On the Way from Matrix to Tensor Computations

Introduction, Basic arithmetics, Tensor decompositions, Hierarchical formats, and Tensor networks

M. Plešinger with a lot of inspiration from collaboration with

I. Hnětynková, D. Kressner, C. Tobler, J. Žáková,

with special thanks to B. N. Khoromskij, and many other colleagues ...

martin.plesinger@tul.cz

Department of Mathematics, Technical University of Liberec, Liberec

SNA '19, Ostrava, January 21—25, 2019

Outline of the tutorial

- **Lecture I**

  Introduction to tensors

  Basic terminology and basic manipulation with tensors

  Rank of a tensor

  Tensor arithmetics
- **Lecture II**

  Basic decompositions of a tensor

  Low-rank arithmetics of tensors

  Graph interpretation: Tensor networks & Hierarchical formats

  Arithmetics of hierarchical Tucker

  An example of practical application

[*T. G. Kolda, B. W. Bader*: **Tensor decompositions and applications**, SIAM Review 51(3), pp. 455–500, 2009]

Introduction to tensors

Introduction

The standard tensor definition

A first (and only) definition of a tensor I met at school:

Tensor $\mathcal{T}$ of order k is a k_1-covariant and k_2-contravariant ($k = k_1 + k_2$) multilinear form on linear vector space $\mathscr{V}$ over $\mathbb{R}$,

$$\mathcal{T} : \underbrace{\mathscr{V} \times \mathscr{V} \times \cdots \times \mathscr{V}}_{k_1\text{-times}} \times \underbrace{\mathscr{V}^* \times \mathscr{V}^* \times \cdots \times \mathscr{V}^*}_{k_2\text{-times}} \longrightarrow \mathbb{R}.$$

In this way tensors are used in many branches of mathematics and physics (differential geometry, solid-state physics, continuum mechanics, general relativity, etc.).

It is something like a matrix, but ...

What is a matrix?

Three (distinct) reference frames

A matrix A can be seen as a mapping between linear vector spaces

$$\begin{aligned} A : \mathbb{R}^n &\longrightarrow \mathbb{R}^m \\ u &\longmapsto w = Au, \end{aligned}$$

as a bilinear form

$$\begin{aligned} A : \mathbb{R}^n \times \mathbb{R}^m &\longrightarrow \mathbb{R} \\ (u, v) &\longmapsto f(u, v) = v^{\mathsf{T}} A u, \end{aligned}$$

and also as an algebraic vector, a member of linear vectors space

$$A \in \mathbb{R}^{m \times n}.$$

What is a matrix?

Transformations of matrices

Let $m = n$ (A is square). We change the basis in $\mathbb{R}^n$ as follows $x = Zx'$, i.e., $x \longmapsto x' = Z^{-1}x$, then

$$\begin{aligned} Au &= w \\ A(Zu') &= Zw' \\ \underbrace{(Z^{-1}AZ)}\, u' &= w' \end{aligned}, \qquad \begin{aligned} f(u,v) &= v^{\mathsf{T}}Au \\ f(Zu',Zv') &= (Zv')^{\mathsf{T}}A(Zu') \\ f'(u',v') &= v'^{\mathsf{T}}\underbrace{(Z^{\mathsf{T}}AZ)}\, u' \end{aligned}.$$

We get two different transf's of A, $A \longmapsto Z^{-1}AZ$ (similarity transf.; eigenvalues) and $A \longmapsto Z^{\mathsf{T}}AZ$ (congruence; quadratic forms), resp.

On the other hand, we can study the matrix itself—e.g., decompositions:

$$A = LU, \quad A = LL^{\mathsf{T}}, \quad A = QR, \quad A = XDX^{-1}, \quad A = U\Sigma V^{\mathsf{T}}, \quad \text{etc.}$$

Definition of a tensor

… and its 'justification'

Similarly to matrices, we can observe a tensor from different perspectives: As a (multi)linear mapping(s) between different vector spaces, or form on $\mathscr{V}$ (and its dual $\mathscr{V}^*$).

In many applications, however, we are focused more on the '*interior structure*' of the tensor (e.g., we are looking for some decomposition), than on its interactions with its 'surroundigs'.

Definition. Tensor $\mathcal{T}$ of order k is a k-way *array* of real numbers of the given dimension,

$$\mathcal{T} = (t_{i_1,i_2,\dots,i_k}) \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}.$$

Note that $n_i \neq n_j$ for $i \neq j$, in general, thus we do not need to distinguish the co- and contravariant indices.

Why tensors?

- Tensors in this form was introduced in psychometrics and chemometrics while analysis of large multidim. arrays of data
- The goal is to find some structure in the data (big data) that allows to analyze (interpret, understand) the data, and simplifies it in such a way, we can easier manipulate it; c.f. the singular value decomposition (SVD) in the case of matrix.
- The memory consumption while storing the tensor as it is, scales exponentialy with k, so-called "curse of dimensionality",

$$\sim n^k \quad \text{where} \quad n = \max\{n_1, n_2, \ldots, n_k\}.$$

- We want to employ basic linear algebra tools (matrix decompositions, etc.).
- In the optimal case, we would like to find a structure (decomposition) that scales linearly with the tensor order k.

Basic terminology and basic manipulation with tensors

Order and shape of tensor

Tensors of small orders

By the order of tensor $\mathcal{T} = (t_{i_1,i_2,\ldots,i_k}) \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$ we understood the number of its indices, i.e., the number k. Tensors of small orders have special names, for

- $k = 0$ we call them scalars (and denote by α, β, etc.);
- $k = 1$ we call them vectors (and denote by x, y, etc.);
- $k = 2$ we call them matrices (and denote by A, B, etc.);
- $k \geq 3$ we call them just tensors (and denote by $\mathcal{T}$, $\mathcal{S}$, etc.).

By the dimension, we understood the k-tuple $(n_1, n_2, \ldots, n_k)$. If

- $k = 2$ and $n_1 = n_2$, we call them square matrices;
- $k \geq 3$ and $n_1 = n_2 = \cdots = n_k$, we call them cubic tensors.

Moreover, we denote $N = \prod_{\kappa=1}^{k} n_\kappa = n_1 \cdot n_2 \cdot \cdots \cdot n_k$.

Tensors and subtensors

General subtensors

Our tensor $\mathcal{T}$ is an *ordered* set of numbers $t_{i_1,i_2,\ldots,i_k} \in \mathbb{R}$ with indices

$$i_\kappa \in \{1, 2, \ldots, n_\kappa\} \equiv \mathscr{I}_\kappa, \quad \text{for} \quad \kappa = 1, 2, \ldots, k,$$

or, equivalently, with multiindices

$$(i_1, i_2, \ldots, i_k) \in \mathscr{I}_1 \times \mathscr{I}_2 \times \cdots \times \mathscr{I}_k.$$

Let $\mathscr{I}'_\kappa \subseteq \mathscr{I}_\kappa$. The subarray of $\mathcal{T}$ obtained by employing only the multiindices in the subset $\mathscr{I}'_1 \times \mathscr{I}'_2 \times \cdots \times \mathscr{I}'_k$ is called a subtensor.

There are several kinds of subtensors of particular importance, e.g., so-called fibres, slices, and co-fibres.

Subtensors: Fibres

Rows, columns, tubes, and the others...

Let $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$, let for some fixed ℓ

$$\mathscr{I}'_\ell = \mathscr{I}_\ell = \{1, 2, \ldots, n_\ell\}, \quad \text{and} \quad \mathscr{I}'_\kappa = \{i_\kappa\} \quad \text{for all} \quad \kappa \neq \ell.$$

The associated subtensor is called the ℓ-mode fibre specified by the $(k–1)$-tuple of indices $(i_1, \ldots, i_{\ell-1}, i_{\ell+1}, \ldots, i_k)$. We denote it

$$\mathcal{T}_{i_1, \ldots, i_{\ell-1}, \star, i_{\ell+1}, \ldots, i_k} \in \mathbb{R}^{1 \times \cdots \times 1 \times n_\ell \times 1 \times \cdots \times 1},$$

it is isomorphic to an n_ℓ-vector. There is N/n_ℓ of ℓ-mode fibres.

The ℓ-mode fibres, $\ell = 1, 2, \ldots, k$ are for

- $k = 2$ called the columns and rows, respectively;
- $k = 3$ called the columns, rows, and tubes, respectively.

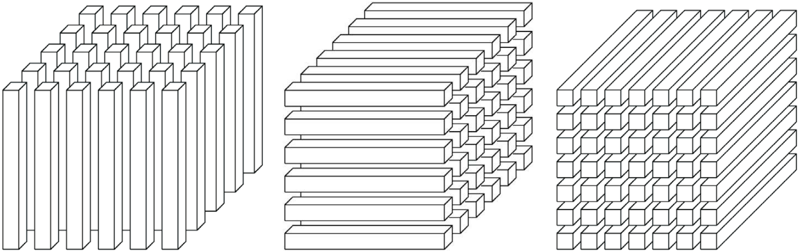
Subtensors: Fibres

Rows, columns, tubes, and the others...

For $k = 3$, the ℓ-mode fibres, $\ell = 1, 2, 3$, i.e.,

$$\mathcal{T}_{\star, i_2, i_3} \in \mathbb{R}^{n_1 \times 1 \times 1}, \quad \mathcal{T}_{i_1, \star, i_3} \in \mathbb{R}^{1 \times n_2 \times 1}, \quad \mathcal{T}_{i_1, i_2, \star} \in \mathbb{R}^{1 \times 1 \times n_3}$$

are called the columns, rows, and tubes, respectively.



Subtensors: Slices

Horizontal, lateral, frontal, and the others...

Let $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$, let for some fixed τ and ß ($\tau \neq$ ß)

$$\mathscr{I}'_\tau = \mathscr{I}_\tau, \quad \mathscr{I}'_ß = \mathscr{I}_ß \quad \text{and} \quad \mathscr{I}'_\kappa = \{i_\kappa\} \quad \text{for all} \quad \kappa \neq \tau \text{ and } \kappa \neq ß.$$

If $\tau <$ ß, the subtensor is called the (τ, ß)-mode slice given by the (k–2)-tuple $(i_1, \ldots, i_{\tau-1}, i_{\tau+1}, \ldots, i_{ß-1}, i_{ß+1}, \ldots, i_k)$. We denote it

$$\mathcal{T}_{i_1,\ldots,i_{\tau-1},\star,i_{\tau+1},\ldots,i_{ß-1},\star,i_{ß+1},\ldots,i_k} \in \mathbb{R}^{1\times\cdots\times 1\times n_\tau \times 1\times\cdots\times 1\times n_ß \times 1\times\cdots\times 1},$$

it is isomorphic to an n_τ-by-$n_ß$ matrix. There is $N/(n_\tau \cdot n_ß)$ of them.

Sometimes, the fibers and slices are considered to be the vectors and matrices. Then we can introduce both, the (τ, ß)- and (ß, τ)-mode slices. Since they are matrices, they are mutually transposed.

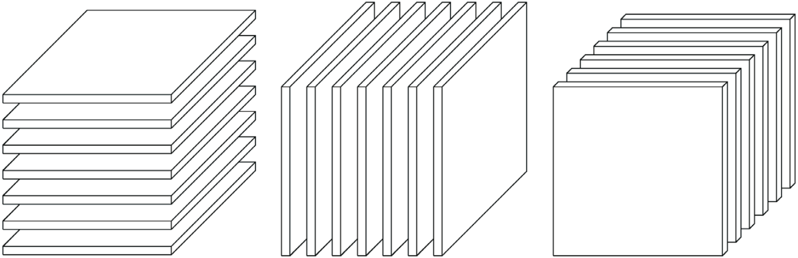
Subtensors: Slices

Horizontal, lateral, frontal, and the others...

For $k = 3$, the $(\tau, \text{ß})$-mode slices, $(\tau, \text{ß}) = (2,3), (1,3), (1,2)$, i.e.,

$$\mathcal{T}_{i_1,\star,\star} \in \mathbb{R}^{1\times n_2\times n_3}, \quad \mathcal{T}_{\star,i_2,\star} \in \mathbb{R}^{n_1\times 1\times n_3}, \quad \mathcal{T}_{\star,\star,i_3} \in \mathbb{R}^{n_1\times n_2\times 1},$$

are called the horizontal, lateral, and frontal, respectively.



Subtensors: Co-fibres

We see that it is easier to identify the type (i.e., horizontal, lateral, frontal) slices of 3-way by the ‘missing index’ than by the pair $(\tau, ß)$ of ‘generating indices’.

Thus we also introduce the ℓ-mode co-fibres such that,

$$\mathscr{I}'_\ell = \{i_\ell\} \quad \text{and} \quad \mathscr{I}'_\kappa = \mathscr{I}_\kappa \quad \text{for all} \quad \kappa \neq \ell,$$

specified by the single index (i_ℓ), denoted

$$\mathcal{T}_{\star,\dots,\star,i_\ell,\star,\dots,\star} \in \mathbb{R}^{n_1\times\cdots\times n_{\ell-1}\times 1\times n_{\ell+1}\times\cdots\times n_k}.$$

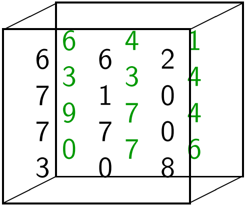
For $k = 3$, the ℓ-mode co-fibres = the $(\tau, ß)$-mode slices ($\ell \neq \tau$, $\ell \neq ß$, $\tau < ß$).

We can continue in a similar manner, but...

Matricization

Unfolding a tensor into a matrix

Collection of all ℓ-mode fibres (handled as vectors) of the given tensor $\mathcal{T}$ into a single matrix $\mathcal{T}^{\{\ell\}} \in \mathbb{R}^{n_\ell \times (N/n_\ell)}$ in the *inverse* lexicographical order is called the ℓ-mode matricization. For

$\mathcal{T} =$  $\in \mathbb{R}^{4\times3\times2}$, we get

$$\mathcal{T}^{\{1\}} = [\mathcal{T}_{\star,1,1}, \mathcal{T}_{\star,2,1}, \mathcal{T}_{\star,3,1}, \mathcal{T}_{\star,1,2}, \mathcal{T}_{\star,2,2}, \mathcal{T}_{\star,3,2}] = \left[\begin{array}{ccc|ccc} 6 & 6 & 2 & 6 & 4 & 1 \\ 7 & 1 & 0 & 3 & 3 & 4 \\ 7 & 7 & 0 & 9 & 7 & 4 \\ 3 & 0 & 8 & 0 & 7 & 6 \end{array}\right],$$

$$\mathcal{T}^{\{2\}} = \left[\begin{array}{cccc|cccc} 6 & 7 & 7 & 3 & 6 & 3 & 9 & 0 \\ 6 & 1 & 7 & 0 & 4 & 3 & 7 & 7 \\ 2 & 0 & 0 & 8 & 1 & 4 & 4 & 6 \end{array}\right], \quad \mathcal{T}^{\{3\}} = \left[\begin{array}{cccc|cccc|cccc} 6 & 7 & 7 & 3 & 6 & 1 & 7 & 0 & 2 & 0 & 0 & 8 \\ 6 & 3 & 9 & 0 & 4 & 3 & 7 & 7 & 1 & 4 & 4 & 6 \end{array}\right].$$

Generalized matricization

Unfolding a tensor into a matrix

Let $\mathcal{T}$ be a k-way tensor and

$$\mathscr{R} = \{r_1, r_2, \ldots, r_\mu\}, \quad r_1 < r_2 < \cdots < r_\mu,$$
$$\mathscr{C} = \{c_1, c_2, \ldots, c_\nu\}, \quad c_1 < c_2 < \cdots < c_\nu,$$

such that $\mathscr{R} \cup \mathscr{C} = \{1, 2, \ldots, k\}$ and $\mathscr{R} \cap \mathscr{C} = \varnothing$. Then

$$\mathcal{T}^{\mathscr{R}} = \mathcal{T}^{\{r_1, r_2, \ldots, r_\mu\}} \in \mathbb{R}^{n_R \times n_C}, \qquad n_R = \prod_{i=1}^{\mu} r_i, \quad n_C = \prod_{j=1}^{\nu} c_j.$$

The entry $t_{i_1, i_2, \ldots, i_k}$ of $\mathcal{T}$ is in the matrix $\mathcal{T}^{\mathscr{R}}$ in the row and column specified by multiindices

$$(r_1, r_2, \ldots, r_\mu) \quad \text{and} \quad (c_1, c_2, \ldots, c_\nu), \quad \text{respectively.}$$

Rows and columns are in $\mathcal{T}^{\mathscr{R}}$ sorted in the *inverse* lexicographical order w.r.t. their multiindices.

Generalized matricization

Examples

Clearly, in general

$$(\mathcal{T}^{\mathscr{R}})^{\mathsf{T}} = \mathcal{T}^{\mathscr{C}}.$$

For our $4 \times 3 \times 2$ tensor,

$$\mathcal{T}^{\{1\}} = \left[\begin{array}{ccc|ccc} 6 & 6 & 2 & 6 & 4 & 1 \\ 7 & 1 & 0 & 3 & 3 & 4 \\ 7 & 7 & 0 & 9 & 7 & 4 \\ 3 & 0 & 8 & 0 & 7 & 6 \end{array}\right] = (\mathcal{T}^{\{2,3\}})^{\mathsf{T}},$$

$$\mathcal{T}^{\{2\}} = \left[\begin{array}{cccc|cccc} 6 & 7 & 7 & 3 & 6 & 3 & 9 & 0 \\ 6 & 1 & 7 & 0 & 4 & 3 & 7 & 7 \\ 2 & 0 & 0 & 8 & 1 & 4 & 4 & 6 \end{array}\right] = (\mathcal{T}^{\{1,3\}})^{\mathsf{T}},$$

$$\mathcal{T}^{\{3\}} = \left[\begin{array}{cccc|cccc|cccc} 6 & 7 & 7 & 3 & 6 & 1 & 7 & 0 & 2 & 0 & 0 & 8 \\ 6 & 3 & 9 & 0 & 4 & 3 & 7 & 7 & 1 & 4 & 4 & 6 \end{array}\right] = (\mathcal{T}^{\{1,2\}})^{\mathsf{T}}.$$

But there are two more matricizations...

