, we know that $\{f_\lambda(X^k)\}$ and $\{\Phi_{k,\rho}\}$ are both bounded from below, and then $\{\Phi_{k,\rho}\}$ is convergent.

(ii) From Lemma 3.1 and (3.16), we obtain

$$0 \leq \left[\rho - \frac{\beta_k^2}{2}\left(L + \frac{1}{\alpha}(\frac{1}{\mu} - L)\right)\right]\|X^k - X^{k-1}\|_F^2 \leq \Phi_{k,\rho} - \Phi_{k+1,\rho}. \tag{3.17}$$

Summing both sides of (3.17) for k from 0 to N, we have

$$\begin{aligned}
0 &\leq \sum_{k=0}^{N}\left[\rho - \frac{\beta_k^2}{2}\left(L + \frac{1}{\alpha}(\frac{1}{\mu} - L)\right)\right]\|X^k - X^{k-1}\|_F^2 \\
&\leq \sum_{k=0}^{N}(\Phi_{k,\rho} - \Phi_{k+1,\rho}) = \Phi_{0,\rho} - \Phi_{N+1,\rho}.
\end{aligned}$$

Let $N \to \infty$, then the convergence of $\{\Phi_{k,\rho}\}$ implies that

$$\sum_{k=0}^{\infty}\left[\rho - \frac{\beta_k^2}{2}\left(L + \frac{1}{\alpha}(\frac{1}{\mu} - L)\right)\right]\|X^k - X^{k-1}\|_F^2 < \infty. \tag{3.18}$$

Conditions (3.7) and (3.8) yield that

$$\left[\rho - \frac{\beta_k^2}{2}\left(L + \frac{1}{\alpha}(\frac{1}{\mu} - L)\right)\right] \geq \left[\rho - \frac{\gamma^2}{2}\left(L + \frac{1}{\alpha}(\frac{1}{\mu} - L)\right)\right] > 0.$$

Hence, from (3.18), we get $\sum\limits_{k=0}^{\infty} \|X^{k+1} - X^k\|_F^2 < \infty$. □

Theorem 3.3 (Global subsequential convergence) *Let $\{X^k\}$ be a sequence generated by Algorithm 1. Then the following statements hold.*

(i) *The sequence $\{X^k\}$ is bounded;*

(ii) *Any accumulation point of $\{X^k\}$ is both a critical point and a global stationary point of problem (1.5).*

(iii) *There exists a constant Φ^* such that $f_\lambda(X^*) \equiv \Phi(X^*, X^*) \equiv \Phi^*$ for any accumulation point X^* of $\{X^k\}$.*

Proof (i) By the definition of $\Phi_{k,\rho}$ (3.6), we have

$$\begin{aligned}\Phi_{k,\frac{1}{2}(\frac{1}{\mu}-L)(1-\alpha)} &= \Phi_{k,\frac{1}{2}(\frac{1}{\mu}-L)(1-\alpha)}(X^k, X^{k-1}) \\ &= f_\lambda(X^k) + \frac{1}{2}(\frac{1}{\mu} - L)(1-\alpha)\|X^k - X^{k-1}\|_F^2.\end{aligned}$$

Since $\{\Phi_{k,\frac{1}{2}(\frac{1}{\mu}-L)(1-\alpha)}\}$ is non-increasing and $X^{-1} = X^0$ as well as condition (3.7), then

$$f_\lambda(X^k) \leq \Phi_{k,\frac{1}{2}(\frac{1}{\mu}-L)(1-\alpha)}(X^k, X^{k-1}) \leq \Phi_{0,\frac{1}{2}(\frac{1}{\mu}-L)(1-\alpha)}(X^0, X^{-1}) = f_\lambda(X^0).$$

That is, $\{X^k\} \subset \{X : f_\lambda(X^k) \leq f_\lambda(X^0)\}$. By the lower boundedness of $\theta(\cdot)$, we have

$$\lim_{\|X\|\to\infty} f_\lambda(X) \;\geq\; \inf \theta(X) + \lim_{\|X\|\to\infty} \lambda\|X\|_p^p \quad = \infty.$$

Consequently, both $\{X : f_\lambda(X^k) \leq f_\lambda(X^0)\}$ and $\{X^k\}$ are bounded.

(ii) Let X^* be any accumulation point of $\{X^k\}$, then there exists a subsequence $\{X^{k_i}\}$ such that $\lim_{i\to\infty} X^{k_i} = X^*$. By (3.1),

$$\begin{aligned}X^{k_i+1} &\in H_{\lambda\mu}\left(Y^{k_i} - \mu\nabla\theta(Y^{k_i})\right) \\ &= \mathrm{Arg}\min_{X\in\mathbb{R}^{m\times n}} \frac{1}{2\mu\lambda}\|X - (Y^{k_i} - \mu\nabla\theta(Y^{k_i}))\|_F^2 + \|X\|_p^p,\end{aligned}$$

then for any $X \in \mathbb{R}^{m\times n}$ we have that

$$\begin{aligned}&\frac{1}{2\mu\lambda}\|X^{k_i+1} - (Y^{k_i} - \mu\nabla\theta(Y^{k_i}))\|_F^2 + \|X^{k_i+1}\|_p^p \\ \leq &\frac{1}{2\mu\lambda}\|X - (Y^{k_i} - \mu\nabla\theta(Y^{k_i}))\|_F^2 + \|X\|_p^p. \qquad (3.19)\end{aligned}$$

Since $Y^{k_i} = X^{k_i} + \beta_{k_i}(X^{k_i} - X^{k_i-1})$ and $\|X^{k_i+1} - X^{k_i}\| \to 0$ as $i \to \infty$ (Lemma 3.2 (ii)), then $X^{k_i+1} \to X^*$ and $Y^{k_i} \to X^*$ as $i \to \infty$. Letting $i \to \infty$ in (3.19) and using the continuity of $\nabla\theta(\cdot)$ and $\|\cdot\|_p^p$, we obtain

$$\frac{1}{2\mu\lambda}\|X^* - (X^* - \mu\nabla\theta(X^*))\|_F^2 + \|X^*\|_p^p \leq \frac{1}{2\mu\lambda}\|X - (X^* - \mu\nabla\theta(X^*))\|_F^2 + \|X\|_p^p,$$

which implies that

$$X^* \in \operatorname{Arg} \min_{X\in\mathbb{R}^{m\times n}} \frac{1}{2\mu\lambda}\|X-(X^*-\mu\nabla\theta(X^*))\|_F^2+\|X\|_p^p. \tag{3.20}$$

From Lemma 2.4, we obtain $X^* \in H_{\lambda\mu}(X^*-\mu\nabla\theta(X^*))$, that is, X^* is a global stationary point of problem (1.5).

In addition, the optimality condition of (3.20) gives that

$$0 \in \frac{1}{\mu}\Big(X^*-(X^*-\mu\nabla\theta(X^*))\Big)+\partial(\lambda\|\cdot\|_p^p)_{X=X^*}=\nabla\theta(X^*)+\partial(\lambda\|\cdot\|_p^p)_{X=X^*}=\partial f_\lambda(X^*).$$

That is, X^* is also a critical point of problem (1.5).

(iii) According to (3.17), there exists $\gamma>0$ such that

$$\Phi_\rho(X^k,X^{k-1})-\Phi_\rho(X^{k+1},X^k)\geq\gamma\|X^k-X^{k-1}\|_F^2. \tag{3.21}$$

That is, $\{\Phi_\rho(X^k,X^{k-1})\}$ is non-increasing with respect to k. Note that $\{\Phi_\rho(X^k,X^{k-1})\}$ is bounded from below since $\theta(\cdot)$ is bounded from below. Hence, there exists Φ^* such that $\lim_{k\to\infty}\Phi_\rho(X^k,X^{k-1})=\Phi^*$. For the accumulation point X^* and the subsequence $\{X^{k_i}\}$ converging to X^*, since $\Phi_\rho(\cdot,\cdot)$ is continuous, we have $\Phi^*\equiv\Phi_\rho(X^*,X^*)\equiv f_\lambda(X^*)$. □

Remark 3.4 *Note that it can be deduced from (3.20) that X^* is a critical point of problem (1.5), but the opposite is not true. That is, if X^* is a global stationary point, it must be a critical point of problem (1.5), but the opposite is not true. The global stationary point possesses some global optimality, which can exclude too many stationary points even some local minimizers of the nonconvex problem. One can see an interesting example [62, Example 4.5] for details.*

The following theorem shows that $\{\mathrm{rank}(X^k)\}$ will become a constant when k is sufficiently large and that all the accumulation points of $\{X^k\}$ have a same rank.

Theorem 3.5 (Rank invariance) *Let $\{X^k\}$ be a sequence generated by Algorithm 1. Then there exist two positive integers k_0 and r such that, whenever $k>k_0$,*

$$\mathrm{rank}(X^k)=\mathrm{rank}(X^*)\equiv r,$$

where X^* is any accumulation point of $\{X^k\}$.

