J Sci Comput
DOI 10.1007/s10915-015-0045-0

An Alternating Direction Method with Continuation for Nonconvex Low Rank Minimization

Zheng-Fen Jin[1] · **Zhongping Wan**[1] · **Yuling Jiao**[2] · **Xiliang Lu**[1,3]

Received: 10 November 2014 / Revised: 1 April 2015 / Accepted: 13 May 2015

Abstract In this paper we consider a nonconvex model of recovering low-rank matrices from the noisy measurement. The problem is formulated as a nonconvex regularized least square optimization problem, in which the rank function is replaced by a matrix minimax concave penalty function. An alternating direction method with a continuation (ADMc) technique (on the regularization parameter) is proposed to solve this nonconvex low rank matrix recovery problem. Moreover, under some mild assumptions, the convergence behavior of the alternating direction method for the proposed nonconvex problems is proved. Finally, comprehensive numerical experiments show that the proposed nonconvex model and the ADM algorithm are competitive with the state-of-the-art models and algorithms in terms of efficiency and accuracy.

Keywords Low rank minimization · Alternating direction method · Continuation · Minimax concave penalty function

✉ Xiliang Lu
xllv.math@whu.edu.cn

Zheng-Fen Jin
jinzhengfen@whu.edu.cn

Zhongping Wan
mathwanzhp@whu.edu.cn

Yuling Jiao
yulingjiaomath@whu.edu.cn

1 School of Mathematics and Statistics, Wuhan University, Wuhan 430072, People's Republic of China

2 School of Statistics and Mathematics, Zhongnan University of Economics and Law, Wuhan 430063, People's Republic of China

3 Computational Science Hubei Key Laboratory, Wuhan University, Wuhan 430072, People's Republic of China

Published online: 21 May 2015

1 Introduction

The matrix completion (MC) problem is to recover an unknown matrix from a small amount of observations. If the desired matrix has a low rank structure, this approach is possible. The mathematical formula reads:

$$\min_{X\in\mathbb{R}^{m\times n}} \operatorname{rank}(X), \quad \text{s.t.} \sum_{(i,j)\in\Omega} |X_{i,j} - M_{i,j}|^2 \le \delta, \tag{1}$$

where Ω is a given set of the index pairs (i, j), and $\delta > 0$ admits the possible noise in the measurement. The MC problem has a lot of applications in online recommendation system and collaborative filtering [1], such as the Joster joke data [2], DNA data [3]and the famous Netflex problem [4]. A general form of the MC problem is the affine rank minimization problem, which can be expressed as

$$\min_{X\in\mathbb{R}^{m\times n}} \operatorname{rank}(X), \quad \text{s.t. } \|\mathcal{A}(X) - b\|_2 \le \delta, \tag{2}$$

where $X \in \mathbb{R}^{m\times n}$ is an unknown low rank matrix, $\mathcal{A}$:$\mathbb{R}^{m\times n} \to \mathbb{R}^p$ is a linear map and $b \in \mathbb{R}^p$ is a given measurement vector. This problem is also widely applied in system identification [5], optimal control [6] and face recognition [7].

Due to the combinational nature of the function "rank(·)" , these rank minimization problems (1) and (2) are NP-hard problems in general [8]. One popular approach is utilizing the nuclear norm as convex relaxation [9–11]. The nuclear norm of X defined as the sum of its singular values, i.e., $\|X\|_* = \sum_{i=1}^{r} \sigma_i(X)$, where we assume the matrix X has r positive singular values of $\sigma_1 \ge \sigma_2 \ge \ldots \ge \sigma_r > 0$, is the best convex approximation of the rank function over the unit ball of matrices with norm less than one [8]. Therefore, the problem (2) can be transformed into the following nuclear norm minimization problem:

$$\min_{X\in\mathbb{R}^{m\times n}} \|X\|_*, \quad \text{s.t. } \|\mathcal{A}(X) - b\|_2 \le \delta, \tag{3}$$

or its equivalent Lagrangian form:

$$\min_{X\in\mathbb{R}^{m\times n}} \|X\|_* + \frac{\gamma}{2}\|\mathcal{A}(X) - b\|_2^2, \tag{4}$$

where $\delta \ge 0$ is the noise level and γ is the regularization parameter which balances the fidelity term and the rank of the solution. The nuclear norm minimization problems are convex optimization problems, which admit many efficient algorithms such as SDPT3 [12], singular value thresholding (SVT) [13], fixed point continuation with approximate SVD (FPCA) method [8], proximal point algorithm [14], accelerated proximal gradient(APG) algorithm [15] and ADM type algorithms [16,17], just to name a few.

However, there are some strict conditions needed for recovering successfully the low rank matrix via the nuclear norm [9,18]. Moreover, the nuclear norm minimization problem may yield the matrix with much higher rank than the real one, and can not recover a low rank target with minimum measurements [9]. Another limitation of the nuclear norm minimization is its bias caused by shrinking all the singular values toward zero simultaneously, thus the rank function may not be approximated well [19].

Therefore, more efficient models are needed for solving the low rank matrix recovery problems. Due to the recent development of nonconvex penalties in sparsity modeling, many researchers have shown that using the nonconvex term to approximate the ℓ_0 norm is better than adopting the ℓ_1 norm, where the nonconvex penalties may be ℓ_p, $p \in (0, 1)$ [20], capped-ℓ_1 [21], smoothly clipped absolute deviation (SCAD) [22] and a minimax concave penalty (MCP) [23]. These methods have been extended to the low rand matrix restoration recently. For example, the ℓ_p penalty form is studied by many researchers (see e.g. [24–26]) and the matrix MCP penalty is proposed in [27] for the robust principle component analysis. The work of Fan and Li [22] showed that the nonconvex sparse recovery models enjoy better properties than the convex one and the sparse vector MCP model also performs well [23]. Therefore, we are interested in studying the matrix MCP penalty on low rank minimization problem in this paper. Firstly, the sparse scalar MCP function is defined by

$$\rho_{\lambda,\tau}(t) = \lambda \int_0^{|t|} \left(1 - \frac{x}{\lambda\tau}\right)_+ dx = \begin{cases} \lambda|t| - \frac{t^2}{2\tau}, & \text{if } |t| \leq \lambda\tau, \\ \frac{1}{2}\tau\lambda^2, & \text{if } |t| > \lambda\tau, \end{cases}$$

for $\tau \in (1, \infty)$. Then similar as the nuclear norm and the rank of matrices [18], the MCP function of a given matrix X is defined by

$$\|X\|_{\lambda,\tau} = \sum_{i=1}^{r} \rho_{\lambda,\tau}\left(\sigma_i(X)\right) = \sum_{i=1}^{r} \lambda \int_0^{\sigma_i(X)} \left(1 - \frac{x}{\lambda\tau}\right)_+ dx, \tag{5}$$

where $\sigma_i(X)$ is the singular values of X and $\rho_{\lambda,\tau}(\cdot)$ is a scalar MCP function defined above.

One can observe that (see Sect. 2.1 for detail) the thresholding function of the scalar MCP converges to that of ℓ_1 function and ℓ_0 function, as $\tau \to \infty$ and $\tau \to 1^+$, respectively. Moreover, it has been shown in [23] that the MCP function satisfies: unbiasedness, sparsity, and continuity at the origin, which implies the MCP to be a good sparsity promoted penalty function [22]. Some efficient algorithms for the MCP sparse optimization can be found in [21,28]. Due to the good properties of the scalar MCP function, the $\|X\|_{\lambda,\tau}$ has been used in robust principal component analysis problem [27]. Some other properties of the matrix MCP function are also obtained in [27]:

(1) $\|X\|_{\lambda,\tau} \geq 0$, with equality holds if and only if $X = \mathbf{0}$;
(2) $\|X\|_{\lambda,\tau}$ is an increasing function of τ, $\|X\|_{\lambda,\tau} \leq \|X\|_*$, and $\lim_{\tau\to\infty} \|X\|_{\lambda,\tau} = \|X\|_*$;
(3) $\|X\|_{\lambda,\tau}$ is unitarily invariant; that is, $\|UXV\|_{\lambda,\tau} = \|X\|_{\lambda,\tau}$ whenever $U_{m\times m}$ and $V_{n\times n}$ are orthogonal matrix.

We will consider the following matrix MCP regularized least square (MCP-RLS) problem:

$$\min_{X\in\mathbb{R}^{m\times n}} \|X\|_{\lambda,\tau} + \frac{1}{2}\|\mathcal{A}(X) - b\|_2^2. \tag{6}$$

If we choose linear operator $\mathcal{A}$ as a componentwise projection, it becomes the matrix completion problem:

$$\min_{X\in\mathbb{R}^{m\times n}} \|X\|_{\lambda,\tau} + \frac{1}{2} \sum_{(i,j)\in\Omega} |X_{i,j} - M_{i,j}|^2, \tag{7}$$

where Ω is the given set of index pairs, λ is the regularization parameter.

