Journal of Industrial and Management Optimization
Vol. 19, No. 4, April 2023, pp. 2354–2366
doi:10.3934/jimo.2022045

LOW RANK MATRIX MINIMIZATION WITH A TRUNCATED DIFFERENCE OF NUCLEAR NORM AND FROBENIUS NORM REGULARIZATION

Huiyuan Guo, Quan Yu, Xinzhen Zhang* and Lulu Cheng

School of Mathematics, Tianjin University,Tianjin 300350, China

(Communicated by Liqun Qi)

Abstract. In this paper, we present a novel regularization with a truncated difference of nuclear norm and Frobenius norm of form $L_{t,*-\alpha F}$ with an integer t and parameter α for rank minimization problem. The forward-backward splitting (FBS) algorithm is proposed to solve such a regularization problem, whose subproblems are shown to have closed-form solutions. We show that any accumulation point of the sequence generated by the FBS algorithm is a first-order stationary point. In the end, the numerical results demonstrate that the proposed FBS algorithm outperforms the existing methods.

1. **Introduction.** Rank minimization problems have wide applications in science and engineering [1, 3, 10]. In this paper, we focus on the following nonconvex nonsmooth rank minimization problem:

$$\min_{X\in\mathbb{R}^{m\times n}} r(X)+l(X), \tag{1.1}$$

where $r(X)$ is the rank of matrix X and $l(X):\mathbb{R}^{m\times n}\to\mathbb{R}$ has a Lipschitz continuous gradient with Lipschitz constant L_l. For solving (1.1), plenty of relaxation models and numerical methods have been proposed in [6, 7, 8, 13, 14, 15, 16, 18, 19]. The well known nuclear norm minimization problem is the most widely used convex relaxation problem, which is convex and thus computationally tractable. However, it may sometimes yield suboptimal performance due to the biased approximation to rank minimization. The reason for this is that nuclear norm is dominated by singular values with large magnitudes, while singular values make equal contributions to rank function.

To overcome this drawback, the regularization term $L_{t,*-\alpha F}$ is adopted to relax the rank function such that problem (1.1) is relaxed as

$$\min_{X\in\mathbb{R}^{m\times n}} L(X) := \|X\|_{t,*-\alpha F}+l(X). \tag{1.2}$$

2020 *Mathematics Subject Classification.* Primary: 15A83, 15A29, 65F55.
Key words and phrases. Low rank matrix minimization, forward-backward splitting.
The third author is supported by NSFC grant 11871369.
*Corresponding author: Xinzhen Zhang.

Here, "$*$" denotes the nuclear norm and "F" denotes the Frobenius norm. Specifically, given $X \in \mathbb{R}^{m\times n}$, $t \le m \le n$,

$$\|X\|_{t,*-\alpha F} := \|\sigma(X)\|_{t,l_1-\alpha l_2} = \sum_{i=t+1}^{m} \sigma_i(X) - \alpha\sqrt{\sum_{i=t+1}^{m} \sigma_i^2(X)}, \tag{1.3}$$

where $\sigma(X) = (\sigma_1(X), \cdots, \sigma_m(X))^T$ is a vector composed of X's singular values with $\sigma_1(X) \ge \cdots \ge \sigma_m(X) \ge 0$ and $0 < \alpha \le 1$ is a given parameter. Clearly, $L_{t,*-\alpha F}$ reduces to $L_{*-\alpha F}$ when $t = 0$, i.e., $\|X\|_{*-\alpha F} := \|X\|_* - \alpha\|X\|_F$, and $L_{t,*-\alpha F}$ reduces to the common used $l_1 - \alpha l_2$ when $t = 0$ and matrix X is additionally diagonal. For this case, problem (1.2) can be rewritten as the regularized vector minimization problem in the form of

$$\min_{x\in\mathbb{R}^q} \|x\|_1 - \alpha\|x\|_2 + l(x), \tag{1.4}$$

where $q = \min(m, n)$ and $\|x\|_p = (\sum_{i=1}^{q} |x_i|^p)^{1/p}$ for any $x \in \mathbb{R}^q$, here $p = 1, 2$. Problem (1.4) and its variants have been widely studied for recovering sparse vector [4, 11, 20]. Efficient minimization algorithms of forward-backward splitting (FBS) were proposed for finding an approximate solution to (1.4) or its variants. Recently, a fast approach [11] for minimizing (1.4) was to combine FBS algorithm with proximal operator, which is particularly useful in convex optimization [17] and defined as

$$\mathbf{prox}_{\lambda,\alpha}(y) = \arg\min_x \|x\|_1 - \alpha\|x\|_2 + \frac{1}{2\lambda}\|x - y\|_2^2. \tag{1.5}$$

Based on proximal operator, each iteration of forward-backward splitting for solving (1.4) is expressed as

$$\begin{aligned} x^{k+1} &\in \mathbf{prox}_{\lambda,\alpha}(x^k - \lambda\nabla l(x^k)) \\ &= \arg\min_x \|x\|_1 - \alpha\|x\|_2 + \tfrac{1}{2\lambda}\|x - (x^k - \lambda\nabla l(x^k))\|_2^2, \end{aligned} \tag{1.6}$$

where $\lambda > 0$ is the stepsize. Such algorithm [11] was shown to be much more efficient than the methods based on a difference-of-convex approach in the numerical experiments.

Motivated by these facts, we wonder whether FBS algorithm can be applied to solve (1.2). For this aim, we consider

$$X^{k+1} = \arg\min_{X\in\mathbb{R}^{m\times n}} \|X\|_{t,*} - \alpha\|X\|_{t,F} + \frac{1}{2\lambda}\|X - (X^k - \lambda\nabla l(X^k))\|_F^2. \tag{1.7}$$

This updating rule is thus referred to as a FBS algorithm. For this updating rule, two questions arise. One immediate question is whether (1.7) has a closed-form solution. Another is the performance of the FBS algorithm applied to the low rank matrix recovery problem.

In this paper, we first propose FBS algorithm for (1.2) with closed-form solutions to (1.7) and present convergence analysis. We also conduct numerical experiments to compare the proposed method with SVT [2], FPCA [13] and LMaFit [19]. The computational results demonstrate the FBS algorithm generally outperforms those methods.

