Adv Comput Math (2016) 42:1209–1224
DOI 10.1007/s10444-016-9459-y

A modified augmented lagrange multiplier algorithm for toeplitz matrix completion

Chuanlong Wang[1] · Chao Li[1] · Jin Wang[2]

Received: 11 September 2014 / Accepted: 22 March 2016 /
Published online: 2 April 2016

Abstract In this paper, a modified scheme is proposed for iterative completion matrices generated by the augmented Lagrange multiplier (ALM) method based on the mean value. So that the iterative completion matrices generated by the new algorithm are of the Toeplitz structure, which decrease the computation of SVD and have better approximation to solution. Convergence is discussed. Finally, the numerical experiments and inpainted images show that the new algorithm is more effective than the accelerated proximal gradient (APG) algorithm, the singular value thresholding (SVT) algorithm and the ALM algorithm, in CPU time and accuracy.

Keywords Toeplitz matrix · Matrix completion · Augmented Lagrange multiplier · Mean value

Mathematics Subject Classification (2010) 15A83 · 15A18 · 15B05 · 90C25

Communicated by: Raymond H. Chan

This work was supported by National Natural Science Foundation of China (Grant No. 11371275).

✉ Chuanlong Wang
clwang1964@163.com

Chao Li
l_chao1989@163.com

Jin Wang
wjyz91@163.com

1 Higher Education Key Laboratory of Engineering Science Computing in Shanxi Province, Taiyuan Normal University, Taiyuan 030012, China

2 School of Computer Science and Technology, Beihang University, Beijing 100191, People's Republic of China

1 Introduction

Recovering an unknown low-rank or approximately low-rank matrix from a sampling of its entries is a challenging problem arising in many real world, such as machine learning [1, 2], control [25], image inpainting [5], computer vision [30] and so on. This problem is well known as the matrix completion (MC) problem. Its typical application is the famous Netflix problem [4].

The MC problem was first proposed by Candès and Recht [9]. In [9] they showed that most low-rank matrices can be recovered from a sampling of its entries by solving the nuclear norm minimization problem,

$$\begin{aligned} \min \ & \|A\|_*, \\ \text{s.t.} \ & A_{ij} = M_{ij}, \ (i, j) \in \Omega. \end{aligned} \tag{1}$$

where $\|A\|_* = \sum_{k=1}^{r} \sigma_k(A), \sigma_k(A)$ denotes the k-th largest singular value of $A \in \mathbb{R}^{n_1 \times n_2}$ of rank r, $M \in \mathbb{R}^{n_1 \times n_2}$,and $\{M_{ij} : (i, j) \in \Omega\}$ are known, Ω is a random subset of cardinality m which is the number of sampled entries. They also proved that if m obeys, $m \geq Cn^{1.2} r \log n$, $n = max(n_1, n_2)$, for some positive numerical constant C, most matrices of rank r can be recovered with high probability. Then Candès and Tao [10], Keshavan, Montanari, and Oh [20], Recht [26] improved the bound.

Many algorithms have been proposed to solve the optimization problem (1). Fazel [12, 13] recast (1) as a semidefinite programming. Toh et al. [29] gave an accelerated proximal gradient (APG) algorithm. Ma and Goldfarb [24] proposed a new algorithm, which is called fixed point continuation with approximate SVD (FPCA) by adopting a Monte approximate SVD in the fixed point continuation algorithm (FPC) [16]. Almost simultaneously, Cai et al. [6] presented a singular value thresholding (SVT) algorithm, which has been shown to be simple and efficient for large low-rank matrix completion problems. Then the augmented Lagrange multiplier (ALM) algorithm proposed by Lin et al. [21] is proved to be considerably faster than the SVT algorithm and the APG algorithm under some conditions. Many other algorithms for MC problems, we can refer to [7, 8, 17] for details.

On the other hand, the Toeplitz matrix plays an important role in a wide variety of scientific and engineering fields, especially in signal and image processing [3, 15, 18]. Such as: in most medical imaging systems, the encoding matrix is a block Toeplitz matrix [27]; The blurring matrices constructed from the point spread functions for the reflexive boundary conditions have a Toeplitz structure [23]; In a special type of signal restoration problem where a limited number of sampling data are missing, when the missing samples are equidistant, the formulation related to samples of the function and its derivative leads to a possibly large linear system associated to a nonsymmetric block Toeplitz matrix [11]. Besides, some scholars have studied the approximation of Toeplitz and Hankel matrices [28]. A Toeplitz matrix $T \in \mathbb{R}^{n \times n}$ is of the form

$$T = \begin{pmatrix} t_0 & t_{-1} & \cdots & t_{-n+2} & t_{-n+1} \\ t_1 & t_0 & \cdots & t_{-n+3} & t_{-n+2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t_{n-2} & t_{n-3} & \cdots & t_0 & t_{-1} \\ t_{n-1} & t_{n-2} & \cdots & t_1 & t_0 \end{pmatrix},$$

which is determined by $2n - 1$ entries, that is the first row and first column. For the special structure of Toeplitz matrices, [19] presented an $O(n \log n)$ algorithm for the Toeplitz matrix-vector multiplication using the fast Fourier transform (FFT) [31], which contributes to the singular value decomposition (SVD) [14] of Toeplitz matrices by using the Lanczos method [14]. In [22, 32], Qiao et al. proposed an $O(n^2 \log n)$ algorithm for the SVD of Hankel matrices by using the Lanczos method, in contrast with existing $O(n^3)$ SVD algorithms. Since a Toeplitz matrix can be transformed into a Hankel matrix by reversing the columns or rows [19], we can straightforwardly obtain an $O(n^2 \log n)$ algorithm for computing the SVD of Toeplitz matrices. Since the efficiency of the algorithms mentioned above depends highly on the performance of the computation of SVD, it's significant to the study of Toeplitz matrix completion.

In this paper, we focus on the Toeplitz matrix completion problem. First, we give some definitions.