Generalized matricization

Examples

The last two case for 3-way tensor are for $\mathscr{R} = \{1, 2, 3\}$ and $\varnothing$,

$$\mathcal{T}^{\{1,2,3\}} = \left[\begin{array}{c} t_{1,1,1} \\ t_{2,1,1} \\ t_{3,1,1} \\ t_{4,1,1} \\ \hline t_{1,2,1} \\ t_{2,2,1} \\ t_{3,2,1} \\ t_{4,2,1} \\ \hline t_{1,3,1} \\ t_{2,3,1} \\ t_{3,3,1} \\ t_{4,3,1} \\ \hline\hline t_{1,1,2} \\ t_{2,1,2} \\ t_{3,1,2} \\ t_{4,1,2} \\ \hline t_{1,2,2} \\ t_{2,2,2} \\ t_{3,2,2} \\ t_{4,2,2} \\ \hline t_{1,3,2} \\ t_{2,3,2} \\ t_{3,3,2} \\ t_{4,3,2} \end{array}\right] = \left[\begin{array}{c} 6 \\ 7 \\ 7 \\ 3 \\ \hline 6 \\ 1 \\ 7 \\ 0 \\ \hline 2 \\ 0 \\ 0 \\ 8 \\ \hline\hline 6 \\ 3 \\ 9 \\ 0 \\ \hline 4 \\ 3 \\ 7 \\ 7 \\ \hline 1 \\ 4 \\ 4 \\ 6 \end{array}\right] = (\mathcal{T}^{\varnothing})^{\mathsf{T}} \equiv \mathrm{vec}(\mathcal{T}).$$

We call this ↑ the vectorization of a tensor (or matrix).

Generalized matricization

Matricization–vectorization relation

Recall that the ℓ-mode matricization is a matrix that contain the ℓ-mode fibres as columns (particularly sorted).

The rows of ℓ-mode matricization are then vectorizations of ℓ-mode co-fibres.

In our case, columns of $\mathcal{T}^{\{1\}}$ are the 1-mode fibres (columns) of $\mathcal{T}$,

$$\mathcal{T}^{\{1\}} = [\mathcal{T}_{\star,1,1}, \mathcal{T}_{\star,2,1}, \mathcal{T}_{\star,3,1}, \mathcal{T}_{\star,1,2}, \mathcal{T}_{\star,2,2}, \mathcal{T}_{\star,3,2}].$$

and rows of $\mathcal{T}^{\{1\}}$ (i.e., transposed columns of $\mathcal{T}^{\{2,3\}}$) are the transposed vectorizations of the 1-mode co-fibrer (i.e., actually the $(2,3)$-slices (the horizontal slices)) of $\mathcal{T}$.

Note on transposition

The matrix transposition

$$A \in \mathbb{R}^{m \times n} \quad \longmapsto \quad A^{\mathsf{T}} \in \mathbb{R}^{n \times m}$$

exchanges the roles of columns (1-mode) and rows (2-mode fib's).

Tensors can be manipulated in a similar fashion, in general, by an arbitrary permutation of roles of individual fibres. Let

$$\Pi = \begin{pmatrix} 1 & 2 & \cdots & k \\ \pi(1) & \pi(2) & \cdots & \pi(k) \end{pmatrix},$$

then

$$\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k} \quad \longmapsto \quad \mathcal{T}^{\Pi} \in \mathbb{R}^{n_{\pi(1)} \times n_{\pi(2)} \times \cdots \times n_{\pi(k)}},$$

$$(\mathcal{T}^{\Pi})_{i_1, i_2, \ldots, i_k} = t_{i_{\pi(1)}, i_{\pi(2)}, \ldots, i_{\pi(k)}}.$$

Norm and scalar product of tensors

We use the simplest available norm

$$\|\mathcal{T}\| = \left(\sum_{j_1=1}^{n_1} \sum_{j_2=1}^{n_2} \cdots \sum_{j_k=1}^{n_k} |t_{j_1,j_2,\dots,j_k}|^2 \right)^{\frac{1}{2}} = \left(\mathrm{vec}(\mathcal{T})^\mathsf{T} \mathrm{vec}(\mathcal{T})\right)^{\frac{1}{2}}$$

which directly generalizes the standard

- Euclidean norm of vectors and
- Frobenius norm of matrices.

Moreover, it is induced by the inner product

$$\langle \mathcal{T}, \mathcal{S} \rangle = \sum_{j_1=1}^{n_1} \sum_{j_2=1}^{n_2} \cdots \sum_{j_k=1}^{n_k} s_{j_1,j_2,\dots,j_k} \cdot t_{j_1,j_2,\dots,j_k} = \mathrm{vec}(\mathcal{S})^\mathsf{T} \mathrm{vec}(\mathcal{T})$$

which directly generalizes the standard

- Euclidean scalar product of vectors $\langle x, y \rangle = y^\mathsf{T} x$ and
- commonly used scalar prod. of matrices $\langle A, B \rangle = \mathrm{trace}(B^\mathsf{T} A)$.

Rank of a tensor

Rank of a matrix

Let start gently...

What is the rank of a matrix $A \in \mathbb{R}^{m\times n}$?

- The order of the largerst nonzero minor of A ;-).
- The maximal number of linearly independent columns of A.
- The maximal number of linearly independent rows of A.
- The minimal number of pairs $(x_j, y_j) \in \mathbb{R}^m \times \mathbb{R}^n$, such that

$$A = x_1 y_1^{\mathsf{T}} + x_2 y_2^{\mathsf{T}} + \cdots \quad = \sum_{\varrho} x_\varrho y_\varrho^{\mathsf{T}},$$

  i.e., the length of the shortest dyadic expansion of A.

Note that the SVD of A serves the shortest dyadic expansion with mutually orthogon(norm)al x_ϱ's and y_ϱ's.

Number of linearly independent fibres...

The ℓ-rank

Since columns and rows are the 1-mode and 2-mode fibres of a matrix, there is a straightforward generalization:

The ℓ-mode rank of the tensor $\mathcal{T}$ is the maximal number of linearly independent ℓ-mode fibres, i.e.,

$$\mathrm{rank}_{\{\ell\}}(\mathcal{T}) \equiv \mathrm{rank}(\mathcal{T}^{\{\ell\}}), \quad \mathcal{T}^{\{\ell\}} \in \mathbb{R}^{n_\ell \times (N/n_\ell)}, \quad N = \prod_{\kappa=1}^{k} n_\kappa.$$

Since $\mathcal{T}^{\{\ell\}}$ is a matrix, whose rows are transposed vectorizations of ℓ-mode co-fibres, we get:

the maximal number of linearly independent ℓ-mode fibres
= the maximal number of linearly independent ℓ-mode co-fibres.

Recall that for $k = 2$ (in the matrix case), the 1-mode co-fibres are the 2-mode fibres (rows) and vice versa.

Number of linearly independent fibres...

The vector rank of tensor

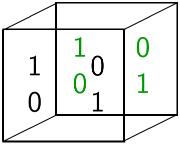
Consequently, for $\ell \neq$ ß, there is no direct relation between

$$\mathrm{rank}_{\{\ell\}}(\mathcal{T}) \quad \text{and} \quad \mathrm{rank}_{\{\text{ß}\}}(\mathcal{T}).$$

The different-mode ranks may be different. Therefore we introduce the vector rank of the tensor,

$$\overrightarrow{\mathrm{rank}}(\mathcal{T}) \equiv (\mathrm{rank}_{\{1\}}(\mathcal{T}), \mathrm{rank}_{\{1\}}(\mathcal{T}), \ldots, \mathrm{rank}_{\{k\}}(\mathcal{T})).$$

For example

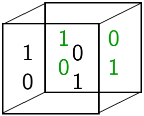
$\mathcal{T} =$ 

$\in \mathbb{R}^{2\times2\times2}$ is of $\overrightarrow{\mathrm{rank}}(\mathcal{T}) = (2, 2, 1)$.

Number of linearly independent fibres...

The vector rank of tensor

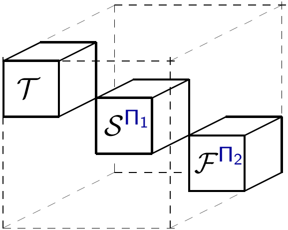
Consider now three of such vectors but of diferent dimensions,

$\mathcal{T} =$ 

$\in \mathbb{R}^{2\times2\times2}$ and similarly $\mathcal{S} \in \mathbb{R}^{3\times3\times3}$, $\mathcal{F} \in \mathbb{R}^{4\times4\times4}$,

i.e., $\overrightarrow{\mathrm{rank}}(\mathcal{T}) = (2,2,1)$, $\overrightarrow{\mathrm{rank}}(\mathcal{S}) = (3,3,1)$, $\overrightarrow{\mathrm{rank}}(\mathcal{F}) = (4,4,1)$.
Their permutations and direct sum (i.e., block-diagonal assembly),

$$\mathrm{diag}_3\big(\mathcal{T}, \mathcal{S}^{\left(\begin{smallmatrix}1&2&3\\3&1&2\end{smallmatrix}\right)}, \mathcal{F}^{\left(\begin{smallmatrix}1&2&3\\2&3&1\end{smallmatrix}\right)}\big) \equiv \mathcal{T} \oplus \mathcal{S}^{\left(\begin{smallmatrix}1&2&3\\3&1&2\end{smallmatrix}\right)} \oplus \mathcal{F}^{\left(\begin{smallmatrix}1&2&3\\2&3&1\end{smallmatrix}\right)} =$$



,

is of vector rank $(2,2,1)+(3,1,3)+(1,4,4)=(6,7,8)$.

Shortest polyadic expansion

Polyadic rank of a tensor

Any matrix A, $r \equiv \mathrm{rank}(A)$ can be written in the dyadic expansion,

$$A = x_1 y_1^{\mathsf{T}} + x_2 y_2^{\mathsf{T}} + \cdots \; = \sum_{\varrho=1}^{r} x_\varrho y_\varrho^{\mathsf{T}}, \qquad \text{where}$$

$$A_\varrho \equiv x_\varrho {y_\varrho}^{\mathsf{T}} =$$

$$, \qquad (A_\varrho)_{i,j} = (x_\rho)_i \cdot (y_\rho)_j$$

is the rank-one matrix—the outer product of two vectors

This motivates the polyadic expansion of k-way tensor as the sum of rank-one terms—the outer products of k vectors; e.g., for $k = 3$

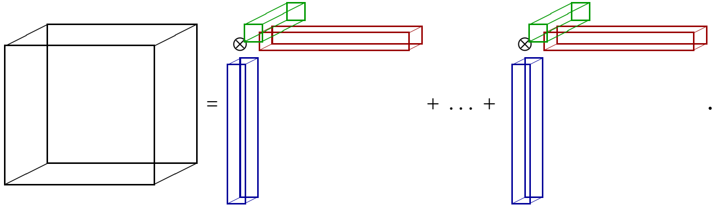
$$\mathcal{T}_\varrho \equiv (x_\varrho, y_\varrho, z_\varrho)_\otimes, \quad \text{where} \quad x_\varrho \in \mathbb{R}^{n_1}, \quad y_\varrho \in \mathbb{R}^{n_2}, \quad z_\varrho \in \mathbb{R}^{n_3},$$

$$(\mathcal{T}_\varrho)_{i_1,i_2,i_3} = (x_\rho)_{i_1} \cdot (y_\rho)_{i_2} \cdot (z_\rho)_{i_3}\,.$$

Shortest polyadic expansion

Polyadic rank of a tensor

Then the polyadic expansion takes form $\mathcal{T} = \sum_\varrho (x_\varrho, y_\varrho, z_\varrho)_\otimes$,



It represents our first kind of tensor decomposition into three matrices $X = [x_1, x_2, \ldots] \in \mathbb{R}^{n_1 \times ?}$, $Y = [y_1, y_2, \ldots] \in \mathbb{R}^{n_2 \times ?}$, $Z = [z_1, z_2, \ldots] \in \mathbb{R}^{n_3 \times ?}$.

This decomposition is intensively studied and it is known under names CanDeComp (Canonic DeComposition), ParaFac (Paralel Factorization), or CP decomposition (CanDeComp–ParaFac).

Shortest polyadic expansion

Polyadic rank of a tensor

In the case of matrices:

- The polyadic expansion can be done in such a way that both $X \in \mathbb{R}^{n\times r}$ and $Y \in \mathbb{R}^{m\times r}$ have orthogon(norm)al columns (via the SVD).
- Rank of A is the minimal number of terms (length of the shortest dyadic exp.).
- The Eckart–Young–Mirsky theorem shows that the difference between A and its best approximation obtained by using only q dyadic terms, $q < r = \mathrm{rank}(A)$, is in the norm equal to $\sigma_{q+1}(A)$, i.e., this approximation problem has (well defined) minimum.

What about tensors?

Shortest polyadic expansion

Polyadic rank of a tensor

We can play with the orthogonality by employing QR decomp's of X, Y, Z, etc. It will be *briefly* mentioned later.

The number of rank-one terms is bounded by N, thus there is the minimal number, defining the polyadic rank,

$$\max_{\ell=1,2,\dots,k} \operatorname{rank}_{\{\ell\}}(\mathcal{T}) \leq \operatorname{polyrank}(\mathcal{T}) \leq \operatorname{nnz}(\mathcal{T}) \leq N = n_1 \cdot n_2 \cdot \cdots \cdot n_k.$$

This rank, however, is **not** robust. Let

$$X = [x', x', x''] \in \mathbb{R}^{n_1 \times 3}, \; Y = [y', y'', y'] \in \mathbb{R}^{n_2 \times 3}, \; Z = [z'', z', z'] \in \mathbb{R}^{n_3 \times 3},$$

and $\operatorname{rank}(X) = \operatorname{rank}(Y) = \operatorname{rank}(Z) = 2$. Consider

$$\mathcal{T} = (x', y', z'')_\otimes + (x', y'', z')_\otimes + (x'', y', z')_\otimes ,$$

$$\mathcal{T}_\varepsilon = \frac{1}{\varepsilon}(x' + \varepsilon x'', y' + \varepsilon y'', z' + \varepsilon z'')_\otimes - \frac{1}{\varepsilon}(x', y', z')_\otimes , \quad \text{then}$$

$$\|\mathcal{T} - \mathcal{T}_\varepsilon\| = \varepsilon \|(x'', y'', z')_\otimes + (x'', y', z'')_\otimes + (x', y'', z'')_\otimes + \varepsilon(x'', y'', z'')_\otimes\|.$$

[*P. Paatero*, J. of Chemometrics 14(3), pp. 285–299, 2000].

Sum of rank-one terms

Another generalization of dyadic expansion

Note that rank-one (rank-at-most-one) terms

$$(x_\varrho, y_\varrho)_\otimes = xy^\mathsf{T}, \quad (x_\varrho, y_\varrho, z_\varrho)_\otimes, \quad x_\varrho \in \mathbb{R}^{n_1},\ y_\varrho \in \mathbb{R}^{n_2},\ z_\varrho \in \mathbb{R}^{n_3},$$

form submanifolds within $\mathbb{R}^{n_1 \times n_2}$ and $\mathbb{R}^{n_1 \times n_2 \times n_3}$, respectively.

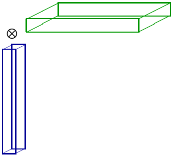
We can take another suitable submanifold and its members consider to be the rank-one terms. For example,

$$\mathcal{T}_\varrho = (x_\varrho, M_\varrho)_\otimes, \quad \text{where} \quad x_\varrho \in \mathbb{R}^{n_1},\ M_\varrho \in \mathbb{R}^{n_2 \times n_3},$$

and $(\mathcal{T}_\varrho)_{i_1,i_2,i_3} = (x_\varrho)_{i_1} \cdot (M_\varrho)_{i_2,i_3}$.

Then rank of $\mathcal{T}$ can be defined as the length of shortest sum

$\mathcal{T} = \sum_\varrho \mathcal{T}_\varrho = \sum_\varrho$



; this rank $= \operatorname{rank}_{\{1\}}(\mathcal{T}) = \operatorname{rank}(\mathcal{T}^{\{1\}})$.

Another example

4-way tensor & the Kronecker product

Let $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times n_3 \times n_4}$ and $\mathcal{T} = \sum_\varrho \mathcal{T}_\varrho$, where

$$\mathcal{T}_\varrho \equiv (K_\varrho, M_\varrho)_\otimes \quad \text{such that} \quad (\mathcal{T}_\varrho)_{i_1,i_2,i_3,i_4} = (K_\varrho)_{i_1,i_2} \cdot (M_\varrho)_{i_3,i_4},$$

and $K_\varrho \in \mathbb{R}^{n_1 \times n_2}$, $M_\varrho \in \mathbb{R}^{n_3 \times n_4}$.

The length of the shortest sum can be observed after rearraging to

$$\mathcal{T}^{\{1,2\}} = \sum_\varrho T_\varrho^{\{1,2\}} = \sum_\varrho \operatorname{vec}(K_\varrho)\big(\operatorname{vec}(M_\varrho)\big)^\mathsf{T} \in \mathbb{R}^{(n_1 \cdot n_2) \times (n_3 \cdot n_4)};$$

it is the rank of this matrix, in general $\operatorname{rank}_{\mathscr{R}}(\mathcal{T}) \equiv \operatorname{rank}(T^{\mathscr{R}})$.

