Journal of Computational Mathematics
Vol.xx, No.x, 2025, 1–20.
http://www.global-sci.org/jcm
doi:10.4208/jcm.2504-m2024-0005

TENSOR COMPLETION VIA MINIMUM AND MAXIMUM OPTIMIZATION WITH NOISE*

Chuanlong Wang and Rongrong Xue[1)]
Shanxi Key Laboratory for Intelligent Optimization Computing and Block-chain Technology, Taiyuan Normal University, Jinzhong 030619, China
Emails: wangcl19641010@163.com, 17836206546@163.com

Abstract

In this paper, the novel optimization model for solving tensor completion with noise is proposed, its objective function is a convex combination of the minimum nuclear norm and maximum nuclear norm. The necessary condition and sufficient condition of the stationary point and optimal solution are discussed. Based on the proximal gradient algorithm and feasible direction method, we design the new algorithm for solving the proposed nonconvex and nonsmooth optimization problem and prove that the sub-sequence generated by the new algorithm converges to the stationary point. Finally, experimental results on the random sample completions and images show that the proposed optimization and algorithm are superior to the compared algorithms in CPU time or precision.

Mathematics subject classification: 90C26, 90C47, 15A69.
Key words: Tensor completion with noise, Minimum and maximum optimization, Proximal gradient algorithm, Feasible direction method.

1. Introduction

With the emergence of various high-dimensional data in many fields such as image analysis [2,16], computer vision [1], signal processing [18] etc., tensor as the higher-order generalization of vector and matrix plays an increasingly important role. In particular, tensor completion which has many applications in image recovery [24]. A gray image is a matrix, which is a 2D data. A color image is a three order tensor, which is a 3D data. A color video is a four order tensor, which is a 4D data. Tensor completion refers to the technique of completing the tensor with part sample data by minimizing the its rank. Tensor completion can be expressed by the following optimization:

$$\begin{aligned} &\min_{\mathcal{X}} \ \operatorname{rank}(\mathcal{X}) \\ &\text{s.t.} \quad P_\Omega(\mathcal{X}) = P_\Omega(\mathcal{T}), \end{aligned} \tag{1.1}$$

where $\mathcal{T}, \mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ are input and output N-order tensors, Ω is an index set of the known samples, and $P_\Omega(\cdot)$ is an orthogonal projection onto the set Ω, $\operatorname{rank}(\mathcal{X})$ is the rank of $\mathcal{X}$. In contrast to the rank of a matrix, the rank of a tensor is much more complicated. In the literature, tensor rank can be expressed in various forms such as CANDECOMP/PARAFAC (CP) rank [15], Tucker rank [28], tensor train (TT) rank [11], tensor ring (TR) rank [35], and so on. Since the model (1.1) is a discrete and discontinuous programming, the computational complexity is non-deterministic polynomial (NP). Usually, the model (1.1) is relaxed into continuous convex

* Received January 9, 2024 / Revised version received September 25, 2024 / Accepted April 3, 2025 / Published online May 21, 2025 /
1) Corresponding author

(or nonconvex) programming as matrix completion [6]. Liu *et al.* [19] first put the optimization problem (1.1) for Tucker rank to convex relaxation as follows:

$$\begin{aligned} &\min_{\mathcal{X}} \sum_{i=1}^{N} \alpha_i \|\mathcal{X}_{(i)}\|_* \\ &\text{s.t.} \quad P_\Omega(\mathcal{X}) = P_\Omega(\mathcal{T}), \end{aligned} \tag{1.2}$$

where $\mathcal{X}_{(i)} \in R^{I_i \times \prod_{j\neq i} I_j}$ is the mode-i unfolding of tensor $\mathcal{X}, \alpha_i \geq 0$ and $\sum_{i=1}^{N} \alpha_i = 1$. For $i = 1, 2, \ldots, N$,

$$\|\mathcal{X}_{(i)}\|_* = \sum_{j=1}^{I_i} \sigma_j(\mathcal{X}_{(i)})$$

denotes the nuclear norm of $\mathcal{X}_{(i)}$ and $\sigma_j(\mathcal{X}_{(i)})$ denotes the j-th largest singular value of $\mathcal{X}_{(i)}$. For more models such as the adaptive weighted Tucker rank or TT-rank or TR-rank and more methods such as difference of convex (DC) optimization algorithm or Riemannian optimization method, etc. see [7, 12, 17, 20, 22, 23].

The above-mentioned tensor completion models and methods assume that the observed entries are noise-free. But in practice these data will also be damaged by noise, so tensor completion with noise was studied, some literatures proposed the corresponding models to solve the tensor completion with noise problem (see [9,13,25,32–34]) as matrix completion with noise problem (see [5,14,27]). The convex relaxation model of the tensor completion with noise for Tucker rank is in the following:

$$\begin{aligned} &\min_{\mathcal{X}} \sum_{i=1}^{N} \alpha_i \|\mathcal{X}_{(i)}\|_* \\ &\text{s.t.} \quad \|P_\Omega(\mathcal{X}) - P_\Omega(\mathcal{T})\|_F \leq \delta, \end{aligned} \tag{1.3}$$

where $\delta \geq 0$ measures the noise level. Obviously, the model (1.2) is a special case of model (1.3), where the noise level $\delta = 0$.

Generally, solving (1.3) requires to unfold the tensor $\mathcal{X}$ into N-modes and to perform a large number of singular value decompositions (SVDs), which are highly time consuming. To overcome the difficulty, we propose the minimum and maximum nuclear norm optimization model in the next subsection. Furthermore, In the study of existing, the nuclear norm model (1.3) was discussed in [9], the combination model of total variation and nuclear norm was studied in [32], the tensor train rank and tensor ring rank model were explored in [25] and [13, 33, 34], respectively. But all algorithms in [9,13,25,32–34] use same techniques as the constraint without noise ($P_\Omega(\mathcal{X}) = P_\Omega(\mathcal{T})$). In the other word, the inequality constraint is not dealt. Thus, its effect is not good when the inequality constraint is active. Hence, in the paper we use the feasible direction method to deal with the inequality constraint and the sequence $\{\mathcal{X}^k\}$ produced by our algorithm is guaranteed to satisfy the inequality constraint.

1.1. The Proposed Model

In this section, we propose the minimum and maximum nuclear norm model as follows:

$$\begin{aligned} &\min_{\mathcal{X}} \left\{ \alpha_1 \min_{1\leq i\leq N} \|\mathcal{X}_{(i)}\|_* + \alpha_2 \max_{1\leq i\leq N} \|\mathcal{X}_{(i)}\|_* \right\} \\ &\text{s.t.} \quad \|P_\Omega(\mathcal{X}) - P_\Omega(\mathcal{T})\|_F \leq \delta, \end{aligned} \tag{1.4}$$

where $\mathcal{T}, \mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, $\alpha_1 + \alpha_2 = 1, \alpha_1, \alpha_2 \geq 0$. This model does not need to complete the SVDs of all modes in each iteration, but it needs to compute nuclear norm of all mode.

Here, one might ask if the model well characterizes the rank of tensors or not? Actually, since the Tucker rank of a tensor is a vector, computing the rank of a general tensor (mode number > 2) is an NP hard problem. Therefore, there is no explicit expression for the tightest convex envelop of the tensor rank so far as pointed out in the literature [19]. The sum of nuclear norm is only an approximation to Tucker rank of tensor. On the other hand, minimizing Tucker rank of a tensor is a multi-objective optimization. As we known, there are many methods to transform multi-objective optimization into single objective optimization such as the weighting sum, maximation, etc. Liu *et al.* [19] used the weighting sum of nuclear norms of all mode, we use the weighting sum of minimum and maximum nuclear norms. Hence, the model (1.3) and (1.4) are only different approximation forms to tensor rank. We mainly focus on convergence analysis and numerical performance of the new model in this paper.

In order to use the alternating direction method, we introduce a new tensor $\mathcal{Y}$. Thus, the problem (1.4) is replaced by the following separable optimization problem:

$$\begin{aligned} &\min_{\mathcal{X},\mathcal{Y}} \left\{ \alpha_1 \min_{1 \leq i \leq N} \|\mathcal{X}_{(i)}\|_* + \alpha_2 \max_{1 \leq i \leq N} \|\mathcal{Y}_{(i)}\|_* \right\} \\ &\text{s.t.} \quad \left\| \frac{1}{2}\big(P_\Omega(\mathcal{X}) + P_\Omega(\mathcal{Y})\big) - P_\Omega(\mathcal{T}) \right\|_F \leq \delta, \\ &\qquad \mathcal{X} = \mathcal{Y}. \end{aligned} \tag{1.5}$$

The above optimization model increases to compare the nuclear norm of all mode matrices, but the algorithm complexity analysis show that its computation complexity is still less than the computation complexity of (1.3) at the end of the Section 2.

1.2. Notation

For convenience, $R^{I_1 \times I_2}$ denotes the whole of a real matrix of $I_1 \times I_2, \mathcal{X} \in R^{I_1 \times I_2 \times \cdots \times I_N}$ denotes a tensor. Its elements are denoted as $x_{i_1, i_2, \ldots, i_N}$, where $1 \leq i_k \leq I_k, 1 \leq k \leq N$. The Tucker rank of tensor $\mathcal{X}$ can be expressed as

$$\text{rank}_T(\mathcal{X}) = \big(\text{rank}(\mathcal{X}_{(1)}), \text{rank}(\mathcal{X}_{(2)}), \cdots, \text{rank}(\mathcal{X}_{(N)})\big).$$

$P_\Omega(\cdot)$ denotes the orthogonal projection operator onto an index set Ω, and

$$\big(P_\Omega(\mathcal{X})\big)_{i_1, i_2 \ldots, i_N} = \begin{cases} x_{i_1, i_2, \ldots, i_N}, & \text{if} \quad (i_1, i_2, \ldots, i_N) \in \Omega, \\ 0, & \text{if} \quad (i_1, i_2, \ldots, i_N) \in \overline{\Omega}, \end{cases}$$

where $\overline{\Omega}$ is the complementary set of Ω.

Definition 1.1 (Singular Value Decomposition, [10]). *The singular value of a matrix $X \in R^{n_1 \times n_2}$ with rank r is decomposed into*

$$X = U \Sigma_r V^\top, \quad \Sigma_r = \text{diag}(\sigma_1, \cdots, \sigma_r),$$

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

Definition 1.2 (Singular Value Shrink Operator, [4]). *For any parameter $\tau \geq 0$, singular value shrink operator D_τ is defined as*

$$D_\tau(X) := UD_\tau(\Sigma)V^\top, \quad D_\tau(\Sigma) = \operatorname{diag}(\{\sigma_i - \tau\}_+),$$

where

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

For a N-order tensor $\mathcal{X} \in R^{I_1 \times I_2 \times \cdots \times I_N}$,

$$\operatorname{unfold}_k(\mathcal{X}) = \mathcal{X}_{(k)} \in R^{I_k \times (\prod_{j \neq k} I_j)}$$

denotes the mode-k unfolding of tensor $\mathcal{X}$, the inverse operator $\operatorname{fold}_k(\mathcal{X}_{(k)}) = \mathcal{X}$ (it is also abbreviated that $\operatorname{fold}(\mathcal{X}_{(k)}) = \mathcal{X}$). The inner product of two tensor with the same order and same dimension is denoted by

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i_1, i_2, \ldots, i_N} \mathcal{X}_{i_1, i_2, \ldots, i_N} \mathcal{Y}_{i_1, i_2, \ldots, i_N},$$

and the F-norm of the tensor is expressed as $\|\mathcal{X}\|_F = \langle \mathcal{X}, \mathcal{X} \rangle^{1/2}$.

