Hierarchical Singular Value Decomposition of Tensors

L. Grasedyck

Preprint 20
Edited by

AG Numerik/Optimierung
Fachbereich 12 - Mathematik und Informatik
Philipps-Universität Marburg
Hans-Meerwein-Str.
35032 Marburg
Hierarchical Singular Value Decomposition of Tensors

L. Grasedyck

Preprint 20
The consecutive numbering of the publications is determined by their chronological order.

The aim of this preprint series is to make new research rapidly available for scientific discussion. Therefore, the responsibility for the contents is solely due to the authors. The publications will be distributed by the authors.
Hierarchical Singular Value Decomposition of Tensors

Lars Grasedyck

July 8, 2009

We define the hierarchical singular value decomposition (SVD) for tensors of order $d \geq 2$. This hierarchical SVD has properties like the matrix SVD (and collapses to the SVD in $d = 2$), and we prove these. In particular, one can find low rank (almost) best approximations in a hierarchical format ($\mathcal{H}$-Tucker) which requires only $O((d-1)k^3 + dnk)$ data, where $d$ is the order of the tensor, $n$ the size of the modes and $k$ the rank. The $\mathcal{H}$-Tucker format is a specialization of the Tucker format and it contains as a special case all (canonical) rank $k$ tensors. Based on this new concept of a hierarchical SVD we present algorithms for hierarchical tensor calculations allowing for a rigorous error analysis. The complexity of the truncation (finding lower rank approximations to hierarchical rank $k$ tensors) is in $O(((d-1)k^4 + dnk^2)$ and the attainable accuracy is just 2–3 digits less than machine precision.

1 Introduction

Several problems of practical interest in physical, chemical, biological or mathematical applications naturally lead to high-dimensional (multivariate) approximation problems and thus are essentially not tractable when the dimension $d$ grows beyond $d = 10$. Examples are partial differential equations with many stochastic parameters, computational chemistry computations, the multiparticle electronic Schrödinger equation etc. This is due to the fact that the computational complexity or error bounds must depend exponentially on the dimension parameter $d$, which is coined by Bellman the curse of dimensionality. In order to make the setting more concrete we consider a multivariate function

$$f : [0, 1]^d \to \mathbb{R}$$

discretized by tensor basis functions

$$\phi_{(i_1, \ldots, i_d)}(x_1, \ldots, x_d) := \prod_{\mu=1}^{d} \phi_{i_\mu}(x_\mu), \quad \phi_{i_\mu} : [0, 1] \to \mathbb{R}, \quad i_\mu = 1, \ldots, n_\mu, \mu = 1, \ldots, d.$$ 

The total number $N$ of basis functions scales exponentially in $d$ as $N = \prod_{\mu=1}^{d} n_\mu$. One strategy to overcome this curse (in complexity) is to assume some sort of smoothness of the
function or object to be approximated so that one can choose a subspace \( \tilde{V} \) of
\[
V = \text{span}\{\phi_{(i_1, \ldots, i_d)} \mid (i_1, \ldots, i_d) \in \{1, \ldots, n_1\} \times \cdots \times \{1, \ldots, n_d\}\}.
\]
This leads to the sparse grids method \([13, 6]\) which chooses (adaptively \([7]\) or non-adaptively) combinations of basis functions. An alternative way to approximate the multivariate function \( f \) is to separate the variables, i.e. to seek for an approximation of the form
\[
f(x_1, \ldots, x_d) \approx \sum_{i=1}^{k} \prod_{\mu=1}^{d} f_{\mu,i}(x_\mu)
\]
where each of the univariate functions \( f_{\mu,i}(x) : [0, 1] \to \mathbb{R} \) is discretized by the full one-dimensional set of basis functions \( \phi_j(x), j = 1, \ldots, n_\mu \). If the separation rank \( k \) is small compared to \( N \), then this is an efficient data-sparse representation. However, whereas sparse grids define a linear space, the set of functions representable with separation rank \( k \) is not a linear space. Therefore, numerical algorithms based on such a format require a truncation of the results of basic steps to this set, i.e. finding low separation rank approximations:
\[
\text{For given } f \in V \text{ find } \tilde{f} \in V \text{ of rank } k \text{ s.t. } \|f - \tilde{f}\| \approx \inf_{v \in V, \text{rank}(v) = k} \|f - v\|.
\]
This approximation problem suffers from the following difficulties:

1. A minimizer \( \tilde{f} \) does not necessarily exist (problem is ill-posed), cf. \([3]\). The corresponding minimizing sequence consists of factors with increasing norm (and leads to severe cancellation effects). This can easily be overcome by Lagrange multipliers or penalty terms involving the norm of the factors.

2. There are no known algorithms allowing for an a priori estimate of the truncation error, see e.g. \([9]\) for an overview on tensor algorithms. This is a severe bottleneck, because even in model problems one cannot be sure to find approximations of almost optimal rank — despite the fact that one might be able to prove that such a low rank approximation exists.

3. The approximation problem is rather difficult to solve if one wants to obtain an accuracy suitable for standard numerical applications, see e.g. \([1, 5]\) for the state of the art of efficient algorithms. Linearly converging algorithms like ALS seem to be favorable for coarse accuracy, but for high accuracy one has to employ Newton-type iterations which are difficult to implement efficiently.

Thus, for some cases it is known how to construct a low separation rank approximation with high accuracy and stable representation but in order to use this low rank format as a basic format in numerical algorithms one needs a rigorous (black box) arithmetic.

A new kind of separation scheme was introduced by Hackbusch and Kühn \([12]\) and is coined \emph{hierarchical} low rank tensor format (cf. also the tree Tucker format \([8, 10]\)). This new format allows the representation of order \( d \) tensors with \((d - 1)k^3 + k \sum_{\mu=1}^{d} n_\mu\) data, where \( k \) is the involved — implicitly defined — representation rank.

In this article we will define the hierarchical rank of a tensor by singular value decompositions (SVD). The hierarchical format is then characterized by a nestedness of subspaces that stem from the SVDs. We present a corresponding hierarchical SVD which has a similar
property as the higher order SVD (HOSVD) by De Lathauwer et al. [2], namely that the best approximation up to a factor of $\sqrt{2d - 3}$ is obtained via cutting off the hierarchical singular values. We then derive a truncation procedure for (1.) dense or unstructured tensors as well as (2.) those already given in hierarchical format. In the first case almost linear complexity with respect to the number of input data is achieved, in the latter case the truncation is of complexity $O((d - 1)k^4 + k^2 \sum_{\mu=1}^{d} n_\mu)$. At last we present numerical examples that underline the attainable accuracy which is close to machine precision (roughly $10^{-13}$ in double precision arithmetic) and apply the truncation for hierarchical tensors of order $d = 1,000,000$.

## 2 Tucker Format

**Notation 1 (Index set)** Let $d \in \mathbb{N}$ and $n_1, \ldots, n_d \in \mathbb{N}$. We consider tensors as vectors over product index sets. For this purpose we introduce the $d$-fold product index set

$$I := I_1 \times \cdots \times I_d, \quad I_\mu := \{1, \ldots, n_\mu\}, \quad (\mu \in \{1, \ldots, d\}).$$

The order of the index sets is in principle important, but since it will always be clear which index belongs to which index set we will treat them without specifying the order. If the ordering becomes important it will be mentioned.

**Definition 2 (Mode, matricization, fibre)** Let $A \in \mathbb{R}^I$. The dimension directions $\mu = 1, \ldots, d$ are called the modes. Let $\mu \in \{1, \ldots, d\}$. We define the index set

$$I^\mu := I_1 \times \cdots \times I_{\mu - 1} \times I_{\mu + 1} \times \cdots \times I_d$$

and the corresponding $\mu$-mode matricization by

$$\mathcal{M}_\mu : \mathbb{R}^I \to \mathbb{R}^{I_\mu \times I^\mu}, \quad (\mathcal{M}_\mu(A))_{i_\mu, (i_1, \ldots, i_{\mu - 1}, i_{\mu + 1}, \ldots, i_d)} := A(i_1, \ldots, i_d).$$

We use the short notation

$$A^\mu := \mathcal{M}_\mu(A)$$

and call this the $\mu$-mode matricization of $A$. The columns of $A^\mu$ define the $\mu$-mode fibres of $A$.

The $\mu$-mode matricization $A^\mu$ is in one-to-one correspondence with the tensor $A$. The vector 2-norm $\|A\|_2$ corresponds to the matrix Frobenius norm: $\|A^\mu\|_F = \|A\|_2$.

**Definition 3 (Multilinear multiplication $\circ$)** Let $A \in \mathbb{R}^I$, $\mu \in \{1, \ldots, d\}$ and $U_\mu \in \mathbb{R}^{J_\mu \times I_\mu}$. Then the $\mu$-mode multiplication $U_\mu \circ A$ is defined by the matricization

$$(U_\mu \circ A)^\mu := U_\mu A^\mu \in \mathbb{R}^{J_\mu \times I^\mu},$$

with entries

$$(U_\mu \circ A)_{(i_1, \ldots, i_{\mu - 1}, j, i_{\mu + 1}, \ldots, i_d)} := \sum_{i_\mu = 1}^{n_\mu} (U_\mu)_{j, i_\mu} A_{(i_1, \ldots, i_d)}.$$
The order of the mode multiplications is irrelevant for the multilinear multiplication.