Proof (i) On the one hand, from (2.2) and (3.4), we know that

$$\sigma_i(X^k)=h_{\lambda\mu}(\sigma_i(G^k))\geq\tau_p:=[2\lambda\mu(1-p)]^{\frac{1}{(2-p)}},\quad \text{if}\quad \sigma_i(X^k)\neq 0, \tag{3.22}$$

for each $i=1,\cdots,\min\{m,n\}$ and any $k>0$. Let $r^k:=\mathrm{rank}(X^k)$. If $\mathrm{rank}(X^{k+1})\neq\mathrm{rank}(X^k)$ (i.e., $\mathrm{rank}(X^{k+1})>\mathrm{rank}(X^k)$ or $\mathrm{rank}(X^{k+1})<\mathrm{rank}(X^k)$), then from (3.22) and the monotonicity of $h_{\lambda\mu}(\cdot)$ (Proposition (2.3)), we have

$$\|\sigma(X^{k+1})-\sigma(X^k)\|_2\geq\min\{\sigma_{r^{k+1}}(X^{k+1}),\sigma_{r^k}(X^k)\}\geq\tau_p. \tag{3.23}$$

On the other hand, by Lemma 3.2 (ii), $\|X^{k+1}-X^k\|_F\to 0$, then there exists a sufficiently large positive integer k_0 such that whenever $k>k_0$ it holds

$$\|X^{k+1}-X^k\|_F<\tau_p,$$

which together with Lemma 2.1 yields that

$$\|\sigma(X^{k+1})-\sigma(X^k)\|_2\leq\|X^{k+1}-X^k\|_F<\tau_p, \tag{3.24}$$

which contradicts to (3.23). This contradiction shows that the equality $\text{rank}(X^{k+1}) = \text{rank}(X^k)$ must hold for any sufficiently large positive integer k. That is, $\text{rank}(X^k)$ must be a constant r whenever $k > k_0$ for some sufficiently large positive integer k_0.

(ii) For any accumulation point X^*, there exists a subsequence $\{X^{k_j}\}$ converging to X^*. Then there exists a sufficiently large positive integer $j_0 > k_0$ such that whenver $j > j_0$,

$$\text{rank}(X^{k_j}) = r \text{ and } \|X^{k_j} - X^*\|_F < \tau_p.$$

Similar to (i), we can obtain

$$\text{rank}(X^{k_j}) = \text{rank}(X^*).$$

Thus, we get $\text{rank}(X^*) = r$. This together with (i) yields that $\text{rank}(X^k) = \text{rank}(X^*) \equiv r$ whenever $k > k_0$. □

3.3 Global whole sequence convergence and convergence rate of Algorithm 1

In this subsection, we discuss the global whole sequence convergence and convergence rate of Algorithm 1 under the additional assumption that the function $\Phi_\rho(\cdot,\cdot)$ (defined in (3.5)) is a KŁ function.

Theorem 3.6 (Global whole sequence convergence) *Let $\{X^k\}$ be a sequence generated by Algorithm 1. Suppose Φ_ρ is a KŁ function. Then the following statements hold.*

(i) $\lim_{k\to\infty} \text{dist}((0,0), \partial\Phi_\rho(X^k, X^{k-1})) = 0$;

(ii) $\sum_{k=1}^{\infty} \|X^k - X^{k-1}\|_F < \infty$;

(iii) *The whole sequence $\{X^k\}$ converges to a critical point and global stationary point of problem (1.5).*

Proof (i) We first consider the subdifferential of Φ_ρ at (X^k, X^{k-1}). Note that for any $k > 0$, we have

$$\begin{aligned}\partial\Phi_\rho(X^k, X^{k-1}) &= \partial_X\Phi_\rho(X^k, X^{k-1}) \times \partial_Y\Phi_\rho(X^k, X^{k-1}) \\ &= \Big\{\nabla\theta(X^k) + \partial(\lambda\|X^k\|_p^p) + 2\rho(X^k - X^{k-1})\Big\} \times \Big\{-2\rho(X^k - X^{k-1})\Big\}.\end{aligned} \tag{3.25}$$

The optimality condition of subproblem (3.1) yields that

$$0 \in \frac{1}{\mu}(X^k - Y^{k-1}) + \nabla\theta(Y^{k-1}) + \partial(\lambda\|X^k\|_p^p). \tag{3.26}$$

Substitute (3.26) into (3.25), we obtain

$$\Big\{2\rho(X^k - X^{k-1}) - \frac{1}{\mu}(X^k - Y^{k-1}) + \nabla\theta(X^k) - \nabla\theta(Y^{k-1})\Big\} \times \Big\{-2\rho(X^k - X^{k-1})\Big\} \\ \in \partial\Phi_\rho(X^k, X^{k-1}).$$

By $Y^{k-1} = X^{k-1} + \beta_{k-1}(X^{k-1} - X^{k-2})$, we have

$$\Big\{(2\rho - \frac{1}{\mu})(X^k - X^{k-1}) + \frac{\beta_{k-1}}{\mu}(X^{k-1} - X^{k-2}) + \nabla\theta(X^k) \\ - \nabla\theta\left(X^{k-1} + \beta_{k-1}(X^{k-1} - X^{k-2})\right)\Big\} \times \Big\{-2\rho(X^k - X^{k-1})\Big\} \in \partial\Phi_\rho(X^k, X^{k-1}).$$

Considering the Lipschitz continuity of $\nabla\theta(\cdot)$, the above relation implies that there exists $M>0$ such that

$$\mathrm{dist}((0,0),\partial\Phi_\rho(X^k,X^{k-1}))\leq M(\|X^k-X^{k-1}\|_F+\|X^{k-1}-X^{k-2}\|_F). \tag{3.27}$$

From Lemma 3.2 (ii), $\|X^{k+1}-X^k\|_F\to 0$ $(k\to\infty)$, we obtain that

$$\lim_{k\to\infty}\mathrm{dist}((0,0),\partial\Phi_\rho(X^k,X^{k-1}))=0.$$

(ii) From Theorem 3.3 (iii) and its proof, for any accumulation point X^* of $\{X^k\}$, we have

$$\lim_{k\to\infty}\Phi_\rho(X^k,X^{k-1})=\Phi_\rho(X^*,X^*)=f_\lambda(X^*)=\Phi^*.$$

Note that $\Phi_\rho(X^k,X^{k-1})=f_\lambda(X^k)+\rho\|X^k-X^{k-1}\|_F^2$ and $\|X^k-X^{k-1}\|_F\to 0$ as $k\to\infty$ (Lemma 3.2 (ii)), then by the continuity of f_λ and $\Phi_\rho(\cdot,\cdot)$, we get

$$\lim_{k\to\infty}f_\lambda(X^k)=\lim_{k\to\infty}\Phi_\rho(X^k,X^{k-1})=f_\lambda(X^*)=\Phi^*. \tag{3.28}$$

Since $\{\Phi_\rho(X^k,X^{k-1})\}$ is non-increasing, then $\Phi_\rho(X^k,X^{k-1})\geq\Phi^*$ for all k.

We next consider the following two cases:

- $\Phi_\rho(X^k,X^{k-1})=\Phi^*$;
- $\Phi_\rho(X^k,X^{k-1})>\Phi^*$.

(1) Consider the case that $\Phi_\rho(X^k,X^{k-1})=\Phi^*$ for some $k=\hat{k}$. Since $\Phi_\rho(X^k,X^{k-1})$ is non-increasing and converges to Φ^*, then $\Phi_\rho(X^k,X^{k-1})\equiv\Phi^*$ for all $k\geq\hat{k}$. From (3.21), we obtain that $X^{k+\hat{k}}=X^{\hat{k}}$ for all $k\geq 1$. This means that $\{X^k\}$ has only finite number of elements.

(2) Consider the case that $\Phi_\rho(X^k,X^{k-1})>\Phi^*$ for all $k\geq 0$. Denote the set of accumulation points of $\{X^k\}$ by Γ. If $X^{k_i}\to X^*$, then from $\|X^{k_i+1}-X^{k_i}\|\to 0$, we can get $X^{k_i+1}\to X^*$, which together with the definition of Γ implies that $\Omega:=\{(X,X):X\in\Gamma\}$ is the set of accumulation points of $\{(X^k,X^{k-1})\}$. Consequently, we have $\lim_{k\to\infty}\mathrm{dist}((X^k,X^{k-1}),\Omega)=0$. Thus, there exists $N_1>0$ such that

$$\mathrm{dist}((X^k,X^{k-1}),\Omega)<\epsilon \tag{3.29}$$

for any $\epsilon>0$ and any $k\geq N_1$. From Theorem 3.3 (iii), we know that Φ_ρ is a constant on Ω. Since Φ_ρ is a KŁ function, by the property of KŁ functions (Lemma 2.8), there exist $a\in(0,+\infty]$, $\epsilon>0$ and a continuous concave function $\psi:[0,a)\to\mathbb{R}_+$ such that

$$\psi'(\Phi_\rho(X,Y)-\Phi^*)\cdot\mathrm{dist}((0,0),\partial\Phi_\rho(X,Y))\geq 1 \tag{3.30}$$

for all $(X,Y)\in\mathcal{U}$, where

$$\begin{aligned}\mathcal{U}=&\Big\{(X,Y)\in\mathbb{R}^{m\times n}\times\mathbb{R}^{m\times n}:\ \mathrm{dist}((X,Y),\Omega)<\epsilon\Big\}\\ &\bigcap\Big\{(X,Y)\in\mathbb{R}^{m\times n}\times\mathbb{R}^{m\times n}:\ \Phi^*<\Phi_\rho(X,Y)<\Phi^*+a\Big\}.\end{aligned}$$

Similarly, since $\{\Phi_\rho(X^k,X^{k-1})\}$ is non-increasing and converges to Φ^* with $\Phi_\rho(X^k,X^{k-1})>\Phi^*$ for all k, then there exists $N_2>0$ such that $\Phi^*<\Phi_\rho(X^k,X^{k-1})<\Phi^*+a$ whenever $k\geq N_2$. This together with (3.29) gives that

$$\{(X^k,X^{k-1})\}_{k\geq N_3}\subset\mathcal{U} \tag{3.31}$$

whenever $k \geq N_3 := \max\{N_1, N_2\}$. Thus, from (3.30), we obtain that

$$\psi'(\Phi_\rho(X^k, X^{k-1}) - \Phi^*) \cdot \mathrm{dist}((0,0), \partial\Phi_\rho(X^k, X^{k-1})) \geq 1 \tag{3.32}$$

whenever $k \geq N_3$. By use of the concavity of ψ and (3.32), we have that

$$\begin{aligned} &[\psi(\Phi_\rho(X^k, X^{k-1}) - \Phi^*) - \psi(\Phi_\rho(X^{k+1}, X^k) - \Phi^*)] \cdot \mathrm{dist}((0,0), \partial\Phi_\rho(X^k, X^{k-1})) \\ \geq\ & \psi'(\Phi_\rho(X^k, X^{k-1}) - \Phi^*) \cdot \mathrm{dist}((0,0), \partial\Phi_\rho(X^k, X^{k-1})) \cdot (\Phi_\rho(X^k, X^{k-1}) - \Phi_\rho(X^{k+1}, X^k)) \\ \geq\ & \Phi_\rho(X^k, X^{k-1}) - \Phi_\rho(X^{k+1}, X^k). \end{aligned} \tag{3.33}$$