1.3. Organization

The rest of the article is organized as follows. Section 2 gives the new algorithm for solving the minimum and maximum optimization model with noise in detail, Section 3 gives the necessary condition and sufficient condition of the optimal solution of the optimization problem (1.4), Section 4 discusses the convergence theory of the proposed algorithm for solving the optimization problem (1.5), Section 5 compares the proposed algorithm with the several state-of-the-art tensor completion methods by numerical experiments. Finally, Section 6 summarizes the full text.

2. The Proposed Algorithm

In this section, we use the proximal gradient (PG) type algorithms and the feasible direction algorithm to approximately solve the problem (1.5). Here, we use partly regular function, other regular functions see [21]. Thus, the optimization problem (1.5) is approached by the following constraint optimization:

$$\begin{aligned} &\min_{\mathcal{X}, \mathcal{Y}} \lambda \left\{ \frac{1}{2} \min_{1 \leq i \leq N} \|\mathcal{X}_{(i)}\|_* + \frac{1}{2} \max_{1 \leq i \leq N} \|\mathcal{Y}_{(i)}\|_* \right\} + \frac{1}{2} \|\mathcal{X} - \mathcal{Y}\|_F{}^2 \\ &\text{s.t.} \quad \left\| \frac{1}{2}\big(P_\Omega(\mathcal{X}) + P_\Omega(\mathcal{Y})\big) - P_\Omega(\mathcal{T}) \right\|_F \leq \delta. \end{aligned} \tag{2.1}$$

For convenience, we rewrite it as

$$F(\mathcal{X}, \mathcal{Y}, \lambda) = f(\mathcal{X}, \mathcal{Y}) + g(\mathcal{X}, \mathcal{Y}),$$

where

$$f(\mathcal{X}, \mathcal{Y}) = \frac{\lambda}{2}\big(f_1(\mathcal{X}) + f_2(\mathcal{Y})\big), \quad f_1(\mathcal{X}) = \min_{1 \leq i \leq N} \|\mathcal{X}_{(i)}\|_*,$$

$$f_2(\mathcal{Y}) = \max_{1 \leq i \leq N} \|\mathcal{Y}_{(i)}\|_*, \qquad g(\mathcal{X}, \mathcal{Y}) = \frac{1}{2} \|\mathcal{X} - \mathcal{Y}\|_F{}^2.$$

Since the programming (2.1) is nonconvex and nonsmooth, we solve the sub-problem by the following convex programming:

$$
\begin{aligned}
&\min_{\mathcal{X}} \frac{\lambda^k}{2}\|\mathcal{X}_{(k,\min)}\|_* + g(\mathcal{X},\mathcal{Y}^k)\\
&\text{s.t.} \quad \left\|\frac{1}{2}\big(P_\Omega(\mathcal{X}) + P_\Omega(\mathcal{Y}^k)\big) - P_\Omega(\mathcal{T})\right\|_F \le \delta,
\end{aligned} \tag{2.2}
$$

$$
\begin{aligned}
&\min_{\mathcal{Y}} \frac{\lambda^k}{2}\|\mathcal{Y}_{(k,\max)}\|_* + g(\tilde{\mathcal{X}}^{k+1},\mathcal{Y})\\
&\text{s.t.} \quad \left\|\frac{1}{2}\big(P_\Omega(\tilde{\mathcal{X}}^{k+1}) + P_\Omega(\mathcal{Y})\big) - P_\Omega(\mathcal{T})\right\|_F \le \delta,
\end{aligned} \tag{2.3}
$$

where

$$
(k,\min) \in \Big\{\arg_i \min_{1\le i\le N} \|\mathcal{X}^k_{(i)}\|_*\Big\}, \quad (k,\max) \in \Big\{\arg_j \max_{1\le j\le N} \|\mathcal{Y}^k_{(j)}\|_*\Big\},
$$

$\tilde{\mathcal{X}}^{k+1}$ is the optimal solution of (2.2). Obviously, two sub-problems are strong convex programming, they exist the unique optimal solution separately.

Because the constraint is a nonlinear inequality, the exact optimal solution is difficult to be obtained, we compute an approximate optimal solution $(\mathcal{X}^{k+1},\mathcal{Y}^{k+1})$ as follows:

$$
\widetilde{\mathcal{X}}^{k+1} = \operatorname*{argmin}_{\mathcal{X}} \Big\{\frac{\lambda^k}{2}\|\mathcal{X}_{(k,\min)}\|_* + g(\mathcal{X},\mathcal{Y}^k)\Big\}, \tag{2.4}
$$

$$
\widetilde{\mathcal{Y}}^{k+1} = \operatorname*{argmin}_{\mathcal{Y}} \Big\{\frac{\lambda^k}{2}\|\mathcal{Y}_{(k,\max)}\|_* + g(\widetilde{\mathcal{X}}^{k+1},\mathcal{Y})\Big\}. \tag{2.5}
$$

If

$$
\beta^{k+1} = \frac{\delta}{\big\|\big(P_\Omega\widetilde{\mathcal{X}}^{k+1} + P_\Omega(\widetilde{\mathcal{Y}}^{k+1})\big)/2 - P_\Omega(\mathcal{T})\big\|_F} \ge 1,
$$

then

$$
\mathcal{X}^{k+1} = \widetilde{\mathcal{X}}^{k+1}, \quad \mathcal{Y}^{k+1} = \widetilde{\mathcal{Y}}^{k+1},
$$

otherwise,

$$
\begin{aligned}
\mathcal{X}^{k+1} &= \widetilde{\mathcal{X}}^{k+1} + (1-\beta^{k+1})P_\Omega(\mathcal{T} - \widetilde{\mathcal{X}}^{k+1}),\\
\mathcal{Y}^{k+1} &= \widetilde{\mathcal{Y}}^{k+1} + (1-\beta^{k+1})P_\Omega(\mathcal{T} - \widetilde{\mathcal{Y}}^{\mathrm{k}+1}).
\end{aligned}
$$

Thus, we give the Algorithm 2.1 in detail.

Analysis of the complexity: Suppose that $I_i \le \prod_{j\ne i} I_j, M = \prod_{j=1}^N I_j$ and the computational complexity $\mathcal{O}(I_i^2) + \mathcal{O}(I_i), i = 1,2,\dots,N$ is ignored.

(1) Computing $\mathcal{X}_{(i)}\mathcal{X}_{(i)}^\top$, $i = 1,2,\dots,N$ take about $\sum_{i=1}^N MI_i$;

(2) Computing all singular value takes about $\sum_{i=1}^N 8I_i^3/3$;

(3) Two singular value decomposition and shrink processing take about $4M^2/I_{i_1} + 9MI_{i_1} + 9I_{i_1}^3 + 4M^2/I_{i_2} + 9MI_{i_2} + 9I_{i_2}^3$, $I_{i_1} \times \prod_{j\ne i_1} I_j, I_{i_2} \times \prod_{j\ne i_2} I_j$ are row and column of the minimum and maximum mode matrices, respectively.

Thus, total computation complexity is about $\sum_{i=1}^N (MI_i + 8I_i^3/3) + 4M^2/I_{i_1} + 9MI_{i_1} + 9I_{i_1}^3 + 4M^2/I_{i_2} + 9MI_{i_2} + 9I_{i_2}^3$. However, the traditional weighting nuclear norm algorithm takes about $\sum_{i=1}^N (4M^2/I_i + 9MI_i + 9I_i^3)$. Obviously, which is higher than the complexity of our algorithm under $N \ge 3$ (on computing all singular value and SVD, see [31, Note 3.2.3, Third Lecture, pp. 116]).

Algorithm 2.1: The MIN-MAX algorithm for solving (2.1).

Input : Index set $\Omega, P_\Omega(\mathcal{T})$, parameters $\lambda^0 > 0$, $t^0 = 1$, $\varepsilon > 0$, $1 > \rho > 0$ and an integer Maxiter.

Initialize $\mathcal{X}^{-1} = \mathcal{X}^0 = \mathbf{0}$ (the zero tensor).

for $k = 0$: Maxiter **do**

  Choose an i_k such that $\|\mathcal{X}^k_{(i_k)}\|_* = \min_{1\le i\le N} \|\mathcal{X}^k_{(i)}\|_*$.

  $\mathcal{Y}^k_{(i_k)} = U^k \Sigma_{\mathcal{Y}^k} (V^k)^\top$.

  $\widetilde{\mathcal{X}}^{k+1} = \text{fold}_{i_k}(U^k D_{\lambda^k/2}(\Sigma_{\mathcal{Y}^k})(V^k)^\top)$.

  Choose a j_k such that $\|\mathcal{Y}^k_{(j_k)}\|_* = \max_{1\le j\le N} \|\mathcal{Y}^k_{(j)}\|_*$.

  $\widetilde{\mathcal{X}}^{k+1}_{(j_k)} = U^{k+1} \Sigma_{\tilde{\mathcal{X}}^{k+1}} (V^{k+1})^\top$.

  $\widetilde{\mathcal{Y}}^{k+1} = \text{fold}_{j_k}(U^{k+1} D_{\lambda^k/2}(\Sigma_{\tilde{\mathcal{X}}^{k+1}})(V^{k+1})^\top)$.

  $\beta^{k+1} = \dfrac{\delta}{\|(P_\Omega(\widetilde{\mathcal{X}}^{k+1}) + P_\Omega(\widetilde{\mathcal{Y}}^{k+1}))/2 - P_\Omega(\mathcal{T})\|_F}$.

  When $\beta^{k+1} \ge 1$, $\mathcal{X}^{k+1} = \widetilde{\mathcal{X}}^{k+1}$, $\mathcal{Y}^{k+1} = \widetilde{\mathcal{Y}}^{k+1}$

   Else $\mathcal{X}^{k+1} = P_{\overline{\Omega}}(\widetilde{\mathcal{X}}^{k+1}) + \beta^{k+1} P_\Omega(\widetilde{\mathcal{X}}^{k+1}) + (1-\beta^{k+1})P_\Omega(\mathcal{T})$,

   $\mathcal{Y}^{k+1} = P_{\overline{\Omega}}(\widetilde{\mathcal{Y}}^{k+1}) + \beta^{k+1} P_\Omega(\widetilde{\mathcal{Y}}^{k+1}) + (1-\beta^{k+1})P_\Omega(\mathcal{T})$.

  If $\|\mathcal{X}^{k+1} - \mathcal{Y}^{k+1}\|_F / P_\Omega(\mathcal{T}) < \varepsilon$, break the for loop, $\mathcal{U}^{k+1} = (\mathcal{X}^{k+1} + \mathcal{Y}^{k+1})/2$

   Otherwise, set $\lambda^{k+1} = \rho\lambda^k$.

end

Output: $\mathcal{U}^{k+1}$.