The outline is as follows. In Section 2, we recall preliminaries that will be used in this paper. In Section 3, we propose the FBS algorithm for (1.2) and establish the convergence. Experimental results in Section 4 show advantages of our method over the state-of-art methods in matrix completion. Finally, conclusions are presented in Section 5.

2. **Preliminaries.** In this section, we review some basic definitions. The set of all n-dimensional nonnegative vectors is denoted by $\mathbb{R}^n_+$, that is, $x \geq 0$ means that $x \in \mathbb{R}^n_+$. $Diag(x)$ or $Diag(x_1, x_2, \ldots, x_n)$ denotes a diagonal matrix with ith diagonal entry x_i. $\mathbb{R}^{m\times n}$ means the space of $m \times n$ matrices, and I_n means the space of $n \times n$ unitary matrix. Throughout the paper, it is assumed that $m \leq n$.

Now we recall notations on matrix norm. For a matrix $X \in \mathbb{R}^{m\times n}$, the Frobenius norm of X is denoted by $\|X\|_F$, namely, $\|X\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |x_{ij}|^2} = \sqrt{tr(XX^T)}$, where $tr(\cdot)$ denotes the trace of a matrix. The nuclear norm is defined as $\|X\|_* := \sum_{i=1}^m \sigma_i(X)$ and the p-norm of X is denoted by $\|X\|_p$, that is, $\|X\|_p = (\sum_{i=1}^m \sigma_i^p(X))^{1/p}$.

Denote

$$\|X\|_{t,*} = \sum_{i=t+1}^{m} \sigma_i(X), \quad \|X\|_{t,F} = \sqrt{\sum_{i=t+1}^{m} \sigma_i^2(X)}$$

and

$$\mathcal{M}(X) = \{(U, V) \in \mathbb{R}^{m\times m} \times \mathbb{R}^{n\times n} : \ U^TU = I_m, \ V^TV = I_n, \ X = UDiag(\sigma(X))V^T\}. \tag{2.1}$$

Let $X^* = UDiag(\sigma(X^*))V^T$ be any singular value decomposition with the matrices partitioned as

$$U = [U^{(1)}, U^{(2)}], \quad V = [V^{(1)}, V^{(2)}], \tag{2.2}$$

with $U^{(1)}$ and $V^{(1)}$ having t columns.

For a given positive integer t and $x, y \in \mathbb{R}^n$, we denote

$$x = (s^T, w^T)^T, \quad s = (x_1, \ldots, x_t)^T, \quad w = (x_{t+1}, \ldots, x_n)^T. \tag{2.3}$$

Similarly, denote

$$y = (p^T, q^T)^T, \quad p = (y_1, \ldots, y_t)^T, \quad q = (y_{t+1}, \ldots, y_n)^T. \tag{2.4}$$

Lemma 1 [11] describes the closed-form solutions of $\mathbf{prox}_{\lambda,\alpha}(y)$. For the reader's convenience, we present it here.

Lemma 2.1. *Given $y \in \mathbb{R}^n$, $\lambda > 0$ and $\alpha > 0$, we have the following statements about the optimal solution x^* to the optimization problem in (1.5):*

(1) When $\|y\|_\infty > \lambda$, $x^* = z(\|z\|_2 + \alpha\lambda)/\|z\|_2$ for $z = S_1(y, \lambda) \in \mathbb{R}^n$. Here, $S_1(y, \lambda)$ is defined with its entries

$$(S_1(y, \lambda))_i = \begin{cases} y_i - \lambda & y_i > \lambda, \\ 0 & |y_i| \leq \lambda, \\ y_i + \lambda & y_i < -\lambda. \end{cases} \tag{2.5}$$

(2) When $\|y\|_\infty = \lambda$, x^* is an optimal solution if and only if it satisfies $x_i^* = 0$ if $|y_i| < \lambda$, $\|x^*\|_2 = \alpha\lambda$ and $x_i^* y_i^* \geq 0$ for all i. When there are more than one component having the maximum absolute value λ, the optimal solution is not unique; in fact, there are infinitely many optimal solutions.

(3) When $(1 - \alpha)\lambda < \|y\|_\infty < \lambda$, x^* is an optimal solution if and only if it is a 1-sparse vector satisfying $x_i^* = 0$ if $|y_i| < \|y\|_\infty$, $\|x^*\|_2 = \|y\|_\infty + (\alpha - 1)\lambda$ and $x_i^* y_i \geq 0$ for all i. The number of optimal solutions is the same as the number of components having the maximum absolute value $\|y\|_\infty$.

(4) When $\|y\|_\infty \leq (1 - \alpha)\lambda$, $x^* = 0$.

To end this section, we present the following assumptions on the loss function l:
Assumption 1. The loss function $l : \mathbb{R}^{m\times n} \to \mathbb{R}^+$ is a smooth function of type $C^{1,1}$, i.e., continuously differentiable with L_l-Lipschitz-continuous gradient, that is,

$$\|\nabla l(X) - \nabla l(Y)\|_F \le L_l\|X - Y\|_F, \quad \forall X, Y \in \mathbb{R}^{m\times n}. \tag{2.6}$$

Such $L_l > 0$ is called Lipschitz constant of ∇l.
Assumption 2. The objective function $l(X)$ is coercive, i.e., $l(X) \to \infty$ iff $\|X\|_F \to \infty$.

Throughout this paper, the loss function $l(X)$ is possibly nonconvex.

3. Forward-backward splitting method and its convergence analysis.

In this section, we first introduce a class of first-order stationary points of problem (1.2). Then we propose a FBS algorithm for solving (1.2). Furthermore, the convergence analysis is established.

Denote $\Phi(X) = \phi(\sigma(X))$. Assume that ϕ is locally Lipschitz around $\sigma(X)$, that is, $\Phi(X) = \phi(\sigma(X))$ is locally Lipschitz on X. It follows from [9, Theorem 3.7] that the Clarke subdifferential [5] of Φ at X is given by

$$\partial\Phi(X) = \{UDiag(d)V^T : d \in \partial\phi(\sigma(X)), \quad (U, V) \in \mathcal{M}(X)\},$$

where $\partial\phi$ is the Clarke subdifferential of ϕ and $\mathcal{M}(X)$ is defined as in (2.1). Then we have the following definition.

