Computational Methods in Applied Mathematics
Vol. 11 (2011), No. 3, pp. 291–304

An Introduction to Hierarchical ($\mathcal{H}$-) Rank and TT-Rank of Tensors with Examples

Lars Grasedyck · Wolfgang Hackbusch

Abstract — We review two similar concepts of hierarchical rank of tensors (which extend the matrix rank to higher order tensors): the TT-rank and the $\mathcal{H}$-rank (hierarchical or $\mathcal{H}$-Tucker rank). Based on this notion of rank, one can define a data-sparse representation of tensors involving $\mathcal{O}(dnk + dk^3)$ data for order d tensors with mode sizes n and rank k. Simple examples underline the differences and similarities between the different formats and ranks. Finally, we derive rank bounds for tensors in one of the formats based on the ranks in the other format.

2010 Mathematical subject classification: 15A69; 65F99.

Keywords: hierarchical tucker; tensor rank; tensor approximation; tensor train.

1. Introduction

For matrices $A \in \mathbb{R}^{\mathcal{I}_1 \times \mathcal{I}_2}$ the standard definition of rank is based on maximal linear independency of rows

$$\{x \in \mathbb{R}^{\mathcal{I}_2} \mid \exists i \in \mathcal{I}_1 \text{ s.t. } \forall j \in \mathcal{I}_2 : \, x_j = A_{i,j}\},$$

or columns

$$\{x \in \mathbb{R}^{\mathcal{I}_1} \mid \exists j \in \mathcal{I}_2 \text{ s.t. } \forall i \in \mathcal{I}_1 : \, x_i = A_{i,j}\},$$

of the matrix (cf. Fig. 1.1) thus, we form vectors from A by keeping one of the indices fixed. For order d tensors

$$A \in \mathbb{R}^{\mathcal{I}}, \qquad \mathcal{I} := \mathcal{I}_1 \times \cdots \times \mathcal{I}_d, \qquad D := \{1, \ldots, d\},$$

there are many indices involved and not just two. As a consequence one can define the t-rank of the tensor A for any subset $t \subset D, t \neq \emptyset$, and the complement $s := D \setminus t$. For this purpose we introduce the notation

$$\mathcal{I}_t := \bigtimes_{\mu \in t} \mathcal{I}_\mu \tag{1.1}$$

and define the t-rank by

$$\operatorname{rank}_t(A) := \dim\bigl\{x \in \mathbb{R}^{\mathcal{I}_t} \bigm| \exists (j_\mu)_{\mu \in s} \in \mathcal{I}_s \text{ s.t. } \forall (j_\mu)_{\mu \in t} \in \mathcal{I}_t : \, x_{(j_\nu)_{\nu \in t}} = A_{j_1,\ldots,j_d}\bigr\}. \tag{1.2}$$

Lars Grasedyck
Institut für Geometrie und Praktische Mathematik, RWTH Aachen, Templergraben 55, 52056 Aachen, Germany
E-mail: lgr@igpm.rwth-aachen.de.

Wolfgang Hackbusch
Max Planck Institute for Mathematics in the Sciences, Inselstraße 22–26, 04103 Leipzig, Germany
E-mail: wh@mis.mpg.de.

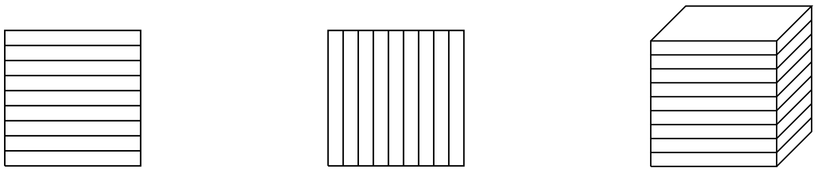


Figure 1.1. Left: rows $x = (A_{i,j})_{j\in\mathcal{I}_2}, i \in \mathcal{I}_1$, and columns $x = (A_{i,j})_{i\in\mathcal{I}_1}, j \in \mathcal{I}_2$, of a matrix $A \in \mathbb{R}^{\mathcal{I}_1\times\mathcal{I}_2}$. Right: for each index $i \in \mathcal{I}_1$ of an order three tensor $A \in \mathbb{R}^{\mathcal{I}_1\times\mathcal{I}_2\times\mathcal{I}_3}$ the second and third index form vectors $x = (A_{i,j,k})_{(j,k)\in\mathcal{I}_2\times\mathcal{I}_3}$

For every so-called *dimension cluster* $t \subset D, t \neq \emptyset$, we obtain a separate rank, cf. Figure 1.1 (right) for $t = \{2, 3\}$. Typically all of these are different — as opposed to the row-rank and column-rank of a matrix. Since there are quite many possible subsets of D, there are numerous ranks of tensors that can be considered.

Of particular interest are small systems of clusters $T \subset \mathfrak{P}(D)$, $\mathfrak{P}(D)$ being the powerset of D, that — provided all ranks $(\mathrm{rank}_t(A))_{t\in T}$ are bounded by k — allow for a data-sparse representation of the tensor and possibly for fast approximation schemes with a complexity scaling polynomially (with small degree) in the order d, the cardinalities of the one-dimensional index sets $n_\mu = \#\mathcal{I}_\mu$, and the rank bound k. At the same time the format should be rich enough to allow for an approximation of interesting classes of tensors.

An entirely different kind of rank that is not based on a matrix rank is the *tensor rank* or canonical rank or CP[1] rank of a tensor defined as the smallest $k \in \mathbb{N}_0$ such that the tensor can be represented in the CP format (cf. [8])

$$A = \sum_{\nu=1}^{k} a_\nu^1 \otimes \cdots \otimes a_\nu^d, \qquad a_\nu^\mu \in \mathbb{R}^{\mathcal{I}_\mu}. \tag{1.3}$$

The CP format requires — for the a_ν^μ — a storage complexity of only $\mathcal{O}(k(\#\mathcal{I}_1 + \cdots + \#\mathcal{I}_d))$, and it seems to be the most data-sparse tensor format that allows for a useful approximation (for an overview see [10] and the references therein). Any reasonable low rank tensor format should extend this class, sometimes for the purpose of enriching the approximation space, sometimes for complexity or stability reasons.

Remark 1.1. For every tensor A and every subset $t \subset D, t \neq \emptyset$, the t-rank is bounded by k from (1.3). Hence, tensor formats based on t-ranks are always at least as rich as the CP format, independently of the choice of the system $T \subset \mathfrak{P}(D)$.