By (3.21), (3.27) and (3.33), we get that

$$\begin{aligned} \|X^k - X^{k-1}\|_F^2 &\leq \frac{1}{\gamma}[\Phi_\rho(X^k, X^{k-1}) - \Phi_\rho(X^{k+1}, X^k)] \\ &\leq \frac{1}{\gamma}[\psi(\Phi_\rho(X^k, X^{k-1}) - \Phi^*) - \psi(\Phi_\rho(X^{k+1}, X^k) - \Phi^*)] \\ &\quad \cdot \mathrm{dist}((0,0), \partial\Phi_\rho(X^k, X^{k-1})) \\ &\leq \frac{M}{\gamma}[\psi(\Phi_\rho(X^k, X^{k-1}) - \Phi^*) - \psi(\Phi_\rho(X^{k+1}, X^k) - \Phi^*)] \\ &\quad \cdot (\|X^k - X^{k-1}\|_F + \|X^{k-1} - X^{k-2}\|_F). \end{aligned} \tag{3.34}$$

Taking square root on both sides of (3.34), and by the inequality $\sqrt{|ab|} \leq \frac{|a|+|b|}{2}$, we obtain

$$\begin{aligned} \|X^k - X^{k-1}\|_F &\leq \sqrt{\frac{2M}{\gamma}\Big[\psi(\Phi_\rho(X^k, X^{k-1}) - \Phi^*) - \psi(\Phi_\rho(X^{k+1}, X^k) - \Phi^*)\Big]} \\ &\quad \cdot \sqrt{\frac{\|X^k - X^{k-1}\|_F + \|X^{k-1} - X^{k-2}\|_F}{2}} \\ &\leq \frac{M}{\gamma}\Big[\psi(\Phi_\rho(X^k, X^{k-1}) - \Phi^*) - \psi(\Phi_\rho(X^{k+1}, X^k) - \Phi^*)\Big] \\ &\quad + \frac{1}{4}\|X^k - X^{k-1}\|_F + \frac{1}{4}\|X^{k-1} - X^{k-2}\|_F. \end{aligned}$$

Rearranging the terms of the above inequality, we get

$$\begin{aligned} \frac{1}{2}\|X^k - X^{k-1}\|_F &\leq \frac{M}{\gamma}\Big[\psi(\Phi_\rho(X^k, X^{k-1}) - \Phi^*) - \psi(\Phi_\rho(X^{k+1}, X^k) - \Phi^*)\Big] \\ &\quad + \frac{1}{4}(\|X^{k-1} - X^{k-2}\|_F - \|X^k - X^{k-1}\|_F). \end{aligned} \tag{3.35}$$

Summing both sides of (3.35) for k from N_3 to ∞, and by the continuity of ψ, we obtain

$$\sum_{k=N_3}^{\infty} \|X^k - X^{k-1}\|_F \leq \frac{2M}{\gamma}\psi(\Phi_\rho(X^{N_3}, X^{N_3-1}) - \Phi^*) + \frac{1}{2}\|X^{N_3-1} - X^{N_3-2}\|_F < \infty.$$

This means that the statement (ii) holds and that $\{X^k\}$ is a Cauchy sequence which implies that $\{X^k\}$ is convergent. Then by Theorem 3.3 (ii), $\{X^k\}$ converges to a critical point and global stationary point of problem (1.5). □

At the end of this subsection, we derive the convergence rate of Algorithm 1 under the assumption that Φ_ρ is a KŁ function. Our analysis is similar to that in [1,25,54].

Theorem 3.7 (Convergence rate) *Let $\{X^k\}$ be a sequence generated by Algorithm 1. Suppose Φ_ρ is a KŁ function such that the function ψ in the KŁ inequality (3.30) takes the form $\psi(s) = cs^{1-\vartheta}$ for some $\vartheta \in [0, 1)$ and $c > 0$. Then the following statements hold.*

(i) [Finite termination] *If $\vartheta = 0$, then X^k is a constant for all sufficiently large k;*

(ii) [R-linear convergence rate] *If $\vartheta \in (0, \frac{1}{2}]$, then there exists $\eta \in (0, 1)$ and $\kappa_1 > 0$ such that $\|X^k - X^*\|_F < \kappa_1 \eta^k$ for all sufficiently large k;*

(iii) [R-sublinear convergence rate] *If $\vartheta \in (\frac{1}{2}, 1)$, then there exists $\kappa_2 > 0$ such that $\|X^k - X^*\|_F < \kappa_2 k^{-\frac{1-\vartheta}{2\vartheta-1}}$ for all sufficiently large k.*

Proof (i) If $\vartheta = 0$, we deduce that there must exists $k_0 > 0$ such that $\Phi_\rho(X^{k_0}, X^{k_0-1}) = \Phi^*$, where Φ^* is given in Theorem 3.3. Assume on the contrary that $\Phi_\rho(X^k, X^{k-1}) > \Phi^*$ for all $k > 0$ since $\{\Phi_\rho(X^k, X^{k-1})\}$ is non-increasing and convergent to Φ^* by Theorem 3.3. It follows from $\psi(s) = cs$ and the KŁ inequality (3.30) that

$$\mathrm{dist}((0, 0), \partial\Phi_\rho(X^k, X^{k-1}) \geq \frac{1}{c} \tag{3.36}$$

for all sufficiently large k, which contradicts (3.29). Since $\Phi_\rho(X^k, X^{k-1})$ is non-increasing and converges to Φ^*, then $\Phi_\rho(X^k, X^{k-1}) \equiv \Phi^*$ for all $k \geq k_0$. From (3.21), we obtain that $X^{k+k_0} = X^{k_0}$ for all $k \geq 1$. This means that X^k is a constant for all sufficiently large k.

(ii) When $\vartheta \in (0, 1)$, let $d_k = \sum_{t=k}^{\infty} \|X^{t+1} - X^t\|_F$, then from Theorem 3.6 (ii), d_k is well defined and non-increasing. It follows from (3.35) that for all $k \geq N_3$,

$$\begin{aligned} d_k &\leq \frac{2M}{\gamma}\psi(\Phi_\rho(X^k, X^{k-1}) - \Phi^*) + \frac{1}{2}\|X^{k-1} - X^{k-2}\|_F \\ &= \frac{2M}{\gamma}\psi(\Phi_\rho(X^k, X^{k-1}) - \Phi^*) + \frac{1}{2}(d_{k-2} - d_{k-1}). \end{aligned} \tag{3.37}$$

From (3.27), (3.37) and d_k is non-increasing, we have

$$d_k \leq \frac{2M}{\gamma}\psi(\Phi_\rho(X^k, X^{k-1}) - \Phi^*) + \frac{1}{2}(d_{k-2} - d_k) \tag{3.38}$$

and

$$\mathrm{dist}((0, 0), \partial\Phi_\rho(X^k, X^{k-1})) \leq M(d_{k-2} - d_k). \tag{3.39}$$

From (3.30), $\lim_{k\to\infty} X^k = X^*$ and $\psi'(s) = c(1 - \vartheta)s^{-\vartheta}$, it follows that

$$c(1 - \vartheta)(\Phi_\rho(X^k, X^{k-1}) - \Phi^*)^{-\vartheta} \cdot \mathrm{dist}((0, 0), \partial\Phi_\rho(X^k, X^{k-1})) \geq 1 \tag{3.40}$$

for all $k \geq N_3$. Combining (3.39) and (3.40), we have

$$(\Phi_\rho(X^k, X^{k-1}) - \Phi^*)^{\vartheta} \leq M \cdot c(1 - \vartheta)(d_{k-2} - d_k). \tag{3.41}$$

Due to $\psi(s) = cs^{1-\vartheta}$ and (3.41), we get

$$\begin{aligned} \psi(\Phi_\rho(X^k, X^{k-1}) - \Phi^*) &= c(\Phi_\rho(X^k, X^{k-1}) - \Phi^*)^{1-\vartheta} \\ &\leq c\big(M \cdot c(1 - \vartheta)(d_{k-2} - d_k)\big)^{(1-\vartheta)/\vartheta}. \end{aligned} \tag{3.42}$$

From (3.37) and (3.42), for all $k \geq N_3$ it follows that

$$\begin{aligned} d_k &\leq \frac{2M}{\gamma}\psi(\Phi_\rho(X^k, X^{k-1}) - \Phi^*) + \frac{1}{2}(d_{k-2} - d_k) \\ &\leq \frac{2M}{\gamma}c\big(M \cdot c(1-\vartheta)(d_{k-2} - d_k)\big)^{(1-\vartheta)/\vartheta} + \frac{1}{2}(d_{k-2} - d_k) \\ &= C(d_{k-2} - d_k)^{(1-\vartheta)/\vartheta} + \frac{1}{2}(d_{k-2} - d_k). \end{aligned} \tag{3.43}$$

where $C := \frac{2M}{\gamma}c(M \cdot c(1-\vartheta))^{\frac{1-\vartheta}{\vartheta}} > 0$. By Theorem (3.6) (ii), we have

$$d_{k-2} - d_k = \|X^k - X^{k-1}\|_F + \|X^{k-1} - X^{k-2}\|_F \to 0. \tag{3.44}$$

