A Matlab toolbox for tensors in hierarchical Tucker format

Christine Tobler and Daniel Kressner
Seminar for applied mathematics
ETH Zürich

c tobler@math.ethz.ch
ProDoc Workshop Disentis, 19.08.2011
Motivation

Low-rank matrix:

\[ X = U V^T \]

- Reduced storage cost: \( O(n_1 r + n_2 r) \) instead of \( O(n_1 n_2) \)
- Basic operations (addition, multiplication by a matrix) possible
- Best lower-rank approximation by truncated SVD

We define a tensor as an array \( \mathcal{X} \in \mathbb{R}^{n_1 \times \cdots \times n_d} \)

Goal:
- Concept of low-rank tensor, similar to that of low-rank matrix
- Storage cost linearly dependent on \( d \)
Contents

CP and Tucker formats
Hierarchical Tucker format
MATLAB \texttt{htucker} toolbox

Basic operations
- Matrix $\times$ tensor, addition
- Orthogonalization
- Inner product

Advanced operations
- Truncation of explicitly given tensor
- Truncation of $\mathcal{H}$-Tucker tensor

Conclusions
CP and Tucker formats
CP format

\[ \text{vec}(\mathcal{X}) = \sum_{j=1}^{R} u_j^{(d)} \otimes \cdots \otimes u_j^{(1)}, \quad u_j^{(\mu)} \in \mathbb{R}^{n_\mu}. \]

Pro and Contra

+ Storage requirements linear in \( d \): \( dnR \)

– Approximation in CP format cumbersome
Tucker format

\[ \text{vec}(\mathcal{X}) = \sum_{j_1=1}^{r_1} \cdots \sum_{j_d=1}^{r_d} C_{j_1, \ldots, j_d} u^{(d)}_{j_d} \otimes \cdots \otimes u^{(1)}_{j_1} = (U_d \otimes \cdots \otimes U_1)\text{vec}(\mathcal{C}), \]

with \( \mathcal{C} \in \mathbb{R}^{r_1 \times \cdots \times r_d} \), \( U_{\mu} = [u^{(\mu)}_{1}, \ldots, u^{(\mu)}_{r_\mu}] \in \mathbb{R}^{n_\mu \times r_\mu} \).

Pro and Contra

+ Efficient quasi-best approximation in Tucker format (HOSVD)

- Storage requirements exponential in \( d \): \( dnr + r^d \)
Hierarchical Tucker format


Matricization

1-mode matricization:

\[ X \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d} \]

\[ X^{(1)} \in \mathbb{R}^{n_1 \times (n_2 n_3 \cdots n_d)} \]

\( \mu \)-mode matricization:

\[ X^{(\mu)} \in \mathbb{R}^{n_\mu \times (n_1 \cdots n_{\mu-1} n_{\mu+1} \cdots n_d)}, \quad \mu = 1, \ldots, d. \]

General matricization for mode decomposition \( \{1, \ldots, d\} = t \cup s \):

\[ X^{(t)} \in \mathbb{R}^{(n_{t_1} \cdots n_{t_k}) \times (n_{s_1} \cdots n_{s_{d-k}})} \]

with \( \left( X^{(t)} \right)_{(i_{t_1}, \ldots, i_{t_k}), (i_{s_1}, \ldots, i_{s_{d-k}})} := x_{i_1, \ldots, i_d}. \)
Hierarchical construction

**Definition** of $\mathcal{H}$-Tucker rank-$\{r_t\}_{t \in T}$ tensor:

$$\mathcal{H} - \text{Tucker}(r_t)_{t \in T} = \{ X \in \mathbb{R}^{n_1 \times \cdots \times n_d} : \text{rank}(X(t)) \leq r_t \ \forall t \in T \}$$

Singular value decomposition: $X(t) = U_t \Sigma_t U_t^T$.

Column spaces are nested $\leadsto$

$$t = t_1 \cup t_2 \ \Rightarrow \ \text{span}(U_t) \subset \text{span}(U_{t_2} \otimes U_{t_1})$$

$$\Rightarrow \ \exists B_t : U_t = (U_{t_2} \otimes U_{t_1}) B_t.$$ 

Size of $U_t$:

$$U_t \in \mathbb{R}^{n_1 \cdots n_k \times r_t} \quad \text{with} \quad r_t = \text{rank}(X(t)).$$

Size of $B_t$:

$$B_t \in \mathbb{R}^{r_1 r_2 \times r_t} \quad \text{with} \quad r_t = \text{rank}(X(t)).$$
Hierarchical construction

Singular value decomposition: $X^{(t)} = U_t \Sigma_t U_s^T$.