An extreme example for the set T is

$$T = \{\{1\}, \ldots, \{d\}\}$$

which defines the Tucker format [15] with Tucker ranks $(\mathrm{rank}_\mu(A))_{\mu\in D}$. In this format an almost best low rank approximation can be obtained by the HOSVD [4] (higher order singular value decomposition). However, the storage complexity for the representation scales exponentially in the dimension d. We have to look for other sets T where also clusters t of larger cardinalities are used. When the cardinalities are large, then the vectors x from (1.2) cannot be trivially stored since their length scales exponentially in $\#t$. The remedy for this

[1] CP stands for canonical polyadic, in the literature also CANDECOMP and PARAFAC.

is the hierarchical ($\mathcal{H}$-) Tucker format [7, 5] based on a tree structure where each node t with complement $s := D \setminus t$ gives rise to a subspace (cf. (1.1))

$$V_t := \operatorname{span}\big\{x \in \mathbb{R}^{\mathcal{I}_t} \;\big|\; \exists (j_\mu)_{\mu\in s} \in \mathcal{I}_s \text{ s.t. } \forall (j_\mu)_{\mu\in t} \in \mathcal{I}_t : x_{(j_\nu)_{\nu\in t}} = A_{j_1,\ldots,j_d}\big\}, \tag{1.4}$$

and if the cardinality of t is larger than one, we split t into sons

$$t = t_1 \dot\cup t_2$$

so that the space V_t naturally decouples into

$$V_t = V_{t_1} \otimes V_{t_2},$$

i.e., every vector $x \in V_t$ can be written as a linear combination

$$x = \sum_{j=1}^{k_{t_1}} \sum_{\ell=1}^{k_{t_2}} c_{j,\ell} \cdot\; y_j \otimes z_\ell, \tag{1.5}$$

for every basis $(y_j)_{j=1}^{k_{t_1}}$ of V_{t_1}, and $(z_\ell)_{\ell=1}^{k_{t_2}}$ of V_{t_2} respectively. This *nestedness* property allows us to store only the $k_{t_1} \cdot k_{t_2}$ coefficients $c_{j,\ell}$ instead of the whole (possibly high-dimensional) vector x. The nestedness translates directly into a tree structure among the nodes $t \in T \subset \mathfrak{P}(D)$. Within such a tree structure, efficient, stable and reliable arithmetic operations based on the SVD (singular value decomposition) are possible, just like the HOSVD for the Tucker format.

Since every set $T \subset \mathfrak{P}(D)$ gives rise to different ranks $(\operatorname{rank}_t(A))_{t\in T}$, it is natural to ask where the differences between those lie. Our main results are the following:

1. The ranks for different choices of the tree can — even if both trees are binary and modes are permuted in an optimal way — differ by an exponential factor $k^{\log_2(d)/2-1}$ (or roughly $d^{\log_2(k)/2}$).
2. There is a specific (degenerate) tree, respectively set $T \subset \mathfrak{P}(D)$, with corresponding ranks $(\operatorname{rank}_t(A))_{t\in T}$ such that the ranks of all contiguous sets $u \subset D$ are bounded by $\operatorname{rank}_u(A) \leqslant k^2$, $k := \max_{t\in T} \operatorname{rank}_t(A)$.
3. The set T from above for the degenerate tree defines the so-called TT (tensor train) format, and this allows for a simplified representation which can be advantageous in practice.

The rest of this article is organised as follows: In Section 2 we give the detailed definition of the hierarchical Tucker format followed by the TT format in Section 3. In Section 4.1 we provide a simple example where the difference between the formats is visible. Finally, in Section 4.2 we provide rank bounds for the conversion between the formats.

2. The Hierarchical Tucker Format

In the hierarchical Tucker format, the sparsity of the representation of a tensor is determined by the hierarchical rank (1.2) for subsets t from a dimension tree. If one puts the vectors x from (1.4) into the columns of a matrix (cf. Fig. 2.1), then this is called a matricization of the tensor, and the rank of this matrix is the t-rank. The matricization is particularly useful to obtain reliable approximations for full rank tensors based on its singular values. However, in this article we will mention the practical and computational aspects only briefly and focus on theoretical aspects of the exact representation of tensors.

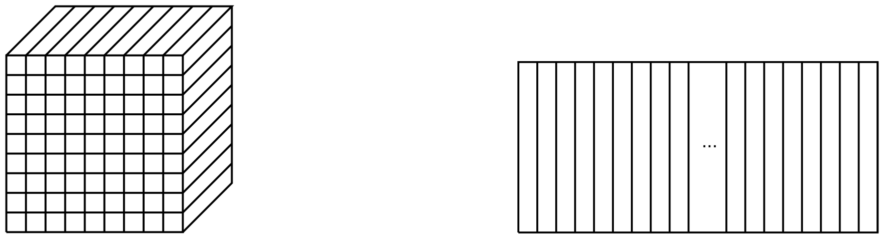


Figure 2.1. The third order ($\mathcal{I} = \mathcal{I}_1 \times \mathcal{I}_2 \times \mathcal{I}_3$) tensor (left) is matricized with respect to the third mode $t = \{3\}$, the resulting matrix (right) has $\#\mathcal{I}_1 \cdot \#\mathcal{I}_2 = 9 \cdot 9 = 81$ columns. Both, the tensor and the matrix have in total the same $9 \cdot 9 \cdot 9 = 729$ entries

2.1. Definition of $\mathcal{H}$-rank and $\mathcal{H}$-Tucker Tensors

For the rest of the article we use the notation

$$\mathcal{I} := \mathcal{I}_1 \times \cdots \times \mathcal{I}_d, \qquad \mathcal{I}_\mu := \{1, \ldots, n_\mu\}, \quad \mu \in D := \{1, \ldots, d\}.$$

Definition 2.1 (Matricization)**.** For a tensor $A \in \mathbb{R}^{\mathcal{I}}$, a collection of dimension indices $t \subset D, t \neq \emptyset$, and the complement $s := \{1, \ldots, d\} \setminus t$, the matricization

$$A^{(t)} \in \mathbb{R}^{\mathcal{I}_t \times \mathcal{I}_s}, \qquad \mathcal{I}_t := \bigtimes_{\mu \in t} \mathcal{I}_\mu, \quad \mathcal{I}_s := \bigtimes_{\mu \in s} \mathcal{I}_\mu,$$