For any given regularization parameter λ, the MCP-RLS problem is highly nonlinear, nonsmooth and nonconvex, it is not trivial to find an efficient algorithm. In this work we apply an alternating direction method (ADM) for solving it. The ADM is very efficient method for solving a lot of sparse promoted optimization problems and has widely application in signal and image processing, machine learning, statistics and matrix completion problems, see [29–31] and the references cited there. The matrix MCP function also admits a separable property, which can be efficiently solved by ADM.

In addition, the regularization parameter λ balances the low rank property of the solution and the fidelity of the measurement, it plays an important role in getting a satisfactory reconstruction. To obtain a good regularization parameter, we couple a continuation technique with ADM, i.e., given a decreasing sequence of parameter $\{\lambda_s\}_s$, and apply the ADM to solve the λ_{s+1}-problem with the initial guess from the λ_s-problem. Moreover, the regularization parameter λ can be chosen automatically without much additional cost when it is equipped with a proper stop rule. In this paper, we adopt a discrepancy principle (DP) or Bayesian information criterion (BIC) as the parameter selection rule to select a suitable $\widehat{\lambda}$ and solution $X(\widehat{\lambda})$ during the continuation process, see similar approach as in [32,33].

The main contribution of this paper is to propose the matrix MCP penalty for low rank matrix restoration and to extent the ADM (ADMc) for solving the nonconvex matrix low rank recovery problems and give certain theoretical analysis. To the best of our knowledge, this is the first paper for the MCP-RLS model. The ADM is very efficient and popular for the convex nuclear norm minimization problems, but the work on the ADM for solving the nonconvex rank minimization is very limited. Moreover, the ADM combining with the continuation technique on the regularization parameter is easy to implement and efficient for the nonconvex matrix rank minimization problems.

The rest of this paper is organized as follows. In Sect. 2, we firstly give some preliminaries on the different penalty functions and their corresponding thresholding operators, which are basis for developing the proposed ADM in this paper. Then we construct the ADM for solving the proposed nonconvex problem and give the convergence result of the proposed method. In Sect. 3, some numerical comparison with state-of-the-art models and algorithms on both simulated and real data are reported. Finally, we conclude the paper with some remarks in Sect. 4.

2 Algorithm and Convergence Analysis

In this section, we mainly construct our ADM to solve the problem (6). We begin with reviewing some preliminaries on the sparse promoted scalar functions including ℓ_1, ℓ_0, MCP to illustrate the advantage of MCP function over the other two. Subsequently, we show that the resulting subproblem has closed-form solution. Finally, we give the proposed ADM and its continuation version ADMc for solving (6) and show its convergence.

2.1 Preliminaries

Firstly, we consider the sparse promoted scalar functions $\rho_{\lambda,\tau} = \ell_1, \ell_0$, MCP and their thresholding operators defined as

$$S^{\rho}_{\lambda,\tau}(v) = \operatorname*{argmin}_{u\in\mathbb{R}} \left\{(u-v)^2/2 + \rho_{\lambda,\tau}(u)\right\}.$$

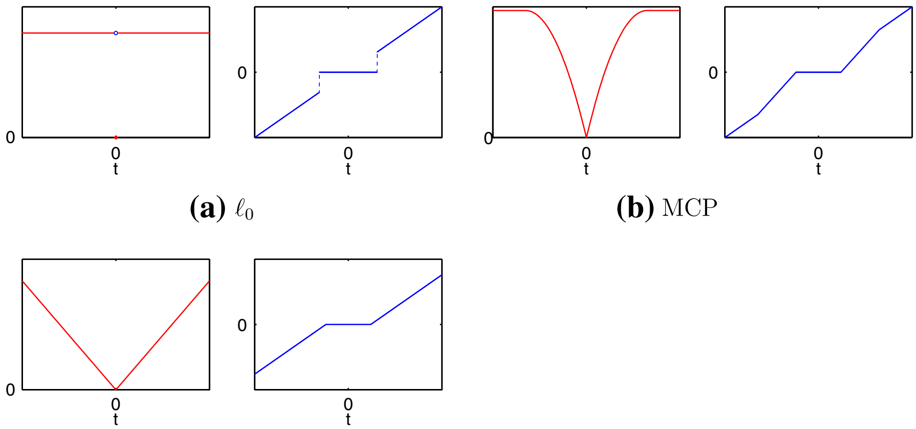
The three penalties and their thresholding operators are shown in Table 1 and Fig. 1.

Form Table 1 and Fig. 1, one can observe that among the three functions, MCP function is the only one that is continuous, sparsity promoting and unbiasedness. And these three properties together guarantee the MCP regularized problem enjoys nice statistical property [22,23]. Intuitively, the advantage of the scalar MCP function can be inherited to matrix MCP case, which is the key motivation to consider using MCP-RLS model to recover low rank matrices

Next, we review a result that was established by Lu and Zhang [37] which is helpful to get closed form solutions of the resulting subproblem.

Table 1 The penalty functions $\rho_{\lambda,\tau}(t)$ and their thresholding operators $S^{\rho}_{\lambda,\tau}(v)$

Sparse vector case		
Penalty	$\rho_{\lambda,\tau}(t)$	$S^{\rho}_{\lambda,\tau}(v)$
Lasso [34,35]	$\lambda\|t\|$	$sgn(v)\max(\|v\|-\lambda,0)$
ℓ^0 [36]	$\begin{cases} \lambda & t \neq 0 \\ 0 & t = 0 \end{cases}$	$\begin{cases} 0 & \|v\| < \sqrt{2\lambda} \\ \{0, sgn(v)\sqrt{2\lambda}\} & \|v\| = \sqrt{2\lambda} \\ v & \|v\| > \sqrt{2\lambda} \end{cases}$
MCP [23]	$\begin{cases} \lambda\left(\|t\| - \frac{t^2}{2\lambda\tau}\right) & \|t\| < \tau\lambda \\ \frac{\lambda^2\tau}{2} & \|t\| \geq \tau\lambda \end{cases}$	$\begin{cases} 0 & \|v\| \leq \lambda \\ sgn(v)\frac{\tau(\|v\|-\lambda)}{\tau-1} & \lambda \leq \|v\| \leq \lambda\tau \\ v & \|v\| \geq \lambda\tau \end{cases}$



(a) ℓ_0 **(b)** MCP **(c)** ℓ_1

Fig. 1 Functions (*left panel*) and their thresholding operators (*right panel*). Here $\lambda = 1.2$, and $\tau = 2.7$ for the MCP function

Proposition 2.1 *Let $\|\cdot\|$ be a unitarily invariant norm on $\mathbb{R}^{m\times n}$, and let $F : \mathbb{R}^{m\times n} \to \mathbb{R}$ be a unitarily invariant function. Suppose that $\mathcal{X} \subseteq \mathbb{R}^{m\times n}$ is a unitarily invariant set. Let matrix $A \in \mathbb{R}^{m\times n}$ be given, $q = min(m,n)$, and let ϕ be a non-decreasing function on* $[0,\infty)$. *Suppose that $U\Sigma(A)V^{\top}$ is the singular value decomposition(SVD) of A, then $X^* = U\mathcal{D}(x^*)V^{\top}$ is an optimal solution of the problem*

$$\min F(X) + \phi(\|X - A\|) \quad s.t.\ X \in \mathcal{X}, \tag{8}$$

where $x^* \in \mathbb{R}^q$ is an optimal solution of the problem

$$\min F(\mathcal{D}(x)) + \phi(\|\mathcal{D}(x) - \Sigma(A)\|) \quad s.t.\ \mathcal{D}(x) \in \mathcal{X}. \tag{9}$$

where $\mathcal{D}(x)$ denotes a $m \times n$ matrix with $\mathcal{D}_{ij}(x) = x_i$ if $i = j$, and $\mathcal{D}_{ij}(x) = 0$; $\Sigma_{ii}(A) = \sigma_i(A)$ for $1 \leq i \leq q$ and $\Sigma_{i,j}(A) = 0$ for all $i \neq j$.

The proof of this proposition is given in [37]. As a consequence of Proposition 2.1, we can show that the following problem has a closed-form solution. It is a key step for solving the one subproblem of the ADM.