**Definition 4 (Tucker rank, Tucker format, mode frames)** The Tucker rank of a tensor $A \in \mathbb{R}^I$ is the tuple $(k_1, \ldots, k_d)$ with (elementwise) minimal entries $k_\mu \in \mathbb{N}_0$ such that there exist (columnwise) orthonormal matrices $U_\mu \in \mathbb{R}^{n_\mu \times k_\mu}$ and a so-called core tensor $C \in \mathbb{R}^{k_1 \times \cdots \times k_d}$ with

$$A = (U_1, \ldots, U_d) \circ C. \quad (1)$$

The representation of the form (1) is called the orthogonal Tucker format, or in short we say $A = (U_1, \ldots, U_d) \circ C$ is an orthogonal Tucker tensor. We call a representation of the form (1) with arbitrary $U_\mu \in \mathbb{R}^{n_\mu \times \tilde{k}_\mu}$ the Tucker format. The set of tensors of Tucker rank at most $(k_1, \ldots, k_d)$ is denoted by Tucker$(k_1, \ldots, k_d)$. The matrices $U_\mu$ are called mode frames for the Tucker tensor representation.

For fixed orthonormal mode frames $U_\mu \in \mathbb{R}^{n_\mu \times k_\mu}$ the unique core tensor $C$ minimizing $\|A - (U_1, \ldots, U_d) \circ C\|$ is

$$C = (U_1^T, \ldots, U_d^T) \circ A.$$

**Definition 5 (Tucker truncation)** Let $A \in \mathbb{R}^I$. Let

$$A^{(\mu)} = U_\mu \Sigma_\mu V_\mu^T, \quad U_\mu \in \mathbb{R}^{n_\mu \times n_\mu},$$

be a singular value decomposition with diagonal matrix $\Sigma_\mu = \text{diag}(\sigma_{\mu,1}, \ldots, \sigma_{\mu,n_\mu})$. Then the truncation of $A$ to Tucker rank $(k_1, \ldots, k_d)$ is defined by

$$\mathcal{T}_{(k_1, \ldots, k_d)}(A) := (\bar{U}_1 \bar{U}_1^T, \ldots, \bar{U}_d \bar{U}_d^T) \circ A = (\bar{U}_1, \ldots, \bar{U}_d) \circ \left((\bar{U}_1^T, \ldots, \bar{U}_d^T) \circ A\right),$$

where $\bar{U}_\mu$ is the matrix of the first $k_\mu$ columns of $U_\mu$.

The truncation $\mathcal{T}_{(k_1, \ldots, k_d)}(A)$ yields an orthogonal Tucker tensor ($\bar{U}_\mu$ is orthogonal). Since the core tensor is uniquely defined by the orthonormal mode frames $U_\mu$, the approximation of a tensor $A$ in Tucker$(k_1, \ldots, k_d)$ is a minimization problem on a (product) Grassmann manifold. A best approximation $A^{\text{best}}$ always exists. The geometry of the Grassmann manifold can be exploited to develop efficient Newton and quasi-Newton methods for a local optimization [4, 11]. As an initial guess one can use the Tucker truncation which allows for an explicit a priori error bound given next.

**Lemma 6 (Tucker approximation)** Let $A \in \mathbb{R}^I$. We denote the best approximation of $A$ in Tucker$(k_1, \ldots, k_d)$ by $A^{\text{best}}$. The error of the truncation is bounded by

$$\|A - \mathcal{T}_{(k_1, \ldots, k_d)}(A)\| \leq \sqrt{\sum_{\mu=1}^d \sum_{i=k_\mu+1}^{n_\mu} \sigma_{\mu,i}^2} \leq \sqrt{d} \|A - A^{\text{best}}\|,$$

where the $\sigma_{\mu,i}$ are the $\mu$-mode singular values from Definition 5.

**Proof:** Property 10 in [2].

The error bound stated in Lemma 6 is an a priori upper bound for the truncation error in terms of the best approximation error. The truncation is in general not a best approximation (but it may serve as an initial guess for a subsequent optimization). In the following section we will provide an elegant proof for this Lemma.
Figure 1: Left: A dimension tree for $d = 6$. Right: The interior nodes $I(T_d)$ are coloured dark brown and the leaves $L(T_d)$ are green.

### 3 Hierarchical Tucker Format

The hierarchical Tucker format is a multilevel variant of the Tucker format — multilevel in terms of the order of the tensor. In order to define the format we have to introduce a hierarchy among the modes $\{1, \ldots, d\}$.

**Definition 7 (Dimension tree)** A dimension tree $T_d$ for dimension $d \in \mathbb{N}$ is a tree with root $\text{Root}(T_d) = \{1, \ldots, d\}$ and depth $p = \lceil \log_2(d) \rceil := \min \{i \in \mathbb{N}_0 \mid i \geq \log_2(d)\}$ such that each node $t \in T_d$ is either

1. a leaf and singleton $t = \{\mu\}$ on level $\ell \in \{p-1, p\}$ or
2. the disjoint union of two successors $S(t) = \{s_1, s_2\}$:
   $$t = s_1 \cup s_2.$$

The level $\ell$ of the tree is defined as the set of all nodes having a distance of exactly $\ell$ to the root, cf. Figure 1. We denote the level $\ell$ of the tree by

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

The set of leaves of the tree is denoted by $L(T_d)$ and the set of interior (non-leaf) nodes is denoted by $I(T_d)$. A node of the tree is a so-called mode cluster.

The dimension tree is almost a complete binary tree, except that on the last but one level there may appear leaves. In principle one could base the following considerations on arbitrary non-binary dimension trees, but for the ease of presentation we have restricted this. The canonical dimension tree is of the form presented in Figure 1 where each node $t = \{\mu_1, \ldots, \mu_q\}, q > 1$, has two successors

$$t_1 := \{\mu_1, \ldots, \mu_r\}, \quad r := \lfloor q/2 \rfloor := \max \{i \in \mathbb{N}_0 \mid i \leq q/2\}, \quad t_2 := \{\mu_{r+1}, \ldots, \mu_q\}.$$  

**Lemma 8** On each level $\ell$ of the dimension tree $T_d$ of depth $p$ the nodes are disjoint subsets of $\{1, \ldots, d\}$. The number of nodes on level $\ell$ is

$$\#T^\ell_d = \begin{cases} 2^\ell & \text{for } \ell < p \text{ and} \\ 2d - 2^p & (\leq d) \text{ for } \ell = p. \end{cases}$$  

For a complete binary tree $2d - 2^p = 2^{p+1} - 2^p = 2^p$ holds. The total number of nodes is $2d - 1$, the number of leaves is $d$ and the number of interior nodes is $d - 1$.  

5
**Proof:** The disjointness follows by (2). For levels \( \ell = 0, \ldots, p - 1 \) the tree is binary and thus the number of nodes doubles for each level. On the last but one level there are \( 2^{p-1} \) (disjoint) nodes, these can be either singletons \( (s) \) or 2-tuples \( (t) \), thus \( \#s + \#t = 2^{p-1} \). The total number of modes is \( d \), thus \( \#s + 2\#t = d \). Together we have \( \#t = d - 2^{p-1}, \) i.e. \( 2\#t = 2d - 2^p \) nodes (singletons) on level \( p \). The total number of nodes is

\[
\sum_{\ell=0}^{p-1} 2^\ell + 2d - 2^p = 2^p - 1 + 2d - 2^p = 2d - 1.
\]

\[\blacksquare\]

**Definition 9 (Matricization)** For a mode cluster \( t \) in a dimension tree \( T_I \) we define the complementary cluster \( t' := \{1, \ldots, d\} \setminus t \),

\[
I_t := \bigotimes_{\mu \in t} I_{\mu}, \quad I_{t'} := \bigotimes_{\mu \in t'} I_{\mu},
\]

and the corresponding \( t \)-matricization

\[
M_t : \mathbb{R}^T \to \mathbb{R}^{I_t \times I_{t'}}, \quad (M_t(A))_{(i_{\mu})_{\mu \in t}, (i_{\nu})_{\nu \in t'}} := A_{(i_1, \ldots, i_d)},
\]

where the special case is \( M_{\emptyset}(A) := M_{\{1, \ldots, d\}}(A) := A \). We use the short notation \( A^{(t)} := M_t(A) \).

We provide a simple example: let the tensor \( A \) be of the form

\[
A = a \otimes b \otimes q \otimes r \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}.
\]

Then the matricizations with respect to \( \{1, 2\} \) and \( \{2, 3\} \) are

\[
A^{([1,2])} = (a \otimes b)(q \otimes r)^T \in \mathbb{R}^{(I_1 \times I_2) \times (I_3 \times I_4)},
A^{([2,3])} = (b \otimes q)(a \otimes r)^T \in \mathbb{R}^{(I_2 \times I_3) \times (I_1 \times I_4)}.
\]

**Definition 10 (Hierarchical rank)** Let \( (k_t)_{t \in T_I} \) be a rank distribution for the dimension tree \( T_I \). The hierarchical rank \( (k_t)_{t \in T_I} \) of a tensor \( A \in \mathbb{R}^T \) is defined by

\[
\forall t \in T_I : \quad k_t := \text{rank}(A^{(t)}).
\]

The set of all tensors of hierarchical rank (node-wise) at most \( (k_t)_{t \in T_I} \) is denoted by

\[
\mathcal{H}-\text{Tucker}((k_t)_{t \in T_I}) := \{ A \in \mathbb{R}^T \mid \forall t \in T_I : \text{rank}(A^{(t)}) \leq k_t \}.
\]

According to the definition of the hierarchical rank one can define the hierarchical SVD by the node-wise SVDs of the matrices \( A^{(t)} \), cf. Figure 2. However, it is not obvious why and how this should lead to an efficient representation and correspondingly efficient algorithms. Instead, we will introduce a nested representation and reveal the connection to the node-wise SVDs later.
Figure 2: Depicted are the largest 24 singular values of $A(t)$ for each node $t \in T_I$ (the dimension tree from Figure 1) except the root in logarithmic scale ranging from 1 down to $10^{-16}$. The first number is the number of singular values larger than $10^{-14}$ and the number in brackets is the cardinality of $I_t$.

