

Tensor completion in hierarchical tensor representations

Holger Rauhut, Reinhold Schneider and Željka Stojanac

Abstract Compressed sensing extends from the recovery of sparse vectors from undersampled measurements via efficient algorithms to the recovery of matrices of low rank from incomplete information. Here we consider a further extension to the reconstruction of tensors of low multi-linear rank in recently introduced hierarchical tensor formats from a small number of measurements. Hierarchical tensors are a flexible generalization of the well-known Tucker representation, which have the advantage that the number of degrees of freedom of a low rank tensor does not scale exponentially with the order of the tensor. While corresponding tensor decompositions can be computed efficiently via successive applications of (matrix) singular value decompositions, some important properties of the singular value decomposition do not extend from the matrix to the tensor case. This results in major computational and theoretical difficulties in designing and analyzing algorithms for low rank tensor recovery. For instance, a canonical analogue of the tensor nuclear norm is NP-hard to compute in general, which is in stark contrast to the matrix case. In this book chapter we consider versions of iterative hard thresholding schemes adapted to hierarchical tensor formats. A variant builds on methods from Riemannian optimization and uses a retraction mapping from the tangent space of the manifold of low rank tensors back to this manifold. We provide first partial convergence results based on a tensor version of the restricted isometry property (TRIP) of the measurement map. Moreover, an estimate of the number of measurements is provided that ensures the TRIP of a given tensor rank with high probability for Gaussian measurement maps.

Holger Rauhut
RWTH Aachen University, Lehrstuhl C für Mathematik (Analysis), Templergraben 55, 52062
Aachen Germany, e-mail: rauhut@mathc.rwth-aachen.de

Reinhold Schneider
Technische Universität Berlin, Straße des 17. Juni 136, 10623 Berlin, e-mail: schneidr@math.tu-berlin.de

Željka Stojanac
RWTH Aachen University, Lehrstuhl C für Mathematik (Analysis), Templergraben 55, 52062
Aachen Germany, e-mail: stojanac@mathc.rwth-aachen.de

1 Introduction

As outlined in the introductory chapter of this book, *compressed sensing* allows the recovery of (approximately) sparse vectors from a small number of linear random measurements via efficient algorithms including ℓ_1 -minimization and iterative hard thresholding, see also [19, 15] for introductory material. This theory was later extended to the reconstruction of low rank matrices from random measurements in [55, 10, 11, 25]. An important special case includes the *matrix completion problem*, where one seeks to fill in missing entries of a low rank matrix [9, 10, 25, 56]. Corresponding algorithms include *nuclear norm minimization* [55, 10, 18] and versions of *iterative hard thresholding* [64].

In the present article we pursue a further extension of compressed sensing. We consider the recovery of a low rank tensor from a relatively small number of measurements. In contrast to already existing work in this direction [22, 36], we will understand low rank tensors in the framework of recently introduced *hierarchical tensor formats* [27]. This concept includes the classical *Tucker format* [65, 38, 14] as well as *tensor trains* [49, 50]. These hierarchical tensors can be represented in a data sparse way, i.e., they require only a very low number of data for their representation compared to the dimension of the full tensor space.

Let us recall the setup of low rank matrix recovery first. Given a matrix $\mathbf{X} \in \mathbb{R}^{n_1 \times n_2}$ of rank at most $r \ll \min\{n_1, n_2\}$, the goal of low rank matrix recovery is to reconstruct \mathbf{X} from linear measurements $b_i = 1, \dots, m$, i.e., $\mathbf{b} = \mathcal{A}(\mathbf{X})$, where $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \rightarrow \mathbb{R}^m$ with $m \ll n_1 n_2$ is a linear sensing operator.

This problem setting can be transferred to the problem to recover higher order tensors $\mathbf{u} \in \mathcal{H}_d := \mathbb{R}^{n_1 \times n_2 \times \dots \times n_d}$, $\boldsymbol{\mu} = (\mu_1, \dots, \mu_d) \mapsto \mathbf{u}(\mu_1, \dots, \mu_d)$ from the linear measurements $\mathbf{b} = \mathcal{A}(\mathbf{u})$, where $\mathcal{A} : \mathbb{R}^{n_1 \times n_2 \times \dots \times n_d} \rightarrow \mathbb{R}^m$, is a sensing operator $m \ll n_1 n_2 \dots n_d$. Here d denotes the order of the tensor (number of modes of a tensor) and we remark that, for easier readability, we use the notation $\mathbf{u}(\mu_1, \dots, \mu_d)$ referring to the entries of the tensor. In the present article we assume that the tensors to be reconstructed belong to a class of hierarchical tensors of a given low multilinear rank $\mathbf{r} = (r_j)_{j=1}^p$, where p depends on the specific tensor format [13], see also below. Of particular interest is the special case of *tensor completion*, where the measurement operator samples entries of the tensor, i.e.,

$$(\mathcal{A}\mathbf{u})_i = \mathbf{u}(\boldsymbol{\mu}_i) = \mathbf{u}(\mu_{1,i}, \dots, \mu_{d,i}) = b_i, \quad i = 1, \dots, m,$$

where the $\boldsymbol{\mu}_i \in \Omega$, $|\Omega| = m$, are given (multi-)indices [4, 62, 40]. Tensors, even of high order $d \gg 3$, appear frequently in data and signal analysis. For example, a video signal is a tensor of order $d = 3$. High order tensors of relatively low rank may also arise in vector-tensorization [28, 51] of a low dimensional signal. The present article tries to present a framework for tensor recovery from the perspective of recent developments in tensor product approximation [38, 5, 27], in particular the development of hierarchical tensors [27, 49]. The *canonical format* (CANDECOMP, PARAFAC) representing a tensor of order d as a sum of elementary tensor products, or rank one tensors (see [38, 5])

$$\begin{aligned}
\mathbf{u}(\mu_1, \dots, \mu_d) &= \sum_{k=1}^r (\mathbf{c}_k^1 \otimes \dots \otimes \mathbf{c}_k^d)(\mu_1, \dots, \mu_d) \\
&= \sum_{k=1}^r \mathbf{c}_k^1(\mu_1) \cdots \mathbf{c}_k^d(\mu_d) \quad , \quad \mu_i = 1, \dots, n_i \quad , \quad i = 1, \dots, d \quad ,
\end{aligned} \tag{1}$$

with $\mathbf{c}_k^i \in \mathbb{R}^{n_i}$, suffers from severe difficulties, unless $d \leq 2$. For example, the tensor rank is not well defined, and the set of tensors of the above form with fixed r is not closed [33] and does not form an algebraic variety. However, we obtain a closed subset, if we impose further conditions. Typical examples for such conditions are e.g. symmetry [41] or bounds $|\mathbf{c}_k^i| \leq \alpha$ for some fixed α [68].

However, it has been experienced that computations within the *Tucker tensor format* behave relatively robust and stable, whereas the complexity unfortunately still suffers from the curse of dimensionality. A first important observation may be summarized in the fact that the set of Tucker tensors with a Tucker rank at most $\mathbf{r} = (r_1, \dots, r_d)$ forms an algebraic variety, i.e., a set of common zeros of multivariate polynomials. Recently developed *hierarchical tensors*, introduced by Hackbusch and coworkers (HT tensors) [29, 23] and the group of Tyrtshnikov (tensor trains, TT) [49, 50] have extended the Tucker format [65, 38, 14] into a multi-level framework, that no longer suffers from high order scaling w.r.t. the order d , as long as the ranks are moderate. For $d = 3$ there is no essential difference, whereas for larger $d \geq 4$ one benefits from the use of the novel formats. This makes the Tucker format [62, 36, 76] and, in particular, its hierarchical generalization [13, 40, 53], the *hierarchical tensor format*, a proper candidate for tensor product approximation in the sense that it serves as an appropriate model class in which we would like to represent or approximate tensors of interest in a data sparse way. Several algorithms developed in compressed sensing and matrix recovery or matrix completion can be easily transferred to this tensor setting (with the exception of nuclear norm minimization, which poses some fundamental difficulties). However, we already note at this point that the analysis of algorithms is much harder for tensors than for matrices as we will see below.

Historically, the hierarchical tensor framework has evolved in the quantum physics community hidden in the renormalization group ideas [75], and became clearly visible in the framework of matrix product and tensor network states [61]. An independent source of these developments can be found in quantum dynamics as the multi-layer multiconfigurational time-dependent Hartree (MCTDH) method [3, 73, 46]. Only after the recent introduction of hierarchical tensor representations in numerics, namely Hierarchical Tucker (HT) [27, 29] and Tensor Trains (TT) [49, 50], its relationship to already existing concepts in quantum physics has been realized [42]. We refer the interested reader to the recent survey articles [26, 24, 42, 30] and the monograph [27]. In the present paper we would like to provide a fairly self-contained introduction, and demonstrate how these concepts can be applied for tensor recovery.

There are several essential difficulties when passing from matrix to tensor recovery. In matrix recovery, the original problem can be reformulated to finding the

solution of the optimization problem

$$\text{minimize } \text{rank}(\mathbf{Z}) \text{ s.t. } \mathcal{A}(\mathbf{Z}) = \mathbf{b}, \mathbf{Z} \in \mathbb{R}^{n_1 \times n_2},$$

i.e., to finding the matrix \mathbf{Z} with the lowest rank consistent with the measurements. While this problem is NP-hard [18], it can be relaxed to the convex optimization problem of constrained nuclear norm minimization

$$\text{minimize } \|\mathbf{Z}\|_* \text{ s.t. } \mathcal{A}(\mathbf{Z}) = \mathbf{b}, \mathbf{Z} \in \mathbb{R}^{n_1 \times n_2}.$$

Here, the nuclear norm is the sum of the singular values $\sigma_j(\mathbf{Z})$ of \mathbf{Z} , i.e., $\|\mathbf{Z}\|_* = \sum_j \sigma_j(\mathbf{Z})$. The minimizer of this problem reconstructs \mathbf{X} exactly under suitable conditions on \mathcal{A} [55, 11, 9, 25, 19].

In the hierarchical tensor setting, we are dealing with a rank tuple $\mathbf{r} = (r_1, \dots, r_p)$, which we would like to minimize simultaneously. However, this is not the only difficulty arising from the non-linear tensor ansatz. In fact, the tensor nuclear norm is NP-hard to compute [32, 20] and therefore, tensor nuclear norm minimization is computationally prohibitive. Another difficulty arises because in contrast to the matrix case, also the best rank \mathbf{r} -approximation to a given tensor is NP-hard to compute [32, 20].

Our model class of hierarchical tensors of fixed multi-linear rank \mathbf{r} is a smooth embedded manifold, and its closure constitutes an algebraic variety. These are properties on which one can build local optimization methods [1, 47, 60], subsumed under the moniker *Riemannian optimization*. Moreover, for *hierarchical tensor representation* efficient numerical tools for finding at least a quasi-best approximation are available, namely the *higher order singular value decomposition (HOSVD)*, related to the Tucker model [14], or the *hierarchical singular value decomposition (HSVD)*, which is an extension of the HOSVD to hierarchical Tucker models [52, 27, 23, 72]. All these methods proceed via successive computations of the SVD of certain matricisations of the original tensor.

The HSVD (and the HOSVD as a special case) enables us to compute rank \mathbf{r} approximations to a given tensor via truncation of the decomposition. This allows to extend a particular class of greedy type algorithms, namely *iterative hard thresholding algorithms* to the present tensor setting. In a wider sense, this class of techniques includes also related Riemannian manifold techniques [40, 13] and alternating least squares methods [35]. First numerical tests show promising results [13, 40, 53, 54]. For a convergence analysis in the tensor case, and in applications, however, we have to struggle with more and harder difficulties than in the matrix case. The most fundamental of these consists in the fact, that truncations of the HSVD only provide quasi-best low rank approximations. Although bounds of the approximation error are known [23], they are not good enough for our purposes, which is the main reason why we are only able to provide partial convergence results in this chapter. Another difficulty with iterative hard thresholding algorithms is that the rank, here a rank tuple $\mathbf{r} = (r_1, \dots, r_p)$, has to be fixed a priori. In practice this rank tuple is not known in advance, and a strategy for specifying appropriate ranks is required. Well known strategies borrowed from matrix recovery consist in increasing the rank during the

approximation or starting with overestimating the rank and reduce the ranks through the iteration [77]. For our seminal treatment, we simply assume that the multi-linear rank \mathbf{r} is known in advance, i.e., the sought tensor \mathbf{u} is of exact rank \mathbf{r} . Moreover, we assume noiseless measurements $(\mathcal{A}\mathbf{u})_j = b_j \in \mathbb{R}$, $j = 1, \dots, m$. The important issues of adapting ranks and obtaining robustness will be deferred to future research [54].

Our chapter is related to the one on *two algorithms for compressed sensing of sparse tensors* by S. Friedland, Q. Li, D. Schonfeld and E.E. Bernal. The latter chapter also considers the recovery of mode d -tensors from incomplete information using efficient algorithms. However, in contrast to our chapter, the authors assume usual sparsity of the tensor instead of the tensor being of low rank. The tensor structure is used in order to simplify the measurement process and to speed up the reconstruction rather than to work with the smallest possible number of measurements and to exploit low-rankness.

