SIAM J. Imaging Sciences
Vol. 17, No. 4, pp. 2242–2276

A Low-Rank Tensor Completion Method via Strassen–Ottaviani Flattening*

Tiantian He†, Shenglong Hu‡, and Zheng-Hai Huang§

Abstract. In this paper, a tensor completion method is proposed based on the Strassen–Ottaviani flattening, which can reveal the underlying tensor rank intrinsically. The resulting tensor completion optimization problem is formulated by spectral functions (convex or nonconvex) as surrogates for the rank function. An exact recovery result for the nuclear norm surrogate is given. An efficient method is proposed for this problem, and adaptive parameters for the spectral functions are allowed during the iterations. The global convergence is established under mild assumptions on the spectral functions and the adaptive parameter updates. In particular, we show that the class of weighted nuclear norms (either with nonincreasing or nondecreasing weights), the ε-sparsity index, and a class of weighted nuclear norms with adaptive weights, which are all widely employed in the literature, all fulfill the assumptions, and thus the global convergence is valid without any assumption. Numerical experiments on color images show that the proposed methods are promising, and always return images with better quality than some state-of-the-art methods.

Key words. tensor completion, Strassen–Ottaviani flattening, global convergence, adaptive parameters

MSC codes. 15A69, 90C25, 90C26

DOI. 10.1137/23M158975X

1. Introduction.

Tensors, as higher order equivalents of vectors and matrices, have been becoming one of the most powerful and prevalent tools for representing multidimensional data in diverse fields, especially in image sciences. Actually, a color image is naturally represented as a third order tensor. Therefore, computations on third as well as higher order tensors have great impact on processing for color images, videos, etc. Our primary focus of this paper is in this category, and we actually consider a more practical situation. There are several scenarios that parts of the data tensor are missing, possibly due to physical damage or acquisition strategies, etc. Immediately, an important postprocessing task is to recover the missing entries from the given incomplete tensor. This is the well-known tensor completion (TC) problem, which has been finding numerous applications in a broad range fields, such as image and video inpainting, MRI data recovery, etc. This inverse problem cannot be solved unless some structural hypotheses are put on the underlying tensor. A common and basic assumption is that the ground truth tensor has a low-rank structure, modeling the geometric

*Received by the editors July 27, 2023; accepted for publication (in revised form) July 30, 2024; published electronically December 2, 2024.
https://doi.org/10.1137/23M158975X
Funding: This work is partially supported by the National Science Foundation of China (grant 12171128 and grant 12371309) and the Natural Science Foundation of Zhejiang Province, China (grant LY22A010022).
†Department of Mathematics, School of Science, Hangzhou Dianzi University, Hangzhou, 310018 Zhejiang, China (hetiantian@hdu.edu.cn).
‡Corresponding author. Department of Mathematics, College of Science, National University of Defense Technology, Changsha, 410072 Hunan, China (hushenglong@nudt.edu.cn).
§School of Mathematics and KL-AAGDM, Tianjin University, Tianjin, 300350 China (huangzhenghai@tju.edu.cn).

sparsity. Thus, the low-rank TC (LRTC) problem is employed in this task, and it has been becoming a research frontier in the last decade. Methods for LRTC can be roughly categorized into two groups: rank minimization based on various matrix flattenings or representation techniques of tensors (see [5, 6, 16, 17, 26, 31, 32, 34, 36, 42, 52, 53, 54, 55] and references therein), and low-rank tensor approximation based on several tensor rank decompositions (see [1, 3, 6, 12, 19, 29, 47, 51, 56, 57] and references therein).

In this paper, we will follow the former approach on rank minimization. Unlike matrices, there are several notions of ranks for tensors. This paper considers canonical polyadic (CP) rank, and it is well known that its calculation is in general NP-hard [23]. Even in some models, such as Tucker, where the rank can be computed out, optimization with rank functions is still a hard problem. Thus, various surrogates for the rank functions are proposed in the literature. On the other hand, most of them are based on certain matrix flattenings of the tensor. Different flattenings give different rank functions and their corresponding performances vary accordingly. In the literature, mode-n matricization or flattening is widely used, which is based on Tucker representation [45]. The others include tensor train (TT) rank [37] which utilizes a line network, tensor ring representation using more complicated networks [54], Tubal rank [27] which applies a circular strategy, square deal [36] which flattens the tensor into a matrix as square as possible, etc. One ingredient of our proposed method is the Strassen–Ottaviani flattening of tensors. It is a matrix flattening different from the ones employed in LRTC literature. Actually, the Strassen–Ottaviani flattening has a deep geometric description, which was first investigated by Strassen [44] and then by Ottaviani [38]; we refer to [28] and references herein for more details. It has an intrinsic relation to the CP rank of the underlying tensor, especially for tensors with higher ranks (cf. section 3).

In this paper, we propose a new TC method based on the Strassen–Ottaviani flattening for the underlying tensor. As this flattening can intrinsically reveal the tensor rank, it is expected that this TC method can be valuable in certain scenarios. As in the literature, we use surrogates for the rank function. While convex surrogates are widely applied, nonconvex surrogates are also valuable in some situations; see [17, 25, 32, 52] and references therein. Thus, we employ both convex and nonconvex spectral functions as surrogates. Moreover, we utilize an adaptive scheme for the surrogates and the discrete cosine transformation (DCT) for the data. Actually, favorable performance for adaptive weighted nuclear norms is reported; see [20, 21] and references therein. Meanwhile, DCT has been widely used in image processing, such as the popular image compression standard JPEG. The underlying rationale is that DCT can usually obtain (or at least approximately) a sparse representation of the image, since usually a nature image has only a few dominant coefficients in a Fourier basis [2]. It also provides an efficient computational method. These techniques are integrated into a unified model, and the resulting optimization problem is solved by an inexact augmented Lagrangian method, and the subproblem is solved by a proximal alternating minimization method. The global convergence is established under hypotheses merely on the spectral functions and the adaptive parameter updates, which are shown to be satisfied for several widely used classes. More precisely, the theoretical contributions are the following:

- A new TC method TCSO based on the Strassen–Ottaviani flattening is proposed. An exact recovery result for a tensor with prescribed CP rank is established under mild assumptions for the first time.

- An optimization problem based on spectral functions with adaptive parameters for the rank function and DCT is proposed to solve the TCSO. An efficient algorithm is proposed for this optimization problem, and global convergence to a Karush-Kuhn-Tucker (KKT) point is established under mild assumptions. This is the first convergence guarantee to a KKT point within adaptive schemes for TC problems.
- For the class of weighted nuclear norms (either nonincreasing weights or nondecreasing weights), the ε-sparsity index, and a class of weighted nuclear norms with adaptive weights, we show that the assumptions guaranteeing global convergence are valid. Then the algorithm converges globally without any assumption.

The remainder of this paper is organized as follows. In section 2, some basic notions and necessary properties on spectral functions and optimization are stated. The Strassen–Ottaviani flattening and the projection onto a polyhedron associated with this flattening are given in section 3. In section 4, we propose the TCSO model based on the Strassen–Ottaviani flattening and develop an inexact augmented Lagrangian method (ALM) to solve the associated optimization problem. The global convergence is established under mild assumptions on the surrogates for the rank function and the adaptive parameter updates. Details on subproblem solving and concrete surrogates will be addressed in sections 5 and 6. The former deals with nonconvex surrogates, including weighted nuclear norms with nondecreasing weights, the ε-sparsity index, and a class of weighted nuclear norms with adaptive weights, and the latter deals with convex surrogates. For convex problems, we employ a proximal alternating direction method of multipliers (proximal ADMM), which is simpler than the inexact ALM and its global convergence is given without any assumption. Numerical experiments to illustrate the theoretical results established in previous sections are given in section 7, compared with some state-of-the-art methods in the literature. Some final conclusions are given in the last section.

2. Preliminaries.

In this section, we will present some basic notations and definitions, as well as some optimization techniques.

2.1. Notations.

In this paper, scalars are denoted by lowercase letters, e.g., λ, vectors by bold lowercase letters, e.g., $\mathbf{a}$, matrices by uppercase letters, e.g., A, and higher order (three or higher) tensors by boldface Euler script letters, e.g., $\mathcal{X}$. The ith entry of a vector $\mathbf{a}$ is denoted as a_i, the (i,j)th element of a matrix A is denoted as a_{ij}, and the $(i_1,\ldots,i_N)$th element of an Nth order tensor $\mathcal{X}$ is denoted as $x_{i_1\cdots i_N}$.

Let $\mathbb{R}^{m\times n}$ be the linear space of all $m\times n$ real matrices equipped with the standard trace inner product $\langle X,Y\rangle := \mathrm{trace}(X^\intercal Y)$, where $X^\intercal$ is the transpose of X and $\mathrm{trace}(\cdot)$ is the trace of the matrix. For any $X\in\mathbb{R}^{m\times n}$ with $m\le n$, let $\sigma(X)\in\mathbb{R}^m$ be the vector of singular values $\sigma_i(X)$'s of X arranged in nonincreasing order. The Frobenius norm and the spectral norm of X are denoted as $\|X\|_F$ (which is the default norm used in this paper, and is often denoted as $\|\cdot\|$ for simplicity) and $\|X\|_2$, respectively. $\|X\|_\infty$ denotes the infinity norm, i.e., $\|X\|_\infty := \max_{i,j}|x_{ij}|$.

2.2. Spectral functions.

In the literature, a *spectral function* is an extended real-valued function defined on $\mathbb{R}^{m\times n}$ of the form $f\circ\sigma$ for an absolutely symmetric function $f:\mathbb{R}^m\to\mathbb{R}\cup\{\infty\}$. For more details, we refer the reader to [30] and references therein. Actually, any

function $g:\mathbb{R}^{m\times n}\to\mathbb{R}\cup\{\infty\}$ depending only on the singular values of the underlying matrix over $\mathbb{R}^{m\times n}$ (also known as unitarily invariant) is a spectral function.

In this paper, we will employ spectral functions to approximate the rank function of a matrix, since the rank of a matrix is the cardinality of its nonzero singular values. A commonly used spectral function along this line is the nuclear norm $\|X\|_*$, which is the sum of all singular values of X. This belongs to the class of convex spectral functions. There are also nonconvex spectral functions.

Definition 2.1 (weighted nuclear norm). *Let $m\le n$ be given positive integers, and $\mathbf{w}:=(w_1,\dots,w_m)^\top$ be a nonnegative weight vector. The weighted nuclear norm of a matrix $X\in\mathbb{R}^{m\times n}$ is defined as*

$$\|X\|_{\mathbf{w},*}:=\sum_{i=1}^{m} w_i\sigma_i(X). \tag{2.1}$$

Of particular interests are the weighted nuclear norms with nondecreasing weights; see [10]. This class of weighted nuclear norms is nonconvex when the weights are not the same constant, but explicit proximal mapping formulas exist (cf. section 5.1). If the weights are nonincreasing, then the corresponding weighted nuclear norms are convex, including the classical nuclear norm and the Ky-Fan norms.

Weighted nuclear norms take in the consideration of different magnitudes between the singular values, while they have fixed weights and thus the flexibility is limited. To remedy this, an adaptive weights strategy was proposed in [20]; moreover, we may also consider ε-sparsity index [35] defined as follows.

Definition 2.2 (ε-sparsity index). *Let $m\le n$ be given positive integers, and ε be a positive scalar. For a matrix $X\in\mathbb{R}^{m\times n}$, the ε-sparsity index of X is defined as*

$$\|X\|_{*,\varepsilon}:=\sum_{i=1}^{m}\frac{\sigma_i(X)}{\sigma_i(X)+\varepsilon}. \tag{2.2}$$

We see from (2.2) that the singular values play more balanced roles in the ε-sparsity index, which is helpful for inferring the rank of the underlying matrix.

It is easy to see that all the functions mentioned above are spectral functions, and they provide surrogates for the rank function. An intrinsic geometric interpretation for the nuclear norm is established in [13].

2.3. Optimization techniques. In order to study the optimality of the involved optimization problems, we recall the notions of subdifferentials of a function [41].

Definition 2.3. *Given a function $f:\mathbb{R}^n\to\mathbb{R}\cup\{\infty\}$, the regular subdifferential of f at $\mathbf{x}\in\mathbb{R}^n$ is defined as*

$$\hat\partial f(\mathbf{x}):=\left\{\mathbf{h}\in\mathbb{R}^n:\liminf_{\mathbf{x}\ne\mathbf{y}\to\mathbf{x}}\frac{f(\mathbf{y})-f(\mathbf{x})-\langle\mathbf{h},\mathbf{y}-\mathbf{x}\rangle}{\|\mathbf{y}-\mathbf{x}\|}\ge 0\right\}$$

and the (limiting) subdifferential of f at $\mathbf{x}$ is defined as

$$\partial f(\mathbf{x}):=\left\{\mathbf{h}\in\mathbb{R}^n:\exists\{\mathbf{x}^{(k)}\}\to\mathbf{x}\text{ and }\{\mathbf{h}^{(k)}\}\to\mathbf{h}\text{ satisfying }\mathbf{h}^{(k)}\in\hat\partial f(\mathbf{x}^{(k)})\text{ for all }k\right\}.$$

It is known that the (limiting) subdifferential reduces to the classical subdifferential in convex analysis when f is convex, i.e.,

$$\partial f(\mathbf{x}) = \{\mathbf{h} \in \mathbb{R}^n : f(\mathbf{y}) - f(\mathbf{x}) - \langle \mathbf{h}, \mathbf{y} - \mathbf{x}\rangle \geq 0 \text{ for all } \mathbf{y} \in \mathbb{R}^n\}.$$

We refer to [41] and references therein for more details on variational analysis concepts. An important fact about the subdifferential of the indicator function δ_{S} of a closed convex set S at $\mathbf{x}$ is $\partial\delta_{\mathsf{S}}(\mathbf{x}) = N_{\mathsf{S}}(\mathbf{x})$, where $N_{\mathsf{S}}(\mathbf{x})$ is the normal cone of S at $\mathbf{x}$ (cf. [41]).

A necessary (but not sufficient) condition for $\mathbf{x} \in \mathbb{R}^n$ to be a minimizer of f over $\mathbb{R}^n$ is

$$0 \in \partial f(\mathbf{x}). \tag{2.3}$$

A point that satisfies (2.3) is called a critical point. For constrained optimization problems, we always consider KKT points. The following proposition [41, Exercise 10.52] gives a relationship between local optimal solutions and KKT points.

Proposition 2.4 (nonsmooth Lagrange multiplier rule). *For a nonempty, closed set $\mathsf{S} \subseteq \mathbb{R}^n$ and strictly continuous functions $f_0 : \mathbb{R}^n \to \mathbb{R}$ and $F : \mathbb{R}^n \to \mathbb{R}^m$ with $F := (f_1, \ldots, f_m)$, consider the problem*

$$\min \ f_0(\mathbf{x}) + \theta(F(\mathbf{x})) \ \textit{s.t.} \ \mathbf{x} \in \mathsf{S}, \tag{2.4}$$

where $\theta : \mathbb{R}^m \to \mathbb{R} \cup \{\infty\}$ is proper, lower semicontinuous, and convex with effective domain D. (As a special case, one could have $\theta = \delta_{\mathsf{D}}$). Suppose $\bar{\mathbf{x}}$ is a local optimal solution at which the following constraint qualification is satisfied:

$$0 \in \partial(\mathbf{y}^{\mathsf{T}}F)(\bar{\mathbf{x}}) + N_{\mathsf{S}}(\bar{\mathbf{x}}) \ \textit{and} \ \mathbf{y} \in N_{\mathsf{D}}(F(\bar{\mathbf{x}})) \Longrightarrow \mathbf{y} = \mathbf{0}.$$

Then there exists a vector $\bar{\mathbf{y}}$ such that

$$0 \in \partial(f_0 + \bar{\mathbf{y}}^{\mathsf{T}}F)(\bar{\mathbf{x}}) + N_{\mathsf{S}}(\bar{\mathbf{x}}) \ \textit{and} \ \bar{\mathbf{y}} \in \partial\theta(F(\bar{\mathbf{x}})). \tag{2.5}$$

A vector $\bar{\mathbf{y}}$ satisfying (2.5) is called a Lagrange multiplier, and the pair $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ satisfying (2.5) is a KKT pair with $\bar{\mathbf{x}}$ a KKT point.

3. Strassen–Ottaviani flattening.

3.1. Matrix representation.

The Strassen–Ottaviani flattening is an intrinsic method for determining the rank of a given tensor. We will scratch the basic ideas for the Strassen–Ottaviani flattening here. Suppose that, after suitable embedding, the tensor is presented as a third order tensor $\mathcal{T}$ in the tensor space $\mathbb{A} \otimes \mathbb{B} \otimes \mathbb{C}$ for three vector spaces $\mathbb{A}$, $\mathbb{B}$, and $\mathbb{C}$, where $\otimes$ is the tensor product between vector spaces [28]. We know that the rank of $\mathcal{T}$, denoted as $\operatorname{rank}(\mathcal{T})$, is equal to the smallest number of rank one elements of $\mathbb{A} \otimes \mathbb{C}$ needed to span a linear subspace containing $\mathcal{T}(\mathbb{B}^*) \subseteq \mathbb{A} \otimes \mathbb{C}$, where $\mathbb{B}^*$ is the dual vector space of $\mathbb{B}$. The classical mode matrix flattening is representing the linear mapping $\mathcal{T} : \mathbb{B}^* \to \mathbb{A} \otimes \mathbb{C}$ via a matrix. Actually, we can go further and consider the mapping

$$\mathcal{T} \otimes \mathrm{id}_{\mathbb{A}} : \mathbb{A} \otimes \mathbb{B}^* \to \mathbb{A} \otimes \mathbb{A} \otimes \mathbb{C}.$$

This mapping is linear with respect to the tensor $\mathcal{T}$. Moreover, we can do this for either one of $\mathbb{A}$, $\mathbb{B}$, or $\mathbb{C}$. In order to make it clear, we always write it with a subscript $\mathbb{A}$ for the above operator. Let $\mathrm{GL}(\mathbb{A})$ be the general linear group over the vector space $\mathbb{A}$. Then, the tensor rank is $\mathrm{GL}(\mathbb{A}) \times \mathrm{GL}(\mathbb{B}) \times \mathrm{GL}(\mathbb{C})$ invariant, and we can then consider the irreducible module $\wedge^2\mathbb{A}\otimes\mathbb{C}$, i.e., the projection of the operator $\mathcal{T}\otimes \mathrm{id}_{\mathbb{A}}$ onto the submodule $\wedge^2\mathbb{A}\otimes\mathbb{C} \subset \mathbb{A}\otimes\mathbb{A}\otimes\mathbb{C}$. We will denote this linear operator as $\mathcal{T}_{\mathbb{A}}^{\wedge}$. Surprisingly, there are pleasant properties on the operator $\mathcal{T}_{\mathbb{A}}^{\wedge}$ and they are related to the rank of the tensor $\mathcal{T}$.