Definition 1 (Singular value decomposition (SVD) [14]) The singular value decomposition of a matrix $X \in \mathbb{R}^{n_1 \times n_2}$ of rank r is:

$$X = U \Sigma_r V^T, \quad \Sigma_r = diag(\sigma_1, \cdots, \sigma_r),$$

where $U \in \mathbb{R}^{n_1 \times r}$ and $V \in \mathbb{R}^{n_2 \times r}$ are column orthonormal matrices, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$.

Definition 2 (Singular value thresholding operator [6]) For each $\tau \geq 0$, the singular value thresholding operator $\mathcal{D}_\tau$ is defined as follows:

$$\mathcal{D}_\tau(X) := U \mathcal{D}_\tau(\Sigma) V^T, \quad \mathcal{D}_\tau(\Sigma) = diag(\{\sigma_i - \tau\}_+),$$

where $X = U \Sigma_r V^T \in \mathbb{R}^{n_1 \times n_2}$, $\{\sigma_i - \tau\}_+ = \begin{cases} \sigma_i - \tau, & if \quad \sigma_i > \tau \\ 0, & if \quad \sigma_i \leq \tau \end{cases}$.

The rest of this paper is organized as follows. In Section 2, we first sketch the previous work, then we give the proposed algorithm for Toeplitz matrices in details. Convergence analysis is given in Section 3. In Section 4, we compare our algorithm with the ALM algorithm, the SVT algorithm, and the APG algorithm through numerical experiments. Finally, we conclude the paper in Section 5.

Notation. For convenience, $\mathbb{R}^{n_1 \times n_2}$ denotes the set of $n_1 \times n_2$ real matrices, X_{ij} denotes the (i, j)th entry of a matrix X, the nuclear norm of a matrix X is denoted by $\|X\|_*$, the Frobenius norm by $\|X\|_F$. The transpose of a matrix X is X^T. The standard inner product between two matrices is denoted by $\langle X, Y\rangle$=trace$(X^T Y)$, $\Omega \subset \{-n_1+1, \cdots, n_2-1\}$ is the indices of observed diagonals of a matrix X, $\bar{\Omega}$ is the complementary set of Ω. Vector $diag(X, l)$ denotes the l-th diagonal of a Toeplitz matrix $X, l \in \{-n_1+1, \cdots, n_2-1\}$. P_Ω is the orthnogonal projector on Ω, satisfying, $diag(P_\Omega(X), l) = \begin{cases} diag(X, l), & l \in \Omega \\ 0, & l \notin \Omega \end{cases}$.

2 Algorithm

2.1 Previous algorithms for matrix compeltion

In this subsection, for completeness and comparison, we briefly introduce other existing algorithms for matrix completion.

2.1.1 The SVT approach

The SVT approach [6] solves the following convex optimization problem:

$$\begin{array}{ll} \min & \tau\|A\|_* + \frac{1}{2}\|A\|_F^2, \\ \text{s.t.} & \mathcal{P}_\Omega(A) = \mathcal{P}_\Omega(M). \end{array} \tag{2}$$

In [6], Cai et al. showed that when $\tau \to \infty$, the solution of (2) converges to the solution of (1).

Algorithm 1 (The SVT Approach)

Step 0: Set sampled set Ω, sampled matrix $\mathcal{P}_\Omega(M)$. Set parameters τ, δ, k_0, ϵ. Set initial matrix $Y_0 = k_0\delta\mathcal{P}_\Omega(M)$, $k := 0$;
Step 1: Compute the SVD of the matrix Y_k,

$$[U_k, \Sigma_k, V_k] = svd(Y_k);$$

Step 2: Set

$$A_{k+1} = U_k\mathcal{D}_\tau(\Sigma_k)V_k^T;$$

Step 3: If $\|\mathcal{P}_\Omega(A_{k+1} - M)\|_F / \|\mathcal{P}_\Omega(M)\|_F \le \epsilon$, stop; Otherwise, go to Step 4;
Step 4: $Y_{k+1} = Y_k + \delta\mathcal{P}_\Omega(M - A_{k+1})$,go to Step 1.

2.1.2 The APG approach

The APG approach [29] solves the following unconstrained convex problem:

$$\min\ F(A) = \mu\|A\|_* + \frac{1}{2}\|\mathcal{P}_\Omega(A - M)\|_F^2. \tag{3}$$

Algorithm 2 (The APG Approach)

Step 0: Set parameters $\mu_0 > 0$, $\bar{\mu} > 0$, L_f, $\eta \in (0, 1)$. Set $A_0 = A_{-1} = 0$, $Y_0 = 0$, $t_0 = t_{-1} = 1$, $k := 0$;
Step 1: Compute the SVD of the matrix $G_k = Y_k - \frac{1}{L_f}(Y_k - \mathcal{P}_\Omega(M))$,

$$[U_k, \Sigma_k, V_k] = svd(G_k);$$

Step 2: Set

$$A_{k+1} = U_k \mathcal{D}_{\frac{\mu_k}{L_f}}(\Sigma_k) V_k^T;$$

Step 3: If $\|A_{k+1} - A_k\|_F / \|A_{k+1}\|_F < \epsilon$,stop; Otherwise, go to Step 4;
Step 4: Compute $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$, $\mu_{k+1} = max(\eta\mu_k, \bar{\mu})$; Set $Y_k = A_k + \frac{t_{k-1}-1}{t_k}(A_k - A_{k-1})$,go to Step 1.

2.1.3 The ALM approach

The ALM approach [21] solves the following convex optimization problem:

$$\begin{aligned} &\min \ \|A\|_*, \\ &\text{s.t.} \ \ A + E = D, \ \mathcal{P}_\Omega(E) = 0. \end{aligned} \tag{4}$$

The augmented Lagrangian function:

$$\mathcal{L}(A, E, Y, \mu) = \|A\|_* + \langle Y, D - A - E\rangle + \frac{\mu}{2}\|D - A - E\|_F^2,$$

where $Y \in \mathbb{R}^{n_1 \times n_2}$.