2 Hierarchical Tensors

2.1 Tensor product spaces

We start with some preliminaries. In the sequel, we consider only the real field $\mathbb{K} = \mathbb{R}$, but most parts are easy to extend to the complex case as well. We will confine ourselves to finite dimensional linear spaces $V_i = \mathbb{R}^{n_i}$ from which the tensor product space

$$\mathcal{H}_d = \bigotimes_{i=1}^d V_i := \bigotimes_{i=1}^d \mathbb{R}^{n_i} ,$$

is built [27]. If it is not stated explicitly, the $V_i = \mathbb{R}^{n_i}$ are supplied with the canonical basis $\{\mathbf{e}_1^i, \dots, \mathbf{e}_{n_i}^i\}$ of the vector space \mathbb{R}^{n_i} . Then any $\mathbf{u} \in \mathcal{H}_d$ can be represented as

$$\mathbf{u} = \sum_{\mu_1=1}^{n_1} \dots \sum_{\mu_d=1}^{n_d} \mathbf{u}(\mu_1, \dots, \mu_d) \mathbf{e}_{\mu_1}^1 \otimes \dots \otimes \mathbf{e}_{\mu_d}^d .$$

Using this basis, with a slight abuse of notation, we can identify $\mathbf{u} \in \mathcal{H}_d$ with its representation by a d -variate function, often called hyper matrix,

$$\boldsymbol{\mu} = (\mu_1, \dots, \mu_d) \mapsto \mathbf{u}(\mu_1, \dots, \mu_d) \in \mathbb{R} , \quad \mu_i = 1, \dots, n_i , \quad i = 1, \dots, d ,$$

depending on discrete variables, usually called indices $\mu_i = 1, \dots, n_i$, and $\boldsymbol{\mu}$ is called a multi-index. Of course, the actual representation $\mathbf{u}(\dots)$ of $\mathbf{u} \in \mathcal{H}_d$ depends on the chosen bases of V_1, \dots, V_d . With $n = \max\{n_i : i = 1, \dots, d\}$, the number of possibly non-zero entries in the representation of \mathbf{u} is $n_1 \dots n_d = \mathcal{O}(n^d)$. This is often referred to as the *curse of dimensions*. We equip the linear space \mathcal{H}_d with the ℓ_2 -norm $\|\mathbf{u}\| =$

$\sqrt{\langle \mathbf{u}, \mathbf{u} \rangle}$ and the inner product

$$\langle \mathbf{u}, \mathbf{v} \rangle := \sum_{\mu_1=1}^{n_1} \cdots \sum_{\mu_d=1}^{n_d} \mathbf{u}(\mu_1, \dots, \mu_d) \mathbf{v}(\mu_1, \dots, \mu_d) .$$

We distinguish linear operators between vector spaces and their corresponding representation by matrices, which are written by capital bold letters \mathbf{U} . Throughout this chapter, all tensor contractions or various tensor–tensor products are either defined explicitly, by summation over corresponding indices, or by introducing corresponding matricisations of the tensors and performing matrix–matrix products.

2.2 Subspace approximation

The essence of the classical Tucker format is that, given a tensor \mathbf{u} and a rank-tuple $\mathbf{r} = (r_j)_{j=1}^d$, one is searching for optimal subspaces $U_i \subset \mathbb{R}^{n_i}$ such that

$$\min \|\mathbf{u} - \mathbf{v}\|, \text{ where } \mathbf{v} \in U_1 \otimes \cdots \otimes U_d,$$

is minimized over U_1, \dots, U_d with $\dim U_i = r_i$. Equivalently, we are looking for corresponding bases $\mathbf{b}_{k_i}^i$ of U_i , which can be written in the form

$$\mathbf{b}_{k_i}^i := \sum_{\mu_i=1}^{n_i} \mathbf{b}^i(\mu_i, k_i) \mathbf{e}_{\mu_i}^i, \quad k_i = 1, \dots, r_i < n_i, \quad (2)$$

where $\mathbf{b}^i(k_i, \mu_i) \in \mathbb{R}$, for each coordinate direction $i = 1, \dots, d$. With a slight abuse of notation we often identify the basis vector with its representation

$$\mathbf{b}_{k_i}^i \simeq (\mu_i \mapsto \mathbf{b}^i(\mu_i, k_i)), \quad \mu_i = 1, \dots, n_i, \quad k_i = 1, \dots, r_i,$$

i.e., a discrete function or an n_i -tuple. This concept of subspace approximation can be used either for an approximation \mathbf{u} of a single tensor, as well as for an ensemble of tensors \mathbf{u}_j , $j = 1, \dots, m$, in tensor product spaces. Given the bases $\mathbf{b}_{k_i}^i$, \mathbf{u}_j can be represented by

$$\mathbf{u}_j = \sum_{k_1=1}^{r_1} \cdots \sum_{k_d=1}^{r_d} \mathbf{c}(j, k_1, \dots, k_d) \mathbf{b}_{k_1}^1 \otimes \cdots \otimes \mathbf{b}_{k_d}^d \in \bigotimes_{i=1}^d U_i \subset \mathcal{H}_d = \bigotimes_{i=1}^d \mathbb{R}^{n_i}. \quad (3)$$

In case \mathbf{b}^i 's form orthonormal bases, the core tensor $\mathbf{c} \in \mathbb{R}^m \otimes \bigotimes_{i=1}^d \mathbb{R}^{r_i}$ is given entry-wise by

$$\mathbf{c}(j, k_1, \dots, k_d) = \langle \mathbf{u}_j, \mathbf{b}_{k_1}^1 \otimes \cdots \otimes \mathbf{b}_{k_d}^d \rangle .$$

We call a representation of the form (3) with some $\mathbf{b}_{k_i}^i$, \mathbf{c} a Tucker representation, and the Tucker representations the Tucker format. In this formal parametrization,

the upper limit of the sums may be larger than the ranks and $\{\mathbf{b}_{k_i}^i\}_{k_i}$ may not be linearly independent. Noticing that a Tucker representation of a tensor is not uniquely defined, we are interested in some normal form.

Since the core tensor contains $r_1 \cdots r_d \sim r^d$, $r := \max\{r_i : i = 1, \dots, d\}$, possibly nonzero entries, this concept does not prevent the number of free parameters from scaling exponentially with the dimensions $\mathcal{O}(r^d)$. Setting $n := \max\{n_i : i = 1, \dots, d\}$, the overall complexity for storing the required data (including the basis vectors) is bounded by $\mathcal{O}(ndr + r^d)$. Since n_i is replaced by r_i , one obtains a dramatical compression $\frac{r_1}{n_1} \cdots \frac{r_d}{n_d} \sim \left(\frac{r}{n}\right)^d$. Without further sparsity of the core tensors the Tucker format is appropriate for low order tensors $d < 4$.

2.3 Hierarchical tensor representation

The *hierarchical Tucker format* (HT) in the form introduced by Hackbusch and Kühn in [29], extends the idea of subspace approximation to a hierarchical or multi-level framework. Let us proceed in a hierarchical way. We first consider $V_1 \otimes V_2 = \mathbb{R}^{n_1} \otimes \mathbb{R}^{n_2}$ or preferably the subspaces $U_1 \otimes U_2$ introduced in the previous section. For the approximation of $\mathbf{u} \in \mathcal{H}_d$ we only need a subspace $U_{\{1,2\}} \subset U_1 \otimes U_2$ with dimension $r_{\{1,2\}} < r_1 r_2$. Indeed, $V_{\{1,2\}}$ is defined through a new basis

$$V_{\{1,2\}} = \text{span} \{\mathbf{b}_{k_{\{1,2\}}}^{\{1,2\}} : k_{\{1,2\}} = 1, \dots, r_{\{1,2\}}\},$$

with basis vectors given by

$$\mathbf{b}_{k_{\{1,2\}}}^{\{1,2\}} = \sum_{k_1=1}^{r_1} \sum_{k_2=1}^{r_2} \mathbf{b}^{\{1,2\}}(k_1, k_2, k_{\{1,2\}}) \mathbf{b}_{k_1}^1 \otimes \mathbf{b}_{k_2}^2, \quad k_{\{1,2\}} = 1, \dots, r_{\{1,2\}}.$$

One may continue in several ways, e.g. by building a subspace $U_{\{1,2,3\}} \subset U_{\{1,2\}} \otimes U_3 \subset U_1 \otimes U_2 \otimes U_3 \subset V_1 \otimes V_2 \otimes V_3$, or $U_{\{1,2,3,4\}} \subset U_{\{1,2\}} \otimes U_{\{3,4\}}$, where $U_{\{3,4\}}$ is defined analogously to $U_{\{1,2\}}$ and so on.

For a systematic treatment, this approach can be cast into the framework of a partition tree, with leaves $\{1\}, \dots, \{d\}$, simply abbreviated here by $1, \dots, d$, and vertices $\alpha \subset D := \{1, \dots, d\}$, corresponding to the partition $\alpha = \alpha_1 \cup \alpha_2$, $\alpha_1 \cap \alpha_2 = \emptyset$. Without loss of generality, we can assume that $i < j$, for all $i \in \alpha_1$, $j \in \alpha_2$. We call α_1, α_2 the *sons* of the *father* α and D is called the *root* of the tree. In the example above we have $\alpha := \{1, 2, 3\} = \alpha_1 \cup \alpha_2 = \{1, 2\} \cup \{3\}$, where $\alpha_1 := \{1, 2\}$ and $\alpha_2 := \{3\}$.

In general, we do not need to restrict the number of sons, and define the *coordination number* by the number of sons +1 (for the father). Restricting to a binary tree so that each node contains two sons for non-leaf nodes (i.e. $\alpha \neq \{i\}$), is often the common choice, which we will also consider here. Let $\alpha_1, \alpha_2 \subset D$ be the two sons of $\alpha \subset D$, then $U_\alpha \subset U_{\alpha_1} \otimes U_{\alpha_2}$, is defined by a basis

$$\mathbf{b}_\ell^\alpha = \sum_{i=1}^{r_{\alpha_1}} \sum_{j=1}^{r_{\alpha_2}} \mathbf{b}^\alpha(i, j, \ell) \mathbf{b}_i^{\alpha_1} \otimes \mathbf{b}_j^{\alpha_2}. \quad (4)$$

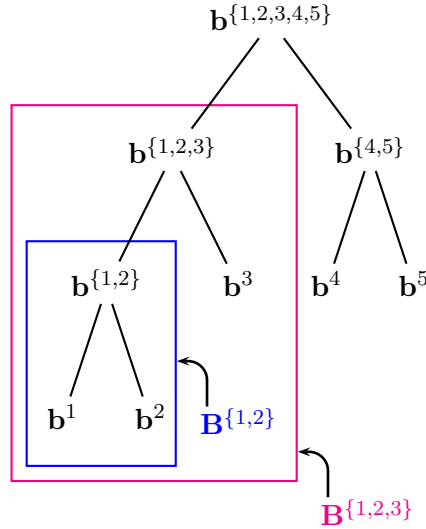
They can also be considered as matrices $(\mu_\alpha, \ell) \mapsto \mathbf{B}^\alpha(\mu_\alpha, \ell) \in \mathbb{R}^{n_\alpha \times r_\alpha}$ with $n_\alpha = \prod_{\ell \in \alpha} n_\ell$, for $\alpha \neq \{i\}$. Without loss of generality, all basis vectors, e.g. $\{\mathbf{b}_\ell^\alpha : \ell = 1, \dots, r_\alpha\}$, can be constructed to be orthonormal, as long as $\alpha \neq D$ is not the root. The tensors $(\ell, i, j) \mapsto \mathbf{b}^\alpha(i, j, \ell)$ will be called *transfer* or *component tensors*. For a leaf $\{i\} \simeq i$, the tensor $(\mu_i, k_i) \mapsto \mathbf{b}^i(\mu_i, k_i)$ in (2) denotes a *transfer* or *component tensor*. The component tensor $\mathbf{b}^D = \mathbf{b}^{\{1, \dots, d\}}$ at the root is called the *root tensor*.

Since the matrices \mathbf{B}^α are too large, we avoid computing them. We store only the *transfer* or *component tensors* which, for fixed $\ell = 1, \dots, r_\alpha$, can also be casted into *transfer matrices* $(i, j) \mapsto \mathbf{B}_\alpha(\ell)_{i,j} \in \mathbb{R}^{r_{\alpha_1} \times r_{\alpha_2}}$.

Proposition 1 ([27]). A tensor $\mathbf{u} \in \mathcal{H}_d$ is completely parametrized by the transfer tensors \mathbf{b}^α , $\alpha \in \mathbb{T}$, i.e., by a multi-linear function τ

$$(\mathbf{b}^\alpha)_{\alpha \in \mathbb{T}} \mapsto \mathbf{u} = \tau(\{\mathbf{b}^\alpha : \alpha \in \mathbb{T}\}).$$

Indeed τ is defined by applying (4) recursively. Since \mathbf{b}^α depends bi-linearly on \mathbf{b}^{α_1} and \mathbf{b}^{α_2} , the composite function τ is multi-linear in its arguments \mathbf{b}^α .



Hierarchical Tensor representation of an order 5 tensor

Data complexity Let $n := \max\{n_i : i = 1, \dots, d\}$, $r := \max\{r_\alpha : \alpha \in \mathbb{T}\}$. Then the number of data required for the representation is $\mathcal{O}(ndr + dr^3)$, in particular does not scale exponentially w.r.t. the order d .