Definition 11 (Frame tree, $t$-frame, transfer tensor) Let $t \in T_I$ be a mode cluster and $(k_t)_{t \in T_I}$ a rank distribution. We call a matrix $U_t \in \mathbb{R}^{I_t \times k_t}$ a $t$-frame and a tuple $(U_s)_{s \in T_I}$ of frames a frame tree. A frame is called orthogonal if the columns are orthonormal and a frame tree is called orthogonal if each frame is. A frame tree is nested if for each interior mode cluster $t$ with successors $S(t) = \{t_1, t_2\}$ the following relation holds:

$$\text{image}(U_t) \subset \text{image}(U_{t_1}) \otimes \text{image}(U_{t_2}).$$

The tensor $B_t \in \mathbb{R}^{k_t \times k_{t_1} \times k_{t_2}}$ of coefficients for the representation of the columns $(U_t)_i$ of $U_t$ by the columns of $U_{t_1}, U_{t_2}$,

$$(U_t)_i = \sum_{j=1}^{k_{t_1}} \sum_{l=1}^{k_{t_2}} (B_t)_{i,j,l} (U_{t_1})_j \otimes (U_{t_2})_l,$$  \hspace{1cm} (3)

is called the transfer tensor.

For a nested frame tree it is sufficient to provide the transfer tensors for all interior mode clusters $t \in I(T_T)$ and the $t$-frames for the leaves $t \in L(T_T)$. Note that we have not yet imposed an orthogonality condition on the $t$-frames.

Definition 12 (Hierarchical Tucker format) Let $(k_t)_{t \in T_T}$ be a rank distribution for the dimension tree $T_T$ and let $A \in \mathcal{H}$-Tucker$((k_t)_{t \in T_T})$. Let $(U_t)_{t \in T_T}$ be a nested frame tree with transfer tensors $(B_t)_{t \in I(T_T)}$ and

$$\forall t \in T_T : \text{image}(A(t)) = \text{image}(U_t), \quad A = U_{\{1, \ldots, d\}}.$$
Then the representation \(((B_t)_{t \in I(T_2)}, (U_t)_{t \in L(T_2)})\) is a hierarchical Tucker representation of \(A\). Note that the columns of \(U_t\) need not be linear independent and that the size of \(B_t\) is not prescribed.

The representation of a tensor \(A \in \mathcal{H}\text{-Tucker}((k_t)_{t \in T_2})\) in the hierarchical Tucker format with orthogonal mode frames is unique up to orthogonal transformation of the \(t\)-frames.

**Lemma 13 (Storage complexity)** Let \(T_2\) be a dimension tree and \(A \in \mathcal{H}\text{-Tucker}((k_t)_{t \in T_2})\) given in hierarchical Tucker representation \(((B_t)_{t \in I(T_2)}, (U_t)_{t \in L(T_2)})\) and \(B_t \in \mathbb{R}^{k_t \times k_t \times k_t}\) for \(S(t) = \{t_1, t_2\}\), i.e. \(B_t\) of minimal size. Then the total storage for all transfer tensors \((B_t)_{t \in I(T_2)}\) and frames \((U_t)_{t \in L(T_2)}\) in terms of number of entries is bounded by

\[
\text{Storage}((B_t)_{t \in I(T_2)}, (U_t)_{t \in L(T_2)}) \leq (d - 1)k^3 + k \sum_{\mu=1}^{d} n_{\mu}, \quad k := \max_{t \in T_2} k_t, \tag{4}
\]

i.e. linearly in the dimension \(d\) (provided \(k\) is uniformly bounded).

**Proof:** For each leaf \(t = \{\mu\}\) of the dimension tree we have to store the \(t\)-frame \(U_t \in \mathbb{R}^{n_{\mu} \times k_t}\) which yields the second term in (4). For all \(d - 1\) interior mode clusters (Lemma 8) we have to store the transfer tensors \(B_t \in \mathbb{R}^{k_t \times k_t \times k_t}\), each has at most \(k^3\) entries.

**Lemma 14 (Successive truncation)** Let \(A \in \mathbb{R}^I\) and \(\pi_t, \pi_s\) orthogonal projections. Then

\[
\|A - \pi_t \pi_s A\|^2 \leq \|A - \pi_t A\|^2 + \|A - \pi_s A\|^2.
\]

**Proof:** We have

\[
\|A - \pi_t \pi_s A\| = \|(I - \pi_t)A + \pi_t (A - \pi_s A)\|.
\]

Due to the orthogonality of \((I - \pi_t), \pi_t\) we conclude

\[
\|A - \pi_t \pi_s A\|^2 = \|(I - \pi_t)A\|^2 + \|\pi_t (A - \pi_s A)\|^2 \leq \|(I - \pi_t)A\|^2 + \|A - \pi_s A\|^2.
\]

In particular Lemma 14 proves Lemma 6: let \(U_t, t = \{\mu\}\), denote the matrix of the \(k_t\) singular vectors of \(A^{(t)}\) corresponding to the largest singular values and

\[
(\pi_t A)^{(t)} := U_t U_t^T A^{(t)}, \quad \text{i.e. } \pi_t A := M_t^{-1}(U_t U_t^T M_t(A)).
\]

Then

\[
\|A - \pi_t A\|^2 = \sum_{i=k_{\mu}+1}^{n_{\mu}} \sigma_{i,t}^2.
\]

Since \(\pi_t A\) is the best approximation of \(A\) with \(\mu\)-mode rank \(k_t\), we also have \(\|A - \pi_t A\|^2 \leq \|A - A^{\text{best}}\|^2\) and thus

\[
\|A - T_{(k_1, ..., k_d)}(A)\| \leq \sqrt{d} \|A - A^{\text{best}}\|.
\]

**Definition 15 (Orthogonal frame projection)** Let \(T_2\) be a dimension tree, \(t \in T_2\) and \(U_t\) an orthogonal \(t\)-frame. Then we define the orthogonal frame projection \(\pi_t : \mathbb{R}^I \to \mathbb{R}^I\) in matricized form by

\[
(\pi_t A)^{(t)} := U_t U_t^T A^{(t)}, \quad \pi_0 A := A, \quad \pi_{\{1, ..., d\}} A := A.
\]
The order of the projections in a product of the form \((\prod_{t \in T} \pi_t)\) is relevant (the \(\pi_t\) do not necessarily commute). One has to be careful with the ordering, because the result of the product of the projections differs structurally.

**Lemma 16** Let \(T_I\) be a dimension tree and \(A \in \mathbb{R}^I\). For all \(t \in T\) let \(U_t \in \mathbb{R}^{I_t \times k_t}\) be orthogonal \(t\)-frames. Then for any order of the projections \(\pi_t\) holds

\[
\|A - \prod_{t \in T_I} \pi_t A\|^2 \leq \sum_{t \in T_I} \|A - \pi_t A\|^2.
\]

**Proof:** Apply Lemma 14 successively for all nodes of the dimension tree. 

**Theorem 17 (Hierarchical truncation error)** Let \(T_I\) be a dimension tree and \(A \in \mathbb{R}^I\). Let \(A_{\text{best}}\) denote the best approximation of \(A\) in \(\mathcal{H}\)-Tucker(\((k_t)_{t \in T}\)) and let \(\pi_t\) be the orthogonal frame projection for the \(t\)-frame \(U_t\) that consists of the left singular vectors of \(A^{(t)}\) corresponding to the \(k_t\) largest singular values \(\sigma_{t,i}\) of \(A^{(t)}\). Then for any order of the projections \(\pi_t, t \in T_I\), holds

\[
\|A - \prod_{t \in T_I} \pi_t A\| \leq \sqrt{\sum_{t \in T_I} \sum_{i > k_t} \sigma_{t,i}^2} \leq \sqrt{2d - 2}\|A - A_{\text{best}}\|.
\]

**Proof:** For any of the projections holds \(\|A - \pi_t A\|^2 = \sum_{i > k_t} \sigma_{t,i}^2 \leq \|A - A_{\text{best}}\|^2\) and for the root \(\|A - \pi_{\{1,\ldots,d\}} A\| = 0\) (w.l.o.g. \(k_{\{1,\ldots,d\}} = 1\)). Applying Lemma 16 and Lemma 8 yields

\[
\|A - \prod_{t \in T_I} \pi_t A\|^2 \leq \sum_{t \in T_I} \sum_{i > k_t} \sigma_{t,i}^2 \leq (2d - 2)\|A - A_{\text{best}}\|^2.
\]

**Remark 18** The estimate given in the previous theorem is not optimal and it can be improved as follows: for the root \(t\) of the dimension tree and its successors \(t_1,t_2\) one can combine both projections \(\pi_{t_1},\pi_{t_2}\) into a single projection via the SVD. This combined projection (with the pairs of the singular vectors) then has the same error as any of the two projections \(\pi_{t_1}\) or \(\pi_{t_2}\). Thereby, the error of the truncation can be estimated by

\[
\|A - \prod_{t \in T_I} \pi_t A\| \leq \sqrt{2d - 3}\|A - A_{\text{best}}\|.
\]

In dimension \(d = 2\) this coincides with the SVD estimate and in \(d = 3\) this coincides with the one-level Tucker estimate.