Algorithm 3 (The ALM Approach)

Step 0: Set sampled set Ω, sampled matrix D. Set parameters $\mu_0 > 0$, $\rho > 1$. Set initial matrices $Y_0 = 0$, $E_0 = 0$, $k := 0$;
Step 1: Compute the SVD of the matrix $(D - E_k + \mu_k^{-1} Y_k)$,

$$[U_k, \Sigma_k, V_k] = svd(D - E_k + \mu_k^{-1} Y_k);$$

Step 2: Set

$$A_{k+1} = U_k \mathcal{D}_{\mu_k^{-1}}(\Sigma_k) V_k^T,$$

$$E_{k+1} = P_{\bar{\Omega}}(D - A_{k+1} + \mu_k^{-1} Y_k);$$

Step 3: If $\|D - A_{k+1} - E_{k+1}\|_F / \|D\|_F < \epsilon_1$ and $\mu_k \|E_{k+1} - E_k\|_F / \|D\|_F < \epsilon_2$, stop; Otherwise, go to Step 4;
Step 4: Set $Y_{k+1} = Y_k + \mu_k(D - A_{k+1} - E_{k+1})$.
If $\mu_k \|E_{k+1} - E_k\|_F / \|D\|_F < \epsilon_2$, set $\mu_{k+1} = \rho\mu_k$; Otherwise, go to Step 1.

2.2 The method of modified augmented Lagrange multiplier

In this section, we focus on the Toeplitz matrix completion problem. Based on the special structure and properties of Toeplitz matrices, we use a mean-value technique in the original ALM algorithm. Therefore, the iterative matrices keep a Toeplitz structure that ensure the fast SVD of Toeplitz matrices. Then our problem is expressed as the following convex programming,

$$\begin{aligned} &\min\ \|A\|_*, \\ &\text{s.t.}\ \ A + E = D,\ \mathcal{P}_\Omega(E) = 0. \end{aligned} \tag{5}$$

where A, $D = \mathcal{P}_\Omega(M) \in \mathbb{R}^{n\times n}$ are Toeplitz matrices (M is the real Toeplitz matrix), $\Omega \subset \{-n+1, \cdots, n-1\}$.

Then the partial augmented Lagrangian function of (5) is

$$L(A, E, Y, \mu) = \|A\|_* + \langle Y, D - A - E\rangle + \frac{\mu}{2}\|D - A - E\|_F^2, \tag{6}$$

where $Y \in \mathbb{R}^{n\times n}$. Let

$$R_l = (r_{ij})_{n\times n} = \begin{cases} 1, & i - j = l \\ 0, & i - j \neq l \end{cases}, l = -n+1, \cdots, n-1. \tag{7}$$

Algorithm 4 (The modified augmented Lagrange multiplier (MALM) algorithm for Toeplitz matrix completion)

Step 0: Set sampled set Ω, sampled matrix D. Set parameters $\mu_0 > 0$, $\rho > 1$. Set initial matrices $Y_0 = 0$, $E_0 = 0$, $k := 0$;

Step 1: Compute the SVD of the matrix $(D - E_k + \mu_k^{-1} Y_k)$ using the Lanczos method

$$[U_k, \Sigma_k, V_k] = lansvd(D - E_k + \mu_k^{-1} Y_k);$$

Step 2: Set

$$X_{k+1} = U_k \mathcal{D}_{\mu_k^{-1}}(\Sigma_k) V_k^T,$$

compute $a_l = mean(diag(X_{k+1}, l))$, $l \in \{-n+1, \cdots, n-1\}$,

set

$$A_{k+1} = \sum_{l\in\{-n+1,\cdots,n-1\}} a_l R_l,$$

$$E_{k+1} = \mathcal{P}_{\bar{\Omega}}(D - A_{k+1} + \mu_k^{-1} Y_k);$$

Step 3: If $\|D - A_{k+1} - E_{k+1}\|_F / \|D\|_F < \epsilon_1$ and $\mu_k \|E_{k+1} - E_k\|_F / \|D\|_F < \epsilon_2$, stop; Otherwise, go to Step 4;

Step 4: Set $Y_{k+1} = Y_k + \mu_k(D - A_{k+1} - E_{k+1})$. If $\mu_k \|E_{k+1} - E_k\|_F / \|D\|_F < \epsilon_2$, set $\mu_{k+1} = \rho\mu_k$; Otherwise, go to Step 1.

Table 1 Comparison of algorithmic complexity between MALM, ALM, APG, and SVT on main process

algorithm \ process	SVD	mean value
MALM	$O(r_k n \log n)$	$O(n^2 - 2)$
ALM	$O(r_k n^2)$	0
APG	$O(r_k n^2)$	0
SVT	$O(r_k n^2)$	0

3 Convergence

In this section, we analyze the convergence of Algorithm 4.

Lemma 1 *The sequence* $\{Y_k\}$ *is bounded.*

Proof Let $B = \mu_k(D - E_k + \mu_k^{-1} Y_k - X_{k+1})$, $E(B) = (\bar{b}_{ij}) = \sum\limits_{l \in \Omega} b_l R_l$, where $b_l = mean(diag(B, l))$, R_l is defined as (7).

Table 2 Comparison between MALM, ALM, APGL, and SVT on the MC problem

size(n)	rank(r)	p	algorithm	#iter	time(s)	t(SVD)	$\frac{\|A^* - M\|_F}{\|M\|_F}$
500	10	0.35	MALM	43	3.2775	0.652	4.1735e-09
			ALM	37	20.1378	17.615	0.0040
			APGL	93	9.7610	2.995	3.8521e-04
			SVT	469	12.8218	6.779	0.0046
800	10	0.35	MALM	50	7.6157	1.040	8.6439e-09
			ALM	47	37.3529	21.172	0.0107
			APGL	334	89.5870	23.504	9.7338e-04
			SVT	847	57.3956	25.483	0.0122
1000	10	0.35	MALM	59	13.3861	2.274	2.1027e-08
			ALM	52	42.4018	27.457	7.2572e-04
			APGL	74	27.7341	6.095	2.2396e-04
			SVT	544	54.8488	22.741	0.0040
1500	10	0.35	MALM	65	29.8996	3.031	3.4911e-05
			ALM	61	129.9868	91.067	0.0119
			APGL	566	514.0216	107.957	0.0013
			SVT	1382	301.0895	112.427	0.0258
2000	10	0.35	MALM	68	46.2109	3.290	1.6951e-09
			ALM	69	159.6407	81.728	6.6649e-05
			APGL	65	93.2820	17.594	2.9535e-04
			SVT	397	147.7427	53.815	0.0027
2500	10	0.35	MALM	71	75.7758	7.380	2.7159e-09
			ALM	75	276.4907	145.738	2.7068e-04
			APGL	62	139.1909	26.705	3.8564e-04
			SVT	787	458.4402	167.491	9.7556e-04
3000	10	0.35	MALM	73	109.0327	8.360	6.1703e-09
			ALM	77	417.3302	197.962	0.0043
			APGL	206	716.4112	137.064	3.8815e-04
			SVT	1102	937.1803	345.818	0.0117