2.4 Tensor trains and matrix product representation

We now highlight another particular case of hierarchical tensor representations, namely *Tyrtysnikov tensors (TT)* or *tensor trains* and *matrix product representations* defined by taking $U_{\{1,\dots,p+1\}} \subset U_{\{1,\dots,p\}} \otimes V_{\{p+1\}}$, developed as TT tensors (tensor trains) by [50, 52] and known as matrix product states (MPS) in physics. Therein, we abbreviate $i \simeq \{1, \dots, i\}$ and consider the unbalanced tree $\mathbb{T} = \{\{1\}, \{2\}, \{1, 2\}, \{3\}, \{1, 2, 3\}, \dots, \{d\}, \{1, \dots, d\}\}$ and setting $r_0 = r_d = 1$. The transfer tensor \mathbf{b}^α for a leaf $\alpha \in \{\{2\}, \{3\}, \dots, \{d\}\}$ is usually defined as identity matrix of appropriate size and therefore the tensor $\mathbf{u} \in \mathcal{H}_d$ is completely parametrized by transfer tensors $(\mathbf{b}^\alpha)_{\alpha \in \mathbb{T}'}$, where $\mathbb{T}' = \{1, \dots, d\} = \{\{1\}, \{1, 2\}, \dots, \{1, 2, \dots, d\}\}$. Applying the recursive construction, the tensor \mathbf{u} can be written as

$$\begin{aligned} (\mu_1, \dots, \mu_d) &\mapsto \mathbf{u}(\mu_1, \dots, \mu_d) \\ &= \sum_{k_1=1}^{r_1} \dots \sum_{k_{d-1}=1}^{r_{d-1}} \mathbf{b}^1(\mu_1, k_1) \mathbf{b}^2(k_1, \mu_2, k_2) \dots \mathbf{b}^d(k_{d-1}, \mu_d). \end{aligned} \quad (5)$$

Introducing the matrices $\mathbf{B}_i(\mu_i) \in \mathbb{R}^{r_{i-1} \times r_i}$,

$$(\mathbf{B}_i(\mu_i))_{k_{i-1}, k_i} = \mathbf{b}^i(k_{i-1}, \mu_i, k_i), \quad 1 \leq i \leq d,$$

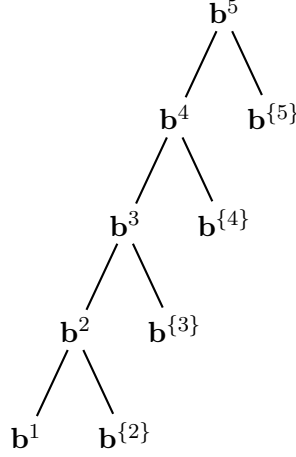
and with the convention $r_0 = r_d = 1$

$$(\mathbf{B}_1(\mu_1))_{k_1}^* = \mathbf{b}^1(\mu_1, k_1) \quad \text{and} \quad (\mathbf{B}_d(\mu_d))_{k_{d-1}} = \mathbf{b}^d(k_{d-1}, \mu_d),$$

the formula (5) can be rewritten entry-wise by matrix–matrix products

$$\mathbf{u}(\mu_1, \dots, \mu_d) = \mathbf{B}_1(\mu_1) \cdots \mathbf{B}_i(\mu_i) \cdots \mathbf{B}_d(\mu_d) = \tau(\mathbf{b}^1, \dots, \mathbf{b}^d). \quad (6)$$

This representation is by no means unique. In general, there exist $\mathbf{b}^\alpha \neq \mathbf{c}^\alpha$ such that $\tau(\{\mathbf{b}^\alpha : \alpha \in \mathbb{T}\}) = \tau(\{\mathbf{c}^\alpha : \alpha \in \mathbb{T}\})$.



TT representation of an order 5 tensor with abbreviation $i \simeq \{1, \dots, i\}$

The tree is ordered according to the father-son relation into a hierarchy of levels, where \mathbf{b}^d is the root tensor. Let us observe that we can rearrange the hierarchy in such a way that any node $p = 1, \dots, d$ can form the root of the tree, i.e., \mathbf{b}^p becomes the root tensor. Using only orthogonal basis vectors, which is the preferred choice, this ordering reflects left and right hand orthogonalization in matrix product states [34].

A tensor in canonical form

$$\mathbf{u} = \sum_{k=1}^R \mathbf{u}_k^1 \otimes \dots \otimes \mathbf{u}_k^d$$

can be easily written in the TT form, by setting $r_i = R$, for all $i = 1, \dots, d-1$ and

$$\mathbf{b}^i(k_{i-1}, \mu_i, k_i) = \begin{cases} \mathbf{u}_k^i(\mu_i) & \text{if } k_{i-1} = k_i = k, i = 2, \dots, d-1 \\ 0 & \text{if } k_{i-1} \neq k_i, i = 2, \dots, d-1 \\ \mathbf{u}_k^i(\mu_i) & \text{if } k_i = k, i = 1 \\ \mathbf{u}_k^i(\mu_i) & \text{if } k_{i-1} = k, i = d \end{cases}.$$

Data complexity: Let $n := \max\{n_i : i = 1, \dots, d\}$, $r := \max\{r_j : j = 1, \dots, d-1\}$. Then the number of data required for the presentation is $\mathcal{O}(dnr^2)$. Computing a single entry of a tensor requires the matrix multiplication of d matrices of size at most $r \times r$. This can be performed in $\mathcal{O}(ndr^3)$ operations.

Since the parametrization τ can be written in the simple matrix product form (6), we will consider the TT format often as a prototype model, and use it frequently for our explanations. We remark that most properties can easily be extended to the general hierarchical case with straightforward modifications [27], and we leave those modifications to the interested reader.

2.5 Matricisation of a tensor and its multi-linear rank

Let \mathbf{u} be a tensor in \mathcal{H}_d . Given a fixed dimension tree \mathbb{T} , for each node $\alpha \in \mathbb{T}$, $\alpha \neq D$, we can build a matrix \mathbf{U}^α from \mathbf{u} by grouping the indices μ_i , with $i \in \alpha$ into a row index I and the remaining indices μ_j with $j \in D \setminus \alpha$ into the column index J of the matrix $\mathbf{U}^\alpha = (\mathbf{U}_{I,J}^\alpha)$. For the root $\alpha = D$ we simply take the vectorized tensor $\mathbf{U}^D \in \mathbb{R}^{n_1 \cdots n_d \times 1}$. Since the rank of this matrix is one, it is often omitted.

For example, in the Tucker case, for $\alpha = \{i\}$ being a leaf, we set $I = \mu_i$ and $J = (\mu_1, \dots, \mu_{i-1}, \mu_{i+1}, \dots, \mu_d)$ providing a matrix

$$\mathbf{U}_{\mu_i;(\mu_1, \dots, \mu_{i-1}, \mu_{i+1}, \dots, \mu_d)}^\alpha = \mathbf{u}(\mu_1, \dots, \mu_d).$$

Similar, in the TT-format, with the convention that $i \simeq \{1, \dots, i\}$, we obtain matrices $\mathbf{U}^i \in \mathbb{R}^{n_1 \cdots n_i \times n_{i+1} \cdots n_d}$ with entries

$$\mathbf{U}_{(\mu_1, \dots, \mu_i);(\mu_{i+1}, \dots, \mu_d)}^i = \mathbf{u}(\mu_1, \dots, \mu_d).$$

Definition 1. Given a dimension tree \mathbb{T} with p nodes, we define the *multi-linear rank* by the p -tuple $\mathbf{r} = (r_\alpha)_{\alpha \in \mathbb{T}}$ with $r_\alpha = \text{rank}(\mathbf{U}^\alpha)$, $\alpha \in \mathbb{T}$. The set of tensors $\mathbf{u} \in \mathcal{H}_d$ of given multi-linear rank \mathbf{r} will be denoted by $\mathcal{M}_{\mathbf{r}}$. The set of all tensors of rank \mathbf{s} at most \mathbf{r} , i.e., $s_\alpha \leq r_\alpha$ for all $\alpha \in \mathbb{T}$ will be denoted by $\mathcal{M}_{\leq \mathbf{r}}$.

Unlike the matrix case, it is possible that for some tuples \mathbf{r} , $\mathcal{M}_{\mathbf{r}} = \emptyset$ [12]. However, since our algorithm works on a closed nonempty set $\mathcal{M}_{\leq \mathbf{r}}$, this issue does not concern us.

In contrast to the canonical format (1), also known as CANDECOMP/PARAFAC, see [37, 38] and the border rank problem [41], in the present setting the rank is a well defined quantity. This fact makes the present concept highly attractive for tensor recovery. On the other hand, if a tensor \mathbf{u} is of rank \mathbf{r} then there exists a component tensor \mathbf{b}^α of the form (4) where $\ell = 1, \dots, r_\alpha$.

It is well known that the set of all matrices $\mathbf{A} \in \mathbb{R}^{n \times m}$ of rank at most r is a set of common zeros of multi-variate polynomials, i.e., an algebraic variety. The set $\mathcal{M}_{\leq \mathbf{r}}$ is the set of all tensors $\mathbf{u} \in \mathcal{H}_d$, where the matrices \mathbf{U}^α have a rank at most r_α . Therefore, it is again a set of common zeros of multivariate polynomials.

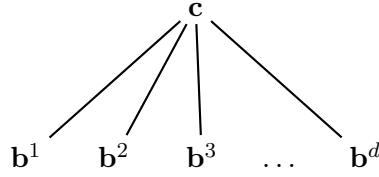
2.6 Higher order singular value decomposition

Let us provide more details about the rather classical higher order singular value decomposition. Above we have considered only binary dimension trees \mathbb{T} , but we can extend the considerations also to N -ary trees with $N \geq 3$. The d -ary tree \mathbb{T} (the tree with a root with d sons $i \simeq \{i\}$) induces the so-called Tucker decomposition and the corresponding higher order singular value decomposition (HOSVD). The Tucker decomposition was first introduced by Tucker in 1963 [66] and has been refined later on in many works, see e.g. [43, 67, 66].

Definition 2 (Tucker decomposition). Given a tensor $\mathbf{u} \in \mathbb{R}^{n_1 \times \dots \times n_d}$, the decomposition

$$\mathbf{u}(\mu_1, \dots, \mu_d) = \sum_{k_1=1}^{r_1} \dots \sum_{k_d=1}^{r_d} \mathbf{c}(k_1, \dots, k_d) \mathbf{b}_{k_1}^1(\mu_1) \dots \mathbf{b}_{k_d}^d(\mu_d),$$

$r_i \leq n_i$, $i = 1, \dots, d$, is called a Tucker decomposition. The tensor $\mathbf{c} \in \mathbb{R}^{r_1 \times \dots \times r_d}$ is called the core tensor and the $\mathbf{b}_{k_i}^i \in \mathbb{R}^{n_i}$, for $i = 1, \dots, d$, form a basis of the subspace $U_i \subset \mathbb{R}^{n_i}$. They can also be considered as transfer or component tensors $(\mu_i, k_i) \mapsto \mathbf{b}^i(\mu_i, k_i) \in \mathbb{R}^{n_i \times r_i}$.



Tucker representation of an order d tensor

Notice that the Tucker decomposition is highly non-unique. For an $i \in \{1, \dots, d\}$ and invertible matrix $\mathbf{Q}_i \in \mathbb{R}^{r_i \times r_i}$, one can define a matrix $\bar{\mathbf{B}}^i = \mathbf{B}^i \mathbf{Q}_i$ and the tensor $\bar{\mathbf{c}}_i$

$$\bar{\mathbf{c}}_i(k_1, \dots, k_d) = \sum_{\bar{k}_i=1}^{r_i} \mathbf{c}_i(k_1, \dots, \bar{k}_i, \dots, k_d) \mathbf{Q}_i^{-1}(\bar{k}_i, k_i)$$

such that the tensor \mathbf{u} can also be written as

$$\mathbf{u}(\mu_1, \dots, \mu_d) = \sum_{k_1=1}^{r_1} \dots \sum_{k_d=1}^{r_d} \bar{\mathbf{c}}_i(k_1, \dots, k_d) \mathbf{b}_{k_1}^1(\mu_1) \dots \bar{\mathbf{b}}_{k_i}^i(\mu_i) \dots \mathbf{b}_{k_d}^d(\mu_d).$$

Similarly to the matrix case and the singular value decomposition, one can impose orthogonality conditions on the matrices \mathbf{B}^i , for all $i = 1, \dots, d$, i.e., we assume that $\{\mathbf{b}_{k_i}^i : k_i = 1, \dots, r_i\}$ are orthonormal bases. However, in this case one does not obtain a super-diagonal core tensor \mathbf{c} .

Definition 3 (HOSVD decomposition). The HOSVD decomposition of a given tensor $\mathbf{u} \in \mathcal{H}_d$ is a special case of the Tucker decomposition where

- the bases $\{\mathbf{b}_{k_i}^i \in \mathbb{R}^{n_i} : k_i = 1, \dots, r_i\}$ are orthogonal and normalized, for all $i = 1, \dots, d$,
- the tensor $\mathbf{c} \in \mathcal{H}_d$ is all orthogonal, i.e., $\langle \mathbf{c}_{k_i=p}, \mathbf{c}_{k_i=q} \rangle = 0$, for all $i = 1, \dots, d$ and whenever $p \neq q$,
- the subtensors of the core tensor \mathbf{c} are ordered according to their ℓ_2 norm, i.e., $\|\mathbf{c}_{k_i=1}\| \geq \|\mathbf{c}_{k_i=2}\| \geq \dots \geq \|\mathbf{c}_{k_i=n_i}\| \geq 0$.

Here, the subtensor $\mathbf{c}_{k_i=p} \in \mathbb{R}^{n_1 \times \dots \times n_{i-1} \times n_{i+1} \times \dots \times n_d}$ is a tensor of order $d - 1$ defined as

$$\mathbf{c}_{k_i=p}(\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_{i-1}, \boldsymbol{\mu}_{i+1}, \dots, \boldsymbol{\mu}_d) = \mathbf{c}(\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_{i-1}, p, \boldsymbol{\mu}_{i+1}, \dots, \boldsymbol{\mu}_d).$$

The HOSVD can be computed via successive SVDs of appropriate unfoldings or matricisations $\mathbf{U}^{(i)} = \mathbf{U}^i$, see e.g. [38] and below for the more general HSVD. For more information on this decomposition, we refer the interested reader to [14, 27].

2.7 Hierarchical singular value decomposition and truncation

The singular value decomposition of the matricisation \mathbf{U}^α , $\alpha \in \mathbb{T}$, is factorizing the tensor into two parts. Thereby, we separate the tree into two subtrees. Each part can be treated independently in an analogous way as before by applying a singular value decomposition. This procedure can be continued in a way such that one ends up with an explicit description of the component tensors. There are several sequential orders one can proceed including top-down and bottom-up strategies. We will call every decomposition of the above type a *higher order singular value decomposition (HOSVD)* or in the hierarchical setting a *hierarchical singular value decomposition (HSVD)*. As long as no approximation, i.e., no truncation, has been applied during the corresponding SVDs, one obtains an exact recovery of the original tensor at the end. The situation changes if we apply truncations (via thresholding). Then the result may depend on the way and the order we proceed as well as on the variant of the HSVD.