Note another rearranging gives

$$\mathcal{T}^{\{1,3\}} \in \mathbb{R}^{(n_1 \cdot n_3) \times (n_2 \cdot n_4)}, \quad \mathcal{T}^{\{1,3\}} = \sum_{\varrho=1}^{\operatorname{rank}_{\{1,2\}}(\mathcal{T})} M_\varrho \otimes K_\varrho \,,$$

where $\otimes$ is the Kronecker product of matrices.

Note on Kronecker product

For matrices, the standard matrix and Kronecker products we have

$$(AB) \otimes (CD) = (A \otimes C)(B \otimes D).$$

Thus, if any two of the following three matrices

$$A, \quad C, \quad E = A \otimes C$$

are invertible, then the third is also invertible.

We can intepret E as the $\{1,3\}$-matricization of a 4-way tensor $\mathcal{E}$, i.e., $E = \mathcal{E}^{\{1,3\}} = A \otimes C$. Then its $\{1,2\}$-matricization takes form

$$\mathcal{E}^{\{1,2\}} = \mathrm{vec}(A)\big(\mathrm{vec}(C)\big)^{\mathsf{T}}$$

All three $\mathcal{E}$, $\mathcal{E}^{\{1,3\}}$, $\mathcal{E}^{\{1,2\}}$ represent the same rank-one object (just differently rearranged) in the given submanifold of 4-way tensors.

But $\mathcal{E}^{\{1,3\}}$ may be invertible whereas $\mathrm{rank}(\mathcal{E}^{\{1,2\}}) = 1$ always.

Final note on ranks

For a given tensor $\mathcal{T}$, we have

- $\mathrm{rank}_{\{\ell\}}(\mathcal{T}) \equiv \mathrm{rank}(\mathcal{T}^{\{\ell\}})$ for $\ell = 1, 2, \ldots, k$,
- $\overrightarrow{\mathrm{rank}}(\mathcal{T}) \equiv \left(\mathrm{rank}_{\{1\}}(\mathcal{T}), \mathrm{rank}_{\{2\}}(\mathcal{T}), \ldots, \mathrm{rank}_{\{k\}}(\mathcal{T})\right)$,
- $\mathrm{rank}_{\mathscr{R}}(\mathcal{T}) \equiv \mathrm{rank}(T^{\mathscr{R}})$ for $\mathscr{R} \subseteq \{1, 2, \ldots, k\}$,
- clearly

$$\left\{\mathrm{rank}_{\{\ell\}}(\mathcal{T})\ ,\ \ell = 1, 2, \ldots, k\right\} \subseteq \left\{\mathrm{rank}_{\mathscr{R}}(\mathcal{T})\ ,\ \mathscr{R} \subseteq \{1, 2, \ldots, k\}\right\},$$

- $\mathrm{polyrank}(\mathcal{T})$:

$$\max_{\mathscr{R} \subseteq \{1,2,\ldots,k\}} \mathrm{rank}_{\mathscr{R}}(\mathcal{T}) \leq^{(*)} \mathrm{polyrank}(\mathcal{T})\,;$$

(∗)

$$\left((x', y', z'')_{\otimes} + (x', y'', z')_{\otimes} + (x'', y', z')_{\otimes}\right)^{\{1,2\}}$$
$$= \left[(y' \otimes x'), (y'' \otimes x') + (y' \otimes x'')\right]\left[z'', z'\right]^{\mathsf{T}}.$$

Tensor arithmetics

Basic operations

Linear combinations, direct sum, outer product

We already know some basic operations.

- Since tensors of the given fixed dimensions form a linear vector space, we can do componentwisely

$$\alpha \mathcal{T}, \quad \mathcal{T} + \mathcal{S}, \quad \alpha \mathcal{T} + \beta \mathcal{S}, \quad \sum_\ell \alpha_\ell \mathcal{T}_\ell .$$

- We can do the direct sum of tensors of the same$^{(?!)}$ order k

$$\mathcal{T} \oplus \mathcal{S} = \operatorname{diag}_k(\mathcal{T}, \mathcal{S}) \in \mathbb{R}^{(n_1+m_1)\times(n_2+m_2)\times\cdots\times(n_k+m_k)}.$$

- We can do the outer product (a.k.a. tensor or Kronecker p.) of any two (or more) tensors

$$\mathcal{S} \otimes \mathcal{T} = (\mathcal{T}, \mathcal{S})_\otimes \in \mathbb{R}^{n_1\times n_2\times\cdots\times n_k\times m_1\times m_2\times\cdots\times m_t}$$

$$(\mathcal{S} \otimes \mathcal{T})_{i_1,i_2,\ldots,i_k,j_1,j_2,\ldots,j_t} = (\mathcal{T})_{i_1,i_2,\ldots,i_k} \cdot (\mathcal{S})_{j_1,j_2,\ldots,j_t}$$

$$(\mathcal{S} \otimes \mathcal{T})^{\{i_1,i_2,\ldots,i_k\}} = \operatorname{vec}(\mathcal{T}) \left(\operatorname{vec}(\mathcal{S})\right)^{\mathsf{T}}$$

Multiplication: Tensor-matrix (TM) product

The basic structure of TM is the same as for matrices: Sums of products of individual entries of given fibres and col's or rows. Let

$$\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}, \quad S \in \mathbb{R}^{c \times n_\ell}, \quad M \in \mathbb{R}^{n_\ell \times d}.$$

The ℓ-mode (pre-/post-)multiplication of tensor by a matrix

$$S \times_\ell \mathcal{T} \in \mathbb{R}^{n_1 \times \cdots \times n_{\ell-1} \times c \times n_{\ell+1} \times \cdots \times n_k}, \quad \mathcal{T} {}_\ell\!\times M \in \mathbb{R}^{n_1 \times \cdots \times n_{\ell-1} \times d \times n_{\ell+1} \times \cdots \times n_k}$$

is defined as

$$(S \times_\ell \mathcal{T})_{i_1,\dots,i_{\ell-1},j,i_{\ell+1},\dots,i_k} \equiv \sum_{i_\ell=1}^{n_\ell} (S)_{j,i_\ell} \cdot (\mathcal{T})_{i_1,\dots,i_{\ell-1},i_\ell,i_{\ell+1},\dots,i_k},$$
$$(\mathcal{T} {}_\ell\!\times M)_{i_1,\dots,i_{\ell-1},j,i_{\ell+1},\dots,i_k} \equiv \sum_{i_\ell=1}^{n_\ell} (\mathcal{T})_{i_1,\dots,i_{\ell-1},i_\ell,i_{\ell+1},\dots,i_k} \cdot (M)_{i_\ell,j}.$$

Clearly $\mathcal{T} {}_\ell\!\times M = M^\mathsf{T} \times_\ell \mathcal{T}$, thus we focus on the pre-multiplication. (The so-called Einstein's notation omits the 'sum' signs.)

Multiplication: Tensor-matrix (TM) product

We can see it as MV-product of S with all the ℓ-mode fibres, i.e.,

$$(S \times_\ell \mathcal{T})^{\{\ell\}} = S\mathcal{T}^{\{\ell\}} \in \mathbb{R}^{c \times ((\prod_{\kappa=1^k} n_\kappa)/n_\ell)}.$$

Tensor-matrix product is associative in the following two meanings

$$P \times_\ell (S \times_\ell \mathcal{T}) = (PS) \times_\ell \mathcal{T}$$

$$P \times_\tau (S \times_ß \mathcal{T}) = S \times_ß (P \times_\tau \mathcal{T}), \quad \text{for} \quad \tau \neq ß.$$

Multiplication by two matrices in two different modes can be again rearranged by matricization as follows:

$$\left(P \times_\tau (S \times_ß \mathcal{T})\right)^{\{\tau,ß\}} = (S \otimes P)\mathcal{T}^{\{\tau,ß\}} \quad \text{or} \quad (P \otimes S)\mathcal{T}^{\{\tau,ß\}}$$

for $\tau < ß$, or $ß > \tau$, respectively (recall the inverse *lexicographical* ordering of multiindices while matricization).

Linear transformation of a tensor

Employing the associativity while multiplication in different modes, we get for

$$\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}, \quad S_\kappa \in \mathbb{R}^{c_\kappa \times n_\kappa}, \quad \kappa = 1, 2, \ldots, k,$$

$$(S_1, S_2, \ldots, S_k \,|\, \mathcal{T}) \equiv S_1 \times_1 (S_2 \times_2 (\cdots (S_k \times_k \mathcal{T}) \cdots)) \in \mathbb{R}^{c_1 \times c_2 \times \cdots \times c_k}$$

a general linear transformation of $\mathcal{T}$. In the post-mult. fashion it takes form $(\mathcal{T} \,|\, M_1, M_2, \ldots, M_k)$ for $M_\kappa \in \mathbb{R}^{n_\kappa \times d_\kappa}$.

A single tensor-matrix product can be written as

$$P \times_\ell \mathcal{T} = (I_{n_1}, \ldots, I_{n_{\ell-1}}, P, I_{n_{\ell+1}}, \ldots, I_{n_k} \,|\, \mathcal{T}).$$

Employing vectorization gives

$$\mathrm{vec}\big((S_1, S_2, \ldots, S_k \,|\, \mathcal{T})\big) = (S_k \otimes \cdots \otimes S_2 \otimes S_1)\, \mathrm{vec}(\mathcal{T});$$

recall that $\mathrm{vec}(\mathcal{T}) = \mathcal{T}^{\{1,2,\ldots,k\}}$.

Note on tensors of order two

Matrix-matrix product treated as tensor-matrix

First note that $A^{\{1\}} = A$, $A^{\{2\}} = A^{\mathsf{T}}$. Since:

$$\begin{aligned}
(S_1 \times_1 A)^{\{1\}} &= S_1 A^{\{1\}}, \quad \text{then} \quad S_1 \times_1 A = S_1 A,\\
(S_2 \times_2 A)^{\{2\}} &= S_2 A^{\{2\}}, \quad \text{then} \quad S_2 \times_2 A = A S_2^{\mathsf{T}},\\
(S_1, S_2 \,|\, A) &= S_1 \times_1 (S_2 \times_2 A), \quad \text{then} \quad (S_1, S_2 \,|\, A) = S_1 A S_2^{\mathsf{T}},
\end{aligned}$$

for the pre-multiplication and

$$\begin{aligned}
A \;{}_1\!\times M_1 &= M_1^{\mathsf{T}} \times_1 A, \quad \text{then} \quad A \;{}_1\!\times M_1 = M_1^{\mathsf{T}} A,\\
A \;{}_2\!\times M_2 &= M_2^{\mathsf{T}} \times_2 A, \quad \text{then} \quad A \;{}_2\!\times M_2 = A M_2,\\
(A \,|\, M_1, M_2) &= (A \;{}_1\!\times M_1) \;{}_2\!\times M_2, \quad \text{then} \quad (A \,|\, M_1, M_2) = M_1^{\mathsf{T}} A M_2,
\end{aligned}$$

for the post-mutliplication.

For tensors of order one (vectors): $S_1 \times_1 v = S_1 v$, $\quad v \;{}_1\!\times M_1 = M_1^{\mathsf{T}} v$.

Tensor-tensor (TT) product a.k.a. Contraction

Let $\mathcal{T}$ and $\mathcal{F}$ be tensors of orders k and s,

$$\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}, \quad \mathcal{F} \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_s}, \quad \text{and} \quad n_\ell = m_ß.$$

Then their $(\ell, ß)$-mode product is a tensor of order $(k + s - 2)$,

$$\mathcal{T} \times_{(\ell,ß)} \mathcal{F} \in \mathbb{R}^{n_1 \times \cdots \times n_{\ell-1} \times n_{\ell+1} \times \cdots \times n_k \times m_1 \times \cdots \times m_{ß-1} \times m_{ß+1} \times \cdots \times m_s},$$

where
$$\left(\mathcal{T} \times_{(\ell,ß)} \mathcal{F}\right)_{i_1,\dots,i_{\ell-1},i_{\ell+1},\dots,i_k,j_1,\dots,j_{ß-1},j_{ß+1},\dots,j_s} = \sum_{\alpha=1}^{n_\ell} (\mathcal{T})_{i_1,\dots,i_{\ell-1},\alpha,i_{\ell+1},\dots,i_k} \cdot (\mathcal{F})_{j_1,\dots,j_{ß-1},\alpha,j_{ß+1},\dots,j_s}.$$

The other available product is

$$\mathcal{F} \times_{(ß,\ell)} \mathcal{T} = \left(\mathcal{T} \times_{(\ell,ß)} \mathcal{F}\right)^{\Pi}, \quad \text{where} \quad \Pi = \begin{pmatrix} 1 & 2 & \cdots & & & & \cdots & k+s-2 \\ k & k+1 & \cdots & k+s-2 & 1 & 2 & \cdots & k-1 \end{pmatrix}.$$

Alternatively

$$\left(\mathcal{T} \times_{(\ell,ß)} \mathcal{F}\right)^{\{1,2,\dots,k-1\}} = (\mathcal{T}^{\{\ell\}})^{\mathsf{T}} \mathcal{F}^{\{ß\}},$$
$$\left(\mathcal{F} \times_{(ß,\ell)} \mathcal{T}\right)^{\{1,2,\dots,s-1\}} = (\mathcal{F}^{\{ß\}})^{\mathsf{T}} \mathcal{T}^{\{\ell\}}.$$

Tensor-tensor (TT) product a.k.a. Contraction

Analogously, we can introduce mutiplication (contraction) in two pairs of indices at once. For

$$\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}, \quad \mathcal{F} \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_s}, \quad \text{and} \quad n_\ell = m_ß, \; n_\tau = m_\sigma, \; \ell < \tau,$$

we get the $(k+s-4)$-way tensor

$$\mathcal{T} \times_{((\ell,\tau),(ß,\sigma))} \mathcal{F},$$

with entries (depending on relations between ß and σ) either / or

$$\sum\nolimits_{\alpha\beta} (\mathcal{T})_{i_1,\ldots,i_{\ell-1},\alpha,i_{\ell+1},\ldots,i_{\tau-1},\beta,i_{\tau+1},\ldots,i_k} \cdot (\mathcal{F})_{j_1,\ldots,j_{ß-1},\alpha,j_{ß+1},\ldots,j_{\sigma-1},\beta,j_{\sigma+1},\ldots,j_s},$$
$$\sum\nolimits_{\alpha\beta} (\mathcal{T})_{i_1,\ldots,i_{\ell-1},\alpha,i_{\ell+1},\ldots,i_{\tau-1},\beta,i_{\tau+1},\ldots,i_k} \cdot (\mathcal{F})_{j_1,\ldots,j_{\sigma-1},\beta,j_{\sigma+1},\ldots,j_{ß-1},\alpha,j_{ß+1},\ldots,j_s}.$$

Again,

$$\left(\mathcal{T} \times_{((\ell,\tau),(ß,\sigma))} \mathcal{F}\right)^{\{1,2,\ldots,k-2\}} = (\mathcal{T}^{\{\ell,\tau\}})^{\mathsf{T}} (\mathcal{F}^{\Pi})^{\{ß,\sigma\}},$$

and $\Pi = \mathrm{Id}$ or $\left(\begin{smallmatrix} \cdots & \sigma & \cdots & ß & \cdots \\ \cdots & ß & \cdots & \sigma & \cdots \end{smallmatrix}\right)$. Similarly for several pairs of indices.

MM- and TM-products as TT-products

If matrices treated as tensors

Note that TM and TT have different ordering of indices,

$$S \times_\ell \mathcal{T} = \left(S \times_{(2,\ell)} \mathcal{T}\right)^{\left(\begin{smallmatrix} 1 & 2 & \cdots & \ell & \ell+1 & \cdots \\ \ell & 1 & \cdots & \ell-1 & \ell+1 & \cdots \end{smallmatrix}\right)} = \left(\mathcal{T} \times_{(\ell,2)} S\right)^{\left(\begin{smallmatrix} \cdots & \ell-1 & \ell & \cdots & k-1 & k \\ \cdots & \ell-1 & \ell+1 & \cdots & k & \ell \end{smallmatrix}\right)},$$

$$\mathcal{T} {}_\ell\!\times M = M^\mathsf{T} \times_\ell \mathcal{T} = \left(M \times_{(1,\ell)} \mathcal{T}\right)^\Pi = \left(\mathcal{T} \times_{(\ell,1)} M\right)^\Pi.$$

For MM-products we get

$$\begin{aligned} AB &= A \times_{(2,1)} B = A^\mathsf{T} \times_{(1,1)} B = A \times_{(2,2)} B^\mathsf{T} = A^\mathsf{T} \times_{(1,2)} B^\mathsf{T} \\ &= (B \times_{(1,2)} A)^\Pi = (B \times_{(1,1)} A^\mathsf{T})^\Pi = (B^\mathsf{T} \times_{(2,2)} A)^\Pi = (B^\mathsf{T} \times_{(2,1)} A^\mathsf{T})^\Pi \\ &= (B^\mathsf{T} A^\mathsf{T})^\mathsf{T} \end{aligned}$$

where $\Pi = \left(\begin{smallmatrix} 1 & 2 \\ 2 & 1 \end{smallmatrix}\right)$. Similarly for

$$A^\mathsf{T} B = A \times_{(1,1)} B = \dots, \quad AB^\mathsf{T} = A \times_{(2,2)} B = \dots, \quad A^\mathsf{T} B^\mathsf{T} = A \times_{(1,2)} B = \dots.$$

Relation between outer and tensor product

Recall that a vector can be interpreted as a single-column matrix, a matrix as a single-front-slice 3-way tensor, etc.

We formalize that in the form of ‘uparrow’ operator

$$\begin{aligned} \uparrow : \quad v \in \mathbb{R}^n &\longmapsto v^{\uparrow} \in \mathbb{R}^{n\times 1}, \\ A \in \mathbb{R}^{n\times d} &\longmapsto A^{\uparrow} \in \mathbb{R}^{n\times d\times 1}, \\ \uparrow^2 = \uparrow\uparrow : \quad v \in \mathbb{R}^n &\longmapsto v^{\uparrow\uparrow} \in \mathbb{R}^{n\times 1\times 1}, \end{aligned}$$

etc.