**Example 19 (Increasing the rank by projection)** We consider the tensor \(A \in \mathbb{R}^{3 \times 3 \times 3}\) in matricized form

\[
A^{(\{1,2\})} := \begin{bmatrix}
u_1 \otimes q_1 & u_2 \otimes q_2 & u_1 \otimes q_2
\end{bmatrix}
\]

with vectors

\[
u_1 = \begin{bmatrix}1 \\ 0 \\ 0 \end{bmatrix}, \quad u_2 = \begin{bmatrix}0 \\ 1 \\ 0 \end{bmatrix}, \quad q_1 = \begin{bmatrix}1/\sqrt{2} \\ 0 \\ 1/\sqrt{2} \end{bmatrix}, \quad q_2 = \begin{bmatrix}0 \\ 1 \end{bmatrix}.
\]
The mode cluster \( t = \{1,2\} \) has the two successors \( t_1 = \{1\}, t_2 = \{2\} \) and we consider the orthogonal mode frames

\[
U_t := \begin{bmatrix} u_1 \otimes q_1 & u_2 \otimes q_2 \end{bmatrix}, \quad U_{t_1} := \begin{bmatrix} u_1 \end{bmatrix}, \quad U_{t_2} := \begin{bmatrix} q_1 \end{bmatrix}.
\]

Clearly, \( \pi_{t_1} \) will project to \( t_1 \)-rank \( \text{rank}(\pi_{t_1}A^{(1)}) = 2 \). We will now show that the rank is at least 3 if we apply all three projectors. The matrix \( Q \) for the projection \( \pi_{t_1} \pi_{t_2} \) is given by

\[
Q = U_{t_1}U_{t_1}^T \otimes U_{t_2}U_{t_2}^T = (u_1u_1^T + u_2u_2^T) \otimes (q_1q_1^T + q_2q_2^T).
\]

We thus obtain

\[
QU_t = \begin{bmatrix} u_1 \otimes q_1 & u_2 \otimes q_2 \end{bmatrix}.
\]

The combined projection reads

\[
(\pi_{t_1} \pi_{t_2} A)^{(1,2)} = U_tU_t^TQA^{(1,2)} = U_t(QU_t^T)A^{(1,2)}
\]

\[
= U_t \begin{bmatrix} u_1 \otimes q_1 & u_2 \otimes q_2 \end{bmatrix}^T \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} = \begin{bmatrix} u_1 \otimes q_1 & u_2 \otimes q_2 & \frac{1}{\sqrt{2}} q_1 \otimes q_2 \end{bmatrix}.
\]

The matricization with respect to \( t_1 = \{1\} \) is of rank three,

\[
(\pi_{t_1} \pi_{t_2} A)^{(1)} = \begin{bmatrix} u_1q_1^T & u_2q_2^T & q_1(\frac{1}{\sqrt{2}} q_2)^T \end{bmatrix},
\]

because \( u_1, u_2, q_1 \) are linearly independent. We conclude: the first projection \( \pi_{t_1} \pi_{t_2} \) maps \( A \) into Tucker(2, 2, 3), but after the coarser projection \( \pi_t \) the 1-mode rank is three and thus \( \pi_{t_1} \pi_{t_2} \notin \text{Tucker}(2, 2, 3) \). This is because \( \pi_t \) mixes the \( t_1 \)-frame and the \( t_2 \)-frame. In the case of a nested frame tree this is not possible.

Lemma 20 (Structure of the hierarchical truncation) Let \( T_T \) be a dimension tree of depth \( p \), \( A \in \mathbb{R}^{T_T} \) and \( (k_t)_{t \in T_T} \). Let \( (U_t)_{t \in T_T}, U_t \in \mathbb{R}^{T_T \times k_t} \), be an orthogonal frame tree (not necessarily nested). Then the tensor

\[
A_{T_T} := \prod_{t \in T_T^p} \pi_t \cdots \prod_{t \in T_T^1} \pi_t A
\]

belongs to \( \mathcal{H} \)-Tucker((\( k_t \))_{t \in T_T}).

Proof: We define the tensors

\[
A_{T_T, \ell} := \prod_{t \in T_T^\ell} \pi_t \cdots \prod_{t \in T_T^1} \pi_t A.
\]

We prove \( \text{rank}(A_{T_T, \ell}) \leq k_t \) for all \( t \in T_T \) with \( \text{level}(t) \leq \ell \) by induction over the level \( \ell = 1, \ldots, p \). Level \( \ell = 1 \) is the Tucker truncation and thus the statement is true for \( \ell = 1 \). Now let \( \ell > 1 \) and assume that

\[
\forall t \in T_T, \ \text{level}(t) \leq \ell - 1 : \quad \text{rank}(A_{T_T, \ell-1}) \leq k_{t-1}.
\]
By construction

\[ A_{\mathcal{H}, \ell} = \prod_{t \in T^2} \pi_t A_{\mathcal{H}, \ell-1}. \]

This is the Tucker truncation on level \( \ell \) and thus for all \( t \in T^2 \) on level \( \ell \) the rank bound is fulfilled. It remains to show that for all levels \( 0, \ldots, \ell - 1 \) the rank bound is fulfilled, i.e., that the rank is not increased by the projections on level \( \ell \). Now let \( t \in T^2_j, j < \ell \). Let \( s \in T^2_j \).

We will show that the rank of \( A_{\mathcal{H}, \ell-1}^{(t)} \) is not increased by the projection \( \pi_s \). Due to the tree structure \( s \) is either a subset of \( t \) or they are disjoint.

\textbf{Case} \( s \subset t \): Let \( \hat{s} := t \setminus s \). Then the projection \( \pi_s \) is of the matricized form

\[ (\pi_s A)^{(t)} = (U_s U_s^T \otimes I) A^{(t)} \]

with \( I \) being the \( I \hat{s} \times I \hat{s} \) identity. The rank is not increased by the multiplication.

\textbf{Case} \( s \cap t = \emptyset \): Let \( \hat{s} := \{1, \ldots, d\} \setminus (t \cup s) \). Then the projection \( \pi_s \) is of the matricized form

\[ (\pi_s A)^{(t)} = A^{(t)} (U_s U_s^T \otimes I) \]

with \( I \) being the \( I \hat{s} \times I \hat{s} \) identity. The rank is not increased by the multiplication.

\textbf{Notation 21} By\n
\[ \psi_{t,k}(A) \]

we denote the \( I_t \times k \) matrix that consists of the left singular vectors of \( A^{(t)} \) corresponding to the \( k \) largest singular values of \( A^{(t)} \).

\textbf{Definition 22 (Hierarchical root-to-leaves truncation)} Let \( T^I \) be a dimension tree of depth \( p \), \( (k_t)_{t \in T} \) a rank distribution and \( A \in \mathbb{R}^{I} \). We define the hierarchical root-to-leaves truncation \( A_{\mathcal{H}} \in \mathcal{H}\text{-Tucker}((k_t)_{t \in T}) \) as follows:

\[ A_{\mathcal{H}} := \prod_{t \in T^2} \pi_t \cdots \prod_{t \in T^2} \pi_t A, \]

where \( \pi_t \) are the projections based on \( U_t := \psi_{t,k_t}(A) \in \mathbb{R}^{I_t \times k_t} \).

The hierarchical Tucker representation of \( A_{\mathcal{H}} \) from the previous definition is obtained by projection of the \( t \)-frames into the span of the sons \( U_{t_1} \otimes U_{t_2} \). The procedure for the construction is given in Algorithm 1. We want to remark that the algorithm is formulated for arbitrary tensors and the specialization to \( \mathcal{H}\text{-Tucker} \) tensors is the topic of the next section.

\textbf{Theorem 23 (Characterization of hierarchical approximability)} Let \( T^I \) be a dimension tree, \( A \in \mathbb{R}^{I} \), \( (k_t)_{t \in T^2} \) a rank distribution and \( \varepsilon > 0 \). If there exists a tensor \( A_{\text{best}}^{\text{best}} \) of hierarchical rank \( (k_t)_{t \in T} \) and \( \|A - A_{\text{best}}^{\text{best}}\| \leq \varepsilon \), then the singular values of \( A^{(t)} \) for each node \( t \) can be estimated by

\[ \sqrt{\sum_{i > k_t} \sigma_i^2} \leq \varepsilon. \]

On the other hand, if the singular values fulfil the bound \( \sqrt{\sum_{i > k_t} \sigma_i} \leq \varepsilon/\sqrt{2d - 2} \), then the truncation yields an \( \mathcal{H}\text{-Tucker} \) tensor \( A_{\mathcal{H}} := \prod_{t \in T^2} \pi_t A \) such that

\[ \|A - A_{\mathcal{H}}\| \leq \varepsilon. \]
Proof: The second part is proven by Theorem 17. The first part follows from the fact that 
\((A^{\text{best}})^{(t)}\) is a rank \(k_t\) approximation of \(A^{(t)}\) with \(\|A^{(t)} - (A^{\text{best}})^{(t)}\|_F \leq \varepsilon\). □

In Algorithm 1 we provide a method for the truncation of an arbitrary tensor to hierarchical 
rank \((k_t)_{t \in T_Z}\), of course one can as well prescribe node-wise tolerances \(\varepsilon_t\) for the truncation of 
singular values: according to Theorem 23 one can prescribe node-wise tolerance 
\(\varepsilon/\sqrt{2d-2}\) in order to obtain a guaranteed error bound of 
\(\|A - A_H\| \leq \varepsilon\). The complexity of Algorithm 1 is estimated in Lemma 24.