In order to become more explicit let us demonstrate an HSVD procedure for the model example of a TT-tensor [49], already introduced in [72] for the matrix product representation. Without truncations this algorithm provides an exact reconstruction with a TT representation provided the multi-linear rank $\mathbf{s} = (s_1, \dots, s_{d-1})$ is chosen large enough. In general, the s_i 's can be chosen to be larger than the dimensions n_i . Via inspecting the ranks of the relevant matricisations, the multilinear rank \mathbf{s} may be determined a priori.

1. Given: $\mathbf{u} \in \mathcal{H}_d$ of multi-linear rank $\mathbf{s} = (s_1, \dots, s_{d-1})$, $s_0 = s_d := 1$.
2. Set $\mathbf{v}^1 = \mathbf{u}$.
3. **For** $i = 1, \dots, d-1$ **do**

- Form matricisation $\mathbf{V}^i \in \mathbb{R}^{s_{i-1} n_i \times n_{i+1} \cdots n_d}$ via

$$\mathbf{V}_{(k_{i-1}, \boldsymbol{\mu}_i); (\boldsymbol{\mu}_{i+1}, \dots, \boldsymbol{\mu}_d)}^i = \mathbf{v}^i(k_{i-1}, \boldsymbol{\mu}_i, \boldsymbol{\mu}_{i+1}, \dots, \boldsymbol{\mu}_d).$$

- Compute the SVD of \mathbf{V}^i :

$$\mathbf{V}_{(k_{i-1}, \boldsymbol{\mu}_i); (\boldsymbol{\mu}_{i+1}, \dots, \boldsymbol{\mu}_d)}^i = \sum_{k_i=1}^{s_i} \sigma_{k_i}^i \mathbf{b}^i(k_{i-1}, \boldsymbol{\mu}_i, k_i) \mathbf{d}^{i+1}(k_i, \boldsymbol{\mu}_{i+1}, \dots, \boldsymbol{\mu}_d),$$

where the $\mathbf{b}^i(\cdot, \cdot, k_i)$ and the $\mathbf{d}^{i+1}(k_i, \dots)$ are orthonormal (i.e., the left and right singular vectors) and the $\sigma_{k_i}^i$ are the nonzero singular values of \mathbf{V}^i .

- Set $\mathbf{v}^{i+1}(k_i, \mu_{i+1}, \dots, \mu_d) = \sigma_{k_i}^i \mathbf{d}^{i+1}(k_i, \mu_{i+1}, \dots, \mu_d)$
- 4. Set $\mathbf{b}^d(k_{d-1}, \mu_d) := \mathbf{v}^d(k_{d-1}, \mu_d)$ and $\mathbf{B}_i(\mu_i)_{k_{i-1}, k_i} = \mathbf{b}^i(k_{i-1}, \mu_i, k_i)$ for $i = 1, \dots, d$.
- 5. Decomposition $\mathbf{u}(\boldsymbol{\mu}) = \mathbf{B}_1(\mu_1) \cdots \mathbf{B}_d(\mu_d)$.

Above, the indices k_i run from 1 to n_i and, for notational consistency, $k_0 = k_d = 1$. Let us notice that the present algorithm is not the only way to use multiple singular value decompositions in order to obtain a hierarchical representation of \mathbf{u} for the given tree, here a TT representation. For example, one may start at the right end separating \mathbf{b}^d first and so on. The procedure above provides some *normal form* of the tensor.

Let us now explain hard thresholding on the example of a TT tensor and the HSVD defined above. This procedure remains essentially the same with the only difference that we apply a thresholding to a target rank $\mathbf{r} = (r_i)_{i=1}^{d-1}$ with $r_i \leq s_i$ at each step by setting $\sigma_{k_i}^i = 0$ for all $k_i > r_i$, $i = 1, \dots, d-1$, where the $(\sigma_{k_i}^i)_{k_i}$ is the monotonically decreasing sequence of singular values of \mathbf{V}^i . This leads to an approximate right factor \mathbf{v}_ε^i , within a controlled ℓ_2 error $\varepsilon_i = \sqrt{\sum_{k_i > r_i} (\sigma_{k_i}^i)^2}$. By the *hard thresholding* HSVD procedure presented above, one obtains a unique approximate tensor

$$\mathbf{u}_\varepsilon := \mathbf{H}_\mathbf{r}(\mathbf{u}) \quad (7)$$

of multi-linear rank \mathbf{r} within a guaranteed error bound

$$\|\mathbf{u}_\varepsilon - \mathbf{u}\| \leq \sum_{i=1}^{d-1} \varepsilon_i.$$

In contrast to the matrix case, this approximation \mathbf{u}_ε , however, may not be the best rank \mathbf{r} approximation of \mathbf{u} , which is in fact NP hard to compute [32, 20]. A more evolved analysis shows the following quasi-optimal error bound.

Theorem 1. *Let $\mathbf{u}_\varepsilon = \mathbf{H}_\mathbf{r}(\mathbf{u})$. Then there exists $C(d) = \mathcal{O}(\sqrt{d})$, such that \mathbf{u}_ε satisfies the quasi-optimal error bound*

$$\inf\{\|\mathbf{u} - \mathbf{v}\| : \mathbf{v} \in \mathcal{M}_{\leq \mathbf{r}}\} \leq \|\mathbf{H}_\mathbf{r}(\mathbf{u})\| \leq C(d) \inf\{\|\mathbf{u} - \mathbf{v}\| : \mathbf{v} \in \mathcal{M}_{\leq \mathbf{r}}\}. \quad (8)$$

The constant satisfies $C(d) = \sqrt{d}$ for the Tucker format [23], $C(d) = \sqrt{d-1}$ for the TT format [50] and $C(d) = \sqrt{2d-3}$ for a balanced tree in the HSVD of [23].

The procedure introduced above can be modified to apply for general hierarchical tensor representations. When we consider the HSVD in the sequel, we have in mind that we have fixed our hierarchical SVD method choosing one of the several variants.

2.8 Hierarchical tensors as differentiable manifolds

It has been shown that, fixing a tree \mathbb{T} , the set of hierarchical tensors of exactly multi-linear rank \mathbf{r} forms an analytical manifold [34, 69, 17, 47]. We will describe its essential features using the TT format or the matrix product representation. For an invertible $r_1 \times r_1$ matrix \mathbf{G}_1 , it holds

$$\begin{aligned} \mathbf{u}(\boldsymbol{\mu}) &= \mathbf{B}^1(\mu_1)\mathbf{B}^2(\mu_2)\cdots\mathbf{B}^i(\mu_i)\cdots\mathbf{B}^d(\mu_d) \\ &= \mathbf{B}^1(\mu_1)\mathbf{G}_1\mathbf{G}_1^{-1}\mathbf{B}^2(\mu_2)\cdots\mathbf{B}^i(\mu_i)\cdots\mathbf{B}^d(\mu_d) \\ &= \tilde{\mathbf{B}}^1(\mu_1)\tilde{\mathbf{B}}^2(\mu_2)\cdots\mathbf{B}^i(\mu_i)\cdots\mathbf{B}^d(\mu_d), \end{aligned}$$

where $\tilde{\mathbf{B}}^1(\mu_1) = \mathbf{B}^1(\mu_1)\mathbf{G}_1$ and $\tilde{\mathbf{B}}^2(\mu_2) = \mathbf{G}_1^{-1}\mathbf{B}^2(\mu_2)$. This provides two different representations of the same tensor \mathbf{u} . In order to remove the redundancy in the above parametrization of the set $\mathcal{M}_{\mathbf{r}}$, let us consider the linear space of parameters $(\mathbf{b}^1, \dots, \mathbf{b}^d) \in \mathcal{X} := \times_{i=1}^d X_i$, $X_i := \mathbb{R}^{r_{i-1}r_i}$, or equivalently $\mathcal{U} := (\mathbf{B}^1(\cdot), \dots, \mathbf{B}^d(\cdot))$, together with a Lie group action. For a collection of invertible matrices $\mathcal{G} = (\mathbf{G}_1, \dots, \mathbf{G}_{d-1})$ we define a transitive group action by

$$\mathcal{G} \circ \mathcal{U} := (\mathbf{B}^1\mathbf{G}_1, \mathbf{G}_1^{-1}\mathbf{B}^2\mathbf{G}_2, \dots, \mathbf{G}_{d-1}^{-1}\mathbf{B}^d).$$

One observes that the tensor \mathbf{u} remains unchanged under this transformation of the component tensors. Therefore, we will identify two representations $\mathcal{U}_1 \sim \mathcal{U}_2$, if there exists \mathcal{G} such that $\mathcal{U}_2 = \mathcal{G} \circ \mathcal{U}_1$. It is easy to see that the equivalence classes $[\mathcal{U}] := \{\mathcal{V} : \mathcal{U} \sim \mathcal{V}\}$ define smooth manifolds in \mathcal{X} . We are interested in the quotient manifold \mathcal{X} / \sim , which is isomorphic to $\mathcal{M}_{\mathbf{r}}$. This construction gives rise to an embedded analytic manifold [1, 47, 69] where a Riemannian metric is canonically defined.

The *tangent space* $\mathcal{T}_{\mathbf{u}}$ at $\mathbf{u} \in \mathcal{M}_{\mathbf{r}}$ is of importance for calculations. It can be easily determined by means of the product rule as follows. A generic tensor $\delta\mathbf{u} \in \mathcal{T}_{\mathbf{u}}$ of a TT tensor \mathbf{u} is of the form

$$\begin{aligned} \delta\mathbf{u}(\mu_1, \dots, \mu_d) &= \mathbf{t}^1(\mu_1, \dots, \mu_d) + \dots + \mathbf{t}^d(\mu_1, \dots, \mu_d) \\ &= \delta\mathbf{B}^1(\mu_1)\mathbf{B}^2(\mu_2)\cdots\mathbf{B}^d(\mu_d) + \dots \\ &\quad + \mathbf{B}^1(\mu_1)\cdots\mathbf{B}^{i-1}(\mu_{i-1})\delta\mathbf{B}^i(\mu_i)\mathbf{B}^{i+1}(\mu_{i+1})\cdots\mathbf{B}^d(\mu_d) + \dots \\ &\quad + \mathbf{B}^1(\mu_1)\cdots\mathbf{B}^{d-1}(\mu_{d-1})\delta\mathbf{B}^d(\mu_d). \end{aligned}$$

This tensor is uniquely determined if we impose *gauging conditions* onto $\delta\mathbf{B}^i$, $i = 1, \dots, d-1$ [34, 69]. There is no gauging condition imposed onto $\delta\mathbf{B}^d$. Typically these conditions are of the form

$$\sum_{k_{i-1}=1}^{r_{i-1}} \sum_{\mu_i=1}^{n_i} \mathbf{b}^i(k_{i-1}, \mu_i, k_i) \delta\mathbf{b}^i(k_{i-1}, \mu_i, k'_i) = 0, \quad \forall k_i, k'_i = 1, \dots, r_i. \quad (9)$$

With this condition at hand an orthogonal projection $P_{\mathcal{T}_{\mathbf{u}}}$ onto the tangent space $\mathcal{T}_{\mathbf{u}}$ is well defined and computable in a straightforward way.

The manifold $\mathcal{M}_{\mathbf{r}}$ is open and its closure is $\mathcal{M}_{\leq \mathbf{r}}$, the set of all tensors with ranks at most r_{α} , $\alpha \in \mathbb{T}$,

$$\text{clos}(\mathcal{M}_{\mathbf{r}}) = \mathcal{M}_{\leq \mathbf{r}}.$$

This important result is based on the observation that the matrix rank is an upper semi-continuous function [16]. The singular points are exactly those for which at least one rank $\tilde{r}_{\alpha} < r_{\alpha}$ is not maximal, see e.g [60]. We remark that, for the root D of the partition tree \mathbb{T} , there is no gauging condition imposed onto $\delta \mathbf{B}^D$. We highlight the following facts without explicit proofs for hierarchical tensors.

Proposition 2. *Let $\mathbf{u} \in \mathcal{M}_{\mathbf{r}}$. Then*

- (a) *the corresponding gauging conditions (9) imply that the tangential vectors \mathbf{t}^i are pairwise orthogonal;*
- (b) *the tensor \mathbf{u} is included in its own tangent space $\mathcal{T}_{\mathbf{u}}$;*
- (c) *the multi-linear rank of a tangent vector is at most $2\mathbf{r}$, i.e., $\delta \mathbf{u} \in \mathcal{M}_{\leq 2\mathbf{r}}$.*

Curvature estimates are given in [47].

3 Tensor completion for hierarchical tensors

3.1 The low rank tensor recovery problem

We pursue on extending methods for solving optimization problems in the calculus of hierarchical tensors to *low rank tensor recovery* and to *tensor completion* as a special case. The latter builds on ideas from the theory of *compressed sensing* [19], which predicts that sparse vectors can be recovered efficiently from incomplete linear measurements via efficient algorithms. Given a linear measurement mapping $\mathcal{A} : \mathcal{H}_d = \bigotimes_{i=1}^d \mathbb{R}^{n_i} \rightarrow \mathbb{R}^m$ our aim is to recover a tensor $\mathbf{u} \in \mathcal{H}_d$ from $m \ll N := n_1 \cdot n_2 \cdots n_d$ measurements $\mathbf{b} \in \mathbb{R}^m$ given by

$$\mathbf{b} = (b_i)_{i=1}^m = \mathcal{A} \mathbf{u}.$$

Since this problem is underdetermined we additionally assume that \mathbf{u} is of low rank, i.e., given a dimension tree \mathbb{T} and a multi-linear rank \mathbf{r} , we suppose that the tensor is contained in the corresponding tensor manifold, $\mathbf{u} \in \mathcal{M}_{\mathbf{r}}$.

The *tensor completion* problem – generalizing the matrix completion problem [9, 10, 25] – is the special case where the measurement map subsamples entries of the tensor, i.e., $b_i = (\mathcal{A} \mathbf{u})_i = \mathbf{u}(\boldsymbol{\mu}_i) = \mathbf{u}(\mu_{1,i}, \dots, \mu_{d,i})$, $i = 1, \dots, m$, with the multi-indices $\boldsymbol{\mu}_i$ being contained in a suitable index set $\Omega \subset [n_1] \times \cdots \times [n_d]$ of cardinality $m \ll n_1 \cdots n_d$.