If $\vartheta \in (0, \frac{1}{2}]$, then $(1-\vartheta)/\vartheta \geq 1$. Then from (3.43) and (3.44), for all sufficiently large k it follows that

$$\begin{aligned} d_k &\leq C(d_{k-2} - d_k)^{(1-\vartheta)/\vartheta} + \frac{1}{2}(d_{k-2} - d_k) \\ &\leq C(d_{k-2} - d_k) + \frac{1}{2}(d_{k-2} - d_k) \leq (C + \frac{1}{2})(d_{k-2} - d_k). \end{aligned}$$

This means that $d_k \leq \frac{2C+1}{2C+3}d_{k-2} = \left(\sqrt{\frac{2C+1}{2C+3}}\right)^2 d_{k-2}$. Thus,

$$\begin{aligned} \|X^k - X^*\|_F &\leq \sum_{t=k}^{\infty} \|X^{t+1} - X^t\|_F = d_k \leq \left(\sqrt{\frac{2C+1}{2C+3}}\right)^2 d_{k-2} \\ &\leq \left(\sqrt{\frac{2C+1}{2C+3}}\right)^4 d_{k-4} \leq \cdots \leq d_{k_1}\left(\sqrt{\frac{2C+1}{2C+3}}\right)^{k-k_1} = \kappa_1 \eta^k, \quad \forall\, k \geq k_1 (\geq N_3), \end{aligned}$$

where $\eta := \sqrt{\frac{2C+1}{2C+3}} \in (0,1)$ and $\kappa_1 := d_{k_1}\eta^{-k_1} > 0$.

(iii) If $\vartheta \in (\frac{1}{2}, 1)$, then $\frac{1-\vartheta}{\vartheta} \in (0,1)$. From (3.43) and (3.44), for all sufficiently large k it follows that

$$\begin{aligned} d_k &\leq C(d_{k-2} - d_k)^{(1-\vartheta)/\vartheta} + \frac{1}{2}(d_{k-2} - d_k) \\ &\leq C(d_{k-2} - d_k)^{(1-\vartheta)/\vartheta} + \frac{1}{2}(d_{k-2} - d_k)^{(1-\vartheta)/\vartheta} \\ &= (C + \frac{1}{2})(d_{k-2} - d_k)^{(1-\vartheta)/\vartheta}. \end{aligned}$$

Note that $\frac{\vartheta}{1-\vartheta} > 1$, then the above inequality yields that

$$d_k^{\frac{\vartheta}{1-\vartheta}} \leq \left(C + \frac{1}{2}\right)^{\frac{\vartheta}{1-\vartheta}}(d_{k-2} - d_k) = C_2(d_{k-2} - d_k) \tag{3.45}$$

for all sufficiently large k, where $C_2 := \left(C + \frac{1}{2}\right)^{\frac{\vartheta}{1-\vartheta}}$.

Consider the subsequence $\{\Delta_k := d_{2k}\}$, then inequality (3.45) yields that

$$\Delta_k^{\frac{\vartheta}{1-\vartheta}} \leq C_2(\Delta_{k-1} - \Delta_k) \tag{3.46}$$

for all sufficiently large k. From (3.46), by the proof similar to that of [1, Theorem 2], especially that from inequality (13) to the second inequality after (15) in [1], we can get that

$$\Delta_k \leq C_3 k^{-\frac{1-\vartheta}{2\vartheta-1}} = C_3 k^{-\xi} \tag{3.47}$$

for all sufficiently large k, where $C_3 > 0$ is a constant and $\xi := \frac{1-\vartheta}{2\vartheta-1} > 0$. Then for all sufficiently large k, we have that

$$\|X^k - X^*\|_F \le d_k \begin{cases} = \Delta_{\frac{k}{2}} \le 2^\xi C_3 k^{-\xi} & \text{if } k \text{ is even,} \\ \le d_{k-1} = \Delta_{\frac{k-1}{2}} \le 2^\xi C_3 (k-1)^{-\xi} \le 4^\xi C_3 k^{-\xi} & \text{if } k \text{ is odd and } k > 2, \end{cases}$$
$$\le \kappa_2 k^{-\xi}$$

where $\kappa_2 := 4^\xi C_3 > 0$. Thus, the proof is complete. □

3.4 Accelerated version of Algorithm 1: FPCAe algorithm

We adopt two technologies to accelerate Algorithm 1. The first one is the continuation technique of the parameter λ, which is similar to that in [32,34,41,42]. In detail, choose a decreasing sequence $\{\lambda_k\} : \lambda_0 > \lambda_1 > \cdots > \lambda_K = \lambda_{final} > 0$, then repeat to apply Algorithm 1 to solve (1.5) with $\lambda = \lambda_k$ until λ reaching λ_{final}. The second one is to compute the approximate SVD of the matrix G^k in each iteration instead of computing its full SVD, which is similar to that in [34,41,42]. Concretely, we adopt the Linear Time SVD algorithm developed by Drineas et al. [13], which computes the approximate SVD by a fast Monte Carlo algorithm, whose details can be referred to [13,34]. Algorithm 1 by employing these two acceleration techniques is called FPCAe algorithm.

4 Numerical experiments

In this section, we conduct numeric experiments to test the performance of our FPCAe algorithm. Concretely, we apply it to solve problem (1.5) with $\theta(X) = \frac{1}{2}\|P_\Omega(X - M)\|_2^2$, i.e.,

$$\min_{X \in \mathbb{R}^{m\times n}} f_\lambda(X) := \frac{1}{2}\|P_\Omega(X - M)\|_2^2 + \lambda\|X\|_p^p, \tag{4.1}$$

where $M \in \mathbb{R}^{m\times n}$, Ω is a subset of index pairs (i, j), and P_Ω is the orthogonal projection onto the subspace of sparse matrices with nonzero entries restricted to the index subset Ω and zero entries outside Ω. Problem (4.1) is also called matrix completion.

The parameters and initial values in FPCAe for problem (4.1) are given as follows: $X_0 = P_\Omega(M)$, $\lambda_0 = \min\{3, mn/|\Omega|\}\|P_\Omega(M)\|_2$, $\mu = 1.9$, $\eta = 0.9$, $\beta = 0.01$, $\lambda_{k+1} = \eta\lambda_k$, λ_{final}=1e-6, xtol=1e-4, maxiter = 1000 and terminate it when

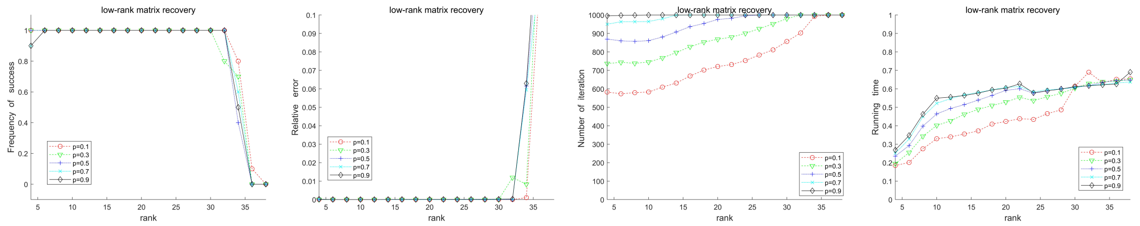
$$\frac{\| X^{k+1} - X^k \|_F}{\max\{1, \| X^k \|_F\}} < \text{xtol}.$$

All experiments are performed in MATLAB R2018a on a 64-bit PC with an Inter(R) Core(TM) i5-7500 CPU (3.40GHz) and 8GB of RAM.

4.1 Choice of p

In this subsection, we use the same way as [23,34] to generate random matrix $M \in \mathbb{R}^{m\times n}$ of rank r by the MATLAB code: `M=randn(m,r)*randn(r,n)`, then sample a subset Ω of q entries uniformly at random. We use `SR` $:= q/(mn)$ to denote the sampling ratio and `os` $:= q/[r(m + n - r)]$ to denote the oversampling rate, which is the ratio between the number of samples to the "degree of freedom" of the $m \times n$ matrix of rank r. If `os` < 1, there is always an infinite number of matrices with rank r having the given entries, so one cannot hope to recover the matrix in this situation, then there must be $\texttt{os} \geq 1$. Furthermore, the closer the os value is to 1, the harder the problem is to solve.

We take $m = n = 100$, $\texttt{os} = 2.5$, and let the rank increase from 4 to 38 by 2 each time. Through this experiment, we select the best value of p among {0.1,0.3,0.5,0.7,0.9}. The results are shown in Figure 4.1.



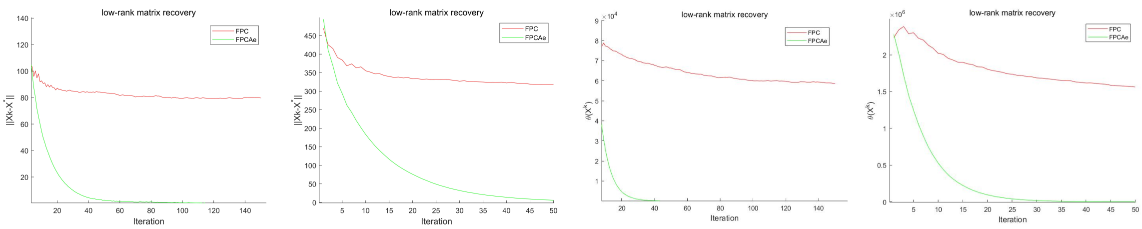
(a) (b) (c) (d)

Fig. 4.1: Results for $m = n = 100$, $\mathrm{os} = 2.5$ with the rank increasing from 4 to 38 by 2 each time.