Algorithm 1 Root-to-leaves truncation of arbitrary tensors to \(\mathcal{H}\)-Tucker format

Require: Input tensor \(A \in \mathbb{R}^I\), dimension tree \(T_Z\) (depth \(p > 0\), rank distribution \((k_t)_{t \in T_Z}\).

for each singleton \(t \in L(T_Z)\) do

Compute an SVD of \(A^{(t)}\) and store the dominant \(k_t\) left singular vectors in the columns 
of the \(t\)-frame \(U_t\).

end for

for \(\ell = p - 1, \ldots, 1\) do

for each mode cluster \(t \in I(T_Z)\) on level \(\ell\) do

Compute an SVD of \(A^{(t)}\) and store the dominant \(k_t\) left singular vectors in the columns 
of the \(t\)-frame \(U_t\).

Let \(U_{t_1}\) and \(U_{t_2}\) denote the frames for the successors of \(t\) on level \(\ell + 1\). Compute the 
entries of the transfer tensor:

\[
(B_t)_{i,j,\nu} := \langle (U_t)_i, (U_{t_1})_j \otimes (U_{t_2})_\nu \rangle
\]

end for

end for

return \(\mathcal{H}\)-Tucker representation \(((U_t)_{t \in L(T_Z)}, (B_t)_{t \in I(T_Z)})\) for \(A_H \in \mathcal{H}\)-Tucker\(((k_t)_{t \in T_Z}))\).

Lemma 24 (Complexity of Algorithm 1) The complexity of Algorithm 1 for a tensor 
\(A \in \mathbb{R}^I\) and dimension tree \(T_Z\) of depth \(p > 0\) is in 
\(O\left(\left(\prod_{\mu=1}^d n_\mu\right)^{3/2}\right)\).

Proof: We have to compute singular value decompositions for all \(A^{(t)}\), and those decompositions have a complexity of 
\(O(\min(\#I_t, \#I_{t'})^2 \max(\#I_t, \#I_{t'}))\), where \(t'\) is the complementary 
mode cluster \(t' := \{1, \ldots\} \setminus t\). Without loss of generality we can assume \(n_\mu \geq 2\) for all modes 
\(\mu\). Then the complexity of the SVD for the root is zero, that for the two successors \(t, t'\) of the 
root is 
\[
C_{\text{SVD}}(\min(\#I_t, \#I_{t'})^2 \max(\#I_t, \#I_{t'})) \leq C_{\text{SVD}} \left(\prod_{\mu=1}^d n_\mu\right)^{3/2},
\]
where \(C_{\text{SVD}}\) is a universal constant for the SVD. For each further level there are at most two 
times more nodes, but the cardinality of \(I_t, I_{t'}\) is reduced by at least a factor of two \((n_\mu \geq 2)\) 
so that the complexity for the SVDs is quartered. Therefore the total complexity is bounded by 
\[
\sum_{\ell=0}^p 2^{-\ell} C_{\text{SVD}} \left(\prod_{\mu=1}^d n_\mu\right)^{3/2} \leq 2C_{\text{SVD}} \left(\prod_{\mu=1}^d n_\mu\right)^{3/2}.
\] □
The truncation presented in Algorithm 1 requires the computation of all (full) SVDs. We want to avoid the superlinear complexity \( \mathcal{O}(\prod_{\mu=1}^{d} n_{\mu})^{3/2} \) and instead work with a core tensor that becomes smaller as we come closer to the root of the tree. This means that we compute the SVDs not for the original tensor but for an already truncated one. The algorithm for this is given in Algorithm 2 and the complexity is estimated in Lemma 26.

**Definition 25 (Hierarchical leaves-to-root truncation)** Let \( T_{I} \) be a dimension tree of depth \( p \), \( (k_{t})_{t \in I} \) a rank distribution and \( A \in \mathbb{R}^{I} \). For all leaves \( t \in \mathcal{L}(T_{I}) \) let \( \pi_{t} \) denote the frame projection for \( U_{t} := \psi_{t,k_{t}}(A) \in \mathbb{R}^{\mathcal{I}_{t} \times k_{t}} \) and

\[
A_{\mathcal{H},p} := \prod_{t \in \mathcal{L}(T_{I})} \pi_{t} A.
\]

For all other levels \( \ell = p - 1, \ldots, 1 \) and \( t \in \mathcal{I}(T_{I}^{\ell}) \) let \( \pi_{t} \) denote the frame projection for \( U_{t} := \psi_{t,k_{t}}(A_{\mathcal{H},\ell+1}) \in \mathbb{R}^{\mathcal{I}_{t} \times k_{t}} \) and

\[
A_{\mathcal{H},\ell} := \prod_{t \in \mathcal{I}(T_{I}^{\ell})} \pi_{t} A_{\mathcal{H},\ell+1}.
\]

Then we define the hierarchical leaves-to-root truncation by

\[
A_{\mathcal{H}} := A_{\mathcal{H},0}.
\]

**Algorithm 2** Leaves-to-root truncation of arbitrary tensors to \( \mathcal{H} \)-Tucker format

**Require:** Input tensor \( A \in \mathbb{R}^{I} \), dimension tree \( T_{I} \) (depth \( p > 0 \)), rank distribution \( (k_{t})_{t \in I} \).

for each singleton \( t \in \mathcal{L}(T_{I}) \) do

Compute an SVD of \( A(t) \) and store the dominant \( k_{t} \) left singular vectors in the columns of the t-frame \( U_{t} \).

end for

Compute the core tensor \( C_{\mathcal{H},p} := (U_{1}^{T}, \ldots, U_{d}^{T}) \circ A \).

for \( \ell = p - 1, \ldots, 0 \) do

Initialize \( C_{\mathcal{H},\ell+1} := C_{\mathcal{H},\ell+1} \).

for each mode cluster \( t \in \mathcal{I}(T_{I}^{\ell}) \) on level \( \ell \) do

Compute an SVD of \( (C_{\mathcal{H},\ell+1})^{(t)} \) and store the dominant \( k_{t} \) left singular vectors in the columns of the t-frame \( U_{t} \in \mathbb{R}^{k_{t1} \times k_{t2} \times k_{t}} \). Let \( U_{t_{1}} \) and \( U_{t_{2}} \) denote the corresponding frames for the successors \( t_{1}, t_{2} \) of \( t \) on level \( \ell + 1 \). Compute the entries of the transfer tensor

\[
(B_{t})_{i,j,\nu} := \langle (U_{t_{1}})_{i} \otimes (U_{t_{2}})_{\nu} \rangle.
\]

Update the core tensor \( C_{\mathcal{H},\ell} := U_{t}^{T} \circ_{t} C_{\mathcal{H},\ell} \).

end for

end for

return \( \mathcal{H} \)-Tucker representation \( ((U_{t})_{t \in \mathcal{L}(T_{I})}, (B_{t})_{t \in \mathcal{I}(T_{I})}) \) for \( A_{\mathcal{H}} \in \mathcal{H} \)-Tucker\( ((k_{t})_{t \in I}) \).

**Lemma 26 (Complexity of leaves-to-root truncation)** The complexity of Algorithm 2 for a tensor \( A \in \mathbb{R}^{I} \) and dimension tree \( T_{I} \) of depth \( p > 0 \) is bounded by

\[
\mathcal{O}\left( \sum_{\mu=1}^{d} n_{\mu} \prod_{\nu=1}^{d} n_{\nu} + dk^{2} \prod_{\nu=1}^{d} n_{\nu} \right), \quad k := \max_{t \in T_{I}} k_{t}.
\]
**Proof:** For all leaves \( t = \{ \mu \} \) we have to compute the singular value decompositions of \( A^{(\mu)} \) which is of complexity \( (C_{SVD} \) being again the generic constant for the SVD)

\[
\sum_{\mu=1}^{d} C_{SVD} n_{\mu}^2 \prod_{\nu \neq \mu} n_{\nu} = C_{SVD} \sum_{\mu=1}^{d} n_{\mu} \prod_{\nu=1}^{d} n_{\nu}.
\]

For all other levels \( \ell = 0, \ldots, p - 1 \) we have to compute SVDs of matrices of size at most \( k_{t_1} k_{t_2} \times \prod_{\nu \in t} n_{\nu} \). The complexity for this is at most

\[
C_{SVD} k_{t_1}^2 k_{t_2}^2 \prod_{\nu \notin t} n_{\nu} \leq C_{SVD} k_{t_1} k_{t_2} \prod_{\nu = 1}^{d} n_{\nu} \leq C_{SVD} k_2^2 \prod_{\nu = 1}^{d} n_{\nu}.
\]

Summing this up over all nodes of the tree yields the estimate. \( \square \)

**Theorem 27 (Leaves-to-root truncation)** Let \( T_I \) be a complete binary dimension tree and \( A \in \mathbb{R}^I \). Let \( A^{\text{best}} \) denote the best approximation of \( A \) in \( \mathcal{H} \)-Tucker((\( k_t \))\( t \in I \)). Then the error of the Leaves-to-Root truncation \( A_{\bar{R}} \) (Algorithm 2) is bounded by

\[
\|A - A_{\bar{R}}\| \leq (2 + \sqrt{2})\sqrt{d} \|A - A^{\text{best}}\|.
\]