We remark that in practice the desired rank \mathbf{r} may not be known in advance and/or the tensor \mathbf{u} is only close to $\mathcal{M}_{\mathbf{r}}$ rather than being exactly contained in $\mathcal{M}_{\mathbf{r}}$.

Moreover, the left hand side \mathbf{b} may not be known exactly because of noise on the measurements. In the present paper we defer from tackling these important stability and robustness issues and focus on the problem in the above form.

The problem of reconstructing $\mathbf{u} \in \mathcal{H}_d$ from $\mathbf{b} = \mathcal{A}\mathbf{u}$ can be reformulated to finding the minimizer of

$$\mathcal{J}(\mathbf{v}) = \frac{1}{2} \|\mathcal{A}\mathbf{v} - \mathbf{b}\|^2 \quad \text{subject to } \mathbf{v} \in \mathcal{M}_{\mathbf{r}}. \quad (10)$$

In words, we are looking for a tensor of multi-linear rank \mathbf{r} , which fits best the given measurements. A minimizer over $\mathcal{M}_{\leq \mathbf{r}}$ always exists, but a solution of the above problem may not exist in general since $\mathcal{M}_{\mathbf{r}}$ is not closed. However, assuming $\mathbf{u} \in \mathcal{M}_{\mathbf{r}}$ and $\mathbf{b} = \mathcal{A}\mathbf{u}$ as above, existence of a minimizer is trivial because setting $\mathbf{v} = \mathbf{u}$ gives $\mathcal{J}(\mathbf{v}) = 0$. We note that finding a minimizer of (10) is NP-hard in general [32, 20].

The necessary first order condition for a minimizer of the problem (10) can be formulated as follows, see e.g. [47]. If $\mathbf{u} \in \mathcal{M}_{\mathbf{r}}$ is a solution of $\operatorname{argmin}_{\mathbf{v} \in \mathcal{M}_{\mathbf{r}}} \mathcal{J}(\mathbf{v})$, then

$$\langle \nabla \mathcal{J}(\mathbf{u}), \delta \mathbf{u} \rangle = 0, \quad \text{for all } \delta \mathbf{u} \in \mathcal{T}_{\mathbf{u}},$$

where $\nabla \mathcal{J}$ is the gradient of \mathcal{J} .

3.2 Optimization approaches

In analogy to compressed sensing and low rank matrix recovery where convex relaxations (ℓ_1 -minimization and nuclear norm minimization) are very successful, a first idea for a tractable alternative to (10) may be to find an analogue of the nuclear norm for the tensor case. A natural approach is to consider the set Q of unit norm rank one tensors in \mathcal{H}_d ,

$$Q = \{\mathbf{u} = \mathbf{b}_1 \otimes \mathbf{b}_2 \cdots \otimes \mathbf{b}_d : \|\mathbf{u}\| = 1\}.$$

Its closed convex hull $B = \overline{\operatorname{conv} Q}$ is taken as the unit ball of the tensor nuclear norm, so that the tensor nuclear norm is the gauge function of B ,

$$\|\mathbf{u}\|_* = \inf\{t : \mathbf{u} \in tB\}.$$

In fact, for the matrix case $d = 2$ we obtain the standard nuclear norm. Unfortunately, for $d \geq 3$, the nuclear tensor norm is NP hard to compute [32, 20].

The contributions [22, 36, 44] proceed differently by considering the matrix nuclear norm of several unfoldings of the tensor \mathbf{u} . Given a dimension tree $\alpha \in \mathbb{T}$ and corresponding matricisations \mathbf{U}^α of \mathbf{u} , we consider the Schatten norm

$$\|\mathbf{U}^\alpha\|_p := \left(\sum_{k_\alpha} (\sigma_{k_\alpha}^\alpha)^p \right)^{\frac{1}{p}}, \quad \alpha \in \mathbb{T}, \quad 1 \leq p < \infty,$$

where the $\sigma_{k_\alpha}^\alpha$ are the singular values of \mathbf{U}^α , see e.g. [6, 59]. Furthermore, for $1 \leq q \leq \infty$ and given $a_\alpha > 0$, e.g. $a_\alpha = 1$, a norm on \mathcal{H}_d can be introduced by

$$\|\mathbf{u}\|_{p,q}^q := \sum_{\alpha \in \mathbb{T}} a_\alpha \|\mathbf{U}^\alpha \mathbf{u}\|_p^q.$$

A prototypical choice of a convex optimization formulation used for tensor recovery consists in finding

$$\operatorname{argmin}\{\mathcal{J}(\mathbf{u}) := \|\mathbf{u}\|_{1,q} : \mathcal{A}\mathbf{u} = \mathbf{b}\}.$$

For $a_\alpha = 1$ and $q = 1$ this functional was suggested in [22, 44] for reconstructing tensors in the Tucker format. Although the numerical results are reasonable, it seems that conceptually this is not the “right” approach and too simple for the present purpose, see corresponding negative results in [48]. For the Tucker case first rigorous results have been shown in [36]. However the approach followed there is based on results from matrix completion and does not use the full potential of tensor decompositions. In fact, their bound on the number of required measurements is $\mathcal{O}(rn^{d-1})$. A more “balanced” version of this approach is considered in [48], where the number of required measurements scales like $\mathcal{O}(r^{d/2}n^{d/2})$, which is better but still far from the expected linear scaling in n , see also Theorem 2 below.

3.3 Iterative hard thresholding schemes

Rather than following the convex optimization approach which leads to certain difficulties as outlined above, we consider versions of the iterative hard thresholding algorithm well-known from compressed sensing [8, 19] and low rank matrix recovery [64]. Iterative hard thresholding algorithms fall into the larger class of *projected gradient methods*. Typically one performs a gradient step in the ambient space \mathcal{H}_d , followed by a mapping \mathcal{R} onto the set of low rank tensors \mathcal{M}_r or $\mathcal{M}_{\leq r}$, formally

$$\begin{aligned} \mathbf{y}^{n+1} &:= \mathbf{u}^n - \alpha_n \nabla \mathcal{J}(\mathbf{u}^n) \quad (\text{gradient step}) \\ &= \mathbf{u}^n - \alpha_n (\mathcal{A}^* (\mathcal{A}\mathbf{u}^n - \mathbf{b})), \\ \mathbf{u}^{n+1} &:= \mathcal{R}(\mathbf{y}^{n+1}) \quad (\text{projection step}). \end{aligned}$$

Apart from specifying the steplength α_n , the above algorithm depends on the choice of the projection operator $\mathcal{R} : \mathcal{H}_d \rightarrow \mathcal{M}_{\leq r}$. An example would be

$$\mathcal{R}(\mathbf{y}^{n+1}) := \operatorname{argmin}\{\|\mathbf{y}^{n+1} - \mathbf{z}\| : \mathbf{z} \in \mathcal{M}_{\leq r}\}.$$

Since this projection is not computable in general [32, 20], we may rather choose the hierarchical singular value (HSVD) thresholding procedure (7)

$$\mathbf{u}^{n+1} := \mathcal{R}(\mathbf{y}^{n+1}) = \mathbf{H}_r(\mathbf{y}^{n+1}) \quad (\text{hard thresholding}),$$

which is only quasi-optimal (8). We will call this procedure *tensor iterative hard thresholding (TIHT)*, or shortly *iterative hard thresholding (IHT)*.

Another possibility for the projection operator relies on the concept of retraction from differential geometry [1]. A retraction maps $\mathbf{u} + \xi$, where $\mathbf{u} \in \mathcal{M}_{\mathbf{r}}$ and $\xi \in \mathcal{T}_{\mathbf{u}}$, smoothly to the manifold. For $R : (\mathbf{u}, \xi) \rightarrow R(\mathbf{u}, \xi) \in \mathcal{M}_{\mathbf{r}}$ being a retraction it is required that R is twice differentiable and $R(\cdot, \mathbf{0}) = \mathbf{I}$ is the identity. Moreover, a retraction satisfies, for $\|\xi\|$ sufficiently small,

$$\begin{aligned} \|\mathbf{u} + \xi - R(\mathbf{u}, \xi)\| &= \mathcal{O}(\|\xi\|^2), \\ \|\mathbf{u} - R(\mathbf{u}, \xi)\| &= \mathcal{O}(\|\xi\|). \end{aligned} \quad (11)$$

Several examples of retractions for hierarchical tensors are known [47, 40], which can be efficiently computed. If a retraction is available, then a nonlinear projection \mathcal{R} can be realized in two steps. First we project (linearly) onto the tangent space $\mathcal{T}_{\mathbf{u}^n}$ at \mathbf{u}^n , and afterwards we apply a retraction R . This leads to the so-called *Riemaniann gradient iteration method (RGI)* defined formally as

$$\begin{aligned} \mathbf{z}^{n+1} &:= P_{\mathcal{T}_{\mathbf{u}^n}}(\mathbf{u}^n - \alpha_n P_{\mathcal{T}_{\mathbf{u}^n}}(\mathcal{A}^*(\mathcal{A}\mathbf{u}^n - \mathbf{b}))) \quad (\text{projected gradient step}) \\ &= P_{\mathcal{T}_{\mathbf{u}^n}}(\mathbf{u}^n - \alpha_n \mathcal{A}^*(\mathcal{A}\mathbf{u}^n - \mathbf{b})) =: \mathbf{u}^n + \xi^n \\ \mathbf{u}_{n+1} &:= R(\mathbf{u}^n, \mathbf{z}^{n+1} - \mathbf{u}^n) = R(\mathbf{u}^n, \xi^n) \quad (\text{retraction step}). \end{aligned}$$

With a slight abuse of notation we will write

$$\mathcal{R}(\mathbf{y}^{n+1}) = R \circ P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{y}^{n+1}$$

for the RGI.

It may happen that an iterate \mathbf{u}^n is of lower rank, i.e., $\mathbf{u}^n \in \mathcal{M}_s$ with $s_\alpha < r_\alpha$ at least for one $\alpha \in \mathbb{T}$. In this case $\mathbf{u}^n \in \mathcal{M}_{\leq \mathbf{r}}$ is a singular point and no longer on our manifold, i.e., $\mathbf{u}^n \notin \mathcal{M}_{\mathbf{r}}$, and our RGI algorithm fails. However, since $\mathcal{M}_{\mathbf{r}}$ is dense in $\mathcal{M}_{\leq \mathbf{r}}$, for arbitrary $\varepsilon > 0$, there exists $\mathbf{u}_\varepsilon^n \in \mathcal{M}_{\mathbf{r}}$, with $\|\mathbf{u}^n - \mathbf{u}_\varepsilon^n\| < \varepsilon$. Practically such a regularized \mathbf{u}_ε^n is not hard to choose. Alternatively, the algorithm described above may be regularized in a sense that it automatically avoids the situation being trapped in a singular point [40]. Here, we do not go into these technical details.

3.4 Restricted isometry property for hierarchical tensors

A crucial sufficient condition for exact recovery in compressed sensing is the *restricted isometry property (RIP)*, see Chapter 1. It has been applied in the analysis of iterative hard thresholding both in the compressed sensing setting [8, 19] as well as in the low rank matrix recovery setting [64]. The RIP can be easily generalized to the present tensor setting. As common, $\|\cdot\|$ denotes the Euclidean norm below.

Definition 4. Let $\mathcal{A} : \mathcal{H}_d = \bigotimes_{i=1}^d \mathbb{R}^{n_i} \rightarrow \mathbb{R}^m$ be a linear measurement map, \mathbb{T} be a dimension tree and for $\mathbf{r} = (r_\alpha)_{\alpha \in \mathbb{T}}$, let $\mathcal{M}_{\mathbf{r}}$ be the associated low rank tensor

manifold. The tensor restricted isometry constant (TRIC) $\delta_{\mathbf{r}}$ of \mathcal{A} is the smallest number such that

$$(1 - \delta_{\mathbf{r}})\|\mathbf{u}\|^2 \leq \|\mathcal{A}\mathbf{u}\|^2 \leq (1 + \delta_{\mathbf{r}})\|\mathbf{u}\|^2, \quad \text{for all } \mathbf{u} \in \mathcal{M}_{\mathbf{r}}. \quad (12)$$

Informally, we say that a measurement map \mathcal{A} satisfies the *tensor restricted isometry property* (TRIP) if $\delta_{\mathbf{r}}$ is small (at least $\delta_{\mathbf{r}} < 1$) for some “reasonably large” \mathbf{r} .

Observing that $\mathbf{u} + \mathbf{v} \in \mathcal{M}_{\leq 2\mathbf{r}}$ for two tensors $\mathbf{u}, \mathbf{v} \in \mathcal{M}_{\leq \mathbf{r}}$ the TRIP (12) of order $2\mathbf{r}$ implies that \mathcal{J} has a unique minimizer on $\mathcal{M}_{\mathbf{r}}$. Indeed, for two tensors $\mathbf{u}_1, \mathbf{u}_2 \in \mathcal{M}_{\mathbf{r}}$ satisfying $\mathcal{A}\mathbf{u}_1 - \mathbf{b} = \mathcal{A}\mathbf{u}_2 - \mathbf{b} = \mathbf{0}$, it follows that $\mathcal{A}(\mathbf{u}_1 - \mathbf{u}_2) = \mathbf{0}$ in contradiction to the TRIP and $\mathbf{u}_1 - \mathbf{u}_2 \in \mathcal{M}_{\leq 2\mathbf{r}}$.