Then for a k-way tensor $\mathcal{T}$ and s-way tensor $\mathcal{F}$ we have

$$(\mathcal{T}, \mathcal{F})_{\otimes} = (\mathcal{T}^{\uparrow}) \times_{(k+1, s+1)} (\mathcal{F}^{\uparrow}).$$

Note again:
The outer product is a.k.a. tensor and Kronecker product.
The tensor (TT) product is a.k.a. contraction.

Basic decompositions of a tensor

Singular value decomposition (SVD)

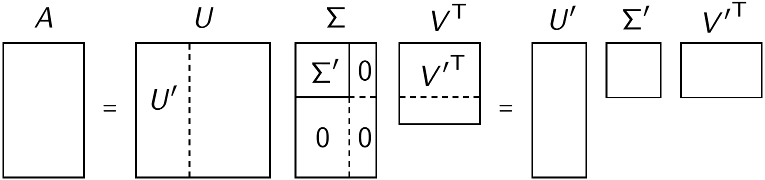
Let start with matrices

Let $A \in \mathbb{R}^{m \times n}$ be a matrix of rank $r = \mathrm{rank}(A)$, then

$$A = U \Sigma V^{\mathsf{T}} = (U, V \,|\, \Sigma) = U' \Sigma' V'^{\mathsf{T}} = (U', V' \,|\, \Sigma')$$

where
$$U^{-1} = U^{\mathsf{T}}, \quad U = [\, U' \,,\, U'' \,] \in \mathbb{R}^{m \times m}, \quad U' \in \mathbb{R}^{m \times r},$$
$$V^{-1} = V^{\mathsf{T}}, \quad V = [\, V' \,,\, V'' \,] \in \mathbb{R}^{n \times n}, \quad V' \in \mathbb{R}^{n \times r},$$

$$\Sigma = \begin{bmatrix} \Sigma' & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{m \times n}, \qquad \Sigma' = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_r) \in \mathbb{R}^{r \times r},$$
$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0.$$



SVDs of ℓ-mode matricizations

Let $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$ of $\overrightarrow{\mathrm{rank}}(\mathcal{T}) \equiv (r_1, r_2, \ldots, r_k)$, where

$$r_\ell = \mathrm{rank}_{\{\ell\}}(\mathcal{T}) = \mathrm{rank}(\mathcal{T}^{\{\ell\}}), \quad \mathcal{T}^{\{\ell\}} \in \mathbb{R}^{n_\ell \times (N/n_\ell)}, \quad N = n_1 \cdot n_2 \cdot \cdots \cdot n_k.$$

Consider then the SVDs

$$\mathcal{T}^{\{\ell\}} = U_\ell \Sigma_\ell V_\ell^{\mathsf{T}} = U'_\ell \Sigma'_\ell {V'_\ell}^{\mathsf{T}}$$

where $U_\ell = [U'_\ell,\ U''_\ell] \in \mathbb{R}^{n_\ell \times n_\ell}$, $U'_\ell \in \mathbb{R}^{n_\ell \times r_\ell}$,

$$\Sigma'_\ell = \mathrm{diag}(\sigma_{1,\ell}, \sigma_{2,\ell}, \ldots, \sigma_{r_\ell,\ell}) \in \mathbb{R}^{r_\ell \times r_\ell}, \quad \sigma_{1,\ell} \geq \sigma_{2,\ell} \geq \cdots \geq \sigma_{r_\ell,\ell} > 0.$$

Thus

$$\begin{bmatrix} {U'_\ell}^{\mathsf{T}} \mathcal{T}^{\{\ell\}} \\ {U''_\ell}^{\mathsf{T}} \mathcal{T}^{\{\ell\}} \end{bmatrix} = {U_\ell}^{\mathsf{T}} \mathcal{T}^{\{\ell\}} = \Sigma_\ell V_\ell^{\mathsf{T}} = \begin{bmatrix} \Sigma'_\ell {V'_\ell}^{\mathsf{T}} \\ 0 \end{bmatrix}.$$

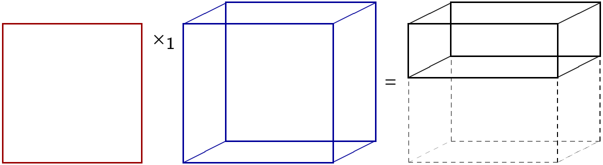
SVDs of ℓ-mode matricizations

Clearly, this is the ℓ-mode product,

$$\left({U_\ell}^{\mathsf{T}} \times_\ell \mathcal{T}\right)^{\{\ell\}} = {U_\ell}^{\mathsf{T}} \mathcal{T}^{\{\ell\}} = \Sigma_\ell V_\ell^{\mathsf{T}} = \begin{bmatrix} \Sigma'_\ell {V'_\ell}^{\mathsf{T}} \\ 0 \end{bmatrix} \in \mathbb{R}^{n_\ell \times (N/n_\ell)},$$

$$\text{and} \quad \left({U'_\ell}^{\mathsf{T}} \times_\ell \mathcal{T}\right)^{\{\ell\}} = {U'_\ell}^{\mathsf{T}} \mathcal{T}^{\{\ell\}} = \Sigma'_\ell {V'_\ell}^{\mathsf{T}} \in \mathbb{R}^{r_\ell \times (N/n_\ell)}.$$

For a three-way tensor and $\ell = 1$:



Note that mutliplication by other $U_ß$s in the other modes ($ß \neq \ell$) does not involve these already made zero co-fibres.

Tucker decompostion a.k.a. high-order SVD (HOSVD)

Finally we get for $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$ a linear transformation

$$\left(U_1^\mathsf{T}, U_2^\mathsf{T}, \ldots, U_k^\mathsf{T} \,\middle|\, \mathcal{T} \right) = \mathrm{diag}_k(\mathcal{C}_\mathcal{T}, 0) \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k},$$

where the subtensor

$$\mathcal{C}_\mathcal{T} = \left({U_1'}^\mathsf{T}, {U_2'}^\mathsf{T}, \ldots, {U_k'}^\mathsf{T} \,\middle|\, \mathcal{T} \right) \in \mathbb{R}^{r_1 \times r_2 \times \cdots \times r_k}$$

is called the Tucker core of tensor $\mathcal{T}$. Since U_ℓ's are invertible and orthogonal the first equation can be rearranged to

$$\mathcal{T} = \left(U_1, U_2, \ldots, U_k \,\middle|\, \mathrm{diag}_k(\mathcal{C}_\mathcal{T}, 0) \right) = \left(U_1', U_2', \ldots, U_k' \,\middle|\, \mathcal{C}_\mathcal{T} \right)$$

that is called the Tucker decomposition or HOSVD of tensor $\mathcal{T}$.

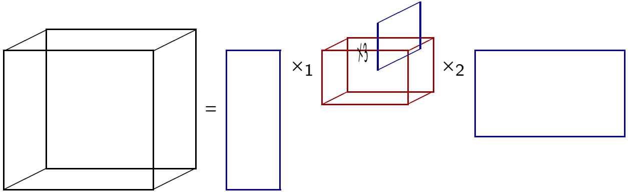
[*L. R. Tucker*, Psychometrika 31(3), pp. 279–311, 1966]

Tucker decompostion a.k.a. high-order SVD (HOSVD)

Thus, for $\mathcal{T}$ with $\overrightarrow{\mathrm{rank}}(r_1, r_2, \ldots, r_k)$ we have decomposition

$$\mathcal{T} = \left(U'_1, U'_2, \ldots, U'_k \,\middle|\, \mathcal{C}_{\mathcal{T}}\right), \qquad \mathcal{C}_{\mathcal{T}} \in \mathbb{R}^{r_1 \times r_2 \times \cdots \times r_k},$$

$$U'_\ell \in \mathbb{R}^{n_\ell \times r_\ell}, \quad {U'_\ell}^{\mathsf{T}} U'_\ell = I_{r_\ell}.$$

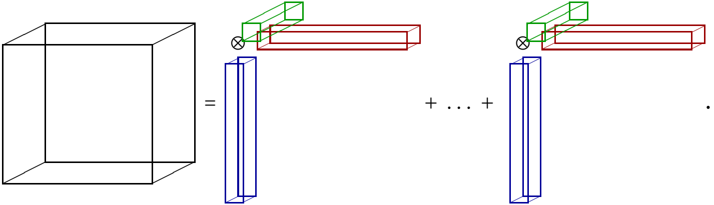


Moreover, the ℓ-mode co-fibres of $\mathcal{C}_{\mathcal{T}}$ are sorted in a nonincreasing sequence w.r.t. their norms equal to $\sigma_{1,\ell}, \sigma_{2,\ell}, \ldots, \sigma_{r_\ell,\ell}$.

This allows to generalize the Eckart–Young–Mirsky theorem.
Compare with the SVD.

Polyadic expansion as the CP decompostion

Recall the polyadic decompostion of $\mathcal{T}$

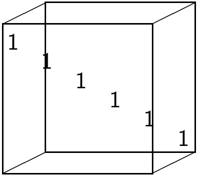


Collecting all the particular vectors into matrices

$$X_1 \in \mathbb{R}^{n_1 \times r}, \quad X_2 \in \mathbb{R}^{n_2 \times r}, \quad \ldots \quad X_k \in \mathbb{R}^{n_k \times r}$$

and using an "identity-like" cubic tensor of order k and dim's r,

$\mathcal{I}_{r,k} =$



$\in \mathbb{R}^{r \times r \times \cdots \times r}$, we get

$$\mathcal{T} = (X_1, X_2, \ldots, X_k \,|\, \mathcal{I}_{r,k}) .$$

Comparison of both basic decompositions

Tucker decomposition (HOSVD)

$$\mathcal{T} = (U'_1, U'_2, \ldots, U'_k \,|\, \mathcal{C}_{\mathcal{T}})$$

- Matrices U'_ℓ with orthonormal columns (+)
- Different numbers of columns equal to $\mathrm{rank}_{\{\ell\}}(\mathcal{T})$ ($\pm$)
- Core of dimensions equal to $\overrightarrow{\mathrm{rank}}(\mathcal{T})$ with the norm "accumulated" in leading principal corner (+)

CP decoposition (CanDeComp, ParaFac)

$$\mathcal{T} = (X_1, X_2, \ldots, X_k \,|\, \mathcal{I}_{r,k})$$

- Matrices X_ℓ may have linearly dependent columns (−)
- The same number of columns equal to $\mathrm{polyrank}(\mathcal{T})$ ($\pm$)
- "Core tensor" is cubic with *very simple* structure; so simple it need not be stored (+ + +)

Note that both decompostitions have similar structure—an inner core tensor of (typically?) smaller dimensions than $\mathcal{T}$, surrounded by k matrices, also called leaves (from graph theory).

Low-rank arithmetics of tensors

Let start with matrices. SVD (re)compression

Let $A \in \mathbb{R}^{m\times n}$ be a (low-rank) matrix given in the form of product of two thin matrices $A = XY^\mathsf{T}$, or, in more general case of three

$$A = XSY^\mathsf{T},\ \ X \in \mathbb{R}^{m\times p},\ m \gg p,\ \ S \in \mathbb{R}^{p\times q},\ \ Y \in \mathbb{R}^{n\times q},\ n \gg q.$$

Our goal is to compute its SVD without evaluating A:
Step 1: Compute economic QR decompositions of *thin* X and Y

$$X = Q_X R_X,\quad Q_X \in \mathbb{R}^{m\times r_X},\quad R_X \in \mathbb{R}^{r_X\times p},\quad r_X = \operatorname{rank}(X),$$
$$Y = Q_Y R_Y,\quad Q_Y \in \mathbb{R}^{n\times r_Y},\quad R_Y \in \mathbb{R}^{r_Y\times q},\quad r_Y = \operatorname{rank}(Y).$$

Thus $A = Q_X W Q_Y{}^\mathsf{T}$ where $W = R_X S R_Y{}^\mathsf{T} \in \mathbb{R}^{r_X\times r_Y}$.
Step 2: Compute the economic SVD of the *small* matrix W

$$W = U'_W \Sigma'_W V'_W{}^\mathsf{T},\quad U'_W \in \mathbb{R}^{r_X\times r},\ \Sigma'_W \in \mathbb{R}^{r\times r},\ V'_W \in \mathbb{R}^{r_Y\times r}.$$

Thus $A = (Q_X U'_W)\,\Sigma'_W\,(Q_Y V'_W)^\mathsf{T}$.

Sum of two low-rank matrices

Let $A, B \in \mathbb{R}^{m \times n}$ be two low-rank matrices given the form of their economic SVDs,

$$A = U'_A \Sigma'_A {V'_A}^\mathsf{T}, \quad B = U'_B \Sigma'_B {V'_B}^\mathsf{T},$$

with $r_A = \mathrm{rank}(A)$, $r_B = \mathrm{rank}(B)$.

Then

$$M = \varphi A + \psi B = \underbrace{\left[\, U'_A \,,\, U'_B \,\right]}_{X \in \mathbb{R}^{m \times (r_A + r_B)}} \underbrace{\begin{bmatrix} \varphi \Sigma'_A & 0 \\ 0 & \psi \Sigma'_B \end{bmatrix}}_{S \in \mathbb{R}^{(r_A + r_B) \times (r_A + r_B)}} \underbrace{\left[\, V'_A \,,\, V'_B \,\right]}_{Y \in \mathbb{R}^{n \times (r_A + r_B)}}^\mathsf{T}.$$

Compression then serves the economic SVD of M.

Product of low-rank matrix with another matrix

Let $A \in \mathbb{R}^{m\times n}$ be a low-rank matrix given the form of its economic SVD,

$$A = U_A' \Sigma_A' {V_A'}^{\mathsf{T}}.$$

If also B is a low-rank matrix given similarly, then

$$M = AB = \underbrace{U_A'}_{Q_X} \underbrace{\left(\Sigma_A' ({V_A'}^{\mathsf{T}} U_B') \Sigma_B'\right)}_{W \in \mathbb{R}^{r_A \times r_B}} \underbrace{V_B'}_{Q_Y}{}^{\mathsf{T}}.$$

If B is a general matrix, then

$$M = AB = \underbrace{U_A'}_{Q_X} \underbrace{\Sigma_A'}_{R_X S} \underbrace{(B^{\mathsf{T}} V_A')}_{Y}{}^{\mathsf{T}}.$$

Compression (which is already partially done) then serves the economic SVD of M.

And similarly for tensors: Compression

Let

$$\mathcal{T} = (X_1, X_2, \ldots, X_k \,|\, \mathcal{S}) \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}, \quad \mathcal{S} \in \mathbb{R}^{p_1 \times p_2 \times \cdots \times p_k}, \quad n_\ell \gg p_\ell$$

(e.g. the CP decomp. / polyadic exp., or another similar product).
Step 1: Compute k economic QR decomp's of *thin* $X_\ell = Q_\ell R_\ell$,

$$(X_1, X_2, \ldots, X_k \,|\, \mathcal{S}) = \Big(Q_1, Q_2, \ldots, Q_k \,\Big|\, \underbrace{(R_1, R_2, \ldots, R_k \,|\, \mathcal{S})}_{\mathcal{W}} \Big).$$

Step 2: Compute the Tucker decomposition of *small* tensor $\mathcal{W}$,

$$\mathcal{W} = (U'_{1,\mathcal{W}}, U'_{2,\mathcal{W}}, \ldots, U'_{k,\mathcal{W}} \,|\, \mathcal{C}_{\mathcal{W}}).$$

This gives

$$\mathcal{T} = \Big(\underbrace{Q_1 U'_{1,\mathcal{W}}}_{U'_{1,\mathcal{T}}}, \underbrace{Q_2 U'_{2,\mathcal{W}}}_{U'_{2,\mathcal{T}}}, \ldots, \underbrace{Q_k U'_{k,\mathcal{W}}}_{U'_{k,\mathcal{T}}} \,|\, \underbrace{\mathcal{C}_{\mathcal{W}}}_{\mathcal{C}_{\mathcal{T}}} \Big)$$

the Tucker decomposition of large tensor $\mathcal{T}$.

Sum of two tensors

Let $\mathcal{T}, \mathcal{F} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$ in Tucker form

$$\mathcal{T} = (U'_{1,\mathcal{T}}, U'_{2,\mathcal{T}}, \ldots, U'_{k,\mathcal{T}} | \mathcal{C}_{\mathcal{T}}), \quad \mathcal{F} = (U'_{1,\mathcal{F}}, U'_{2,\mathcal{F}}, \ldots, U'_{k,\mathcal{F}} | \mathcal{C}_{\mathcal{F}}).$$

Then

$$\mathcal{E} = \varphi\mathcal{T} + \psi\mathcal{F} = \Big(\underbrace{[U'_{1,\mathcal{T}}, U'_{1,\mathcal{F}}]}_{X_1}, \ldots, \underbrace{[U'_{k,\mathcal{T}}, U'_{k,\mathcal{F}}]}_{X_k} \Big| \underbrace{\mathrm{diag}_k(\varphi\mathcal{C}_{\mathcal{T}}, \psi\mathcal{C}_{\mathcal{F}})}_{\mathcal{S}} \Big).$$

The compression then yields the Tucker decomposition of $\mathcal{E}$.

Cost: Instead of n^k of sums of two number, we need to do:

- k-times the economic QR decomposition of $n \times r$ matrix;
- k-times the product of $(r^{\times k})$-tensor with $(r \times r)$-matrix;
- one Tucker decompostion of $(r^{\times k})$-tensor;
- k-times the product of $(n \times r)$-matrix with $(r \times r)$-matrix.

(Here $n = \max\{n_1, n_2, \ldots, n_k\}$ and $r = \max\{r_1, r_2, \ldots, r_k\}$.)