$$t = t_1 \cup t_2 \implies \text{span}(U_t) \subset \text{span}(U_{t_2} \otimes U_{t_1})$$

$$\implies \exists B_t : U_t = (U_{t_2} \otimes U_{t_1})B_t.$$

For $d = 4$:

$$U_{12} = (U_2 \otimes U_1)B_{12}$$

$$U_{34} = (U_4 \otimes U_3)B_{34}$$

$$\text{vec}(X) = X^{(1234)} = (U_{34} \otimes U_{12})B_{1234}$$

$$\implies \text{vec}(X) = (U_4 \otimes U_3 \otimes U_2 \otimes U_1)(B_{34} \otimes B_{12})B_{1234}.$$
Dimension tree

Tree structure for $d = 4$:

\[ U_1 \quad (n_1 \times r_1) \quad B_{12} \quad (r_1 r_2 \times r_{12}) \]
\[ U_2 \quad (n_2 \times r_2) \quad B_{1234} \quad (r_{12} r_{34} \times 1) \]
\[ U_3 \quad (n_3 \times r_3) \quad B_{34} \quad (r_3 r_4 \times r_{34}) \]
\[ U_4 \quad (n_4 \times r_4) \]

Reshape:

\[ B_{12} \in \mathbb{R}^{r_1 r_2 \times r_{12}} \implies B_{12} \in \mathbb{R}^{r_1 \times r_2 \times r_{12}} \]
\[ B_{34} \in \mathbb{R}^{r_3 r_4 \times r_{34}} \implies B_{34} \in \mathbb{R}^{r_3 \times r_4 \times r_{34}} \]
\[ B_{1234} \in \mathbb{R}^{r_{12} r_{34} \times 1} \implies B_{1234} \in \mathbb{R}^{r_{12} \times r_{34}} \]
Storage requirements for general $d$: 

\[ \mathcal{O}(d nr) + \mathcal{O}(dr^3), \]

where $r = \max\{r_t\}$, $n = \max\{n_\mu\}$. 
Tensor network notation

**Tensor network**: Collection of tensors connected by tensor contractions.

**Examples:**
MATLAB h Tucker toolbox
MATLAB \texttt{htucker} toolbox

\url{www.sam.math.ethz.ch/NLAgroup/htucker_toolbox.html}

\textbf{Hierarchical Tucker Toolbox}

A MATLAB Toolbox for the construction and manipulation of tensors in the Hierarchical Tucker (H-Tucker) format, see references [1-3]. The H-Tucker format is an approximate SVD-based data-sparse representation of a tensor, admitting the storage of higher-order tensors. It has similarities with the Tucker decomposition, but avoids exponential growth of storage requirements inherent in the Tucker format.

Download Hierarchical Tucker Toolbox

\texttt{htucker\_toolbox.tar.gz, htucker\_toolbox.zip} (Version 0.8, January 2011)
This code is research code and not intended for production use.

\begin{itemize}
\item \texttt{htucker} provides MATLAB class for storing and manipulating tensor in low-rank format.
\item Several operations overloaded (\texttt{+ - .\!* norm ...})
\item Set of frequently used utilities.
\item \textbf{Main goal:} Painless experimentation with algorithms.
\end{itemize}
Existing MATLAB Toolboxes

- **MATLAB Tensor Toolbox** by T. Kolda and B. Bader
  csmr.ca.sandia.gov/~tkgkolda/TensorToolbox

- **TT Toolbox** by I. Oseledets
  spring.inm.ras.ru/osel/?p=31
Basic functionality for MATLAB class \texttt{htensor}

\begin{verbatim}
x = htenones([4 5 6 7]) constructs htensor of size 4 x 5 x 6 x 7, with all entries one.
x = htenrandn([4 5 6 7]) constructs htensor of size 4 x 5 x 6 x 7, with random ranks and random entries.
x(1, 3, 4, 2) returns entry of \(X\).
x(1, 3, :, :) returns slice of \(X\) as an htensor.
full(x) returns full tensor represented by \(X\). (use with care)
spy(x) displays spy plots of \(U_t, B_t\), on the dimension tree.
disp_tree(x) returns tree structure/ranks:
\end{verbatim}

\begin{verbatim}
ans is an htensor of size 4 x 5 x 6 x 7
  1-4  1; 6 3 1
  1-2  2; 3 4 6
  1    4; 4 3
  2    5; 5 4
  3-4  3; 3 3 3
  3    6; 6 3
  4    7; 7 3
\end{verbatim}
Singular value tree

\texttt{plot\_sv}(x) plots singular values of corresponding matricizations in the dimension tree of a tensor $\mathcal{X}$.