To be more explicit, let the tensor $\mathcal{T} \in \mathbb{A}\otimes\mathbb{B}\otimes\mathbb{C}$ be given, where $\mathbb{A}$, $\mathbb{B}$, and $\mathbb{C}$ are vector spaces of dimensions a, b, and c, respectively. Consider $\mathcal{T}\otimes \mathrm{id}_{\mathbb{A}} : \mathbb{A}\otimes\mathbb{B}^* \to \mathbb{A}\otimes\mathbb{A}\otimes\mathbb{C}$ and its projection $\mathcal{T}_{\mathbb{A}}^{\wedge} : \mathbb{A}\otimes\mathbb{B}^* \to \wedge^2\mathbb{A}\otimes\mathbb{C}$. Assume $a = 3$, and let $\mathbb{A}$ have basis $\{\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3\}$. Then we know that both $\mathbb{A}$ and $\wedge^2\mathbb{A}$ have dimension 3. On the other hand, we can write $\mathcal{T}$ as $\mathbf{a}_1 \otimes X_1 + \mathbf{a}_2 \otimes X_2 + \mathbf{a}_3 \otimes X_3$, where $X_j : \mathbb{B}^* \to \mathbb{C}$ is a $b \times c$ matrix for any $j \in \{1, 2, 3\}$. Thus, the matrix representation of $\mathcal{T}_{\mathbb{A}}^{\wedge}$ is, in terms of block matrices:

$$\mathcal{T}_{\mathbb{A}}^{\wedge} = \begin{bmatrix} 0 & X_3 & -X_2 \\ -X_3 & 0 & X_1 \\ X_2 & -X_1 & 0 \end{bmatrix}. \tag{3.1}$$

The formula (3.1) gives a way of flattening the tensor $\mathcal{T}$ into a matrix. In fact, let $\mathcal{X} \in \mathbb{R}^{3\times b\times c}$ be a third order tensor, as the tensorial representation for a color image of our numerical experiments in section 7. If the three frontal slices of $\mathcal{X}$ are denoted as X_1, X_2, and X_3, i.e., $X_i := \mathcal{X}_{i,:,:} \in \mathbb{R}^{b\times c}$ for $i = 1, 2, 3$, then (3.1) is the Strassen–Ottaviani flattening of $\mathcal{X}$. We see that it is different from the various matrix flattening methods employed in the literature for TCs, e.g., mode-n unfolding [34] and mode-k canonical unfolding [6], etc. This flattening was investigated first by Strassen [44] and then by Ottaviani [38], and is called the *Strassen–Ottaviani flattening*. For more details, we refer to [28] and references herein.

The following result is a corollary of [28, Theorem 3.8.1.1] in which the result is stated for the border rank of the underlying tensor.

Theorem 3.1. *Let $\mathcal{T} \in \mathbb{A}\otimes\mathbb{B}\otimes\mathbb{C}$, where $\mathbb{A}$, $\mathbb{B}$, and $\mathbb{C}$ are vector spaces of dimensions a, b, and c, respectively. Assume $3 \le a \le b \le c$, if $\mathrm{rank}(\mathcal{T}) \le r$, then $\mathrm{rank}(\mathcal{T}_{\mathbb{A}}^{\wedge}) \le r(a-1)$. If $a = 3$, $b = c$, and $\mathcal{T}$ is generic, then $\mathrm{rank}(\mathcal{T}_{\mathbb{A}}^{\wedge}) = 3b$.*

There are formulas in terms of a, b, c for the generic ranks of third order tensors [28]. Thus, the second part of Theorem 3.1 gives a relation between the CP rank of a generic tensor and the rank of its Strassen–Ottaviani flattening matrix. Actually, we can go further and show the following result. Let $\sigma_r \subseteq \mathbb{A}\otimes\mathbb{B}\otimes\mathbb{C}$ be the set of tensors with border rank at most r, i.e., the (Zariski) closure of the set of tensors with rank at most r. Let $\lfloor\alpha\rfloor$ be the largest integer smaller than α.

Theorem 3.2. *Let $\mathcal{T} \in \mathbb{A}\otimes\mathbb{B}\otimes\mathbb{C}$, where $\mathbb{A}$, $\mathbb{B}$, and $\mathbb{C}$ are vector spaces of dimensions a, b, and c, respectively. Assume $3 \le a \le b \le c$ and $r < \min\{\lfloor \frac{abc}{a+b+c-2}\rfloor, \lfloor \frac{a}{a-1}b\rfloor\}$, if $\mathcal{T} \in \sigma_r$ is a generic point, then $\mathrm{rank}(\mathcal{T}_{\mathbb{A}}^{\wedge}) = (a-1)r$.*

Proof. By definition, σ_r is an algebraic set. Let $\mathcal{L}(\mathbb{A}\otimes\mathbb{B}^*, \wedge^2\mathbb{A}\otimes\mathbb{C})$ be the set of linear mappings from $\mathbb{A}\otimes\mathbb{B}^*$ to $\wedge^2\mathbb{A}\otimes\mathbb{C}$, and $\phi : \sigma_r \to \mathcal{L}(\mathbb{A}\otimes\mathbb{B}^*, \wedge^2\mathbb{A}\otimes\mathbb{C})$ the mapping defined by

$$\phi(\mathcal{T}) := \mathcal{T}_{\mathbb{A}}^{\wedge}.$$

It is seen from the definition (cf. (3.1)) that ϕ is a polynomial mapping.

Let $Z \subseteq \mathbb{A} \otimes \mathbb{B} \otimes \mathbb{C}$ be the set of tensors such that $\operatorname{rank}(\mathcal{T}_{\mathbb{A}}^{\wedge}) < (a-1)r$ for all $\mathcal{T} \in Z$. Then Z is an algebraic variety defined via the $(a-1)r \times (a-1)r$ minors of the matrix (3.1) and the polynomial map ϕ.

Let $W := (\mathbb{R}^a \times \mathbb{R}^b \times \mathbb{R}^c)^r$ and $\psi : W \to \sigma_r$ be the mapping

$$\psi((\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i)) := \sum_{i=1}^{r} \mathbf{a}_i \otimes \mathbf{b}_i \otimes \mathbf{c}_i.$$

Let $\widehat{Z} := \psi^{-1}(Z)$. We have that $\widehat{Z}$ is a proper algebraic set of W, since we can find a point $\mathcal{T}$ such that the resulting $\operatorname{rank}(\mathcal{T}_{\mathbb{A}}^{\wedge}) = (a-1)r$. Thus, the mapping $\psi : W \setminus \widehat{Z} \to \sigma_r$ dominates, i.e., the image $\psi(W \setminus \widehat{Z})$ contains a Zariski dense open subset of σ_r [43]. Therefore, for a generic $\mathcal{T} \in \sigma_r$, each of its preimages of ψ is outside Z, and hence $\operatorname{rank}(\mathcal{T}_{\mathbb{A}}^{\wedge}) \geq (a-1)r$. On the other hand, Theorem 3.1 says that $\operatorname{rank}(\mathcal{T}_{\mathbb{A}}^{\wedge}) \leq (a-1)r$ for each tensor $\mathcal{T}$ with rank at most r. By the lower semicontinuity of the rank function, it follows that $\operatorname{rank}(\mathcal{T}_{\mathbb{A}}^{\wedge}) \leq (a-1)r$ for all tensors inside σ_r. In conclusion, it follows that a generic tensor in σ_r has $\operatorname{rank}(\mathcal{T}_{\mathbb{A}}^{\wedge}) = (a-1)r$. ∎

Theorem 3.2 indicates that if we know the underlying tensor is of admissible rank, then we can identify the generic case by merely testing the rank of the Strassen–Ottaviani matrix. The point is that the range of applicable ranks r for Theorem 3.2 can be larger than c, which is impossible for the classical mode flattening, since the latter is upper bounded by c. In applications, low ranks are always presumed for the underlying objects, such as color images, video data, etc, while experiments show that they are not actually of stringent low ranks. On the other hand, they are comprised of accumulation on the principle factors and a long tail distribution on the other featured factors. We could maintain major information by truncating the first factors with large magnitudes, while the other factors may provide crucial characteristic information. As the Strassen–Ottaviani flattening can handle much larger ranks, it allowsthe possibility in this situation.

From Theorems 3.1 and 3.2, we see that the rank of the representation matrix $\mathcal{T}_{\mathbb{A}}^{\wedge}$ can indeed indicate the low-rank property of the underlying tensor. We call $\operatorname{rank}(\mathcal{T}_{\mathbb{A}}^{\wedge})$ the *Strassen-Ottaviani rank* of the tensor $\mathcal{T}$, denoted as $\operatorname{rank}_{\mathrm{SO}}(\mathcal{X})$, an alternative to the rank, Tucker rank, TT rank, and tubal rank of $\mathcal{T}$.

For general $a > 3$, the matrix representation of $\mathcal{T}_{\mathbb{A}}^{\wedge}$ has a pattern similar to (3.1) but more complicated than it. For the sake of clarity, we present the subsequent analysis for the case $a = 3$, while, it extends in the same vein to the general case.

3.2. Projection. With the Strassen–Ottaviani flattening, we can introduce a polyhedron in the space of matrices related to the forthcoming TC problem.

Suppose our target tensor is in the space $\mathbb{R}^{3\times m\times n}$, and the corresponding matrix representation is in the space $\mathbb{R}^{3m\times 3n}$.

Let the variable matrix X be partitioned in the block form

$$X = \begin{bmatrix} X_{11} & X_{12} & X_{13} \\ X_{21} & X_{22} & X_{23} \\ X_{31} & X_{32} & X_{33} \end{bmatrix} \in \mathbb{R}^{3m\times 3n}, \tag{3.2}$$

where $X_{ij} \in \mathbb{R}^{m\times n}, i, j \in \{1, 2, 3\}$. The linear mapping $\mathcal{A} : \mathbb{R}^{3m\times 3n} \to \mathbb{R}^{m\times 6n}$ is defined by

$$\mathcal{A}(X) := \begin{bmatrix} X_{11} & X_{22} & X_{33} & X_{12}+X_{21} & X_{13}+X_{31} & X_{23}+X_{32} \end{bmatrix}. \tag{3.3}$$

Define a polyhedron P as

$$\mathsf{P} := \{X \in \mathbb{R}^{3m\times 3n} : \mathcal{A}(X) = 0, X_\Omega = O_\Omega\}, \tag{3.4}$$

where Ω is a subset of the indices indicating the observed elements of X via O_Ω, and it satisfies the constraints consistency:

$$\{X : X_\Omega = O_\Omega\} \subset \{X : \mathcal{A}(X) = 0\}.$$

In the following, we give the formula for the projection $\Pi_{\mathsf{P}}(Y)$ for any given Y, i.e.,

$$\Pi_{\mathsf{P}}(Y) := \arg\min_{X\in\mathsf{P}} \frac{1}{2}\|X - Y\|_F^2.$$

Proposition 3.3. *Let Y be given as*

$$Y = \begin{bmatrix} Y_1 & Y_2 & Y_3 \\ Y_4 & Y_5 & Y_6 \\ Y_7 & Y_8 & Y_9 \end{bmatrix},$$

where $Y_i \in \mathbb{R}^{m\times n}, i = 1, 2, \ldots, 9$, then

$$\Pi_{\mathsf{P}}(Y) = \begin{bmatrix} 0 & \frac{Y_2 - Y_4}{2} & \frac{Y_3 - Y_7}{2} \\ \frac{Y_4 - Y_2}{2} & 0 & \frac{Y_6 - Y_8}{2} \\ \frac{Y_7 - Y_3}{2} & \frac{Y_8 - Y_6}{2} & 0 \end{bmatrix}_{\Omega^c} + O_\Omega, \tag{3.5}$$

where Ω^c is the complement of Ω.

$$Y = \begin{bmatrix} Y_1 & Y_2 & Y_3 \\ Y_4 & Y_5 & Y_6 \\ Y_7 & Y_8 & Y_9 \end{bmatrix}$$

Proof. According to the definition of projection, computing $\Pi_{\mathsf{P}}(Y)$ is equivalent to solving the following constrained optimization problem:

$$\begin{aligned} \min_X \quad & \frac{1}{2}\|X - Y\|_F^2 \\ \text{s.t.} \quad & \mathcal{A}(X) = 0, \\ & X_\Omega = O_\Omega. \end{aligned} \tag{3.6}$$

We will first eliminate the constraint $\mathcal{A}(X) = 0$ by (3.3). By the linear operator $\mathcal{A}$, the candidate matrix X should take the following form

$$X = \begin{bmatrix} 0 & X_3 & -X_2 \\ -X_3 & 0 & X_1 \\ X_2 & -X_1 & 0 \end{bmatrix},$$

where $X_1, X_2, X_3 \in \mathbb{R}^{m\times n}$. Equipped with this form, by the separability of the Frobenius norm, the objective function of (3.6) becomes

$$\begin{aligned} &\frac{1}{2}(\|Y_1\|_F^2 + \|Y_5\|_F^2 + \|Y_9\|_F^2 + \|X_1 - Y_6\|_F^2 + \|X_1 + Y_8\|_F^2 \\ &\quad + \|X_2 - Y_7\|_F^2 + \|X_2 + Y_3\|_F^2 + \|X_3 - Y_2\|_F^2 + \|X_3 + Y_4\|_F^2) \end{aligned}$$

Note that since Y is given, then a direct calculation shows that (3.6) can be reformulated as

$$\begin{aligned} \min_{X_1,X_2,X_3} \quad & \left\|X_1 - \frac{Y_6 - Y_8}{2}\right\|_F^2 + \left\|X_2 - \frac{Y_7 - Y_3}{2}\right\|_F^2 + \left\|X_3 - \frac{Y_2 - Y_4}{2}\right\|_F^2 \\ \text{s.t.} \quad & X_\Omega = O_\Omega. \end{aligned} \tag{3.7}$$

Due to the independence of variables, problem (3.7) can be decomposed into three separate convex optimization problems as follows:

$$\min_{X_1,\ X_\Omega = O_\Omega} \left\|X_1 - \frac{Y_6 - Y_8}{2}\right\|_F^2, \quad \min_{X_2,\ X_\Omega = O_\Omega} \left\|X_2 - \frac{Y_7 - Y_3}{2}\right\|_F^2, \quad \min_{X_3,\ X_\Omega = O_\Omega} \left\|X_3 - \frac{Y_2 - Y_4}{2}\right\|_F^2.$$

The results follow from the separations of the Frobenius norm and the constraints $X_\Omega = O_\Omega$.

This concludes the proof. ∎

4. Tensor completion via Strassen–Ottaviani flattening.

In this section, based on the Strassen–Ottaviani flattening and the DCT [2], we introduce a new model for LRTC and propose a method for solving it.

4.1. Formulation.

A basic principle for TC is the hypothesis that the underlying tensor has a low-rank structure, a geometric sparsity characterization [28]. Thus, we can recover a tensor $\mathcal{X}$ through the following rank minimization:

$$\begin{aligned} \min \quad & \operatorname{rank}(\mathcal{X}) + r(\mathcal{X}) \\ \text{s.t.} \quad & \mathcal{X}_\Omega = \mathcal{O}_\Omega, \end{aligned} \tag{4.1}$$

where Ω is the index set of the observed elements, $\mathcal{O}_\Omega$ represents the observed elements, and $r(\mathcal{X})$ is a regularizer.

Problem (4.1) is NP-hard. Actually, computing the rank of a higher order tensor is already NP-hard [22]. A common approach is to reformulate the tensor $\mathcal{X}$ by its various matrix flattenings and then apply the matrix rank function and its surrogates to convey the underlying sparsity property. It follows from Theorems 3.1 and 3.2 that the Strassen–Ottaviani rank of $\mathcal{X}$ would be a good candidate. While the principle applies to general tensors, in the following, we will focus on third order tensors with dimension $3 \times m \times n$ for simplicity. Thus, with the characterizations (3.1) and (3.3), we propose the following model:

$$\begin{aligned} \min \quad & \mathcal{F}_\mu(X) + \lambda \|\mathcal{D}(X)\|_1 \\ \text{s.t.} \quad & \mathcal{A}(X) = 0, \\ & X_\Omega = O_\Omega, \end{aligned} \tag{4.2}$$

where $X \in \mathbb{R}^{3m \times 3n}$, $\mathcal{A}$ is the linear mapping given in (3.3), and $\mathcal{A}(X) = 0$ indicates that X is the Strassen–Ottaviani flattening of $\mathcal{X}$ by (3.1), O comes from $\mathcal{O}$ with compatible structure as X, $\mathcal{F}_\mu$ is a surrogate function for the rank of a matrix with a parameter μ, $\lambda \geq 0$ is a regularization parameter, and $\mathcal{D}$ is the discrete cosine transformation regarding X as a long column vector. Note that $\mathcal{D}$ is a given orthogonal linear mapping. Here we choose the DCT as a regularizer, which would be helpful particularly in color image processing [35, 46, 48]. The parameter μ allows an adaptive choice during an algorithm. It can be set as fixed as well.

We will abbreviate problem (4.2) as TCSO.

Our goal here is to recover a $3 \times m \times n$ tensor with partially observed entries indicated by Ω. With the formula (3.1) and the linear mapping (3.3), the rationale for recovery of the underlying tensor by (4.2) would be clear.

We see from (3.3) that the kernel of the linear operator $\mathcal{A}$ captures the Strassen–Ottaviani flattening. The choice of the surrogate function $\mathcal{F}_\mu$ shapes the optimization problem (4.2). If it is convex, such as the commonly employed nuclear norm, then the resulting problem is convex. If it is nonconvex, such as the ε-sparsity index to be applied, then the resulting problem is nonconvex. In general, the function $\mathcal{F}_\mu$ is nonsmooth and hence, nonsmooth techniques should be introduced for solving (4.2).

In the following, we present an exact recovery result for the classical nuclear norm surrogate without regularization, i.e., (4.2) with $\mathcal{F}_\mu(X) := \|X\|_*$ and $\lambda = 0$. Recall that the coherence $\mu(W)$ of a subspace $W \subseteq \mathbb{R}^n$ of dimension r is defined as (cf. [8])

$$\mu(W) := \frac{n}{r} \max_{1 \le i \le n} \|\mathrm{P}_W(\mathbf{e}_i)\|^2,$$

where P_W is the orthogonal projection onto W and $\mathbf{e}_i$ is the ith standard basis vector in $\mathbb{R}^n$.