From Figure 4.1, we can see that FPCAe with $p = 0.1$ outperforms $p = 0.3$, 0.5, 0.7, 0.9 since $p = 0.1$ has the higher success frequencies, the less time, the fewer iterations and the smaller relative errors than those of $p = 0.3, 0.5, 0.7, 0.9$. Thus, we use $p = 0.1$ in the remaining experiments.

4.2 **Comparison of FPCAe algorithm and FPC algorithm**

In this subsection, we compare FPCAe and FPC [41] (without extrapolation) to show the effect of extrapolation. The test problem is generated as follows. For each (m, n, r)=(500,500,50), (1000,1000,50), we take $\texttt{SR} = 0.20$, $\texttt{os} = 5.0$. We plot $\|X^k - X^*\|_F$ against the number of iterations of FPCAe and FPC, where X^* are the real solution of the problems. We also plot the value of loss function $\theta(X^k)$ with respect to the number of iterations of the two algorithms. The results are presented in Figure 4.2.



(a) m=500, n=500, r=50 (b) m=1000, n=1000, r=100 (c) m=500, n=500, r=50 (d) m=1000, n=1000, r=100

Fig. 4.2: Results of $\|X^k - X^*\|_F$ and $\theta(X^k)$ for FPCAe and FPC

Figure 4.2 shows that both FPCAe and FPC are convergent, while FPCAe is R-linearly convergent which is obviously faster than FPC. The results illustrate that the extrapolation technique improves the convergence rate of the algorithm.

4.3 Comparison of four algorithms for low-rank matrix recovery

In this subsection, we compare FPCAe with some state-of-the-art algorithms for low-rank matrix recovery problems such as WNNM [17], APGL [51] and tIRucLq_M [23]. We consider two classes of low-rank matrix recovery problems as below:

- square matrix recovery with different scales, ranks, sampling ratios, and noise levels.
- rectangular matrix recovery with different ratios of length to width.

In all the experiments, the parameters of above compared algorithms are taken the same as given in the original literatures. We set the maximum number of iteration `maxiter` = 1000, `xtol` = 10^{-4} and use the same stopping criteria for inner loops. We compare the behavior of WNNM, APGL, FPCAe and tIRucLq_M in the presence and absence of noise. For each scale of the problems, we run 10 instances, record the average of relative error '`Rel.err`' and the average CPU time '`Time`', and highlight the best results in bold for each problem in the tables.

4.3.1 Square matrix recovery

We first compare the four algorithms for square matrix recovery problems. The parameters of problems and numerical results for noiseless problems ($\sigma = 0$) and noise problems ($\sigma = 0.01$) are presented in Tables 4.1 and 4.2 respectively.

Table 4.1: Comparison results of four algorithms for square matrix recovery problems without noise (i.e., $\sigma = 0$)

Problems			WNNM		APGL		FPCAe		tIRucLq_M	
$m = n$	r	os	Time	Rel.err	Time	Rel.err	Time	Rel.err	Time	Rel.err
500	10	2.5	7.06	2.41e − 4	**0.88**	2.35e − 3	3.04	**3.93e − 5**	8.91	7.88e − 5
	20	2	7.24	2.78e − 4	**2.20**	1.60e − 1	3.76	**8.40e − 6**	32.21	6.52e − 5
	60	1.5	7.21	3.59e − 4	**5.36**	3.34e − 1	5.42	**1.15e − 6**	39.49	3.57e − 5
800	10	2.5	21.24	2.30e − 4	**2.03**	2.04e − 3	7.93	2.69e − 4	83.83	**2.15e − 4**
	20	2	22.55	2.72e − 4	**5.81**	2.00e − 1	8.81	**6.98e − 5**	75.85	1.11e − 4
	60	1.5	24.15	3.71e − 4	**13.27**	4.20e − 1	14.13	**2.29e − 5**	80.35	6.30e − 5
1000	10	2.5	35.00	**2.20e − 4**	**0.81**	1.79e − 3	12.92	5.70e − 4	123.25	3.34e − 4
	40	2	32.27	2.65e − 4	**14.39**	3.30e − 1	15.49	**3.57e − 6**	79.19	2.15e − 5
	110	1.5	50.96	3.53e − 4	**20.43**	3.12e − 3	29.06	**2.27e − 6**	260.19	2.50e − 5
2000	15	2.5	169.47	**2.05e − 4**	**2.25**	1.77e − 3	63.30	5.12e − 4	489.29	7.10e − 4
	40	2	230.35	2.55e − 4	**47.14**	3.07e − 1	56.75	8.50e − 5	804.32	**4.81e − 5**
	110	1.5	258.80	3.63e − 4	92.54	4.86e − 3	**80.94**	6.34e − 5	2786.36	**5.24e − 5**
3000	20	2.5	464.01	**2.00e − 4**	**3.79**	1.58e − 3	151.74	5.65e − 4	1318.98	3.70e − 4
	40	2	498.98	2.43e − 4	**29.73**	4.53e − 2	140.09	3.01e − 4	2732.00	**1.36e − 5**
	110	1.5	619.55	3.54e − 4	**170.96**	4.94e − 3	203.42	2.37e − 4	8881.97	**1.23e − 4**

For the noiseless tested problems in Table 4.1, APGL algorithm is always the fastest one, but its accuracy is the worst. Our FPCAe algorithm win the first place many times in accuracy, while its speed is in general the second fastest. What's more, when FPCAe is not the best, it tends to be very close to the best one regardless of speed or accuracy.

For the noise tested problems in Table 4.2, APGL algorithm is still the fastest but the least accurate one, while tIRucLq_M algorithm possesses the highest accuracy but almost the slowest speed. It can be seen that our FPCAe algorithm is almost as accurate as tIRucLq_M algorithm and only slightly slower than APGL algorithm, while both APGL and FPCAe algorithms are much faster than WNNM and tIRucLq_M algorithms.

Table 4.2: Comparison results of four algorithms for square matrix recovery problems with noise ($\sigma = 0.01$)

Problems			WNNM		APGL		FPCAe		tIRucLq_M	
$m = n$	r	os	Time	Rel.err	Time	Rel.err	Time	Rel.err	Time	Rel.err
500	10	2.5	20.15	5.69e − 3	**0.81**	3.64e − 3	4.98	3.09e − 3	24.15	**2.81e − 3**
	20	2	19.87	4.34e − 3	**2.16**	1.67e − 1	5.47	2.65e − 3	18.03	**2.38e − 3**
	60	1.5	21.62	2.73e − 3	**5.09**	3.31e − 1	9.04	2.01e − 3	17.82	**1.69e − 3**
800	10	2.5	53.11	5.84e − 3	**1.88**	3.51e − 3	11.78	3.15e − 3	61.42	**2.82e − 3**
	20	2	51.91	4.43e − 3	**4.91**	1.52e − 1	12.72	2.65e − 3	49.63	**2.38e − 3**
	60	1.5	55.21	2.94e − 3	**12.39**	4.16e − 1	18.46	2.06e − 3	53.11	**1.84e − 3**
1000	10	2.5	85.24	5.75e − 3	**0.79**	3.25e − 3	18.01	3.15e − 3	120.67	**2.64e − 3**
	40	2	96.27	3.06e − 3	**13.75**	3.33e − 1	25.00	1.84e − 3	41.19	**1.61e − 3**
	110	1.5	151.63	2.07e − 3	**21.44**	3.50e − 3	40.33	1.48e − 3	150.63	**1.25e − 3**
2000	15	2.5	412.25	4.59e − 3	**2.02**	2.95e − 3	70.75	2.56e − 3	521.62	**2.23e − 3**
	40	2	422.31	3.13e − 3	**34.28**	3.51e − 1	83.80	1.86e − 3	456.80	**1.67e − 3**
	110	1.5	602.93	2.22e − 3	**98.41**	5.10e − 3	122.35	1.52e − 3	1780.09	**1.38e − 3**
3000	20	2.5	976.08	3.86e − 3	**3.55**	2.50e − 3	159.23	2.21e − 3	1378.01	**1.95e − 3**
	40	2	1070.18	3.08e − 3	**38.81**	1.27e − 1	206.24	1.86e − 3	2381.94	**1.68e − 3**
	110	1.5	1438.76	2.20e − 3	**173.41**	5.19e − 3	251.60	1.53e − 3	7083.88	**1.46e − 3**

4.3.2 Rectangular matrix recovery

This subsection compares the four algorithms for rectangular matrix recovery problems. The parameters of problems and numerical results for noiseless problems ($\sigma = 0$) and noise problems ($\sigma = 0.01$) are presented in Tables 4.3 and 4.4 respectively.