Tensor matrix product

Let $\mathcal{T}, \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$ in Tucker form

$$\mathcal{T} = (U'_{1,\mathcal{T}}, U'_{2,\mathcal{T}}, \ldots, U'_{k,\mathcal{T}} | \mathcal{C}_{\mathcal{T}}), \quad \text{and} M \in \mathbb{R}^{m \times n_\ell}$$

Then

$$\mathcal{E} = M \times_\ell \mathcal{T} = (U'_{1,\mathcal{T}}, \ldots, \underbrace{MU'_{\ell,\mathcal{T}}}_{X_\ell}, \ldots, U'_{k,\mathcal{T}} | \mathcal{C}_{\mathcal{T}}).$$

The compression then yields the Tucker decomposition of $\mathcal{E}$.

Cost: Instead of n^{k-1} of MV products, we need to do:

- r-times the MV product;
- one economic QR decomposition of $n \times r$ matrix;
- one Tucker decompostion of $(r^{\times k})$-tensor;
- one product of $(r^{\times k})$-tensor with $(r \times r)$-matrix;
- k-times the product of $(n \times r)$-matrix with $(r \times r)$-matrix.

Note on norm and scalar product

Recall that

$$\langle \mathcal{T}, \mathcal{F} \rangle = \mathrm{vec}(\mathcal{F})^{\mathsf{T}} \mathrm{vec}(\mathcal{T}), \qquad \|\mathcal{T}\| = (\langle \mathcal{T}, \mathcal{T} \rangle)^{\frac{1}{2}},$$

$$\mathcal{T} = (U'_{1,\mathcal{T}}, U'_{2,\mathcal{T}}, \dots, U'_{k,\mathcal{T}} | \mathcal{C}_{\mathcal{T}}),$$

$$\mathrm{vec}(\mathcal{T}) = \left(U'_{k,\mathcal{T}} \otimes \cdots \otimes U'_{2,\mathcal{T}} \otimes U'_{1,\mathcal{T}} \right) \mathrm{vec}(\mathcal{C}_{\mathcal{T}}),$$

and similarly for $\mathcal{F}$. Then $\langle \mathcal{T}, \mathcal{F} \rangle$

$$= \mathrm{vec}(\mathcal{C}_{\mathcal{F}})^{\mathsf{T}} \left(U'_{k,\mathcal{F}} \otimes \ldots \otimes U'_{1,\mathcal{F}} \right)^{\mathsf{T}} \left(U'_{k,\mathcal{T}} \otimes \ldots \otimes U'_{1,\mathcal{T}} \right) \mathrm{vec}(\mathcal{C}_{\mathcal{T}})$$

$$= \mathrm{vec}(\mathcal{C}_{\mathcal{F}})^{\mathsf{T}} \left(\left(U'^{\mathsf{T}}_{1,\mathcal{F}} U'_{k,\mathcal{T}} \right) \otimes \cdots \otimes \left(U'^{\mathsf{T}}_{1,\mathcal{F}} U'_{k,\mathcal{T}} \right) \right) \mathrm{vec}(\mathcal{C}_{\mathcal{T}})$$

$$= \mathrm{vec}(\mathcal{C}_{\mathcal{F}})^{\mathsf{T}} \, \mathrm{vec}\left(\left(U'^{\mathsf{T}}_{1,\mathcal{F}} U'_{k,\mathcal{T}} \right), \dots, \left(U'^{\mathsf{T}}_{1,\mathcal{F}} U'_{k,\mathcal{T}} \right) \middle| \mathcal{C}_{\mathcal{T}} \right)$$

but also

$$= \mathrm{vec}\left(\left(U'^{\mathsf{T}}_{1,\mathcal{T}} U'_{k,\mathcal{F}} \right), \dots, \left(U'^{\mathsf{T}}_{1,\mathcal{T}} U'_{k,\mathcal{F}} \right) \middle| \mathcal{C}_{\mathcal{F}} \right)^{\mathsf{T}} \mathrm{vec}(\mathcal{C}_{\mathcal{T}})$$

one of the last two lines needs to be evaluated (note that one core may be smaller than the other).

Why to do such complicated arithmetics?

Consider the following problem

$$\mathscr{A}(\mathcal{X}) = \mathcal{B}, \quad \text{where} \quad \mathscr{A} \in \mathscr{L}(\mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}, \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k})$$

and $\mathcal{B}$ are given and the goal is to find $\mathcal{X}$.

For example: The Lyapunov operator on $\mathbb{R}^{n \times n}$,

$$\mathscr{A}(X) = AX + XA^{\mathsf{T}}, \quad \mathrm{vec}(\mathscr{A}(X)) = (I \otimes A + A \otimes I)\mathrm{vec}(X).$$

For rank-one rhs $B = bb^{\mathsf{T}}$, $b \neq 0$, the solution X is of full rank with exponentially decaying singular values.

If A is SPD, then also $\mathscr{A}$ is SPD, and then, e.g., the method of conjugate gradients (CG) can be used for solving $\mathscr{A}(\mathcal{X}) = \mathcal{B}$. With an initial guess $\mathcal{X}_0 = (0, 0, \ldots, 0 | 0)$ and employing the low-rank arithmetics, we get solution in Tucker format.

Cost of CG iteration is changing, it depends on ranks!
(Truncation, open pbs.)

A final note on Tucker decomposition

First note that the "Tucker-like" decompositions

$$\mathcal{T} = (U_1', U_2', \ldots, U_k', \mathcal{C}_{\mathcal{T}}) \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$$

are not sufficient (from the computational point of view) for handling really large tensors.

Let $\overrightarrow{\mathrm{rank}}(\mathcal{T}) = (r_1, r_2, \ldots, r_k)$, i.e., the Tucker core

$$\mathcal{C}_{\mathcal{T}} \in \mathbb{R}^{r_1 \times r_2 \times \cdots \times r_k} \quad \text{and let} \quad r_1 = r_2 = \cdots = r_k = 2.$$

Then the memory requirement to store $\mathcal{T}$ are roughly

$$\underbrace{k \cdot (n \cdot 2)}_{U_\ell'} + \underbrace{2^k}_{\mathcal{C}_{\mathcal{T}}} \approx 2^k,$$

i.e., for example for $k = 100$ we need to store

$$\approx 2^{100} \approx 1.2677 \cdot 10^{30} \text{ numbers } \approx 9.2234 \cdot 10^{18} \text{ TiB in doubles.}$$

Graph interpretation:

Tensor networks & Hierarchical formats

Tensors & graphs

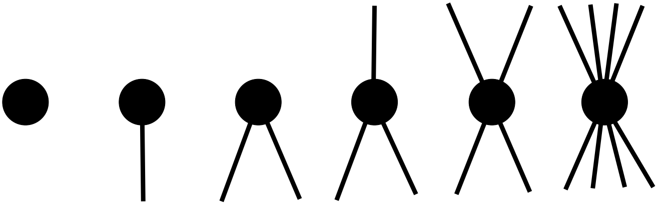
To simplify a bit our notion about tensors, tensor products and tensor decompositions, we employ the graph theory.

Any tensor $\mathcal{T}$ is interpreted as a *graph vertex*, and number of indices of $\mathcal{T}$ as the *degree of the vertex*.

Thus the scalar, vector, matrix, 3-, 4-, and, e.g., 8-way tensors

$$t, \qquad t_i, \qquad t_{i,j}, \qquad t_{i_1,i_2,i_3}, \qquad t_{i_1,\dots,i_4}, \qquad t_{i_1,\dots,i_8}$$

are interpreted as

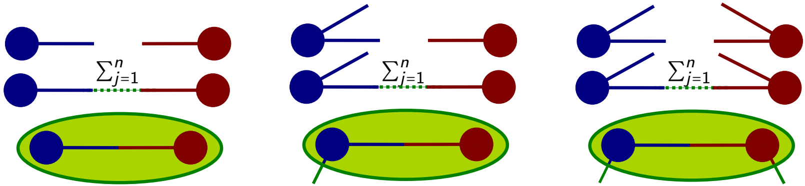


Basic products

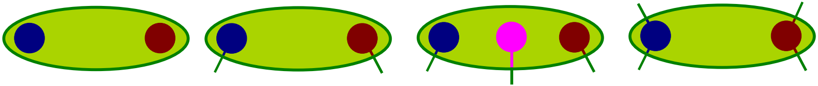
Scalar, MV, and MM-products can be then drawn as follows:

$y \in \mathbb{R}^n,\ x \in \mathbb{R}^n$ $\quad$ $A \in \mathbb{R}^{m\times n},\ x \in \mathbb{R}^n$ $\quad$ $A \in \mathbb{R}^{m\times n},\ B \in \mathbb{R}^{n\times d}$

$y^{\mathsf{T}} x = \alpha \in \mathbb{R}$ $\quad$ $Ax = y \in \mathbb{R}^m$ $\quad$ $AB = C \in \mathbb{R}^{m\times d}$

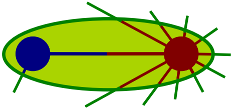


Prod. of scalars, outer prod's. of (two and three) vec's and mat's:



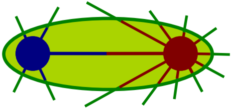
Products involving tensors

- Tensor-matrix product (pre- or post-multiplication)



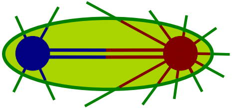
$\sim \qquad \mathcal{W} = M \times_{\ell} \mathcal{T},$

- Tensor-tensor product (contraction)



$\sim \qquad \mathcal{W} = \mathcal{F} \times_{(ß,\ell)} \mathcal{T},$

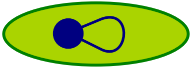
- Tensor-tensor product (contraction) in several pairs of indices at once



$\sim \qquad \mathcal{W} = \mathcal{F} \times_{((ß,\sigma),(\ell,\tau))} \mathcal{T}.$

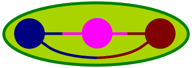
It allows us to be more creative :-)

- A product of matrix $A \in \mathbb{R}^{n \times n}$ with itself?



$$\sim \qquad \sum_{i=1}^{n} a_{i,i} = \mathrm{trace}(A)$$

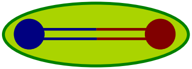
- A circular product of matrices $A \in \mathbb{R}^{m \times n}$, $B \in \mathbb{R}^{n \times d}$, $C \in \mathbb{R}^{d \times m}$?



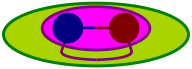
$$\sim \qquad \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{\ell=1}^{d} a_{i,j} \cdot b_{j,\ell} \cdot c_{\ell,i}$$

- But recall the scalar product of tensors! For matrices $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{m \times n}$ it takes form of both—the *circular product* and *product of a matrix with itself* :-)

$$\langle A, B \rangle = \sum_{i=1}^{m} \sum_{j=1}^{n} b_{i,j} \cdot a_{i,j} = \mathrm{trace}(B^{\top} A)$$



=

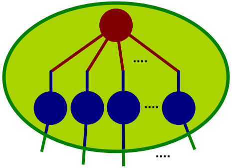


Tucker decomposition

Graph of the Tucker decompostion

$$\mathcal{T} = (U_1', U_2', U_3', \ldots, U_k' \,|\, \mathcal{C}_{\mathcal{T}})$$

takes form

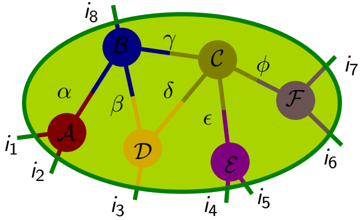


Our goal is to break up the high-order core tensor $\mathcal{C}_{\mathcal{T}}$ to product of several lower-orders tensors. Computationally, we want to replace the core as it is, whos number of entries scales exponentially ($\approx r^k$) with the tensor order k, by a set of tensors, whos number of entries scales linearly or logarithmically with k. How to do it can be easily understood by using graphs.

A general tensor network

By a general tensor network we understand interpretation of a high-order tensor $\mathcal{T}$ as a (prescribed) structured product of a set of lower-order tensors.

The tensor network can be seen as a (de)composition or approximation framework of the tensor $\mathcal{T}$.



$$t_{i_1,i_2,i_3,i_4,i_5,i_6,i_7,i_8} = \sum_{\alpha,\beta,\gamma,\delta,\epsilon,\phi} a_{i_1,i_2,\alpha} \cdot b_{\alpha,\beta,\gamma,i_8} \cdot c_{\gamma,\delta,\epsilon,\phi} \cdot d_{\beta,i_3,\delta} \cdot e_{i_4,i_5,\epsilon} \cdot f_{\phi,i_6,i_7}$$

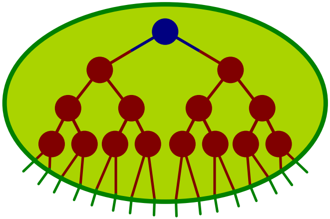
$$n^8 \longrightarrow 4n^3 + 2n^4$$

The simples structure for decomposing tensor is a (binary) tree (it avoids computationally complicated circles).

Tree decomposition of the Tucker core

Recall $\mathcal{T} = (U_1', U_2', \ldots, U_k' \,|\, \mathcal{C}_{\mathcal{T}})$. There are two different extremes:

The balanced (as much as possible) **binary tree**

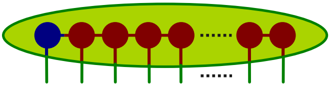


$$r^k \longrightarrow (k-2)r^3 + r^2 \approx kr^3$$

So-called hierarchical Tucker decompostion (HTD).

[*L. Grasedyck*, SIMAX 31(4), 2010]

The most-unbalanced binary tree



$$r^k \longrightarrow (k-2)r^3 + 2r^2 \approx kr^3$$

So-called tensor train decompostion (TTD).

[*I. V. Oseledets*, SISC 33(5), 2011]

The blue two-way tensors (matrices) are roots of these binary trees.

How to find the prescribed tree structure?

The root

The root is always a tensor of second order (a matrix). Let, for simplicity, the indices (modes) of the whole core $\mathcal{C} \in \mathbb{R}^{r_1 \times r_2 \times \cdots \times r_k}$ be ordered in such a way that

$$i_1, i_2, \ldots, i_t \qquad \text{and} \qquad i_{t+1}, i_{t+2}, \ldots, i_k$$

correspond to the left and right branches, respectively.

Thus, for HTD and even k, $t = k/2$; for TTD $t = 1$.

Consider the economic SVD of the matricizaton of $\mathcal{C}$

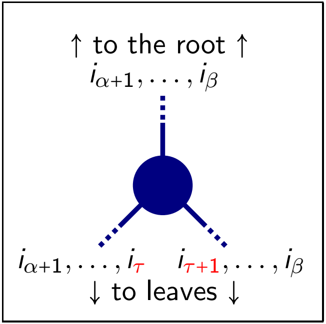
$$\mathcal{C}^{\mathscr{R}} = U'_{\mathscr{R}} \Sigma'_{\mathscr{R}} {V'_{\mathscr{R}}}^{\mathsf{T}}, \qquad \text{where} \qquad \mathscr{R} = \{1, 2, \ldots, t\},$$

- Then the matrix $\Sigma'_{\mathscr{R}}$ is the root of the tree and
- matrices $U'_{\mathscr{R}}$, $V'_{\mathscr{R}} = U'_{\mathscr{C}}$ can be decomposed into left and right branches of the tree, respectively; $\mathscr{C} = \{1, \ldots, k\} \setminus \mathscr{R}$.

How to find the prescribed tree structure?

A single vertex of degree three

Since indices of $\mathcal{C}$ are order properly, any vertex of deg.3 looks like:



Let us consider three corresponding matricizations and their economic SVDs:

$$\mathcal{C}^{\{\alpha+1,\dots,\beta\}} = U'_{\{\alpha+1,\dots,\beta\}} \Sigma'_{\{\alpha+1,\dots,\beta\}} {V'_{\{\alpha+1,\dots,\beta\}}}^{\mathsf{T}},$$

$$\mathcal{C}^{\{\alpha+1,\dots,\tau\}} = U'_{\{\alpha+1,\dots,\tau\}} \Sigma'_{\{\alpha+1,\dots,\tau\}} {V'_{\{\alpha+1,\dots,\tau\}}}^{\mathsf{T}},$$

$$\mathcal{C}^{\{\tau+1,\dots,\beta\}} = U'_{\{\tau+1,\dots,\beta\}} \Sigma'_{\{\tau+1,\dots,\beta\}} {V'_{\{\tau+1,\dots,\beta\}}}^{\mathsf{T}}.$$

The key theorem of all tree-form decomp's (HTD, TTD, ...) says:

$$\mathrm{range}\Big(U'_{\{\alpha+1,\dots,\beta\}}\Big) \subseteq \mathrm{range}\Big(U'_{\{\tau+1,\dots,\beta\}} \otimes U'_{\{\alpha+1,\dots,\tau\}}\Big).$$

How to find the prescribed tree structure?

Tensor-tree-decomposition theorem

Theorem:

$$\operatorname{range}\Big(U'_{\{\alpha+1,\dots,\beta\}}\Big) \subseteq \operatorname{range}\Big(U'_{\{\tau+1,\dots,\beta\}} \otimes U'_{\{\alpha+1,\dots,\tau\}}\Big), \quad \alpha < \tau < \beta.$$

Sketch of the proof: Any column of $\mathcal{C}^{\{\dots\}}$ is a vector $v \in \mathbb{R}^{(\beta-\alpha)}$, that can be reshaped into a matrix $M \in \mathbb{R}^{(\tau-\alpha)\times(\alpha-\beta)}$, $v = \operatorname{vec}(M)$.