Proposition 4.1. *Suppose that $\mathcal{T}$ is a given tensor with the Strassen–Ottaviani flattening M and $m \ge n$. Suppose that M has rank r and a singular value decomposition as $P\Sigma Q^\intercal$ such that*

1. *the row and column spaces have coherence bounded above by μ_0;*
2. *the matrix $PQ^\intercal$ has maximum entry bounded above by $\mu_1\sqrt{\frac{r}{10mn}}$.*

Suppose that K entries of the tensor $\mathcal{T}$ are observed with locations sampled uniformly at random. If

$$K \ge 3 * 32 \max\{\mu_0, \mu_1^2\} r(m+n)\beta \log^2(6m) \tag{4.3}$$

for some $\beta > 1$, then the minimizer to the problem (4.2) with $\mathcal{F}_\mu(X) := \|X\|_*$ and $\lambda = 0$ is unique, which is M, with probability at least $1 - 6\log(3m)(3m+3n)^{2-2\beta} - (3m)^{2-2\sqrt{\beta}}$.

Proof. The proof follows essentially from [39] with some modifications. To avoid repetition, we outline only the necessary changes. Unmentioned notations are consistent with those in [39]. Our problem now is

$$\begin{aligned} \min \quad & \|X\|_* \\ \text{s.t.} \quad & X_\Omega = M_\Omega \end{aligned} \tag{4.4}$$

with the matrix being

$$X = \begin{bmatrix} 0 & X_3 & -X_2 \\ -X_3 & 0 & X_1 \\ X_2 & -X_1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & U \\ V & 0 \end{bmatrix},$$

where U represents the upper triangular part consisting of X_1, X_2, X_3 and V is completely determined by U. Note that the sampling is essentially taken over the part U. The biggest difference between our problem and the classical matrix completion problem is that the diagonal part of our matrix is sampled with certainty. Let $\mathcal{U}_D$ be the operator taking the diagonal part of X, and $\mathcal{U}_U$ be the operator taking the upper triangular part, $\mathcal{R}_\Omega$ the sampling operator on the upper triangular part. Then, for a probability p, it follows that $\mathbb{E}[\mathcal{R}_\Omega] = p\mathcal{U}_U$.

The analysis in [39] is applied to the whole matrix X, as the sampling is made on the whole set of indices of X. For our problem, we have to take the analysis focusing on the upper

triangular part only. An essential change is to separate the diagonal part and the upper triangular part. For example, in the definition of the sequence $\{Y_k\}$ on [39, p. 3243], instead of $Y_k := \frac{n_1 n_2}{q} \sum_{j=1}^{k} \mathcal{R}_{\Omega_j}(W_{j-1})$ (n_1, n_2 corresponding to the matrix size in that paper), we define

$$Y_k := \sum_{j=1}^{k} \left[\mathcal{U}_D(W_{j-1}) + \frac{3mn}{q} (\mathcal{R}_{\Omega_j} + \mathcal{R}_{\Omega_j'})(W_{j-1}) \right],$$

where Ω_j' is the corresponding lower triangular index set of Ω_j. As

$$\mathbb{E}[\mathrm{P}_T(\mathcal{U}_D + \mathcal{R}_\Omega)\,\mathrm{P}_T] = \mathrm{P}_T \mathcal{U}_D\,\mathrm{P}_T + p\,\mathrm{P}_T \mathcal{U}_U\,\mathrm{P}_T,$$

we transferred the analysis from the whole matrix as in [39] to the upper triangular part by the above adjustment.

Following the proof in [39], the conclusion can be established. ∎

Both the sampling size and the probability are direct applications of [39, Theorem 2]. In light of the proof, by the structure of our problem, there should be space to improve these bounds. For the sake of paper length, we would like to defer these constant improvements to future investigations. While, as noted in [39], the sampling bound is optimal in magnitude of the matrix size and, therefore, it is optimal in magnitude of the tensor size for our content.

Combining Theorem 3.2 and Proposition 4.1, we see that under reasonable hypotheses, we can exactly recover a generic tensor in σ_r with high probability by solving the matrix completion problem (4.2) using the Strassen–Ottaviani flattening.

4.2. Solution methods. In this subsection, we apply the classical inexact ALM for solving problem (4.2).

By introducing the indicator function and auxiliary variables, (4.2) can be equivalently rewritten as

$$\begin{aligned} \min \quad & \mathcal{F}_\mu(Y) + \lambda \|Z\|_1 + \delta_{\mathsf{P}}(X) \\ \text{s.t.} \quad & X = Y,\ \mathcal{D}(X) = Z, \end{aligned} \tag{4.5}$$

where the variables can be grouped into $(X, U) := (X, (Y, Z))$, and the indicator function δ_{P} is defined as $\delta_{\mathsf{P}}(X) := 0$ if $X \in \mathsf{P}$ and ∞ otherwise.

Proposition 4.2. *Each local optimal solution of problem* (4.5) *is a KKT point.*

Proof. KKT points of problem (4.5) can be written down combing Proposition 2.4 with results in [41, Exercise 8.8, Theorem 8.2, and Proposition 10.5] as follows:

$$S \in \mathcal{D}^{\intercal}(W) + N_{\mathsf{P}}(X),\ 0 \in S + \partial \mathcal{F}_\mu(Y) \text{ and } W \in \lambda \partial \|Z\|_1 \tag{4.6}$$

for a multiplier pair (S, W). Moreover, the constraint qualification in Proposition 2.4 is satisfied. Actually, if we define $V := (X, Y, Z) \in \mathsf{S}$ with $\mathsf{S} := \mathbb{R}^{3m\times 3n} \times \mathbb{R}^{3m\times 3n} \times \mathbb{R}^{3m\times 3n}$, $F_1(V) := X - Y$, and $F_2(V) := \mathcal{D}(X) - Z$ in (2.4), then all requirements in Proposition 2.4 for the objective and constraint functions and the abstract set S are satisfied. The constraint qualification reduces to

$$\begin{bmatrix} Y_1 \\ -Y_1 \\ 0 \end{bmatrix} + \begin{bmatrix} \mathcal{D}^{\intercal}(Y_2) \\ 0 \\ -Y_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \Longrightarrow Y_1 = Y_2 = 0,$$

which holds trivially. Thus, the conclusion follows from Proposition 2.4. ∎

The augmented Lagrangian function of problem (4.5) is

$$L_{\beta,\mu}(X,U;\Lambda) := \mathcal{F}_\mu(Y) + \lambda\|Z\|_1 + \delta_{\mathsf{P}}(X) + \langle \Lambda_1, X - Y\rangle + \langle \Lambda_2, \mathcal{D}(X) - Z\rangle + \frac{\beta}{2}\|X - Y\|_F^2 + \frac{\beta}{2}\|\mathcal{D}(X) - Z\|_F^2, \tag{4.7}$$

where $\Lambda := (\Lambda_1, \Lambda_2)$ is the Lagrangian multiplier, and $\beta > 0$ is the penalty parameter.

A standard inexact ALM for solving (4.5) is given as follows in Algorithm 4.1.

Algorithm 4.1. Inexact ALM.

Initialization: Given a nonincreasing positive sequence $\{\epsilon^{(k)}\}$ such that $\epsilon^{(k)} \to 0$ as $k \to \infty$, choose initial $(X^{(0)}, U^{(0)}; \Lambda^{(1)})$, parameter $\mu^{(0)}$, penalty parameters $\beta^{(0)} = \beta^{(1)} > 0$, a penalty adjustment parameter $\kappa > 1$, and a penalty adjustment threshold parameter $\gamma \in (0,1)$. Set $k := 1$.

Step 1: Solve the subproblem

$$(X^{(k)}, U^{(k)}) \approx \arg\min_{X,U} L_{\beta^{(k)},\mu^{(k-1)}}(X, U; \Lambda^{(k)}) \tag{4.8}$$

such that $X^{(k)} \in \mathsf{P}$,

$$L_{\beta^{(k)},\mu^{(k)}}(X^{(k)}, U^{(k)}; \Lambda^{(k)}) \le L_{\beta^{(k)},\mu^{(k-1)}}(X^{(k-1)}, U^{(k-1)}; \Lambda^{(k)}), \tag{4.9}$$

where $\mu^{(k)}$ is updated by a certain rule from $Y^{(k)}$, and

$$\operatorname{dist}(0, \partial L_{\beta^{(k)},\mu^{(k-1)}}(X^{(k)}, U^{(k)}; \Lambda^{(k)})) \le \epsilon^{(k)}. \tag{4.10}$$

Step 2: Update the multiplier as

$$\Lambda_1^{(k+1)} := \Lambda_1^{(k)} + \beta^{(k)}(X^{(k)} - Y^{(k)}) \text{ and } \Lambda_2^{(k+1)} := \Lambda_2^{(k)} + \beta^{(k)}(\mathcal{D}(X^{(k)}) - Z^{(k)}). \tag{4.11}$$

Step 3: Update the penalty parameter as

$$\beta^{(k+1)} := \begin{cases} \beta^{(k)} & \text{if } \dfrac{\beta^{(k-1)}\sqrt{\|\Lambda_1^{(k+1)} - \Lambda_1^{(k)}\|^2 + \|\Lambda_2^{(k+1)} - \Lambda_2^{(k)}\|^2}}{\beta^{(k)}\sqrt{\|\Lambda_1^{(k)} - \Lambda_1^{(k-1)}\|^2 + \|\Lambda_2^{(k)} - \Lambda_2^{(k-1)}\|^2}} \le \gamma, \\ \kappa\beta^{(k)} & \text{otherwise.} \end{cases} \tag{4.12}$$

Step 4: Unless a termination criterion is fulfilled, set $k := k + 1$ and go to Step 1.

A key point in Algorithm 4.1 is that an adaptive parameter strategy for $\mu^{(k)}$ is allowed. This is employed as heuristics in the literature, without a solid analysis however. We will give a convergence analysis for it in the following; cf. Theorem 4.7 and Theorem 5.8. The

augmented Lagrangian subproblem (4.8) can be solved by a standard proximal alternating minimization method (PAMM) [4], which deals with a class of nonsmooth and nonconvex optimization problems. In general, the PAMM generates a point of (4.8), satisfies (4.10), and

$$\begin{aligned}(4.13)\qquad &\tau_k\|(X^{(k)},U^{(k)})-(X^{(k-1)},U^{(k-1)})\|^2+L_{\beta^{(k)},\mu^{(k-1)}}(X^{(k)},U^{(k)};\Lambda^{(k)})\\ &\le L_{\beta^{(k)},\mu^{(k-1)}}(X^{(k-1)},U^{(k-1)};\Lambda^{(k)}),\end{aligned}$$

where τ_k depends on the proximal parameters and the number of iterations; see (4.18). Thus, it leaves a lot of leeway for choosing $\mu^{(k)}$ such that (4.9) is further fulfilled.

For simplicity of presentation for PAMM, we will omit the outer iteration index of Algorithm 4.1. Thus, for the given β, μ, and Λ, each subproblem has the following form

$$(4.14)\qquad \min_{X,U} Q(X,U) := L_{\beta,\mu}(X,U;\Lambda).$$

A widely employed PAMM for solving (4.14) is given as follows in Algorithm 4.2.

Algorithm 4.2. PAMM.

Initialization: Choose $(X^{(0)},U^{(0)})$, proximal parameters $\rho_1,\rho_2\ge 0$, and a tolerance $\epsilon\ge 0$. Set $k:=1$.

Step 1: Compute

$$(4.15)\qquad U^{(k)}\in \arg\min_{U} Q(X^{(k-1)},U)+\frac{\rho_1}{2}\|U-U^{(k-1)}\|^2.$$

Step 2: Compute

$$(4.16)\qquad X^{(k)}:=\arg\min_{X} Q(X,U^{(k)})+\frac{\rho_2}{2}\|X-X^{(k-1)}\|^2.$$

Step 3: Unless $\|\Theta^{(k)}\|\le\epsilon$ (see (4.17) for detail), set $k:=k+1$ and go to Step 1.

In Algorithm 4.2, $\Theta^{(k)}$ in the stopping criterion is defined as

$$(4.17)\qquad \Theta^{(k)}:=\begin{bmatrix}\rho_2(X^{(k-1)}-X^{(k)})\\ \beta(X^{(k-1)}-X^{(k)})+\rho_1(Y^{(k-1)}-Y^{(k)})\\ \beta\mathcal{D}(X^{(k-1)}-X^{(k)})+\rho_1(Z^{(k-1)}-Z^{(k)})\end{bmatrix}.$$

4.3. Convergence analysis. In this subsection, we will establish the global convergence of Algorithm 4.1 for solving (4.5) under fairly mild conditions.

Since $\mathcal{F}_\mu$ is a surrogate for the rank function, we assume without loss of generality that $\mathcal{F}_\mu$ takes nonnegative values for all allowable parameters μ throughout this section.

We will first show the convergence of Algorithm 4.2 and the well definiteness of Step 1 in Algorithm 4.1. We first show that Algorithm 4.2 converges.

Lemma 4.3. *Let $\{X^{(k)}, U^{(k)}\}$ be a sequence generated by Algorithm 4.2 for $\epsilon = 0$. Then, the sequence $\{X^{(k)}, U^{(k)}\}$ is bounded and converges to a critical point of the function Q defined in (4.14).*

Proof. From Algorithm 4.2, we obtain

$$Q(X^{(k-1)}, U^{(k)}) + \frac{\rho_1}{2}\left(\|Y^{(k)} - Y^{(k-1)}\|^2 + \|Z^{(k)} - Z^{(k-1)}\|^2\right) \le Q(X^{(k-1)}, U^{(k-1)})$$

and

$$Q(X^{(k)}, U^{(k)}) + \frac{\rho_2}{2}\|X^{(k)} - X^{(k-1)}\|^2 \le Q(X^{(k-1)}, U^{(k)}).$$

By combining the above inequalities, we have

$$\begin{aligned} Q(X^{(k)}, U^{(k)}) + \frac{\rho}{2}\left(\|X^{(k)} - X^{(k-1)}\|^2 + \|Y^{(k)} - Y^{(k-1)}\|^2 + \|Z^{(k)} - Z^{(k-1)}\|^2\right) \\ \le Q(X^{(k-1)}, U^{(k-1)}), \end{aligned} \tag{4.18}$$

where $\rho := \min\{\rho_1, \rho_2\}$. Thus the sequence $\{Q(X^{(k)}, U^{(k)})\}$ is bounded from above. By the definition of Q (cf. (4.14) and (4.7)), we have that both $\{X^{(k)} - Y^{(k)}\}$ and $\{\mathcal{D}(X^{(k)}) - Z^{(k)}\}$ must be bounded, since Q is defined by the summation of nonnegative functions and strongly convex quadratic functions of them. Then, $\{Z^{(k)}\}$ must be bounded by the l_1-norm component, which in turn implies the boundedness of $\{X^{(k)}\}$ and $\{Y^{(k)}\}$.

Obviously, the function Q is a Kurdyka–Łojasiewicz function, and the conclusion follows from [4, Theorem 6.2]. ∎

Proposition 4.4. *If $\epsilon > 0$, then Algorithm 4.2 will terminate in finitely many steps, and compute a point $(X^{(k)}, U^{(k)})$ satisfying*

$$Q(X^{(k)}, U^{(k)}) \le Q(X^{(0)}, U^{(0)}) \text{ and } \|\Theta^{(k)}\| \le \epsilon \text{ with } \Theta^{(k)} \in \partial Q(X^{(k)}, U^{(k)}). \tag{4.19}$$

Proof. It follows from Lemma 4.3 that the sequence $\{X^{(k)}, U^{(k)}\}$ generated by Algorithm 4.2 converges. Thus, it follows from the formula of (4.17) that $\Theta^{(k)} \to 0$ as $k \to \infty$. Since $\epsilon > 0$ is a prescribed positive constant, the conclusion of finite convergence then follows.

The decrease of the objective function follows from the proof of Lemma 4.3. In the following, we show that $\Theta^{(k)} \in \partial Q(X^{(k)}, U^{(k)})$.

Actually, from the optimality conditions of (4.15) and (4.16), we have

$$\begin{aligned} \Lambda_1 + \beta(X^{(k-1)} - Y^{(k)}) - \rho_1(Y^{(k)} - Y^{(k-1)}) &\in \partial \mathcal{F}_\mu(Y^{(k)}), \\ \Lambda_2 + \beta(\mathcal{D}(X^{(k-1)}) - Z^{(k)}) - \rho_1(Z^{(k)} - Z^{(k-1)}) &\in \lambda \partial \|Z^{(k)}\|_1, \\ -\Lambda_1 - \beta(X^{(k)} - Y^{(k)}) - \beta(X^{(k)} - \mathcal{D}^\intercal(Z^{(k)})) & \\ -\mathcal{D}^\intercal(\Lambda_2) - \rho_2(X^{(k)} - X^{(k-1)}) &\in \partial \delta_{\mathsf{P}}(X^{(k)}). \end{aligned}$$

Compared with the critical points of the augmented Lagrangian function (cf. (4.7)), we have

$$\Theta^{(k)} \in \partial L_{\beta,\mu}(X^{(k)}, U^{(k)}; \Lambda) = \partial Q(X^{(k)}, U^{(k)}),$$

where the equality follows from the definition of Q in (4.14).

The proof is complete. ∎

The stopping criterion of Algorithm 4.2 is computing an ϵ-perturbation of the critical point property

$$0 \in \partial L_{\beta,\mu}(X^{(k)}, U^{(k)}; \Lambda).$$

Proposition 4.4 shows that Step 1 in Algorithm 4.1 is well defined and can be realized by Algorithm 4.2 if either μ is fixed or (4.9) can be fulfilled for concrete adaptive choices on μ.

Equipped with this, in the following, we establish the global convergence of Algorithm 4.1, under the following assumptions.

Assumption 4.5. Elements in $\partial \mathcal{F}_\mu$ are bounded uniformly in $\mathrm{dom}(\mathcal{F}_\mu)$ and for all allowable μ. The set-valued mapping $\mu \mapsto \partial \mathcal{F}_\mu(Y)$ is upper semicontinuous for all $Y \in \cup_\mu \mathrm{dom}(\mathcal{F}_\mu)$.

We note that Assumption 4.5 holds for a broad range of surrogate functions. For example, the weighted nuclear norms and ε-sparsity index all satisfy this assumption; the proofs are deferred to section 5.3.

We make the following assumption on the sequence of parameters $\{\mu^{(k)}\}$ in Algorithm 4.1.