For Gaussian (or more generally subgaussian) measurement maps, the TRIP holds with high probability for both the HOSVD and the TT format under a suitable bound on the number of measurements [53, 54], which basically scales like the number of degrees of freedom of a tensor of multi-linear rank \mathbf{r} (up to a logarithmic factor in d). In order to state these results, let us introduce Gaussian measurement maps. A measurement map $\mathcal{A} : \mathcal{H}_d \rightarrow \mathbb{R}^m$ can be identified with a tensor in $\mathbb{R}^m \otimes \bigotimes_{i=1}^d \mathbb{R}^{n_i}$ via

$$(\mathcal{A}\mathbf{u})_{\ell} = \sum_{\mu_1=1}^{n_1} \sum_{\mu_2=1}^{n_2} \cdots \sum_{\mu_d=1}^{n_d} \mathbf{a}(\ell, \mu_1, \mu_2, \dots, \mu_d) \mathbf{u}(\mu_1, \dots, \mu_d), \quad \ell = 1, \dots, m.$$

If all entries of \mathcal{A} are independent realizations of normal distributed random variables with mean zero and variance $1/m$, then \mathcal{A} is called a *Gaussian measurement map*.

Theorem 2 ([53, 54]). *For $\delta, \varepsilon \in (0, 1)$, a random draw of a Gaussian measurement map $\mathcal{A} : \mathcal{H}_d = \bigotimes_{i=1}^d \mathbb{R}^{n_i} \rightarrow \mathbb{R}^m$ satisfies $\delta_{\mathbf{r}} \leq \delta$ with probability at least $1 - \varepsilon$ provided*

- *HOSVD format:* $m \geq C\delta^{-2} \max \{ (r^d + dnr) \log(d), \log(\varepsilon^{-1}) \}$,
- *TT format:* $m \geq C\delta^{-2} \max \{ (dnr^2) \log(dr), \log(\varepsilon^{-1}) \}$,

where $n = \max \{ n_i : i = 1, \dots, d \}$ and $r = \max \{ r_i : i = 1, \dots, d \}$ and $C > 0$ is a universal constant.

The above result extends to subgaussian and in particular to Bernoulli measurement maps, see e.g. [19, 71] for the definition of subgaussian random variables and matrices. Presently, it is not clear whether the logarithmic factor in d above is necessary or whether it is an artefact of the proof. We conjecture that similar bounds hold also for the general hierarchical tensor format. We note that in practice, the application of Gaussian sensing operators acting on the tensor product space \mathcal{H}_d seems to be computationally too expensive except for relatively small dimensions d , (e.g. $d = 2, 3, 4$), and small n_i . A more realistic measurement map for which TRIP bounds can be shown [54] is the decomposition of random sign flips of the tensor entries, a d -dimensional Fourier transform and random subsampling. All these operations can be performed quickly (exploiting) the FFT. For further details we refer to [53, 54].

We finally remark that the TRIP does not hold in the tensor completion setup because sparse and low rank tensor may belong to the kernel of the measurement map. In this scenario, additional incoherence properties on the tensor to be recovered like in the matrix completion scenario [9, 25, 56] are probably necessary.

3.5 Convergence results

Unfortunately, a full convergence (and recovery) analysis of the TIHT and RGI algorithms under the TRIP is not yet available. Nevertheless, we present two partial results. The first concerns the local convergence of the RGI and the second is a convergence analysis of the TIHT under an additional assumption on the iterates.

We assume that $\mathbf{u} \in \mathcal{M}_{\mathbf{r}}$, where the low rank tensor manifold is associated to a fixed hierarchical tensor format. Measurements are given by

$$\mathbf{b} = \mathcal{A}\mathbf{u},$$

where \mathcal{A} is assumed to satisfy the TRIP of order $3\mathbf{r}$ below. Recall that our projected gradient scheme starts with an initial guess \mathbf{u}^0 and forms the iterates

$$\mathbf{y}^{n+1} := \mathbf{u}^n + \mathcal{A}^*(\mathbf{b} - \mathcal{A}\mathbf{u}^n), \quad (13)$$

$$\mathbf{u}^{n+1} := \mathcal{R}(\mathbf{y}^{n+1}), \quad (14)$$

where either $\mathcal{R}(\mathbf{u}^{n+1}) := \mathbf{H}_{\mathbf{r}}(\mathbf{y}^{n+1})$ (TIHT) or $\mathcal{R}(\mathbf{u}^{n+1}) := R \circ P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{y}^{n+1}$ (RGI).

We first show local convergence, in the sense that the iterates \mathbf{u}^n converge to the original tensor \mathbf{u} if the initial guess is sufficiently close to the solution $\mathbf{u} \in \mathcal{M}_{\mathbf{r}}$. Of course, this analysis also applies if one of the later iterates comes close enough to \mathbf{u} .

Theorem 3 (Local convergence). *Let $\mathbf{b} = \mathcal{A}\mathbf{u}$ for $\mathbf{u} \in \mathcal{M}_{\leq \mathbf{r}}$ and let \mathbf{u}^n be the iterates (13), (14) of the Riemannian gradient iterations, i.e., $\mathcal{R}(\mathbf{u}^{n+1}) := R \circ P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{y}^{n+1}$, where R is a retraction. In addition, let's assume that \mathcal{A} satisfies the TRIP of order $3\mathbf{r}$, i.e., $\delta_{3\mathbf{r}} \leq \delta < 1$. Suppose that*

$$\|\mathbf{u} - \mathbf{u}^0\| \leq \varepsilon$$

is sufficiently small and the distance to the singular points $\varepsilon < \text{dist}(\mathbf{u}^0, \partial \mathcal{M}_{\mathbf{r}})$ is sufficiently large. Then, there exists $0 < \rho < 1$ (depending on δ and ε) such that the series $\mathbf{u}^n \in \mathcal{M}_{\leq \mathbf{r}}$ converges linearly to $\mathbf{u} \in \mathcal{M}_{\leq \mathbf{r}}$ with rate ρ ,

$$\|\mathbf{u}^{n+1} - \mathbf{u}\| \leq \rho \|\mathbf{u}^n - \mathbf{u}\|.$$

Proof. We consider the orthogonal projection $P_{\mathcal{T}_{\mathbf{u}^n}}$ onto the tangent space $\mathcal{T}_{\mathbf{u}^n}$. There exists $1 < \gamma = \gamma(\varepsilon)$ and $\kappa > 0$ depending on the curvature of $\mathcal{M}_{\mathbf{r}}$, such that, for all $\|\mathbf{v} - \mathbf{u}^n\| < \varepsilon$, it holds that [47]

$$\gamma^{-1} \|\mathbf{u}^n - \mathbf{v}\| \leq \|P_{\mathcal{T}_{\mathbf{u}^n}}(\mathbf{u}^n - \mathbf{v})\| \leq \gamma \|\mathbf{u}^n - \mathbf{v}\| \quad (15)$$

$$\|(I - P_{\mathcal{T}_{\mathbf{u}^n}})(\mathbf{u}^n - \mathbf{v})\| \leq \kappa \|P_{\mathcal{T}_{\mathbf{u}^n}}(\mathbf{u} - \mathbf{v})\|^2. \quad (16)$$

Using the triangle inequality we estimate

$$\begin{aligned} \|\mathbf{u}^{n+1} - \mathbf{u}\| &= \|R(\mathbf{u}^n, P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{y}^{n+1} - \mathbf{u}^n) - \mathbf{u}\| \\ &\leq \|R(\mathbf{u}^n, P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{y}^{n+1} - \mathbf{u}^n) - P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{y}^{n+1}\| + \|P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{y}^{n+1} - P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{u}\| + \|P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{u} - \mathbf{u}\|. \end{aligned} \quad (17)$$

We will bound each of the three terms in (17) separately. We start with the first term, where we exploit the property (11) of retractions. Moreover, $W^n := \mathcal{T}_{\mathbf{u}^n} \subset \mathcal{M}_{\leq 2\mathbf{r}}$ by Proposition 2(c) and $P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{y}^{n+1} = P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{u}^n + P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{u}^n \mathcal{A}^* \mathcal{A}(\mathbf{u} - \mathbf{u}^n) = \mathbf{u}^n + P_{\mathcal{T}_{\mathbf{u}^n}} \mathcal{A}^* \mathcal{A}(\mathbf{u} - \mathbf{u}^n)$ because $P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{u}^n = \mathbf{u}^n$ by Proposition 2(a). Since $\mathbf{u} - \mathbf{u}^n \in \mathcal{M}_{\leq 2\mathbf{r}}$ we may apply the TRIP of order $2\mathbf{r} < 3\mathbf{r}$ to obtain

$$\begin{aligned} \|R(\mathbf{u}^n, P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{y}^{n+1} - \mathbf{u}^n) - P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{y}^{n+1}\| &\leq C \|P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{y}^{n+1} - \mathbf{u}^n\|^2 \\ &= C \|P_{\mathcal{T}_{\mathbf{u}^n}} \mathcal{A}^* \mathcal{A}(\mathbf{u} - \mathbf{u}^n)\|^2 \leq C(1 + \delta_{3\mathbf{r}})^2 \varepsilon \|\mathbf{u} - \mathbf{u}^n\|, \end{aligned} \quad (18)$$

where in the last estimate we also used that $\|\mathbf{u} - \mathbf{u}^n\| \leq \varepsilon$.

For the second term in (17) observe that $(I - P_{\mathcal{T}_{\mathbf{u}^n}})(\mathbf{u} - \mathbf{u}^n) = \mathbf{u} - P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{u} \in \mathcal{M}_{\leq 3\mathbf{r}}$ by Proposition 2. The TRIP implies therefore that the spectrum of $P_{\mathcal{T}_{\mathbf{u}^n}}(I - \mathcal{A}^* \mathcal{A})|_{W^n}$ is contained in the interval $[-\delta_{3\mathbf{r}}, \delta_{3\mathbf{r}}]$. With these observations and (16) we obtain

$$\begin{aligned} \|P_{\mathcal{T}_{\mathbf{u}^n}}(\mathbf{y}^{n+1} - \mathbf{u})\| &= \|P_{\mathcal{T}_{\mathbf{u}^n}}((\mathbf{u} - \mathbf{u}^n) - \mathcal{A}^*(\mathbf{b} - \mathcal{A}\mathbf{u}^n))\| \\ &\leq \|P_{\mathcal{T}_{\mathbf{u}^n}}((\mathbf{u} - \mathbf{u}^n) - \mathcal{A}^* \mathcal{A}(P_{\mathcal{T}_{\mathbf{u}^n}}(\mathbf{u} - \mathbf{u}^n)))\| \\ &\quad + \|P_{\mathcal{T}_{\mathbf{u}^n}} \mathcal{A}^* \mathcal{A}((I - P_{\mathcal{T}_{\mathbf{u}^n}})(\mathbf{u} - \mathbf{u}^n))\| \\ &\leq \|P_{\mathcal{T}_{\mathbf{u}^n}}((\mathbf{u} - \mathbf{u}^n) - \mathcal{A}^*(\mathcal{A}P_{\mathcal{T}_{\mathbf{u}^n}}(\mathbf{u} - \mathbf{u}^n)))\| \\ &\quad + (1 + \delta_{3\mathbf{r}}) \|(I - P_{\mathcal{T}_{\mathbf{u}^n}})(\mathbf{u} - \mathbf{u}^n)\| \\ &\leq \|P_{\mathcal{T}_{\mathbf{u}^n}}(I - \mathcal{A}^* \mathcal{A}P_{\mathcal{T}_{\mathbf{u}^n}})(\mathbf{u} - \mathbf{u}^n)\| \\ &\quad + (1 + \delta_{3\mathbf{r}}) \kappa \|P_{\mathcal{T}_{\mathbf{u}^n}}(\mathbf{u} - \mathbf{u}^n)\|^2 \\ &\leq \delta_{3\mathbf{r}} \|P_{\mathcal{T}_{\mathbf{u}^n}}(\mathbf{u} - \mathbf{u}^n)\| + (1 + \delta_{3\mathbf{r}}) \kappa \|P_{\mathcal{T}_{\mathbf{u}^n}}(\mathbf{u} - \mathbf{u}^n)\|^2. \end{aligned}$$

Hence, for ε sufficiently small, there exists a factor $0 < \tilde{\rho} < 1$ such that

$$\|P_{\mathcal{T}_{\mathbf{u}^n}}(\mathbf{y}^{n+1} - \mathbf{u})\| \leq \tilde{\rho} \|P_{\mathcal{T}_{\mathbf{u}^n}}(\mathbf{u} - \mathbf{u}^n)\|. \quad (19)$$

For the third term in (17), first notice that by the Pythagorean theorem

$$\|\mathbf{u}^n - \mathbf{u}\|^2 = \|\mathbf{u}^n - P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{u}\|^2 + \|P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{u} - \mathbf{u}\|^2.$$

Since $P_{\mathcal{T}_{\mathbf{u}^n}} \mathbf{u}^n = \mathbf{u}^n$, using (15) one obtains

$$\|P_{\mathcal{F}_{\mathbf{u}^n}} \mathbf{u} - \mathbf{u}\| \leq \sqrt{1 - \gamma^{-2}} \|\mathbf{u}^n - \mathbf{u}\|. \quad (20)$$

Combining the estimates (18), (19) and (20) yields

$$\|\mathbf{u}^{n+1} - \mathbf{u}\| \leq \rho \|\mathbf{u} - \mathbf{u}^n\|,$$

where $\rho = C(1 + \delta_{3\mathbf{r}}) \varepsilon + \tilde{\rho} + \sqrt{1 - \gamma^{-2}} < 1$ for ε and κ small and γ close enough to 1. Consequently, the sequence \mathbf{u}^n converges linearly to \mathbf{u} .

The weak point of the previous theorem is that this convergence can be guaranteed only in a very narrow neighborhood of the solution. To obtain global convergence, now for TIHT, we ask for an additional assumption on the iterates.