Table 4.3: Comparison results of four algorithms for rectangular matrix recovery problems without noise (i.e., $\sigma = 0$)

Problems				WNNM		APGL		FPCAe		tIRucLq_M	
m	n	r	os	Time	Rel.err	Time	Rel.err	Time	Rel.err	Time	Rel.err
200	100	10	2.0	**0.29**	3.10e − 4	0.45	4.55e − 3	0.35	**7.99e − 5**	39.02	3.91e − 4
		15	1.5	**0.28**	**1.32e − 3**	0.85	7.69e − 3	0.46	1.44e − 3	37.28	2.39e − 3
400	300	15	2.0	4.18	2.84e − 4	1.01	2.34e − 3	**0.84**	**9.44e − 6**	34.12	1.31e − 4
		20	1.5	5.07	4.07e − 4	1.70	5.43e − 1	**1.30**	**3.39e − 4**	62.68	8.70e − 4
600	400	15	2.0	6.45	2.89e − 4	**1.38**	2.77e − 3	2.72	**1.14e − 4**	65.86	2.23e − 4
		20	2.0	6.72	2.78e − 4	**1.37**	2.32e − 4	2.69	**9.21e − 6**	43.12	8.37e − 5
		50	1.5	6.40	3.66e − 4	**4.27**	3.98e − 1	4.78	**4.42e − 6**	46.69	5.74e − 5
800	450	15	2.0	8.12	**2.85e − 4**	**1.67**	4.67e − 3	4.47	3.05e − 4	92.26	6.46e − 4
		20	2.0	9.46	2.82e − 4	**3.55**	2.45e − 1	3.96	**7.34e − 5**	75.25	1.14e − 4
		50	1.5	9.65	3.75e − 4	**5.93**	4.19e − 1	6.20	**2.36e − 5**	217.49	7.57e − 5
1000	800	15	2.0	28.28	**2.68e − 4**	**1.35**	2.29e − 3	11.42	5.60e − 4	169.23	6.58e − 4
		20	2.0	27.24	2.69e − 4	**3.53**	1.51e − 1	8.99	**1.57e − 4**	166.15	1.67e − 4
		50	1.5	30.56	3.73e − 4	**12.05**	4.87e − 1	12.84	**9.11e − 5**	224.45	1.10e − 4
3000	2000	50	2.0	298.28	2.53e − 4	**22.02**	2.22e − 3	80.40	8.12e − 5	2970.79	**2.43e − 5**
		100	2.0	278.38	2.57e − 4	**46.44**	1.24e − 3	87.33	**1.23e − 6**	2684.71	1.19e − 5
		150	1.5	516.04	3.71e − 4	**109.22**	3.37e − 3	148.61	**4.19e − 5**	6031.38	4.39e − 5
5000	3000	50	2.0	849.70	**2.40e − 4**	**56.27**	6.77e − 3	250.55	2.97e − 4	8158.50	2.96e − 4
		100	2.0	794.65	2.60e − 4	**123.80**	2.15e − 3	237.43	2.08e − 5	10003.90	**1.13e − 5**
		150	1.5	1375.44	3.52e − 4	**205.55**	4.04e − 3	437.79	1.84e − 4	21669.32	**1.62e − 4**

For the noiseless rectangular matrix recovery problems in Table 4.3, FPCAe algorithm outperforms tIRucLq_M and WNNM algorithms in terms of running time and accuracy, and attains higher accuracy than APGL algorithm but needs slightly more time. For the noise rectangular matrix recovery problems in Table 4.4, APGL algorithm is still the fastest one with the lowest accurate, while tIRucLq_M algorithm has the highest accuracy but the slowest speed. Obviously, our FPCAe algorithm is almost as accurate as tIRucLq_M algorithm and

Table 4.4: Comparison results of four algorithms for rectangular matrix recovery problems with noise ($\sigma = 0.01$)

Problems				WNNM		APGL		FPCAe		tIRucLq_M	
m	n	r	os	Time	Rel.err	Time	Rel.err	Time	Rel.err	Time	Rel.err
200	100	10	2.0	0.51	5.57e − 3	**0.35**	4.82e − 3	0.53	3.91e − 3	78.60	**3.65e − 3**
		15	1.5	**0.48**	5.55e − 3	0.66	8.77e − 3	0.59	**4.53e − 3**	28.66	4.84e − 3
400	300	15	2.0	11.67	5.01e − 3	**0.81**	3.81e − 3	1.57	3.07e − 3	21.75	**2.81e − 3**
		20	1.5	9.51	5.16e − 3	1.65	5.37e − 1	**1.62**	3.71e − 3	50.56	**3.47e − 3**
600	400	15	2.0	19.07	5.10e − 3	**1.34**	4.26e − 3	5.04	3.06e − 3	44.33	**2.82e − 3**
		20	2.0	18.75	4.35e − 3	**1.91**	1.24e − 1	5.52	2.64e − 3	25.83	**2.38e − 3**
		50	1.5	18.86	3.07e − 3	**3.99**	3.99e − 1	7.74	2.23e − 3	26.52	**1.97e − 3**
800	450	15	2.0	25.34	5.16e − 3	**1.66**	4.47e − 3	7.35	3.27e − 3	68.05	**2.83e − 3**
		20	2.0	26.04	4.41e − 3	**3.11**	2.42e − 1	8.20	2.65e − 3	60.79	**2.36e − 3**
		50	1.5	25.12	3.18e − 3	**5.92**	4.17e − 1	10.35	2.23e − 3	54.56	**2.00e − 3**
1000	800	15	2.0	66.76	5.16e − 3	**1.27**	4.00e − 3	15.30	3.10e − 3	139.58	**2.87e − 3**
		20	2.0	72.78	4.47e − 3	**2.85**	5.43e − 2	16.03	2.64e − 3	96.47	**2.43e − 3**
		50	1.5	67.31	3.32e − 3	**11.87**	4.88e − 1	20.56	2.27e − 3	138.83	**2.13e − 3**
3000	2000	50	2.0	691.47	2.14e − 3	**11.25**	2.22e − 3	130.14	1.64e − 3	1589.73	**1.45e − 3**
		100	2.0	948.16	1.89e − 3	**43.49**	1.60e − 3	195.33	1.15e − 3	1372.97	**9.96e − 4**
		150	1.5	1141.34	1.88e − 3	**106.04**	3.38e − 3	245.55	1.29e − 3	4319.47	**1.11e − 3**
5000	3000	50	2.0	2051.01	2.74e − 3	**19.07**	2.47e − 3	323.06	1.64e − 3	7702.01	**1.46e − 4**
		100	2.0	2417.49	1.94e − 3	**116.09**	2.42e − 3	462.58	1.15e − 3	5770.84	**1.03e − 3**
		150	1.5	2737.64	1.89e − 3	**207.16**	4.46e − 3	563.88	1.30e − 3	17683.42	**1.24e − 3**

only slightly slower than APGL algorithm, while both APGL and FPCAe algorithms are much faster than WNNM and tIRucLq_M algorithms.

Therefore, our FPCAe algorithm is competitive with the three compared algorithms in terms of speed, accuracy, robustness and anti-noise.

4.4 **Experiments on grayscale images recovery**

In this subsection, we apply FPCAe to recover the grayscale images and continue to compare it with WNNM, APGL and tIRucLq_M in different sampling ratios. We use the following four indicators: peak signal to noise ratio (PSNR) ([18]), structural similarity (SSIM) ([53]), root mean square error (RMSE) ([53]) and CPU time (Time) to evaluate the numerical performance of the compared algorithms for image recovery problems, where PSNR is defined as follows:

$$\text{PSNR} := 10 * \log_{10}\left(\frac{mn * 255^2}{\|M - X^*\|_F^2}\right). \tag{4.2}$$

The higher PSNR and SSIM values and the smaller RMSE and Time values represent better recovery performance.

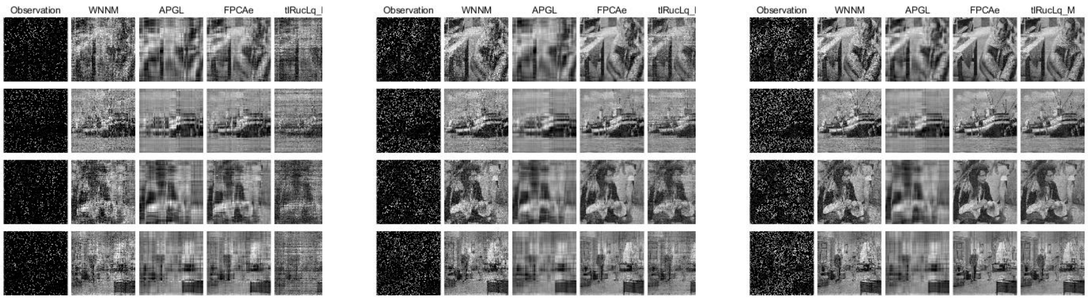
The four grayscale images in Figure 4.3 are all of 512×512, and all downloaded from the CVG-UGR image database at https://ccia.ugr.es/cvg/CG/base.htm.

Since these original images themselves do not have the low-rank structure, we randomly sample 10.0%, 20.0%, 30.0% pixels respectively of each image to reconstruct an image of rank 80 by each algorithm as low-rank approximation to the original image, that is, $\texttt{sr} = 0.1, 0.2, 0.3$ respectively and all with $r = 80$. The recovered images are presented in Figure 4.4 with $\texttt{sr} = 0.1, 0.2, 0.3$ respectively, while the numerical results of PSNR, SSIM, RMSE and Time for the recovered images are reported in Table 4.5.