Assumption 4.6. If a subsequence $\{Y^{(i_k)}\}$ converges, then $\{\mu^{(i_k)}\}$ has an accumulation point.

Assumption 4.6 is trivial for fixed μ, and it also holds for a wide range of choices.

Theorem 4.7. *Suppose that $\lambda > 0$, and Assumptions 4.5 and 4.6 hold. Let the sequence $\{X^{(k)}, U^{(k)}, \Lambda^{(k)}\}$ be generated by Algorithm 4.1. Then, it is bounded and any accumulation point of $\{X^{(k)}, U^{(k)}\}$ is a KKT point of problem (4.5) for some μ^*.*

Proof. By Algorithm 4.1, there exists a vector $\Theta^{(k)} \in \partial L_{\beta^{(k)},\mu^{(k-1)}}(X^{(k)}, U^{(k)}; \Lambda^{(k)})$ such that $\|\Theta^{(k)}\| \le \epsilon^{(k)}$ which is guaranteed by (4.10). Thus, by [41, Exercise 8.8 and Proposition 10.5], we obtain

$$\Theta^{(k)} = \begin{bmatrix} N^{(k)} + \Lambda_1^{(k)} + \mathcal{D}^\intercal(\Lambda_2^{(k)}) + \beta^{(k)}(X^{(k)} - Y^{(k)}) + \beta^{(k)}(X^{(k)} - \mathcal{D}^\intercal(Z^{(k)})) \\ H^{(k)} - \Lambda_1^{(k)} - \beta^{(k)}(X^{(k)} - Y^{(k)}) \\ \lambda V^{(k)} - \Lambda_2^{(k)} - \beta^{(k)}(\mathcal{D}(X^{(k)}) - Z^{(k)}) \end{bmatrix} \tag{4.20}$$

for some

$$N^{(k)} \in \partial \delta_{\mathsf{P}}(X^{(k)}),\ H^{(k)} \in \partial \mathcal{F}_{\mu^{(k-1)}}(Y^{(k)}) \text{ and } V^{(k)} \in \partial \|Z^{(k)}\|_1.$$

It follows from the last row in (4.20) and the uniform boundedness of the subdifferential $\partial \|Z^{(k)}\|_1$ that $\{\Lambda_2^{(k)} + \beta^{(k)}(\mathcal{D}(X^{(k)}) - Z^{(k)})\}$ is bounded. It then follows from the multiplier update rule (4.11) that $\{\Lambda_2^{(k)}\}$ is bounded. Likewise, by Assumption 4.5 and (4.11), it follows from the second row in (4.20) that $\{\Lambda_1^{(k)}\}$ is bounded.

Therefore, we have that both the sequences

$$\{\beta^{(k)}(X^{(k)} - Y^{(k)})\} \text{ and } \{\beta^{(k)}(\mathcal{D}(X^{(k)}) - Z^{(k)})\} \tag{4.21}$$

are bounded. Take a common upper bound $M > 0$ for them, i.e., for all k, it holds that

$$\beta^{(k)}\|X^{(k)} - Y^{(k)}\| \le M \text{ and } \beta^{(k)}\|\mathcal{D}(X^{(k)}) - Z^{(k)}\| \le M. \tag{4.22}$$

In the following, we will show that the sequence $\{L_{\beta^{(k)},\mu^{(k)}}(X^{(k)}, U^{(k)}; \Lambda^{(k)})\}$ is bounded from above. For the sake of notational simplicity, let

$$E^{(k)} := \sqrt{\|X^{(k)} - Y^{(k)}\|^2 + \|\mathcal{D}(X^{(k)}) - Z^{(k)}\|^2}.$$

It follows from (4.22) that

$$\beta^{(k)} E^{(k)} \le \sqrt{2} M. \tag{4.23}$$

Then, if we define

$$\begin{aligned}
\triangle_k &:= L_{\beta^{(k)},\mu^{(k)}}(X^{(k)}, U^{(k)}; \Lambda^{(k+1)}) - L_{\beta^{(k)},\mu^{(k)}}(X^{(k)}, U^{(k)}; \Lambda^{(k)}) \\
&= \beta^{(k)} \left(\|X^{(k)} - Y^{(k)}\|^2 + \|\mathcal{D}(X^k) - Z^{(k)}\|^2 \right), \\
\triangle'_k &:= L_{\beta^{(k+1)},\mu^{(k)}}(X^{(k)}, U^{(k)}; \Lambda^{(k+1)}) - L_{\beta^{(k)},\mu^{(k)}}(X^{(k)}, U^{(k)}; \Lambda^{(k+1)}) \\
&= \frac{\beta^{(k+1)} - \beta^{(k)}}{2} \left(\|X^{(k)} - Y^{(k)}\|^2 + \|\mathcal{D}(X^k) - Z^{(k)}\|^2 \right), \\
\eta_k &:= L_{\beta^{(k+1)},\mu^{(k)}}(X^{(k)}, U^{(k)}; \Lambda^{(k+1)}) - L_{\beta^{(k+1)},\mu^{(k+1)}}(X^{(k+1)}, U^{(k+1)}; \Lambda^{(k+1)}),
\end{aligned}$$

then we have

$$\begin{aligned}
&L_{\beta^{(k+1)},\mu^{(k+1)}}(X^{(k+1)}, U^{(k+1)}; \Lambda^{(k+1)}) - L_{\beta^{(k)},\mu^{(k)}}(X^{(k)}, U^{(k)}; \Lambda^{(k)}) \\
&= \triangle_k + \triangle'_k - \eta_k = \frac{\beta^{(k)} + \beta^{(k+1)}}{2} (E^{(k)})^2 - \eta_k \le \frac{\beta^{(k)} + \beta^{(k+1)}}{2} (E^{(k)})^2,
\end{aligned}$$

where the last inequality follows from the nonnegativity of η_k (cf. (4.9)). This, together with the penalty parameter update rule (4.12), implies that

$$L_{\beta^{(k+1)},\mu^{(k+1)}}(X^{(k+1)}, U^{(k+1)}; \Lambda^{(k+1)}) - L_{\beta^{(k)},\mu^{(k)}}(X^{(k)}, U^{(k)}; \Lambda^{(k)}) \le \frac{(\kappa+1)\beta^{(k)}}{2} (E^{(k)})^2. \tag{4.24}$$

(4.23), together with (4.24), implies that

$$L_{\beta^{(k+1)},\mu^{(k+1)}}(X^{(k+1)}, U^{(k+1)}; \Lambda^{(k+1)}) - L_{\beta^{(k)},\mu^{(k)}}(X^{(k)}, U^{(k)}; \Lambda^{(k)}) \le C E^{(k)}, \tag{4.25}$$

where $C := \frac{(\kappa+1)\sqrt{2}M}{2}$.

In the following, we consider the following two cases:

1. if $\beta^{(k+1)} = \beta^{(k)}$, then $E^{(k)} \le \gamma E^{(k-1)}$ (cf. (4.12)); and
2. if $\beta^{(k+1)} = \kappa\beta^{(k)}$, then $E^{(k)} \le \frac{\sqrt{2}M}{\beta^{(k)}}$ (cf. (4.23)).

In the following, we analyze the penalty parameter sequence $\{\beta^{(k)}\}$ generated by Algorithm 4.1. Suppose $\{\beta^{i_j}\}$ is exactly the subsequence with penalty parameter adjustments, with $\beta^{i_0} = \beta^{(1)}$. Thus, we have that $\beta^{i_j} = \kappa\beta^{i_{j-1}}$ and $\beta^{i_j} = \beta^{i_j+1} = \cdots = \beta^{i_{j+1}-1} = \kappa^j \beta^{(1)}$. Consequently, inside the iterations from i_j to i_{j+1}, the sequence $\{E^{(k)}\}$ is dominated by a

geometric progression with ratio γ by case 1; the sequence corresponding to $\{\beta^{i_j}\}$ is also dominated by a geometric progression with ratio $\frac{1}{\kappa}$ by case 2.

For a given integer $K > 0$, assume that $i_{s-1} \le K < i_s$ with some $s \ge 1$. Thus, summing (4.25) from $k = 0$ to $k = K - 1$, we have

$$\begin{aligned}
&L_{\beta^{(K)},\mu^{(K)}}(X^{(K)}, U^{(K)}; \Lambda^{(K)}) - L_{\beta^{(0)},\mu^{(0)}}(X^{(0)}, U^{(0)}; \Lambda^{(0)}) \\
&\quad \le \frac{\sqrt{2}MC}{\beta^{(1)}}(1 + \gamma + \cdots + \gamma^{i_1 - 2}) + \frac{\sqrt{2}MC}{\beta^{(1)}}(1 + \gamma + \cdots + \gamma^{i_2 - i_1 - 1}) \\
&\qquad + \frac{\sqrt{2}MC}{\kappa\beta^{(1)}}(1 + \gamma + \cdots + \gamma^{i_3 - i_2 - 1}) + \cdots + \frac{\sqrt{2}MC}{\kappa^{s-3}\beta^{(1)}}(1 + \gamma + \cdots + \gamma^{i_{s-1} - i_{s-2} - 1}) \\
&\qquad + \frac{\sqrt{2}MC}{\kappa^{s-2}\beta^{(1)}}(1 + \gamma + \cdots + \gamma^{K - i_{s-1}}) \\
&\quad = \frac{(1 - \gamma^{i_1 - 1})\sqrt{2}MC}{(1 - \gamma)\beta^{(1)}} + \frac{(1 - \gamma^{i_2 - i_1})\sqrt{2}MC}{(1 - \gamma)\beta^{(1)}} + \frac{(1 - \gamma^{i_3 - i_2})\sqrt{2}MC}{(1 - \gamma)\kappa\beta^{(1)}} + \cdots \\
&\qquad + \frac{(1 - \gamma^{i_{s-1} - i_{s-2}})\sqrt{2}MC}{(1 - \gamma)\kappa^{s-3}\beta^{(1)}} + \frac{(1 - \gamma^{K - i_{s-1} + 1})\sqrt{2}MC}{(1 - \gamma)\kappa^{s-2}\beta^{(1)}} \\
&\quad \le \frac{\sqrt{2}MC}{(1 - \gamma)\beta^{(1)}} + \frac{\sqrt{2}MC}{(1 - \gamma)\beta^{(1)}} + \frac{\sqrt{2}MC}{(1 - \gamma)\kappa\beta^{(1)}} + \cdots + \frac{\sqrt{2}MC}{(1 - \gamma)\kappa^{s-3}\beta^{(1)}} + \frac{\sqrt{2}MC}{(1 - \gamma)\kappa^{s-2}\beta^{(1)}} \\
&\quad = \frac{\sqrt{2}MC}{(1 - \gamma)\beta^{(1)}} + \frac{\kappa\sqrt{2}MC(1 - \frac{1}{\kappa^{s-1}})}{(\kappa - 1)(1 - \gamma)\beta^{(1)}} \le \frac{\sqrt{2}MC}{(1 - \gamma)\beta^{(1)}} + \frac{\kappa\sqrt{2}MC}{(\kappa - 1)(1 - \gamma)\beta^{(1)}}.
\end{aligned}$$

Note that for all large k, if only case 1 or only case 2 occurs, the analysis works as well. Consequently, $\{L_{\beta^{(k)},\mu^{(k)}}(X^{(k)}, U^{(k)}; \Lambda^{(k)})\}$ is bounded from above.

In the following, we show that the iteration sequence $\{X^{(k)}, U^{(k)}\}$ is bounded. First of all, we claim that $\{Z^{(k)}\}$ must be bounded. Assume on the contrary that $\{Z^{(k)}\}$ is unbounded. By the formula of the augmented Lagrangian function (cf. (4.7)), $\{\mathcal{D}(X^{(k)}) - Z^{(k)}\}$ must be bounded, since otherwise the quadratic term will dominate and the function value will be unbounded from the above. Guaranteed by this, terms involving $\mathcal{D}(X^{(k)}) - Z^{(k)}$ are bounded. Thus, $\{Z^{(k)}\}$ is bounded, since otherwise $\{\lambda\|Z^{(k)}\|_1\}$ will be unbounded for positive $\lambda > 0$, which will violate the boundedness of $\{L_{\beta^{(k)},\mu^{(k)}}(X^{(k)}, U^{(k)}; \Lambda^{(k)})\}$.

Thus, $\{X^{(k)}\}$ must be bounded since $\mathcal{D}$ is an orthogonal linear operator. A similar argument shows that $\{X^{(k)} - Y^{(k)}\}$ must be bounded and hence $\{Y^{(k)}\}$ is bounded. Thus, the boundedness follows.

In the following, let (X^*, U^*) be an accumulation point of the sequence $\{X^{(k)}, U^{(k)}\}$.

Depending on whether the sequence $\{\beta^{(k)}\}$ is bounded or not, we have the following two cases.

Case I. Assume that $\{\beta^{(k)}\}$ is unbounded, i.e., $\beta^{(k)} \to \infty$ as $k \to \infty$. In this case, it follows from (4.22) that

$$X^{(k)} - Y^{(k)} \to 0, \text{ and } \mathcal{D}(X^{(k)}) - Z^{(k)} \to 0.$$

Case II. Assume that $\{\beta^{(k)}\}$ is bounded, that is, $\beta^{(k)}$ stabilizes after some k_0, i.e., $\beta^{(k)} = \beta^{(k_0)}$ for all $k \ge k_0$. Thus, by the penalty parameter update rule (4.12), we obtain

$$\sqrt{\|X^{(k)}-Y^{(k)}\|^2+\|\mathcal{D}(X^k)-Z^{(k)}\|^2}$$
$$\leq \gamma^{k-k_0+1}\sqrt{\|X^{(k_0-1)}-Y^{(k_0-1)}\|^2+\|\mathcal{D}(X^{k_0-1})-Z^{(k_0-1)}\|^2}$$

for all $k \geq k_0$. Therefore, we have $X^{(k)}-Y^{(k)} \to 0$ and $\mathcal{D}(X^{(k)})-Z^{(k)} \to 0$. Consequently, the accumulation point (X^*, U^*) is a feasible point of problem (4.5).

Along the subsequence for the accumulation point (X^*, U^*), taking a subsequence if necessary, let W^* and S^* be accumulation points of the sequences in (4.21), respectively, and Λ^* be an accumulation point of the sequence $\{\Lambda^{(k)}\}$. Note that the sequence $\{N^{(k)}\}$ in (4.20) is then bounded as well. Let the converging subsequence be indexed by $\{i_k\}$. Then, we have that $\{Y^{(i_k-1)}\}$ is also a bounded sequence, since it is a subsequence of the bounded sequence of $\{Y^{(k)}\}$. Taking a subsequence if necessary, we can assume that $\{Y^{(i_k-1)}\}$ converges. Thus, we assume that the corresponding accumulation point of $\{\mu^{(i_k-1)}\}$ is μ^* by Assumption 4.6.

By the fact that $\Theta^{(k)} \to 0$ (cf. $\epsilon^{(k)} \to 0$) and the upper semicontinuousness of the subdifferentials and Assumption 4.5, we have from (4.20) that

$$-(\Lambda_1^* + \mathcal{D}^\intercal(\Lambda_2^* + S^*) + W^*) \in \partial\delta_{\mathsf{P}}(X^*),\ \Lambda_1^* + W^* \in \partial\mathcal{F}_{\mu^*}(Y^*),\ \text{and}\ \Lambda_2^* + S^* \in \lambda\partial\|Z^*\|_1.$$

Hence, it follows that (X^*, U^*) is a KKT point of problem (4.5) with Lagrange multiplier $(\Lambda_1^* + W^*, \Lambda_2^* + S^*)$ (cf. (4.6)). ∎

By a careful analysis on the penalty parameters and an elaboration on bounding the decrease of the augmented objective functions, Theorem 4.7 is established without any assumptions except certain minimal assumptions on the parameterized surrogate function $\mathcal{F}_\mu$, which can be waived for a large class of $\mathcal{F}_\mu$ (cf. Theorem 5.8). It improves the literature [50]. Different choices of $\mathcal{F}_\mu$ in (4.5) correspond to different models. In the following sections, we will present more details on subproblem solving in PAMM. We will introduce two classes of models: nonconvex models that are based on nonconvex surrogates and convex models that rely on convex surrogates.

5. Nonconvex spectral functions.

In this section, we will present more details on the subproblem solving for PAMM (i.e., Algorithm 4.2). The parameter μ appears only when we discuss the convergence in section 5.3. Thus, we will omit it in the other subsections for simplicity. We can see from the analysis that it holds for both convex and nonconvex surrogates $\mathcal{F}$. However, we would like to put more emphasis on nonconvex surrogates. In particular, the ε-sparsity index, weighted nuclear norms via nondecreasing weights and adaptive weights will be given.

If the surrogate $\mathcal{F}$ is convex, the optimization problem (4.5) is convex as well. In this case, we have a simpler algorithmic implementation, which will be given in section 6.

5.1. Proximal mappings.

For a proper, lower semicontinuous function $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ and a parameter $\lambda > 0$, the proximal mapping $\text{prox}_\lambda(f)$ is defined as

$$\text{prox}_\lambda(f)(\mathbf{x}) := \arg\min_{\mathbf{w}} \left\{ f(\mathbf{w}) + \frac{1}{2\lambda}\|\mathbf{w}-\mathbf{x}\|^2 \right\}. \tag{5.1}$$

For nonconvex f, $\mathrm{prox}_\lambda(f)$ is a set-valued mapping in the general case. By abuse of notation, we will also use $\mathrm{prox}_\lambda(f)(\mathbf{x})$ to refer to one solution in (5.1). This will not bring any ambiguity in this paper.

The following is a classical result; see, for example, [35, Proposition 2.1].

Lemma 5.1. *For a positive $\lambda>0$ and a given vector $\mathbf{y}\in\mathbb{R}^n$, we have*

$$\mathrm{prox}_\lambda(\|\cdot\|_1)(\mathbf{y})=\mathrm{shrink}(\mathbf{y},\lambda):=\mathrm{sgn}(\mathbf{y})\odot\max(|\mathbf{y}|-\lambda,\mathbf{0}),$$

where sgn *takes the sign of a vector component-wisely, $\odot$ is the Hadamard product, and $|\mathbf{y}|$ takes the absolute values of $\mathbf{y}$ component-wisely.*

We note that when shrink is applied to a matrix variable, we mean the application to the vectorization of the underlying matrix followed by the inverse matricization.

The next lemma gives the proximal mapping for the ε-sparsity index.