Corresponding to each triplet $\{n, r, m/(2n-1)\}$, the MC problem was solved for the same data matrices M and $P_\Omega(M)$ using the four different algorithms

First, we prove that Y_k, E_k, $k = 1, 2, \cdots$,are Toeplitz matrices. From Algorithm 4, $Y_0 = 0$, $E_0 = 0$ are Toeplitz matrices. Assume that Y_k, E_k are Toeplitz matrices, then $E_{k+1} = \mathcal{P}_{\bar{\Omega}}(D - A_{k+1} + \frac{1}{\mu_k} Y_k)$ is a Toeplitz matrix. Thus, Y_{k+1} is also a Toeplitz matrix by Step 4 in Algorithm 4.

$$\begin{aligned} Y_{k+1} &= Y_k + \mu_k(D - A_{k+1} - E_{k+1}) \\ &= Y_k + \mu_k(D - A_{k+1} - E_k) + \mu_k(E_k - E_{k+1}). \end{aligned}$$

Table 3 Comparison between MALM, ALM, APGL, and SVT on the MC problem

size(n)	rank(r)	p	algorithm	#iter	time(s)	t(SVD)	$\frac{\Vert A^* - M\Vert_F}{\Vert M\Vert_F}$
500	10	0.4	MALM	43	3.8063	1.220	2.9960e-06
			ALM	39	24.8547	22.153	1.7350
			APGL	195	22.5710	6.254	5.5711e-04
			SVT	1107	31.5542	15.469	0.0066
800	10	0.4	MALM	50	7.4551	1.200	4.4314e-06
			ALM	47	36.5402	27.236	0.0094
			APGL	585	166.7816	39.607	7.6463e-04
			SVT	750	53.2880	22.579	0.0119
1000	10	0.4	MALM	56	12.9775	2.886	1.9074e-05
			ALM	51	49.0338	31.075	0.0015
			APGL	141	59.1304	12.435	2.1702e-04
			SVT	702	73.5963	27.680	0.0040
1500	10	0.4	MALM	62	26.2751	3.620	3.3785e-08
			ALM	62	99.1826	61.486	0.0038
			APGL	305	288.1202	56.227	9.0188e-04
			SVT	1388	345.2863	110.972	0.0175
2000	10	0.4	MALM	68	50.6570	3.840	2.3823e-06
			ALM	67	117.6600	75.754	1.1095e-04
			APGL	72	110.8651	19.252	3.2971e-04
			SVT	520	209.0315	70.837	0.0023
2500	10	0.4	MALM	71	73.1580	5.900	1.4313e-08
			ALM	74	213.1340	90.839	0.0013
			APGL	190	480.2522	85.495	4.9145e-04
			SVT	840	514.7787	176.139	0.0062
3000	10	0.4	MALM	69	97.2530	7.410	1.1861e-08
			ALM	76	341.9672	128.716	0.0042
			APGL	266	962.8713	173.708	5.9774e-04
			SVT	880	777.9870	274.509	0.0095

Corresponding to each triplet $\{n, r, m/(2n-1)\}$, the MC problem was solved for the same data matrices M and $P_\Omega(M)$ using the four different algorithms

Computing $\mu_k(E_k - E_{k+1})$,

$$\begin{aligned}
\mu_k(E_k - E_{k+1}) &= \mu_k \mathcal{P}_{\bar{\Omega}}(E_k - (D - A_{k+1} + \frac{1}{\mu_k} Y_k)) \\
&= \mu_k \mathcal{P}_{\bar{\Omega}}(E_k - (D - E(X_{k+1}) + \frac{1}{\mu_k} Y_k) \\
&= \mu_k \mathcal{P}_{\bar{\Omega}}(E_k - (D - E(D - E_k + \frac{1}{\mu_k} Y_k - \frac{1}{\mu_k} B) + \frac{1}{\mu_k} Y_k)) \\
&= \mu_k \mathcal{P}_{\bar{\Omega}}(E_k - (D - (D - E_k + \frac{1}{\mu_k} Y_k - \frac{1}{\mu_k} E(B)) + \frac{1}{\mu_k} Y_k)) \\
&= \mathcal{P}_{\bar{\Omega}}(E(B)).
\end{aligned}$$

By Step 1 and Step 2 in Algorithm 4, we have

$$D - E_k + \mu_k^{-1} Y_k = \bar{U}_k \bar{\Sigma}_k \bar{V}_k^T + \tilde{U}_k \tilde{\Sigma}_k \tilde{V}_k^T$$

where $\bar{U}_k, \bar{V}_k$ (resp. $\tilde{U}_k, \tilde{V}_k$) are the singular vectors associated with singular values greater than μ_k^{-1} (resp. smaller than or equal to μ_k^{-1}), the diagonal elements of $\bar{\Sigma}_k$ (resp. $\tilde{\Sigma}_k$) are greater than μ_k^{-1} (resp. smaller than or equal to μ_k^{-1}). Then, we have $X_{k+1} = \bar{U}_k(\bar{\Sigma}_k - \mu_k^{-1} I)\bar{V}_k^T$. Thus,

$$\begin{aligned}
\|B\|_F &= \|\mu_k(D - E_k + \mu_k^{-1} Y_k - X_{k+1})\|_F \\
&= \|\mu_k(\mu_k^{-1} \bar{U}_k \bar{V}_k^T + \tilde{U}_k \tilde{\Sigma}_k \tilde{V}_k^T)\|_F \\
&= \|\bar{U}_k \bar{V}_k^T + \mu_k \tilde{U}_k \tilde{\Sigma}_k \tilde{V}_k^T\|_F \\
&\le \sqrt{n}.
\end{aligned}$$