3. Optimality Condition

In this section, we discuss the optimality condition including the necessary condition and sufficient condition of optimal solution and the stationary point. Since the constraint is a quadratic inequality, we introduce some definition and property on the convex cone and conjugate cone. We define convex cone in the following:

$$S(\mathcal{X}^*) = \{\beta P_\Omega(\mathcal{X} - \mathcal{X}^*) \,|\, \mathcal{X} \in \mathcal{D},\ \mathcal{X}^* \in \widetilde{\mathcal{D}}\}, \tag{3.1}$$

where $\beta \ge 0$,

$$\mathcal{D} = \left\{(\mathcal{X}, \mathcal{Y}) \,\middle|\, \left\|\frac{1}{2}\big(P_\Omega(\mathcal{X}) + P_\Omega(\mathcal{Y})\big) - P_\Omega(\mathcal{T})\right\|_F \le \delta\right\},$$

and its boundary

$$\widetilde{\mathcal{D}} = \left\{(\mathcal{X}, \mathcal{Y}) \,\middle|\, \left\|\frac{1}{2}\big(P_\Omega(\mathcal{X}) + P_\Omega(\mathcal{Y})\big) - P_\Omega(\mathcal{T})\right\|_F = \delta\right\}.$$

$S^*(\mathcal{X}^*)$ denotes the conjugate cone of $S(\mathcal{X}^*)$, which is defined as follows (see [3]):

$$S^*(\mathcal{X}^*) = \{\mathcal{Z} \,|\, \langle \mathcal{X}, \mathcal{Z}\rangle \ge 0,\ \mathcal{X} \in S(\mathcal{X}^*)\}.$$

We definite the active set

$$I_1(\mathcal{X}) = \left\{i \mid \|\mathcal{X}_{(i)}\|_* = \min_{1\le j\le N} \|\mathcal{X}_{(j)}\|_*\right\}, \quad I_2(\mathcal{X}) = \left\{i \mid \|\mathcal{X}_{(i)}\|_* = \max_{1\le j\le N} \|\mathcal{X}_{(j)}\|_*\right\}.$$

Meanwhile, we introduce the subgradient of nuclear norm as follows, more nonsmooth analyses see [8, 26].

Lemma 3.1 (Subgradient of the Matrix Trace Norm, [30]). *Let X be a real $m \times n$ matrix with* $\mathrm{rank}(X) = r$. *Then, the subdifferential of the nuclear norm $\|X\|_*$ is given by*

$$\partial\|X\|_* = \{UV^\top + W \mid U^\top W = 0,\ WV = 0,\ \|W\|_2 \leq 1\},$$

where $U \in \mathbb{R}^{m\times r}$ and $V \in \mathbb{R}^{n\times r}$ have orthogonal columns formed by the left and the right singular vectors of X, respectively.

Theorem 3.1. *Assume $\mathcal{X}^* \in \mathcal{D}$. $\mathcal{X}^*$ is a global optimal solution of the optimization problem* (1.4)*, if the following conditions hold:*

(i) $\mathcal{X}^* \in \mathcal{D} - \widetilde{\mathcal{D}}$ *and* $0 \in \mathrm{fold}(\partial\|\mathcal{X}^*_{(i)}\|_*),\ i = 1, 2, \ldots, N.$

(ii) $\mathcal{X}^* \in \widetilde{\mathcal{D}}$, $0 \in P_{\overline{\Omega}}(\mathrm{fold}(\partial\|\mathcal{X}^*_{(i)}\|_*))$ *and* $P_\Omega(\mathrm{fold}(\partial\|\mathcal{X}^*_{(i)}\|_*)) \in S^*(\mathcal{X}^*),\ i = 1, 2, \ldots, N.$

Proof. We prove $f_1(\mathcal{X}) + f_2(\mathcal{X}) \geq f_1(\mathcal{X}^*) + f_2(\mathcal{X}^*)$ for any $\mathcal{X} \in \mathcal{D}$.

(i) Let $\mathcal{X}^* \in \mathcal{D} - \widetilde{\mathcal{D}}$ and $i \in I_1(\mathcal{X}), j \in I_2(\mathcal{X}^*)$, we have

$$\begin{aligned}
& f_1(\mathcal{X}) + f_2(\mathcal{X}) \\
\geq\ & \|\mathcal{X}_{(i)}\|_* + \|\mathcal{X}_{(j)}\|_* \\
\geq\ & \big\|\mathcal{X}^*_{(i)}\big\|_* + \big\|\mathcal{X}^*_{(j)}\big\|_* + \big\langle \partial\big\|\mathcal{X}^*_{(i)}\big\|_*, \mathcal{X}_{(i)} - \mathcal{X}^*_{(i)}\big\rangle + \big\langle \partial\big\|\mathcal{X}^*_{(j)}\big\|_*, \mathcal{X}_{(j)} - \mathcal{X}^*_{(j)}\big\rangle \\
=\ & \big\|\mathcal{X}^*_{(i)}\big\|_* + \big\|\mathcal{X}^*_{(j)}\big\|_* + \big\langle \mathrm{fold}\big(\partial\big\|\mathcal{X}^*_{(i)}\big\|_*\big) + \mathrm{fold}\big(\partial\big\|\mathcal{X}^*_{(j)}\big\|_*\big), \mathcal{X} - \mathcal{X}^*\big\rangle \\
\geq\ & \big\|\mathcal{X}^*_{(i)}\big\|_* + \big\|\mathcal{X}^*_{(j)}\big\|_* \geq f_1(\mathcal{X}^*) + f_2(\mathcal{X}^*).
\end{aligned}$$

The first inequality is from the definitions of $f_1(\mathcal{X})$ and $f_2(\mathcal{X})$, the second inequality follows from convex function, the third inequality is due to the assumption (i), the last inequality is from the definitions of $f_1(\mathcal{X}^*)$ and $f_2(\mathcal{X}^*)$.

(ii) Let $i \in I_1(\mathcal{X}), j \in I_2(\mathcal{X}^*)$, we have

$$\begin{aligned}
& f_1(\mathcal{X}) + f_2(\mathcal{X}) \\
\geq\ & \|\mathcal{X}_{(i)}\|_* + \|\mathcal{X}_{(j)}\|_* \\
\geq\ & \big\|\mathcal{X}^*_{(i)}\big\|_* + \big\|\mathcal{X}^*_{(j)}\big\|_* + \big\langle \partial\big\|\mathcal{X}^*_{(i)}\big\|_*, \mathcal{X}_{(i)} - \mathcal{X}^*_{(i)}\big\rangle + \big\langle \partial\big\|\mathcal{X}^*_{(j)}\big\|_*, \mathcal{X}_{(j)} - \mathcal{X}^*_{(j)}\big\rangle \\
=\ & \big\|\mathcal{X}^*_{(i)}\big\|_* + \big\|\mathcal{X}^*_{(j)}\big\|_* + \big\langle \mathrm{fold}\big(\partial\big\|\mathcal{X}^*_{(i)}\big\|_*\big) + \mathrm{fold}\big(\partial\big\|\mathcal{X}^*_{(j)}\big\|_*\big), \mathcal{X} - \mathcal{X}^*\big\rangle \\
=\ & \big\|\mathcal{X}^*_{(i)}\big\|_* + \big\|\mathcal{X}^*_{(j)}\big\|_* \\
& + \big\langle \mathrm{fold}\big(\partial\big\|\mathcal{X}^*_{(i)}\big\|_*\big) + \mathrm{fold}\big(\partial\big\|\mathcal{X}^*_{(j)}\big\|_*\big), P_{\overline{\Omega}}(\mathcal{X} - \mathcal{X}^*) + P_\Omega(\mathcal{X} - \mathcal{X}^*)\big\rangle \\
=\ & \big\|\mathcal{X}^*_{(i)}\big\|_* + \big\|\mathcal{X}^*_{(j)}\big\|_* + \big\langle P_{\overline{\Omega}}\big(\mathrm{fold}\big(\partial\big\|\mathcal{X}^*_{(i)}\big\|_*\big) + \mathrm{fold}\big(\partial\big\|\mathcal{X}^*_{(j)}\big\|_*\big)\big), P_{\overline{\Omega}}(\mathcal{X} - \mathcal{X}^*)\big\rangle \\
& + \big\langle P_\Omega\big(\mathrm{fold}\big(\partial\big\|\mathcal{X}^*_{(i)}\big\|_*\big) + \mathrm{fold}\big(\partial\big\|\mathcal{X}^*_{(j)}\big\|_*\big)\big), P_\Omega(\mathcal{X} - \mathcal{X}^*)\big\rangle \\
\geq\ & \big\|\mathcal{X}^*_{(i)}\big\|_* + \big\|\mathcal{X}^*_{(j)}\big\|_* \geq f_1(\mathcal{X}^*) + f_2(\mathcal{X}^*).
\end{aligned}$$

The third inequality is from assumption (ii). Thus, we obtain the theorem. □

Theorem 3.2. *Assume $\mathcal{X}^* \in \mathcal{D}$. $\mathcal{X}^*$ is a local optimal solution of the optimization problem* (1.4)*, if the following conditions hold:*

(i) $\mathcal{X}^* \in \mathcal{D} - \widetilde{\mathcal{D}}$ *and* $0 \in \mathrm{fold}(\partial\|\mathcal{X}^*_{(i)}\|_*),\ i \in I_1(\mathcal{X}^*) \cup I_2(\mathcal{X}^*)$.

(ii) $\mathcal{X}^* \in \widetilde{\mathcal{D}},\ 0 \in P_{\overline{\Omega}}(\mathrm{fold}(\partial\|\mathcal{X}^*_{(i)}\|_*))$ *and* $P_\Omega(\mathrm{fold}(\partial\|\mathcal{X}^*_{(i)}\|_*)) \in S^*(\mathcal{X}^*),\ i \in I_1(\mathcal{X}^*) \cup I_2(\mathcal{X}^*)$.