Definition 3.1. Let $(U, V) \in \mathcal{M}(X^*)$ and $U^{(2)}, V^{(2)}$ be defined as in (2.2). Denote $f(w) = \|w\|_1 - \alpha\|w\|_2$ with (2.3) and $w^* := (\sigma_{t+1}(X^*), \ldots, \sigma_m(X^*))^T$. Then X^* is a critical point if

$$0 \in \{\nabla l(X^*) + U^{(2)}Diag(\partial f(w^*))(V^{(2)})^T\}. \tag{3.1}$$

Here $Diag(\partial f(w^*)) \in \mathbb{R}^{(m-t)\times(n-t)}$.

In the following, we extend the FBS algorithm via proximal operator proposed in [11] to (1.2) and then establish its convergence. Before proceeding, we need the closed-form solution for the proximal operator of $\|\cdot\|_{t,l_1-\alpha l_2}$.

By direct computation, together with (2.3) and (2.4), we have

$$\|x\|_{t,1} - \alpha\|x\|_{t,2} + \frac{1}{2\lambda}\|x - y\|^2 = \|w\|_1 - \alpha\|w\|_2 + \frac{1}{2\lambda}\|w - q\|_2^2 + \frac{1}{2\lambda}\|s - p\|_2^2.$$

Hence, it follows

$$(p^T, (\mathbf{prox}_{\lambda,\alpha}(q))^T)^T = \arg\min_{s,w}\|w\|_1 - \alpha\|w\|_2 + \frac{1}{2\lambda}\|w - q\|_2^2 + \frac{1}{2\lambda}\|s - p\|_2^2. \tag{3.2}$$

As stated in Lemma 2.1, $(\mathbf{prox}_{\lambda,\alpha}(q))^T$ is not unique. In order to find the sparsest vector, we choose the sparsest one obtained by Lemma 2.1 as $(\mathbf{prox}_{\lambda,\alpha}(q))^T$.

Consider

$$\min_{X\in\mathbb{R}^{m\times n}} \left\{\langle C, X - B\rangle + \frac{1}{2\lambda}\|X - B\|_F^2 + \|X\|_{t,*} - \alpha\|X\|_{t,F}\right\} \tag{3.3}$$

for some $B, C \in \mathbb{R}^{m\times n}$, $\lambda > 0$. We now show that problem (3.3) has a closed-form solution.

Theorem 3.2. *Given $B, C \in \mathbb{R}^{m\times n}$, $\lambda > 0$, let $UDiag(y)V^T$ be the singular value decomposition of $B - \lambda C$ and $x^* = argmin_{x\in\mathbb{R}^n_+}\{\|x\|_{t,1} - \alpha\|x\|_{t,2} + \frac{1}{2\lambda}\|x - y\|_2^2\}$. Then $X^* = UDiag(x^*)V^T$ is an optimal solution to problem (3.3).*

Proof. By direct computation, (3.3) can be rewritten as

$$\min_{X\in\mathbb{R}^{m\times n}} \left\{ \frac{1}{2\lambda}\|X-(B-\lambda C)\|_F^2 + \|X\|_{t,*} - \alpha\|X\|_{t,F} \right\}. \tag{3.4}$$

Since $x^* = \arg\min_{x\in\mathbb{R}^n_+} \left\{\|x\|_{t,1} - \alpha\|x\|_{t,2} + \frac{1}{2\lambda}\|x-y\|_2^2\right\}$, $X^* = UDiag(x^*)V^T$ is an optimal solution to problem (3.4) based on Proposition 2.1 in [12] with $F(X) = \|X\|_{t,*} - \alpha\|X\|_{t,F}$ and $\phi(t) = \frac{t^2}{2\lambda}$. Hence X^* is also an optimal solution of (3.3). □

Now we are in the position to present FBS algorithm for solving (1.2) in detail.

Algorithm 3.3. (FBS algorithm)

Input: Function $l(\cdot)$ and parameters $\lambda > 0$, $\alpha \in (0,1]$, positive integer t. Let $X^0 \in \mathbb{R}^{m\times n}$ and $k := 0$.

while not converge **do**

***Step* 1.** Compute $\nabla l(X^k)$ and $B^k := X^k - \lambda\nabla l(X^k)$.

***Step* 2.** Compute $U^k Diag(y^k)(V^k)^T$, which is the singular value decomposition of B^k.

***Step* 3.** Let $x^{k+1} = ((p^k)^T, (\mathbf{prox}_{\lambda,\alpha}(q^k))^T)^T$ by (3.2) with $p^k := (y_1^k, \ldots, y_t^k)^T$ and $q^k := (y_{t+1}^k, \ldots, y_n^k)^T$.

***Step* 4.** Update X^{k+1} by $X^{k+1} = U^k Diag(x^{k+1})(V^k)^T$.

***Step* 5.** Let $k := k+1$ and go to Step 1.

end while

Output: X^k.

Theorem 3.4. *Assume that $l(x)$ satisfies Assumption 1,2 and $\lambda < \frac{1}{2L_l}$. Let $\{X^k\}$ be generated by FBS algorithm and y^* be an accumulation point of $\{y^k\}$, where $\{y^k\}$ is the singular value vector of $X^k - \lambda\nabla l(X^k)$. Then*

(1) $L(X^k)$ is monotonically decreasing. Indeed, $L(X^{k+1}) - L(X^k) \le (L_l - \frac{1}{2\lambda})\|X^{k+1} - X^k\|_F^2$.

(2) The sequence $\{X^k\}$ is bounded.

(3) $\lim_{k\to\infty}(X^k - X^{k+1}) = 0$.

(4) Let X^* be an accumulation point of $\{X^k\}$. Then X^* is a first-order stationary point of (1.2), i.e., (3.1) holds at X^*.