Note that columns of M are in $\operatorname{range}(U'_{\{\dots\}}) = \operatorname{range}(\mathcal{C}^{\{\dots\}})$ and rows of M in $\operatorname{range}(U'_{\{\dots\}}) = \operatorname{range}(\mathcal{C}^{\{\dots\}})$. Thus

$$\underbrace{M = \mathcal{C}^{\{\dots\}}\mathcal{C}^{\{\dots\}\dagger} M \qquad \text{and} \qquad M^\mathsf{T} = \mathcal{C}^{\{\dots\}}\mathcal{C}^{\{\dots\}\dagger} M^\mathsf{T}}$$

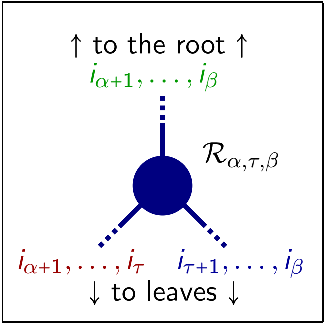
$$M = \mathcal{C}^{\{\dots\}} \underbrace{\mathcal{C}^{\{\dots\}\dagger} M \, \mathcal{C}^{\{\dots\}\dagger\mathsf{T}}} \mathcal{C}^{\{\dots\}\mathsf{T}}$$

giving $\operatorname{vec}(M) = v = \Big(\mathcal{C}^{\{\dots\}} \otimes \mathcal{C}^{\{\dots\}}\Big)\Big(\mathcal{C}^{\{\dots\}\dagger} \otimes \mathcal{C}^{\{\dots\}\dagger}\Big) v.$ □

How to find the prescribed tree structure?

How to employ the tensor-tree-decomposition theorem?

Denote the three-way tensor $\mathcal{R}_{\alpha,\tau,\beta}$. Since



$$\mathrm{range}(U'_{\{\dots\}}) \subseteq \mathrm{range}(U'_{\{\dots\}} \otimes U'_{\{\dots\}})$$

there exists a matrix R such that

$$U'_{\{\dots\}} = (U'_{\{\dots\}} \otimes U'_{\{\dots\}})R, \quad R^\top R = I$$

$$R \in \mathbb{R}^{(\mathrm{rank}_{\{\dots\}}(\mathcal{C}) \cdot \mathrm{rank}_{\{\dots\}}(\mathcal{C})) \times (\mathrm{rank}_{\{\dots\}}(\mathcal{C}))}$$

It remains to interpret $R = \mathcal{R}^{\{1,2\}}_{\alpha,\tau,\beta}$ so

$$\mathcal{R}_{\alpha,\tau,\beta} \in \mathbb{R}^{(\mathrm{rank}_{\{\dots\}}(\mathcal{C})) \times (\mathrm{rank}_{\{\dots\}}(\mathcal{C})) \times (\mathrm{rank}_{\{\dots\}}(\mathcal{C}))}$$

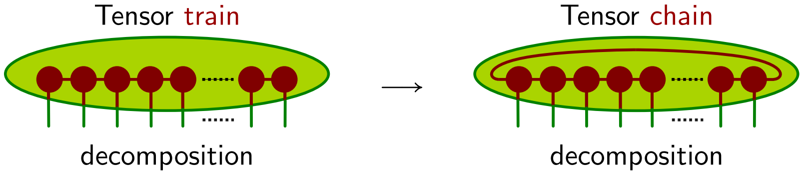
Doing this with all deg.3 vertices yields the HTD with any binary tree (recall the matrices on leaves). The last tensor of order two in TTD is just an identity matrix.

It can be applied on any (not necessarily binary) tree-form decomp.

A few notes on hierarchical / tree-form decompositions

- There is a lot of different ranks of $\mathcal{T}$ in the game (dimensions of cubes).
- To be efficent, these ranks needs to be small.
- To be effective, $\mathcal{T}$ has to be *either* of low rank, *or* well approximable by a such low rank tensor.
- Otherwise we are not able to manage $\mathcal{T}$ in this way.
- Design of the tree should reflect knowledge about the problem.
- Employ symmetries between modes (if there are; $t_{i,j,\ell} = t_{j,i,\ell}$).

Note that there are also cyclic decompositions:

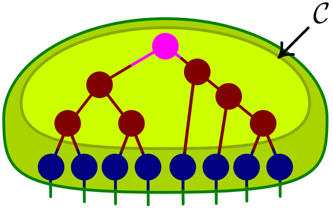


A few notes on hierarchical / tree-form decompositions

Recall that we first did the Tucker decomposition of a tensor and now the tree-form decomposition of the Tucker core.

Both together gives the HTD with structure like:

$$\mathcal{T} = (U'_1, U'_2, \dots, U'_k \,|\, \mathcal{C}) =$$



Note that in this particular case $\mathscr{R} = \{1, 2, 3, 4\}$,

$$\mathcal{T}^{\mathscr{R}} = \overbrace{\left(U'_4 \otimes U'_3 \otimes U'_2 \otimes U'_1\right)\left(\mathcal{R}^{\{1,2\}}_{2,3,4} \otimes \mathcal{R}^{\{1,2\}}_{0,1,2}\right)\left(\mathcal{R}^{\{1,2\}}_{0,2,4}\right)}^{U'_{\mathscr{R}}} \Sigma'_{\mathscr{R}}$$

$$\Big(\underbrace{\left(U'_8 \otimes U'_7 \otimes U'_6 \otimes U'_5\right)\left(\mathcal{R}^{\{1,2\}}_{6,7,8} \otimes I\right)\left(\mathcal{R}^{\{1,2\}}_{5,6,8} \otimes I\right)\left(\mathcal{R}^{\{1,2\}}_{4,5,8}\right)}_{V'_{\mathscr{R}}}\Big)^{\mathsf{T}}$$

Arithmetics of hierarchical Tucker

Motivation

Recall that we want to solve, e.g.,

$$\mathscr{A}(\mathcal{X}) = \mathcal{B}, \qquad \text{where} \qquad \mathcal{X},\ \mathcal{B} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k},$$

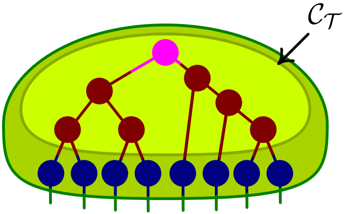
where $\mathscr{A}$ is symmetric positive definite (SPD) typically represented by one or more sparse matrices in outer (Kronecker) product, and the low-rank right-hand side $\mathcal{B}$ is given in HTD.

By taking $\mathcal{X}_0 = 0$ and storing it in the same tree structure as $\mathcal{B}$ (e.g., by replacing all numbers by zeros), we can start to search for $\mathcal{X}$ for example by the method of conjugate gradients (CG).

We need to know how to (i) do linear combinations, (ii) TM-product, and (iii) calculate scalar products and norms in HTD.

A sum (a linear combination) of two HTDs

Let $\mathcal{T}$ and $\mathcal{F}$ be of the same order k, of the same dimensions, and with HTDs of the same structure:

$$\mathcal{T} = (U'_{1,\mathcal{T}}, U'_{2,\mathcal{T}}, \ldots, U'_{k,\mathcal{T}} \,|\, \mathcal{C}_{\mathcal{T}}) =$$ 

.

In the top, there is one root matrix $\Sigma'_{\mathcal{T}}$, in the middle, there is bunch of inner cubes (3-way tensors) $\mathcal{R}_{\alpha,\tau,\beta,\mathcal{T}}$, and in the bottom k leaves matrices $U'_{j,\mathcal{T}}$.

Recall that

$$(\mathcal{R}_{\alpha,\tau,\beta,\mathcal{T}}{}^{\{1,2\}})^{\mathsf{T}} \mathcal{R}^{\{1,2\}}_{\alpha,\tau,\beta,\mathcal{T}} = I = I_{\mathrm{rank}_{\{\alpha+1,\ldots,\beta\}}(\mathcal{T})} \quad \text{for all } \alpha < \tau < \beta,$$

$$U'_{j,\mathcal{T}}{}^{\mathsf{T}} U'_{j,\mathcal{T}} = I = I_{\mathrm{rank}_{\{j\}}(\mathcal{T})} \qquad \text{for } j = 1, 2, \ldots, k.$$

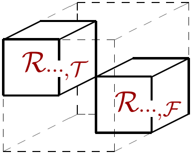
A sum (a linear combination) of two HTDs

A linear combination

$$\mathcal{E} = \varphi \mathcal{T} + \psi \mathcal{F}$$

will be done in several steps: **Step 1:** Concatenation of leaves, block diagonal composition (direct sum) of inner cubes and roots:

$$\left[U'_{j,\mathcal{T}} \, , \, U'_{j,\mathcal{F}} \right],$$



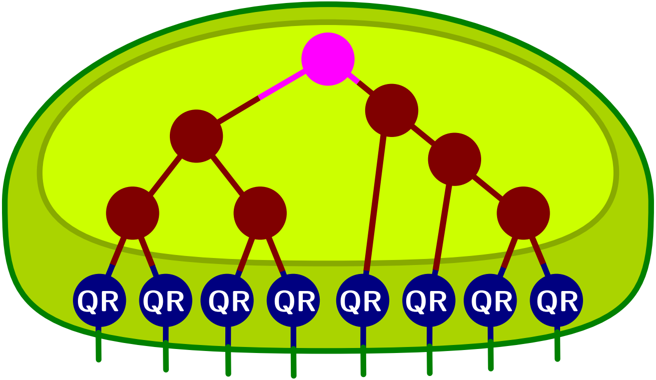
$$, \quad \begin{bmatrix} \varphi \Sigma'_{\mathcal{T}} & 0 \\ 0 & \psi \Sigma'_{\mathcal{F}} \end{bmatrix},$$

gives the sum $\mathcal{E}$ formally in the same HTD structure. However, dimensions of all objects are twice as large and $U'_{\ldots}$'s and $\mathcal{R}^{\{1,2\}}_{\ldots}$'s do not have orthonormal columns.

Step 2: (Re)compression of the sum enforing wanted properties requires plenty of QR's, TM-prod's and one SVD.

A sum (a linear combination) of two HTDs

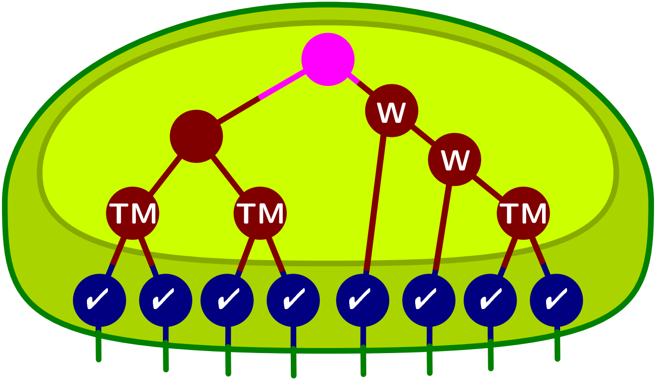
Recompression



e-QR decomp's of leaves matrices; triangular factors go up to cubes

A sum (a linear combination) of two HTDs

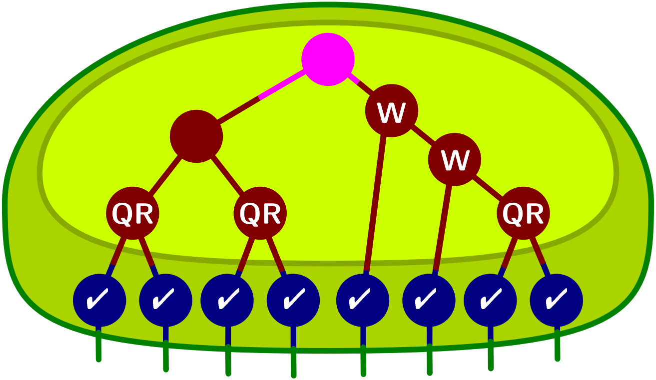
Recompression



Multiplication of cubes by triangular factors (two are waiting)

A sum (a linear combination) of two HTDs

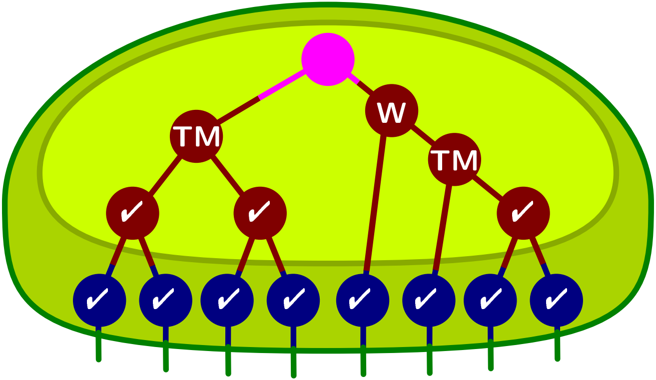
Recompression



$\{1,2\}$-ma'tions & e-QR decomp's of cubes; triangular factors go up

A sum (a linear combination) of two HTDs

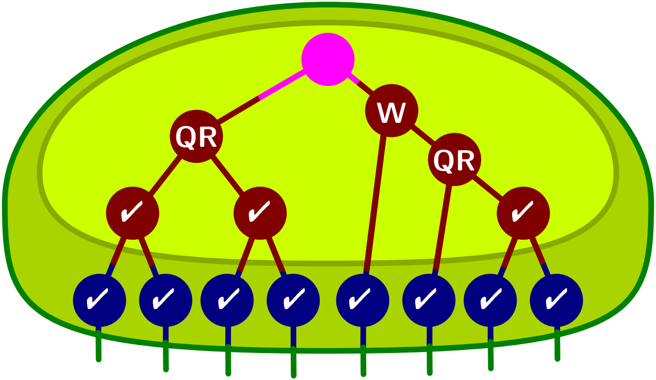
Recompression



Multiplication of cubes by triangular factors (one is waiting)

A sum (a linear combination) of two HTDs

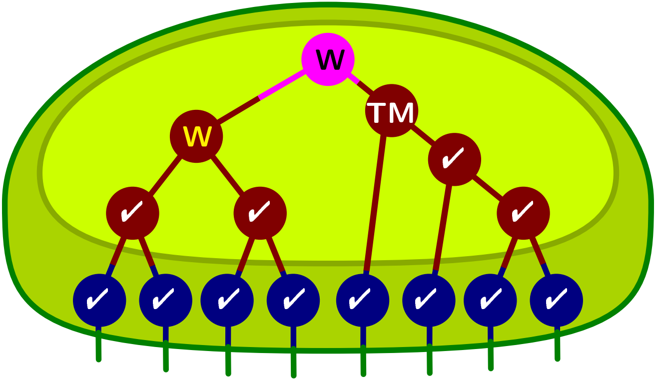
Recompression



$\{1, 2\}$-ma'tions & e-QR decomp's of cubes; triangular factors go up

A sum (a linear combination) of two HTDs

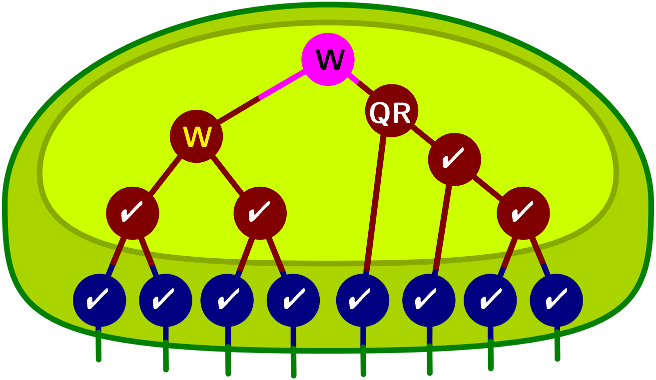
Recompression



Multiplication the last cube by triangular factors (root is waiting)

A sum (a linear combination) of two HTDs

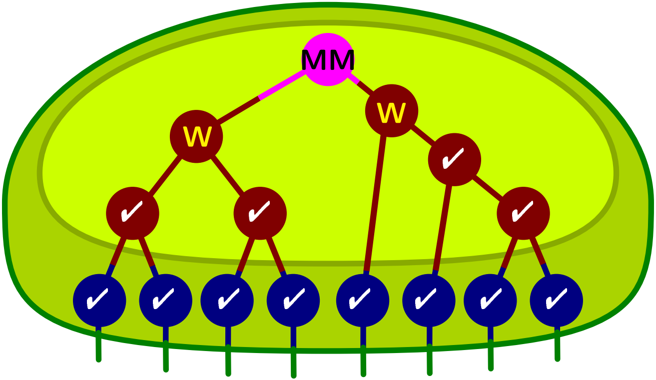
Recompression



$\{1,2\}$-ma'tions & e-QR decomp's of the last cube

A sum (a linear combination) of two HTDs

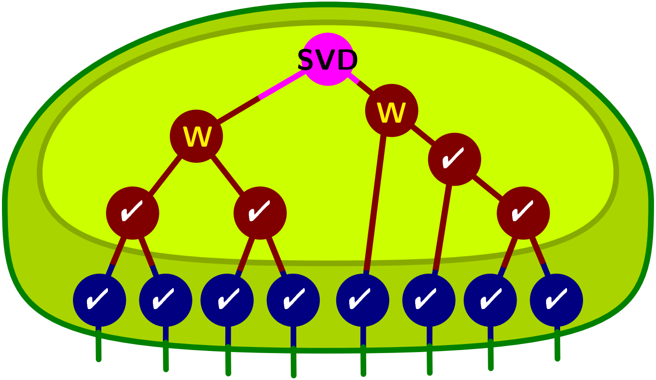
Recompression



Multiplication the root by triangular factors

A sum (a linear combination) of two HTDs

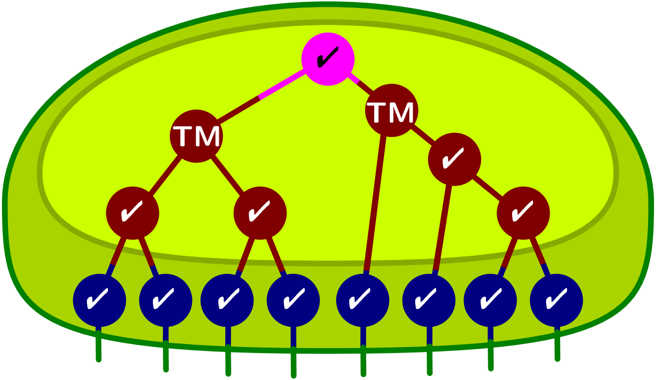
Recompression



e-SVD of the root; we've the root $\Sigma'_{\mathcal{E}}$; U' and V' are going down

A sum (a linear combination) of two HTDs

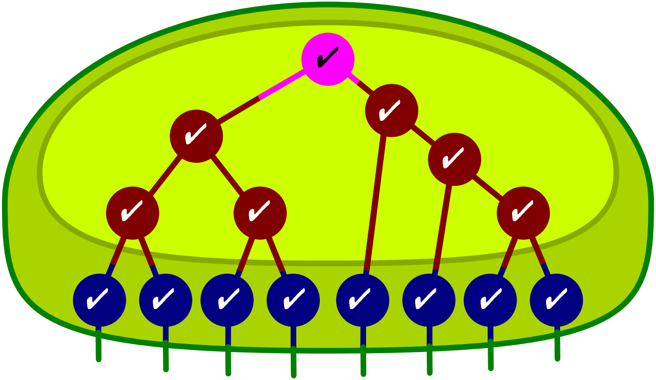
Recompression



The last two multiplications of cubes.