Proof. If there exists a neighborhood $O(\mathcal{X}^*, \delta)$ of $\mathcal{X}^*$ such that

$$I_1(\mathcal{X}) \subset I_1(\mathcal{X}^*), \quad I_2(\mathcal{X}) \subset I_2(\mathcal{X}^*), \quad \forall\, \mathcal{X} \in O(\mathcal{X}^*, \delta),$$

then we use the same proof technique as Theorem 3.1 to prove

$$f_1(\mathcal{X}) + f_2(\mathcal{X}) \geq f_1(\mathcal{X}^*) + f_2(\mathcal{X}^*), \quad \forall\, \mathcal{X} \in O(\mathcal{X}^*, \delta).$$

From the definition of $I_1(\mathcal{X}^*)$, we have $\|\mathcal{X}^*_{(j)}\|_* > \|\mathcal{X}^*_{(i)}\|_*,\ j \neq i,\ i \in I_1(\mathcal{X}^*)$. Thus, there exists a neighborhood $O(\mathcal{X}^*, \delta_1)$ of $\mathcal{X}^*$ such that $\|\mathcal{X}_{(j)}\|_* > \|\mathcal{X}_{(i)}\|_*, \mathcal{X} \in O(\mathcal{X}^*, \delta_1)$, since the function $\|\mathcal{X}_{(i)}\|_*$ is a continuous function. So from the definition of $I_1(\mathcal{X})$ we have $j \notin I_1(\mathcal{X})$, which implies $I_1(\mathcal{X}) \subset I_1(\mathcal{X}^*), \mathcal{X} \in O(\mathcal{X}^*, \delta_1)$. Similarly, there exists another neighborhood $O(\mathcal{X}^*, \delta_2)$ of $\mathcal{X}^*$ such that $I_2(\mathcal{X}) \subset I_2(\mathcal{X}^*), \mathcal{X} \in O(\mathcal{X}^*, \delta_2)$. Thus, let $\delta = \min\{\delta_1, \delta_2\}$, $\mathcal{X} \in O(\mathcal{X}^*, \delta)$ satisfies $I_1(\mathcal{X}) \subset I_1(\mathcal{X}^*)$ and $I_2(\mathcal{X}) \subset I_2(\mathcal{X}^*)$, Theorem is proved. □

Theorem 3.3. *The stationary point $\mathcal{X}^*$ of the optimization problem* (1.4) *satisfies*

$$0 \in \mathrm{fold}\big(\partial f_1(\mathcal{X}^*)\big) + \mathrm{fold}\big(\partial f_2(\mathcal{X}^*)\big) + \mu^* P_\Omega(\mathcal{X}^* - \mathcal{T}), \quad \mu^* \geq 0, \tag{3.2}$$

where μ^* is the Lagrange multiplier of the inequality constraint.

Proof. The Lagrange function of the programming (1.4) is in the following:

$$L(\mathcal{X}, \mu) = \left\{\frac{1}{2}\min_{1\leq i\leq N}\|\mathcal{X}_{(i)}\|_* + \frac{1}{2}\max_{1\leq i\leq N}\|\mathcal{X}_{(i)}\|_*\right\} + \mu\big(\|P_\Omega(\mathcal{X}) - P_\Omega(\mathcal{T})\|_F - \delta\big),$$

where μ is the Lagrange multiplier. As we know, the stationary point $\mathcal{X}^*$ satisfies $0 \in \partial L(\mathcal{X}, \mu)/\partial\mathcal{X}|_{\mathcal{X}=\mathcal{X}^*}$. So

$$\begin{aligned} 0 \in \frac{\partial L(\mathcal{X}, \mu)}{\partial \mathcal{X}}\Big|_{\mathcal{X}=\mathcal{X}^*} &= \frac{1}{2}\Big(\mathrm{fold}\big(\partial f_1(\mathcal{X}^*)\big) + \mathrm{fold}\big(\partial f_2(\mathcal{X}^*)\big)\Big) \\ &\quad + \frac{\mu}{\|P_\Omega(\mathcal{X}^* - \mathcal{T})\|_F} P_\Omega(\mathcal{X}^* - \mathcal{T}), \quad \mu \geq 0. \end{aligned}$$

Let $\mu^* = 2\mu/\|P_\Omega(\mathcal{X}^* - \mathcal{T})\|_F \geq 0$. We obtain the theorem. □

4. Convergence Analysis

In this section, we establish the convergence of Algorithm 2.1. First, the objective function is proved to be coercible and descent, then convergence theory is established based on the optimal condition.

Lemma 4.1. *For all $\lambda > 0$, $F(\mathcal{X}, \mathcal{Y}, \lambda) \to +\infty$ as $\mathcal{X}, \mathcal{Y} \to \infty$.*

Proof. Since $\mathcal{X}, \mathcal{Y}$ satisfy the constraint condition, we only consider $\mathcal{P}_{\overline{\Omega}}(\mathcal{X}), \mathcal{P}_{\overline{\Omega}}(\mathcal{Y}) \to \infty$. For convenience, the fixed $\mathcal{X}$ and $\mathcal{Y}$ satisfy $\mathcal{P}_{\overline{\Omega}}(\mathcal{X}) \neq 0, \mathcal{P}_{\overline{\Omega}}(\mathcal{Y}) \neq 0$, we observe $c\mathcal{P}_{\overline{\Omega}}(\mathcal{X}) \neq 0$, $c\mathcal{P}_{\overline{\Omega}}(\mathcal{Y}) \neq 0$ when $c \to +\infty$.

$$\begin{aligned} F(cX, cY, \lambda) \geq \lambda\bigg(& \frac{1}{2}\min_{1\leq i\leq N} \|cP_{\overline{\Omega}}(\mathcal{X}_{(i)}) + P_{\Omega}(\mathcal{X}_{(i)})\|_* + \frac{1}{2}\max_{1\leq i\leq N} \|cP_{\overline{\Omega}}(\mathcal{Y}_{(i)}) + P_{\Omega}(\mathcal{Y}_{(i)})\|_*\bigg) \\ & + \frac{c}{2}\|P_{\overline{\Omega}}(\mathcal{X} - \mathcal{Y})\|_F{}^2. \end{aligned}$$

Then

$$\begin{aligned} \frac{1}{2}\min_{1\leq i\leq N} \|cP_{\overline{\Omega}}(\mathcal{X}_{(i)}) + P_{\Omega}(\mathcal{X}_{(i)})\|_* &\to +\infty, \\ \frac{1}{2}\max_{1\leq i\leq N} \|cP_{\overline{\Omega}}(\mathcal{Y}_{(i)}) + P_{\Omega}(\mathcal{Y}_{(i)})\|_* &\to +\infty. \end{aligned}$$

So, $F(c\mathcal{X}, c\mathcal{Y}, \lambda) \to +\infty$, which implies the theorem holds. □

Lemma 4.2. *Let* $\{\mathcal{X}^k, \mathcal{Y}^k, \lambda^k\}$ *represent the sequence produced by the Algorithm* 2.1*. Then*

$$F(\mathcal{X}^k, \mathcal{Y}^k, \lambda^k) - F(\widetilde{\mathcal{X}}^{k+1}, \widetilde{\mathcal{Y}}^{k+1}, \lambda^k) \geq \frac{1}{2}(\|\mathcal{X}^k - \widetilde{\mathcal{X}}^{k+1}\|_F^2 + \|\mathcal{Y}^k - \widetilde{\mathcal{Y}}^{k+1}\|_F^2),$$

where $\{\mathcal{X}^k, \mathcal{Y}^k, \widetilde{\mathcal{X}}^{k+1}, \widetilde{\mathcal{Y}}^{k+1}\}$ *is generated by Algorithm* 2.1*.*

Proof. Let

$$\begin{aligned} (k, \min) \in I_1^k &= \Big\{i \mid \|\mathcal{X}_{(i)}^k\|_* = \min_{1\leq j\leq N} \|\mathcal{X}_{(j)}^k\|_*\Big\}, \\ (k, \max) \in I_2^k &= \Big\{i \mid \|\mathcal{Y}_{(i)}^k\|_* = \max_{1\leq j\leq N} \|\mathcal{Y}_{(j)}^k\|_*\Big\}, \end{aligned}$$

$\mathcal{X}_{(k,\min)}^{k+1}$ represents the $(k, \min)$-mode matrix of the $(k+1)$-th iteration tensor $\mathcal{X}^{k+1}$, $\mathcal{Y}_{(k,\max)}^{k+1}$ represents the $(k, \max)$-mode matrix of the $(k+1)$-th iteration tensor $\mathcal{Y}^{k+1}$. Then

$$\begin{aligned} & F(\mathcal{X}^k, \mathcal{Y}^k, \lambda^k) - F(\widetilde{\mathcal{X}}^{k+1}, \widetilde{\mathcal{Y}}^{k+1}, \lambda^k) \\ = & \frac{\lambda^k}{2}(\|\mathcal{X}^k{}_{(k,\min)}\|_* + \|\mathcal{Y}^k{}_{(k,\max)}\|_*) + \frac{1}{2}\|\mathcal{X}^k - \mathcal{Y}^k\|_F^2 \\ & - \frac{\lambda^k}{2}(\|\widetilde{\mathcal{X}}_{(k,\min)}^{k+1}\|_* + \|\widetilde{\mathcal{Y}}_{(k,\max)}^{k+1}\|_*) - \frac{1}{2}\|\widetilde{\mathcal{X}}^{k+1} - \widetilde{\mathcal{Y}}^{k+1}\|_F^2. \end{aligned}$$

We first compute the following difference:

$$\begin{aligned} & \frac{1}{2}\|\mathcal{X}^k - \mathcal{Y}^k\|_F^2 - \frac{1}{2}\|\widetilde{\mathcal{X}}^{k+1} - \widetilde{\mathcal{Y}}^{k+1}\|_F^2 \\ = & \frac{1}{2}(\|\mathcal{X}^k - \mathcal{Y}^k\|_F^2 - \|\widetilde{\mathcal{X}}^{k+1} - \mathcal{Y}^k\|_F^2 + \|\widetilde{\mathcal{X}}^{k+1} - \mathcal{Y}^k\|_F^2 - \|\widetilde{\mathcal{X}}^{k+1} - \widetilde{\mathcal{Y}}^{k+1}\|_F^2) \\ = & \frac{1}{2}\|\mathcal{X}^k - \widetilde{\mathcal{X}}^{k+1}\|_F^2 + \langle \mathcal{X}^k - \widetilde{\mathcal{X}}^{k+1}, \widetilde{\mathcal{X}}^{k+1} - \mathcal{Y}^k\rangle + \frac{1}{2}\|\mathcal{Y}^k - \widetilde{\mathcal{Y}}^{k+1}\|_F^2 \\ & + \langle \mathcal{Y}^k - \widetilde{\mathcal{Y}}^{k+1}, \widetilde{\mathcal{X}}^{k+1} - \widetilde{\mathcal{Y}}^{k+1}\rangle. \end{aligned}$$

On the other hand, the optimal solutions of the sub-problems (2.2) and (2.3) satisfy the following equations:

$$\begin{cases} 0 \in \dfrac{\lambda^k}{2}\text{fold}\Big(\dfrac{\partial\|\widetilde{\mathcal{X}}^{k+1}_{(k,\min)}\|_*}{\partial\mathcal{X}}\Big) + \widetilde{\mathcal{X}}^{k+1} - \mathcal{Y}^k, \\ 0 \in \dfrac{\lambda^k}{2}\text{fold}\Big(\dfrac{\partial\|\widetilde{\mathcal{Y}}^{k+1}_{(k,\max)}\|_*}{\partial\mathcal{Y}}\Big) + \widetilde{\mathcal{Y}}^{k+1} - \widetilde{\mathcal{X}}^{k+1}. \end{cases}$$

Thus there exist a subgradient $\mathcal{W}_1^{k+1} \in \text{fold}(\partial\|\widetilde{\mathcal{X}}^{k+1}_{(k,\min)}\|_*/\partial\mathcal{X})$ and a subgradient $\mathcal{W}_2^{k+1} \in \text{fold}(\partial\|\widetilde{\mathcal{Y}}^{k+1}_{(k,\max)}\|_*/\partial\mathcal{Y})$ such that

$$\begin{cases} \dfrac{\lambda^k}{2}\mathcal{W}_1^{k+1} + \widetilde{\mathcal{X}}^{k+1} - \mathcal{Y}^k = 0, \\ \dfrac{\lambda^k}{2}\mathcal{W}_2^{k+1} + \widetilde{\mathcal{Y}}^{k+1} - \widetilde{\mathcal{X}}^{k+1} = 0. \end{cases} \tag{4.1}$$