($A^{(D)} \in \mathbb{R}^{\mathcal{I}}$, respectively), is defined by its entries

$$\left(A^{(t)}\right)_{(i_\mu)_{\mu \in t}, (i_\mu)_{\mu \in s}} := A_{i_1, \ldots, i_d}.$$

It should be noted that in the definition of the matricization we have not specified an ordering of the row-indices and column-indices. For the definition of the rank the ordering is irrelevant, but for the visualization of examples we need a fixed ordering. For sake of simplicity we choose a lexicographic ordering: an index $(i_p, \ldots, i_q) \in \mathcal{I}_p \times \cdots \times \mathcal{I}_q$ is mapped to

$$\ell := i_p + (i_{p+1} - 1)n_p + (i_{p+2} - 1)n_p n_{p+1} + \cdots + i_q n_p \cdots n_{q-1}.$$

Example 2.1. The matricizations of the tensor

$$A_{i_1, i_2, i_3, i_4} := i_1 + 2(i_2 - 1) + 4(i_3 - 1) + 8(i_4 - 1), \qquad i_1, i_2, i_3, i_4 \in \{1, 2\}$$

are

$$A^{(\{1\})} = \begin{bmatrix} 1 & 3 & 5 & 7 & 9 & 11 & 13 & 15 \\ 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 \end{bmatrix}, \quad A^{(\{2\})} = \begin{bmatrix} 1 & 2 & 5 & 6 & 9 & 10 & 13 & 14 \\ 3 & 4 & 7 & 8 & 11 & 12 & 15 & 16 \end{bmatrix},$$

$$A^{(\{3\})} = \begin{bmatrix} 1 & 2 & 3 & 4 & 9 & 10 & 11 & 12 \\ 5 & 6 & 7 & 8 & 13 & 14 & 15 & 16 \end{bmatrix}, \quad A^{(\{4\})} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \end{bmatrix},$$

$$A^{(\{2,3,4\})} = (A^{(\{1\})})^T, \quad A^{(\{1,2\})} = \begin{bmatrix} 1 & 5 & 9 & 13 \\ 2 & 6 & 10 & 14 \\ 3 & 7 & 11 & 15 \\ 4 & 8 & 12 & 16 \end{bmatrix}, \quad A^{(\{3,4\})} = (A^{(\{1,2\})})^T.$$

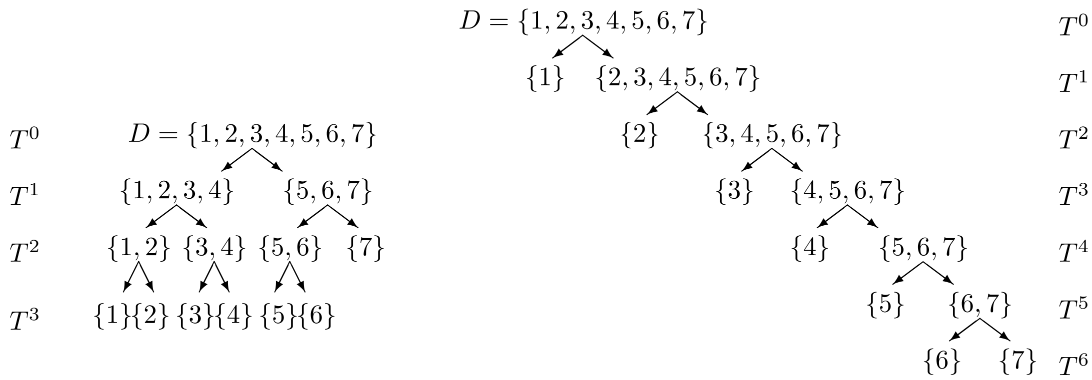


Figure 2.2. Left: The balanced canonical dimension tree. Right: The unbalanced TT-tree

Based on the matricization of a tensor A with respect to several sets $t \subset D$ one can define the hierarchical rank and the hierarchical Tucker format. In order to be able to perform efficient arithmetics, we require the sets t to be organized in a tree (the nestedness property (1.5) from the introduction).

Definition 2.2 (Dimension tree)**.** A dimension (cluster) tree T for dimension d is a finite tree with root D and depth p such that each node $t \in T$ is either

1. a leaf and singleton $t = \{\mu\}$ or
2. the union of two disjoint successors $S(t) = \{t_1, t_2\}$:

$$t = t_1 \dot{\cup} t_2. \tag{2.1}$$

The level ℓ of the tree is defined as the set of all nodes having a distance of exactly ℓ to the root. We denote the level ℓ of the tree by (cf. Figure 2.2)

$$T^\ell := \{t \in T \mid \text{level}(t) = \ell\}.$$

A node of the tree is a so-called cluster (a union of directions μ (sometimes called modes)).

In Fig. 2.2 we give two typical and, at the same time, extreme examples. The first is the **canonical dimension tree** which is of minimal depth subdividing each node $t = \{p, \ldots, p+q\}$ at the median $\lfloor q/2 \rfloor = \max_{r \in \mathbb{N}_{\leqslant q/2}} r$ into

$$S(t) = \{t_1, t_2\}, \qquad t_1 := \{p, \ldots, p + \lfloor q/2 \rfloor\}, \quad t_2 := \{p + \lfloor q/2 \rfloor + 1, \ldots, p+q\}.$$

The second is the **TT-tree** with nodes $t = \{p, \ldots, d\}$ subdivided into

$$S(t) = \{t_1, t_2\}, \qquad t_1 := \{p\}, \quad t_2 := \{p+1, \ldots, d\}.$$

Definition 2.3 (Hierarchical rank, $\mathcal{H}$-Tucker)**.** Let T be a dimension tree. The *hierarchical rank* or $\mathcal{H}$-rank $\underline{k} = (k_t)_{t \in T}$ of a tensor $A \in \mathbb{R}^{\mathcal{I}}$ is defined by

$$\forall t \in T: \quad k_t := \text{rank}(A^{(t)}).$$

The set of all tensors of hierarchical rank (node-wise) at most $\underline{k}$, the so-called $\mathcal{H}$-Tucker tensors, is defined as

$$\mathcal{H}\text{-Tucker}(T, \underline{k}) := \{A \in \mathbb{R}^{\mathcal{I}} \mid \forall t \in T: \ \text{rank}(A^{(t)}) \leqslant k_t\}.$$