From lemma 1 and lemma 4 in [21], we have $Y_k + \mu_k(D - A_{k+1} - E_k) \in \partial \|A_{k+1}\|_*$. Let $A_{k+1} = U \Sigma V^T$,it is known [6] that

$$\partial \|A_{k+1}\|_* = \{UV^T + W : W \in \mathbb{R}^{n \times n}, U^T W = 0, WV = 0, \|W\|_2 \le 1\}.$$

Thus,

$$\|UV^T + W\|_F^2 = trace((UV^T + W)^T(UV^T + W)) = trace(VV^T + W^T W) \le n.$$

Therefore,

$$\|Y_k + \mu_k(D - A_{k+1} - E_k)\|_F \le \sqrt{n},$$

$$\|\mathcal{P}_{\bar{\Omega}}(E(B))\|_F \le \|E(B)\|_F \le \|B\|_F \le \sqrt{n}.$$

So the sequence $\{Y_k\}$ is bounded. □

Theorem 1 *If $\mu_k \to \infty$ and $\sum_{k=1}^{\infty} \mu_k^{-1} = +\infty$, then the sequence $\{A_k\}$ converges to the solution of (5).*

Proof Since $\mu_k^{-1}(Y_{k+1} - Y_k) = D - A_{k+1} - E_{k+1}$, by Lemma 1, we have

$$\lim_{k \to \infty} (D - A_{k+1} - E_{k+1}) = 0.$$

Let (A^*, E^*) be the solution of (5), since $A^* + E^* = D$, A_{k+1}, Y_{k+1}, E_{k+1} are Toeplitz matrices for $k = 1, 2, \cdots$,we first prove that

$$\begin{aligned} &\|E_{k+1} - E^*\|_F^2 + \mu_k^{-2}\|Y_{k+1} - Y^*\|_F^2 \\ = &\|E_k - E^*\|_F^2 - \|E_{k+1} - E_k\|_F^2 + \mu_k^{-2}\|Y_k - Y^*\|_F^2 - \mu_k^{-2}\|Y_{k+1} - Y_k\|_F^2 \\ &-2\mu_k^{-1}\langle A_{k+1} - A^*, \hat{Y}_{k+1} - Y^*\rangle, \end{aligned} \tag{8}$$

Table 4 Comparison between MALM, ALM, APGL, and SVT on the MC problem

size(n)	rank(r)	p	algorithm	#iter	time(s)	t(SVD)	$\frac{\|A^*-M\|_F}{\|M\|_F}$
500	10	0.5	MALM	42	3.1901	0.825	2.3564e-07
			ALM	40	24.1743	18.791	2.9903
			APGL	187	23.3999	5.886	3.7470e-04
			SVT	823	25.5050	11.911	0.0068
800	10	0.5	MALM	54	8.5332	1.855	3.5990e-05
			ALM	47	29.9279	25.017	0.0011
			APGL	588	186.9982	40.257	7.2905e-04
			SVT	671	52.8481	20.336	0.0169
1000	10	0.5	MALM	55	11.8331	2.094	2.1442e-08
			ALM	52	34.1414	29.075	4.5571e-04
			APGL	106	48.5309	9.169	2.0680e-04
			SVT	691	82.2254	29.749	0.0036
1500	10	0.5	MALM	62	25.4994	3.070	3.1976e-07
			ALM	61	66.5100	33.278	0.0091
			APGL	370	387.1082	68.600	7.9480e-04
			SVT	1293	329.4267	104.797	0.0211
2000	10	0.5	MALM	67	43.7179	3.150	1.0703e-07
			ALM	67	99.9049	37.346	1.1530e-04
			APGL	66	112.4825	17.213	2.8046e-04
			SVT	355	158.2011	50.461	0.0027
2500	10	0.5	MALM	70	73.0632	5.710	3.2989e-08
			ALM	72	158.1567	54.187	5.4778e-04
			APGL	108	299.4840	48.059	3.1326e-04
			SVT	576	397.4815	124.700	0.0066
3000	10	0.5	MALM	69	96.9950	7.470	1.1860e-08
			ALM	75	293.9793	128.716	0.0017
			APGL	153	614.7142	95.706	4.7547e-04
			SVT	550	543.5487	177.879	0.0083

Corresponding to each triplet $\{n, r, m/(2n-1)\}$, the MC problem was solved for the same data matrices M and $P_\Omega(M)$ using the four different algorithms

where $\hat{Y}_{k+1} = Y_k + \mu_k(D - A_{k+1} - E_k)$, Y^* is the optimal solution to the dual problem (9) in [21].

$$\begin{aligned} \|E_k - E^*\|_F^2 &= \|\mathcal{P}_{\bar{\Omega}}(E_k - E^*)\|_F^2 \\ &= \|\mathcal{P}_{\bar{\Omega}}(E_{k+1} - E^* - E_{k+1} + E_k)\|_F^2 \\ &= \|\mathcal{P}_{\bar{\Omega}}(E_{k+1} - E^*)\|_F^2 + \|\mathcal{P}_{\bar{\Omega}}(E_{k+1} - E_k)\|_F^2 \\ &\quad -2\langle \mathcal{P}_{\bar{\Omega}}(E_{k+1} - E^*), \mathcal{P}_{\bar{\Omega}}(E_{k+1} - E_k)\rangle \\ &= \|E_{k+1} - E^*\|_F^2 + \|E_{k+1} - E_k\|_F^2 \\ &\quad +2\mu_k^{-1}\langle \mathcal{P}_{\bar{\Omega}}(A_{k+1} - A^*), \hat{Y}_{k+1} - Y^*\rangle. \end{aligned}$$