Transpose both sides of the above equations and multiply by $(\mathcal{X}^k - \widetilde{\mathcal{X}}^{k+1})$ and $(\mathcal{Y}^k - \widetilde{\mathcal{Y}}^{k+1})$, it yields

$$\begin{cases} \dfrac{\lambda^k}{2}\langle\mathcal{W}_1^{k+1}, \mathcal{X}^k - \widetilde{\mathcal{X}}^{k+1}\rangle + \langle\widetilde{\mathcal{X}}^{k+1} - \mathcal{Y}^k, \mathcal{X}^k - \widetilde{\mathcal{X}}^{k+1}\rangle = 0, \\ \dfrac{\lambda^k}{2}\langle\mathcal{W}_2^{k+1}, \mathcal{Y}^k - \widetilde{\mathcal{Y}}^{k+1}\rangle + \langle\widetilde{\mathcal{Y}}^{k+1} - \widetilde{\mathcal{X}}^{k+1}, \mathcal{Y}^k - \widetilde{\mathcal{Y}}^{k+1}\rangle = 0. \end{cases} \tag{4.2}$$

The convex function $\|\mathcal{X}_{(k,\min)}\|_*$ and $\|\mathcal{Y}_{(k,\max)}\|_*$ have the following inequalities:

$$\frac{\lambda^k}{2}\big(\|\mathcal{X}^{\text{k}}_{(\text{k,min})}\|_* - \|\widetilde{\mathcal{X}}^{\text{k+1}}_{(\text{k,min})}\|_*\big) \geq \frac{\lambda^{\text{k}}}{2}\langle\mathcal{W}_1^{\text{k+1}}, \mathcal{X}^{\text{k}}_{(\text{k,min})} - \widetilde{\mathcal{X}}^{\text{k+1}}_{(\text{k,min})}\rangle,$$
$$\frac{\lambda^k}{2}\big(\|\mathcal{Y}^{\text{k}}_{(\text{k,max})}\|_* - \|\widetilde{\mathcal{Y}}^{\text{k+1}}_{(\text{k,max})}\|_*\big) \geq \frac{\lambda^{\text{k}}}{2}\langle\mathcal{W}_2^{\text{k+1}}, \mathcal{Y}^{\text{k}}_{(\text{k,max})} - \widetilde{\mathcal{Y}}^{\text{k+1}}_{(\text{k,max})}\rangle.$$

Combining above equations and inequalities, we have

$$\begin{aligned} & F(\mathcal{X}^k, \mathcal{Y}^k, \lambda^k) - F(\widetilde{\mathcal{X}}^{k+1}, \widetilde{\mathcal{Y}}^{k+1}, \lambda^k) \\ = & \frac{1}{2}\big(\|\mathcal{X}^k - \widetilde{\mathcal{X}}^{k+1}\|_F^2 + \|\mathcal{Y}^k - \widetilde{\mathcal{Y}}^{k+1}\|_F^2\big) \\ & + \frac{\lambda^k}{2}\Big(\|\mathcal{X}^k{}_{(k,\min)}\|_* - \|\widetilde{\mathcal{X}}^{\text{k+1}}_{(\text{k,min})}\|_* - \langle\mathcal{W}_1^{\text{k+1}}, \mathcal{X}^{\text{k}}_{(\text{k,min})} - \widetilde{\mathcal{X}}^{\text{k+1}}_{(\text{k,min})}\rangle \\ & \qquad + \|\mathcal{Y}^k{}_{(k,\max)}\|_* - \|\widetilde{\mathcal{Y}}^{\text{k+1}}_{(\text{k,max})}\|_* - \langle\mathcal{W}_2^{\text{k+1}}, \mathcal{Y}^{\text{k}}_{(\text{k,max})} - \widetilde{\mathcal{Y}}^{\text{k+1}}_{(\text{k,max})}\rangle\Big) \\ \geq & \frac{1}{2}\big(\|\mathcal{X}^k - \widetilde{\mathcal{X}}^{k+1}\|_F^2 + \|\mathcal{Y}^k - \widetilde{\mathcal{Y}}^{k+1}\|_F^2\big). \end{aligned}$$

The inequality is from the above inequality of the convex function $\|\mathcal{X}_{(k,\min)}\|_*$ and $\|\mathcal{Y}_{(k,\max)}\|_*$. So the theorem holds. □

Lemma 4.3. *Let* $\{\mathcal{X}^k, \mathcal{Y}^k, \lambda^k\}$ *represent the sequence produced by the Algorithm* 2.1 *and let*

$$\lambda^{k+1}\big(\|\mathcal{X}^{k+1}_{(k+1,\min)}\|_* + \|\mathcal{Y}^{k+1}_{(k+1,\max)}\|_*\big) \leq \lambda^k\big(\|\widetilde{\mathcal{X}}^{k+1}_{(k,\min)}\|_* + \|\widetilde{\mathcal{Y}}^{k+1}_{(k,\max)}\|_*\big). \tag{4.3}$$

Then the following inequality:

$$F(\mathcal{X}^k, \mathcal{Y}^k, \lambda^k) - F(\mathcal{X}^{k+1}, \mathcal{Y}^{k+1}, \lambda^{k+1})$$

$$\geq \frac{1}{2}\big(\big\|P_{\overline{\Omega}}(\mathcal{X}^k-\mathcal{X}^{k+1})\big\|_F^2+\big\|P_{\overline{\Omega}}(\mathcal{Y}^k-\mathcal{Y}^{k+1})\big\|_F^2\big)+\frac{\delta^2(1-\beta^{k+1})^2}{(\beta^{k+1})^2} \tag{4.4}$$

holds.

Proof. By Lemma 4.2, we have the following inequality:

$$F(\mathcal{X}^k,\mathcal{Y}^k,\lambda^k)-F(\widetilde{\mathcal{X}}^{k+1},\widetilde{\mathcal{Y}}^{k+1},\lambda^k)\geq \frac{1}{2}\big(\|\mathcal{X}^k-\widetilde{\mathcal{X}}^{k+1}\|_F^2+\|\mathcal{Y}^k-\widetilde{\mathcal{Y}}^{k+1}\|_F^2\big).$$

Meanwhile, we compute the difference

$$\begin{aligned}
&F(\widetilde{\mathcal{X}}^{k+1},\widetilde{\mathcal{Y}}^{k+1},\lambda^k)-F(\mathcal{X}^{k+1},\mathcal{Y}^{k+1},\lambda^{k+1})\\
=&\frac{\lambda^k}{2}\big(\big\|\widetilde{\mathcal{X}}^{k+1}_{(k,\min)}\big\|_*+\big\|\widetilde{\mathcal{Y}}^{k+1}_{(k,\max)}\big\|_*\big)+\frac{1}{2}\|\widetilde{\mathcal{X}}^{k+1}-\widetilde{\mathcal{Y}}^{k+1}\|_F^2\\
&-\frac{\lambda^{k+1}}{2}\big(\big\|\mathcal{X}^{k+1}{}_{(k+1,\min)}\big\|_*+\big\|\mathcal{Y}^{k+1}{}_{(k+1,\max)}\big\|_*\big)-\frac{1}{2}\|\mathcal{X}^{k+1}-\mathcal{Y}^{k+1}\|_F^2\\
\geq&\frac{1}{2}\big(\|\widetilde{\mathcal{X}}^{k+1}-\widetilde{\mathcal{Y}}^{k+1}\|_F^2-\|\mathcal{X}^{k+1}-\mathcal{Y}^{k+1}\|_F^2\big),
\end{aligned}$$

the inequality is from the condition (4.3).

(i) If $\|P_\Omega(\widetilde{\mathcal{X}}^{k+1}+\widetilde{\mathcal{Y}}^{k+1})/2-P_\Omega(\mathcal{T})\|_F\leq\delta$. From Algorithm 2.1 we have $\mathcal{X}^{k+1}=\widetilde{\mathcal{X}}^{k+1}$, $\mathcal{Y}^{k+1}=\widetilde{\mathcal{Y}}^{k+1}$. Then the Lemma 4.3 holds.

(ii) If $\|P_\Omega(\widetilde{\mathcal{X}}^{k+1}+\widetilde{\mathcal{Y}}^{k+1})/2-P_\Omega(\mathcal{T})\|_F>\delta$. From Algorithm 2.1 we have

$$\begin{aligned}
\mathcal{X}^{k+1}&=\widetilde{\mathcal{X}}^{k+1}+(1-\beta^{k+1})\big(P_\Omega(\mathcal{T})-P_\Omega(\widetilde{\mathcal{X}}^{k+1})\big),\\
\mathcal{Y}^{k+1}&=\widetilde{\mathcal{Y}}^{k+1}+(1-\beta^{k+1})\big(P_\Omega(\mathcal{T})-P_\Omega(\widetilde{\mathcal{Y}}^{k+1})\big),
\end{aligned}$$

where

$$\beta^{k+1}=\frac{\delta}{\big\|P_\Omega(\widetilde{\mathcal{X}}^{k+1}+\widetilde{\mathcal{Y}}^{k+1})/2-P_\Omega(\mathcal{T})\big\|_F}.$$

Further, the following inequalities:

$$\begin{aligned}
&F(\widetilde{\mathcal{X}}^{k+1},\widetilde{\mathcal{Y}}^{k+1},\lambda^k)-F(\mathcal{X}^{k+1},\mathcal{Y}^{k+1},\lambda^{k+1})\\
\geq&\frac{1}{2}\big(\|\widetilde{\mathcal{X}}^{k+1}-\widetilde{\mathcal{Y}}^{k+1}\|_F^2-\|\mathcal{X}^{k+1}-\mathcal{Y}^{k+1}\|_F^2\big)\\
=&\frac{1}{2}\big(1-(\beta^{k+1})^2\big)\big\|P_\Omega(\widetilde{\mathcal{X}}^{k+1})-P_\Omega(\widetilde{\mathcal{Y}}^{k+1})\big\|_F^2\geq 0
\end{aligned}$$

hold.