Lemma 5.2 (Proposition 2.2 in [35]). *Let $Y\in\mathbb{R}^{m\times n}$ be given, and $\lambda>0$ and $\varepsilon>0$ be two given parameters. If $Y=U\Sigma(Y)V^\top$ is the singular value decomposition of Y with $\Sigma(Y)=\mathrm{diag}(\sigma(Y))$ and $r=\mathrm{rank}(Y)$, then we have*

$$\mathrm{prox}_\lambda(\|\cdot\|_{*,\varepsilon})(Y)=\mathbb{S}_{\lambda,\varepsilon}(Y):=U\,\mathrm{S}_{\lambda,\varepsilon}(\sigma(Y))V^\top, \tag{5.2}$$

where

$$\mathrm{S}_{\lambda,\varepsilon}(\sigma(Y)):=\mathrm{diag}(S_{\lambda,\varepsilon}(\sigma_1(Y)),S_{\lambda,\varepsilon}(\sigma_2(Y)),\dots,S_{\lambda,\varepsilon}(\sigma_r(Y))),$$

and $S_{\lambda,\varepsilon}(\cdot)$ is the singular value threshold operator, defined as

$$S_{\lambda,\varepsilon}(\eta):=\begin{cases}\mathrm{r}(\eta,\lambda,\varepsilon) & \text{if } \eta>\frac{\lambda}{\varepsilon},\\ \arg\min\{h_\eta(0),h_\eta(\mathrm{r}(\eta,\lambda,\varepsilon))\} & \text{if } \varepsilon<\sqrt[3]{2\lambda\varepsilon}<\frac{2}{3}(\eta+\varepsilon) \text{ and } \eta\le\frac{\lambda}{\varepsilon},\\ 0 & \text{otherwise,}\end{cases} \tag{5.3}$$

where $h_\eta(\alpha):=\frac{1}{2}(\alpha-\eta)^2+\lambda\frac{\alpha}{\alpha+\varepsilon}$ and $\mathrm{r}(\eta,\lambda,\varepsilon)\in(0,\eta)$ is the root closest to η of the cubic equation $\alpha^3+(2\varepsilon-\eta)\alpha^2+(\varepsilon^2-2\eta\varepsilon)\alpha+\lambda\varepsilon-\eta\varepsilon^2=0$ in the interval $(0,\eta)$, and the solution in the interval is unique whenever $\eta>\frac{\lambda}{\varepsilon}$.

If $\varepsilon<\min\{\sqrt{\frac{\lambda}{2}},\frac{\lambda}{2\eta}\}$, then we can approximate $S_{\lambda,\varepsilon}(\eta)$ by (cf. [20, 35])

$$S_{\lambda,\varepsilon}(\eta)\approx\begin{cases}0 & \text{if } c_2<0,\\ \frac{c_1+\sqrt{c_2}}{2} & \text{if } c_2\ge0,\end{cases} \tag{5.4}$$

where $c_1:=\eta-\varepsilon$ and $c_2:=(\eta+\varepsilon)^2-2\lambda$.

The next lemma gives the proximal mapping for weighted nuclear norms with ordered (i.e., nonincreasing or nondecreasing) weights.

Lemma 5.3. *Let $Y\in\mathbb{R}^{m\times n}$ be given, $\lambda>0$ be a given parameter, and $\mathbf{w}$ be either a nonincreasing or a nondecreasing nonnegative weight. If $Y=U\Sigma(Y)V^\top$ is the singular value decomposition of Y and $r=\mathrm{rank}(Y)$, then we have*

$$\mathrm{prox}_\lambda(\|\cdot\|_{\mathbf{w},*})(Y)=\mathrm{wshrink}(Y,\mathbf{w},\lambda):=U\,\mathrm{diag}(\max\{\sigma_i(Y)-\lambda w_i,0\}_{1\le i\le r})V^\top.$$

Proof. The nonincreasing weights correspond to convex spectral functions. In this case, the proximal mapping can be directly checked by the optimality via subdifferential calculus. The nondecreasing case follows from [10, Theorem 2]. ∎

5.2. Explicit formulas for PAMM subproblems. In this subsection, we present explicit formulas for solutions of the subproblems in PAMM (cf. Algorithm 4.2) for realizations of the surrogate function $\mathcal{F}$.

Recall that, for given β and Λ, the function Q is expressed as

$$Q(X,U) = \mathcal{F}(Y) + \lambda\|Z\|_1 + \delta_{\mathsf{P}}(X) + \langle \Lambda_1, X - Y\rangle + \langle \Lambda_2, \mathcal{D}(X) - Z\rangle + \frac{\beta}{2}\|X - Y\|^2 + \frac{\beta}{2}\|\mathcal{D}(X) - Z\|^2.$$

Proposition 5.4. *For a penalty parameter β, a multiplier Λ, and an iteration $(X^{(k)}, U^{(k)})$, the solution of (4.15) is given by*

$$U^{(k+1)} = \left(\operatorname{prox}_{\frac{1}{\beta+\rho_1}}(\mathcal{F}) \left(\frac{\Lambda_1 + \beta X^{(k)} + \rho_1 Y^{(k)}}{\beta + \rho_1} \right), \operatorname{shrink}\left(\frac{\Lambda_2 + \beta\mathcal{D}(X^{(k)}) + \rho_1 Z^{(k)}}{\beta + \rho_1}, \frac{\lambda}{\beta + \rho_1} \right) \right).$$

Proof. By separation of the variables, the subproblem (4.15) can be equivalently transformed into the following two independent problems:

$$\begin{aligned} Y^{(k+1)} &\in \arg\min_Y \mathcal{F}(Y) - \langle \Lambda_1, Y\rangle + \frac{\beta}{2}\|X^{(k)} - Y\|^2 + \frac{\rho_1}{2}\|Y - Y^{(k)}\|^2 \\ &= \arg\min_Y \mathcal{F}(Y) + \frac{\beta + \rho_1}{2}\|Y\|^2 - \langle Y, \Lambda_1 + \beta X^{(k)} + \rho_1 Y^{(k)}\rangle \\ &= \arg\min_Y \mathcal{F}(Y) + \frac{\beta + \rho_1}{2}\left\| Y - \frac{\Lambda_1 + \beta X^{(k)} + \rho_1 Y^{(k)}}{\beta + \rho_1} \right\|^2 \end{aligned}$$

and

$$\begin{aligned} Z^{(k+1)} &= \arg\min_Z \lambda\|Z\|_1 - \langle \Lambda_2, Z\rangle + \frac{\beta}{2}\|\mathcal{D}(X^{(k)}) - Z\|^2 + \frac{\rho_1}{2}\|Z - Z^{(k)}\|^2 \\ &= \arg\min_Z \lambda\|Z\|_1 + \frac{\beta + \rho_1}{2}\|Z\|^2 - \langle Z, \Lambda_2 + \beta\mathcal{D}(X^{(k)}) + \rho_1 Z^{(k)}\rangle \\ &= \arg\min_Z \lambda\|Z\|_1 + \frac{\beta + \rho_1}{2}\left\| Z - \frac{\Lambda_2 + \beta\mathcal{D}(X^{(k)}) + \rho_1 Z^{(k)}}{\beta + \rho_1} \right\|^2 \\ &= \arg\min_Z \|Z\|_1 + \frac{\beta + \rho_1}{2\lambda}\left\| Z - \frac{\Lambda_2 + \beta\mathcal{D}(X^{(k)}) + \rho_1 Z^{(k)}}{\beta + \rho_1} \right\|^2. \end{aligned}$$

Consequently, the conclusion follows from (5.1) and Lemma 5.1. ∎

Proposition 5.5. *For a penalty parameter β, a multiplier Λ, and an iteration $(X^{(k)}, U^{(k+1)})$, the solution of (4.16) is given by*

$$X^{(k+1)} = \Pi_{\mathsf{P}}\left(\frac{\mathcal{D}^{\intercal}(\beta Z^{(k+1)} - \Lambda_2) + \rho_2 X^{(k)} + \beta Y^{(k+1)} - \Lambda_1}{2\beta + \rho_2}\right),$$

where Π_{P} (see (3.5) for details) is the projection onto P.

Proof. The subproblem (4.16) can be equivalently transformed into the following problem:

(5.5)

$$\begin{aligned}
X^{(k+1)} &= \arg\min_X \delta_{\mathsf{P}}(X) + \langle \Lambda_1, X\rangle + \langle \mathcal{D}(X), \Lambda_2\rangle + \frac{\beta}{2}\|X - Y^{(k+1)}\|^2 \\
&\quad + \frac{\beta}{2}\|\mathcal{D}(X) - Z^{(k+1)}\|^2 + \frac{\rho_2}{2}\|X - X^{(k)}\|^2 \\
&= \arg\min_X \delta_{\mathsf{P}}(X) - \langle X, \mathcal{D}^{\intercal}(\beta Z^{(k+1)} - \Lambda_2) + \rho_2 X^{(k)} + \beta Y^{(k+1)} - \Lambda_1\rangle + \frac{2\beta + \rho_2}{2}\|X\|^2 \\
&= \arg\min_X \delta_{\mathsf{P}}(X) + \frac{2\beta + \rho_2}{2}\left\|X - \frac{\mathcal{D}^{\intercal}(\beta Z^{(k+1)} - \Lambda_2) + \rho_2 X^{(k)} + \beta Y^{(k+1)} - \Lambda_1}{2\beta + \rho_2}\right\|^2 \\
&= \Pi_{\mathsf{P}}\left(\frac{\mathcal{D}^{\intercal}(\beta Z^{(k+1)} - \Lambda_2) + \rho_2 X^{(k)} + \beta Y^{(k+1)} - \Lambda_1}{2\beta + \rho_2}\right),
\end{aligned}$$

where the second equality follows from the fact that the DCT $\mathcal{D}$ is an orthogonal transformation and hence $\|\mathcal{D}(X)\| = \|X\|$ for any X. ∎

Equipped with Propositions 5.4 and 5.5, Lemmas 5.2 and 5.3, we can implement Algorithm 4.1 through Algorithm 4.2 smoothly.

5.3. Convergence analysis. According to [35, Lemma 2.1], we have that the ε-sparsity index satisfies Assumption 4.5. For weighted nuclear norms, we have the following lemma.

Lemma 5.6. *Let $\mathbf{w}$ be a nonnegative weight vector. For a given matrix $X \in \mathbb{R}^{m\times n}$ with $m \le n$, $\|X\|_{\mathbf{w},*}$ defined as (2.1) is a Lipschitz continuous function with modulus $\sqrt{m}\|\mathbf{w}\|_\infty$ and, hence, each element in $\partial\|X\|_{\mathbf{w},*}$ is uniformly bounded.*

Proof. Suppose the singular values of X and Y are, respectively, $\sigma_i(X)$ and $\sigma_i(Y)$ for $i = 1, \ldots, m$, ordered in a nonincreasing manner. It follows from [24] that

$$\sum_{i=1}^{m}(\sigma_i(X) - \sigma_i(Y))^2 \le \|X - Y\|^2.$$

Thus,

$$\begin{aligned}
|\|X\|_{\mathbf{w},*} - \|Y\|_{\mathbf{w},*}| &= |\sum_{i=1}^{m} w_i(\sigma_i(X) - \sigma_i(Y))| \le \sum_{i=1}^{m} w_i|\sigma_i(X) - \sigma_i(Y)| \\
&\le \|\mathbf{w}\|_\infty \sum_{i=1}^{m}|\sigma_i(X) - \sigma_i(Y)| \le \sqrt{m}\|\mathbf{w}\|_\infty\|X - Y\|.
\end{aligned}$$

The conclusion then follows. ∎

Note that $\mathbf{w}$ is not required to be ordered.

If adaptive weights are chosen, we have to guarantee both Assumptions 4.5 and 4.6. Actually, there are a variety of choices.

Proposition 5.7. *Let $m \le n$ and $c_i : \mathbb{R}^{m\times n} \to \mathbb{R}_+$ be continuous and bounded for all $i = 1,\dots,m$. Then $\mathcal{F}_{\mu(Z)}(Y) := \|Y\|_{\mu(Z),*}$ with*

$$\mu_i(Z) := \frac{c_i(Z)}{\sigma_i(Z)+\varepsilon} \text{ for all } i = 1,\dots,m \tag{5.6}$$

satisfies Assumptions 4.5 and 4.6.

Proof. The continuity (and hence existence of accumulation points) and boundedness follow from Lemma 5.6, the continuity and the boundedness of c_i's. In the following, we show upper semicontinuity. For two different Z_1, Z_2, we have

$$f(Y) := \mathcal{F}_{\mu(Z_1)}(Y) - \mathcal{F}_{\mu(Z_2)}(Y) = \sum_{i=1}^{m} \left(\frac{c_i(Z_1)}{\sigma_i(Z_1)+\varepsilon} - \frac{c_i(Z_2)}{\sigma_i(Z_2)+\varepsilon} \right) \sigma_i(Y). \tag{5.7}$$

A similar argument as Lemma 5.6 shows that f is Lipschitz continuous with modulus

$$l_{Z_1,Z_2} := \sqrt{m} \max_{1\le i\le m} \left| \frac{c_i(Z_1)}{\sigma_i(Z_1)+\varepsilon} - \frac{c_i(Z_2)}{\sigma_i(Z_2)+\varepsilon} \right|.$$

By continuity, it is easy to show that this quantity approaches to zero as $Z_1 - Z_2 \to 0$. Meanwhile, the subdifferential of f is bounded by l_{Z_1,Z_2}.

On the other hand, $\mathcal{F}_{\mu(Z_1)}(Y) = \mathcal{F}_{\mu(Z_2)}(Y) + f(Y)$, which implies (cf. [41])

$$\partial \mathcal{F}_{\mu(Z_1)}(Y) \subseteq \partial \mathcal{F}_{\mu(Z_2)}(Y) + \partial f(Y).$$

Consequently, the upper semicontinuity then follows [11]. ∎

We see from (4.13) and (5.7) that when c_i is bounded by a number with consistency to those in (4.13), (4.9) can be fulfilled by (5.6) as well.

Therefore, by using an ε-sparsity index, or weighted nuclear norms, or the adaptive weights in (5.6) as surrogates for the rank function, we can obtain convergence results as demonstrated in Theorem 4.7 without any assumption.

Theorem 5.8. *Let $\{X^{(k)}, U^{(k)}, \Lambda^{(k)}\}$ be a sequence generated by Algorithm 4.1 with $\lambda > 0$ and $\mathcal{F}_\mu$ being either the ε-sparsity index or the weighted nuclear norms or the weighted nuclear norms by the adaptive weights in (5.6). Then, it is bounded and any accumulation point of $\{X^{(k)}, U^{(k)}\}$ is a KKT point of problem (4.5).*

6. Convex spectral functions. In this section, we consider convex surrogate $\mathcal{F}$. The problem

$$\begin{aligned} \min \quad & \mathcal{F}(Y) + \lambda\|Z\|_1 + \delta_{\mathsf{P}}(X) \\ \text{s.t.} \quad & X = Y,\ \mathcal{D}(X) = Z, \end{aligned} \tag{6.1}$$

is a convex problem with separable structures.

Algorithm 6.1. Proximal ADMM.

Initialization: Choose $(X^{(0)}, U^{(0)}, \Lambda^{(0)})$, penalty parameter $\beta > 0$, step length $\kappa \in (0, (1+\sqrt{5})/2)$, and proximal parameters $\rho_1, \rho_2 \geq 0$. Set $k := 0$.

Step 1: Solve the U-subproblem

$$U^{(k+1)} := \arg\min_{U} L_\beta(X^{(k)}, U; \Lambda^{(k)}) + \frac{\rho_1}{2}\|U - U^{(k)}\|^2. \tag{6.2}$$

Step 2: Solve the X-subproblem

$$X^{(k+1)} := \arg\min_{X} L_\beta(X, U^{(k+1)}; \Lambda^{(k)}) + \frac{\rho_2}{2}\|X - X^{(k)}\|^2. \tag{6.3}$$

Step 3: Update the multiplier as

$$\Lambda_1^{(k+1)} := \Lambda_1^{(k)} + \kappa\beta(X^{(k+1)} - Y^{(k+1)}) \text{ and } \Lambda_2^{(k+1)} := \Lambda_2^{(k)} + \kappa\beta(\mathcal{D}(X^{(k+1)}) - Z^{(k+1)}). \tag{6.4}$$

Step 4: Unless a termination criterion is fulfilled, set $k := k+1$ and go to Step 1.

6.1. Algorithm. In this case, we can simplify Algorithms 4.1 and 4.2 to a proximal ADMM. Actually, algorithmically, it can be interpreted as a simplified ALM of Algorithm 4.1 with the subproblem (4.8) being solved by only one iteration of Algorithm 4.2, and keeping the penalty parameter fixed.

A commonly applied proximal ADMM for solving (6.1) is given as follows in Algorithm 6.1.

When $\rho_1 = 0$ and $\rho_2 = 0$, the proximal ADMM reduces to the classical ADMM introduced by Gabay and Mercier [15] and Glowinski and Marroco [18].

We note that the solution for each subproblem in Algorithm 6.1 can usually be obtained in a relative efficient way. The subproblem (6.3) is identical to (5.5), and can therefore be solved by Proposition 5.5. The subproblem (6.2) depends on the function $\mathcal{F}$. If the proximal mapping of $\mathcal{F}$ can be computed efficiently, then this subproblem can be solved as well. For example, if $\mathcal{F} = \|\cdot\|_*$ is the nuclear norm, then (6.2) with Lemma 5.3 gives an explicit formula, since the weight $\mathbf{w} = \mathbf{e}$ is valid for Lemma 5.3.

6.2. Convergence analysis. In this subsection, we will show that the sequence generated by Algorithm 6.1 is convergent. The convergence follows from [14, Theorem B.1]; the point here is that all hypotheses of that theorem will be verified, in what follows, to be valid for our problem (6.1) without any assumption.

The linear constraints in (6.1) can be reformulated as the following form,

$$\mathcal{A}(U) + \mathcal{L}(X) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{6.5}$$

where $\mathcal{A}:\mathbb{R}^{3m\times 3n}\times\mathbb{R}^{3m\times 3n}\to\mathbb{R}^{6m\times 3n}$ is defined by $\mathcal{A}(U):=\left[\begin{smallmatrix}-Y\\-Z\end{smallmatrix}\right]$, and $\mathcal{L}:\mathbb{R}^{3m\times 3n}\to\mathbb{R}^{6m\times 3n}$ is defined by $\mathcal{L}(X):=\left[\begin{smallmatrix}X\\\mathcal{D}(X)\end{smallmatrix}\right]$. We denote the set defined by (6.5) as C. The objective function in (6.1) can be decomposed as $f+g$ with $f(X):=\delta_{\mathsf{P}}(X)$ and $g(U):=\mathcal{F}(Y)+\lambda\|Z\|_1$.