**Proof:** The first truncation step on level \( \ell = p \) is the Tucker truncation which yields \( t \)-frames \( U_t \) for all nodes \( t \in T_k^p \) and an error bound of the form

\[
\|A - A_{\bar{R},p}\| = \|A - \prod_{\ell \in T_k^p} \pi_t A\| \leq \sqrt{2^p} \|A - A^{\text{best}}\|,
\]

where \( A^{\text{best}} \) is the best approximation (worse than the one-level best approximation) in \( \mathcal{H} \)-Tucker((\( k_t \))\( t \in I \)). On any level \( \ell = p - 1, \ldots, 0 \) we construct the \( t \)-frames \( U_t \) for all nodes \( t \in T_k^\ell \) that yield a Tucker truncation of \( A_{\bar{R},\ell+1} \) the error of which is bounded in terms of the best possible approximation \( A^{\text{best}}_\ell \) of \( A_{\bar{R},\ell+1} \) using frames on level \( \ell \):

\[
\|A_{\bar{R},\ell+1} - A_{\bar{R},\ell}\| \leq \sqrt{2^\ell} \|A_{\bar{R},\ell+1} - A^{\text{best}}_\ell\|.
\]

Now let \( \pi_t^*, t \in T_k^\ell \) be projections that yield the best approximation of \( A \) in the Tucker format defined by the nodes \( t \) and ranks \( k_t \) on level \( \ell \) of the dimension tree. Then \( \prod_{\ell \in T_k^\ell} \pi_t^* A \) fulfills the rank bound on level \( \ell \) and due to Lemma 20 also the additional projection to the finer nodes \( \prod_{\ell = \ell+1} \prod_{t \in T_k^\ell} \pi_t \prod_{t \in T_k^\ell} \pi_t^* A \) fulfills the rank bound. This constructed approximation is not better than the best approximation on level \( \ell \):

\[
\|A_{\bar{R},\ell+1} - A^{\text{best}}_\ell\| \leq \|\prod_{\ell = \ell+1} \prod_{t \in T_k^\ell} \pi_t A - \prod_{\ell = \ell+1} \prod_{t \in T_k^\ell} \pi_t \prod_{t \in T_k^\ell} \pi_t^* A\|
\leq \|A - \prod_{t \in T_k^\ell} \pi_t^* A\| \leq \|A - A^{\text{best}}\|.
\]

Thus we can estimate

\[
\|A - A_{\bar{R}}\| \leq \|A - A_{\bar{R},p}\| + \sum_{\ell = 1}^{p-1} \|A_{\bar{R},\ell+1} - A_{\bar{R},\ell}\|
\leq (\sqrt{2^p} + \sum_{\ell = 1}^{p-1} \sqrt{2^\ell}\|A - A^{\text{best}}\| \leq (2 + \sqrt{2})\sqrt{d} \|A - A^{\text{best}}\|.
\]

\( \square \)
4 Truncation of Hierarchical Tucker Tensors

In this Section we want to derive an efficient realization of the truncation procedures from the previous section for the special case that the input tensor is already given in a data-sparse format, namely the hierarchical Tucker format. For the canonical rank format (as input) one can as well derive such an efficient truncation.

**Definition 28 (Brother of a mode cluster)** Let $T_I$ be a dimension tree and $t \in T_I$ a non-root mode cluster with father $f$. Then we define the unique mode cluster $\bar{t} \in T_I$ such that $f = t \sqcup \bar{t}$ as the brother of $t$.

**Lemma 29** Let $T_I$ a dimension tree and $t \in \mathcal{I}(T_I)$ an interior node with two successors $t = t_1 \sqcup t_2$. Further, let

$$A^{(t)} = \sum_{\nu=1}^{k} u_{\nu} v_{\nu}^T$$

be a matricization of $A$. Let

$$u_{\nu} = \sum_{j=1}^{k_1} \sum_{l=1}^{k_2} c_{\nu,j,l} x_j \otimes y_l, \quad x_j \in \mathbb{R}^{T_{t_1}}, y_l \in \mathbb{R}^{T_{t_2}}, \quad \nu = 1, \ldots, k$$

be a representation of the $u_{\nu}$. Then the matricization of $A$ with respect to $t_1$ is given by

$$A^{(t_1)} = \sum_{j=1}^{k_1} x_j \left( \sum_{\nu=1}^{k} \sum_{l=1}^{k_2} c_{\nu,j,l} y_l \otimes v_{\nu} \right)^T.$$

**Proof:** For the first matricization holds

$$A_{(t_1)\ldots(t_d)} = A_{(t_1)\ldots(t_p)\ldots(t_d)}^{(t)} = \sum_{\nu=1}^{k} \sum_{j=1}^{k_1} \sum_{l=1}^{k_2} c_{\nu,j,l} (x_j)_{(t_\nu)\nu \in t_1} (y_l)_{(t_\nu)\nu \in t_2} (v_{\nu})_{(t_\nu)\nu \in t_3}.$$

**Lemma 30 (Matricization of tensors in hierarchical Tucker format)** Let $T_I$ be a dimension tree, $(k_\nu)_{\nu \in I}$ a rank distribution, $A \in \mathbb{H}$-Tucker($(k_\nu)_{\nu \in I}$) with nested orthogonal frame tree $(U_\nu)_{\nu \in T_I}$ and corresponding transfer tensors $(B_\nu)_{\nu \in T_I}$. Let $t \in T_I^{(p)}$, $p \geq 1$, and $\text{Root}(T_I) = t_0, t_1, \ldots, t_{p-1}, t_p = t$ a path of length $p$. Let $U^1, \ldots, U^p$ denote the frames of the corresponding brothers, $B^0, \ldots, B^{p-1}$ the corresponding transfer tensors and $k_0, \ldots, k_p$ the corresponding ranks. For convenience of notation we assume that the brother $I_\ell$ is always the first and $t_\ell$ the second successor, i.e.

$$(U_{t_\nu})_{\nu} = \sum_{j} B_{\nu,i,j} U^1_{i} \otimes U_{t_{\nu+1}}.$$
Then the t-matricization has the form

\[ A^{(t)} = \sum_{\nu=1}^{k_t} (U^{(t)}_{\nu}(V^{(t)}_{\nu})^T = U^{(t)}V^{(t)}^T, \]

where the complementary frame \( V_t \) is defined by

\[ (V_t)_{jp} = \left( \sum_{i_1=1}^{k_1} \sum_{j_1=1}^{k_1} \cdots \sum_{i_p=1}^{k_p} \sum_{j_p=1}^{k_p} B^0_{i_1,j_1} \cdots B^{p-1}_{i_p,j_p} \right) \bar{U}^{1}_{i_1} \otimes \cdots \otimes \bar{U}^p_{j_p}. \]

**Proof:** We prove the statement by induction over the level \( p \) of the mode cluster \( t \). The start \( p = 1 \) is trivial: the tensor \( A \) has the representation (Lemma 29)

\[ A = \sum_{i=1}^{k_t} \sum_{j=1}^{k_t} B^0_{i,j} \bar{U}^1_{i} \otimes \bar{U}^1_{j}, \quad A^{(t)} = \sum_{j=1}^{k_t} U^1_{jp} \left( \sum_{i=1}^{k_t} B^0_{i,j} \bar{U}^1_{i} \right)^T. \]

For the node \( t_{p-1} \) holds by induction

\[ A^{(t_{p-1})} = \sum_{\nu=1}^{k_{p-1}} U^{p-1}_{\nu}(V^{p-1}_{\nu})^T, \quad U^{p-1}_{\nu} = \sum_{i_p=1}^{k_p} \sum_{j_p=1}^{k_p} B^{p-1}_{\nu,i_p,j_p} \bar{U}^p_{i_p} \otimes U^p_{j_p}. \]

Together we obtain by Lemma 29

\[ A^{(t_{p-1})} = \sum_{j_p=1}^{k_p} U^p_{j_p} \left( \sum_{\nu=1}^{k_{p-1}} \sum_{i_p=1}^{k_p} B^{p-1}_{\nu,i_p,j_p} \bar{U}^p_{i_p} \otimes V^{p-1}_{\nu} \right)^T = \sum_{j_p=1}^{k_p} U^p_{j_p} (V^{p}_{j_p})^T. \]

---

**Definition 31 (Accumulated transfer tensors)** Let \( T_t \) be a dimension tree, \((k_t)_{t \in T} \) a rank distribution, \((B_t)_{t \in T_t} \) transfer tensors. Let \( t \in T^{(p)}_t, p \geq 1 \), and \( \text{Root}(T_t) = t_0, t_1, \ldots, t_{p-1}, t_p = t \) a path of length \( p \). Let \( B^0, \ldots, B^{p-1} \) denote the corresponding transfer tensors (assuming that the brother \( t_i \) is always the first and \( t_j \) the second successor). Let \( k_0, \ldots, k_p \) be the corresponding ranks and \( \bar{k}_0, \ldots, \bar{k}_p \) the ranks of the brothers. Then we define the accumulated transfer tensor \( \hat{B}_t \) by

\[ (\hat{B}^1)_{i_1, j_1} := \sum_{i_1=1}^{k_1} B^0_{i_1,j_1} B^0_{i_1,s_1}, \]

\[ (\hat{B}^\ell)_{i_\ell, s_\ell} := \sum_{s_{\ell-1}=1}^{k_{\ell-1}} \sum_{i_{\ell-1}=1}^{k_{\ell-1}} (\hat{B}^{\ell-1})_{i_{\ell-1}, s_{\ell-1}} B^{\ell-1}_{i_{\ell-1}, i_\ell, s_\ell}, \]

\[ \hat{B}_t := \hat{B}^p. \]