Corollary 2.1 *Let $\mu > 1/\tau$, $A \in \mathbb{R}^{m\times n}$, and $q = min(m, n)$. Suppose that $U\Sigma V^T$ is the SVD of A. Then $X^* = \mathcal{S}_{\lambda,\tau,\mu}(A) := U\mathcal{D}(x^*)V^T$ is an optimal solution of the problem*

$$\min \|X\|_{\lambda,\tau} + \frac{\mu}{2}\|X - A\|_F^2, \tag{10}$$

where

$$x_i^* = \begin{cases} \max\left(\Sigma_{ii} - \frac{\lambda}{\mu}, 0\right) \cdot \frac{\tau\mu}{\tau\mu - 1}, & \text{if} \quad \Sigma_{ii} \le \tau\cdot\lambda \\ \Sigma_{ii}, & \text{if} \quad \Sigma_{ii} > \tau\cdot\lambda. \end{cases} \tag{11}$$

Proof Let $F(X) = \|X\|_{\lambda,\tau}$, $\phi(t) = \frac{\mu t^2}{2}$ and $\|\cdot\| = \|\cdot\|_F$. Using the Proposition 2.1, we can obtain that the $X^* = U\mathcal{D}(x^*)V^T$ is an optimal solution of (10) with $x^* \in \mathbb{R}^q$ be the optimal solution of the problem

$$\min \sum_{i=1}^{n} \rho_{\lambda,\tau}(x_i) + \frac{\mu}{2}\left(x_i - \Sigma_{ii}\right)^2.$$

Observing the above optimization problem is separable and the corresponding one dimensional optimization is strictly convex under the assumption $\mu > 1/\tau$. We can verify that (11) holds, detail can be found in [28].

2.2 Alternating Direction Method for MCP-RLS Problem

In this subsection, we propose an alternating direction method for solving the MCP-RLS problem. Firstly, by introducing an auxiliary variable Y, MCP-RLS problem can be equivalently transformed into

$$\begin{aligned} &\min_{X,Y} \|X\|_{\lambda,\tau} + \frac{1}{2}\|\mathcal{A}(Y) - b\|_2^2, \\ &\quad s.t.\ X = Y, \quad X \in \mathbb{R}^{m\times n}, Y \in \mathbb{R}^{m\times n}. \end{aligned} \tag{12}$$

The corresponding augmented Lagrangian function of problem (12) is

$$L_\mu(X, Y, Z) = \|X\|_{\lambda,\tau} - \langle Z, X - Y\rangle + \frac{1}{2}\|\mathcal{A}(Y) - b\|_2^2 + \frac{\mu}{2}\|X - Y\|_F^2, \tag{13}$$

where $Z \in \mathbb{R}^{m\times n}$ is the Lagrangian multiplier, and $\mu > 0$ is the penalty parameter for the violation of the linear constraint.

Given (X^k, Y^k, Z^k), the iteration scheme of the ADM for problem (12) can be described as follows:

$$X^{k+1} \in \underset{X\in\mathbb{R}^{m\times n}}{\operatorname{argmin}}\, L_\mu\left(X, Y^k, Z^k\right), \tag{14}$$

$$Y^{k+1} \in \underset{Y\in\mathbb{R}^{m\times n}}{\operatorname{argmin}}\, L_\mu\left(X^{k+1}, Y, Z^k\right), \tag{15}$$

$$Z^{k+1} = Z_k - \mu\left(X^{k+1} - Y^{k+1}\right), \tag{16}$$

where argmin denotes the minimal set to an optimization problem.

It is easy to see that the X-subproblem (14) can be reformulated as

$$\begin{aligned} X^{k+1} &\in \underset{X}{\operatorname{argmin}} \|X\|_{\lambda,\tau} - \langle Z^k, X - Y^k\rangle + \frac{\mu}{2}\|X - Y^k\|_F^2 \\ &= \underset{X}{\operatorname{argmin}} \|X\|_{\lambda,\tau} + \frac{\mu}{2}\left\|X - \left(Y^k + \frac{1}{\mu}Z^k\right)\right\|_F^2. \end{aligned} \tag{17}$$

Let $\widehat{Y} = Y^k + \frac{1}{\mu}Z^k$, according to the Corollary 2.1, it is easy to show that the closed-form solutions of (17) can be described as

$$X^{k+1} = \mathcal{S}_{\lambda,\tau,\mu}(\widehat{Y}). \tag{18}$$

On the other hand, the Y-subproblem (15) can be reformulated as follows

$$Y^{k+1} = \underset{Y}{\operatorname{argmin}} \langle Z^k, Y - X^{k+1}\rangle + \frac{1}{2}\|\mathcal{A}(Y) - b\|_2^2 + \frac{\mu}{2}\|X^{k+1} - Y\|_F^2. \tag{19}$$

The Y-subproblem is a quadratical optimization problem and admits a unique solution Y^{k+1} satisfying

$$\left(\mu I + \mathcal{A}^*\mathcal{A}\right) Y^{k+1} = \mu X^{k+1} - Z^k + \mathcal{A}^* b, \tag{20}$$

where I is an identity matrix, and the $\mathcal{A}^*$ is the adjoint of $\mathcal{A}$. In practice, it may be expensive to solve the above linear system directly, we can apply the conjugate gradient method [38] for solving it. For the sake of simplicity, let $C = \mu I + \mathcal{A}^*\mathcal{A}$, and $D_k = \mu X^{k+1} - Z^k + \mathcal{A}^* b$. Let $\widehat{Y}_0 = Y^k$, $\widehat{R}_0 = C\widehat{Y}_0 - D_k$ and $\widehat{P}_0 = -\widehat{R}_0$, and then the sequence $\{\widehat{Y}_i\}$ can be computed iteratively as

$$\begin{cases} \alpha_i = -\frac{\langle \widehat{R}_i, \widehat{P}_i\rangle}{\langle \widehat{P}_i, C\widehat{P}_i\rangle}, \\ \widehat{Y}_{i+1} = \widehat{Y}_i + \alpha_i \widehat{P}_i, \\ \widehat{R}_{i+1} = C\widehat{Y}_{i+1} - D_k, \\ \beta_{i+1} = \frac{\langle \widehat{R}_{i+1}, C\widehat{P}_i\rangle}{\langle \widehat{P}_i, C\widehat{P}_i\rangle}, \\ \widehat{P}_{i+1} = -\widehat{R}_{i+1} + \beta_{i+1}\widehat{P}_i, \end{cases} \tag{21}$$

and set $Y^{k+1} = \widehat{Y}_i$, where $\langle\cdot,\cdot\rangle$ denotes the standard inner trace product of matrix. It is worth being mentioned that the linear conjugate gradient method is very efficient to solve the linear system, and its convergence properties are well studied. For further details, we can refer to [39].

Based on the analysis above, we give the basic framework of the ADM for solving MCP-RLS problem with fixed regularization parameter λ as follows:

Remark 2.1 In the Step 3, (1)–(3) are the iterative process of linear conjugate gradient method, where $\bar{i} = 5$. Because of the simple structure of the coefficient matrix C, it can get the approximate optimal solution X^* of the subproblem within 5 steps.

In the Step 1, the terminated condition can be the relative error between the original matrix M and the optimal solution produced by the ADM, that is

$$RelErr = \frac{\|X^* - M\|_F}{\|M\|_F} \le \epsilon \tag{22}$$

for some $\epsilon > 0$. Similarly, we can terminate the algorithm if the relative change of the sequence $\{X^k, Y^k\}$ is less than ϵ.

Algorithm 2.1. (ADM)

Initialization: *Input* X^0, Y^0, *and* Z^0. *Given constants* μ, τ *and* λ. *Set* $k = 0$.
Step 1. *Stop if some terminated condition is satisfied. Otherwise, continue.*
Step 2. *Compute* X^{k+1} *via* (18) with fixed Y^k *and* Z^k;
Step 3. *Compute* Y^{k+1} *with fixed* X^{k+1} *and* Z^k.
(1) Let $\widehat{Y}_0 = Y^k$ *and* $\epsilon > 0$, *set* $i = 0$.
(2) While $\widehat{R}_i > \epsilon$ *and* $i < \bar{i}$,
compute the $\widehat{Y}_i$ *via* (21);
Let $i = i + 1$;
(3) Set $Y^{k+1} = \widehat{Y}_i$.
Step 4. *Update* Z^{k+1} *via* (16) *with fixed* X^{k+1} *and* Y^{k+1};
Step 5. *Let* $k = k + 1$. *Go to* **Step 1**.

Algorithm 2.1 is designed for solving MCP-RLS problem with a fixed regularization parameter λ. However, a good regularization parameter that leads to good approximation to the low rank target may not be known in advance. Here we propose novel data driven regularization parameter selection rules during the continuation process. To precise, given a decreasing sequence of parameter $\{\lambda_s\}_s$, we run ADM to solve the λ_{s+1}-problem initialized with the solution of λ_s-problem. Meanwhile, the discrepancy $d^s =: \|\mathcal{A}(X(\lambda_s)) - b\|_2$ and the Bayesian information criterion (BIC) value $B^s =: \frac{1}{2}\|\mathcal{A}(X(\lambda_s)) - b\|_2^2 + \ln(n * m) * rank(X(\lambda_s))$ are calculated and stored at each λ_s. The regularization parameter is selected as the first one that makes $d^s > \delta$ when the noise level δ is given (DP), or the one that makes minimum BIC value when the noise level is not known (BIC). We summarize the above ideas in the following ADM with continuation (ADMc) algorithm for solving the MCP-RLS problem:

Algorithm 2.2. (ADMc)

Initialization: *Input* $\lambda_0 \geq \|\mathcal{A}^* b\|_\infty$, $X(\lambda_0) = \mathbf{0}$, $Y(\lambda_0) = \mathcal{A}^* b$, $\rho \in (0, 1)$.
1. **for** *s=1, 2, 3, . . .* **do**
2. Set $\lambda_s = \lambda_0 \rho^s$ *and* $(X^0, Y^0) = (X(\lambda_{s-1}), Y(\lambda_{s-1}))$.
3. Find $X(\lambda_s)$ *and* $Y(\lambda_s)$ *by* Algorithm 2.1.
4. Compute the discrepancy d^s *and BIC value* B^s.
5. **End for**
6. Select $\hat{\lambda}$ *by discrepancy principle (DP) or BIC value.*

Remark 2.2 At the line 4 of Algorithm 2.2, the stopping rule for the regularization parameter λ can be chosen as either discrepancy principle (DP) or Bayesian information criterion (BIC), see [32] for more details.