Proof. (1) Denote $F(X) = \|X\|_{t,*} - \alpha\|X\|_{t,F}$. For some ξ^k between X^k and X^{k+1}, we have

$$\begin{aligned}
& L(X^{k+1}) - L(X^k) = F(X^{k+1}) - F(X^k) + l(X^{k+1}) - l(X^k) \\
\le\ & \tfrac{1}{2\lambda}\|\lambda\nabla l(X^k)\|_F^2 - \tfrac{1}{2\lambda}\|X^{k+1} - X^k + \lambda\nabla l(X^k)\|_F^2 + l(X^{k+1}) - l(X^k) \\
=\ & l(X^{k+1}) - l(X^k) - \langle\nabla l(X^k), X^{k+1} - X^k\rangle - \tfrac{1}{2\lambda}\|X^{k+1} - X^k\|_F^2 \\
=\ & \langle\nabla l(\xi^k), X^{k+1} - X^k\rangle - \langle\nabla l(X^k), X^{k+1} - X^k\rangle - \tfrac{1}{2\lambda}\|X^{k+1} - X^k\|_F^2 \\
=\ & \langle\nabla l(\xi^k) - \nabla l(X^k), X^{k+1} - X^k\rangle - \tfrac{1}{2\lambda}\|X^{k+1} - X^k\|_F^2 \\
\le\ & \|\nabla l(\xi^k) - \nabla l(X^k)\|_F\|X^{k+1} - X^k\|_F - \tfrac{1}{2\lambda}\|X^{k+1} - X^k\|_F^2 \\
\le\ & L_l\|X^{k+1} - X^k\|_F^2 - \tfrac{1}{2\lambda}\|X^{k+1} - X^k\|_F^2 \\
=\ & (L_l - \tfrac{1}{2\lambda})\|X^{k+1} - X^k\|_F^2 \\
\le\ & 0.
\end{aligned} \tag{3.5}$$

The first inequality comes from (1.7), the second one is from Cauchy-Schwarz inequality, the third one is due to Assumption 1 and the last inequality comes from $\lambda < \frac{1}{2L_l}$. Hence, (1) is asserted.

(2) From (1), $L(X^k)$ is bounded above. Since $\|X\|_* \geq \|X\|_F \geq \alpha\|X\|_F$, $F(X) \geq 0$, which implies that $L(X) = F(X) + l(X) \geq l(X)$. So we can assert that $l(X^k)$ is bounded above. From Assumption 2, we have the boundness of $\{X^k\}$.

(3) By summing inequalities in (3.5), there holds

$$(\frac{1}{2\lambda} - L_l)\sum_{k=1}^{\infty}\|X^{k+1} - X^k\|_F^2 \leq L(X^1) - \lim_{k\to\infty} L(X^k) < \infty.$$

The last inequality is due to the fact that $\{X^k\}$ is bounded from result (2). Thus $\lim_{k\to\infty}(X^k - X^{k+1}) = 0$.

(4) Let $B^k = X^k - \lambda\nabla l(X^k)$, $U^k Diag(y^k)(V^k)^T$ be the singular value decomposition of B^k. Let x^{k+1} and X^{k+1} be as in Step 3 and Step 4. Then

$$\begin{aligned} X^{k+1} - X^k &= X^{k+1} - B^k + B^k - X^k \\ &= U^k Diag(x^{k+1} - y^k)(V^k)^T + (X^k - \lambda\nabla l(X^k)) - X^k \\ &= U^k Diag(x^{k+1} - y^k)(V^k)^T - \lambda\nabla l(X^k) \\ &= U^{(2,k)} Diag(\mathbf{prox}_{\lambda,\alpha}(q^k) - q^k)(V^{(2,k)})^T - \lambda\nabla l(X^k), \end{aligned}$$

where the last equality comes from (3.2).

Since $\mathbf{prox}_{\lambda,\alpha}(q^k) = \arg\min_x \|x\|_1 - \alpha\|x\|_2 + \frac{1}{2\lambda}\|x - q^k\|^2$, it holds

$$\partial f(\mathbf{prox}_{\lambda,\alpha}(q^k)) + \frac{1}{\lambda}(\mathbf{prox}_{\lambda,\alpha}(q^k) - q^k) = 0,$$

which leads to $\mathbf{prox}_{\lambda,\alpha}(q^k) - q^k = -\lambda\partial f(\mathbf{prox}_{\lambda,\alpha}(q^k))$. Now we can assert that $U^{(2,k)}\partial f(\mathbf{prox}_{\lambda,\alpha}(q^k))(V^{(2,k)})^T - \nabla l(X^k) \to 0$ since $\|X^{k+1} - X^k\|_F \to 0$.

Denote $\mathcal{K}$ is an index set such that $\{X^k\}_{\mathcal{K}} \to X^*$. Without loss of generality, it is assumed that $\{U^{(k)}\}_{\mathcal{K}} \to U^*$, $\{V^{(k)}\}_{\mathcal{K}} \to V^*$ and $\{x^k\}_{\mathcal{K}} \to x^*$. Then $X^* = U^* Diag(x^*)(V^*)^T$. Hence $U^{(2,*)} Diag(\partial f(\mathbf{prox}_{\lambda,\alpha}(q^*)))(V^{(2,*)})^T - \nabla l(X^*) = 0$, which means that X^* is a first-order stationary point.