First, we show that $\mathcal{A}^*\mathcal{A}$ and $\mathcal{L}^*\mathcal{L}$ are positive definite.

Lemma 6.1. *Both $\mathcal{A}^*\mathcal{A}$ and $\mathcal{L}^*\mathcal{L}$ are positive definite.*

Proof. From the definition of $\mathcal{L}$, we have $\mathcal{L}^*\mathcal{L}(X)=2X$. Thus, for any nonzero matrix $X\in\mathbb{R}^{3m\times 3n}$, we have $\langle X,\mathcal{L}^*\mathcal{L}(X)\rangle=2\langle X,X\rangle>0$, and then $\mathcal{L}^*\mathcal{L}$ is positive definite. The case for $\mathcal{A}^*\mathcal{A}$ is similar. ∎

Second, we show that the optimal solution set of (6.1) is nonempty and compact under reasonable hypotheses.

Lemma 6.2. *Suppose that $\lambda>0$ and $\mathcal{F}$ is bounded from below. Then the optimal solution set of (6.1) is nonempty and compact.*

Proof. Recall that $\mathsf{C}=\{(X,U)|X=Y,\mathcal{D}(X)=Z\}$. Then problem (6.1) can be rewritten as follows:

$$\min h(X,U):=\mathcal{F}(Y)+\lambda\|Z\|_1+\delta_{\mathsf{P}}(X)+\delta_{\mathsf{C}}(X,U).$$

It is clear that h is proper and lower semicontinuous. Then, according to [41, Theorem 1.9], it is sufficient to show that $h(X,U)$ is level-bounded, i.e., $h(X,U)\to\infty$ as $\|X\|_F+\|Y\|_F+\|Z\|_F\to\infty$. We prove it by contradiction. Suppose that there exists a sequence $\{X^{(k)},U^{(k)}\}$ such that $\|X^{(k)}\|_F+\|Y^{(k)}\|_F+\|Z^{(k)}\|_F\to\infty$ while $\{h(X^{(k)},U^{(k)})\}$ is bounded. Then we have that $(X^{(k)},U^{(k)})\in\mathsf{C}$, which implies that all of $\{X^{(k)}\}$, $\{Y^{(k)}\}$, and $\{Z^{(k)}\}$ are unbounded, while, the boundedness of $\{h(X^{(k)},U^{(k)})\}$ and the lower boundedness of $\{\mathcal{F}(Y^{(k)})\}$ imply that $\{\|Z^{(k)}\|_1\}$ is bounded, which is a contradiction. Thus, the optimal solution set of (6.1) is nonempty and compact. ∎

Last, we will show that $\mathrm{ri}(\mathrm{dom}(f)\times\mathrm{dom}(g))\cap\mathsf{C}\neq\emptyset$. Recall that $\mathcal{F}$ has domain $\mathbb{R}^{3m\times 3n}$ and, hence, $\mathrm{dom}(g)=\mathbb{R}^{3m\times 3n}\times\mathbb{R}^{3m\times 3n}=\mathrm{ri}(\mathrm{dom}(g))$, and $\mathrm{dom}(f)=\mathsf{P}$ and has a nonempty relative interior whenever $\mathsf{P}\neq\emptyset$ [40].

Lemma 6.3. *There exists $(X_0,Y_0,Z_0)\in ri(\mathrm{dom}(f)\times\mathrm{dom}(g))\cap\mathsf{C}$.*

Proof. Let $X_0\in\mathrm{ri}(\mathrm{dom}(f))$. Let $Y_0=X_0$ and $Z_0=\mathcal{D}(X_0)$, then we have

$$(X_0,Y_0,Z_0)\in(\mathrm{ri}(\mathrm{dom}(f))\times\mathbb{R}^{3m\times 3n}\times\mathbb{R}^{3m\times 3n})\cap\mathsf{C}.$$

This completes the proof. ∎

The problem (6.1) is composed of a convex objective function and linear equality constraints, thus the convergence result of Algorithm 6.1 can be derived from [14, Theorem B.1]. We summarize this in the following.

Theorem 6.4. *Suppose that $\lambda>0$ and $\mathcal{F}$ is bounded from below. Let $\{X^{(k)},U^{(k)},\Lambda^{(k)}\}$ be a sequence generated by Algorithm 6.1. Then $\{X^{(k)},U^{(k)}\}$ converges to an optimal solution to (6.1) and $\{\Lambda^{(k)}\}$ converges to an optimal solution to the dual problem of (6.1).*

Proof. By Lemmas 6.1, 6.2, and 6.3, the conclusion then follows from [14, Theorem B.1]. ∎

The lower boundedness of $\mathcal{F}$ is satisfied for a large variety of convex surrogates, such as weighted nuclear norms with nonincreasing weights. In particular, it includes the nuclear norm, the Ky-Fan norms, etc.

7. Numerical experiments. In this section, for numerical illustration, we implement (4.5) via four choices of $\mathcal{F}_\mu$: the ε-sparsity index (denoted as TCSO-SI), the weighted nuclear norm by a nondecreasing weight (denoted as TCSO-WNN), an adaptive strategy with nondecreasing weights (denoted as TCSO-aWNN), and the nuclear norm (denoted as TCSO-NN). Note that the former three are nonconvex and the last one is convex. We compare the proposed methods with two groups of state-of-the-art methods. One group of three classical unfolding methods for tensors, includes SiLRTC-TT [6] of mode-n canonical unfolding, TRNNM [54] of circular unfolding, and TNN [55] of tensor T-product unfolding. The other group consists of methods with total variation (TV) regularization, including LRTC-TV-I [31] which is related to HaLRTC [34] (for mode-n unfolding), and TNN-TV [26] which is related to TNN.

The numerical tests are performed on a standard set of color images: airplane, baboon, barbara, sailboat, as collected in Figure 1.

All numerical experiments are implemented in MATLAB R2019b on a Lenovo PC with i7-5500U CPU and RAM 8 GB in 64 bit Windows operating system. The quality of recovered images is measured by the peak signal-to-noise ratio (PSNR), the relative square error (RSE) and the structural similarity index (SSIM). Given a tensor $\mathcal{X} \in \mathbb{R}^{n_1\times n_2\times n_3}$, PSNR and RSE are defined as

$$\mathrm{PSNR} := 10\log_{10}\frac{n_1n_2n_3\|\mathcal{X}_{\mathrm{true}}\|_\infty^2}{\|\mathcal{X}_{\mathrm{true}}-\mathcal{X}_{\mathrm{rec}}\|^2} \quad \text{and } \mathrm{RSE} := \frac{\|\mathcal{X}_{\mathrm{true}}-\mathcal{X}_{\mathrm{rec}}\|}{\|\mathcal{X}_{\mathrm{true}}\|},$$

where $\mathcal{X}_{\mathrm{true}}$ is the ground truth tensor, $\mathcal{X}_{\mathrm{rec}}$ is the recovered tensor, $\|\mathcal{X}_{\mathrm{true}}\|_\infty$ denotes the infinity norm of $\mathcal{X}_{\mathrm{true}}$. We use $\mathsf{ssim}(\mathcal{X}_{\mathsf{true}}, \mathcal{X}_{\mathsf{rec}})$ in MATLAB to calculate SSIM. The supreme values of PSNR, SSIM, and RSE are infinity, 1, and 0, respectively. The higher the PSNR/SSIM values and the lower the RSE values, the better the quality of the recovered images. The sampling ratio (SR) is defined as

$$\mathrm{SR} := \frac{|\Omega|}{n_1n_2n_3},$$

where Ω is generated at random and $|\Omega|$ represents the cardinality of the set Ω.

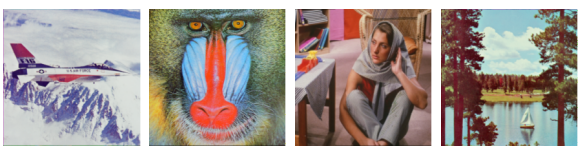


Figure 1. *The set of test images in the numerical experiments.*

Both Algorithms 4.1 and 6.1 are terminated either when the maximum iteration exceeds 150 or the following stopping criterion is met:

$$\frac{\|\mathcal{X}^{(k+1)} - \mathcal{X}^{(k)}\|}{\|\mathcal{X}^{(k)}\|} < 10^{-4}.$$

Algorithm 4.2 is terminated either when the maximum iteration exceeds 1 for high SRs and exceeds 5 for low SRs or the following stopping criterion is met:

$$\|\Theta^{(k)}\| \le \epsilon^{(k)},$$

where $\epsilon^{(k)} := \max\{10^{-1}, 0.998^k\}$.

7.1. Parameter setting. In this subsection, we present the performance dependence on the parameters in Algorithms 4.1, 4.2 and 6.1. The complete landscape would be rather complicated, so we give some illustration on a tip of this iceberg.

For TCSO-WNN, we set

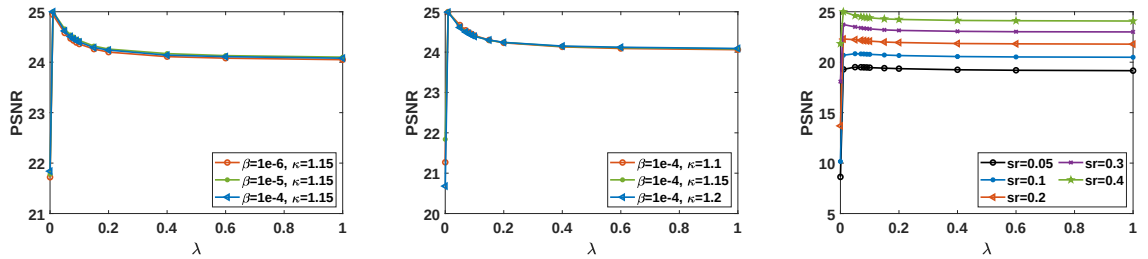
$$w_i := \frac{1}{\sigma_i(Y^{(0)})} \text{ for } i = 1, \ldots, r \text{ and } w_i := \frac{1}{\sigma_r(Y^{(0)})} \text{ for } i = r+1, \ldots, \min\{m, n\},$$

where $r = \operatorname{rank}(Y^{(0)})$. For TCSO-aWNN, $\mathcal{F}_\mu$ is chosen as the weighted nuclear norm with adaptive weights [20], i.e., the parameter $\mu^{(k)}$ indicates the adaptive weights as follows:

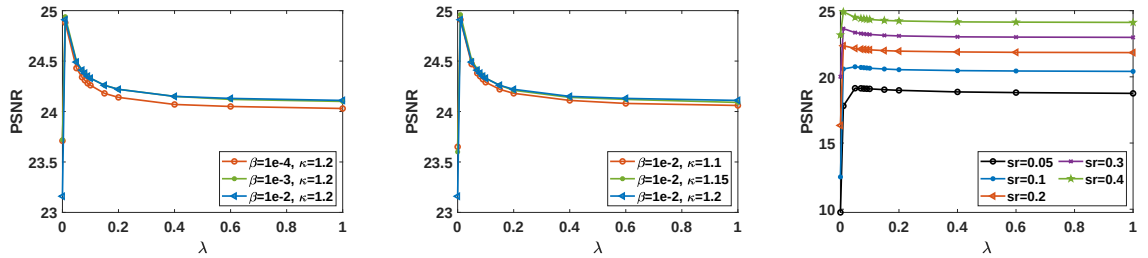
$$\mu_i^{(k)} := \frac{1}{\sigma_i(Y^{(k)}) + \varepsilon} \text{ for } i = 1, \ldots, \min\{m, n\}.$$

First, we show the consequences of the penalty parameter β, the penalty adjustment parameter κ, and the regularization parameter λ on baboon with SR $= 0.4$. From Figure 2, it is observed that different choices of β and κ give similar performances, while the performance varies significantly along the regularization parameter in Algorithm 4.1. We set $\beta = 10^{-4}$ and $\kappa = 1.15$ for TCSO-SI, and $\beta = 10^{-2}$ and $\kappa = 1.2$ for TCSO-WNN and TCSO-aWNN. The experimental optimal values of λ are around 0.05, 0.01, and 0.1 for TCSO-SI, TCSO-WNN, and TCSO-aWNN, respectively. Moreover, for different SRs, the performances of TCSO-SI and TCSO-aWNN are similar as in Figures 2(c) and 2(i). Therefore, we set $\lambda = 0.05$ for TCSO-SI and $\lambda = 0.1$ for TCSO-aWNN in our experiments. While, for TCSO-WNN, it is more subtle; see Figure 2(f). Then we set $\lambda = 0.1$ for SR $= \{0.05, 0.1\}$ and set $\lambda = 0.01$ for SR $= \{0.2, 0.3, 0.4\}$ in experiments for random missing, and set $\lambda = 0.1$ in experiments for structural missing.

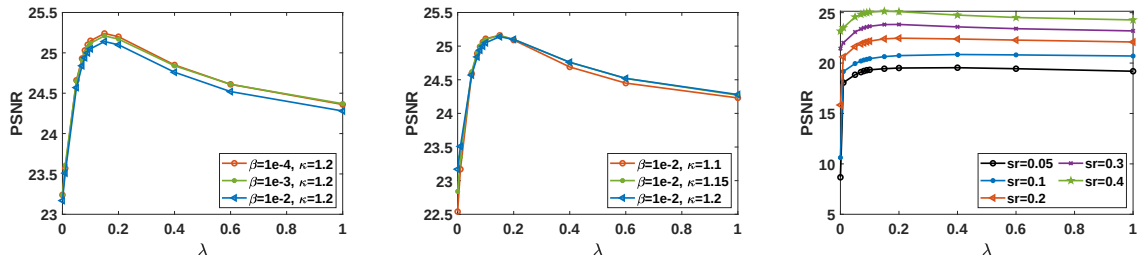
For Algorithm 6.1, the theoretical range of κ is $(0, (\sqrt{5}+1)/2)$, and we set $\kappa = 1.15$ for TCSO-NN. As suggested in [9] for matrix completion problem, we set the penalty parameter $\beta = C/\sqrt{9mn}$ in the two block proximal ADMM. In our experiments, we find that when $C \in [10, 80]$, it gives the best performance. Thus, we set $\beta = 30/\sqrt{9mn}$ for TCSO-NN. The PSNR values of TCSO-NN with different regularization parameters λ are shown in Figure 3(a), from which we see that PSNR values reach their peak when λ is around the interval $[0.1, 0.2]$. So we set $\lambda = 0.1$ for TCSO-NN in experiments for random missing and set $\lambda = 0.2$ in experiments for structural missing.



(a) (TCSO-SI, β) (b) (TCSO-SI, κ) (c) (TCSO-SI, SR)

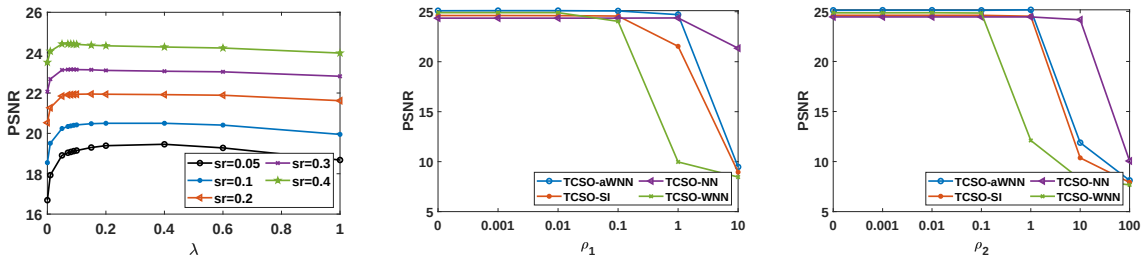


(d) (TCSO-WNN, β) (e) (TCSO-WNN, κ) (f) (TCSO-WNN, SR)



(g) (TCSO-aWNN, β) (h) (TCSO-aWNN, κ) (i) (TCSO-aWNN, SR)

Figure 2. *Left and middle: PSNR values of TCSO-SI, TCSO-WNN, and TCSO-aWNN with different β's and κ's on baboon with $SR = 0.4$. Right: PSNR values of different SRs.*



(a) (TCSO-NN, λ) (b) (PSNR, ρ_1) (c) (PSNR, ρ_2)

Figure 3. *Left: PSNR values of TCSO-NN with different λ's and SRs. Middle and right: PSNR values of TCSO-SI, TCSO-WNN, TCSO-aWNN, and TCSO-NN with different ρ_i's on baboon with $SR = 0.4$.*

Second, we test the influence of the proximal parameters ρ_1 and ρ_2 in Algorithm 4.2 on baboon with the same SR = 0.4. From Figures 3(b) and 3(c), we can observe that PSNR

Table 1
PSNR (dB), SSIM, and RSE obtained by TCSO-SI and TCSO-aWNN on baboon with different ε's and SRs.