Table 5 Comparison between MALM, ALM, APGL, and SVT on the MC problem

size(n)	rank(r)	p	algorithm	#iter	time(s)	t(SVD)	$\frac{\|A^*-M\|_F}{\|M\|_F}$
500	10	0.6	MALM	41	3.2158	0.698	1.6253e-09
			ALM	38	15.8501	8.917	0.0012
			APGL	35	4.4266	0.889	3.8879e-04
			SVT	361	12.7537	5.269	0.0027
800	10	0.6	MALM	50	6.9207	1.075	9.5633e-09
			ALM	47	16.3682	10.561	0.0060
			APGL	301	105.8264	20.818	4.1101e-04
			SVT	753	64.4577	22.929	0.0079
1000	10	0.6	MALM	54	11.0180	1.776	9.0936e-09
			ALM	51	27.2914	18.116	2.6971e-04
			APGL	78	38.4526	6.362	2.0277e-04
			SVT	302	37.7752	12.063	0.0030
1500	10	0.6	MALM	58	22.5574	3.020	3.7890e-09
			ALM	59	59.2528	35.210	0.0028
			APGL	199	221.6222	35.493	5.6397e-04
			SVT	399	108.9084	34.096	0.0144
2000	10	0.6	MALM	68	41.9135	3.310	1.7307e-09
			ALM	64	103.1202	57.796	5.3243e-05
			APGL	58	109.2345	15.240	2.1368e-04
			SVT	160	74.8539	21.463	0.0017
2500	10	0.6	MALM	70	66.4277	6.090	2.4727e-09
			ALM	69	172.2073	95.162	1.5696e-04
			APGL	62	184.0663	26.622	2.8670e-04
			SVT	627	456.2409	133.465	0.0011
3000	10	0.6	MALM	68	91.9454	7.590	1.1492e-08
			ALM	74	296.4603	177.771	4.0690e-04
			APGL	145	649.3658	93.785	3.5473e-04
			SVT	464	481.4966	142.546	0.0067

Corresponding to each triplet $\{n, r, m/(2n-1)\}$, the MC problem was solved for the same data matrices M and $P_\Omega(M)$ using the four different algorithms

We obtain the following results with the same analysis,

$$\begin{aligned}\mu_k^{-2}\|Y_k - Y^*\|_F^2 &= \mu_k^{-2}\|\mathcal{P}_\Omega(Y_k - Y^*)\|_F^2 \\ &= \mu_k^{-2}\|\mathcal{P}_\Omega(Y_{k+1} - Y^*)\|_F^2 + \mu_k^{-2}\|\mathcal{P}_\Omega(Y_{k+1} - Y_k)\|_F^2 \\ &\quad -2\mu_k^{-1}\langle \mathcal{P}_\Omega(Y_{k+1} - Y^*), \mathcal{P}_\Omega(Y_{k+1} - Y_k)\rangle \\ &= \mu_k^{-2}\|Y_{k+1} - Y^*\|_F^2 + \mu_k^{-2}\|Y_{k+1} - Y_k\|_F^2 \\ &\quad +2\mu_k^{-1}\langle \mathcal{P}_\Omega(A_{k+1} - A^*), \hat{Y}_{k+1} - Y^*\rangle.\end{aligned}$$

So (8) holds.

Since $\|A\|_*$ is a convex function and $\hat{Y}_{k+1} \in \partial\|A_{k+1}\|_*$, $\langle A_{k+1} - A^*, \hat{Y}_{k+1} - Y^*\rangle \geq 0$, so $\sum_{k=1}^{\infty} \mu_k^{-1}\langle A_k - A^*, \hat{Y}_k - Y^*\rangle < +\infty$, and $\|E_k - E^*\|^2 + \mu_k^{-2}\|Y_k - Y^*\|^2$ is non-increasing. On the other hand, from Algorithm 4,

$$\langle Y_k, E_k - E^*\rangle = 0,$$

$$\langle Y^*, E_k - E^*\rangle = 0.$$

Thus, using the same proof of Theorem 2 in [21], we obtain that A^* is the solution of (5). □

Theorem 2 *Let $X = (x_{ij}) \in \mathbb{R}^{n\times n}$. $E(X) = (\bar{x}_{ij})$ is a Toeplitz matrix with $E(X) = \sum_{l\in\{-n+1,\cdots,n-1\}} a_l R_l$, where $a_l = mean(diag(X, l))$, $l \in \{-n+1, \cdots, n-1\}$, R_l is defined as (7). If Y is a Toeplitz matrix, then $\langle X - E(X), Y\rangle = 0$.*

Proof By the definition of $E(X)$, we have $\sum_{i-j=l} (x_{ij} - \bar{x}_{ij}) = 0$, $l = -n+1, \cdots, n-1$. Since Y is a Toeplitz matrix, and $y_l = y_{ij}$, $l = i - j$, $i, j = 1, 2, \cdots, n$. So

$$\begin{aligned}\langle X - E(X), Y\rangle &= \sum_{i=1}^{n}\sum_{j=1}^{n}(x_{ij} - \bar{x}_{ij})y_{ji} \\ &= \sum_{l=-n+1}^{n-1} (y_l \sum_{i-j=l} (x_{ij} - \bar{x}_{ij})) \\ &= 0.\end{aligned}$$

□

Theorem 3 *In Algorithm 4, A_k is a Toeplitz matrix generated by X_k. Then*

$$\|A_k - A^*\|_F < \|X_k - A^*\|_F,$$

where A^* is the solution of (5).