Example: Singular value tree of an order 5 tensor.
Basic operations

- Matrix $\times$ tensor
- Addition
- Inner product
Matrix $\times$ tensor

Application of matrix $A \in \mathbb{R}^{m \times n_\mu}$ to mode $\mu$ of $X \in \mathbb{R}^{n_1 \times \cdots \times n_d}$:

$$Y = A \circ_\mu X \iff Y^{(\mu)} = AX^{(\mu)}.$$  

Nearly trivial if $X$ is in $\mathcal{H}$-Tucker format:

$$A \circ_\mu X = A \circ_\mu ((U_1, \ldots, U_d) \circ C) = (U_1, \ldots, U_{\mu-1}, AU_\mu, U_{\mu+1}, \ldots, U_d) \circ C$$

- Almost no operations required.
- Ranks stay the same.
- Orthogonality destroyed.

```
ttm(x, A, 2) applies matrix A to htensor X in mode 2.
y = ttm(x, {A, B, C}, [2, 3, 4])
y = ttm(x, @(x)(fft(x)), 2) applies FFT in mode 2.
```
Addition of low-rank matrices

Addition of two matrices in low-rank format:

\[ A = U_1 \Sigma_A U_2^T, \quad B = V_1 \Sigma_B V_2^T \]

\[ \Rightarrow \]

\[ A + B = \begin{bmatrix} U_1 & V_1 \end{bmatrix} \begin{bmatrix} \Sigma_A & 0 \\ 0 & \Sigma_B \end{bmatrix} \begin{bmatrix} U_2 \\ V_2 \end{bmatrix}^T \]

- No operations required.
- Rank increases.
- Orthogonality destroyed.
Addition of low-rank tensors

Addition of four tensors $X_1, X_2, X_3, X_4$ in $\mathcal{H}$-Tucker format:

$$X_1 + X_2 + X_3 + X_4.$$

Proceed as in matrix case by embedding factors in larger matrices.

- No operations required.
- $\mathcal{H}$-Tucker rank increases.
- Orthogonality destroyed.

Command in $\texttt{htucker}$: $x1 + x2 + x3 + x4$
\[ U_1^{[1]} U_1^{[2]} U_1^{[3]} U_1^{[4]} \]

\[ U_2^{[1]} U_2^{[2]} U_2^{[3]} U_2^{[4]} \]

\[ U_3^{[1]} U_3^{[2]} U_3^{[3]} U_3^{[4]} \]

\[ U_4^{[1]} U_4^{[2]} U_4^{[3]} U_4^{[4]} \]
Orthogonalization

Any tensor $\mathcal{X}$ in $\mathcal{H}$-Tucker format can be orthogonalized in the sense that all factors in the dimension tree, except for the root node, contain orthonormal columns, $U_t^T U_t = I$.

Example: $\text{vec}(\mathcal{X}) = (U_4 \otimes U_3 \otimes U_2 \otimes U_1)(B_{34} \otimes B_{12})B_{1234}$.

Step 1: QR decompositions $U_t = Q_t R_t \rightsquigarrow$

$$\text{vec}(\mathcal{X}) = (Q_4 \otimes Q_3 \otimes Q_2 \otimes Q_1)(\tilde{B}_{34} \otimes \tilde{B}_{12})B_{1234}$$

with $\tilde{B}_{34} := (R_4 \otimes R_3)B_{34}$, $\tilde{B}_{12} := (R_2 \otimes R_1)B_{12}$.

Step 2: QR decompositions $\tilde{B}_{34} = Q_{34} R_{34}, \tilde{B}_{12} = Q_{12} R_{12} \rightsquigarrow$

$$\text{vec}(\mathcal{X}) = (Q_4 \otimes Q_3 \otimes Q_2 \otimes Q_1)(Q_{34} \otimes Q_{12})\tilde{B}_{1234}$$

with $\tilde{B}_{1234} := (R_{34} \otimes R_{12})B_{1234}$.

Compt. requirements for general $d$: $O(dnr^2) + O(dr^4)$.

Command in $\texttt{htucker}$: $x = \text{orthog}(x)$
Inner product

Inner product of two tensors $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$:

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \langle \text{vec}(\mathcal{X}), \text{vec}(\mathcal{Y}) \rangle = \sum_{i_1=1}^{n_1} \cdots \sum_{i_d=1}^{n_d} x_{i_1, \ldots, i_d} y_{i_1, \ldots, i_d}.$$

Can be performed efficiently in $\mathcal{H}$-Tucker, provided that $\mathcal{X}, \mathcal{Y}$ have compatible dimension trees.