	SR	0.05		0.2		0.4	
Method	ε-value	PSNR/SSIM	RSE	PSNR/SSIM	RSE	PSNR/SSIM	RSE
TCSO-SI	$\varepsilon = 10^{-1}$	17.94/0.5191	0.2348	22.27/0.7552	0.1428	24.93/0.8675	0.1050
	$\varepsilon = 10^{-2}$	17.75/0.5126	0.2402	22.21/0.7541	0.1438	24.89/0.8663	0.1055
	$\varepsilon = 10^{-3}$	17.87/0.5235	0.2367	22.21/0.7517	0.1437	24.91/0.8674	0.1053
	$\varepsilon = 10^{-4}$	17.80/0.5207	0.2386	22.24/0.7548	0.1431	24.94/0.8684	0.1049
	$\varepsilon = 10^{-5}$	17.73/0.5168	0.2406	22.22/0.7514	0.1436	24.89/0.8671	0.1055
	$\varepsilon = 10^{-6}$	17.90/0.5264	0.2360	22.23/0.7514	0.1434	24.94/0.8682	0.1049
TCSO-aWNN	$\varepsilon = 10^{-1}$	18.63/0.5463	0.2170	22.20/0.7575	0.1439	25.07/0.8728	0.1034
	$\varepsilon = 10^{-2}$	16.93/0.4661	0.2638	19.45/0.6383	0.1975	23.78/0.8382	0.1199
	$\varepsilon = 10^{-3}$	16.98/0.4671	0.2624	19.47/0.6374	0.1970	23.84/0.8418	0.1191
	$\varepsilon = 10^{-4}$	19.40/0.5820	0.1985	22.27/0.7585	0.1427	24.83/0.8657	0.1063
	$\varepsilon = 10^{-5}$	19.64/0.5985	0.1931	22.39/0.7595	0.1408	24.81/0.8637	0.1065
	$\varepsilon = 10^{-6}$	19.62/0.5941	0.1936	22.35/0.7585	0.1414	24.87/0.8658	0.1058

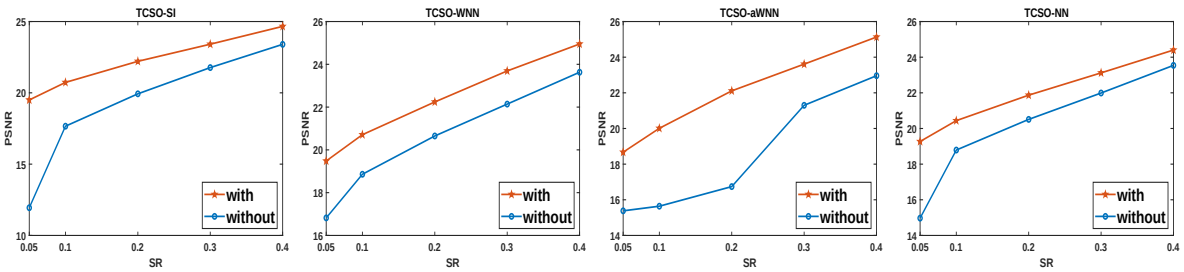


Figure 4. *Numerical performance with and without regularization.*

values are stable when the proximal parameters $\rho_1 \leq 0.1$ and $\rho_2 \leq 0.1$, i.e., the performance of TCSO-SI, TCSO-WNN, TCSO-aWNN, and TCSO-NN are robust for $\rho_1, \rho_2 \leq 0.1$. Thus, we set the proximal parameters $\rho_1 = 0.001$ and $\rho_2 = 0.001$ for TCSO-SI, TCSO-WNN, TCSO-aWNN, and TCSO-NN in all experiments.

In the next, we test the influence of different ε's in TCSO-SI and TCSO-aWNN. The corresponding numerical results are presented in Table 1. According to Table 1, the PSNR and SSIM values of TCSO-WNN and TCSO-aWNN reach their peaks when the ε's are, respectively, around 0.1 and 10^{-5}, and we set $\varepsilon = 0.1$ for both.

At the end of this subsection, we show the effectiveness of the regularization term. We test baboon on SR=$\{0.05, 0.1, 0.2, 0.3, 0.4\}$. Figure 4 shows the improvements of the regularization.

7.2. Color image for random missing. In this subsection, we randomly sample the images with size $256 \times 256 \times 3$ to get the set Ω with the SR in $\{0.05, 0.1, 0.2, 0.3, 0.4\}$. Elaborate numerical tests are carried out, while only a small sample is presented here for the sake of the paper length. Table 2 lists the numerical results of the different methods, and the best PSNR, RSE, and SSIM values are shown in bold font for easy identification. Data with underlines refer to suboptimal performance.

It can be seen from Table 2 that the proposed methods are recommendable compared with the state-of-the-art methods without regularization (i.e., TNN, SiLRTC-TT, TRNNM)

Table 2

PSNR (dB), RSE, SSIM, and Time (seconds) obtained by 9 algorithms for color images with different SRs.

Image	SR	0.2		0.3		0.4	
	Method	PSNR/RSE/SSIM	Time	PSNR/RSE/SSIM	Time	PSNR/RSE/SSIM	Time
airplane	TNN	23.14/0.0951/0.6719	18.22	25.77/0.0722/0.7842	14.05	27.72/0.0555/0.8592	17.44
	SiLRTC-TT	22.93/0.0949/0.6313	53.97	24.37/0.0752/0.7512	52.67	26.40/0.0596/0.8326	39.96
	TRNNM	23.64/0.0818/0.8392	66.14	26.08/0.0618/0.9036	72.64	28.46/0.0466/0.9409	83.75
	LRTC-TV-I	23.68/0.0774/0.8558	52.11	26.30/0.0602/0.9117	57.55	28.48/0.0467/0.9427	67.73
	TNN-TV	24.74/0.0720/0.8749	88.91	26.94/0.0559/0.9217	99.36	29.21/0.0437/0.9459	89.20
	TCSO-NN(our)	24.73/0.0721/0.8289	141.08	26.77/0.0570/0.8804	183.41	28.97/0.0443/0.9189	171.11
	TCSO-SI(our)	**25.70/0.0645**/0.8372	319.84	**27.69/0.0513**/0.8914	235.73	**29.92/0.0397**/0.9290	260.31
	TCSO-WNN(our)	**25.32/0.0674**/0.8437	87.89	**27.44/0.0528**/0.8957	101.23	**29.69/0.0408**/0.9295	82.56
	TCSO-aWNN(our)	**26.32/0.0600**/0.8523	268.19	**28.71/0.0456**/0.9082	137.42	**31.15/0.0349**/0.9404	131.05
baboon	TNN	20.46/0.1862/0.6541	14.56	22.12/0.1552/0.7494	15.52	23.99/0.1286/0.8245	14.97
	SiLRTC-TT	20.67/0.1716/0.6641	65.36	22.19/0.1441/0.7586	61.30	23.62/0.1222/0.8261	44.69
	TRNNM	21.17/0.1618/0.7173	88.66	22.70/0.1358/0.7974	81.45	24.21/0.1142/0.8565	81.47
	LRTC-TV-I	21.95/0.1481/0.7346	67.83	23.19/0.1284/0.7992	66.61	24.31/0.1129/0.8467	65.81
	TNN-TV	22.06/0.1462/0.7386	92.73	23.33/0.1263/0.8058	100.94	24.52/0.1101/0.8548	113.36
	TCSO-NN(our)	21.90/0.1487/0.7352	158.19	23.09/0.1299/0.7988	175.11	24.33/0.1126/0.8496	179.38
	TCSO-SI(our)	**22.25/0.1430/0.7527**	362.13	**23.36/0.1258/0.8098**	256.97	**24.57/0.1095/0.8568**	264.17
	TCSO-WNN(our)	**22.27/0.1427/0.7554**	95.64	**23.58/0.1228/0.8180**	111.72	**24.86/0.1059/0.8664**	95.80
	TCSO-aWNN(our)	**22.17/0.1444/0.7530**	258.19	**23.62/0.1221/0.8220**	136.05	**25.06/0.1035/0.8728**	137.89
barbara	TNN	22.58/0.1513/0.7633	16.39	25.34/0.1104/0.8544	12.75	27.97/0.0819/0.9121	15.80
	SiLRTC-TT	23.38/0.1388/0.7959	63.64	25.63/0.1072/0.8631	50.33	27.57/0.0857/0.9082	39.78
	TRNNM	24.25/0.1257/0.8462	85.84	26.71/0.0947/0.9028	96.03	28.97/0.0730/0.9397	90.02
	LRTC-TV-I	25.09/0.1141/0.8608	60.22	27.27/0.0888/0.9041	64.36	28.96/0.0731/0.9305	66.56
	TNN-TV	25.77/0.1055/0.8735	85.81	27.92/0.0823/0.9135	92.91	29.72/0.0669/0.9388	92.39
	TCSO-NN(our)	25.62/0.1074/0.8641	160.70	27.81/0.0834/0.9073	172.13	29.63/0.0676/0.9362	189.38
	TCSO-SI(our)	**26.57/0.0961/0.8744**	362.83	**28.85/0.0740/0.9182**	253.22	**30.86/0.0587/0.9468**	245.39
	TCSO-WNN(our)	**26.39/0.0982/0.8774**	120.53	**28.57/0.0764/0.9168**	89.50	**30.57/0.0607/0.9453**	96.30
	TCSO-aWNN(our)	**26.86/0.0930/0.8779**	270.31	**29.28/0.0704/0.9208**	127.33	**31.51/0.0544/0.9501**	129.95
sailboat	TNN	21.47/0.1600/0.7305	13.95	23.54/0.1256/0.8266	14.91	25.28/0.0996/0.8910	14.97
	SiLRTC-TT	21.17/0.1533/0.7821	59.67	23.23/0.1209/0.8638	44.88	25.14/0.0970/0.9123	36.66
	TRNNM	21.92/0.1406/0.8353	82.09	24.02/0.1104/0.8963	91.81	26.05/0.0874/0.9350	90.44
	LRTC-TV-I	22.69/0.1287/0.8710	54.42	24.73/0.1017/0.9167	64.08	26.44/0.0836/0.9441	67.00
	TNN-TV	23.10/0.1228/0.8769	88.45	25.22/0.0961/0.9233	96.50	26.82/0.0800/0.9473	88.00
	TCSO-NN(our)	**23.16/0.1218**/0.8625	163.03	25.10/0.0975/0.9087	168.22	26.77/0.0804/0.9375	188.80
	TCSO-SI(our)	**23.76/0.1137/0.8792**	323.13	**25.70/0.0909**/0.9195	240.53	**27.39/0.0749**/0.9449	236.59
	TCSO-WNN(our)	**23.62/0.1156/0.8801**	89.48	**25.71/0.0908/0.9239**	90.89	**27.52/0.0738/0.9488**	93.36
	TCSO-aWNN(our)	**24.35/0.1063/0.8889**	240.42	**26.48/0.0832/0.9304**	125.14	**28.27/0.0677/0.9537**	131.88

in different SRs. They always return recovered images with better quality within affordable time, especially in low sampling. Moreover, the proposed methods are comparable to the regularization methods LRTC-TV-I and TNN-TV. Figure 5 shows the visual results of each method at SRs of 0.05 and 0.1. As can be seen from Figure 5, the proposed methods are visually better than the compared methods. Figure 6 shows the PSNR, RSE, and SSIM curves of unpictured baboon for a clearer visual comparison.

7.3. Color image for structural missing. In this subsection, we present preliminary numerical experiments for color images with three types of structural missing: barbara with a text mask (denoted as barbara-text), sailboat with a stripe mask (denoted as sailboat-stripe), and baboon with a graph mask (denoted as baboon-graph). Figure 7 presents the visual pictures for different methods and Table 3 is the corresponding PSNR, RSE, and SSIM values. From Figure 7 as well as Table 3, it is seen that the proposed methods have a favorite performance in all of the PSNR values, RSE values, and SSIM values, as well as visual results.

7.4. Color image for random missing and structural missing without regularization. Finally, we implement (4.5) via nuclear norm and weighted nuclear norm by a nondecreasing weight without regularization, i.e., $\lambda = 0$ and test baboon and barbara with the SR in

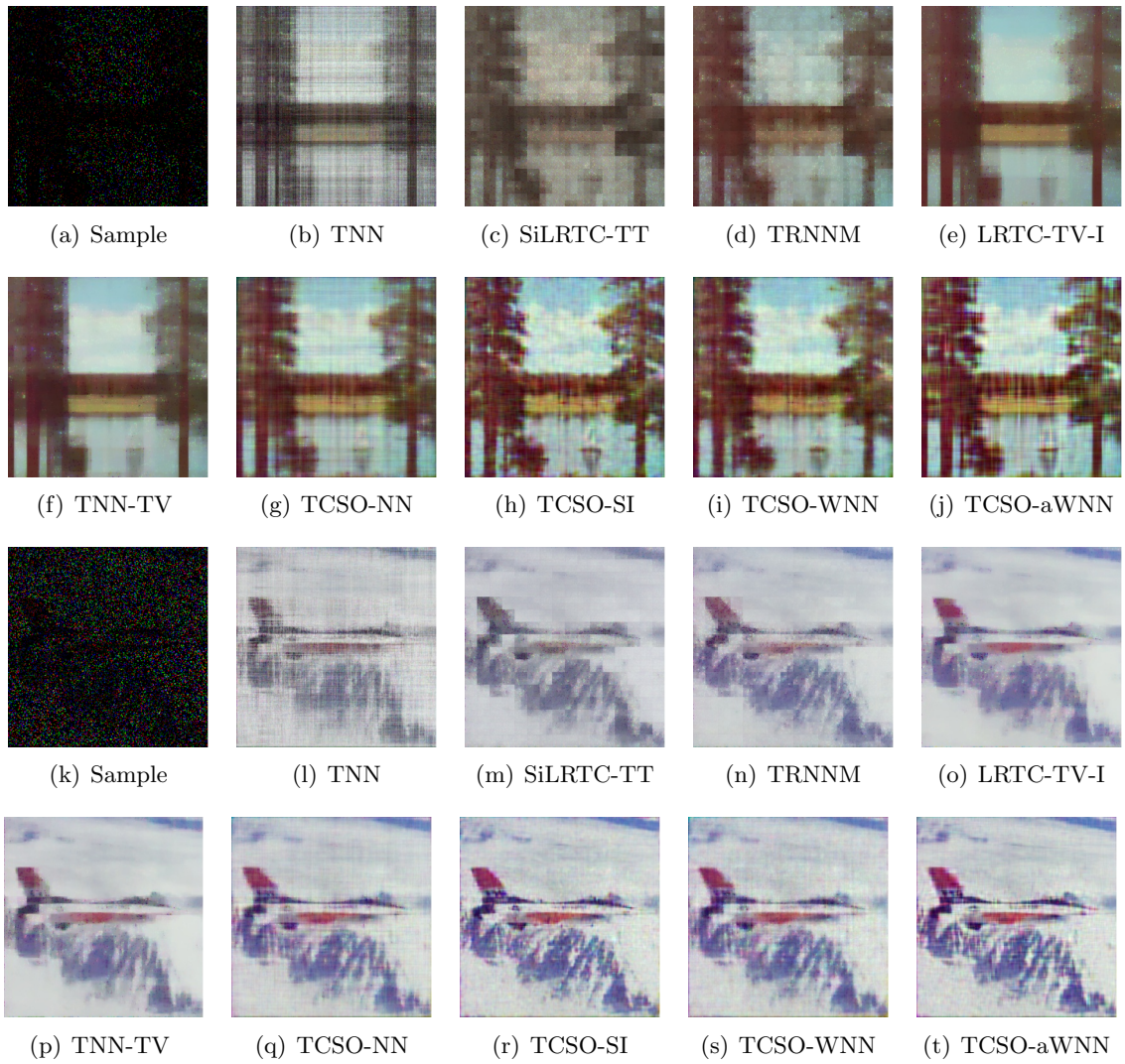


Figure 5. *The top two rows are the visual results at* $SR = 0.05$ *and the bottom two rows are the visual results at* $SR = 0.1$.

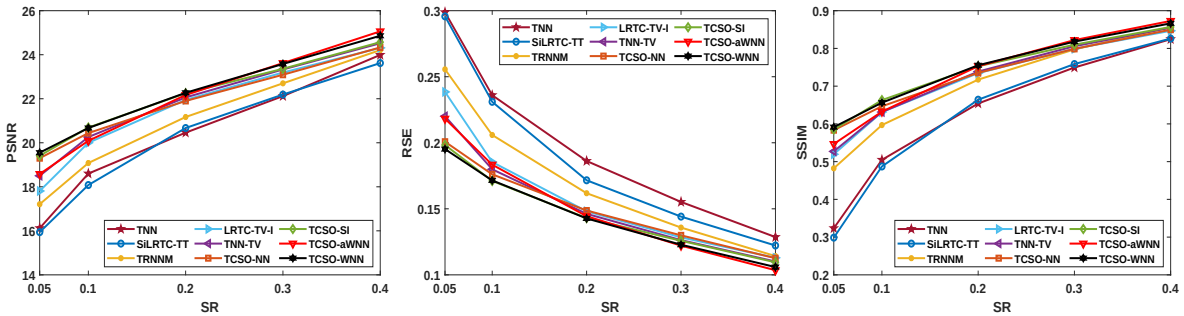
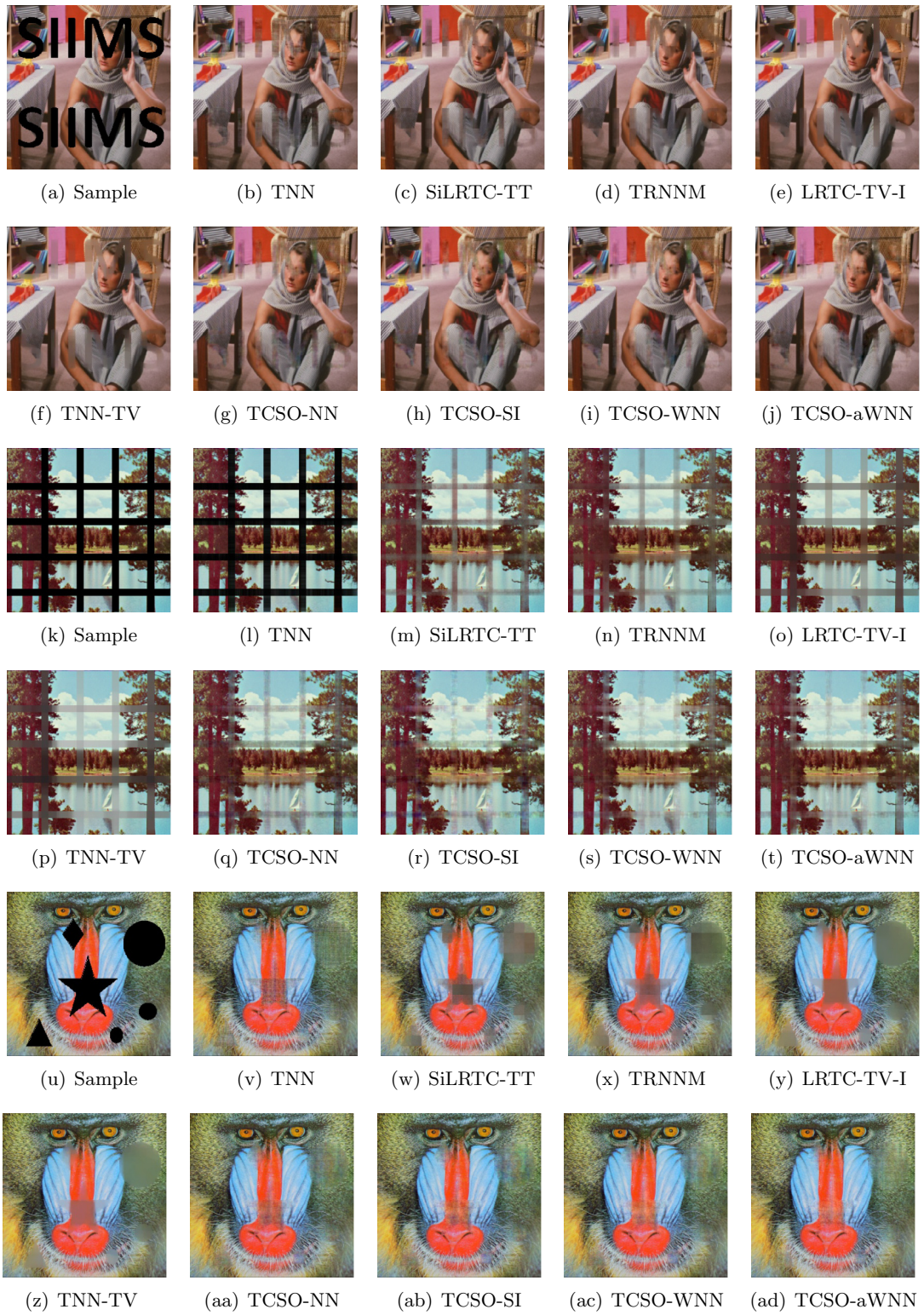


Figure 6. *PSNR, RSE, and SSIM comparisons of baboon.*



(a) Sample (b) TNN (c) SiLRTC-TT (d) TRNNM (e) LRTC-TV-I

(f) TNN-TV (g) TCSO-NN (h) TCSO-SI (i) TCSO-WNN (j) TCSO-aWNN

(k) Sample (l) TNN (m) SiLRTC-TT (n) TRNNM (o) LRTC-TV-I

(p) TNN-TV (q) TCSO-NN (r) TCSO-SI (s) TCSO-WNN (t) TCSO-aWNN

(u) Sample (v) TNN (w) SiLRTC-TT (x) TRNNM (y) LRTC-TV-I

(z) TNN-TV (aa) TCSO-NN (ab) TCSO-SI (ac) TCSO-WNN (ad) TCSO-aWNN

Figure 7. *Images recovered by the 9 algorithms with structural missing information.*

Table 3
PSNR (dB), RSE, and SSIM corresponding to Figure 7.