Theorem 4 (Conditionally global convergence). *Let $\mathbf{b} = \mathcal{A} \mathbf{u}$ for $\mathbf{u} \in \mathcal{M}_{\leq \mathbf{r}}$ and let \mathbf{u}^n be the iterates (13), (14) of the tensor iterative hard thresholding algorithm, i.e., $\mathcal{R}(\mathbf{u}^{n+1}) := \mathbf{H}_{\mathbf{r}}(\mathbf{y}^{n+1})$. In addition, let's assume that \mathcal{A} satisfies the TRIP of order $3\mathbf{r}$, i.e., $\delta_{3\mathbf{r}} \leq 1/2$. We further assume that the iterates satisfy, for all $n \in \mathbb{N}$,*

$$\|\mathbf{u}^{n+1} - \mathbf{y}^{n+1}\| \leq \|\mathbf{u} - \mathbf{y}^{n+1}\|. \quad (21)$$

Then the sequence $\mathbf{u}^n \in \mathcal{M}_{\leq \mathbf{r}}$ converges linearly to a unique solution $\mathbf{u} \in \mathcal{M}_{\leq \mathbf{r}}$ with rate $\rho < 1$, i.e.,

$$\|\mathbf{u}^{n+1} - \mathbf{u}\| \leq \rho \|\mathbf{u}^n - \mathbf{u}\|.$$

For details of the proof, we refer to [54]. We note that the above result can be extended to robustness under noise on the measurements and to tensors being only approximately of low rank, i.e., being close to $\mathcal{M}_{\mathbf{r}}$ but not necessarily on $\mathcal{M}_{\mathbf{r}}$.

Let us comment on the essential condition (21). In the case that \mathcal{R} computes the best rank \mathbf{r} approximation then this condition holds since

$$\inf\{\|\mathbf{v} - \mathbf{y}^{n+1}\| : \mathbf{v} \in \mathcal{M}_{\leq \mathbf{r}}\} \leq \|\mathbf{u} - \mathbf{y}^{n+1}\|$$

is trivially true. However, the best approximate is not numerically available and the truncated HSVD only ensures the worst case error estimate

$$\|\mathbf{u}^{n+1} - \mathbf{y}^{n+1}\| \leq C(d) \inf\{\|\mathbf{v} - \mathbf{y}^{n+1}\| : \mathbf{v} \in \mathcal{M}_{\leq \mathbf{r}}\}.$$

Nevertheless, in practice this bound may be pessimistic for a generic tensor so that (21) may be likely to hold. In any case, the above theorem may at least *explain* why we observe recovery by TIHT in practice.

3.6 Alternating least squares scheme (ALS)

An efficient and fairly simple method for computing (at least a local) minimizer of $\|\mathbf{u} - \mathbf{v}\|$ subject to $\mathbf{v} \in \mathcal{M}_{\mathbf{r}}$ is based on *alternating least squares (ALS)*, which is a

variant of block Gauß-Seidel optimization. In contrast to poor convergence experienced with the canonical format (CANDECOMP, PARAFAC) [38], ALS implemented appropriately in the hierarchical formats has been observed to be surprisingly powerful [35]. Furthermore, and quite importantly, it is robust with respect to overfitting, i.e., allows optimization in the set $\mathcal{M}_{\leq \mathbf{r}}$ [35]. As a local optimization scheme, like the Riemannian optimization it converges only to a local minimum at best. This scheme applied to TT tensors is basically a one-site DMRG (density matrix renormalization group) algorithm introduced for quantum spin systems in [75, 61]. The basic idea for computing $\mathbf{u} = \tau(\{\mathbf{b}_\alpha : \alpha \in \mathbb{T}\})$ by the ALS or block Gauß-Seidel method is to compute the required components \mathbf{b}_α , one after each other. Fixing the components \mathbf{b}_α , $\alpha \in \mathbb{T}$, $\alpha \neq t$, only the component \mathbf{b}_t is left to be optimized in each iteration step. Before passing to the next iteration step, the new iterate has to be transformed into the normal form \mathbf{b}_t^{n+1} by orthogonalization e.g. by applying an SVD (without truncation) or simply by QR factorization.

Let us assume that the indices $\alpha \in \mathbb{T}$ are in a linear ordering $<$, which is consistent with the hierarchy. The components given by the present iterate are denoted by \mathbf{b}_α^n , if $\alpha \geq t$ respectively, \mathbf{b}_α^{n+1} for $\alpha < t$. We optimize over \mathbf{b}_t and introduce a corresponding tensor by

$$\mathbf{u}_{-t}^{n+1} := \tau(\{\mathbf{b}_\alpha^{n+1} : \alpha < t\} \cup \{\mathbf{b}_t\} \cup \{\mathbf{b}_\alpha^n : \alpha > t\}) \in \mathcal{H}_d.$$

Since the parametrization $\tau(\{\mathbf{b}_\alpha : \alpha \in \mathbb{T}\}) \in \mathcal{M}_{\leq \mathbf{r}}$ is multi-linear in its arguments \mathbf{b}_α , the map τ_t^{n+1} defined by $\mathbf{b}_t \mapsto \tau_t^{n+1}(\mathbf{b}_t) := \mathbf{u}_{-t}^{n+1}$ is linear. The first order optimality condition for the present minimization is

$$0 = \nabla \mathcal{J} \circ \tau_t^n(\mathbf{b}_t) = (\tau_t^n)^* \mathcal{A}^* (\mathcal{A} \tau_t^n(\mathbf{b}_t) - \mathbf{b}),$$

which constitutes a linear equation for the unknown \mathbf{b}_t . It is not hard to show the following facts.

Theorem 5. *1. Suppose that \mathcal{A} satisfies the TRIP of order \mathbf{r} with TRIC $\delta_{\mathbf{r}} < 1$ and that $\{\mathbf{b}_\alpha : \alpha \neq t\}$ is orthogonalized as above. Then, since $\|\tau_t^n(\mathbf{b}_t)\| = \|\mathbf{b}_t\|$, the TRIP reads as*

$$(1 - \delta_{\mathbf{r}})\|\mathbf{b}_t\|^2 \leq \|\mathcal{A} \tau_t^n(\mathbf{b}_t)\|^2 \leq (1 + \delta_{\mathbf{r}})\|\mathbf{b}_t\|^2.$$

In addition, the functional $\mathcal{J} \circ \tau_t^n$ is strictly convex, and $\mathcal{J} \circ \tau_t^n$ possesses a unique minimizer \mathbf{b}_t^n .

- 2. For $\mathbf{u}_{-t}^n := \tau_t^n(\mathbf{b}_t^n)$, the sequence $\mathbf{J}(\mathbf{u}_{-t}^n)$ is nonincreasing with n , and it is decreasing unless \mathbf{u}_{-t}^n is a stationary point, i.e., $\nabla \mathbf{J}(\mathbf{u}_{-t}^n) \perp \mathcal{T}_{\mathbf{u}_{-t}^n}$: In the latter case the algorithm stagnates.*
- 3. The sequence of iterates \mathbf{u}_{-t}^n is uniformly bounded.*

This result implies at least the existence of a convergent subsequence. However, no conclusion can be drawn whether this algorithm recovers the original low rank tensor \mathbf{u} from $\mathbf{b} = \mathcal{A}\mathbf{u}$. For further convergence analysis of ALS, we refer e.g. to [57].

In [77] convergence of a Block Gauß Seidel method was shown by means of the Lojasiewicz-Kurtyka inequality. Also nonnegative tensor completion has been discussed there. It is likely that these arguments apply also to the present setting. The ALS is simplified if one rearranges the tree in each micro-iteration step such that one optimizes always the root. This can be easily done for TT tensors with left and right-orthogonalization [34, 35], and can be modified for general hierarchical tensors as well. Often, it is preferable to proceed in an opposite order after the optimization of all components (half-sweep). For the Gauss–Southwell variant, where one optimizes the component with the largest defect, convergence estimates from gradient based methods can be applied [60]. Although the latter method converges faster, one faces a high computational overhead.

Let us remark that the Block Gauß-Seidel method and ALS strategy can be used in various situations, in particular, as one ingredient in the TIHT and RGI algorithms from the previous section. For instance, ALS can be applied directly after a gradient step defining the operator \mathcal{R} or one can use a simple half-sweep for approximating the gradient correction \mathbf{y}^{n+1} by a rank \mathbf{r} tensor in order to define the nonlinear projection \mathcal{R} .

4 Numerical results

For numerical tests, we concentrate on the HOSVD and the tensor iterative hard-thresholding (TIHT) algorithm for recovering order $d = 3$ tensors from Gaussian measurement maps $\mathcal{A} : \mathcal{H}_3 = \bigotimes_{i=1}^3 \mathbb{R}^{n_i} \rightarrow \mathbb{R}^m$, i.e., the entries of \mathcal{A} identified with a tensor in $\mathbb{R}^m \otimes \bigotimes_{i=1}^3 \mathbb{R}^{n_i}$ are i.i.d. $\mathcal{N}(0, \frac{1}{m})$ random variables.

For these tests, we generate tensors $\mathbf{u} \in \mathcal{H}_3$ of rank $\mathbf{r} = (r_1, r_2, r_3)$ via its Tucker decomposition. Let us suppose that

$$\mathbf{u}(\mu_1, \mu_2, \mu_3) = \sum_{k_1=1}^{r_1} \sum_{k_2=1}^{r_2} \sum_{k_3=1}^{r_3} \mathbf{c}(k_1, k_2, k_3) \mathbf{b}_{k_1}^1(\mu_1) \mathbf{b}_{k_2}^2(\mu_2) \mathbf{b}_{k_3}^3(\mu_3)$$

is the corresponding Tucker decomposition. Each entry of the core tensor is taken independently from the normal distribution, $\mathcal{N}(0, 1)$, and the component tensors $\mathbf{b}^j \in \mathbb{R}^{n_j \times r_j}$ are the first r_j left singular vectors of a matrix $\mathbf{M}^j \in \mathbb{R}^{n_j \times n_j}$ whose elements are also drawn independently from the normal distribution $\mathcal{N}(0, 1)$.

We then form the measurements $\mathbf{b} = \mathcal{A}\mathbf{u}$ and run the TIHT algorithm with the specified multi-linear rank $\mathbf{r} = (r_1, r_2, r_3)$ on \mathbf{b} . We test whether the algorithm successfully reconstructs the original tensor and say that the algorithm converged if $\|\mathbf{u} - \hat{\mathbf{u}}\| < 10^{-4}$. We stop the algorithm if it did not converge after 5000 iterations.

Figures 1–3 present the recovery results for low rank tensors of size $10 \times 10 \times 10$ (Figures 1 and Figure 2) and $6 \times 10 \times 15$ (Figure 3). The horizontal axis represents the number of measurements taken with respect to the number of degrees of freedom of an arbitrary tensor of this size. To be more precise, for a tensor of size $n_1 \times n_2 \times n_3$, the number \bar{n} on the horizontal axis represents $m = \lceil n_1 n_2 n_3 \frac{\bar{n}}{100} \rceil$ measurements. The

vertical axis represents the percentage of the successful recovery. For fixed tensor dimensions $n_1 \times n_2 \times n_3$, fixed HOSVD-rank $\mathbf{r} = (r_1, r_2, r_3)$ and fixed number of measurements m , we performed 200 simulations.

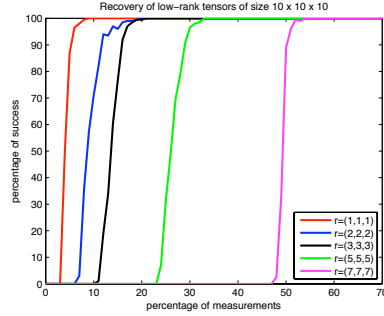


Fig. 1: Numerical results for $10 \times 10 \times 10$ tensors with same k -ranks.

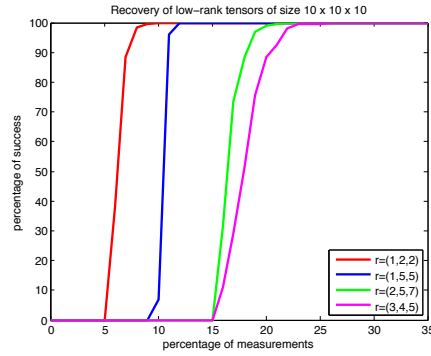


Fig. 2: Numerical results for $10 \times 10 \times 10$ tensors with different k -ranks.

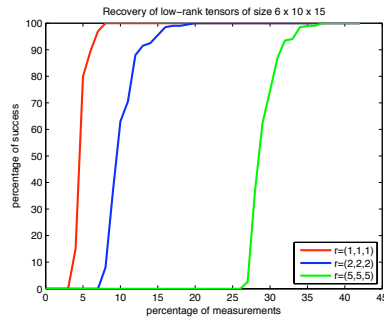


Fig. 3: Numerical results for $6 \times 10 \times 15$ tensors with same k -ranks.

Fig. 4: Numerical results for tensor IHT algorithm.

Table 1 complements Figure 4. With $\%_{\max}$ we denote the maximal percentage of measurements for which we did not manage to recover even one tensor out of 200. The minimal percentage of measurements for full recovery is denoted by $\%_{\min}$. The last column represents the number of iterations needed for full recovery with $m = \lceil n_1 n_2 n_3 \frac{\%_{\min}}{100} \rceil$ number of measurements.

Table 1: Numerical results for tensor IHT algorithm.

$n_1 \times n_2 \times n_3$	rank	$\%_{\max}$	$\%_{\min}$	# of iterations for $\%_{\min}$
$10 \times 10 \times 10$	(1, 1, 1)	3	9	321
$10 \times 10 \times 10$	(2, 2, 2)	6	20	185
$10 \times 10 \times 10$	(3, 3, 3)	10	21	337
$10 \times 10 \times 10$	(5, 5, 5)	23	33	547
$10 \times 10 \times 10$	(7, 7, 7)	47	54	1107
$10 \times 10 \times 10$	(1, 2, 2)	5	10	588
$10 \times 10 \times 10$	(1, 5, 5)	9	12	1912
$10 \times 10 \times 10$	(2, 5, 7)	15	22	696
$10 \times 10 \times 10$	(3, 4, 5)	15	26	384
$6 \times 10 \times 15$	(1, 1, 1)	3	8	511
$6 \times 10 \times 15$	(2, 2, 2)	7	20	214
$6 \times 10 \times 15$	(5, 5, 5)	26	37	501

5 Concluding remarks

In this chapter we considered low rank tensor recovery for hierarchical tensors extending the classical Tucker format to a multi-level framework. For low ranks, this model can break the curse of dimensionality. Its number of degrees of freedom scale like $\mathcal{O}(ndr + dr^3) \ll \mathcal{O}(n^d)$ and $\mathcal{O}(ndr^2)$ for TT tensors instead of $\mathcal{O}(n^d)$. Under the assumption of a tensor restricted isometry property, we have shown local convergence for Riemannian gradient iterations and global convergence under a certain condition on the iterates of the tensor iterative hard thresholding algorithm for hierarchical tensors, including the classical Tucker format as well as tensor trains. For instance for TT tensors, an estimate of the TRIP for Gaussian measurement maps was provided that requires the number of measurements to scale like $m \sim ndr^2 \log(dr)$. However, it is still not clear whether the logarithmic factor is needed.