□

4. **Numerical results.** In this section, we conduct numerical experiments to test the performance of the FBS algorithm. In particular, we apply it to solve the problem (1.2) with $l(X) = \theta\|\mathcal{P}_\Omega(X - M)\|_F^2$, that is,

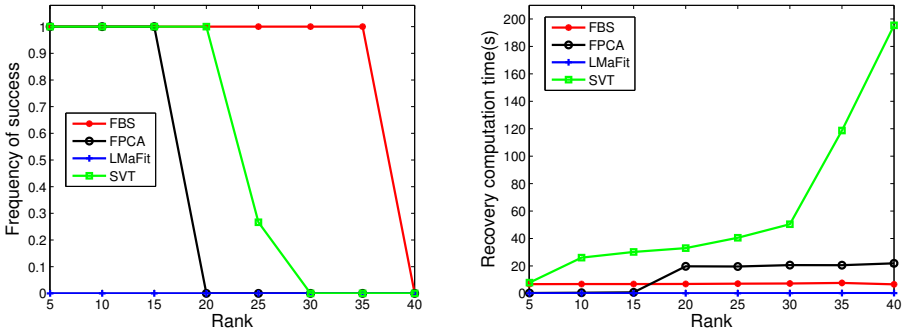
$$\min_{X\in\mathbb{R}^{m\times n}} L(X) := \|X\|_{t,*} - \alpha\|X\|_{t,F} + \theta\|\mathcal{P}_\Omega(X - M)\|_F^2. \tag{4.1}$$

Here $M \in \mathbb{R}^{m\times n}$, Ω is an index set which locates the observed data, $\mathcal{P}_\Omega$ is a linear operator that extracts the entries in Ω and fills the entries not in Ω with zeros. We conduct extensive experiments to evaluate our method and then compare it with some existing methods, including SVT [2], FPCA [13] and LMaFit [19]. The platform is Matlab R2014a under Windows 10 on a desktop of a 3.6GHz CPU and 8GB memory. We adopt the relative error as evaluation metrics

$$rel.err := \frac{\|X^* - M\|_F}{\|M\|_F}.$$

4.1. **Numerical simulation.** In this subsection, we aim to recover a random matrix $M \in \mathbb{R}^{m\times n}$ with rank r based on a subset of entries $\{M_{ij}\}_{(i,j)\in\Omega}$. In detail, we first generate random matrices $M_L = rand(m, r) \in \mathbb{R}^{m\times r}$ and $M_R = rand(n, r) \in \mathbb{R}^{n\times r}$, then let $M = M_L M_R^T$. We then sample a subset with sampling ratio p uniformly at random, where $p = |\Omega|/(mn)$. In our experiment, we set $m = n = 200$ and $p = 0.5$. For each sample ratio p and rank r, we apply FBS, FPCA, LMaFit

and SVT to solve (4.1) on 30 instances that are randomly generated as above. In particular, we set 1000 as the maximum of iterative numbers and $tol = 1e-6$ for all algorithms. In addition, we set $\tau = 5\sqrt{mn}$ and $\delta = 1.2/p$ for SVT and the initial rank $K = \lfloor 1.5r \rfloor$ for LMaFit. We set $\alpha(k) = \frac{1}{1+e^{-0.02k}}$, $\theta = 0.9/\lambda$ and $t = 1$ for FBS.



(A) Frequency of success (B) Recovery computation time(s)

FIGURE 1. Comparison of random data with $p = 0.5$.

From Figure 1, we can see that as the rank increases, the recovery rate of the four algorithms gradually decreases and the required running time gradually increases. More specifically, it can be seen from Figure 1(A) that no matter what the rank is, LMaFit cannot recover the matrix; when the rank reaches 20, FPCA fails; when the rank reaches 30, SVT fails; when the rank is 30, FBS has a 60% chance of recovering the matrix. Furthermore, Figure 1(B) shows that the running time of FBS is always less than 25s, which is not much different from the minimum running time. As the rank increases, the running time of SVT increases dramatically. In summary, the FBS algorithm is the best one for matrix completion.

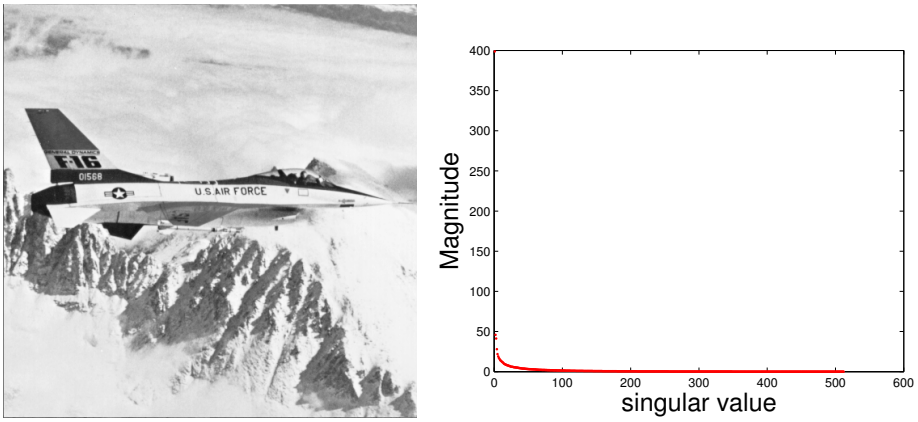
4.2. **Image simulation.** Note that a grayscale image can be expressed by a matrix. The grayscale image inpainting problem can be modeled as matrix completion problem when the matrix data is of low rank or numerically low rank.

In this subsection, we apply FBS, FPCA, LMaFit and SVT to solve a grayscale image inpainting problem. We use the USC-SIPI image database[1] to evaluate our proposed algorithm with $t = 50$ for image inpainting. We randomly select a picture from it and plot the singular values of the picture. Figure 2 shows that the matrix corresponding to the picture is numerically low rank. In our test, we randomly select 4 images from this database and test them with entries missed randomly by sampling ratio $p = 0.6$.

In general, there is no drastic changes in the image data between two adjacent rows and two adjacent columns. To study such stability, we calculate the data pairs of two adjacent rows and two adjacent columns. We measure the stability between two adjacent rows of the images at the ith row as

$$\Delta\text{row}(i,j) = \frac{|M_{ij} - M_{(i+1)j}|}{\max\limits_{1\le i\le m-1} |M_{ij} - M_{(i+1)j}|},$$

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



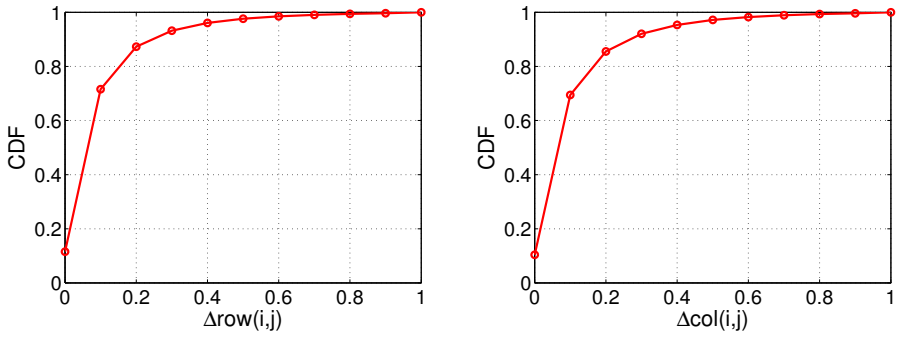
(A) A 512×512 image example. (B) The singular values of image.