- **htucker command:** `innerprod(x, y)`
- **Overall cost:** $O(dnr^2) + O(dr^4)$. 
Computation of inner product

\[ \langle X, Y \rangle = \sum_{i_1=1}^{n_1} \cdots \sum_{i_d=1}^{n_d} x_{i_1,...,i_d} y_{i_1,...,i_d}. \]
Computation of inner product

\[ \langle X, Y \rangle = \sum_{i_1=1}^{n_1} \cdots \sum_{i_d=1}^{n_d} x_{i_1, \ldots, i_d} y_{i_1, \ldots, i_d}. \]
Computation of inner product
Computation of inner product
Computation of inner product
Computation of inner product
Advanced operations

- Truncation of explicitly given tensor
- Truncation of $\mathcal{H}$-Tucker tensor
Truncation of explicit tensor

Let $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$ be explicitly given.

- For each tree node $t$, let $W_t$ contain the $r_t$ dominant left singular vectors of $X^{(t)}$ and define projection

$$\pi_t \mathcal{X} = W_t W_t^H \circ_t \mathcal{X} \iff \pi_t X^{(t)} = W_t W_t^H X^{(t)}.$$

- Truncated tensor:

$$\tilde{\mathcal{X}} := \left( \prod_{t \in T_\ell} \pi_t \right) \cdots \left( \prod_{t \in T_1} \pi_t \right) \mathcal{X},$$

where $T_\ell$ contains all nodes on level $\ell$.

- [Grasedyck’2010]: $\| \mathcal{X} - \tilde{\mathcal{X}} \| \leq \sqrt{2d - 3} \| \mathcal{X} - \mathcal{X}_{\text{best}} \|.$
Truncation of explicit tensor

Example:

\[
\text{vec} \tilde{\mathcal{X}} = (W_4 W_4^H \otimes W_3 W_3^H \otimes W_2 W_2^H \otimes W_1 W_1^H)(W_{34} W_{34}^H \otimes W_{12} W_{12}^H)\text{vec} \mathcal{X}
\]

\[
= (W_4 \otimes W_3 \otimes W_2 \otimes W_1) \cdots \]

\[
\underbrace{([W_4^H \otimes W_3^H] W_{34}^H \otimes [W_2^H \otimes W_1^H] W_{12}^H)}_{=: B_{34}} \underbrace{([W_{34}^H \otimes W_{12}^H] \text{vec} \mathcal{X})}_{=: B_{1234}}.
\]

opts.max_rank = 10 maximal rank at truncation.
opts.rel_eps = 1e-6 maximal relative truncation error.
opts.abs_eps = 1e-6 maximal absolute truncation error.
Condition max_rank takes precedence over rel_eps and abs_eps.
x_t = htensor.truncate_rtl(x, opts) returns truncated tensor \( \mathcal{X} \) of a multidimensional array.

Remark: There is also a significantly faster htensor.truncate_ltr (proceeds successively from leaves to roots), for which the same error bound holds [Tobler’10].
Truncation of $\mathcal{H}$-Tucker tensor

Let $X \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$ be in $\mathcal{H}$-Tucker format and orthogonalized.

- Compute left singular vectors of $X^{(t)} = U_t V_t^T$ from eigenvectors of
  
  $$X^{(t)} (X^{(t)})^T = U_t V_t^T V_t U_t^T, \quad = G_t$$

  with reduced Gramian $G_t$ (calculated recursively for all $t$).

  If $S_t$ contains $r_t$ dominant eigenvectors of $G_t \leadsto W_t = U_t S_t$.

- Remember: $\tilde{B}_t = [W_{t_2}^H \otimes W_{t_1}^H] W_t$

  $$\tilde{B}_t = [S_{t_2}^H U_{t_2}^H \otimes S_{t_1}^H U_{t_1}^H] U_t S_t$$

  $$= [S_{t_2}^H U_{t_2}^H \otimes S_{t_1}^H U_{t_1}^H](U_{t_2} \otimes U_{t_1}) B_t S_t = [S_{t_2}^H \otimes S_{t_1}^H] B_t S_t.$$  

- In $\text{htucker: } \text{truncate}(x, \text{opts})$. Complexity $\mathcal{O}(dnr^2 + dr^4)$. 

Conclusions

htucker:

▶ Convenient MATLAB implementation of hierarchical Tucker format.

▶ Plenty of utilities and examples hopefully get you started.

  i.e., there is no documentation yet.

▶ Current version 0.8. (We are open to suggestions!)

Coming soon:

▶ Application to high-dimensional parabolic PDEs (R. Andreev)
Conclusions

htucker:

▶ Convenient MATLAB implementation of hierarchical Tucker format.
▶ Plenty of utilities and examples hopefully get you started. i.e., there is no documentation yet.
▶ Current version 0.8. (We are open to suggestions!)

Coming soon:

▶ Application to high-dimensional parabolic PDEs (R. Andreev)

Thank you for your attention!