Next, we compute the difference of $\|\mathcal{X}^k-\widetilde{\mathcal{X}}^{k+1}\|_F$. Since

$$\begin{aligned}
\mathcal{X}^k-\widetilde{\mathcal{X}}^{k+1}&=P_{\overline{\Omega}}(\mathcal{X}^k-\mathcal{X}^{k+1})+P_\Omega(\mathcal{X}^k-\widetilde{\mathcal{X}}^{k+1}),\\
\mathcal{Y}^k-\widetilde{\mathcal{Y}}^{k+1}&=P_{\overline{\Omega}}(\mathcal{Y}^k-\mathcal{Y}^{k+1})+P_\Omega(\mathcal{Y}^k-\widetilde{\mathcal{Y}}^{k+1}),
\end{aligned}$$

we have

$$\frac{1}{2}\big(\big\|P_\Omega(\mathcal{X}^k-\widetilde{\mathcal{X}}^{k+1})\big\|_F^2+\big\|P_\Omega(\mathcal{Y}^k-\widetilde{\mathcal{Y}}^{k+1})\big\|_F^2\big)$$

$$
\begin{aligned}
&\geq \frac{1}{4}\left\|P_\Omega(\mathcal{X}^k+\mathcal{Y}^k-\widetilde{\mathcal{X}}^{k+1}-\widetilde{\mathcal{Y}}^{k+1})\right\|_F^2\\
&\geq \left(\left\|\frac{1}{2}\big(P_\Omega(\widetilde{\mathcal{X}}^{k+1}+\widetilde{\mathcal{Y}}^{k+1}-\mathcal{T})\big)\right\|_F-\left\|\frac{1}{2}\big(P_\Omega(\mathcal{X}^k+\mathcal{Y}^k-\mathcal{T})\big)\right\|_F\right)^2\\
&\geq \left(\frac{1}{\beta^{k+1}}\delta-\delta\right)^2=\frac{(1-\beta^{k+1})^2}{(\beta^{k+1})^2}\delta^2,
\end{aligned}
$$

which implies that the theorem holds. □

Remark 4.1. Obviously, when $P_\Omega(\mathcal{X})=P_\Omega(\mathcal{T})=P_\Omega(\mathcal{Y})$, then $\delta=0$, this is no other than tensor completion without noise.

Theorem 4.1. *Suppose that the sequence $\{\mathcal{X}^k,\mathcal{Y}^k\}$ is generated by Algorithm 2.1, the following statements hold:*

(i) The sequence $\{\mathcal{X}^k,\mathcal{Y}^k\}$ is bounded, $\lim_{\lambda^k\to 0}(1-\beta^k)=0$, $\lim_{\lambda^k\to 0}(\mathcal{X}^k-\mathcal{Y}^k)=0$.

(ii) There exists a sub-sequence $\{\mathcal{X}^{k_i},\mathcal{Y}^{k_i}\}$ converges to the stationary point of the optimization problem (1.4).

Proof. First, we prove (i). From Lemma 4.3, we get that the sequence $\{F(\mathcal{X}^k,\mathcal{Y}^k,\lambda^k)\}$ is monotonic non-increasing, so

$$
\sum_{k\geq 1}\frac{1}{2}\left(\left\|P_{\overline{\Omega}}(\mathcal{X}^k-\mathcal{X}^{k+1})\right\|_F^2\right)+\left\|P_{\overline{\Omega}}(\mathcal{Y}^k-\mathcal{Y}^{k+1})\right\|_F^2+\frac{\delta^2(1-\beta^{k+1})^2}{(\beta^{k+1})^2}<\infty.
$$

From Lemma 4.1 we get that the sequence $\{\mathcal{X}^k,\mathcal{Y}^k\}$ is bounded and $\lim_{\lambda^k\to 0}(1-\beta^k)=0$. Meanwhile, (4.1) and $\lim_{\lambda^k\to 0}(1-\beta^k)=0$ imply that $\lim_{\lambda^k\to 0}(\mathcal{X}^k-\mathcal{Y}^k)=0$.

Next, we prove (ii). From (i), there exists a sub-sequence $\{k_i\}$ such that $\{\mathcal{X}^{k_i},\mathcal{Y}^{k_i},\widetilde{\mathcal{X}}^{k_i},\widetilde{\mathcal{Y}}^{k_i}\}$ converges to $\mathcal{X}^*$. The following proves that the limit point $\mathcal{X}^*$ is a stationary point of the optimization problem (1.4).

Since $\mathcal{X}^k=\widetilde{\mathcal{X}}^k+(1-\beta^k)(P_\Omega(\mathcal{T}-\widetilde{\mathcal{X}}^k)$ and $\lim_{\lambda^k\to 0}(1-\beta^k)=0$, so $\lim_{\lambda^k\to 0}\widetilde{\mathcal{X}}^{k_i}=\lim_{\lambda^k\to 0}\mathcal{X}^{k_i}=\mathcal{X}^*$. Thus, from references [8,26,30], we have

$$
\lim_{\lambda^k\to 0}\partial\left\|\widetilde{\mathcal{X}}^{k_i}_{(i)}\right\|_*\subseteq\partial\left\|\mathcal{X}^*_{(i)}\right\|_*,\quad i=1,2,\ldots,N, \tag{4.5}
$$

which implies that

$$
\lim_{\lambda^{k_i}\to 0}\operatorname{fold}\big(\partial f_1(\mathcal{X}^{k_i})\big)+\operatorname{fold}\big(\partial f_2(\mathcal{X}^{k_i})\big)\subseteq\operatorname{fold}\big(\partial f_1(\mathcal{X}^*)\big)+\operatorname{fold}\big(\partial f_2(\mathcal{X}^*)\big).
$$

On the other hand, combining (4.1) and (4.2), we have

$$
\begin{aligned}
\operatorname{fold}\left(\frac{\partial f_1(\widetilde{\mathcal{X}}^{k_i})}{\partial\mathcal{X}}\right)&=\frac{2(\mathcal{Y}^{k_i-1}-\widetilde{\mathcal{X}}^{k_i})}{\lambda^{k_i-1}}=\frac{2(\mathcal{Y}^{k_i-1}-\mathcal{X}^{k_i}+\mathcal{X}^{k_i}-\widetilde{\mathcal{X}}^{k_i})}{\lambda^{k_i-1}}\\
&=\frac{2(\mathcal{Y}^{k_i-1}-\mathcal{X}^{k_i})}{\lambda^{k_i-1}}+\frac{2(\mathcal{X}^{k_i}-\widetilde{\mathcal{X}}^{k_i})}{\lambda^{k_i-1}}\\
&=\frac{2(\mathcal{Y}^{k_i-1}-\mathcal{X}^{k_i})}{\lambda^{k_i-1}}+\frac{2(1-\beta^{k_i})P_\Omega(\mathcal{T}-\widetilde{\mathcal{X}}^{k_i})}{\lambda^{k_i-1}},
\end{aligned}
$$

where $\beta^{k_i} < 1$, otherwise $\mathcal{X}^{k_i} = \widetilde{\mathcal{X}}^{k_i}$. Similarly, we have

$$\text{fold}\left(\frac{\partial f_2(\widetilde{\mathcal{Y}}^{k_i})}{\partial \mathcal{Y}}\right) = \frac{2(\mathcal{X}^{k_i} - \mathcal{Y}^{k_i})}{\lambda^{k_i - 1}} + \frac{2(1-\beta^{k_i})P_\Omega(2\mathcal{T} - \widetilde{\mathcal{X}}^{k_i} - \widetilde{\mathcal{Y}}^{k_i})}{\lambda^{k_i-1}}.$$

Since the $\{\mathcal{X}^{k_i}, \mathcal{Y}^{k_i}, \widetilde{\mathcal{X}}^{k_i}, \widetilde{\mathcal{Y}}^{k_i}\}$ is convergent, then

$$\lim_{\lambda^{k_i}\to 0} \frac{2(\mathcal{Y}^{k_i-1} - \mathcal{X}^{k_i})}{\lambda^{k_i-1}} = E, \quad \lim_{\lambda^{k_i}\to 0} \frac{2(\mathcal{X}^{k_i} - \mathcal{Y}^{k_i})}{\lambda^{k_i-1}} = -E.$$

In order to obtain $2(1-\beta^{k_i})/\lambda^{k_i-1}$, we compute

$$1 - \beta^{k_i} = \frac{\left\|\left(P_\Omega(\widetilde{\mathcal{X}}^{k_i}) + P_\Omega(\widetilde{\mathcal{Y}}^{k_i})\right)/2 - P_\Omega(\mathcal{T})\right\|_F - \delta}{\left\|\left(P_\Omega(\widetilde{\mathcal{X}}^{k_i}) + P_\Omega(\widetilde{\mathcal{Y}}^{k_i})\right)/2 - P_\Omega(\mathcal{T})\right\|_F}.$$

Further,

$$\begin{aligned}
&\left\|\frac{1}{2}P_\Omega(\widetilde{\mathcal{X}}^{k_i} + \widetilde{\mathcal{Y}}^{k_i} - 2\mathcal{T})\right\|_F \\
=& \left\|\frac{1}{2}P_\Omega(\mathcal{X}^{k_i-1} + \mathcal{Y}^{k_i-1} - 2\mathcal{T} + \widetilde{\mathcal{Y}}^{k_i} - \mathcal{Y}^{k_i-1} + \widetilde{\mathcal{X}}^{k_i} - \mathcal{Y}^{k_i-1} + \mathcal{Y}^{k_i-1} - \mathcal{X}^{k_i-1})\right\|_F, \\
&\left\|\frac{1}{2}P_\Omega(\mathcal{X}^{k_i-1} + \mathcal{Y}^{k_i-1}) - P_\Omega(\mathcal{T})\right\|_F = \delta.
\end{aligned}$$

Then the following inequality:

$$\begin{aligned}
&\left\|\frac{1}{2}P_\Omega(\widetilde{\mathcal{X}}^{k_i} + \widetilde{\mathcal{Y}}^{k_i} - 2\mathcal{T})\right\|_F - \delta \\
\le& \frac{1}{2}\left(\left\|P_\Omega(\widetilde{\mathcal{Y}}^{k_i} - \mathcal{Y}^{k_i-1})\right\|_F + \left\|P_\Omega(\widetilde{\mathcal{X}}^{k_i} - \mathcal{Y}^{k_i-1})\right\|_F + \left\|P_\Omega(\mathcal{Y}^{k_i-1} - \mathcal{X}^{k_i-1})\right\|_F\right) \\
\le& \frac{1}{2}\left(\|\mathcal{B}^{k_i} + \mathcal{C}^{k_i}\|_F + \|\mathcal{B}^{k_i}\|_F\right) + \frac{\beta^{k_i-1}}{2}\left\|P_\Omega(\widetilde{\mathcal{Y}}^{k_i-1} - \widetilde{\mathcal{X}}^{k_i-1})\right\|_F \\
\le& \frac{1}{2}\left(\|\mathcal{B}^{k_i} + \mathcal{C}^{k_i}\|_F + \|\mathcal{B}^{k_i}\|_F\right) + \frac{\beta^{k_i-1}}{2}\|\mathcal{C}^{k_i-1}\|_F
\end{aligned}$$

holds, where

$$\mathcal{B}^{k_i} = \widetilde{\mathcal{X}}^{k_i} - \mathcal{Y}^{k_i-1}, \quad \mathcal{C}^{k_i} = \widetilde{\mathcal{Y}}^{k_i} - \widetilde{\mathcal{X}}^{k_i}, \quad \mathcal{C}^{k_i-1} = \widetilde{\mathcal{Y}}^{k_i-1} - \widetilde{\mathcal{X}}^{k_i-1}.$$

From (4.1),

$$\mathcal{B}^{k_i} = -\frac{\lambda^{k_i-1}}{2}\mathcal{W}_1^{k_i}, \quad \mathcal{C}^{k_i} = -\frac{\lambda^{k_i-1}}{2}\mathcal{W}_2^{k_i}, \quad \mathcal{C}^{k_i-1} = -\frac{\lambda^{k_i-2}}{2}\mathcal{W}_2^{k_i-1}.$$