**Remark 32** The first accumulated tensors \( \hat{B}_{t_1}, \hat{B}_{t_2} \) for the two sons of the root \( t \) can be computed in \( O(k_1k_t,k_t) \) each. For each further node the second formula in Definition 31 has to be applied and it involves inside the bracket a matrix multiplication of complexity
Lemma 33 (Gram matrices of complementary frames) Let $T_Z$ be a dimension tree, $(k_t)_{t \in T}$ a rank distribution, $A \in \mathcal{H}$-Tucker($(k_t)_{t \in T}$) with nested orthogonal frame tree $(U_t)_{t \in T_Z}$ and corresponding transfer tensors $(B_t)_{t \in T_Z}$. For each $t \in T_Z$ let $V_t$ be the complementary frame from Lemma 30. Then $\hat{B}_t$ is the Gram matrix for $V_t$:

$$V_t^T V_t = \hat{B}_t, \quad \langle (V_t)_\nu, (V_t)_\mu \rangle = \langle \hat{B}_t \rangle_{\nu, \mu}.$$ 

Proof: We use the definitions and notations from Lemma 30. According to Lemma 30 and due to the orthogonality of each of the $k$-frames $U_t$ we obtain

$$\langle (V_t)_\nu, (V_t)_\mu \rangle = \sum_{i=1}^{k_1} \sum_{j=1}^{k_2} \sum_{s=1}^{k_3} \cdots \sum_{i_1=1}^{k_1} \sum_{i_2=1}^{k_1} \sum_{i_3=1}^{k_1} \sum_{i_4=1}^{k_1} \sum_{i_5=1}^{k_1} B^0_{1, i_1, i_2} \cdots B^{p-2}_{j_1, i_1, i_2} B^0_{j_1, i_1, i_2} \cdots B^{p-2}_{j_1, i_1, i_2, \mu} B^0_{1, i_1, i_2, \mu} B_{1, i_1, i_2, \mu} = \langle \hat{B}_t \rangle_{\nu, \mu}.$$ 

According to the previous Lemma we can easily compute the left singular vectors of $V_t^T$ which are the eigenvectors of the $k_t \times k_t$ matrix $\hat{B}_t$. The matrix $Q_t$ of singular vectors is the transformation matrix such that $U_t Q_t$ is the matrix of the left singular vectors of $A^{(t)}$ the singular values of which are the square roots of the eigenvalues of $\hat{B}_t$. Thus, one can truncate either to fixed rank or one can determine the rank adaptively in order to guarantee a truncation accuracy of $\varepsilon$.

The nested mode frames were required to be orthogonal. If this is not yet the case, one has to orthogonalize the frame tree. The procedure for this is explained next and the complexity is estimated afterwards.

Lemma 34 (Frame transformation) Let $t \in T_Z$ be a mode cluster with $t$-frame $U_t$, transfer tensor $B_t$ and two sons $t_1, t_2$ with frames $U_{t_1}, U_{t_2}$, such that the columns fulfill

$$(U_t)_i = \sum_{j=1}^{k_1} \sum_{l=1}^{k_2} (B_t)_{i,j,l} (U_{t_1})_j \otimes (U_{t_2})_l, \quad i = 1, \ldots, k.$$ 

Let $X \in \mathbb{R}^{k_1 \times k_2}, Y \in \mathbb{R}^{k_1 \times k_1}, Z \in \mathbb{R}^{k_2 \times k_2}$. Then we can rewrite the transformed frames as

$$(U_t X)_i = \sum_{j=1}^{k'_1} \sum_{l=1}^{k'_2} (B'_t)_{i,j,l} (U_{t_1} Y)_j \otimes (U_{t_2} Z)_l, \quad B'_t := \langle X, Y^{-1}, Z^{-1} \rangle \circ B_t.$$
Algorithm 3 Orthogonalization of hierarchical Tucker tensors

**Require:** Input tensor $A \in \mathcal{H}$-Tucker($((k_t)_{t \in T})$) represented by $((U_t)_{t \in \mathcal{L}(T_1)}, (B_t)_{t \in \mathcal{I}(T_2)})$.

for each singleton $t \in \mathcal{L}(T_1)$ do

- Compute a QR-decomposition of the $t$-frame $U_t$ and define $U_t := Q$, $B_f := \begin{cases} (I, I, R) \circ B_f & \text{if } t \text{ is the second successor}, \\ (I, R, I) \circ B_f & \text{if } t \text{ is the first successor} \end{cases}$

for the father $f$ of $t$.
end for

for each mode cluster $t \in \mathcal{I}(T_2)$ do

- Compute a QR-decomposition of $B_t^{1,2}$:
  \[
  (B_t)^{1,2} = (Q_t)^{1,2}) R,
  \]
  and set $B_t := Q_t$, $B_f := \begin{cases} (I, I, R) \circ B_f & \text{if } t \text{ is the second successor}, \\ (I, R, I) \circ B_f & \text{if } t \text{ is the first successor} \end{cases}$

of the father $f$ of $t$.
end for

return nested orthogonal frames $(U_t)_{t \in \mathcal{L}(T_1)}$ and transfer tensors $(B_t)_{t \in \mathcal{I}(T_2)}$.

**Proof:** The formula follows from elementary matrix multiplications.

**Lemma 35 (Complexity for the orthogonalization of nested frame trees)** The complexity of Algorithm 3 for a tensor $A \in \mathcal{H}$-Tucker($((k_t)_{t \in T})$) with nested frames $(U_t)_{t \in \mathcal{L}(T_1)}$ and transfer tensors $(B_t)_{t \in \mathcal{I}(T_2)}$ is bounded by

\[
O \left( \sum_{\mu=1}^d n_\mu k_\mu^2 + \sum_{t \in \mathcal{I}(T_2), \text{Sons}(t)=\{t_1,t_2\}} k_{t_1}^2 k_{t_2} + k_{t_1} k_{t_2}^2 + k_{t_1} k_{t_2}^2 + k_{t_1} k_{t_2}^2 \right).
\]

**Proof:** For each interior node we have to compute QR decompositions which are of complexity $O(k_{t_1}^2 k_{t_2})$ and perform two mode multiplications $X \circ_\mu B_f$, $\mu = 2, 3$, which is of complexity $O(k_{t_1} k_{t_2}^2 + k_{t_1}^2 k_{t_2}^2)$.

For the leaves $t = \{\mu\}$ a QR-factorization is of complexity $O(n_\mu k_\mu^2)$. The sum over all nodes of the tree yields the desired bound.

**Lemma 36 (Complexity for the $\mathcal{H}$-Tucker truncation)** The complexity for the truncation of an $\mathcal{H}$-Tucker($((k_t)_{t \in T})$)-Tensor $A$ (not necessarily with orthogonal frames) to lower rank is

\[
O(d \max_{t \in T} k_t^4 + \sum_{\mu=1}^d n_\mu k_\mu^2).
\]

**Remark 37 (Converting elementary tensors into $\mathcal{H}$-Tucker format)** Let $A \in \mathbb{R}^T$ be a tensor represented by a short sum of elementary tensors:

\[
A = \sum_{i=1}^k \bigotimes_{\mu=1}^d a_{i,\mu}, \quad a_{i,\mu} \in \mathbb{R}^{I_\mu}.
\]


Then $A$ can immediately be represented in the hierarchical Tucker format by the $t$-frames

$$\forall t = \{\mu\} \in \mathcal{L}(T_T) : (U_t)_i := a_{i,\mu}, \quad i = 1, \ldots, k, \quad k_\mu := k,$$

and the transfer tensors

$$\forall t \in \mathcal{I}(T_T) \setminus \text{Root}(T_T) : (B_t)_{i,j,l} := \begin{cases} 1 & \text{if } i = j = l \\ 0 & \text{otherwise,} \end{cases}, \quad B_t \in \mathbb{R}^{k \times k \times k}, \quad k_t := k.$$

The root transfer tensor is

$$(B_{\{1,\ldots,d\}})_{1,j,l} := \begin{cases} 1 & \text{if } j = l \\ 0 & \text{otherwise,} \end{cases}, \quad B_{\{1,\ldots,d\}} \in \mathbb{R}^{1 \times k \times k}, \quad k_{\{1,\ldots,d\}} := 1.$$

The frames are not yet orthogonal, so a subsequent orthogonalization and truncation is advisable to find a reduced representation. If we store the transfer tensors in sparse format, then the amount of storage is $k(d-1) + \sum_{\mu=1}^{d} n_\mu k_\mu$, i.e. almost the same as for a tensor in canonical format (i.e. sum of elementary tensors).

The opposite conversion from $\mathcal{H}$-Tucker to canonical format is highly non-trivial.

### 5 Numerical Examples

The numerical examples in this section are focussed on three questions:

1. How close to the measurements are the theoretical estimates of the truncation error, i.e. the ratio between nodewise errors and the total error? In particular we are interested in the question whether or not the factor $\sqrt{d}$ appears.