FIGURE 2. As can be seen, the distribution of the singular values has a fast decaying distribution, and the information is dominated by the top 50 singular values.

where $\max_{1\le i\le m-1} |M_{ij} - M_{(i+1)j}|$ is the maximal gap between any two adjacent rows in the image data. We plot the Cumulative Distribution Function (CDF) of $\Delta\text{row}(i,j)$ for Male image in Figure 3(A). The X-axis represents the normalized difference values between two adjacent rows slots, i.e., $\Delta\text{row}(i,j)$. The Y-axis represents the cumulative probability. Similarly, we plot the CDF of $\Delta col(i,j)$ for Male image in Figure 3(B). We can see that both $\Delta\text{row}(i,j) < 0.8$ and $\Delta\text{col}(i,j) < 0.8$ are more than 80%. Based on the above analysis, for the restoration of images, we let

$$l(X) = \theta \left\| \mathcal{P}_\Omega(X - M) \right\|_F^2 + \beta \left(\left\| L^T X \right\|_F^2 + \left\| XR \right\|_F^2 \right),$$

where L and R are $m \times (m-1)$ and $n \times (n-1)$ Toeplitz matrices, respectively. We call this algorithm FBS-TM.

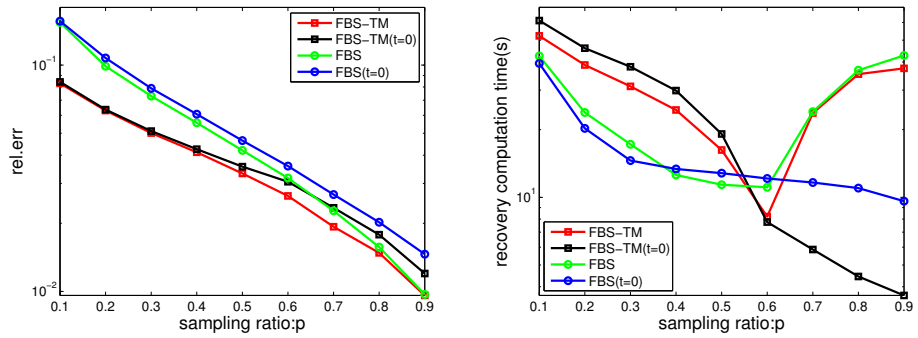


(A) row (B) col

FIGURE 3. An empirical study of Male images data.

In order to compare the performance of FBS, FBS($t = 0$), FBS-TM and FBS-TM($t = 0$), we select the picture of Airplane and set sampling ratio p from 0.1

to 0.9 with increment 0.1. In particular, we set 1000 as the maximum iterative number and $tol = 5e-5$ for all methods. Figure 4 reports the *rel.err* value and running time of such algorithms. We can see that truncated FBS-TM and FBS are superior to FBS-TM($t = 0$) and FBS($t = 0$) in *rel.err* value regardless of the sample rate, which indicates that the truncated norm is meaningful. We can also see that regardless of the sample rate, FBS-TM and FBS-TM($t = 0$) are superior to FBS and FBS($t = 0$) in *rel.err* value, respectively. It explains that the model based on stability is also helpful.



(A) rel.err (B) time

FIGURE 4. Comparison of the four algorithms with different sampling ratio p.

TABLE 1. Numerical results of different images

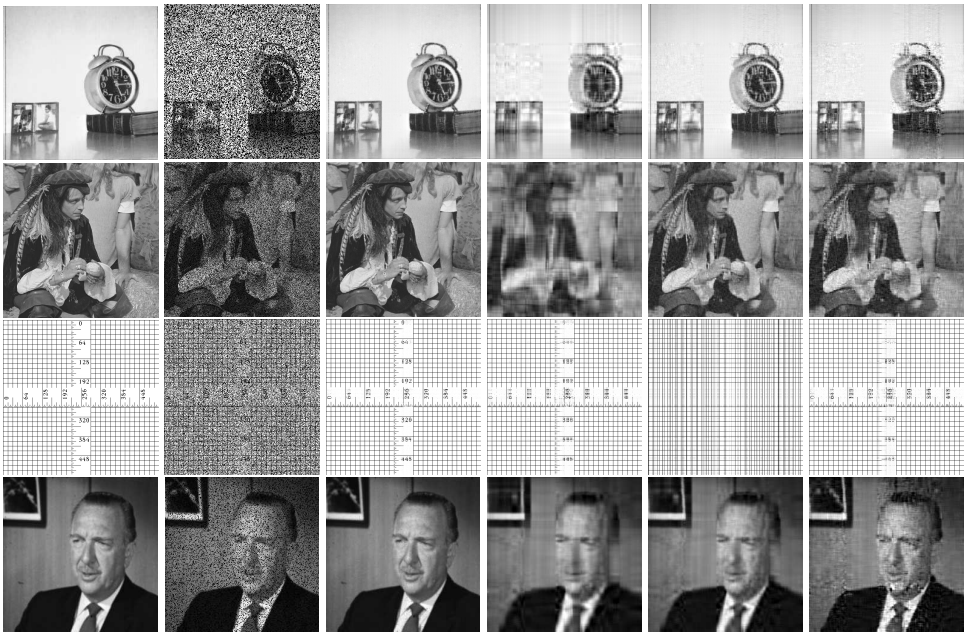
Method	Clock		Male		Pixel ruler		Walter Cronkite	
	rel.err	time	rel.err	time	rel.err	time	rel.err	time
FBS-TM	**2.96e-02**	5.47	**4.63e-02**	40.28	**4.31e-02**	9.66	**1.77e-02**	7.62
LMaFit	7.95e-02	0.05	1.74e-01	0.16	8.74e-02	1.43	1.27e-01	0.05
SVT	4.93e-02	19.05	1.02e-01	39.51	3.09e-01	45.64	7.23e-02	19.39
FPCA	7.60e-02	2.82	1.36e-01	14.69	1.30e-01	14.69	8.29e-02	2.94

Figure 5 shows that FBS-TM is the best one to inpaint image, while images recovered by another three algorithms are not clear. Table 1 presents the numerical results, which indicates that FBS-TM is the best one.