Since $\lim_{k\to\infty}\mathcal{W}_1^{k_i} \in \text{fold}(\partial f_1(\mathcal{X}^*))$ and $\lim_{k\to\infty}\mathcal{W}_2^{k_i} \in \text{fold}(\partial f_2(\mathcal{X}^*))$, so

$$\lim_{\lambda^{k_i}\to 0} \frac{2(1-\beta^{k_i})}{\lambda^{k_i-1}} = \mu \ge 0.$$

Then we have

$$\begin{aligned}
\lim_{\lambda^{k_i}\to 0} \text{fold}\left(\frac{\partial f_1(\widetilde{\mathcal{X}}^{k_i})}{\mathcal{X}}\right) &= \lim_{\lambda^{k_i}\to 0} \frac{2(\mathcal{Y}^{k_i-1} - \mathcal{X}^{k_i})}{\lambda^{k_i-1}} + \lim_{\lambda^{k_i}\to 0} \frac{2(1-\beta^{k_i})P_\Omega(\mathcal{T} - \widetilde{\mathcal{X}}^{k_i})}{\lambda^{k_i-1}} \\
&= E + \mu P_\Omega(\mathcal{T} - \mathcal{X}^*).
\end{aligned}$$

Using the same technique, we obtain that

$$\lim_{\lambda^{k_i}\to 0} \mathrm{fold}\left(\frac{\partial f_2(\widetilde{\mathcal{Y}}^{k_i})}{\mathcal{Y}}\right) = -E + \mu P_\Omega(2\mathcal{T} - \mathcal{X}^* - \mathcal{Y}^*).$$

From $\mathcal{X}^* = \mathcal{Y}^*$ and $\mu^* = 3\mu$, we obtain (3.2). Hence, (ii) is obtained. □

5. Numerical Results

In this section, we compare the proposed algorithm with existing algorithm such as ADM-TR, LRTV-pds, TRLRF, TRNNM (see [9, 32–34], based on random data and real image. All the experiments are performed under Windows 11 and MATLAB R2019a running on an DELL laptop.

5.1. Experiment settings

For simulation data experiments, randomly generated noise-laden tensors are used to demonstrate the optimization effect of the proposed min-max model and algorithm. The random tensor is generated by the Tucker decomposition of the tensor

$$\mathcal{T} = \mathcal{G} \times_1 U_1 \times_2 U_2 \times_3 \cdots \times_N U_N,$$

where $\mathcal{G} \in \mathbb{R}^{r_1\times r_2\times\cdots\times r_N}$ is the nuclear tensor, $U_n \in \mathbb{R}^{I_n\times r_n}, n = 1,2,\ldots,N$ is the matrix. $\delta \geq 0$ measures the noise level. Let the original tensor $\mathcal{Q} = \mathcal{T} + \delta/\|\mathcal{T}\|_F \cdot \mathcal{T}$ to show the noise. Furthermore, every element of $\mathcal{G}, U_i, \mathcal{Q}$ obey the standard Gaussian distribution. We use $sr = s/(I_1 \times I_2 \times \cdots \times I_N)$ to denote sampling rate. In noisy case, we use

$$RSE = \frac{\|(\mathcal{X}^{k+1} + \mathcal{Y}^{k+1})/2 - \mathcal{T}\|_F}{\|\mathcal{T}\|_F}$$

to measure the recovery accuracy of the algorithms.

In the real data experiment, we consider image recovery to illustrate the advantages of the algorithm proposed in this paper. Here, we use PSNR and CPU time to evaluate the numerical performance for the real image denoising.

We set

$$\lambda^0 = \sum_{i=1}^{N} \frac{\|P_\Omega(\mathcal{T})_{(i)}\|_F}{N}, \quad \rho = 0.92, \quad \delta = 0.1, \quad Maxiter = 1000.$$

For ADM-TR, $\beta = 1, \lambda = N, c_\beta = 1, c_\lambda = 1/0.92$.

5.1.1. Experiments on simulation data

Tables 5.1-5.3 show the running results of the simulated data with sampling rates of $sr = 0.4$, $0.3, 0.2$, respectively.

Note: In order to facilitate the calculation and save time, we control the CPU time of the TRNNM algorithm.

Table 5.1: Comparisons of the MIN-MAX algorithm and the other algorithms with $sr = 0.4$.

Size	Rank	Algorithm	Iter	CPU time(s)	RSE
$(100 \times 100 \times 100)$	(2,2,2)	MIN-MAX	175	3.8173	6.1434e-8
		ADM-TR	57	3.8381	2.0720e-3
		LRTC-pds	15	2.1035	3.2817e-3
		TRLRF	31	4.5110	5.0197e-3
		TRNNM	52	102.6003	1.4960e-6
$(150 \times 150 \times 150)$	(5,5,5)	MIN-MAX	195	12.9502	4.6713e-8
		ADM-TR	82	20.6033	4.1248e-4
		LRTC-pds	92	38.4765	8.7323e-4
		TRLRF	29	11.7051	8.3673e-4
		TRNNM	141	260.7580	1.7944e-6
$(100 \times 100 \times 100 \times 100)$	(2,2,2,2)	MIN-MAX	183	192.4233	3.9907e-8
		ADM-TR	66	220.4198	7.0896e-4
		TRLRF	43	185.9719	3.1624e-5
		TRNNM	15	6881.9	1.5823e-2
$(100 \times 100 \times 100 \times 100)$	(2,3,4,5)	MIN-MAX	247	381.7125	4.2138e-8
		ADM-TR	114	397.8545	2.5735e-4
		TRLRF	27	124.7607	2.5529e-6
		TRNNM	25	9090.1	3.2707e-3
$(50 \times 50 \times 50 \times 50 \times 50)$	(2,2,2,2,2)	MIN-MAX	225	974.2706	4.3183e-8
		ADM-TR	97	1066.7939	9.4595e-4
		TRLRF	33	540.9885	1.7920e-4
		TRNNM	40	9.8690e3	1.3828e-2
$(50 \times 60 \times 70 \times 70 \times 70)$	(2,2,2,2,2)	MIN-MAX	235	3304.6540	3.5387e-8
		ADM-TR	100	4502.6834	3.9702e-4
		TRLRF	27	1863.6	2.7407e-6
		TRNNM	40	14569	7.5519e-2

5.1.2. Experiments on real data

In this part, we use a color image “flower”[1] of size $504 \times 756 \times 3$ and a color image “grass”[2] of size $504 \times 756 \times 3$ to evaluate the numerical performance of the comparison algorithm. Tables 5.4 and 5.5 show the effect of CPU time and PSNR.

$$PSNR = 10\log_{10}\frac{n{\mathcal{T}_{\max}}^2}{\|X - \mathcal{T}\|_F^2},$$

where n is the number of pixels in the tensor, and $\mathcal{T}_{\max}$ is the maximum pixel of the original tensor (see [29]). Fig. 5.1 shows the effect of image recovery with sample rates $sr = 0.4, 0.3, 0.2$.

[1] https://www2.eecs.berkeley.edu/
[2] https://www2.eecs.berkeley.edu/

Table 5.2: Comparisons of the MIN-MAX algorithm and the other algorithms with $sr = 0.3$.

Size	Rank	Algorithm	Iter	CPU time(s)	RSE
$(100 \times 100 \times 100)$	(2,2,2)	MIN-MAX	176	5.4038	8.3112e-8
		ADM-TR	78	6.0648	2.0720e-3
		LRTC-pds	24	3.1705	6.0288e-3
		TRLRF	44	4.5167	1.7428e-3
		TRNNM	71	106.6363	1.8337e-6
$(150 \times 150 \times 150)$	(5,5,5)	MIN-MAX	195	12.8491	6.8620e-8
		ADM-TR	108	27.2486	4.1328e-4
		LRTC-pds	99	40.0217	9.4552e-4
		TRLRF	43	16.5785	9.7536e-4
		TRNNM	186	338.3928	1.3628e-6
$(100 \times 100 \times 100 \times 100)$	(2,2,2,2)	MIN-MAX	183	298.3046	5.7011e-8
		ADM-TR	92	308.4272	7.0935e-4
		TRLRF	57	231.2765	3.0806e-5
		TRNNM	20	18362	1.7035e-2
$(100 \times 100 \times 100 \times 100)$	(2,3,4,5)	MIN-MAX	247	382.0222	6.2007e-8
		ADM-TR	155	510.3100	2.5593e-4
		TRLRF	48	211.4977	1.9263e-4
		TRNNM	25	1460.5	5.3942e-1
$(50 \times 50 \times 50 \times 50 \times 50)$	(2,2,2,2,2)	MIN-MAX	225	988.1588	6.0795e-8
		ADM-TR	101	1106.6612	9.4403e-4
		TRLRF	37	588.9137	4.2845e-6
		TRNNM	40	10833	2.1253e-1
$(50 \times 60 \times 70 \times 70 \times 70)$	(2,2,2,2,2)	MIN-MAX	235	3244.3689	5.1351e-8
		ADM-TR	138	6070.7847	3.9284e-4
		TRLRF	39	2968.1	2.3966e-4
		TRNNM	40	24766	2.7412e-1

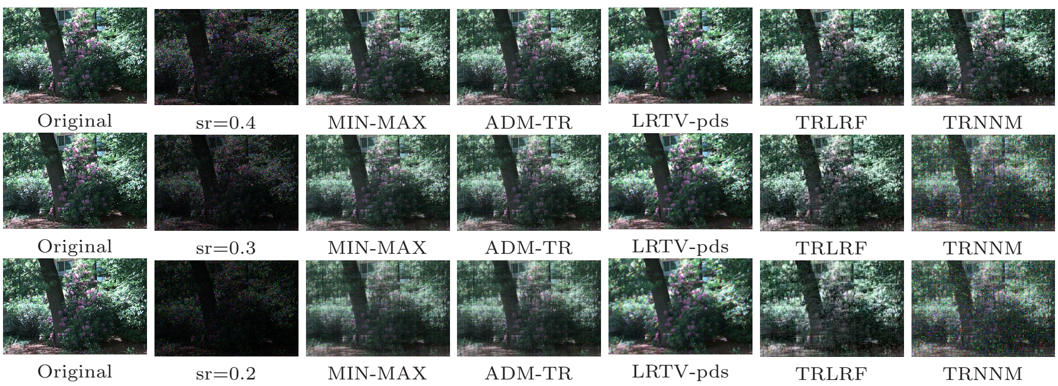

Fig. 5.1. The recovery results for the color image “flower”.

Table 5.3: Comparisons of the MIN-MAX algorithm and the other algorithms with $sr = 0.2$.