2. What is the maximal attainable truncation accuracy, i.e. how close can we get to the machine precision?

3. What are problem sizes that can realistically be tackled by the $\mathcal{H}$-Tucker format in terms of the dimension $d$ and the maximal rank $k$?

All computations are performed on a hush with 2 CPUs of which only one is used for the numerical tests. The CPU peak frequency is 1.83 GHz and the available memory is 1 GB.

#### 5.1 Truncation from dense to $\mathcal{H}$-Tucker format

Our first numerical example is in $d = 5$ with mode size $n_\mu = 25$. The tensor $A$ is a dense tensor with entries

$$A_{(i_1,\ldots,i_d)} := \left( \sum_{\mu=1}^{d} i_\mu^2 \right)^{-1/2},$$

which corresponds to the discretization of the function $1/\|x\|$ on $[1, 25]^5$. The time for the conversion (Algorithm 1) of the dense tensor to $\mathcal{H}$-Tucker format $A_{\mathcal{H}}$, the amount of storage needed for the frames $U_t$ and transfer tensors $B_t$ and the obtained relative approximation accuracy $\|A - A_{\mathcal{H}}\|$ are presented in Table 1. The nodewise SVD is shown in Figure 2. From the truncation we lose roughly 2–3 digits of precision compared to the maximal attainable machine precision $\text{EPS} \approx 10^{-16}$. It seems that the nodewise rank is uniformly bounded (there is almost no variation between the ranks $k_t$) by $k \sim \log(1/\varepsilon)$.
Table 1: Converting a dense tensor to $\mathcal{H}$-Tucker format.

<table>
<thead>
<tr>
<th>$\varepsilon$</th>
<th>$|A - A_H|/|A|$</th>
<th>Storage (KB)</th>
<th>times (Sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1 \times 10^{-2}$</td>
<td>$6.0 \times 10^{-3}$</td>
<td>3.6</td>
<td>105.8</td>
</tr>
<tr>
<td>$1 \times 10^{-4}$</td>
<td>$1.2 \times 10^{-4}$</td>
<td>11.2</td>
<td>104.1</td>
</tr>
<tr>
<td>$1 \times 10^{-6}$</td>
<td>$1.1 \times 10^{-6}$</td>
<td>29.3</td>
<td>103.5</td>
</tr>
<tr>
<td>$1 \times 10^{-8}$</td>
<td>$7.8 \times 10^{-9}$</td>
<td>58.1</td>
<td>104.8</td>
</tr>
<tr>
<td>$1 \times 10^{-10}$</td>
<td>$1.7 \times 10^{-10}$</td>
<td>92.7</td>
<td>108.0</td>
</tr>
<tr>
<td>$1 \times 10^{-12}$</td>
<td>$7.2 \times 10^{-13}$</td>
<td>153.2</td>
<td>104.8</td>
</tr>
<tr>
<td>$1 \times 10^{-14}$</td>
<td>$3.2 \times 10^{-13}$</td>
<td>298.0</td>
<td>104.0</td>
</tr>
<tr>
<td>$1 \times 10^{-16}$</td>
<td>$2.7 \times 10^{-14}$</td>
<td>615.1</td>
<td>106.9</td>
</tr>
</tbody>
</table>

5.2 Truncation from canonical to $\mathcal{H}$-Tucker format

The second example is in higher dimension $d$ with mode size $n_\mu = 1000$. The entries of the tensor $A_H$ are approximations of

$$A_{(i_1,\ldots,i_d)} := \left(\sum_{\mu=1}^{d} i_\mu^2 \right)^{-1/2}, \quad i\mu = 1,\ldots,1000,$$

by exponential sums,

$$(A_E)_{(i_1,\ldots,i_d)} := \sum_{j=1}^{35} \omega_j \prod_{\mu=1}^{d} \alpha_j i_\mu^2,$$

such that each entry is accurate up to $\varepsilon_E = 10^{-10}$,

$$|A_{(i_1,\ldots,i_d)} - (A_E)_{(i_1,\ldots,i_d)}| \leq 7.315 \times 10^{-10}.$$

The weights $\omega_j$ and exponents $\alpha_j$ were obtained from W. Hackbusch and are available via the webpage ($k = 35, R = 1000000$)

http://www.mis.mpg.de/scicomp/EXP_SUM

The tensor $A_E$ (in canonical format or elementary tensor sum) is then converted to $\mathcal{H}$-Tucker format (error zero), which we denote by $A_H$. The hierarchical rank is $k_t = 35$ for every mode cluster $t \in T_I$. From this input tensor we compute truncations $A_{H,\varepsilon}$ to lower hierarchical rank by prescribing the (relative) truncation accuracy $\varepsilon$. In Tables 2 and 3 we report the accuracy $\|A_H - A_{H,\varepsilon}\|/\|A_H\|$, the storage requirements for $A_{H,\varepsilon}$ in MB as well as the time in seconds used for the truncation. We observe that the accuracy is

$$\|A_H - A_{H,\varepsilon}\|/\|A_H\| \approx 3\varepsilon$$

independent of the dimension $d$. The maximal attainable accuracy seems do be roughly $\varepsilon_{\min} \approx 10^{-13}$. 
Here, we setup an notebook computer, cf. Table 5. On a larger desktop machine one can use

5.3 Truncation of $\mathcal{H}$-Tucker tensors

The third test is not any more concerned with the approximation accuracy, but purely on the computational complexity. Here, we setup an $\mathcal{H}$-Tucker tensor with nodewise ranks $k_i \equiv k$ and mode sizes $n_{\mu} \equiv 20$. Then, we vary the rank $k$ and dimension parameter $d$ and measure the storage complexity as well as the complexity for the truncation (which is essentially independent of the target rank or accuracy). The results are reported in Table 4 (dashes mean that for that problem size we have run out of memory (1GB)). We conclude that it is indeed possible to perform reliable numerical computations in dimension $d = 1,000,000$, and also rather large ranks of $k = 50$ are not a problem for dimensions $d = 1000$ on a simple notebook computer, cf. Table 5. On a larger desktop machine one can use $k = 100$ in dimension $d = 10,000$ (uses roughly 80 GB and takes ca. 10 hours).

References


Table 4: Storage complexity (MB) and truncation time for the $\mathcal{H}$-Tucker format with mode size $n_\mu = 20$.

<table>
<thead>
<tr>
<th>k</th>
<th>d=</th>
<th>10</th>
<th>100</th>
<th>1,000</th>
<th>10,000</th>
<th>100,000</th>
<th>1,000,000</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>size</td>
<td>0.00</td>
<td>0.02</td>
<td>0.16</td>
<td>1.60</td>
<td>16.02</td>
<td>160.22</td>
</tr>
<tr>
<td>1</td>
<td>time</td>
<td>0.00</td>
<td>0.00</td>
<td>0.01</td>
<td>0.08</td>
<td>0.86</td>
<td>7.88</td>
</tr>
<tr>
<td>2</td>
<td>size</td>
<td>0.00</td>
<td>0.04</td>
<td>0.37</td>
<td>3.66</td>
<td>36.62</td>
<td>366.21</td>
</tr>
<tr>
<td>2</td>
<td>time</td>
<td>0.00</td>
<td>0.00</td>
<td>0.03</td>
<td>0.22</td>
<td>1.63</td>
<td>15.43</td>
</tr>
<tr>
<td>5</td>
<td>size</td>
<td>0.02</td>
<td>0.17</td>
<td>1.71</td>
<td>17.16</td>
<td>171.66</td>
<td>-</td>
</tr>
<tr>
<td>5</td>
<td>time</td>
<td>0.00</td>
<td>0.02</td>
<td>0.08</td>
<td>0.81</td>
<td>7.60</td>
<td>-</td>
</tr>
<tr>
<td>10</td>
<td>size</td>
<td>0.08</td>
<td>0.90</td>
<td>9.14</td>
<td>91.54</td>
<td>915.51</td>
<td>-</td>
</tr>
<tr>
<td>10</td>
<td>time</td>
<td>0.01</td>
<td>0.10</td>
<td>0.61</td>
<td>5.79</td>
<td>55.86</td>
<td>-</td>
</tr>
<tr>
<td>20</td>
<td>size</td>
<td>0.52</td>
<td>6.29</td>
<td>63.97</td>
<td>640.75</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>20</td>
<td>time</td>
<td>0.20</td>
<td>1.23</td>
<td>7.36</td>
<td>72.12</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 5: Storage complexity (MB) and truncation time for the $\mathcal{H}$-Tucker format with mode size $n_\mu = 100$.

<table>
<thead>
<tr>
<th>k</th>
<th>d=</th>
<th>10</th>
<th>100</th>
<th>1,000</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>size</td>
<td>1.15</td>
<td>13.59</td>
<td>138.05</td>
</tr>
<tr>
<td>25</td>
<td>time</td>
<td>0.24</td>
<td>1.97</td>
<td>19.39</td>
</tr>
<tr>
<td>50</td>
<td>size</td>
<td>8.03</td>
<td>97.29</td>
<td>989.93</td>
</tr>
<tr>
<td>50</td>
<td>time</td>
<td>2.60</td>
<td>30.20</td>
<td>306.02</td>
</tr>
<tr>
<td>100</td>
<td>size</td>
<td>68.74</td>
<td>755.39</td>
<td>-</td>
</tr>
<tr>
<td>100</td>
<td>time</td>
<td>57.05</td>
<td>685.98</td>
<td>-</td>
</tr>
</tbody>
</table>


Preprint Series DFG-SPP 1324

http://www.dfg-spp1324.de

Reports