Proof

$$
\begin{aligned}
\|X_k - A^*\|_F^2 &= \|X_k - A_k + A_k - A^*\|_F^2 \\
&= \langle X_k - A_k, X_k - A_k\rangle + 2\langle X_k - A_k, A_k - A^*\rangle \\
&\quad + \langle A_k - A^*, A_k - A^*\rangle \\
&= \|X_k - A_k\|_F^2 + \|A_k - A^*\|_F^2.
\end{aligned}
$$

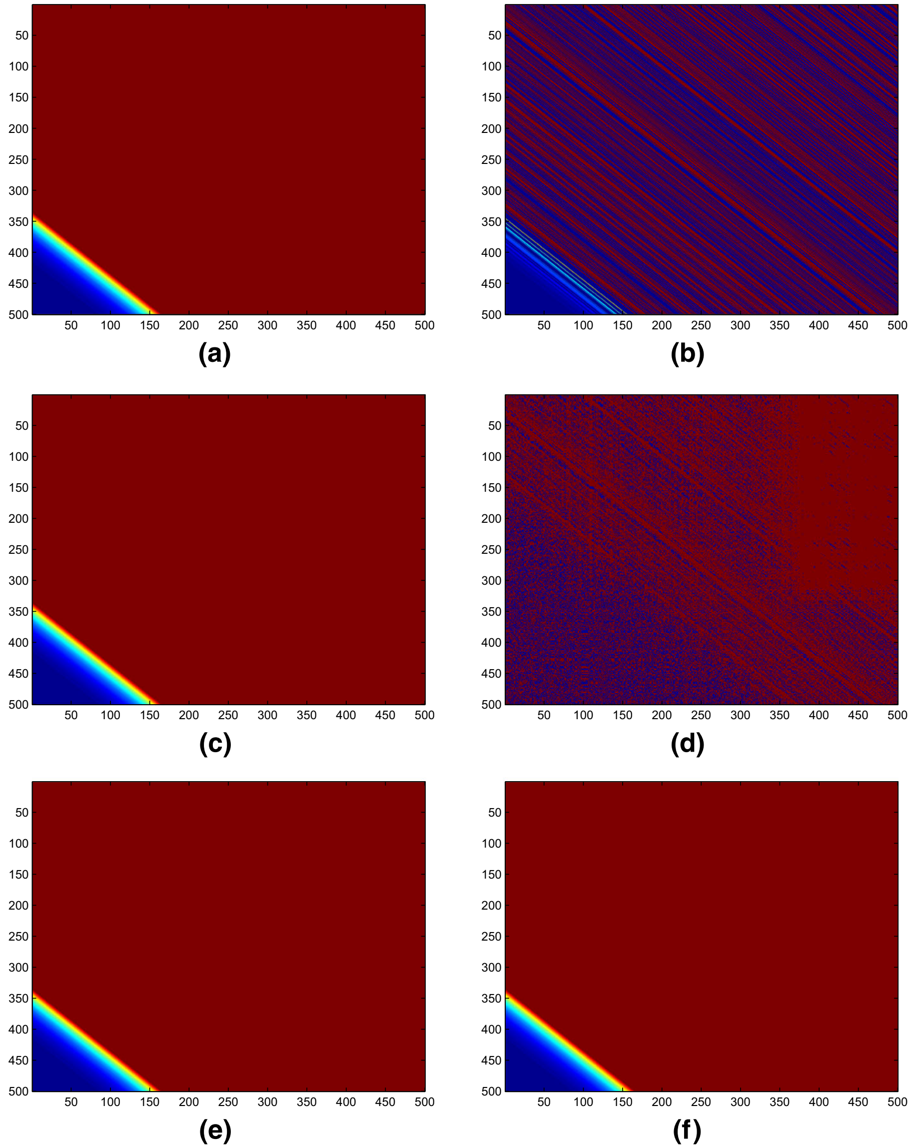

Fig. 1 **a** original image, **b** damaged image, **c** inpainted image by the MALM algorithm, **d** inpainted image by the ALM algorithm, **e** inpainted image by the APGL algorithm, **f** inpainted image by the SVT algorithm

So $\|A_k - A^*\|_F < \|X_k - A^*\|_F$. □

4 Numerical experiments

In this section, we first show a brief comparison of the algorithmic complexity on the main process of each algorithm in Table 1, where r_k is the number of singular values in the kth iteration. Then through numerical experiments and image inpainting, we evaluate the performance of our MALM algorithm with the ALM algorithm, the APGL algorithm (APG with line search), and the SVT algorithm. All the experiments are conducted on the same workstation.

In the experiments, we suggest $p = m/(2n-1)$ as the sampling density, where m is the number of observed entries. Since the special structure of Toeplitz, we have $0 \le m \le 2n-1$. For Algorithm 4, we empirically set the parameters $\tau_0 = 1/\|D\|_2$, $\rho = 1.2172 + 1.8588p$, $\epsilon_1 = 10^{-9}$, $\epsilon_2 = 5 \times 10^{-6}$. The choosing of parameters for ALM are suggested in [21], except the parameters of the stopping criteria are $\epsilon_1 = 10^{-9}$ and $\epsilon_2 = 5 \times 10^{-6}$. For Algorithm 1, we set $\tau = \|\mathcal{P}_\Omega(M)\|_2/2$, $\epsilon = 5 \times 10^{-4}$, $\delta = 1.89$, k_0 is an integer obeying $\frac{\tau}{\delta\|\mathcal{P}_\Omega(M)\|_2} \in (k_0 - 1, k_0]$. For Algorithm 2, we set $\mu_0 = \|\mathcal{P}_\Omega(M)\|_2$, $\bar{\mu} = 10^{-4}\mu_0$, $\eta = 0.8$, $L_f = 1$, $\epsilon = 1 \times 10^{-5}$.

A brief comparison of the four algorithms is presented in Tables 2,3,4,5 where t(SVD) denotes the time of SVD, A^* denotes the output of the four algorithms, M denotes the real Toeplitz matrix. We can see that our algorithm performs well, while the other algorithms only converge for individual experiments. Besides, we apply our algorithms to simply image processing using one of the experiment data. We demonstrate this simulation in Fig. 1.[1]

5 Conclusion

In this paper, we develop a modified augmented Lagrange multiplier algorithm for Toeplitz matrix completion. Compared to the ALM, APGL and SVT algorithms, our algorithm performs better that of much higher accuracy. Throughout the iterative process, the completion matrices keep the Toeplitz structure that ensure the fast SVD of Toeplitz matrices. From Tables 2-5, we can see our algorithm is advantageous over the other three algorithms on the time of SVD. Besides, under the same parameters, the ALM dose not converge to the exact solution of the problem well. Meanwhile, the completion matrices of the other algorithms are not keeping the Toeplitz structure.