Let us finally mention some open problems. One important task is to establish global convergence to the original tensor of any of the discussed algorithms, without additional assumptions such as (21) on the iterates. In addition, robustness and stability for the Riemannian gradient method are still open. Further, also the TRIP related to general HT tensors for Gaussian measurement maps is not yet established. Since the TRIP does not hold for the completion problem, it is not clear yet whether a low rank tensor can be recovered from less than $\mathcal{O}(n^{d/2})$ entries.

References

1. Absil, P.-A., Mahony, R.E., Sepulchre, R.: Optimization algorithms on matrix manifolds. *Foundations of Computational Mathematics* **10**, 241–244 (2010)
2. Arnold, A., Jahnke, T.: On the approximation of high-dimensional differential equations in the hierarchical Tucker format. *BIT Numerical Mathematics* **54**, 305–341 (2014)
3. Beck, M.H., Jäckle, A., Worth, G.A., Meyer, H.-D.: The multi-configuration time-dependent Hartree (MCTDH) method: a highly efficient algorithm for propagating wavepackets. *Phys. Reports* **324**, 1–105 (2000)
4. Beylkin, G., Garcke, J., Mohlenkamp, M.J.: Multivariate regression and machine learning with sums of separable functions. *SIAM J. Sci. Comput.* **31**, 1840–1857 (2009)
5. Beylkin, G., Mohlenkamp, M.J.: Algorithms for numerical analysis in high dimensions. *SIAM J. Sci. Comput.* **26**, 2133–2159 (2005)
6. Bhatia, R.: *Matrix Analysis*. Graduate Texts in Mathematics **169**, Springer (1997)
7. Blumensath, T., Davies, M.: Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis* **27**, 265–274 (2009)
8. Blumensath, T., Davies, M.: Iterative thresholding for sparse approximations. *J. Fourier Anal. Appl.* **14**, 629–654 (2008)
9. Candès, E.J., Recht, B.: Exact matrix completion via convex optimization. *Found. Comput. Math.* **9**, 717–772 (2009)
10. Candès, E.J., Tao, T.: The power of convex relaxation: near-optimal matrix completion. *IEEE Trans. Inform. Theory* **56**, 2053–2080 (2010)
11. Candès, E.J., Plan, Y.: Tight oracle bounds for low-rank matrix recovery from a minimal number of random measurements. *IEEE Trans. Inform. Theory* **57**, 2342–2359 (2011)
12. Carlini, E., Kleppe, J.: Ranks derived from multilinear maps. *Journal of Pure and Applied Algebra* **215**, 1999–2004 (2011)
13. Da Silva, C., Herrmann, F.J.: Hierarchical Tucker tensor optimization - Applications to tensor completion. In *Proc. 10th International Conference on Sampling Theory and Applications* (2013)
14. De Lathauwer, L., De Moor, B., Vandewalle, J.: A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.* **21**: 1253–1278 (2000)
15. Eldar, Y.C., Kutyniok, K. (Eds.): *Compressed Sensing : Theory and Applications*. Cambridge Univ. Press (2012)
16. Falcó, A., Hackbusch, W.: On minimal subspaces in tensor representations. *Found. Comput. Math.* **12**, 765–803 (2012)
17. Falcó, A., Hackbusch, W., Nouy, A.: Geometric structures in tensor representations. *Tech. Rep. 9*, MPI MIS Leipzig (2013)
18. Fazel, M.: *Matrix rank minimization with applications*. PhD thesis, Stanford University, CA (2002)
19. Foucart, S., Rauhut, H.: *A Mathematical Introduction to Compressive Sensing*. Applied and Numerical Harmonic Analysis, Birkhäuser (2013)
20. Friedland, S., Lim, L.-H.: Tensor nuclear norm and bipartite separability. In preparation
21. Friedland, S., Ottaviani, G.: The number of singular vector tuples and uniqueness of best rank-one approximation of tensors. To appear in *Foundations of Computational Mathematics*
22. Gandy, S., Recht, B., Yamada, I.: Tensor completion and low-n-rank tensor recovery via convex optimization. *Inverse Problems* **27**, 025010 (2011)
23. Grasedyck, L.: Hierarchical singular value decomposition of tensors. *SIAM. J. Matrix Anal. & Appl.* **31**, 2029–2054 (2010)
24. Grasedyck, L., Kressner, D., Tobler, C.: A literature survey of low-rank tensor approximation techniques. *GAMM-Mitteilungen* **36**, 53–78 (2013)
25. Gross, D.: Recovering low-rank matrices from few coefficients in any basis. *IEEE Trans. Inform. Theory* **57**, 1548–1566 (2011)
26. Hackbusch, W.: *Numerical tensor calculus*. *Acta Numerica* **23**, 651–742 (2014)

27. Hackbusch, W.: Tensor spaces and numerical tensor calculus. Springer series in computational mathematics **42** (2012)
28. Hackbusch, W.: Tensorisation of vectors and their efficient convolution. Numer. Math. **119**, 465–488 (2011)
29. Hackbusch, W., Kühn, S.: A new scheme for the tensor representation. J. Fourier Anal. Appl. **15**, 706–722 (2009)
30. Hackbusch, W., Schneider, R.: Tensor spaces and hierarchical tensor representations. In preparation
31. Haegeman, J., Osborne, T., Verstraete, F.: Post-matrix product state methods: to tangent space and beyond. Physical Review B **88**, 075133 (2013)
32. Hastad, J.: Tensor rank is NP-complete. J. of Algorithms **11**, 644–654 (1990)
33. Hillar, C.J., Lim, L.-H.: Most tensor problems are NP hard. J. ACM **60**, 45:1–45:39 (2013)
34. Holtz, S., Rohwedder, T., Schneider, R.: On manifolds of tensors of fixed TT rank. Numer. Math. **120**, 701–731 (2012)
35. Holtz, S., Rohwedder, T., Schneider, R.: The alternating linear scheme for tensor optimisation in the tensor train format. SIAM J. Sci. Comput. **34**, A683 – A713 (2012)
36. Huang, B., Mu, C., Goldfarb, D., Wright, J.: Provable low-rank tensor recovery. http://www.optimization-online.org/DB_FILE/2014/02/4252.pdf (2014)
37. Lim, L.-H., De Silva, V.: Tensor rank and the ill-posedness of the best low-rank approximation problem. SIAM J. Matrix Anal. & Appl. **30**, 1084–1127 (2008)
38. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. SIAM Review **51**, 455–500 (2009)
39. Kreimer N., Sacchi, M.D.: A tensor higher-order singular value decomposition for prestack seismic data noise reduction and interpolation. Geophysics **77**, V113-V122 (2012)
40. Kressner, D., Steinlechner, M., Vandereycken, B.: Low-rank tensor completion by Riemannian optimization. BIT Numerical Mathematics **54**, 447–468 (2014)
41. Landsberg, J.M.: Tensors: geometry and applications. Graduate Studies in Mathematics **128**, AMS, Providence, RI (2012)
42. Legeza, Ö., Rohwedder, T., Schneider, R., Szalay, S.: Tensor product approximation (DMRG) and coupled cluster method in quantum chemistry. Many-Electron Approaches in Physics, Chemistry and Mathematics, 53–76, Springer (2014)
43. Levin, J.: Three-Mode Factor Analysis. Ph.D. thesis, University of Illinois, Urbana (1963)
44. Liu, J., Musialski, P., Wonka, P., Ye, J.: Tensor completion for estimating missing values in visual data. Transactions of Pattern Analysis & Machine Intelligence (PAMI) **35**, 208–220 (2012)
45. Liu, Y., Shang, F.: An efficient matrix factorization method for tensor completion. IEEE Signal Processing Letters **20**, 307–310 (2013)
46. Lubich, C.: From quantum to classical molecular dynamics: Reduced methods and numerical analysis. Zürich Lectures in advanced mathematics **12**, EMS (2008)
47. Lubich, C., Rohwedder, T., Schneider, R., Vandereycken, B.: Dynamical approximation by hierarchical Tucker and Tensor-Train tensors. SIAM J. Matrix Anal. Appl. **34**, 470–494 (2013)
48. Mu, C., Huang, B., Wright, J., Goldfarb, D.: Square deal: Lower bounds and improved relaxations for tensor recovery. arxiv.org/abs/1307.5870v2 (2013)
49. Oseledets, I.V.: A new tensor decomposition. Doklady Math. **80**, 495–496 (2009)
50. Oseledets, I.V.: Tensor-train decomposition. SIAM J. Sci. Comput. **33**, 2295–2317 (2011)
51. Oseledets, I.V., Tyrtshnikov, E.E.: Algebraic wavelet transform via quantics tensor train decomposition. SIAM J. Sci. Comput. **33**, 1315–1328 (2011)
52. Oseledets, I.V., Tyrtshnikov, E.E.: Breaking the curse of dimensionality, or how to use SVD in many dimensions. SIAM J. Sci. Comput. **31**, 3744–3759 (2009)
53. Rauhut, H., Schneider, R., Stojanac, Ž.: Tensor recovery via iterative hard thresholding. In Proc. 10th International Conference of Sampling Theory and Appl. (2013)
54. Rauhut, H., Schneider, R., Stojanac, Ž.: Low rank tensor recovery via iterative hard thresholding. In preparation

55. Recht, B., Fazel, M., Parrilo, P.A.: Guaranteed minimum-rank solution of linear matrix equations via nuclear norm minimization. *SIAM Rev.* **52**, 471–501 (2010)
56. Recht, B.: A simpler approach to matrix completion. *J. Mach. Learn. Res.* **12**, 3413–3430 (2011)
57. Rohwedder, T., Uschmajew, A.: On local convergence of alternating schemes for optimization of convex problems in the tensor train format. *SIAM J. Numer. Anal.* **51**, 1134–1162 (2013)
58. Romera-Paredes, B., Pontil, M.: A new convex relaxation for tensor completion. *NIPS* **26**, 2967–2975 (2013)
59. Schneider, R., Uschmajew, A.: Approximation rates for the hierarchical tensor format in periodic Sobolev spaces. *Journal of Complexity* **30**, 56–71 (2014)
60. Schneider, R., Uschmajew, A.: Convergence results for projected line-search methods on varieties of low-rank matrices via Łojasiewicz inequality. arxiv.org/abs/1402.5284v1
61. Schollwöck, U.: The density-matrix renormalization group in the age of matrix product states, *Annals of Physics (NY)* **326**, 96–192 (2011)
62. Signoretto, M., De Lathauwer, L., Suykens, J.A.K.: Nuclear norms for tensors and their use for convex multilinear estimation. *Int. Rep.* 10–186, ESAT-SISTA, K. U. Leuven (2010)
63. Signoretto, M., Tran Dinh, Q., De Lathauwer, L., Suykens, J.A.K.: Learning with tensors: a framework based on convex optimization and spectral regularization. *Machine Learning* **94**, 303–351 (2014)
64. Tanner, J., Wei, K.: Normalized iterative hard thresholding for matrix completion. *SIAM J. Scientific Computing* **35**, S104–S125 (2013)
65. Tucker, L.R.: Some mathematical notes on three-mode factor analysis. *Psychometrika* **31**, 279–311 (1966)
66. Tucker, L.R.: Implications of factor analysis of three-way matrices for measurement of change. *Problems in Measuring Change*. Harris, C.W. (Eds.), University of Wisconsin Press, 122–137 (1963)
67. Tucker, L.R.: The extension of factor analysis to three-dimensional matrices. *Contributions to Mathematical Psychology*. Gulliksen, H., Frederiksen N. (Eds.), Holt, Rinehart & Winston, New York, 110–127 (1964)
68. Uschmajew, A.: Well-posedness of convex maximization problems on Stiefel manifolds and orthogonal tensor product approximations. *Numer. Math.* **115**, 309–331 (2010)
69. Uschmajew, A. and Vandereycken, B.: The geometry of algorithms using hierarchical tensors. *Linear Algebra and its Appl.* **439**, 133–166 (2013)
70. Vandereycken, B.: Low-rank matrix completion by Riemannian optimization. *SIAM J. Optim.* **23**, 1214–1236 (2013)
71. Vershynin, R.: Introduction to the non-asymptotic analysis of random matrices. *Compressed sensing: Theory and Applications*. Eldar, C.Y., Kutyniok, G. (Eds.), Cambridge Univ. Press, Cambridge, 210–268 (2012)
72. Vidal, G.: Efficient classical simulation of slightly entangled quantum computations. *Phys. Rev. Lett.* **91**, 147902 (2003)
73. Wang, H., Thoss, M.: Multilayer formulation of the multi-configuration time-dependent Hartree theory. *J. Chem. Phys.* **119**, 1289–1299 (2003)
74. Wen, Z., Yin, W., Zhang, Y.: Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm. *Math. Prog. Comp.* **4**, 333–361 (2012)
75. White, S.: Density matrix formulation for quantum renormalization groups. *Phys. Rev. Lett.* **69**, 2863–2866 (1992)
76. Xu, Y., Hao, R., Yin, W., Su, Z.: Parallel matrix factorisation for low-rank tensor completion. *UCLA CAM*, 13-77 (2013)
77. Xu, Y., Yin, W.: A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on Imaging Sciences* **6**, 1758–1789 (2013)