A sum (a linear combination) of two HTDs

Recompression



Done! ✔ ✔ ✔

Tensor-matrix multiplication

Similarly we can do the ℓ-mode tensor-matrix multiplication,

$$\mathcal{E} = M \times_\ell \mathcal{T}.$$

It will be done again in sevaral steps: **Step 1:** Multiplication of M with the particular (the ℓth) leaf:

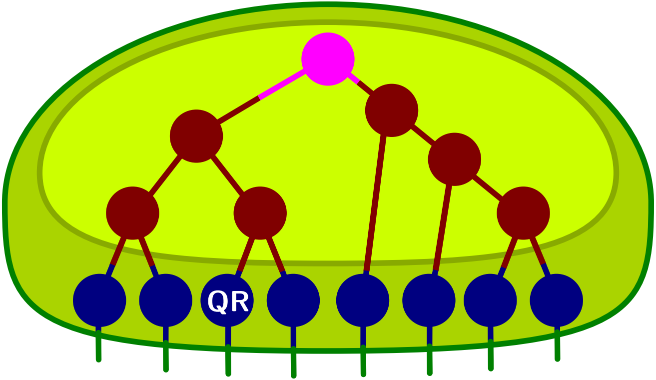
$$\left[M U'_{\ell,\mathcal{T}} \right]$$

that gives the product $\mathcal{E}$ formally in the HTD structure. Similarly as before we can do the:

Step 2: (Re)compression of the product $\mathcal{E}$. Since we multiplied only in one mode, everything is a bit simpler.

Tensor-matrix multiplication (3-mode)

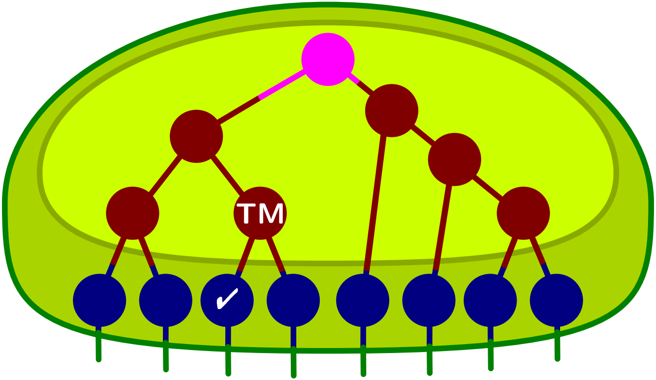
Recompression



e-QR decomp. of the third leaf; triangular factor goes up to cubes

Tensor-matrix multiplication (3-mode)

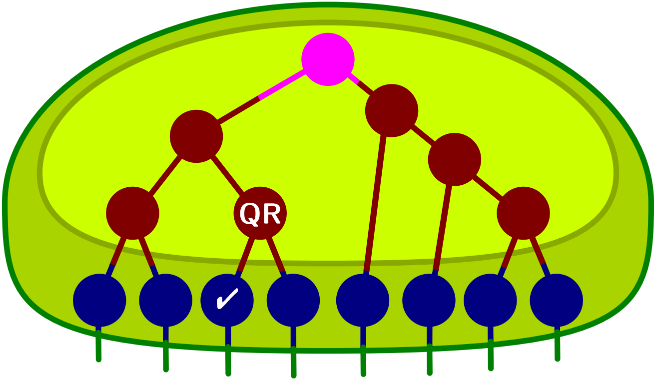
Recompression



Multiplication of cubes by triangular factors (two are waiting)

Tensor-matrix multiplication (3-mode)

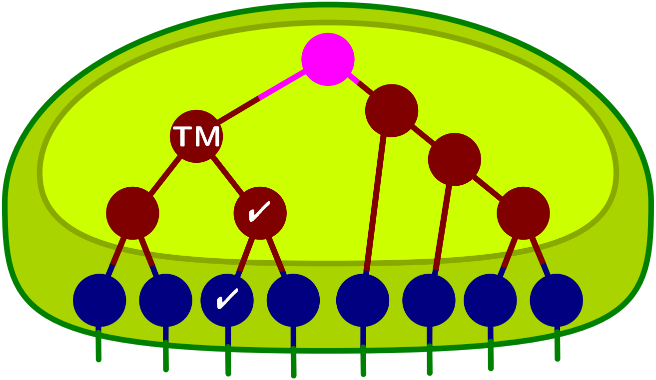
Recompression



$\{1, 2\}$-ma'tions & e-QR decomp's of cubes; triangular factors go up

Tensor-matrix multiplication (3-mode)

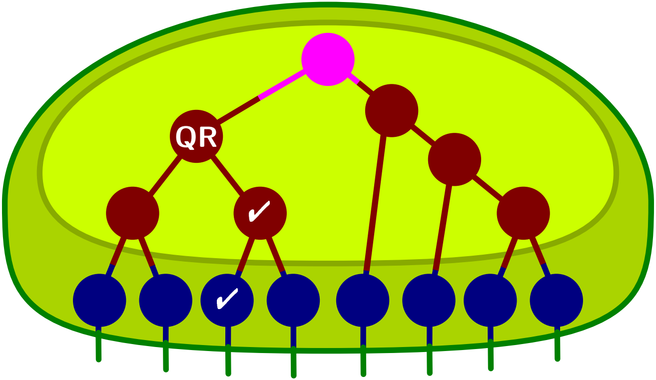
Recompression



Multiplication of cubes by triangular factors (one is waiting)

Tensor-matrix multiplication (3-mode)

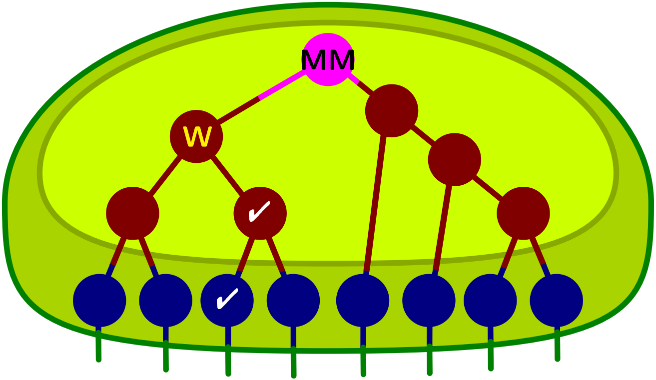
Recompression



$\{1, 2\}$-ma'tions & e-QR decomp's of cubes; triangular factors go up

Tensor-matrix multiplication (3-mode)

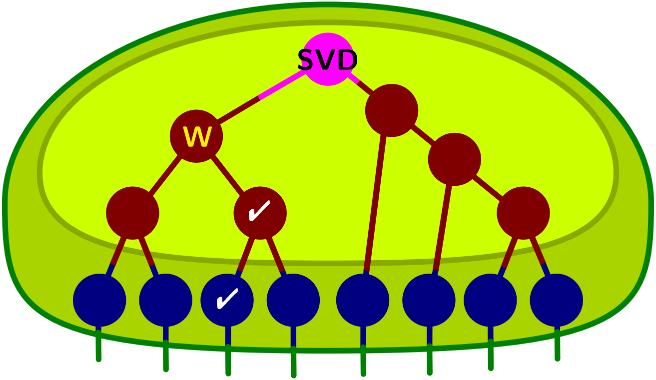
Recompression



Multiplication the root by triangular factors

Tensor-matrix multiplication (3-mode)

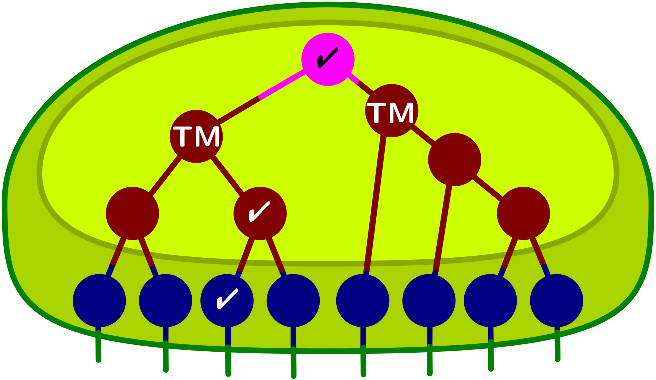
Recompression



e-SVD of the root; we've the root $\Sigma'_{\mathcal{E}}$; U' and V' are going down

Tensor-matrix multiplication (3-mode)

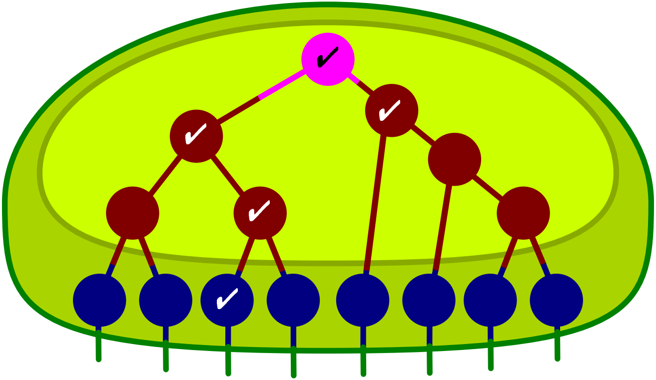
Recompression



The last two multiplications of cubes.

Tensor-matrix multiplication (3-mode)

Recompression



Done! ✔ ✔ ✔

Scalar product of two tensors in HTD

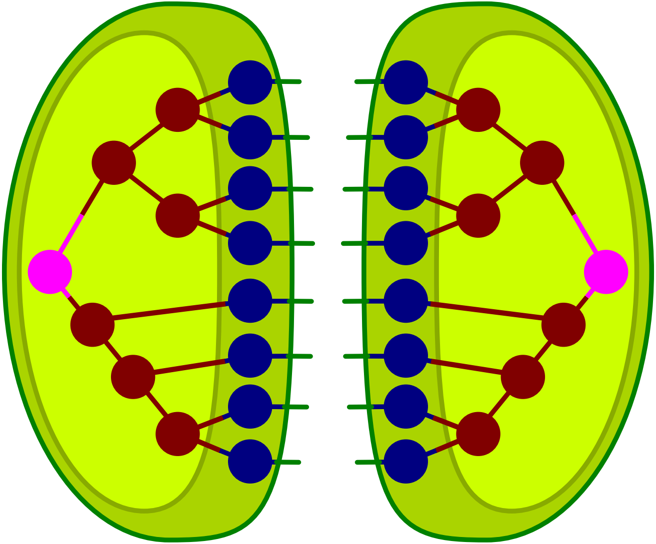
Finally, we present evaluation of the scallar product

$$\langle \mathcal{T}, \mathcal{F} \rangle$$

of two vectors in HTD with the same trees; and also of the norm

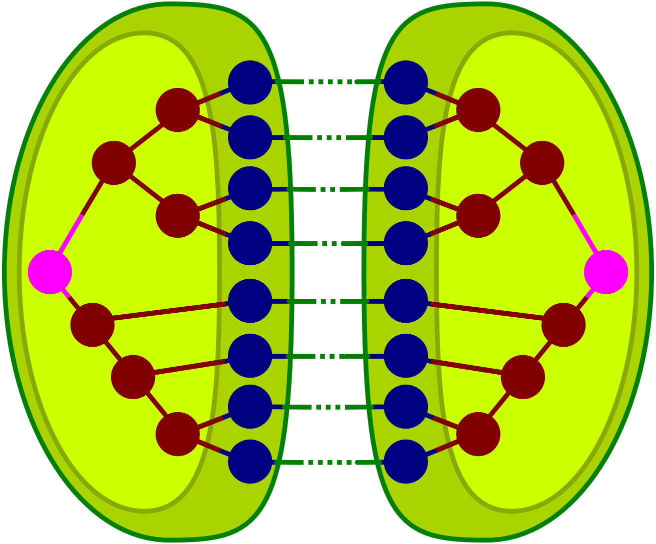
$$\|\mathcal{T}\| = (\langle \mathcal{T}, \mathcal{T} \rangle)^{\frac{1}{2}}.$$

Scalar product of two tensors in HTD



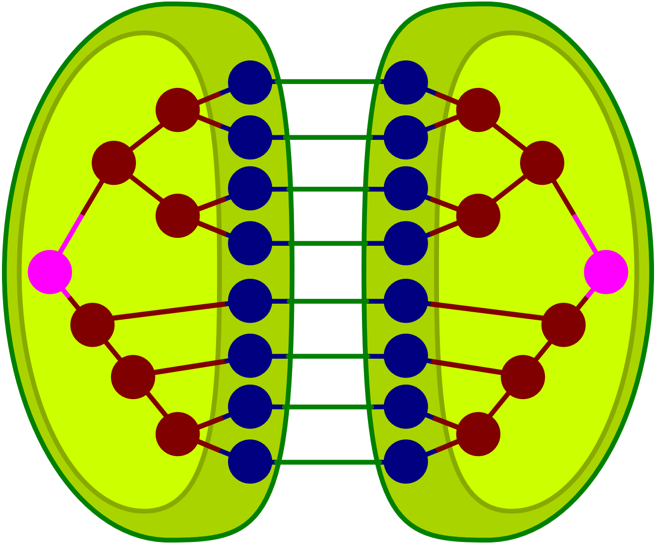
Two tensors with the same tree

Scalar product of two tensors in HTD



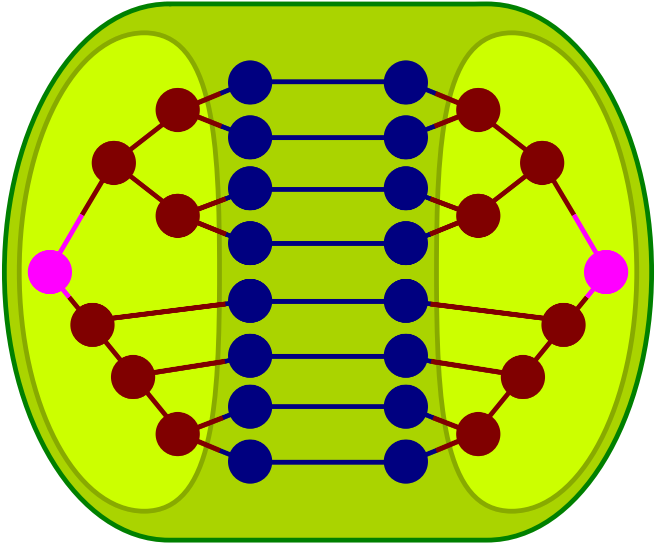
Two tensors with the same tree and their scalar product

Scalar product of two tensors in HTD



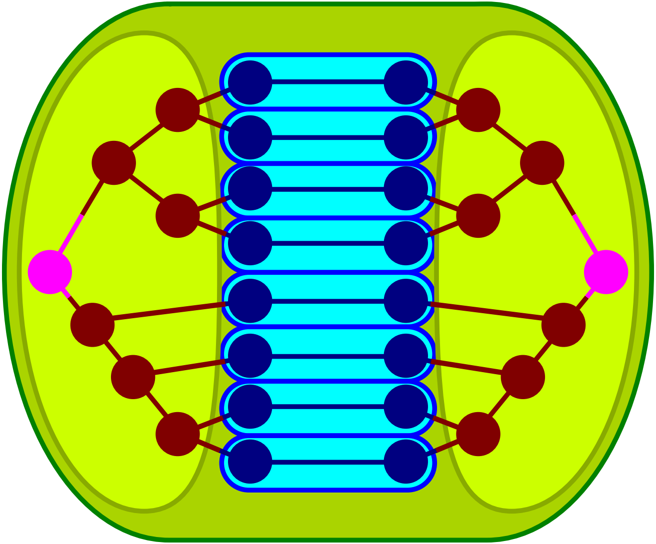
Two tensors with the same tree and their scalar product

Scalar product of two tensors in HTD



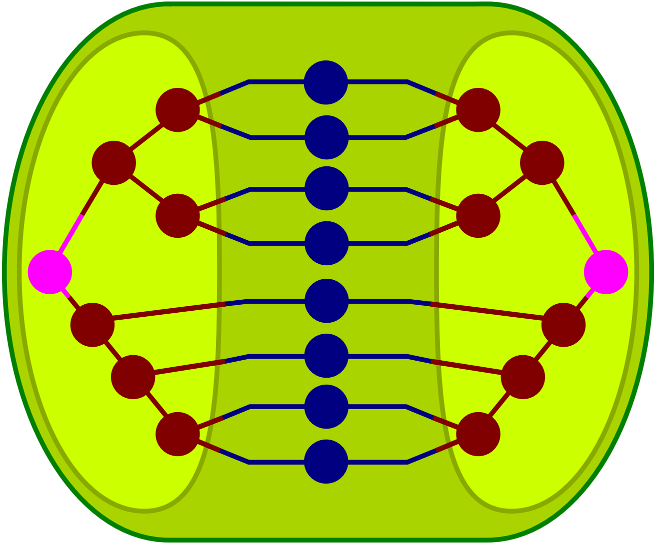
Two tensors with the same tree and their scalar product