References

1. Amit, Y., Fink, M., Srebro, N., Ullman, S.: Uncovering shared structures in multiclass classification. In: Processings of the 24th International Conference on Machine Learing, pp. 17–24. ACM (2007)

[1] After zooming in, a greater difference between each subfigure can be seen.

2. Argyriou, A., Evgeniou, T., Pontil, M.: Multi-task feature learing. Adv. Neural Inf. Process. Syst. **19**, 41–48 (2007)
3. Bajwa, W.U., Haupt, J.D., Raz, G.M., Wright, S.J., Nowak, R.D.: Toeplitz-structures compressed sensing matrices. Statistical signal Processing, 2007, SSP'07, IEEE/SP 14th Workshop on 294–298 (2007)
4. Bennett, J., Lanning, S.: The netflix prize. In: proceedings of KDD Cup and Workshop (2007)
5. Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. Comput. Grapher, (SIGGRAPH) **2000**(34), 417–424 (2000)
6. Cai, J.F., Candès, E.J., Shen, Z.: A singular value thresholding algorithm for matrix completion. SIAM J. Optim. **20**, 1956–1982 (2010)
7. Cai, J.F., Osher, S.: Fast singular value thresholding without singular value decomposition. Methods Appl. Anal. **20**, 335–352 (2013)
8. Candès, E.J., Plan, Y.: Matrix completion with noise. In: Proceedings of the IEEE (2009)
9. Candès, E.J., Recht, B.: Exact matrix completion via convex optimization. Found. Comput. Math. **9**, 717–772 (2009)
10. Candès, E.J., Tao, T.: The power of convex relaxation: Near-optimal matrix completion. IEEE Trans. Inf. Theory **56**, 2053–2080 (2010)
11. Del Prete, V., Di Benedetto, F., Donatelli, M., Serra-Capizzano, S.: Symbol approach in a signal-restoration problem involving block Toeplitz matrices, vol. 272 (2014)
12. Fazel, M.: Matrix Rank Minimization with Applications. PhD thesis, Stanford University (2002)
13. Fazel, M., Hindi, H., Boyd, S.P.: Log-det heuristic for matrix rank minimization with applications to Hankel and Euclidean distance matrices. In: proceedings of the American Control Conference 3, pp. 2156–2162 (2003)
14. Golub, G.H., Van Loan, C.F. Matrix Computations, 3rd edn. The Johns Hopkins University Press, Baltimore, MD (1996)
15. Gutiérrez-Gutiérrez, J., Crespo, P.M., Böttcher, A.: Functions of banded Hermitian block Toeplitz matrices in signal processing. Linear Algebra Appl. **422**, 788–807 (2007)
16. Hale, E.T., Yin, W., Zhang, Y.: Fixed-point continuation for l_1-minimization: methodology and convergence. SIAM J. Optim. **19**, 1107–1130 (2008)
17. Hu, Y., Zhang, D.B., Liu, J., Ye, J.P., He, X.F.: Accelerated singular value thresholding for matrix completion. In: KDD'12: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 298–306 (2012)
18. Huang, J., Huang, T.Z., Zhao, X.L., Xu, Z.B.: Image restoration with shifting reflective boundary conditions. Sci. China Inf. Sci. **56**(6), 1–15 (2013)
19. Kailath, T., Sayed, A.H. (eds.): Fast Reliable Algorithms for Matrices with Structure. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA (1999)
20. Keshavan, R.H., Montanari, A., Oh, S.: Matrix completion from a few entries. IEEE Trans Inf Theory **56**, 2980–2998 (2010)
21. Lin, Z., Chen, M., Wu, L., Ma, Y.: The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices UIUC Technicial Report UIUL-ENG-09-2214 (2010)
22. Luk, F.T., Qiao, S.: A Fast Singular Value Algorithm for Hankel Matrices. In: Olshevsky, V. (ed.) Fast Algorithms for Structured Matrices: Theory and Applications, Contemporary Mathematics, Vol. 323, American Mathematical Society (2003)
23. Lv, X.G., Huang, T.Z., Xu, Z.B., Zhao, X.L.: Kronecker product approximations for image restoration with whole-sample symmetric boundary conditions. Inf. Sci. **186**, 150C163 (2012)
24. Ma, S., Goldfarb, D., Chen, L.: Fixed point and Bregman iterative methods for matrix rank minimization. Math. Program. **128**, 321–353 (2011)
25. Mesbahi, M., Papavassilopoulos, G.P.: On the rank minimization problem over a positive semidefinite linear matrix inequality. IEEE Trans. Autom. Control **42**, 239–243 (1997)
26. Recht, B.: A simpler approach to matrix completion. J. Mach. Learing Res. **12**, 3413–3430 (2011)
27. Sebert, F., Zhou, Y.M., Ying, L.: Toeplitz block matrices in compressed sensing and their applications in imaging. In: International Conference on Information Technology and Applications in biomedicine(ITAB), pp. 47–50. IEEE (2008)
28. Shaw, A.K., Pokala, S., Kumaresan, R.: Toeplitz and Hankel approximation using structured approach. Acoust., Speech Signal Process., Process. IEEE Int. Conf. **2349**, 12–15 (1998)
29. Toh, K.C., Yun, S.: An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems. Pac. J. Optim. **6**, 615–640 (2010)

30. Tomasi, C., Kanade, T.: Shape and motion from image streams under orthography: a factorization method. Int. J. Comput. Vis. **9**, 137–154 (1992)
31. Van Loan, C.F.: Computational frameworks for the fast fourier transform society for industrial and applied mathematics (SIAM) (1992)
32. Xu, W., Qiao, S.: A fast SVD algorithm for square Hankel matrices. Linear Algebra Appl. **428**, 550–563 (2008)