For further comparison, we also recover images of the deterministically masked images by boat and airplane respectively. Clearly, the masked images are no-mean-sampling. The results are displayed in Figure 6 and Table 2, where $t = 60$ in our proposed algorithm. Figure 6 indicates that FBS-TM has the best performance.

4.3. **MRI volume dataset.** The resolution of the MRI volume dataset[2] is of size 217×181 with 181 slices and we pick the first 100 ones. We consider the case where entries are missing at random with sampling ratio $p = 0.8$. From Figure 7, it is asserted that in the restoration of all 100 frames of pictures, the restoration effect

[2] http://www.bic.mni.mcgill.ca/ServicesBrainWeb/HomePage



(A) Original (B) Observation (C) FBS-TM (D) LMaFit (E) SVT (F) FPCA

FIGURE 5. Testing images. "Clock": grayscale image of 256×256 pixels. "Male": grayscale image of 512×512 pixels. "Pixel ruler": grayscale image of 512×512 pixels. "Walter Cronkite": grayscale image of 256×256 pixels.

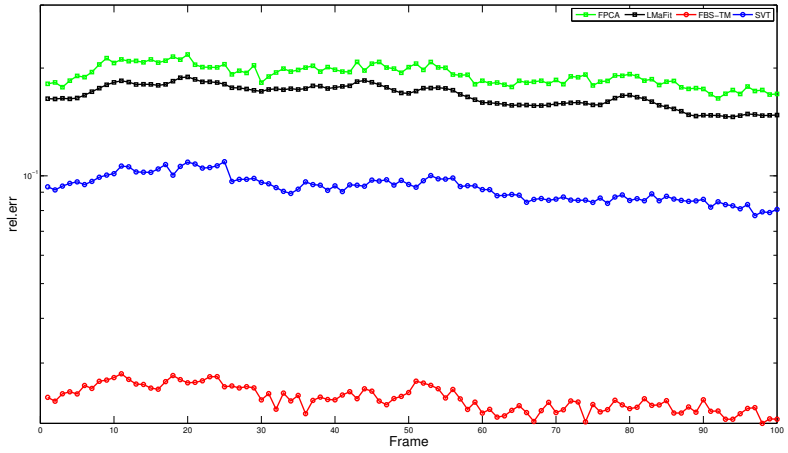
(A) Original (B) Observation (C) FBS-TM (D) LMaFit (E) SVT (F) FPCA

FIGURE 6. Recovered images of the masked images.

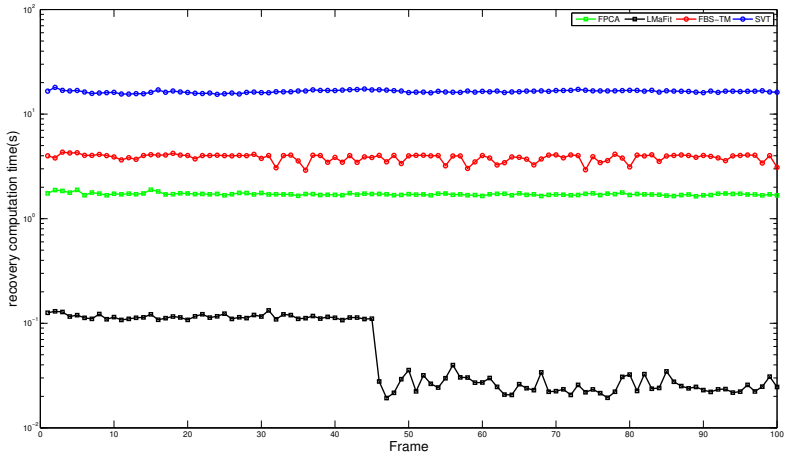
TABLE 2. Numerical results of masked images.

FBS-TM		LMaFit		SVT		FPCA	
rel.err	time	rel.err	time	rel.err	time	rel.err	time
2.95e-02	25.45	1.43e-01	0.49	1.70e-01	11.26	7.26e-01	16.51
1.50e-02	25.10	1.12e-01	0.26	3.04e-02	52.21	5.35e-01	15.43

based on our proposed algorithm with $t = 50$ is far better than another three ones. Moreover, we also present the difference of the restored pictures and the original ones by various algorithms. For better visualization, we add 0.5 to the pixel. It is clear that the pictures obtained by FBS-TM have almost no outliers, indicating the best performance.