2.2. Definition of the $\mathcal{H}$-Tucker Format

From Definition 2.3 of the $\mathcal{H}$-rank of a tensor A based on a dimension tree T, one can directly obtain a data-sparse representation of the tensor. For this, we first notice that a representation of the rank k_t matrix $A^{(t)}$ in the form

$$A^{(t)} = U_t V_t^T, \qquad U_t \in \mathbb{R}^{\mathcal{I}_t \times k_t},$$

is an exact representation of A for *every* node t of the tree T. Since for $S(t) = \{t_1, t_2\}$ the column-vectors $(U_t)_i$ of U_t fulfil the nestedness property (1.5), there exists for every $i \in \{1, \ldots, k_t\}$ a matrix $(B_t)_{i,\cdot,\cdot} \in \mathbb{R}^{k_{t_1} \times k_{t_2}}$ such that

$$(U_t)_i = \sum_{j=1}^{k_{t_1}} \sum_{\ell=1}^{k_{t_2}} (B_t)_{i,j,\ell} \cdot (U_{t_1})_j \otimes (U_{t_2})_\ell.$$

For the root D we have $A^{(D)} = U_D \in \mathbb{R}^{\mathcal{I} \times 1}$ ($k_D = 1$), and for leaves $t = \{\mu\} \in \mathcal{L}(T)$ the matrices $U_t \in \mathbb{R}^{\mathcal{I}_\mu \times k_t}$ are small enough to be stored in dense form. Thus, for all leaves $t \in \mathcal{L}(T)$ we have to store

$$\sum_{t=\{\mu\} \in \mathcal{L}(T)} n_\mu k_{\{\mu\}}$$

entries, for all interior nodes except the root we have to store

$$\sum_{t=\in T \setminus \mathcal{L}(T)} k_t k_{t_1} k_{t_2}$$

and for the root $t = D$ only $k_{t_1} k_{t_2}$ entries. A tensor stored or represented in this form is said to be given in $\mathcal{H}$-Tucker format.

Now we change our perspective: we consider tensors that allow a representation by some B_t and $U_{\{\mu\}}$ as above, but the sizes k_t involved in the representation need not be the t-ranks, i.e., the representation might not be minimal. This is called the $\mathcal{H}$-Tucker format and it is defined by

$$\text{representation ranks} \quad k_t \quad \forall t \in T \tag{2.2}$$

$$\text{transfer tensors} \quad B_t \in \mathbb{R}^{k_t \times k_{t_1} \times k_{t_2}} \quad \forall t \in T \setminus \mathcal{L}(T) \tag{2.3}$$

$$\text{frames} \quad U_{\{\mu\}} \in \mathbb{R}^{n_\mu \times k_{\{\mu\}}} \quad \forall \mu \in D. \tag{2.4}$$

Abbreviating $k := \max_{t \in T} k_t$ and $n := \max_{\mu \in D} n_\mu$ we arrive at $\mathcal{O}(dnk + dk^3)$ storage complexity, in particular linear in the order d and the mode size n of the tensor.

Remark 2.1. The column vectors $(U_{\{\mu\}})_i$ for the leaves in (2.4) might as well be stored in any compressed form via

$$(U_{\{\mu\}})_i = \sum_{j=1}^{\widehat{k}} (B_t)_{i,j} (\widehat{U}_{\{\mu\}})_j. \tag{2.5}$$

The vectors $(\widehat{U}_{\{\mu\}})_j$ could be wavelet representations [1, 3], hierarchical matrices [6, 2] or some other kind of data-sparse systems. If we ignore the costs to store the $\widehat{U}$, then the leaves (the matrices B_t in (2.5)) can be stored in $\mathcal{O}(dk^2)$ instead of $\mathcal{O}(dnk)$.

2.3. Properties of the $\mathcal{H}$-Tucker Format

In the previous section we have seen that a tensor with $\mathcal{H}$-rank bounded by k for all $t \in T$ and index sets of cardinality $\#\mathcal{I}_\mu \leqslant n$ for all $\mu \in D$, can be stored in

$$\mathcal{O}(dkn + dk^3).$$

In addition, the data-sparse representation (2.2-2.4) of a tensor allows for a formatted arithmetic based on singular value decompositions. The term 'formatted' means that arithmetic operations like linear combinations $A = \sum_{i=1}^{q} A_i$ of tensors are not computed in exact arithmetic, but instead the result A is projected to $\pi(A)$ with a prescribed hierarchical rank $\underline{k}$. This rank can as well be determined adaptively to ensure a prescribed accuracy $\|A - \pi(A)\| \leqslant \varepsilon$, just like for single precision or double precision floating point arithmetic. The projection is quasi-optimal [5],

$$\|A - \pi(A)\| \leqslant \sqrt{2d-3} \inf_{\tilde{A} \in \mathcal{H}\text{-Tucker}(T,\underline{k})} \|A - \tilde{A}\|, \tag{2.6}$$

and it can be computed for a tensor A in the above representation with representation ranks bounded by k and mode sizes $n_\mu \leqslant n$ in $\mathcal{O}(dk^2n + dk^4)$. The projection is the straight-forward orthogonal projection in each matricization to it's first singular vectors. The $\mathcal{H}$-Tucker format is thus data-sparse and useful for efficient arithmetics (see [5] for a more detailed description and proofs).

The estimates above hold for any dimension tree T as defined in Definition 2.2. Of course, the choice of the tree is guided by the rank properties of the tensor that is to be represented or approximated. In [13] it is shown that any tensor in Tree Tucker format can be represented in the TT format of the following section (with the same ranks but only after a permutation of the modes). This means that the Tree Tucker format is superfluous and we therefore omit to introduce it. The crucial question that remains is whether or not it is also possible to find a non-trivial transformation, e.g. permutation of the modes, so that a tensor in $\mathcal{H}$-Tucker-format can be represented with the same or similar ranks in the TT format. We will answer this question after the introduction of the TT format.

3. MPS and TT Format

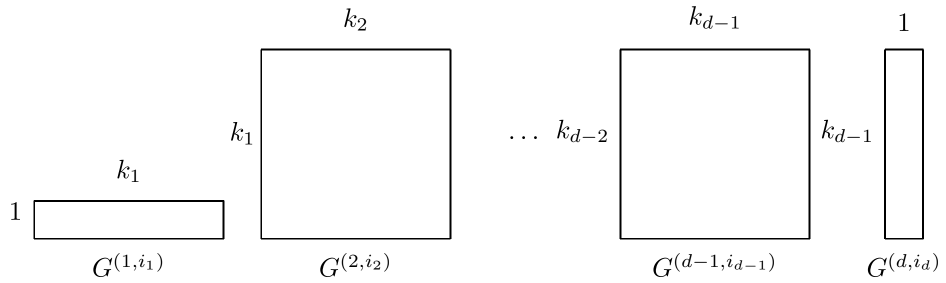
In this section we introduce a special variant of the general $\mathcal{H}$-Tucker format, the so-called TT (tensor train) format [12, 11]. The same format was introduced in the computational chemistry community under the name MPS (matrix product states) [17, 16].