2.3 Convergence Analysis

In this subsection, we give the convergence analysis of the ADM for solving the MCP-RLS problem. Due to the nonconvexity of the problem, it is not easy to prove that the ADM converges to a global minimizer. Wen et al have proposed some preliminary analysis of the convergent behavior of ADM in [40,41]. They show that any limit point of the iteration sequence generated by the ADM is a Karush-Kuhn-Tucker (KKT) point under some suitable assumptions. Following their work, we can establish the convergent behavior of the proposed ADM for the MCP-RLS problem.

A triple (X^*, Y^*, Z^*) is called KKT point of problem (12) if it satisfies the following system:

$$\begin{cases} X^* = \mathcal{S}_{\lambda,\tau,\mu}\left(Y^* + \frac{1}{\mu}Z^*\right), \\ \mathcal{A}^*\left(\mathcal{A}(Y^*) - b\right) + Z^* = 0, \\ X^* = Y^*. \end{cases} \tag{23}$$

Theorem 2.1 *Let $\{(X^k, Y^k, Z^k)\}$ be a sequence generated by ADM. Assume that $lim_{k\to\infty}$ $\|Z^{k+1} - Z^k\|_F = 0$, and $\{Y^k\}$ is bounded, then there exists a subsequence of $\{(X^k, Y^k, Z^k)\}$ such that it converges to a KKT point of problem (12).*

Proof Since $\mu > 0$, and

$$\lim_{k\to\infty} \|Z^{k+1} - Z^k\|_F = 0, \tag{24}$$

we can obtain from (16) that

$$\lim_{k\to\infty} \|Y^{k+1} - X^{k+1}\|_F = \lim_{k\to\infty} \frac{1}{\mu}\|Z^{k+1} - Z^k\|_F = 0. \tag{25}$$

It follows from the boundedness of $\{Y^k\}$ and the condition (25) that $\{X^k\}$ is bounded. We can see that $\{Z^k\}$ is also bounded from (20). Since $\{(X^k, Y^k, Z^k)\}$ is bounded and the augmented Lagrangian function L_μ is continuous, we can obtain that $L_\mu(X^k, Y^k, Z^k)$ is bounded. It is clearly to see that L_μ is strongly convex with respect to the variable Y, so it holds that for any Y and $\triangle Y$,

$$L_\mu\left(X, Y + \triangle Y, Z\right) - L_\mu(X, Y, Z) \geq \nabla_Y L_\mu(X, Y, Z)^T \triangle Y + c\|\triangle Y\|_F^2, \tag{26}$$

where $c > 0$ is a constant. In addition, Y^{k+1} minimize (15), so the following condition holds, which is

$$\nabla_Y L_\mu\left(X^{k+1}, Y^{k+1}, Z^k\right)^T \left(Y^k - Y^{k+1}\right) \geq 0. \tag{27}$$

We note $\triangle Y = Y^k - Y^{k+1}$, then combining (26) with (27), we can obtain that

$$L_\mu\left(X^{k+1}, Y^{k+1} + (Y^k - Y^{k+1}), Z^k\right) - L_\mu\left(X^{k+1}, Y^{k+1}, Z^k\right) \geq c\|Y^{k+1} - Y^k\|_F^2 \tag{28}$$

that is,

$$L_\mu(X^{k+1}, Y^k, Z^k) - L_\mu\left(X^{k+1}, Y^{k+1}, Z^k\right) \geq c\|Y^{k+1} - Y^k\|_F^2 \tag{29}$$

Moreover, since X^{k+1} minimize (14), we have

$$L_\mu\left(X^{k+1}, Y^k, Z^k\right) \leq L_\mu\left(X^k, Y^k, Z^k\right), \tag{30}$$

Thus together with (29), we can get that

$$L_\mu\left(X^k, Y^k, Z^k\right) - L_\mu\left(X^{k+1}, Y^{k+1}, Z^{k+1}\right) + \frac{1}{\mu}\|Z^{k+1} - Z^k\|_F^2 \geq c\|Y^{k+1} - Y^k\|_F^2 \tag{31}$$

For simplicity, this inequation can be rewritten as

$$L_k - L_{k+1} + z_k \geq y_k \geq 0, \tag{32}$$

where $L_k = L_\mu(X^k, Y^k, Z^k)$, $L_{k+1} = L_\mu(X^{k+1}, Y^{k+1}, Z^{k+1})$, $z_k = \frac{1}{\mu}\|Z^{k+1} - Z^k\|_F^2$, $y_k = c\|Y^{k+1} - Y^k\|_F^2$. Since L_k is bounded, there exists a subsequence k_j such that

$$\lim_{k_j \to \infty} L_{k_j} = \underline{\lim}_{k\to\infty} L_k.$$

Due the the nonnegativity of y_k, z_k and $z_k \to 0$, we have

$$\begin{aligned} 0 \le \underline{\lim}_{k_j\to\infty} y_{k_j} \le \underline{\lim}_{k_j\to\infty} \left(L_{k_j} - L_{k_j+1} + z_k\right) = \lim_{k_j\to\infty} \left(L_{k_j} + z_{k_j}\right) \\ -\underline{\lim}_{k_j\to\infty} L_{k_j+1} \le 0, \end{aligned} \tag{33}$$

which shows the following equation:

$$\underline{\lim}_{k_j\to\infty} y_{k_j} = 0 \tag{34}$$

and be equivalent as

$$\underline{\lim}_{k_j\to\infty} \|Y^{k_j+1} - Y^{k_j}\|_F = 0. \tag{35}$$

Together with (25), we can also get that

$$\underline{\lim}_{k_j\to\infty} \|X^{k_j+1} - X^{k_j}\|_F = 0. \tag{36}$$

Then, by the boundedness of $\{X^{k_j}, Y^{k_j}, Z^{k_j}\}$, there exists a convergence subsequence still be denoted by $\{k_j\}$ such that $\{X^{k_j}, Y^{k_j}, Z^{k_j}\}$ converges to (X^*, Y^*, Z^*). From (25) and

$$\lim_{k_j\to\infty} Y^{k_j} = Y^*, \lim_{k_j\to\infty} X^{k_j} = X^*. \tag{37}$$

Thus, we can obtain that the following condition holds, that is,

$$X^* = Y^*. \tag{38}$$

The first order optimality condition associated with the subproblem (15) can be written as (20), which can be transformed into the following form

$$\mathcal{A}^* \left(\mathcal{A}\left(Y^{k+1}\right) - b\right) = Z^{k+1} - Z^k - Z^{k+1} + \mu \left(X^{k+1} - Y^{k+1}\right). \tag{39}$$

Taking the limit of the both sides of (39) on k_j and together with (24) and (37), we can obtain the following condition holds:

$$\mathcal{A}^* \left(\mathcal{A}(Y^*) - b\right) + Z^* = 0. \tag{40}$$

Similarly, we can get the first order optimality conditions associated with the subproblem (18) as follow

$$X^{k+1} = \mathcal{S}_{\lambda,\tau,\mu}\left(Y^k + \frac{1}{\mu} Z^k\right). \tag{41}$$

Using the condition (35) and taking the limit of the both sides of (41) on k_j, we get the following condition holds, i.e.,

$$X^* = \mathcal{S}_{\lambda,\tau,\mu}\left(Y^* + \frac{1}{\mu} Z^*\right). \tag{42}$$

Hence, combining (42) with (38) and (40), we verify that (X^*, Y^*, Z^*) is a KKT point of (12). □

Remark 2.3 To obtain convergence result of using ADM on nonconvex problems, some similar assumptions as the boundedness of $\{Y^k\}$ and (24) are also used in [41,42]. These assumptions are often observed in the numerical tests, see Appendix 1 for the numerical verification. On the other hand, if the $\{Y^k\}$ is unbounded, the produced sequence $\{X^k\}$ may be not convergent. It is closely related to the non-existence of the global minimizer, see Appendix 2 for an example.