From Figure 4.4, we can see that the recovered images by FPCAe algorithm are clearer and closer to the original images than those of other algorithms. Further from Table 4.5, we can see that our FPCAe algorithm always perform better than other three algorithms in

(a) Barbara (b) Boat (c) Man (d) Couple

Fig. 4.3: Four original grayscale images



(a) sr=0.1 (b) sr=0.2 (c) sr=0.3

Fig. 4.4: Recovery results for grayscale images with different sampling ratios

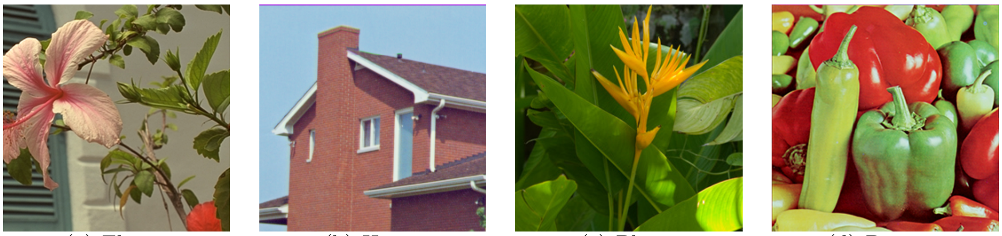
Table 4.5: Numerical results for grayscale images recovery of the four algorithms

Images	Methods	sr=0.1				sr=0.2				sr=0.3			
		PSNR	SSIM	RMSE	Time	PSNR	SSIM	RMSE	Time	PSNR	SSIM	RMSE	Time
Barbara	WNNM	14.61	0.09	47.41	8.33	17.21	0.21	35.14	16.62	19.26	0.34	27.75	27.27
	APGL	17.02	0.22	35.91	**0.83**	17.70	0.32	33.25	**0.66**	19.76	0.40	26.20	**0.78**
	FPCAe	**18.05**	**0.24**	**31.90**	6.07	**20.70**	**0.41**	**23.51**	7.12	**23.23**	**0.58**	**17.56**	9.28
	tIRucLq_M	14.96	0.11	45.55	2.41	18.52	0.26	30.23	11.68	21.95	0.47	20.36	69.53
Boat	WNNM	16.09	0.13	39.99	10.37	19.36	0.27	27.41	17.16	21.62	0.39	21.15	29.30
	APGL	18.55	0.31	30.12	**0.72**	19.60	0.40	26.71	**0.64**	19.82	0.43	26.00	**0.74**
	FPCAe	**19.59**	**0.32**	**26.70**	5.13	**22.54**	**0.48**	**19.02**	7.59	**25.33**	**0.62**	**13.79**	10.45
	tIRucLq_M	16.44	0.15	38.37	2.20	20.18	0.32	24.97	10.12	23.82	0.53	16.42	59.54
Man	WNNM	15.91	0.11	40.81	9.98	19.22	0.23	27.86	16.83	21.52	0.35	21.40	26.56
	APGL	18.30	0.30	30.99	**0.62**	18.97	0.35	28.68	**0.54**	19.15	0.40	28.11	**0.57**
	FPCAe	**19.45**	**0.32**	**27.16**	5.42	**22.51**	**0.42**	**19.08**	7.30	**25.51**	**0.60**	**13.51**	9.20
	tIRucLq_M	15.91	0.15	40.80	2.29	19.96	0.28	25.59	10.89	23.75	0.48	16.54	49.67
Couple	WNNM	16.86	0.17	36.56	10.11	19.65	0.32	26.53	16.34	22.22	0.47	19.73	27.26
	APGL	19.33	0.33	27.52	**0.60**	20.07	0.42	25.28	**0.54**	20.24	0.46	24.79	**0.64**
	FPCAe	**20.05**	**0.35**	**25.35**	5.56	**22.89**	**0.52**	**18.28**	7.20	**25.86**	**0.68**	**12.97**	9.25
	tIRucLq_M	16.71	0.16	37.20	2.31	20.42	0.35	24.27	8.05	24.13	0.58	15.84	63.13

all the cases. Although APGL algorithm is always the fastest one, but its PSNR, SSIM and RMSE are not very good, which makes the quality of its recovered images poor.

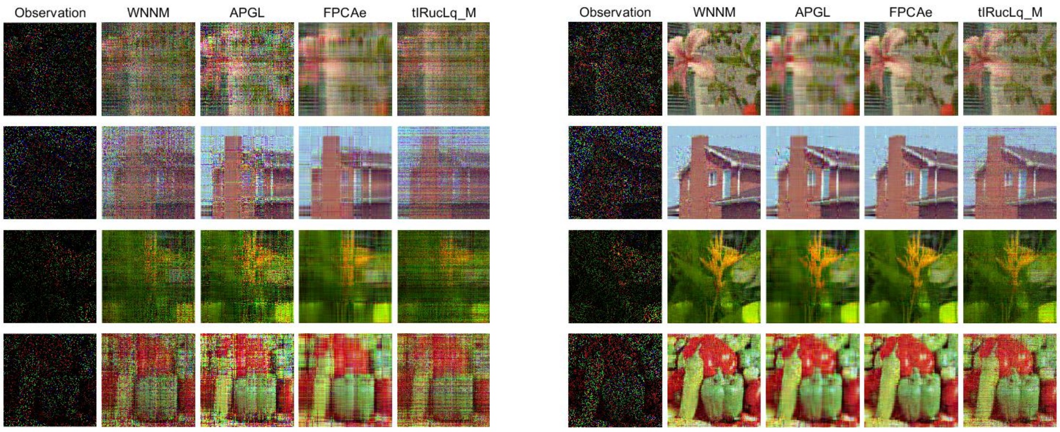
4.5 **Experiments on three-channel images recovery**

In this subsection, we select four popular three-channel images (Figure 4.5) with dimension $256*256*3$ for this experiment, whose entries denote the pixels of the corresponding images. We also randomly sample 10%, 20% pixels respectively of each image to reconstruct an image of rank 30 by each algorithm as low-rank approximations to the original image, that is, $\texttt{sr}$=0.1, 0.2 respectively and all with $r = 30$. The recovered images are presented in Figure 4.6 with $\texttt{sr} = 0.1, 0.2$ respectively, while the numerical results of PSNR, SSIM, RMSE and Time for the recovered images are reported in Table 4.6.



(a) Flower (b) House (c) Plants (d) Peppers

Fig. 4.5: Four original three-channel images



(a) sr=0.1 (b) sr=0.2

Fig. 4.6: Recovery results for three-channel images with different sampling ratios

From Figure 4.6, we can see that when $\texttt{sr} = 0.1$, all the compared algorithms fail to recover the clear images, while the quality of the recovery images of all the compared algorithms becomes better as the sampling ratios becomes larger. When $\texttt{sr} = 0.2$, the recovered images by FPCAe algorithm are clearer and closer to the original images than those of other algorithms. Further from Table 4.6, we can see that FPCAe algorithm always recover the target images stably with higher PSNR and SSIM values and less CPU time compared to WNNM and tIRucLq_M algorithms. Although the speed of APGL algorithm and the RMSE value of tIRucLq_M algorithm are smaller than other algorithms, their other aspects of numerical results are not as good as FPCAe algorithm.

Table 4.6: Numerical results for three-channel images recovery of the four algorithms

Images	Methods	sr=0.1				sr=0.2			
		PSNR	SSIM	RMSE	Time	PSNR	SSIM	RMSE	Time
Flower	WNNM	13.24	0.08	0.03	2.83	19.87	0.34	0.01	4.96
	APGL	11.48	0.06	0.13	1.73	19.49	0.34	0.01	**0.49**
	FPCAe	**17.83**	**0.23**	0.005	**0.74**	**20.96**	**0.41**	**0.001**	1.24
	tIRucLq_M	15.20	0.12	**0.004**	0.98	19.11	0.29	**0.001**	10.63
House	WNNM	12.09	0.07	0.02	2.68	22.05	0.47	0.009	5.39
	APGL	13.83	0.15	0.07	2.26	21.10	0.46	0.01	**0.28**
	FPCAe	**19.32**	**0.34**	**0.002**	**0.81**	**22.88**	**0.52**	**0.001**	1.18
	tIRucLq_M	15.41	0.11	**0.002**	1.08	20.09	0.30	**0.001**	10.40
Plants	WNNM	15.64	0.21	0.08	3.98	22.37	0.45	0.02	5.66
	APGL	14.47	0.19	0.14	2.13	19.31	0.43	0.03	1.21
	FPCAe	**19.91**	**0.38**	**0.01**	**0.81**	**23.40**	**0.54**	0.008	**1.14**
	tIRucLq_M	17.07	0.26	0.02	1.29	20.94	0.41	**0.006**	30.34
Peppers	WNNM	11.66	0.07	0.13	2.43	19.10	0.35	0.07	5.19
	APGL	10.21	0.07	0.20	2.49	18.58	0.36	0.06	**0.55**
	FPCAe	**16.00**	**0.21**	**0.03**	**0.75**	**20.01**	**0.42**	0.05	1.20
	tIRucLq_M	13.45	0.10	**0.03**	0.99	17.55	0.26	**0.04**	15.80

5 Conclusion

In this paper, we studied a general Schatten p-quasi-norm ($0 < p < 1$) regularized matrix optimization problem which is modeled by (1.5). For this nonconvex nonsmooth and non-Lipschitz optimization problem, based on the matrix p-thresholding operator, we propose a fixed point continuation algorithm with extrapolation (FPCAe) for solving it. We prove that any accumulation point of the iterative sequence generated by the proposed algorithm is not only a critical point but also a global stationary point of the problem, where the global stationary point possesses some global optimality which can exclude too many stationary points even some local minimizers of the nonconvex problem. We also prove the rank invariance of the iterative sequence. Further, we established global convergence and R-linear convergence rate of the whole iterative sequence generated by the proposed algorithm under some mild conditions. Finally, a large number of numerical experiments on random square and rectangular matrix completion problems, grayscale image and three-channel image recovery problems demonstrated that the proposed FPCAe algorithm possesses very excellent performance and so is a very competitive algorithm for low-rank matrix optimization problems in comparison with some state-of-the-art algorithms.

Acknowledgements This work is supported by the National Natural Science Foundation of China (12261020), the Guizhou Provincial Science and Technology Program (ZK[2021]009), the Foundation for Selected Excellent Project of Guizhou Province for High-level Talents Back from Overseas ([2018]03), and the Research Foundation for Postgraduates of Guizhou Province (YJSCXJH[2020]085).