For a tensor $A \in \mathbb{R}^{\mathcal{I}}$ the MPS format is a representation of the form

$$A_{i_1,\dots,i_d} = G^{(1,i_1)} \cdots G^{(d,i_d)}, \tag{3.1}$$

where

$$G^{(1,i_1)} \in \mathbb{R}^{1\times k_1}, \qquad G^{(\mu,i_\mu)} \in \mathbb{R}^{k_{\mu-1}\times k_\mu}, \mu = 2,\dots,d-1, \qquad G^{(d,i_d)} \in \mathbb{R}^{k_{d-1}\times 1}:$$



If we write all the matrix products explicitly and use the notation

$$G_\mu(j_{\mu-1}, i_\mu, j_\mu) := (G^{(\mu,i_\mu)})_{j_{\mu-1},j_\mu}, \quad G_1(i_1, j_1) := (G^{(1,i_1)})_{1,j_1}, \quad G_d(j_{d-1}, i_d) := (G^{(d,i_d)})_{j_{d-1},1},$$

then the entries of the tensor are given by

$$A_{i_1,\dots,i_d} = \sum_{j_1=1}^{k_1} \cdots \sum_{j_{d-1}=1}^{k_{d-1}} G_1(i_1, j_1) G_2(j_1, i_2, j_2) \cdots G_{d-1}(j_{d-2}, i_{d-1}, j_{d-1}) G_d(j_{d-1}, i_d),$$

which is the form of the TT format, $TT(\underline{k})$, defined in [12]. The minimal parameters k_μ for the representation are the TT-ranks $\text{rank}_t(A), t = \{1, \dots, \mu\}, \mu \in D$ [11, 9]. Thus, the ranks characterize the complexity to store a tensor in the TT-format. One can easily check that the storage complexity is in

$$\mathcal{O}(dk^2n)$$

for an order d tensor with mode sizes bounded by n and ranks bounded by k. Each $G^{(\mu,\cdot)}_{\cdot,\cdot}$ can be further separated in the Tucker format [15, 4],

$$(G^{(\mu,i)})_{j,l} = \sum_{\nu=1}^{r_\mu} B_{\nu,j,l} U_{\nu,i}$$

with a transfer tensor $B \in \mathbb{R}^{r_\mu \times k_{\mu-1} \times k_\mu}$ and a mode frame $U \in \mathbb{R}^{r_\mu \times n_\mu}$, which is preferable if the separation rank r_μ is considerably smaller than the mode size n_μ. In this case the complexity for the storage (representation) coincides with that in $\mathcal{H}$-Tucker format.

In [14] the approximation error for the projection to nodewise rank k_μ is given as

$$\|A - \pi(A)\| \leqslant \sqrt{d-1} \inf_{\tilde{A} \in TT(\underline{k})} \|A - \tilde{A}\|,$$

which is a corollary of the general $\mathcal{H}$-Tucker estimate when one leaves out the projection in the d singletons except the last one $t = \{d\}$. When also the ranks r_μ for the leaves $t = \{\mu\}$ are to be optimized, then the error bound is again given by (2.6).

4. Rank Bounds

In this section we provide bounds for the TT-rank ($t = \{1, \dots, \mu\}$) and $\mathcal{H}$-rank ($t = \{p, \dots, q\}$) based on given bounds in the respective other format. A first bound for the conversion from TT format to $\mathcal{H}$-Tucker format was given in [5]: If the TT-ranks of A are bounded by k, then the $\mathcal{H}$-Tucker-rank is bounded by k^2. For this estimate the nodes in T are subsets $t = \{q, \dots, r\}$ of D. Indeed, one can immediately construct an example where this bound is sharp: Let $d = 8$ and $n_1 = n_6 = n_7 = n_8 = 1$, $n_2 = n_3 = n_4 = n_5 = 2$. The tree T is a complete binary tree. Since half of the modes are of size 1, we can omit these and end up with a tree where the non-trivial modes $\{3,4\}$ have to be separated from $\{2,5\}$ (for all other nodes the number of rows or columns is at most 2), whereas the TT-ranks require a separation of the modes $\{2,3\}$ from $\{4,5\}$. We define the tensor A by its matricization

$$\begin{array}{cccc} \multicolumn{2}{c}{n_5 = 1} & \multicolumn{2}{c}{n_5 = 2} \\ {\scriptstyle n_2=1} & {\scriptstyle n_2=2} & {\scriptstyle n_2=1} & {\scriptstyle n_2=2} \end{array}.$$

$$A^{(\{3,4\})} := \left[\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right] \begin{array}{ll} {\scriptstyle n_3=1} & n_4{=}1 \\ {\scriptstyle n_3=2} & n_4{=}1 \\ {\scriptstyle n_3=1} & n_4{=}2 \\ {\scriptstyle n_3=2} & n_4{=}2 \end{array}.$$

Obviously, the rank of $A^{(\{3,4\})}$ is 4. On the other hand

$$A^{(\{2,3\})} = \left[\begin{array}{cc|cc} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{array}\right],$$

which is of rank 1 (other matricizations are rank 1 or 2). In this example we did not allow a permutation of the modes, otherwise the ranks are obviously always the same. In the following section we consider an example where the ranks are different even under every possible permutation, i.e., where the TT-rank is always strictly larger than the $\mathcal{H}$-rank.

4.1. A Minimal Example

In this section we provide an example, where the TT-rank and $\mathcal{H}$-rank differ — even if we allow arbitrary permutations of the modes. One can observe a principal structural difference that does not appear for small orders d. The smallest order where this is possible is $d = 6$.

We consider a dimension tree for the hierarchical format as it is provided in Fig. 4.1 (left). The tree has only three non-trivial nodes $\{1,2\}, \{3,4\}, \{5,6\}$, and corresponding ranks.