3 Numerical Experiments

In this section, we report some numerical results for solving the MCP-RLS problem and the MC problem (7) on both simulated and real data sets, which show the well performance of the MCP model and the efficiency of the proposed ADMc (Algorithm 2.2). In our numerical experiments, m and n represent the matrix dimension, r is the rank of original matrix, and p denotes the number of measurements. Given $r \leq \min(m, n)$, we generate $M = M_L M_R^T$, where matrix $M_L \in \mathbb{R}^{m\times r}$ and $M_R \in \mathbb{R}^{n\times r}$ are generated with independent identically distributed Gaussian entries. The subset Ω of p elements is selected uniformly at random entries form $\{(i, j) : i = 1, \ldots, m, j = 1, \ldots, n\}$. The p elements of the known Ω can be generated by Matlab script"$randsmple(p, m\times n)$". And we choose the partial discrete cosine transform (DCT) matrix as the linear map $\mathcal{A}$. Since the DCT matrix-vector multiplication is implemented implicitly by FFT, this enables us to test problem more efficiently. The linear measurements b are set to be $b = \mathcal{A}(M) + \omega$, where ω is the additive Gaussian noise of zero mean and standard deviation σ, which will be specified in different test data sets.

We use $sr = p/(mn)$ to denote the sampling ratio, and $dr = r(m + n - r)$ to denote the number of degree of freedom for a real-valued rank r matrix. As mentioned in [9,43], when the ratio p/dr is greater than 3, the problem can be viewed as an easy problem. On the contrary, it is called as a hard problem. Another ratio is $FR = r(m + n - r)/p$, it is also important for successfully recovering the matrix M. If $FR > 1$, it is impossible to recover matrix because there is an infinite number of matrices X with rank r with the given entries [8]. So the FR varies in (0, 1) in this paper. In addition, we take $\mu = 0.5$, $\tau = 2.7$ and σ as noise level. The regularization parameter λ is chosen by the DP or BIC rule.

In all tests, let X^* be the optimal solution produced by the proposed method, we use the relative error to measure the quality of X^* to original M, i.e.

$$RelErr = \frac{\|X^* - M\|_F}{\|M\|_F}. \tag{43}$$

We say that M is recovered successfully by X^* if the corresponding $RelErr$ is less than 10^{-3}, which has been used in [8,13]. In all the tests, we take the $RelErr = 10^{-4}$ as the terminal condition.

Given the computing of a SVD is needed at each iteration for our proposed method, here we use PROPACK [44] package to evaluate partial SVD. Since PROPACK can not automatically compute only those singular values greater than $\tau\lambda$, we need to choose the predetermined number sv^k of singular values to be computed at the k-th iteration. As in [15], initializing $sv_0 = \min(m, n)/20$, if $svp_k < sv_k$, we set $sv_{k+1} = svp_k + 1$; if $svp_k = sv_k$, we have $sv_{k+1} = svp_k + 5$, where svp_k represents the number of positive singular values of $\widehat{Y}$. All experiments except Table 7 are performed under Window 7 premium and MATLAB v7.8(2009a) running on a Lenovo laptop with an Intel core CPU at 2.4GHz and 2 GB memory.

Table 2 Numerical results of ADM, ADMc, APGL and PD for easy matrix completion problems, $m = n, sr = 0.5$

(m, r)	p/dr	ADM			ADMc		APGL		PD	
		λ	Time	RelErr	Time	RelErr	Time	RelErr	Time	RelErr
(200,10)	5.128	35	1.07	8.79e−6	1.99	7.00e−7	0.42	2.94e−5	5.48	9.51e−6
(300,10)	7.627	45	1.53	8.92e−6	3.44	2.98e−7	0.77	9.53e−5	10.37	7.94e−6
(400,10)	10.127	55	3.12	7.58e−6	6.44	1.77e−7	1.29	1.76e−5	26.32	8.83e−6
(500,10)	12.626	65	4.65	7.67e−6	10.75	9.06e−8	2.18	8.11e−6	43.15	8.31e−6
(600,10)	15.126	75	5.86	8.25e−6	17.09	8.94e−8	3.24	4.14e−6	106.60	7.59e−6
(700,10)	17.626	85	8.88	5.65e−6	20.18	6.44e−8	4.15	2.53e−6	204.26	7.30e−6
(800,10)	20.126	95	12.21	7.03e−6	31.17	6.16e−8	5.47	1.24e−6	294.80	6.32e−6
(900,10)	22.626	105	14.13	5.87e−6	42.96	4.62e−8	7.83	2.38e−7	293.32	8.90e−6
(1000,10)	25.126	115	18.65	9.45e−6	69.39	3.89e−8	16.12	2.12e−7	548.78	8.99e−6

3.1 Test on Matrix Completion Problems

In this subsection, we apply the proposed ADMc for solving the matrix completion problem (7). In order to illustrate the performance of ADM for nonconvex MCP matrix completion problem, we compare the proposed ADM (Algorithm 2.1) and ADMc (Algorithm 2.2) with the state-of-the-art method $APGL$[1] [15] and PD[2] [37]. The PD method solves the rank minimization problem (1) without noise, and the APGL solves the following nuclear norm matrix completion problem

$$\min_{X \in \mathbb{R}^{m\times n}} \mu \|X\|_* + \frac{1}{2} \sum_{(i,j)\in\Omega} |X_{i,j} - M_{i,j}|^2 \quad \forall (i,j) \in \Omega. \tag{44}$$

In running the codes of APGL and PD, default values are used for all parameters. Firstly, we test them for the easy matrix completion problems and report the numerical results in Table 2. From the Table 2, we can see that the ADM with some fixed λ needs less time than the ADMc and PD method, especially in the high dimensional cases, and it also can attain a high accuracy as the others. On the another hand, the ADM is comparable with the state-of-the-art APGL on the running time and accuracy. Thus it shows that the nonconvex MCP matrix completion problem (7) can be solved by ADM efficiently. From the Table 2, it is also clear to see that the ADMc can get higher accuracy than the others with a little more time. Moreover, it dose better than the PD at the time and accuracy. From the above analysis, we can see that our proposed ADM and ADMc are comparable to the APGL and PD for solving the easy matrix completion problems, in which the ADM performs better, but it needs to try for choosing a suitable λ when the tested problem varies. However, the ADMc performs with an automatical choice for the regularization parameter λ, which makes the proposed method is more convenient. Therefore we will use the ADMc to solve the different situations of the matrix MCP minimization problem in the following tests.

For further illustrate the efficiency of the ADMc, we test it with different r in the following test. The results are shown in the Table 3, which shows that the problems become harder and

[1] The APGL code is downloaded form http://www.math.nus.edu.sg/~mattohkc/NNLS.html.

[2] The PD code is downloaded from http://www.sfu.ca/~yza30/homepage/PD_Rank/downloads.html.

Table 3 Numerical results of ADMc, APGL and PD for matrix completion with different r, $m = n$, $sr = 0.5$

(m, r)	p/dr	ADMc		APGL		PD	
		Time	RelErr	Time	RelErr	Time	RelErr
(200,5)	10.127	1.30	3.21e−7	0.36	4.09e−6	3.03	6.26e−6
(200,10)	5.128	1.98	7.00e−7	0.40	2.94e−5	5.44	9.51e−6
(200,15)	3.463	2.48	8.77e−7	0.76	8.32e−5	5.68	1.18e−5
(200,20)	2.632	4.46	1.39e−6	0.97	3.86e−4	10.22	1.90e−5
(200,25)	2.133	5.08	8.94e−6	1.34	1.30e−4	12.47	2.59e−5
(200,30)	1.802	5.50	2.54e−5	3.36	2.66e−1	22.58	3.22e−5
(200,35)	1.566	7.69	5.85e−5	3.61	3.31e−1	33.87	4.09e−5
(200,40)	1.389	17.11	8.64e−5	3.91	3.76e−1	84.58	5.71e−5

Table 4 Numerical results of ADMc, APGL and PD for hard matrix completion problems, $m = n$, $sr = 0.5$

(m, r)	p/dr	ADMc		APGL		PD	
		Time	RelErr	Time	RelErr	Time	RelErr
(100,10)	2.632	1.29	2.41e−6	0.32	5.22e−4	1.62	1.73e−5
(100,15)	1.802	1.99	4.12e−5	0.68	4.27e−4	3.47	3.60e−5
(200,20)	2.632	4.03	1.39e−6	0.90	3.86e−4	9.79	1.90e−5
(200,25)	2.133	4.58	8.94e−6	1.35	1.30e−4	12.18	2.59e−5
(300,30)	2.632	9.25	1.47e−6	2.15	6.40e−5	28.92	2.01e−5
(300,35)	2.276	11.65	2.16e−6	2.39	3.99e−4	37.92	2.16e−5
(400,40)	2.632	20.98	1.15e−6	3.90	2.62e−4	80.47	1.85e−5
(400,45)	2.355	23.12	1.54e−6	5.17	6.75e−5	84.37	2.12e−5
(500,50)	2.632	40.72	1.59e−6	7.58	1.98e−4	206.85	2.00e−5
(500,55)	2.405	41.36	1.62e−6	11.21	9.09e−5	211.61	2.11e−5
(600,60)	2.632	60.60	1.45e−6	11.95	3.18e−4	699.98	1.86e−5
(600,65)	2.440	64.87	1.52e−6	15.35	4.66e−5	537.15	2.12e−5
(700,70)	2.632	98.35	1.50e−6	17.13	3.04e−4	775.14	1.92e−5
(700,75)	2.465	103.03	1.35e−6	21.30	4.73e−5	1178.32	1.98e−5
(800,80)	2.632	138.28	1.15e−6	27.29	4.36e−5	1617.95	1.92e−5
(900,90)	2.632	193.51	1.44e−6	39.57	9.70e−5	2295.38	1.94e−5
(1000,100)	2.632	260.54	1.51e−6	53.02	9.94e−6	−−	−

harder with r increasing. When r is more than 25, the APGL can not obtain a high accuracy, but the others can still solve efficiently. Form this, it shows that the nonconvex model is more efficient than the nuclear norm model for these problems without very low rank. In addition, the ADMc can get a higher accuracy than PD with less time. So we can conclude that the proposed nonconvex model and the proposed ADMc is robust and efficient.