References

1. Attouch, H., Bolte, J.: On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. Math. Program., **116**(1-2), 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-Lojasiewicz inequality. Math. Oper. Res., **35**(2), 438-457 (2010)
3. Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C.: Image Inpainting. ACM SIGGRAPH, 417-424 (2000)
4. Bolte, J., Sabach, S., Teboublle, M.: Proximal alternating linearized minimization for nonconvex and nonsmooth problems. Math. Comput., **146**(1-2), 459-494 (2014)

5. Cai, J., Candès, E., Shen, Z.: A singular value thresholding algorithm for matrix completion. SIAM J. Optim., **20**(4), 1956-1982 (2010)
6. Candès, E., Plan, Y.: Matrix completion with noise. Proceedings of the IEEE, **98**(6), 925-936 (2010)
7. Candès, E., Plan, Y.: Tight Oracle bounds for low-rank matrix recovery from a minimal number of noisy random measurements. IEEE Trans. Inform. Theory, **57**(4), 2342-2359 (2011)
8. Candès, E., Recht, B.: Exact matrix completion via convex optimization. Found. Comput. Math., **9**(6), 717-772 (2009)
9. Candès, E., Tao, T.: The power of convex relaxation: Near-optimal matrix completion. IEEE Trans. Inform. Theory, **56**(5), 2053-2080 (2010)
10. Chen, C., He, B., Yuan, X.: Matrix completion via alternating direction methods. IMA J. Numer. Anal., **32**(32), 227-245 (2012)
11. Chen, X., Ge, D., Wang, Z., Ye, Y.: Complexity of unconstrained l_2-l_p minimization. Math. Program., **143**(1-2), 371-383 (2014)
12. Chen, Y., Xiu, N., Peng, D.: Global solutions of non-Lipschitz S_2-S_p minimization over the positive semidefinite cone. Optim. Lett., **8**, 2053-2064 (2014)
13. Drineas, P., Kannan, R., Mahoney, M.: Fast Monte Carlo algorithms for matrices II: Computing low-rank approximations to a matrix. SIAM J. Comput. **36(1)**, 158-183 (2006)
14. Fazel, M., Hindi, H., Boyd, S.: Rank minimization and applicationsin system theory. Proceedings of the 2004 American Control Conference, **4**, 3273-3278 (2004)
15. Fazel, M.: Matrix rank minimization with applications. Ph.D. thesis, Stanford University, 2002.
16. Fornasier, M., Rauhut, H., Ward, R.: Low-rank matrix recovery via iteratively reweighted least squares minimization. SIAM J. Optim., **21**, 1614-1640 (2011)
17. 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)
18. He, L., Wang, Y., Xiang, Z.: Support driven wavelet frame-based image deblurring. Inf. Sci., **479**, 250-269 (2019)
19. Hoffman, A., Wielandt, H.: The variation of the spectrum of a normal matrix. Duke Math. J., **20**(1), 37-39 (1953)
20. Horn, R.A., Johnson, C.R.: Matrix Analysis, 2nd edn. Cambridge University Press, Cambridge (1985)
21. Korah, T., Rasmussen, C.: Spatio-temporal inpainting for recovering texture maps of occluded building facades. IEEE Trans. Image Process., **16**(7), 2262-2271 (2007)
22. Lai, M.-J., Liu, Y., Li, S., Wang, L.: On the Schatten p-quasi-norm minimization for low-rank matrix recovery, Appl. Comput. Harmon. A., **51**(2), 157-170 (2021)
23. Lai, M.-J., Xu, Y., Yin, W.: Improved iteratively reweighted least squares for unconstrained smoothed l_q minimization. SIAM J. Numer. Anal, **51**(2), 927-957 (2013)
24. Lewis, A.S., Sendov, H.S.: Nonsmooth analysis of singular values. Part I: theory. Set-Valued Anal., **13**(3), 213-241 (2005)
25. 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)
26. Li, X., Luo, Z.: Normal cones intersection rule and optimality analysis for low-rank matrix optimization with affine manifolds. SIAM J. Optim., **33**(3), 1333-1360 (2023)
27. Li, X., Xiu, N., Zhou, S.: Matrix optimization over low-rank spectral sets: stationary points and local and global minimizers. J. Optim. Theory Appl., **184**(3), 895-930 (2020)
28. Li, Y., Shang, K., Huang, Z.: A singular value p-shrinkage thresholding algorithm for low rank matrix recovery. Comput. Optim. Appl. **73**, 453-476 (2019)
29. Liu, Y., Sun, D., Toh, K.: An implementable proximal point algorithmic framework for nuclear norm minimization. Math. Program., **133**(1-2), 399-436 (2012)
30. Liu, Z., Vandenberghe, L.: Interior-point method for nuclear norm approximation with application to system identification. SIAM J. Matrix Anal. Appl., **31**(3), 1235-1256 (2009)
31. Lu, Y., Zhang, L., Wu, J.: A smoothing majorization method for l_2-l_p matrix minimization. Optim. Methods Softw., **30**(4), 682-705 (2015)
32. Lu, Z., Zhang, Y., Lu, J.: ℓ_p regularized low-rank approximation via iterative reweighted singular value minimization. Comput. Optim. Appl., **68**, 619-642 (2017)
33. Marjanovic, G., Solo, V.: On l_p optimization and matrix completion. IEEE Trans. Signal Process, **60**(11), 5714-5724 (2012)
34. Ma, S., Goldfarb, D., Chen, L.: Fixed point and Bregman iterative methods for matrix rank minimization. Math. Program., **128**(1-2), 321-353 (2011)

35. Mohan, K., Fazel, M.: Reweighted nuclear norm minimization with application to system identification. American Control Conference, 2953-2959 (2010)
36. Nesterov, Y.: A method of solving a convex programming problem with convergence rate O($\frac{1}{k^2}$). Sov. Math. Dokl., **27**, 372-376 (1983)
37. Nesterov, Y.: Gradient methods for minimizing composite functions. Math. Program., **140**(1), 125-161 (2013)
38. Nie, F., Huang, H., Ding, C.: Low-rank matrix recovery via efficient Schatten p-norm minimization. Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, **1**(1), 655-661 (2012)
39. Oymak, S., Hassibi, B.: New null space results and recovery thresholds for matrix rank minimization, Eprint Arxiv, **58**(4), 766-773 (2010)
40. Oymak, S., Mohan, K., Fazel, M., Hassibi, B.: A simplified approach to recovery conditions for low rank matrices. Proceedings of IEEE International Symposium on Information Theory Proceedings, 2318-2322 (2011)
41. Peng, D., Xiu, N., Yu, J.: Global optimality conditions and fixed point continuation algorithm for non-Lipschitz ℓ_p regularized matrix minimization. Sci. China Math., **61**, 1139-1152 (2018)
42. Peng, D., Xiu, N., Yu, J.: $S_{1/2}$ regularization methods and fixed point algorithms for affine rank minimization problems. Comput. Optim. Appl., **67**, 543-569 (2017)
43. Polyak, B.T.: Some methods of speeding up the convergence of iteration methods. USSR Comput. Math. Phys., **4**, 1-17 (1964)
44. Rakotomamonjy, A., Flamary, R., Gasso, G.: ℓ_p-ℓ_q penalty for sparse linear and sparse multiple kernel multitask learning. IEEE Trans. Neural Network, **22**(8), 1307-1320 (2011)
45. 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)
46. Recht, B., Xu, W., Hassibi, B.: Necessary and sufficient conditions for success of the nuclear norm heuristic for rank minimization. 2008 47th IEEE Conference on Decision and Control, 3065-3070 (2008)
47. Recht, B., Xu, W., Hassibi, B.: Null space conditions and thresholds for rank minimization. Math. Program., **127**(1), 175-202 (2011)
48. Rockafellar, R., Wets, R.-J.: Variational Analysis. Springer Science & Business Media (2009)
49. Skelton, R., Iwasaki, T., Grigoriadis, K.: A unified algebraic approach to linear control design. Abingdon: Taylor and Francis (1998)
50. Sleem, O.M., Ashour, M.E., Aybat, N.S., Lagoa,C.: Lp quasi-norm minimization: algorithm and applications (2023). https://arxiv.org/abs/2301.12027v1.
51. Toh, K., Yun, S.: An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems. Pacific J. Optim., **6**(3), 615-640 (2010)
52. Vandenberghe, L., Boyd, S.: Semidefinite programming. SIAM Rev., **38**(1), 49-95 (1996)
53. Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process., **13**(4), 600-612 (2004)
54. Wen, B., Chen, X., Pong, T.: A proximal difference-of-convex algorithm with extrapolation. Comput. Optim. Appl., **69**(2), 297-324 (2018)
55. Wen, Y.-W., Li, K., Chen, H.: Accelerated matrix completion algorithm using continuation strategy and randomized SVD. J. Comput. Appl. Math. (2023). https://doi.org/10.1016/j.cam.2023.115215.
56. Yang, J., Yuan, X.: Linearized augmented Lagrangian and alternating direction methods for nuclear norm minimization. Math. Comput., **82**, 301-329 (2013)
57. Yang, L., Pong, T., Chen, X.: Alternating direction method of multipliers for a class of nonconvex and nonsmooth problems with applications to background/foreground extraction. SIAM J. Imaging Sci., **10**(1), 74-110 (2017)
58. Yang, Z., Han, L.: A global exact penalty for rank constrained optimization problem and applications. Comput. Optim. Appl., **84**, 477-508 (2023)
59. Yu, P., Li, G., Pong, T.K.: Kurdyka-Łojasiewicz exponent via INF-projection. Found. Comput. Math., **22**, 1-47 (2021)
60. Yu, Q., Zhang, X.: A smoothing proximal gradient algorithm for matrix rank minimization problem. Comput. Optim. Appl. **81**, 519-538 (2022)
61. Zeng, C.: Proximal linearization methods for Schatten p-quasi-norm minimization. Numer. Math., **153**, 213-248 (2023)
62. Zhang, X., Peng, D., Su, Y.: A singular value shrinkage thresholding algorithm for folded concave penalized low-rank matrix optimization problems. J. Global Optim. (2023). https://doi.org/10.1007/s10898-023-01322-8.
63. Zhou, S., Xiu, X., Wang, Y., Peng, D.: Revisiting Lq ($0 \leq q < 1$) norm regularized optimization (2023). https://doi.org/10.48550/arXiv.2306.14394.