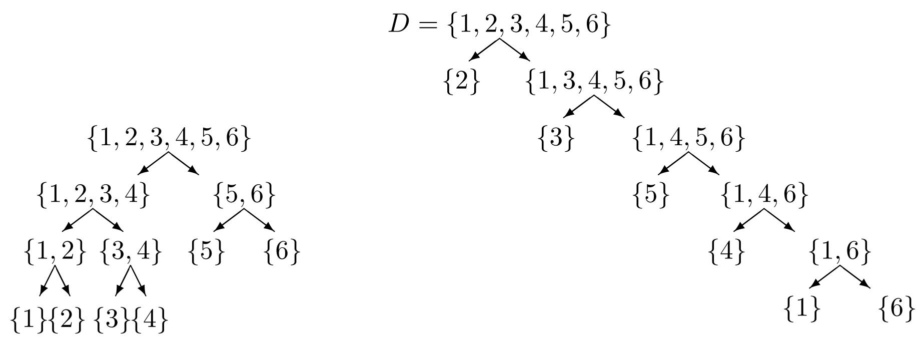


Figure 4.1. The example dimension tree for the hierarchical Tucker format (left) and one of the TT-trees where the node $t = \{1,4,6\}$ and the corresponding rank appears (right)

For a matricization with respect to the node $t = \{\nu_1, \nu_2\}$ ($t' = \{\nu_3, \nu_4, \nu_5, \nu_6\}$) and increasing $\nu_1 < \nu_2, \nu_3 < \nu_4 < \nu_5 < \nu_6$ we use, for the visualization, again the ordering

(indexing)

$$\ell := i_{\nu_1} + 2i_{\nu_2}, \qquad j := i_{\nu_3} + 2i_{\nu_4} + 4i_{\nu_5} + 8i_{\nu_6}, \qquad M_{\ell,j} := A^{(t)}_{(i_{\nu_1},i_{\nu_2}),(i_{\nu_3},i_{\nu_4},i_{\nu_5},i_{\nu_6})}.$$

The rank of $A^{(t)}$ is invariant under the ordering, but for the visualization we need to specify a fixed ordering which is the one by ℓ and j in M. The three matricizations of our example tensor A are:

$$A^{(\{1,2\})} = \begin{bmatrix} 2&5&2&3&2&5&2&3&1&2&1&1&2&5&2&3\\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\ 3&8&3&5&3&8&3&5&3&7&3&4&3&8&3&5\\ 3&8&3&5&3&8&3&5&3&7&3&4&3&8&3&5 \end{bmatrix},$$

$$A^{(\{3,4\})} = \begin{bmatrix} 2&5&2&3&0&0&0&0&3&8&3&5&3&8&3&5\\ 2&5&2&3&0&0&0&0&3&8&3&5&3&8&3&5\\ 1&2&1&1&0&0&0&0&3&7&3&4&3&7&3&4\\ 2&5&2&3&0&0&0&0&3&8&3&5&3&8&3&5 \end{bmatrix},$$

$$A^{(\{5,6\})} = \begin{bmatrix} 2&2&1&2&0&0&0&0&3&3&3&3&3&3&3&3\\ 5&5&2&5&0&0&0&0&8&8&7&8&8&8&7&8\\ 2&2&1&2&0&0&0&0&3&3&3&3&3&3&3&3\\ 3&3&1&3&0&0&0&0&5&5&4&5&5&5&4&5 \end{bmatrix}.$$

One can easily see that the ranks of all three matricizations are exactly two. Now the question is whether there exists an ordering $\nu_1, \ldots, \nu_6$ of the dimension indices $D = \{1, \ldots, d\}$ such that the TT-tree for this ordering with nodes

$$\{\nu_1, \ldots, \nu_6\}, \quad \{\nu_1\}, \{\nu_2, \ldots, \nu_6\}, \quad \{\nu_2\}, \{\nu_3, \ldots, \nu_6\}, \quad \ldots, \quad \{\nu_5\}, \{\nu_6\}$$

yields matricizations with ranks bounded by two. In every TT-tree there appears a node t of cardinality three (such nodes do not appear in the canonical tree). Either t or it's complement contains the first mode $\mu = 1$, and for symmetry reasons we can assume that $1 \in t$. In the following we list the matricizations for the node t for all possible three-element subsets $t \subset D$ that contain the first mode:

$$\boxed{A^{(\{1,2,3\})}} \begin{bmatrix} 2&5&2&3&2&5&2&3\\ 1&2&1&1&2&5&2&3\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 3&8&3&5&3&8&3&5\\ 3&7&3&4&3&8&3&5\\ 3&8&3&5&3&8&3&5\\ 3&7&3&4&3&8&3&5 \end{bmatrix}, \quad \boxed{A^{(\{1,2,4\})}} \begin{bmatrix} 2&5&2&3&1&2&1&1\\ 2&5&2&3&2&5&2&3\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 3&8&3&5&3&7&3&4\\ 3&8&3&5&3&8&3&5\\ 3&8&3&5&3&7&3&4\\ 3&8&3&5&3&8&3&5 \end{bmatrix}, \quad \boxed{A^{(\{1,2,5\})}} \begin{bmatrix} 2&5&2&5&2&3&2&3\\ 1&2&2&5&1&1&2&3\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 3&8&3&8&3&5&3&5\\ 3&7&3&8&3&4&3&5\\ 3&8&3&8&3&5&3&5\\ 3&7&3&8&3&4&3&5 \end{bmatrix},$$

$$\boxed{A^{(\{1,2,6\})}} \begin{bmatrix} 2&2&2&2&1&1&2&2\\ 5&3&5&3&2&1&5&3\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 3&3&3&3&3&3&3&3\\ 8&5&8&5&7&4&8&5\\ 3&3&3&3&3&3&3&3\\ 8&5&8&5&7&4&8&5 \end{bmatrix}, \quad \boxed{A^{(\{1,3,4\})}} \begin{bmatrix} 2&5&2&3&0&0&0&0\\ 2&5&2&3&0&0&0&0\\ 1&2&1&1&0&0&0&0\\ 2&5&2&3&0&0&0&0\\ 3&8&3&5&3&8&3&5\\ 3&8&3&5&3&8&3&5\\ 3&7&3&4&3&7&3&4\\ 3&8&3&5&3&8&3&5 \end{bmatrix}, \quad \boxed{A^{(\{1,3,5\})}} \begin{bmatrix} 2&5&2&5&0&0&0&0\\ 2&3&2&3&0&0&0&0\\ 1&2&2&5&0&0&0&0\\ 1&1&2&3&0&0&0&0\\ 3&8&3&8&3&8&3&8\\ 3&5&3&5&3&5&3&5\\ 3&7&3&8&3&7&3&8\\ 3&4&3&5&3&4&3&5 \end{bmatrix},$$