Next, we choose some hard matrix completion problems for testing. As shown in the Table 4, we can see that the ADMc performs better than PD at running time and accuracy, and gets higher accuracy than APGL but needs more time. $'-'$ represents the running time of PD is more than 3000s, so we force it to stop. In particularly, when the dimension becomes

Table 5 Numerical results of ADMc, APGL and IADM-CG for matrix completion problems with noise, $m = n, \sigma = 1e - 3$

(m, r)	sr	p/dr	ADMc		APGL		IADM-CG	
			Time	RelErr	Time	RelErr	Time	RelErr
(300,30)	0.3	1.579	19.07	5.20e−4	5.89	4.45e−1	9.83	2.50e−1
	0.5	2.632	4.09	1.42e−4	2.35	1.44e−4	9.93	6.95e−4
	0.7	3.684	6.82	1.00e−4	6.82	1.00e−4	10.24	3.75e−4
(500,50)	0.3	1.579	64.60	4.45e−4	24.61	3.68e−1	35.79	2.31e−1
	0.5	2.632	19.26	9.99e−5	7.85	2.22e−4	37.19	4.05e−4
	0.7	3.684	18.59	9.16e−5	6.16	1.10e−4	37.80	2.27e−4
(700,70)	0.3	1.579	157.83	3.59e−4	97.81	2.83e−1	96.61	2.28e−1
	0.5	2.632	44.97	9.71e−5	18.41	3.15e−4	96.97	2.90e−4
	0.7	3.684	39.85	7.88e−5	15.92	6.65e−5	98.16	1.66e−4
(900,90)	0.3	1.579	306.25	3.18e−4	130.30	1.65e−1	196.32	2.29e−1
	0.5	2.632	83.60	9.48e−5	40.53	1.22e−4	200.86	2.29e−4
	0.7	3.684	76.40	7.46e−5	29.06	6.22e−5	208.10	1.32e−4

higher, the APGL is faster than the ADMc and PD. The PD solves these problems with full SVD, so it needs a little more time when the dimension is high. By these limited tests, we can see that the proposed ADMc is comparable with the APGL and PD for solving hard completion problems.

In some practical applications, the observations may contain some noise. For this situation, we test it for some easy and hard matrix completion problems with noise level $\sigma = 1e - 3$ in the following tests. Because the PD method is designed for solving the noiseless problems, in the following tests, we change to compare ADMc with IADM-CG [38] and APGL for the problems with noise. In running the code of IADM-CG, we set the parameters $maxit = 1000$ and $tol\_relchg = 1e - 4$, and default values are used for other parameters. The numerical results is presented in Table 5, where sr is chosen as 0.3, 0.5 and 0.7. Form the Table 5, we can see that the cases of $sr = 0.3$ and 0.5 are hard problem. For the case of $sr = 0.3$, only the ADMc can obtain the optimal solution. This shows that the nonconvex MCP-RLS model may need less measurements to successfully recover a low rank matrix than nuclear norm minimization. When the problem becomes easier, all algorithms can solve successfully, and the ADMc is comparable with the APGL and better than IADM-CG. From the above analysis, we can see that the proposed nonconvex MCP problem can be more robust than the nuclear norm model, and the ADMc performs well for the nonconvex matrix low rank recovery problems with noise (Fig. 2).

Finally, we test the ADMc for recovering two real corrupted gray images. Firstly, we use the SVD to obtain the low rank-50 images. Then we randomly select 40 % samples from the low rank image, which are the corrupted image with noise level $\sigma = 1e - 3$. Finally, these corrupted images are recovered by the proposed ADMc and APGL. From the Fig. 3, it is not hard to see that the quality of the image (c) restored by ADMc is better than the image (d) restored by APGL. In fact, the relative error and the CPU time of the ADMc is less than that of the APGL for recovering the two images. In particularly, the APGL can not recover successfully the second image with a correct rank but the ADMc can get a image with high quality.

Fig. 2 **a** Corresponding low rank images with $m = n = 512, r = 50$; **b** Randomly masked images from rank 50 with $sr = 40\,\%, \sigma = 1e-3$; **c** Recovered images by ADMc method [ErrRel= 5.08e−3 (*first image*), 2.89e−3 (*second image*)]; **d** Recovered images by APGL method [ErrRel= 4.18e−2 (*first image*), 2.30e−1 (*second image*)]

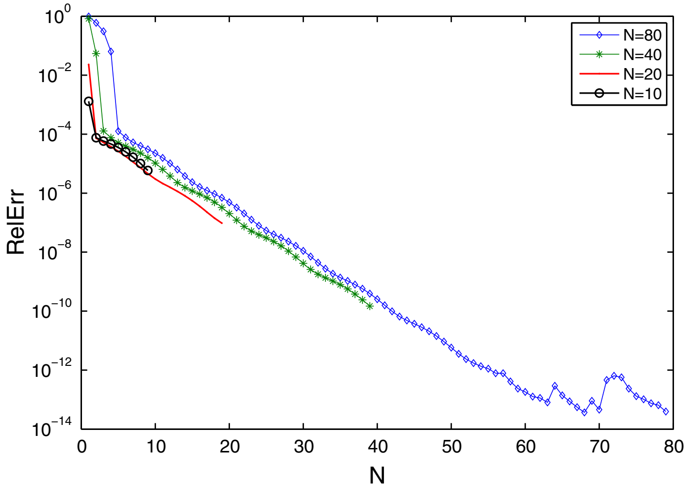


Fig. 3 $m = n = 500$, $r = 10$, $sr = 0.5$, $Time = 122.52, 65.97, 34.24, 23.29$ s (From 80 to 10)

3.2 Test on General Matrix MCP-RLS Problems

In this subsection, we will test the ADMc for the MCP-RLS problem. We choose the partial DCT matrix as an encoder $\mathcal{A}$, and compare the ADMc with the APGL for solving noisy problems. In the first test, we set the noise level σ to be $1e-3$ and let r be from 10 to 100. The numerical results can be seen in Table 6, and it shows that the ADMc uses more time than the APGL when the problem is easy. However, when the problems become hard, i.e. $r > 80$, the APGL becomes inefficient, where "rX"represents the rank of matrix recovered by APGL, and "$\surd$" shows the rank of matrix recovered successfully. For example, when the $r = 90$, the running time of APGL suddenly increases but the optimal solution can not be obtained. In particularly, when $r > 80$, the APGL can not attain the desired low rank matrix but the ADMc can still recover them successfully. From these numerical results, we can conclude that the

Table 6 Numerical results of ADMc and APGL for MCP-RLS problem, $m = n, sr = 0.5, \sigma = 1e-3$

(m, r)	p/dr	ADMc		APGL		
		Time	RelErr	Time	RelErr	rX
(500,10)	12.63	20.72	9.64e−5	11.89	9.29e−5	$\surd$
(500,20)	6.378	26.56	9.74e−5	14.75	9.43e−5	$\surd$
(500,30)	4.296	32.04	9.79e−5	17.56	9.64e−5	$\surd$
(500,40)	3.255	34.83	1.01e−4	19.23	1.82e−4	$\surd$
(500,50)	2.632	48.92	9.98e−5	21.85	2.67e−4	$\surd$
(500,60)	2.216	51.72	1.12e−4	26.96	1.09e−4	$\surd$
(500,70)	1.920	69.79	1.15e−4	29.22	1.67e−4	$\surd$
(500,80)	1.698	87.80	1.88e−4	36.96	6.58e−4	$\surd$
(500,90)	1.526	124.51	2.14e−4	198.97	1.03e−1	150
(500,100)	1.389	199.61	3.26e−4	203.28	1.24e−1	150
(500,100)	1.277	291.79	8.18e−4	196.59	1.54e−1	126

Table 7 Numerical results of ADMc, APGL for matrix completion problems, $m = n$

$(m, p/dr)$	p	r	ADMc		APGL	
			Time	RelErr	Time	RelErr
(200,4)	15665	10	2.39	6.09e−003	0.87	5.47e−002
(400,4)	31548	10	3.55	6.41e−003	1.50	5.74e−002
(800,4)	63397	10	14.11	9.46e−003	3.15	5.85e−002
(1600,4)	127429	10	62.69	2.53e−002	2.75	5.83e−002
(3200,4)	255709	10	249.45	4.74e−002	4.62	5.88e−002

proposed nonconvex MCP-RLS problem can solve more efficiently the rank minimization problems without very low rank than the convex nuclear norm minimization, in other word, the proposed model is more robust.