	Image								
	barbara-text			sailboat-stripe			baboon-graph		
Method	PSNR	RSE	SSIM	PSNR	RSE	SSIM	PSNR	RSE	SSIM
TNN	22.92	0.1465	0.9023	10.63	0.5155	0.6243	24.55	0.1125	0.9202
SiLRTC-TT	24.17	0.1269	0.9096	18.93	0.1983	0.7738	22.84	0.1336	0.9037
TRNNM	24.79	0.1181	0.9237	19.80	0.1794	0.8295	22.89	0.1328	0.9180
LRTC-TV-I	25.32	0.1110	0.9285	19.03	0.1961	0.8108	24.42	0.1114	0.9253
TNN-TV	25.74	0.1058	**0.9308**	20.55	0.1646	0.8109	25.31	0.1006	0.9298
TCSO-NN(our)	**25.78**	**0.1054**	0.9298	**20.57**	**0.1641**	**0.8364**	**25.49**	**0.0985**	0.9292
TCSO-SI(our)	**26.15**	**0.1010**	**0.9308**	**21.05**	**0.1554**	**0.8449**	**25.69**	**0.0962**	0.9289
TCSO-WNN(our)	**26.34**	**0.0988**	**0.9338**	**21.11**	**0.1543**	**0.8606**	**25.97**	**0.0932**	**0.9311**
TCSO-aWNN(our)	**26.58**	**0.0961**	**0.9373**	**21.24**	**0.1519**	**0.8644**	**26.25**	**0.0902**	**0.9320**

Table 4
PSNR (dB), SSIM, and RSE for different SRs and structural missing.

Image	SR	0.3		0.4		1	
Method	PSNR/SSIM	RSE	PSNR/SSIM	RSE	PSNR/SSIM	RSE	
baboon-square	HaLRTC	18.77/0.6882	0.2134	19.26/0.7424	0.2017	21.07/0.8933	0.1638
	TNN	19.94/0.6828	0.2018	20.66/0.7431	0.1862	22.28/0.8944	0.1425
	SiLRTC-TT	18.50/0.6473	0.2203	18.96/0.7054	0.2089	20.17/0.8606	0.1818
	TRNNM	17.43/0.6973	0.2492	17.76/0.7494	0.2398	18.71/0.8742	0.2149
	TCSO-NN(Our)	19.62/**0.7060**	**0.1936**	20.15/**0.7581**	**0.1821**	**22.29/0.9052**	**0.1423**
	TCSO-WNN(Our)	**20.11/0.7101**	**0.1289**	**20.83/0.7631**	**0.1685**	**22.74/0.9059**	**0.1351**
barbara-circle	HaLRTC	16.54/0.7752	0.3052	16.63/0.8135	0.3021	16.96/0.8951	0.2909
	TNN	17.23/0.7750	0.2737	17.58/0.8217	0.2666	17.64/0.8969	0.2549
	SiLRTC-TT	16.73/0.7727	0.2986	16.89/0.8102	0.2932	16.98/0.8890	0.2901
	TRNNM	15.62/0.8078	0.3391	15.73/0.8389	0.3351	15.95/0.8916	0.3269
	TCSO-NN(Our)	**17.70**/0.7857	**0.2672**	**17.78**/0.8243	**0.2648**	**17.97/0.9043**	0.2589
	TCSO-WNN(Our)	**19.65**/0.8015	**0.2133**	**20.00/0.8400**	**0.2049**	**19.82/0.9025**	**0.2092**

$\{0.3, 0.4, 1\}$ and structural missing together, denoted as baboon-square and barbara-circle, respectively. From Figures 2(d) and 2(e), we can observe that TCSO-WNN is sensitive to β and κ when $\lambda = 0$, and TCSO-WNN performs better with small β and κ. Therefore, we set $\beta = 10^{-4}$ and $\kappa = 1.15$ for this case. For TCSO-NN, we adopt an efficient strategy proposed in [33], in which the penalty parameter $\beta^{(k+1)}$ is updated by $\beta^{(k+1)} = \kappa\beta^{(k)}$, and we set $\kappa = 1.15$ and $\beta = 5 \times 10^{-3}$ as the initial values. Illustrative numerical results of HaLRTC, TNN, SiLRTC-TT, TRNNM, TCSO-NN, and TCSO-WNN in different SRs and structural missing information are listed in Table 4 and a visual picture at SR of 0.3 is shown in Figure 8. The numerical results are promising as well.

8. Conclusions. In this paper, a new TC model is proposed, which is different from those in the literature. One ingredient is the Strassen–Ottaviani flattening, which is an intrinsic flattening technique to reveal the tensor rank, and another is an adaptive parameter scheme in the spectral surrogate functions. Integrated by other techniques, such as DCT; the proposed method has favorable performances compared with some state-of-the-art methods in the literature. We propose an inexact ALM method for solving the proposed model, and it

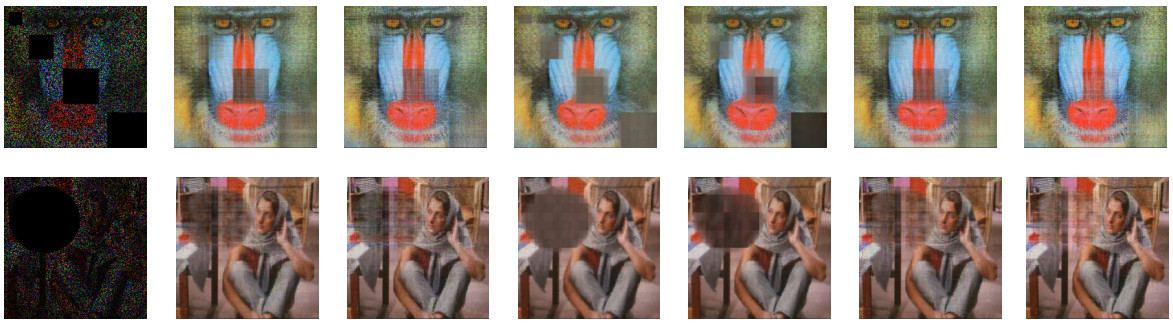


Figure 8. *Recovered images at SR = 0.3. Left to right: sampled, HaLRTC, TNN, SiLRTC, TRNNM, TCSO-NN(our), TCSO-WNN(our).*

can handle adaptive objective functions. The global convergence is established under mild hypotheses, which are shown to be true for a wide class of surrogate functions.

It is worth mentioning that further investigations on more efficient implementations and exact recovery theory for the proposed models can be carried out and are interesting. In fact, it is meaningful to consider an alternative approach to avoid SVD of the involved matrices. For example, we can decompose the matrix into the product of two matrices with smaller dimensions using Burer–Monteiro decompositions [7, 49]. Exact recovery theory based on surrogates other than the nuclear norm, such as weighted nuclear norms, ε-sparsity index, etc., is worth of study as well.

Acknowledgment. We would like to acknowledge the editor and two anonymous referees for their thoughtful suggestions and helpful comments on our manuscript.

REFERENCES

[1] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mørup, *Scalable tensor factorizations for incomplete data*, Chemom. Intell. Lab. Syst., 106 (2011), pp. 41–56.
[2] N. Ahmed, T. Natarajan, and K. R. Rao, *Discrete cosine transform*, IEEE Trans. Comput., 100 (1974), pp. 90–93.
[3] C. A. Andersson and R. Bro, *Improving the speed of multi-way algorithms: Part* I. *Tucker*3, Chemom. Intell. Lab. Syst., 42 (1998), pp. 93–103.
[4] H. Attouch, J. Bolte, and B. F. Svaiter, *Convergence of descent methods for semi-algebraic and tame problems: Proximal algorithms, forward–backward splitting, and regularized Gauss–Seidel methods*, Math. Program., 137 (2013), pp. 91–129.
[5] M. Bai, X. Zhang, G. Ni, and C. Cui, *An adaptive correction approach for tensor completion*, SIAM J. Imaging Sci., 9 (2016), pp. 1298–1323.
[6] J. A. Bengua, H. N. Phien, H. D. Tuan, and M. N. Do, *Efficient tensor completion for color image and video recovery: Low-rank tensor train*, IEEE Trans. Image Process., 26 (2017), pp. 2466–2479.
[7] S. Burer and R. D. Monteiro, *A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization*, Math. Program., 95 (2003), pp. 329–357.
[8] E. J. Candès and B. Recht, *Exact matrix completion via convex optimization*, Found. Comput. Math., 9 (2009), pp. 717–772.
[9] C. Chen, B. He, and X. Yuan, *Matrix completion via an alternating direction method*, IMA J. Numer. Anal., 32 (2012), pp. 227–245.
[10] K. Chen, H. Dong, and K.-S. Chan, *Reduced rank regression via adaptive nuclear norm penalization*, Biometrika, 100 (2013), pp. 901–920.

[11] F. H. Clarke, *Optimization and Nonsmooth Analysis*, Classics Appl. Math. 5, SIAM, Philadelphia, 1990.
[12] M. Ding, T.-Z. Huang, T.-Y. Ji, X.-L. Zhao, and J.-H. Yang, *Low-rank tensor completion using matrix factorization based on tensor train rank and total variation*, J. Sci. Comput., 81 (2019), pp. 941–964.
[13] M. Fazel, *Matrix Rank Minimization with Applications*, Ph.D. thesis, Stanford University, Stanford, CA, 2002.
[14] M. Fazel, T. K. Pong, D. Sun, and P. Tseng, *Hankel matrix rank minimization with applications to system identification and realization*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 946–977.
[15] D. Gabay and B. Mercier, *A dual algorithm for the solution of nonlinear variational problems via finite element approximation*, Comput. Math. Appl., 2 (1976), pp. 17–40.
[16] S. Gandy, B. Recht, and I. Yamada, *Tensor completion and low-n-rank tensor recovery via convex optimization*, Inverse Problems, 27 (2011), 025010.
[17] S. Gao and Q. Fan, *Robust Schatten-p norm based approach for tensor completion*, J. Sci. Comput., 82 (2020), pp. 1–23.
[18] R. Glowinski and A. Marroco, *Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de Dirichlet non linéaires*, Rev. Fr. Autom. Inform. Rech. Opér. Anal Numér., 9 (1975), pp. 41–76.
[19] L. Grasedyck, M. Kluge, and S. Krämer, *Variants of alternating least squares tensor completion in the tensor train format*, SIAM J. Sci. Comput., 37 (2015), pp. A2424–A2450.
[20] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang, *Weighted nuclear norm minimization and its applications to low level vision*, Int. J. Comput. Vis., 121 (2017), pp. 183–208.
[21] S. Gu, L. Zhang, W. Zuo, and X. Feng, *Weighted nuclear norm minimization with application to image denoising*, in IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Piscataway, NJ, 2014, pp. 2862–2869.
[22] J. Håstad, *Tensor rank is NP-complete*, J. Algorithms, 11 (1990), pp. 644–654.
[23] C. J. Hillar and L.-H. Lim, *Most tensor problems are NP-hard*, J. ACM, 60 (2013), pp. 1–39.
[24] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, 2012.
[25] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He, *Fast and accurate matrix completion via truncated nuclear norm regularization*, IEEE Trans. Pattern Anal. Mach. Intell., 35 (2012), pp. 2117–2130.
[26] F. Jiang, X.-Y. Liu, H. Lu, and R. Shen, *Anisotropic total variation regularized low-rank tensor completion based on tensor nuclear norm for color image inpainting*, in IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, Piscataway, NJ, 2018, pp. 1363–1367.
[27] M. E. Kilmer and C. D. Martin, *Factorization strategies for third-order tensors*, Linear Algebra Appl., 435 (2011), pp. 641–658.
[28] J. M. Landsberg, *Tensors: Geometry and Applications*, American Mathematical Society, Providence, RI, 2012.
[29] S. E. Leurgans, R. T. Ross, and R. B. Abel, *A decomposition for three-way arrays*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 1064–1083.
[30] A. S. Lewis and H. S. Sendov, *Nonsmooth analysis of singular values. Part* I*: Theory*, Set-Valued Anal., 13 (2005), pp. 213–241.
[31] X. Li, Y. Ye, and X. Xu, *Low-rank tensor completion with total variation for visual data inpainting*, in Proceedings of the AAAI Conference on Artificial Intelligence, AAAI Press, Palo Alto, CA, 2017, pp. 2210–2216.
[32] Y.-F. Li, K. Shang, and Z.-H. Huang, *Low Tucker rank tensor recovery via ADMM based on exact and inexact iteratively reweighted algorithms*, J. Comput. Appl. Math., 331 (2018), pp. 64–81.
[33] Z. Lin, M. Chen, and Y. Ma, *The Augmented Lagrange Multiplier Method for Exact Recovery of Corrupted Low-Rank Matrices*, preprint, arXiv:1009.5055, 2010.
[34] 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), pp. 208–220.
[35] X. Ma, S. Hu, and J. Wang, *Efficient low-rank regularization-based algorithms combining advanced techniques for solving tensor completion problems with application to color image recovering*, J. Comput. Appl. Math., 423 (2023), 114947.
[36] C. Mu, B. Huang, J. Wright, and D. Goldfarb, *Square deal: Lower bounds and improved relaxations for tensor recovery*, in Proc. Mach. Learn. Res. (PMLR), 32 (2014), pp. 73–81.

[37] I. V. Oseledets, *Tensor-train decomposition*, SIAM J. Sci. Comput., 33 (2011), pp. 2295–2317.
[38] G. Ottaviani, *Symplectic bundles on the plane, secant varieties and Lüroth quartics revisited*, in Vector Bundles and Low-codimensional Subvarieties: State of the Art and Recent Developments, Quad. Mat. 21, Aracne, Rome, 2007, pp. 315–352.
[39] B. Recht, *A simpler approach to matrix completion*, J. Mach. Learn. Res., 12 (2011), pp. 3413–3430.
[40] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
[41] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, Springer, Berlin, 2009.
[42] B. Romera-Paredes and M. Pontil, *A new convex relaxation for tensor completion*, in Advances in Neural Information Processing Systems, Curran Associates, Red Hook, NY, 2013, pp. 2967–2975.
[43] I. R. Shafarevich, *Basic Algebraic Geometry* 1: *Varieties in Projective Space*, Springer, New York, 1994.
[44] V. Strassen, *Rank and optimal computation of generic tensors*, Linear Algebra Appl., 52 (1983), pp. 645–685.
[45] L. R. Tucker, *Some mathematical notes on three-mode factor analysis*, Psychometrika, 31 (1966), pp. 279–311.
[46] P.-P. Wang, L. Li, and G.-H. Cheng, *Low-rank tensor completion with sparse regularization in a transformed domain*, Numer. Linear Algebra Appl., 28 (2021) e2387.
[47] W. Wang, V. Aggarwal, and S. Aeron, *Efficient low rank tensor ring completion*, in Proceedings of the IEEE International Conference on Computer Vision, IEEE Computer Society, Los Alamitos, CA, 2017, pp. 5698–5706.
[48] Y. Wang, C. Xu, S. You, C. Xu, and D. Tao, *DCT regularized extreme visual recovery*, IEEE Trans. Image Process., 26 (2017), pp. 3360–3371.
[49] 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), pp. 333–361.
[50] J. Xu, L. Zhang, D. Zhang, and X. Feng, *Multi-channel weighted nuclear norm minimization for real color image denoising*, in Proceedings of the IEEE International Conference on Computer Vision, IEEE Computer Society, Los Alamitos, CA, 2017, pp. 1096–1104.
[51] Y. Xu, R. Hao, W. Yin, and Z. Su, *Parallel matrix factorization for low-rank tensor completion*, Inverse Probl. Imaging, 9 (2015), pp. 601–624.
[52] S. Xue, W. Qiu, F. Liu, and X. Jin, *Low-rank tensor completion by truncated nuclear norm regularization*, in 24th International Conference on Pattern Recognition, IEEE, Piscataway, NJ, 2018, pp. 2600–2605.
[53] L. Yang, Z.-H. Huang, and X. Shi, *A fixed point iterative method for low n-rank tensor pursuit*, IEEE Trans. Signal Process., 61 (2013), pp. 2952–2962.
[54] 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, IEEE, Piscataway, NJ, 2019, pp. 3142–3146.
[55] Z. Zhang, G. Ely, S. Aeron, N. Hao, and M. Kilmer, *Novel methods for multilinear data completion and de-noising based on tensor-SVD*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Piscataway, NJ, 2014, pp. 3842–3849.
[56] Q. Zhao, L. Zhang, and A. Cichocki, *Bayesian CP factorization of incomplete tensors with automatic rank determination*, IEEE Trans. Pattern Anal. Mach. Intell., 37 (2015), pp. 1751–1763.
[57] Q. Zhao, G. Zhou, S. Xie, L. Zhang, and A. Cichocki, *Tensor Ring Decomposition*, preprint, arXiv:1606.05535, 2016.