$$\boxed{A^{(\{1,3,6\})}} \qquad \boxed{A^{(\{1,4,5\})}} \qquad \boxed{A^{(\{1,4,6\})}}$$

$$\begin{bmatrix} 2&2&2&2&0&0&0&0\\ 5&3&5&3&0&0&0&0\\ 1&1&2&2&0&0&0&0\\ 2&1&5&3&0&0&0&0\\ 3&3&3&3&3&3&3&3\\ 8&5&8&5&8&5&8&5\\ 3&3&3&3&3&3&3&3\\ 7&4&8&5&7&4&8&5 \end{bmatrix}, \quad \begin{bmatrix} 2&5&1&2&0&0&0&0\\ 2&3&1&1&0&0&0&0\\ 2&5&2&5&0&0&0&0\\ 2&3&2&3&0&0&0&0\\ 3&8&3&7&3&8&3&7\\ 3&5&3&4&3&5&3&4\\ 3&8&3&8&3&8&3&8\\ 3&5&3&5&3&5&3&5 \end{bmatrix}, \quad \begin{bmatrix} 2&2&1&1&0&0&0&0\\ 5&3&2&1&0&0&0&0\\ 2&2&2&2&0&0&0&0\\ 5&3&5&3&0&0&0&0\\ 3&3&3&3&3&3&3&3\\ 8&5&7&4&8&5&7&4\\ 3&3&3&3&3&3&3&3\\ 8&5&8&5&8&5&8&5 \end{bmatrix},$$

$$\boxed{A^{(\{1,5,6\})}}$$

$$\begin{bmatrix} 2&2&1&2&0&0&0&0\\ 5&5&2&5&0&0&0&0\\ 2&2&1&2&0&0&0&0\\ 3&3&1&3&0&0&0&0\\ 3&3&3&3&3&3&3&3\\ 8&8&7&8&8&8&7&8\\ 3&3&3&3&3&3&3&3\\ 5&5&4&5&5&5&4&5 \end{bmatrix}.$$

Each of the matricizations has a rank of 4, thus the TT-rank of the tensor has — for any permutation $\nu_1, \dots, \nu_6$ of the modes D — at least one rank parameter of size 4, whereas the $\mathcal{H}$-rank of the tensor is bounded uniformly by 2.

Numerical examples in higher dimension $d > 6$ are quite invovled, not only due to the size of the matrices from a matricization, but also due to the number of possible permutations.

The reason for the possible increase however is now obvious: If the tree T contains only nodes of size 2^ℓ, $d = 2^p$, $p > 1$ odd, and the TT-rank involves nodes of all cardinalities, for example a node t with $\#t = q$,

$$q := 1 + 4 + 16 + 64 + \cdots + 2^{p-1}, \qquad d - q = 1 + 2 + 8 + 32 + \cdots + 2^{p-2},$$

then due to the cardinality of t and $D \setminus t$ (the number of row-modes and column-modes of a matricization for the TT-rank) one can only bound the row-rank or column rank by the product of the ranks of a partition of t or $D \setminus t$ into sets of size 2^ℓ, which in this case requires at least $(p+1)/2$ elements in the partition and thus a rank of $k^{(p+1)/2} = k^{(\log_2(d)+1)/2}$.

4.2. General Rank Bound

The general rank bound consists of two parts: in the first part we prove an upper bound. Afterwards, we want to indicate that the bound is sharp for random tensors, which in particular shows that the difference between the hierarchical formats (trees) can be huge.

Lemma 4.1. *Let* $X \in \mathbb{R}^{\mathcal{J}_1 \times \mathcal{J}_2 \times \mathcal{J}_3}$ *and let*

$$\operatorname{rank}(X^{(1)}) \leqslant k_1, \quad \operatorname{rank}(X^{(2)}) \leqslant k_2, \quad \operatorname{rank}(X^{(3)}) \leqslant k_3.$$

Then there holds

$$\operatorname{rank}(X^{(\{1,2\})}) \leqslant \min\{k_1 k_2,\, k_3\}.$$

Proof. Due to the rank bounds the tensor has the representation

$$A = \sum_{j_1=1}^{k_1} \sum_{j_2=1}^{k_2} \sum_{j_3=1}^{k_3} c_{j_1,j_2,j_3} U^1_{j_1} \otimes U^2_{j_2} \otimes U^3_{j_3}$$

for suitable vectors U^μ_j. For the matricization with respect to the first two modes this reads

$$A^{(\{1,2\})} = \sum_{j_1=1}^{k_1}\sum_{j_2=1}^{k_2} \left(U^1_{j_1} \otimes U^2_{j_2}\right) \cdot \left(\sum_{j_3=1}^{k_3} c_{j_1,j_2,j_3} U^3_{j_3}\right)^T ,$$

and equivalently

$$(A^{(\{1,2\})})^T = A^{(\{3\})} = \sum_{j_3=1}^{k_3} U^3_{j_3} \cdot \left(\sum_{j_1=1}^{k_1}\sum_{j_2=1}^{k_2} c_{j_1,j_2,j_3} U^1_{j_1} \otimes U^2_{j_2}\right)^T .$$

The ranks of the two representations are bounded by the dimension of the image which is bounded by k_1k_2, and k_3 respectively. □

Lemma 4.2. *Let $d = 2^p \geqslant 2$ and let T be a (complete binary) canonical dimension tree of depth p. Let A be a tensor with $\mathcal{H}$-rank bounded by k. Then the TT-rank of the tensor is bounded by $k^{\lceil p/2\rceil}$.*

Proof. Let $t = \{1,\dots,q\}$, $t' = \{q+1,\dots,d\}$, $1 \leqslant q < d$ and

$$q = 2^{q_0} + 2^{q_1} + \dots + 2^{q_{p'}}, \qquad q_0 > q_1 > \dots > q_{p'}, \quad p' < p.$$

Then t and t' are each the union of together at most $p+1$ nodes from T,

$$t = t_0 \dot\cup \cdots \dot\cup t_r, \quad t' = t_{r+1} \dot\cup \cdots \dot\cup t_p, \quad t_j \in T.$$

By repeated application of Lemma 4.1 we conclude

$$\operatorname{rank}(A^{(t)}) \leqslant k^{r+1}, \qquad \operatorname{rank}(A^{(t')}) \leqslant k^{p-r}.$$

One of the two numbers, $r+1$ or $p-r$, is at most $\lceil p/2\rceil$ which proves the assertion. □