Now we should mention the limitation of the proposed method. If the desired matrix is large but with very low rank, the proposed ADMc algorithm is not as efficient as APGL. To test a large scale problem with very low rank, we fix the rank be 10 and $p/dr = 4$ and let the size increase from 200 to 3200. This computation is under Windows 7 premium and MATLAB (2009a) running on a Sony laptop with Intel Core i7 CPU at 2.4GHz and 8 GB memory. We can see from Table 7 that the CPU time for APGL is linearly increasing but for the proposed ADMc is quadratically increasing. This is because the solution to the Y-subproblem in (19) is not necessarily low rank, and hence there is no fast matrix-vector multiplication in partial SVD (e.g. PROPACK) for X-subproblem (18). From all above numerical experiment, we may conclude that when the rank compare with problem size is not very small, the proposed ADMc algorithm is competitive with APGL.

Finally, we give a note on an important parameter N in the continuation technique, which cuts λ into N elements. Then at each outer loop, the N elements of λ is chosen by descendant order. For all the above tests, we set $N = 20$. Here, we choose $N = 10, 20, 40, 80$ respectively for testing the ADMc. From the Fig. 3, we can see that the solution becomes more accurate when N increasing from 10 to 80. The $RelErr$ can reach to 10^{-14} when $N = 80$, which needs more CPU time.

4 Conclusions

In this paper, we developed an alternating direction method for solving the MCP-RLS problem. The proposed MCP-RLS model is extended from the sparse signal case, and is more robust for recovering the low rank minimization problems than the nuclear norm minimization, especially for the problems which have not very low rank. For this nonconvex problem, we showed that the resulting subproblem of ADM has a closed-form solution. Then we demonstrated that the solution sequence produced by the ADM for the nonconvex problem converges to its KKT point. Numerical experiments on random data and real data illustrate that the proposed ADMc method performs well, when it coupled with a continuation strategy. Moreover, the comparing with the state-of-the-art methods have further illustrated that the proposed ADMc is very efficient and promising. There are several avenues for further investigation. For the large scale problem but with very low rank solution, the proposed ADMc algorithm is not be scaling well. To propose an algorithm which can be scaling better deserves further study. And there are a few assumptions in Theorem 2.1, how to remove or validate them are still not clear.

Acknowledgments The authors are grateful to the anonymous referees for their insightful comments, which have led to improved presentation of the paper. The research of Z. Wan is supported by the National Natural Science Foundation of China No. 71471140. Y. Jiao is supported by the Fundamental Research Funds for the Central Universities of China No. 31541411212. X. Lu is supported by the National Natural Science Foundation of China No. 91230108 and No. 11471253.

Appendices

Appendix 1: A Note on Theorem 2.1

In this part, we give some numerical evidences about the assumptions of Theorem 2.1. We choice $m = n = 100$, $r = 5$, $sr = 0.5$, $maxiter = 500$ and $\lambda = 20$. Form the Fig. 4, we find that the $\|Y^k\|_F$ is always bounded and the $\|X^k - Y^k\|_F$ is less than 10^{-15} after 500 iterations. It implies that the condition $\lim_{k\to\infty} \|Z^{k+1} - Z^k\|_F = 0$ holds.

Appendix 2: An Example

We will give an example to show the nonconvex model for the matrix completion problem may not admit a solution if the nonconvex functional is not coercive. Let $m = n = 2$, $\Omega = \{(1, 1); (1, 2); (2, 1)\}$, the observation matrix M be given by $M = \begin{pmatrix} 0 & 1 \\ 1 & - \end{pmatrix}$. We consider the following four nonconvex models:

(1) $\min \sum_{(i,j)\in\Omega} |X_{i,j} - M_{i,j}|^2, \quad s.t.\ \mathrm{rank}(X) \le 1,$
(2) $\min \sum_{(i,j)\in\Omega} |X_{i,j} - M_{i,j}|^2 + \mathrm{rank}(X),$

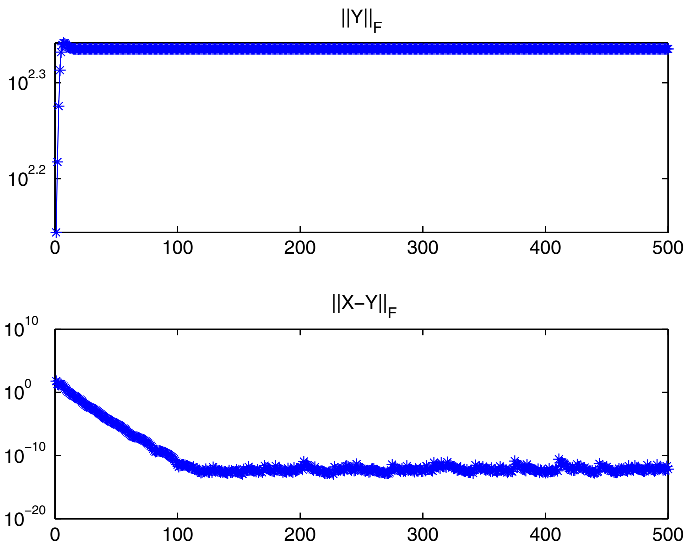


Fig. 4 Test results on the $\|Y^k\|_F$ and $\|X^k - Y^k\|_F$

(3) $\min \sum_{(i,j)\in\Omega} |X_{i,j} - M_{i,j}|^2, \quad s.t. \ \|X\|_{\lambda,\tau} \leq 1,$
(4) $\min \sum_{(i,j)\in\Omega} |X_{i,j} - M_{i,j}|^2 + \|X\|_{\lambda,\tau},$

where $\lambda = 2$ and $\tau = 2$ in the scalar MCP function, then $\rho(t) = \begin{cases} 2t - \frac{t^2}{4}, & |t| < 4 \\ 4, & |t| \geq 4 \end{cases}$ and $\|X\|_{2,2} = \rho(\sigma_1) + \rho(\sigma_2)$, where σ_1 and σ_2 are two singular values of X. Clearly $\rho(t) > t$ for all $0 < t < 4$. We will show that problems (1) to (4) have no solutions.

For problem (1), let $X_n = \begin{pmatrix} 1/n & 1 \\ 1 & n \end{pmatrix}$, then we obtain that the object function has the infimum 0. But it is clear that 0 can not be obtained, which implies the nonexistence of solution to problem (1). The similar argument can be applied to show problem (3) does not admit a solution. To see this, firstly X_n defined as above provides a minimum sequence, it remains to show 0 is not reachable. For any $Z = \begin{pmatrix} 0 & 1 \\ 1 & c \end{pmatrix}$, it has two nonzero singular values $\sigma_1 \geq 1 \geq \sigma_2 > 0$, which implies that $\|Z\|_{\lambda,\tau} \geq \min(4, \sigma_1) + \sigma_2 > 1$. Therefore problem (3) has no solution.

For problem (2), we first notice that the cost functional has a lower bounded 1 and can not obtain this value. Then X_n as above implies that 1 is the exact lower bound, and hence the nonexistence of solution. To see problem (4), we only need to show 1 is its unreachable exact lower bound. To see this, let the cost function of problem (4) be $f(X)$. For any $Z = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$, we can compute $Z^t Z = \begin{pmatrix} a^2 + b^2 & ac + bd \\ ac + bd & c^2 + d^2 \end{pmatrix}$, and hence two eigenvalues λ_1, λ_2 are positive and satisfy

$$\lambda_1 + \lambda_2 = a^2 + b^2 + c^2 + d^2, \quad \lambda_1 \lambda_2 = (ad - bd)^2.$$

From the definition of scalar MCP function, we have

$$\|Z\|_{\lambda,\tau} \geq \min(\sigma_1, 4) + \min(\sigma_2, 4),$$

where σ_1 and σ_2 are two singular values of Z (i.e., $\sigma_i = \sqrt{\lambda_i}$). If the sum of two singular value great than or equal to 1, then $f(Z) \geq 1$ and the equality never happens (since when $1 \geq \sigma_1 > 0$, the inequality $\rho(\sigma_1) > \sigma_1$ holds). Otherwise let us assume the sum of two singular values is less than 1, we have

$$f(Z) \geq \sigma_1 + \sigma_2 + a^2 + (b-1)^2 + (c-1)^2 \geq \lambda_1 + \lambda_2 + a^2 + (b-1)^2 + (c-1)^2.$$

By observing $b^2 + (1-b)^2 \geq 1/2, c^2 + (1-c)^2 \geq 1/2$, we obtain $f(Z) \geq 1$ and the equality can not be obtained (otherwise $a = 0, b = c = 1/2, \sigma_1 = \sigma_2 = 0$, which is a contradiction).