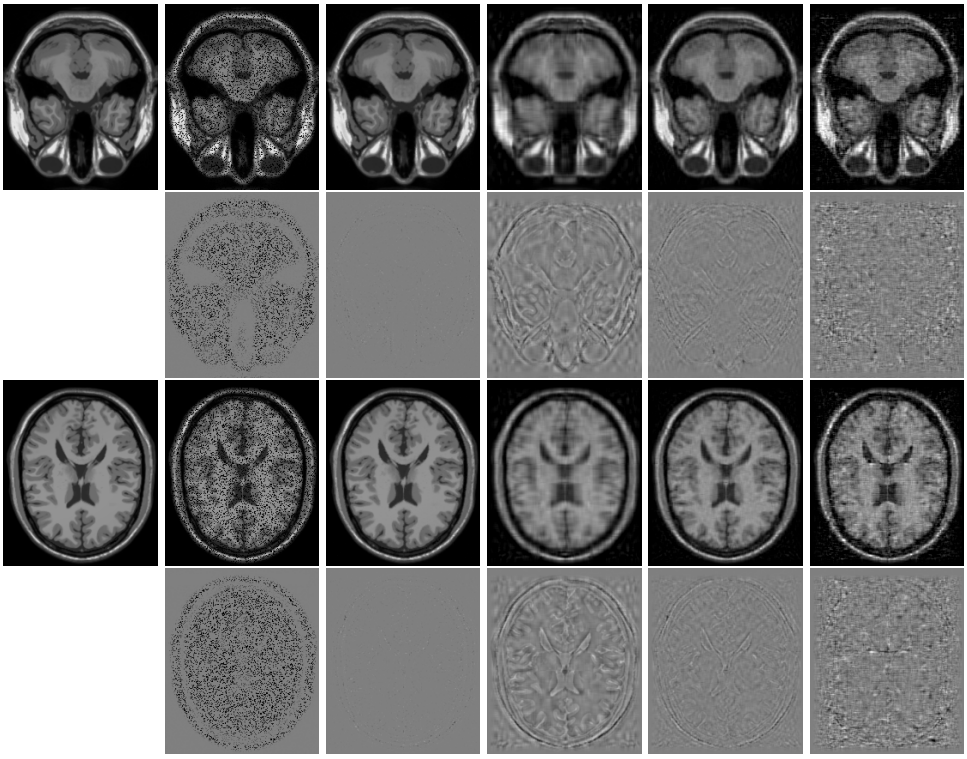
(A) rel.err



(B) time

FIGURE 7. Comparison of four algorithms for the first 100 frames.

Acknowledgments. The authors would like to thank the anonymous referees for their helpful comments.



(A) Original (B) Observation (C) FBS-TM (D) LMaFit (E) SVT (F) FPCA

FIGURE 8. Comparison results of the MRI volume dataset.

5. **Conclusions.** In this paper, we considered the $L_{t,*-\alpha F}$ regularization. We proposed the FBS algorithm. The computational results demonstrated that the FBS algorithm generally outperformed those algorithms.

Besides the $L_{t,*-\alpha F}$, the truncated norm regularizer, there are some other popular nonconvex regularizers for producing a sparse solution of a system or a sparse minimization problem, which may be extended to low rank matrix minimization. In this paper, we studied $L_{t,*-\alpha F}$ regularized low rank matrix minimization problems. An open question is whether we can follow this way with other regularizers for the low rank matrix minimization problems. This will be our future work.

REFERENCES

[1] A. Argyriou, T. Evgeniou and M. Pontil, Convex multi-task feature learning, *Mach. Learn.*, 73 (2008), 243–272.

[2] J. F. Cai, E. J. Candes and Z. Shen, A singular value thresholding algorithm for matrix completion, *SIAM J. Optim.*, **20** (2010), 1956–1928.

[3] E. J. Cands and B. Recht, Exact matrix completion via convex optimization, *Found. Comput. Math.*, 9 (2009), 717–772.

[4] R. Chartrand, Exact reconstruction of sparse signals via nonconvex minimization, *IEEE Signal Process. Lett.*, 14 (2007), 707–710.

[5] F. H. Clarke, *Optimization and Nonsmooth Analysis*, 2^{nd} edition, Society for Industrial and Applied Mathematics, 1990.

[6] M. Fazel, H. Hindi and S. P. Boyd, A rank minimization heuristic with application to minimum order system approximation, *Proceedings of the 2001 American Control Conference*, **6** (2001), 4734–4739.
[7] M. Fornasier, H. Rauhut and R. Ward, Low-rank matrix recovery via iteratively reweighted least squares minimization, *SIAM J. Optim.*, **21** (2011), 1614–1640.
[8] M. Lai, Y. Xu and W. Yin, Improved iteratively reweighted least squares for unconstrained smoothed l_q minimization, *SIAM J. Numer. Anal*, **51** (2013), 927–957.
[9] A. S. Lewis and H. S. Sendov, Nonsmooth analysis of singular values. Part II: Applications, *Set-Valued Analysis*, **13** (2005), 243–264.
[10] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu and Y. Ma, Robust recovery of subspace structures by low-rank representation, *IEEE Trans. Pattern Anal. Mach. Intell.*, **35** (2013), 171–184.
[11] Y. Lou and M. Yan, Fast $L_1 - L_2$ minimization via a proximal operator, *J. Sci. Comput.*, **74** (2018), 767–785.
[12] Z. Lu, Y. Zhang and X. Li, Penalty decomposition methods for rank minimization, *Optim. Methods Softw.*, **30** (2015), 531–558.
[13] S. Ma, D. Goldfarb and L. Chen, Fixed point and Bregman iterative methods for matrix rank minimization, *Math. Program.*, **128** (2011), 321–353.
[14] Z. Ming, L. Zhang, Y. Xu and M. Bakshi, An algorithm for matrix recovery of high-loss-rate network traffic data, *Appl. Math. Model.*, **96** (2021), 645–656.
[15] D. Phan and T. Nguyen, An accelerated IRNN-Iteratively reweighted nuclear norm algorithm for nonconvex nonsmooth low rank minimization problems, *J. Comput. Appl. Math.*, **396** (2021), Paper No. 113602, 14 pp.
[16] B. Recht, M. Fazel and P. A. Parrilo, Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization, *SIAM Rev.*, **52** (2010), 471–501.
[17] Rockafellar. R. T, *Convex Analysis*, Princeton University Press, 1997.
[18] K. Toh and S. Yun, An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems, *Pac. J. Optim.*, **6** (2010), 615–640.
[19] Z. Wen, W. Yin and Y. Zhang, Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm, *Math. Program. Comput.*, **4** (2012), 333–361.
[20] P. Yin, Y. Lou, Q. He and J. Xin, Minimization of l_{1-2} for compressed sensing, *SIAM J. Sci. Comput.*, **37** (2015), 536–563.

Received September 2021; revised January 2022; early access March 2022.

E-mail address: huiyuanguo@tju.edu.cn
E-mail address: quanyu@tju.edu.cn
E-mail address: xzzhang@tju.edu.cn
E-mail address: lulucheng@tju.edu.cn