Example 4.1. Let $d = 2^p$, $p > 1$ odd, and let T be a (complete binary) canonical dimension tree. Let the tensor $A \in \mathcal{H}$-Tucker(T,k) be such that the $U_t \in \mathbb{R}^{n_\mu \times k}$ for all leaves $t = \{\mu\} \in \mathcal{L}(T)$, and all transfer tensors $B_t \in \mathbb{R}^{k_t \times k_{t_1} \times k_{t_2}}$ for $S(t) = \{t_1,t_2\}$ have random entries in $[-1,1]$. Then for any permutation $\pi : D \to D$ the node

$$t := \{\pi(1),\dots,\pi(q)\}, \qquad t' := D \setminus t, \qquad q := 1+4+16+64+\dots+2^{p-1},$$

(due to the binary splitting the tree T contains only nodes of cardinality 2^j) are each the union of at least $(p+1)/2$ maximal nodes $s \in T$ (father of s is not a subset of t):

$$t = t_1 \dot\cup \cdots \dot\cup\, t_r, \qquad t' = t'_1 \dot\cup \cdots \dot\cup\, t'_{r'}, \qquad r, r' \geqslant (p+1)/2, \quad t_i, t'_j \in T.$$

For all nodes t_j and t'_j we can assume that the corresponding frames U_i, U'_j are of rank k. Accordingly, the full outer product of all column vectors from U_i, $i = 1,\dots,r$, spans a k^r-dimensional space. The same holds for t' and r', respectively. Thus, without further rank restrictions apart from the rank for all t_i, t'_j and their respective successors, we would obtain the expected full rank $k_t = \min\{k^r, k^{r'}\} \geqslant k^{(p+1)/2}$,

$$A^{(t)} = \sum_{j_1=1}^{k} \cdots \sum_{j_r=1}^{k} \sum_{j'_1=1}^{k} \cdots \sum_{j'_{r'}=1}^{k} G_{j_1,\dots,j_r,j'_1,\dots,j'_{r'}} \left(\bigotimes_{\mu=1}^{r} (U_\mu)_{j_\mu}\right) \left(\bigotimes_{\mu=1}^{r'} (U'_\mu)_{j'_\mu}\right)^T \tag{4.1}$$

It remains to show that the remaining rank restrictions do not affect the rank k_t. Since this seems to be rather technical and difficult, we do not give a proof here.

5. Conclusions

The hierarchical rank formats (hierarchical Tucker ($\mathcal{H}$-Tucker) and tensor train (TT)) reviewed in this article allow for a data-sparse representation of high order tensors with only $\mathcal{O}(dnk + dk^3)$ or $\mathcal{O}(dnk^2)$ parameters. Additionally, these formats enable a fast truncation procedure in order to find minimal rank approximations in a rank-reducing step.

Although the complexity of these formats scales only linearly in the dimension d, the dependency on the rank k can be a limiting factor (sometimes called 'curse of the rank'). Indeed, the ranks depend strongly on the tree or permutation of modes (unlike the CP or Tucker format) and thus raise the question which format is 'the best'. Of course, there is no straight answer to this. Each format can be advantageous over the other one, depending on the kind of tensor to be represented.

As we have shown, the ranks required for the $\mathcal{H}$-Tucker format based on an arbitrary tree can always be bounded by the ranks in the TT format squared. This is not true in the other direction: tensors with ranks in the $\mathcal{H}$-Tucker format bounded by k are typically such that for every permutation of modes, the ranks required to store the tensor in TT format scale as $k^{\log_2(d)/2}$.

References

[1] G. Beylkin, R. Coifman, and V. Rokhlin, *The fast wavelet transform and numerical algorithms*, Comm. Pure and Appl. Math., **44** (1991), pp. 141–183.

[2] S. Börm, L. Grasedyck, and W. Hackbusch, *Introduction to hierarchical matrices with applications*, Engineering Analysis with Boundary Elements, **27** (2003), pp. 405–422.

[3] W. Dahmen and R. Schneider, *Wavelets on manifolds I: Construction and domain decomposition*, SIAM Journal of Mathematical Analysis, **31** (1999), pp. 184–230.

[4] L. De Lathauwer, B. De Moor, and J. Vandewalle, *A multilinear singular value decomposition*, SIAM Journal on Matrix Analysis and Applications, **21** (2000), no. 4, pp. 1253–1278.

[5] Lars Grasedyck, *Hierarchical singular value decomposition of tensors*, SIAM J. Matrix Anal. Appl., **31** (2010), pp. 2029–2054.

[6] W. Hackbusch, *A sparse matrix arithmetic based on $\mathcal{H}$-matrices, Part I: Introduction to $\mathcal{H}$-matrices*, Computing, **62** (1999), pp. 89–108.

[7] W. Hackbusch and S. Kühn, *A new scheme for the tensor representation*, J. Fourier Anal. Appl., **15** (2009), pp. 706–722.

[8] F. L. Hitchcock, *The expression of a tensor or a polyadic as a sum of products*, Journal of Mathematics and Physics, **6** (1927), pp. 164–189.

[9] S. Holtz, T. Rohwedder, and R. Schneider, *On manifolds of tensors of fixed tt-rank*, Preprint 749/2010, DFG Research Center MATHEON, TU Berlin, 2001.

[10] T. G. Kolda and B. W. Bader, *Tensor decompositins and applications*, SIAM Review, **51** (2009), no. 3, pp. 455–500.

[11] I.V. Oseledets, *Compact matrix form of the d-dimensional tensor decomposition*, submitted to SIAM J. Sci. Comp.

[12] I.V. Oseledets and E.E. Tyrtyshnikov, *Breaking the curse of dimensionality, or how to use svd in many dimensions*, SIAM J. Sci. Comp., **31** (2009), no. 5, pp. 3744–3759.

[13] I.V. Oseledets and E.E. Tyrtyshnikov, *Tensor tree decomposition does not need a tree*, submitted to Lin. Alg. Appl.

[14] I.V. Oseledets and E.E. Tyrtyshnikov, *TT-cross approximation for multidimensional arrays*, Linear Algebra and its Applications, **432** (2010), pp. 70–88.

[15] L. R. Tucker, *Some mathematical notes on three-mode factor analysis*, Psychometrika, **31** (2010), pp. 279–311.

[16] G. Vidal, *Efficient classical simulation of slightly entangled quantum computation*, Phys. Review Letters, **91** (2003), no. 14, pp. 7902–1–7902–4.

[17] S. White, *Density matrix formulation for quantum renormalization groups*, Phys. Review Letters, **63** (1992), no. 19, pp. 2863–2866.