One may also find the following two minimization problems

(5) $\min \operatorname{rank}(X), \quad s.t. \ \sum_{(i,j)\in\Omega} |X_{i,j} - M_{i,j}|^2 \leq \delta,$
(6) $\min \|X\|_{\lambda,\tau}, \quad s.t. \ \sum_{(i,j)\in\Omega} |X_{i,j} - M_{i,j}|^2 \leq \delta$

have solutions, but the solutions are unstable with respect to noise level δ.

In general, if the desirable matrix is not a low rank matrix, its low-rank approximation is either not exist or not stable. This explains that the assumptions in Theorem 2.1 are necessary in general, to avoid the possible non-stable computation. It also explains that some existing matrix completion algorithms work well for easy problem ($p/dr \geq 3$) but may be not so efficient for hard problem. On the other hand, if we are interesting in some local minimizers, it may exist and stable. From our numerical experiments, it seems that the proposed ADMc converges to some stable local minimizer and hence it works well for both easy and hard problems.

References

1. Srebro, N.: Learning with matrix factorizations. Doctoral dissertation, Massachusetts Institute of Technology (2004)
2. Goldberg, K., Roeder, T., Gupta, D., Perkins, C.: Eigentaste: a constant time collaborative filtering algorithm. Inf. Retr. **4**(2), 133–151 (2001)
3. Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer, V.R., Anders, K., Eisen, M.B., Brown, P.O., Botstein, D., Futcher, B.: Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. Mol. Biol. Cell **9**(12), 3273–3297 (1998)
4. Netfix prize website http://www.netflixprize.com
5. Mohan, K., Fazel, M.: Reweighted nuclear norm minimization with application to system identification. In: American Control Conference, 2010, pp. 2953–2959. IEEE (2010)
6. Fazel, M., Hindi, H., Boyd, S.: Rank minimization and applications in system theory. In: American Control Conference, 2004. Proceedings of the 2004, vol. 4, pp. 3273–3278. IEEE (2004)
7. Candès, E.J., Li, X., Ma, Y., Wright, J.: Robust principal component analysis? J. ACM **58**(3), 11 (2011)
8. Ma, S., Goldfarb, D., Chen, L.: Fixed point and bregman iterative methods for matrix rank minimization. Math. Program. **128**(1–2), 321–353 (2011)
9. Candès, E.J., Recht, B.: Exact matrix completion via convex optimization. Found. Comput. Math. **9**(6), 717–772 (2009)
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. Keshavan, R.H., Montanari, A., Oh, S.: Matrix completion from a few entries. IEEE Trans. Inf. Theory **56**(6), 2980–2998 (2010)
12. Tütüncü, R.H., Toh, K.-C., Todd, M.J.: Solving semidefinite-quadratic-linear programs using sdpt3. Math. Program. **95**(2), 189–217 (2003)
13. Cai, J.-F., Candès, E.J., Shen, Z.: A singular value thresholding algorithm for matrix completion. SIAM J. Optim. **20**(4), 1956–1982 (2010)
14. Liu, Y.-J., Sun, D., Toh, K.-C.: An implementable proximal point algorithmic framework for nuclear norm minimization. Math. Program. **133**(1–2), 399–436 (2012)
15. Toh, K.-C., Yun, S.: An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. Pac. J. Optim. **6**(615–640), 15 (2010)
16. Xiao, Y.-H., Jin, Z.-F.: An alternating direction method for linear-constrained matrix nuclear norm minimization. Numer. Linear Algebra Appl. **19**(3), 541–554 (2012)
17. Yang, J., Yuan, X.: Linearized augmented lagrangian and alternating direction methods for nuclear norm minimization. Math. Comput. **82**(281), 301–329 (2013)
18. 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)
19. Hu, Y., Zhang, D., Ye, J., Li, X., He, X.: Fast and accurate matrix completion via truncated nuclear norm regularization. IEEE Trans. Pattern Anal. Mach. Intell. **35**(9), 2117–2130 (2013)
20. Chen, X., Fengmin, X., Ye, Y.: Lower bound theory of nonzero entries in solutions of $l_2 - l_p$ minimization. SIAM J. Sci. Comput. **32**(5), 2832–2852 (2010)
21. Zhang, T.: Analysis of multi-stage convex relaxation for sparse regularization. J. Mach. Learn. Res. **11**, 1081–1107 (2010)
22. Fan, J., Li, R.: Variable selection via nonconcave penalized likelihood and its oracle properties. J. Am. Stat. Assoc. **96**(456), 1348–1360 (2001)
23. Zhang, C.H.: Nearly unbiased variable selection under minimax concave penalty. Ann. Stat. **38**(2), 894–942 (2010)
24. Li, Y.-F., Zhang, Y.-J., Huang, Z.-H.: A reweighted nuclear norm minimization algorithm for low rank matrix recovery. J. Comput. Appl. Math. **263**, 338–350 (2014)
25. Lu, C., Tang, J., Yan, S., Lin, Z.: Generalized nonconvex nonsmooth low-rank minimization. arXiv preprint arXiv:1404.7306 (2014)
26. Lai, M.-J., Yangyang, X., Yin, W.: Improved iteratively reweighted least squares for unconstrained smoothed l_q minimization. SIAM J. Numer. Anal. **51**(2), 927–957 (2013)
27. Wang, S., Liu, D., Zhang, Z.: Nonconvex relaxation approaches to robust matrix recovery. In: Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, pp. 1764–1770. AAAI Press (2013)
28. Jiao, Y., Jin, B., Lu, X.: A primal dual active set algorithm for a class of nonconvex sparsity optimization. arXiv preprint. arXiv:1310.1147 (2013)
29. Yang, J., Zhang, Y.: Alternating direction algorithms for l_1-problems in compressive sensing. SIAM J. Sci. Comput. **33**(1), 250–278 (2011)

30. Xiao, Y., Zhu, H., Soon-Yi, W.: Primal and dual alternating direction algorithms for $l_1 - l_1$-norm minimization problems in compressive sensing. Comput. Optim. Appl. **54**(2), 441–459 (2013)
31. Yuan, X.: Alternating direction method for covariance selection models. J. Sci. Comput. **51**(2), 261–273 (2012)
32. Fan, Q., Jiao, Y., Lu, X.: A primal dual active set algorithm with continuation for compressed sensing. IEEE Trans. Signal Process. **62**, 6276–6285 (2014)
33. Jiao, Y., Jin, B., Lu, X.: A primal dual active set with continuation algorithm for the ℓ^0-regularized optimization problem. Appl. Comput. Harmon. Anal. (2014). doi:10.1016/j.acha.2014.10.001
34. Chen, S.S., Donoho, D.L.: Atomic decomposition by basis pursuit. SIAM J Sci. Comput. **20**(1), 33–61 (1998)
35. Tibshirani, R.: Regression shrinkage and selection via the lasso. J. R. Stat. Soc. Ser. B (Methodological) **58**(1), 267–288 (1996)
36. Akaike, H.: A new look at the statistical model identification. IEEE Trans. Autom. Control **19**(6), 716–723 (1974)
37. Lu, Z., Zhang, Y.: Penalty decomposition methods for rank minimization. Research Paper, Department of Mathematics, Simon Fraser University. Available at http://people.math.sfu.ca/zhaosong/ResearchPapers/pd-rank-rev.pdf (2013)
38. Jin, Z.-F., Wang, Q., Wan, Z.: Recovering low-rank matrices from corrupted observations via the linear conjugate gradient algorithm. J. Comput. Appl. Math. **256**, 114–120 (2014)
39. Kelley, C.T.: Iterative Methods for Linear and Nonlinear Equations. Society for Industrial and Applied Mathematics, Philadelphia (1995)
40. Wen, Z., Yang, C., Liu, X., Marchesini, S.: Alternating direction methods for classical and ptychographic phase retrieval. Inverse Probl. **28**(11), 115010 (2012)
41. Wen, Z., Peng, X., Liu, X., Sun, X., Bai, X.: Asset allocation under the basel accord risk measures. arXiv preprint, arXiv:1308.1321 (2013)
42. Shen, Y., Wen, Z., Zhang, Y.: Augmented lagrangian alternating direction method for matrix separation based on low-rank factorization. Optim. Methods Softw. **29**(2), 239–263 (2014)
43. Malek-Mohammadi, M., Babaie-Zadeh, M., Amini, A., Jutten, C.: Recovery of low-rank matrices under affine constraints via a smoothed rank function. IEEE Trans. Signal Process. **62**(4), 981–992 (2014)
44. Larsen, R.M.: Propack-software for large and sparse svd calculations. Available online, http://sun.stanford.edu/rmunk/PROPACK (2004)