Size	Rank	Algorithm	Iter	CPU time(s)	RSE
$(100 \times 100 \times 100)$	(2,2,2)	MIN-MAX	177	4.2141	1.3554e-7
		ADM-TR	114	7.7268	2.0719e-6
		LRTC-pds	40	4.9073	1.5125e-3
		TRLRF	66	5.6869	3.6515e-2
		TRNNM	106	159.6403	2.9299e-6
$(150 \times 150 \times 150)$	(5,5,5)	MIN-MAX	197	13.5185	1.2997e-7
		ADM-TR	174	17.3654	2.7497e-6
		LRTC-pds	114	46.1975	1.0261e-3
		TRLRF	72	27.5751	1.0232e-3
		TRNNM	264	415.8293	1.9632e-6
$(100 \times 100 \times 100 \times 100)$	(2,2,2,2)	MIN-MAX	185	292.0888	9.9171e-8
		ADM-TR	135	454.7938	7.0802e-4
		TRLRF	88	340.1408	3.2481e-4
		TRNNM	20	17534	4.9759e-2
$(100 \times 100 \times 100 \times 100)$	(2,3,4,5)	MIN-MAX	248	378.3138	1.0839e-7
		ADM-TR	229	748.7921	2.5682e-6
		TRLRF	58	245.0460	1.0141e-5
		TRNNM	25	10162	6.3912e-1
$(50 \times 50 \times 50 \times 50 \times 50)$	(2,2,2,2,2)	MIN-MAX	231	1010.9566	7.1996e-7
		ADM-TR	151	1664.7642	9.3242e-4
		TRLRF	61	905.9137	2.0081e-5
		TRNNM	40	15246	3.0292e-1
$(50 \times 60 \times 70 \times 70 \times 70)$	(2,2,2,2,2)	MIN-MAX	236	3259.0377	8.8906e-8
		ADM-TR	206	9130.5879	3.9391e-4
		TRLRF	103	6080.6	3.5085e-4
		TRNNM	40	15083	4.1116e-1

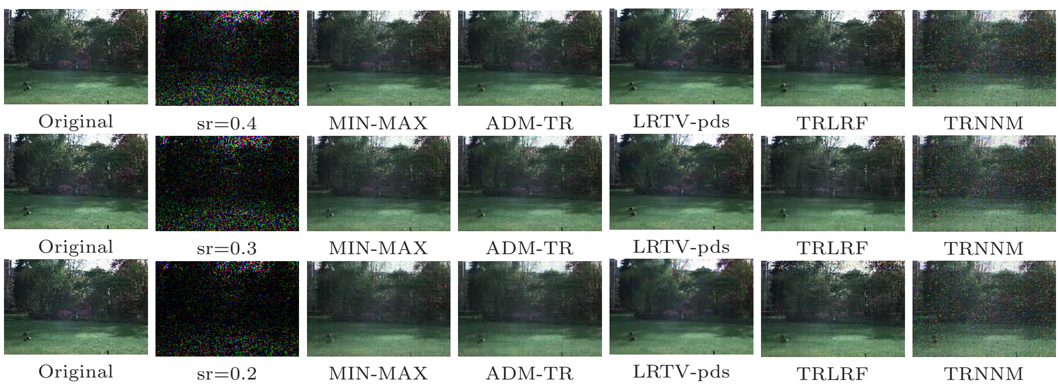

Fig. 5.2. The recovery results for the color image “grass”.

Table 5.4: Comparisons of the algorithms for the color image "flower" at different sampling rates.

Algorithm	Samples	Iter	CPU time(s)	PSNR
MIN-MAX	sr=0.4	254	56.5078	21.3938
ADM-TR		264	73.4049	21.8913
LRTV-pds		321	143.1165	21.6267
TRLRF		185	57.4786	18.8096
TRNNM		76	37.7743	18.5699
MIN-MAX	sr=0.3	252	57.1078	19.0655
ADM-TR		336	93.3372	19.6524
LRTV-pds		391	169.7973	19.9536
TRLRF		382	168.3693	17.9738
TRNNM		92	36.7922	17.2471
MIN-MAX	sr=0.2	249	53.1774	16.7094
ADM-TR		458	129.3386	17.3738
LRTV-pds		507	221.0463	18.1644
TRLRF		423	100.5691	17.0470
TRNNM		115	35.8688	15.8995

Table 5.5: Comparisons of the algorithms for the color image "grass" at different sampling rates.

Algorithm	Samples	Iter	CPU time(s)	PSNR
MIN-MAX	sr=0.4	211	45.7940	28.4158
ADM-TR		189	52.9261	28.6328
LRTV-pds		273	122.7735	27.9377
TRLRF		198	60.9588	26.4692
TRNNM		75	34.3256	23.1929
MIN-MAX	sr=0.3	210	44.9488	26.0192
ADM-TR		248	65.5477	26.4237
LRTV-pds		340	152.5799	26.2648
TRLRF		283	100.5541	25.5099
TRNNM		89	34.3611	22.1953
MIN-MAX	sr=0.2	208	43.3650	23.3589
ADM-TR		343	91.5023	24.1950
LRTV-pds		462	203.3269	24.7157
TRLRF		313	161.8627	24.0510
TRNNM		108	31.0088	18.9862

6. Concluding Remarks

From the above numerical experiments, the proposed algorithm for solving the proposed optimization with Gaussian noise (1.5) is better than the compared algorithms in CPU time or precision. The new algorithm has better accuracy than the compared methods on the random experiment. Although PSNR of the new algorithm is slightly less than ADM-TR and LRTV-pds methods, the new algorithm takes less CPU time than them in real image inpaining. This is mainly because the real image inpaining is not a low-rank tensor.

Acknowledgments. This work is supported by the National Nature Science Foundation (Grant No. 12371381) and by the Special Fund for Science and Technology Innovation Team of Shanxi Province (Grant No. 202204051002018).

References

[1] M. Bertalmio, A.L. Bertozzi, and G. Sapir, Navier-Stokes, fluid dynamics, and image and video inpainting, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, **1** (2001), 355–362.

[2] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, Image inpainting, in: *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, ACM Press, (2000), 417–424.

[3] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[4] J.F. Cai, E.J. Candès, and Z.W. Shen, A singular value thresholding algorithm for matrix completion, *SIAM J. Optim.*, **20** (2010), 1956–1982.

[5] E.J. Candès and Y. Plan, Matrix completion with noise, *Proc. IEEE*, **98** (2010), 925–936.

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

[7] L. Chen, H. Li, and S. Xie, Modified high-order SVD for spatiotemporal modeling of distributed parameter systems, *IEEE Trans. Ind. Electron.*, **69** (2022), 4296–4304.

[8] F.H. Clarke, *Optimization and Nonsmooth Analysis*, SIAM, 1983.

[9] S. Gandy, B. Recht, and I. Yamada, Tensor completion and low-n-rank tensor recovery via convex optimization, *Inverse Problems*, **27** (2011), 025010.

[10] G.H. Golub and C.F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, 2013.

[11] L. Grasedyck, M. Kluge, and S. Kramer, Variants of alternating least squares tensor completion in the tensor train format, *SIAM J. Sci. Comput.*, **37** (2015), A2424–A2450.

[12] L. Grasedyck, D. Kressner, and C. Tobler, A literature survey of low-rank tensor approximation techniques, *GAMM-Mitt.*, **36** (2013), 53–78.

[13] Y. Kapushev, I. Oseledets, and E. Burnaev, Tensor completion via Gaussian process-based initialization, *SIAM J. Sci. Comput.*, **42** (2020), A3812–A3824.

[14] R. Keshavan A. Montanari, and S. Oh, Matrix completion from noisy entries, *J. Mach. Learn. Res.*, **11** (2010), 2057–2078.

[15] T.G. Kolda and B.W. Bader, Tensor decompositions and applications, *SIAM Rev.*, **51** (2009), 455–500.

[16] N. Komodakis, Image completion using global optimization, in: *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, (2006), 442–452.

[17] D. Kressner, M. Steinlechner, and B. Vandereycken, Low-rank tensor completion by Riemannian optimization, *BIT*, **54** (2014), 447–468.

[18] X. Li, H. Shen, H. Li, and L. Zhang, Patch matching-based multitemporal group sparse representation for the missing information reconstruction of remote-sensing images, *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, **9** (2016), 3629–3641.

[19] J. Liu, P. Musialski, P. Wonka, and J. Ye, Tensor completion for estimating missing values in visual data, *IEEE Trans. Pattern Anal. Mach. Intell.*, **35** (2013), 208–220.

[20] T. Liu, T.K. Pong, and A. Takeda, A refined convergence analysis of pDCA$_e$ with applications to simultaneous sparse recovery and outlier detection, *Comput. Optim. Appl.*, **73** (2019), 69–100.

[21] S. Mohaoui, A. Hakim, and S. Raghay, Tensor completion via bilevel minimization with fixed-point constraint to estimate missing elements in noisy data, *Adv. Comput. Math.*, **47** (2021), 10.

[22] C. Mu, B. Huang, J. Wright, and D. Goldfarb, Square deal: Lower bounds and improved relaxations for tensor recovery, in: *Proceedings of the 31st International Conference on International Conference on Machine Learning*, PMLR, **32** (2014), 73–81.

[23] M.K.P. Ng, Q. Yuan, L. Yan, and J. Sun, An adaptive weighted tensor completion method for the recovery of remote sensing images with missing data, *IEEE Trans. Geosci. Remote Sens.*, **55** (2017), 3367–3381.

[24] M. Pauly, N.J. Mitra, J. Giesen, L. Guibas, and M. Gross, Example-based 3D scan completion, in: *Proceedings of the Third Eurographics Symposium on Geometry Processing*, Eurographics Association, (2005), 23–32.

[25] Y. Qiu, G. Zhou, Q. Zhao, and S. Xie, Noisy tensor completion via low-rank tensor ring, *IEEE Trans. Neural Networks Learn. Syst.*, **35** (2024), 1127–1141.

[26] R.T. Rockafellar, *Convex Analysis*, Princeton University Press, 1972.

[27] R. Schmidt, Multiple emitter location and signal parameter estimation, *IEEE Trans. Antennas Propag.*, **34** (1986), 276–280.

[28] L.R. Tucker, R.F. Koopman, and R.L. Linn, Evaluation of factor analytic research procedures by means of simulated correlation matrices, *Psychometrika*, **34** (1969), 421–459.

[29] Z. Wang and A.C. Bovik, Mean squared error: Love it or leave it? A new look at signal fidelity measures, *IEEE Signal Process Mag.*, **26** (2009), 98–117.

[30] G.A. Watson, Characterization of the subdifferential of some matrix norms – ScienceDirect, *Linear Algebra Appl.*, **170** (1992), 33–45.

[31] S.F. Xu and J. Qian, *Matrix Computation. Six Lectures*, Higher Education Press, 2011.

[32] T. Yokota and H. Hontani, Simultaneous tensor completion and denoising by noise inequality constrained convex optimization, *IEEE Access*, **7** (2019), 15669–15682.

[33] J. Yu, C. Li, Q. Zhao, and G. Zhao, Tensor-ring nuclear norm minimization and application for visual: Data Completion, in: *IEEE International Conference on Acoustics, Speech and Signal Processing – Proceedings*, IEEE, (2019), 3142–3146.

[34] L. Yuan, C. Li, J. Cao, and Q. Zhao, Rank minimization on tensor ring: An efficient approach for tensor decomposition and completion, *Mach. Learn.*, **109** (2020), 603–622.

[35] Q. Zhao, G. Zhou, S. Xie, L. Zhang, and A. Cichocki, Tensor ring decomposition, *arXiv: 1606.05535*, 2016.