Scalar product of two tensors in HTD



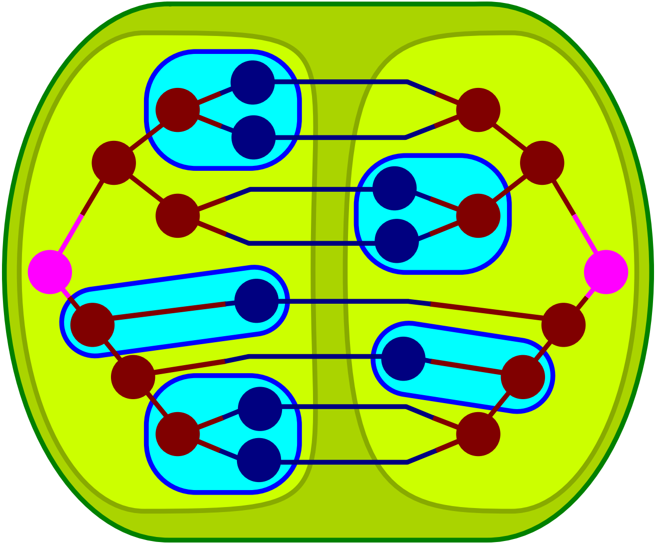
Evaluation starts with bunch of MM-products of leaves

Scalar product of two tensors in HTD



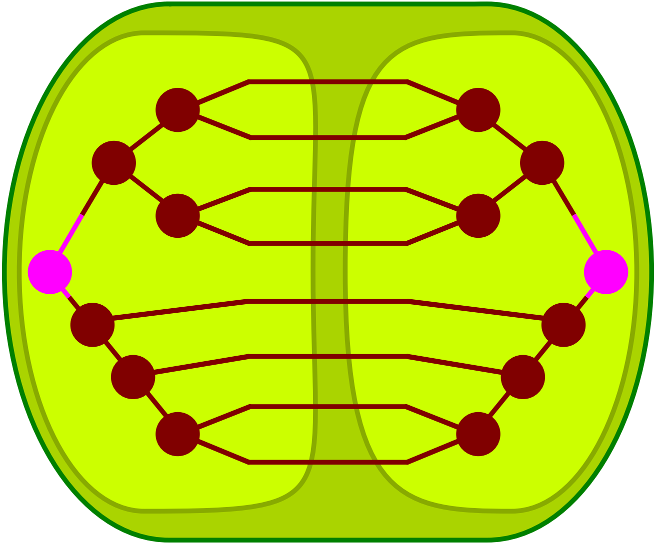
MM-products result in matrices

Scalar product of two tensors in HTD



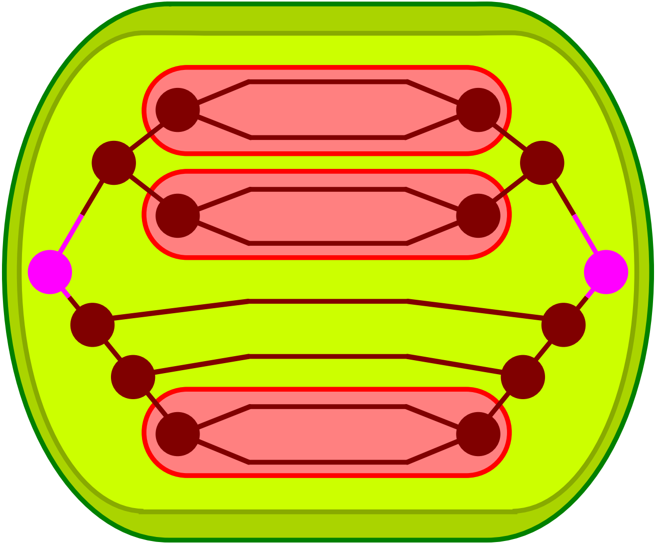
Then comes bunch of TM-prod's; we choose *smaller resulting dim's*

Scalar product of two tensors in HTD



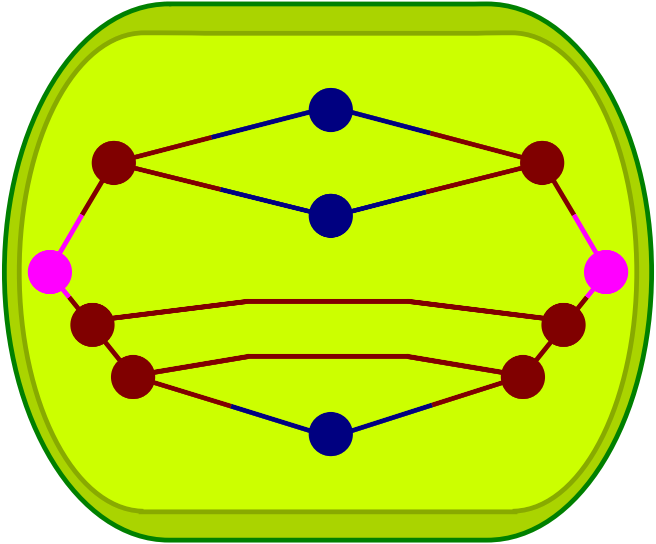
TM-products result in tensors

Scalar product of two tensors in HTD



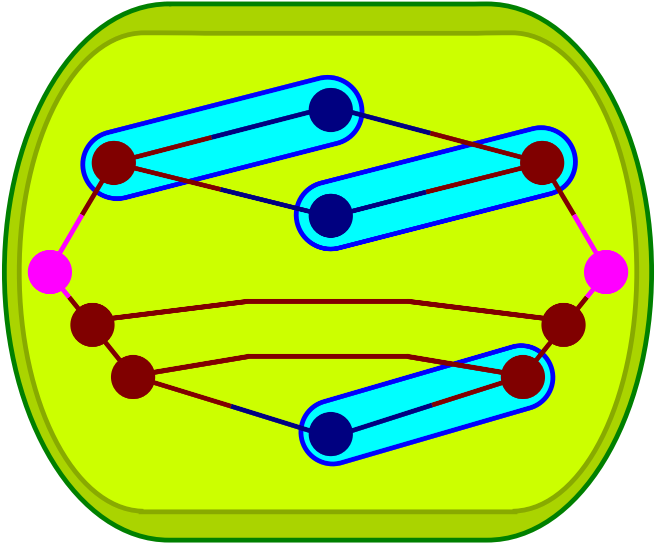
We continue with bunch of two-mode TT-products

Scalar product of two tensors in HTD



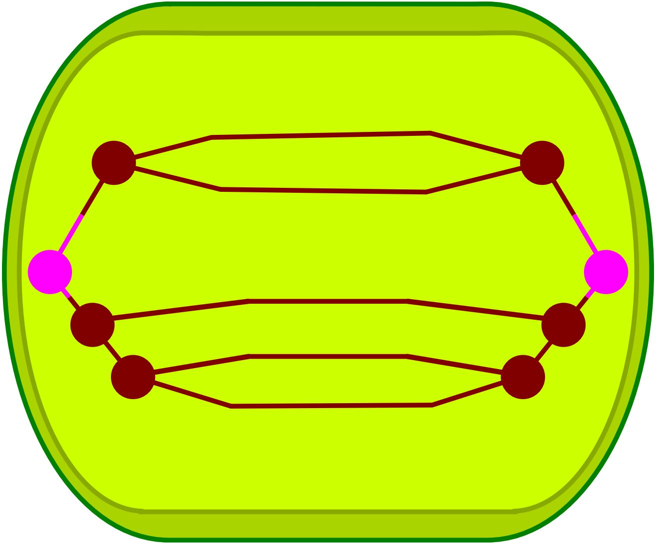
Two-mode TT-products of cubes result in matrices

Scalar product of two tensors in HTD



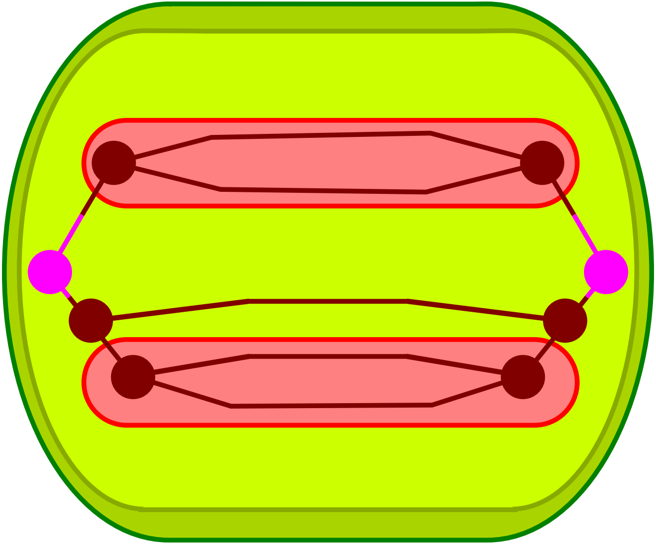
We continue with bunch of TM-products; we can choose *faster way*

Scalar product of two tensors in HTD



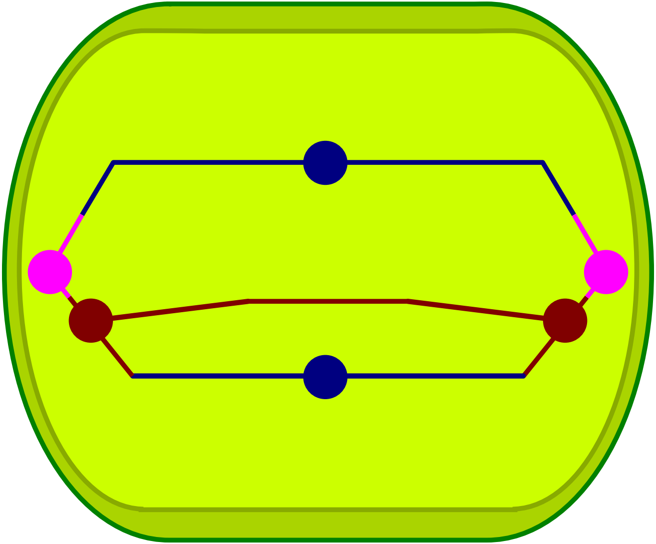
TM-products result in tensors

Scalar product of two tensors in HTD



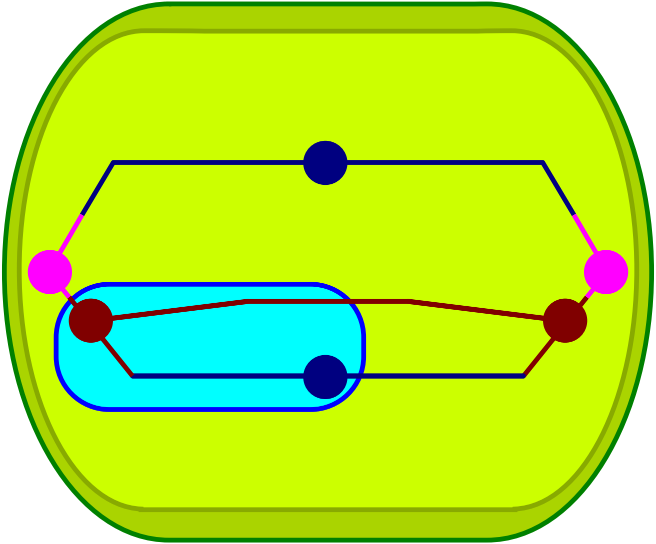
We continue with bunch of two-mode TT-products

Scalar product of two tensors in HTD



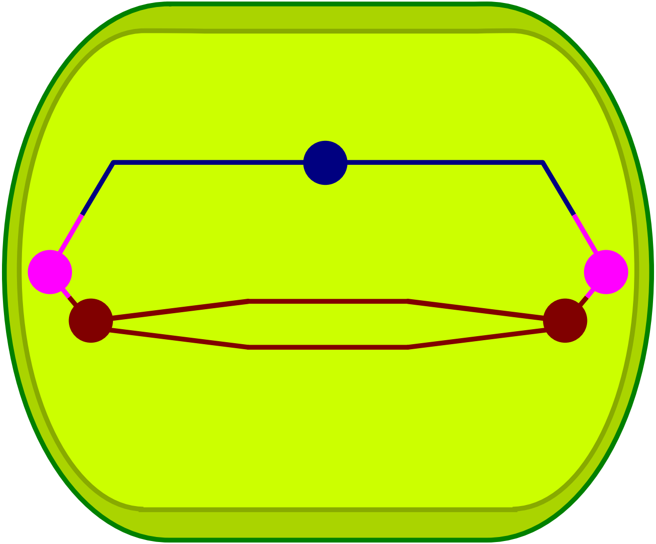
Two-mode TT-products of cubes result in matrices

Scalar product of two tensors in HTD



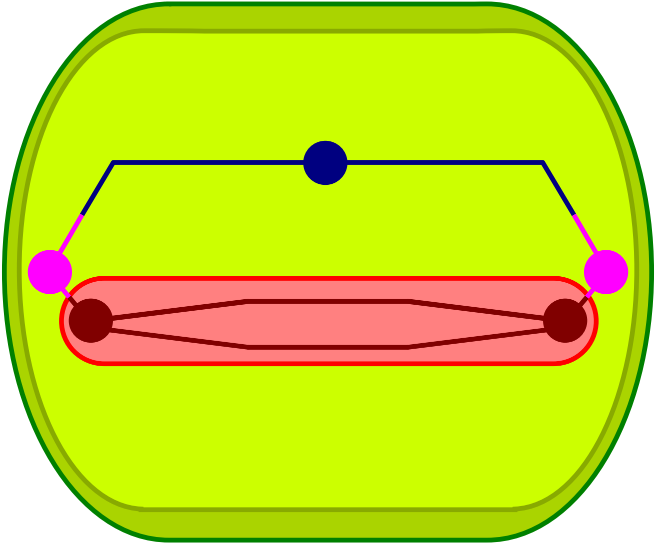
The last TM-product

Scalar product of two tensors in HTD



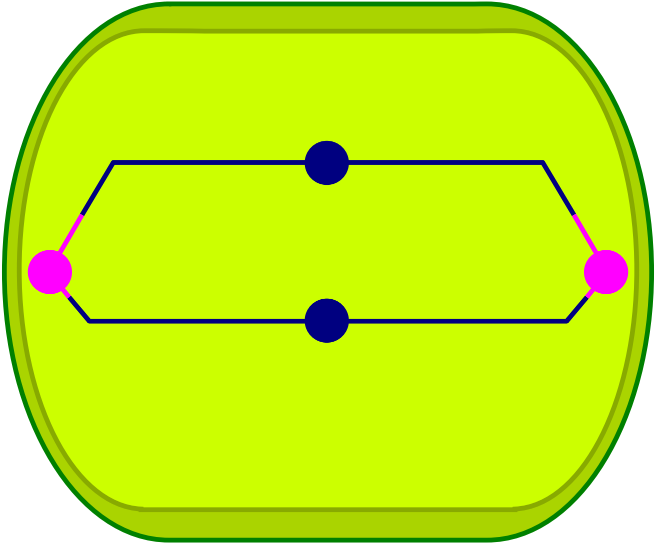
The last TM-product results in tensor as well

Scalar product of two tensors in HTD



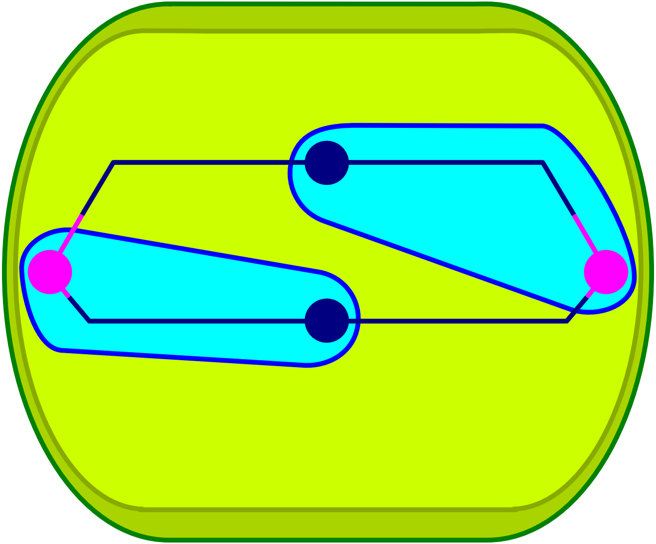
The last two-mode TT-product

Scalar product of two tensors in HTD



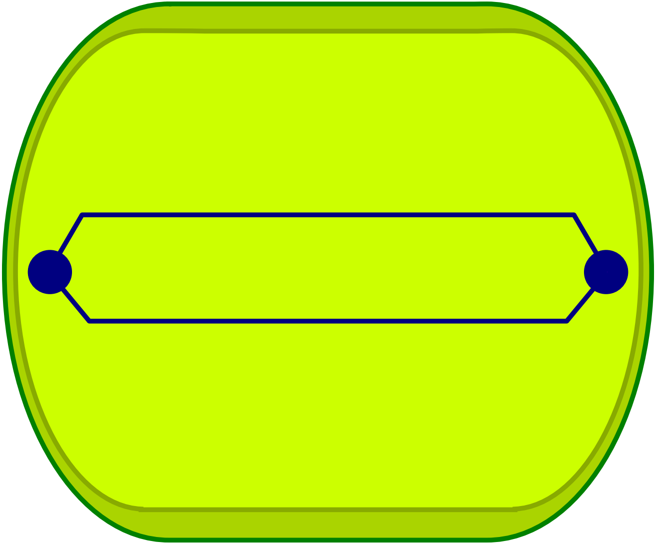
The last two-mode TT-products of cubes results in matrix as well

Scalar product of two tensors in HTD



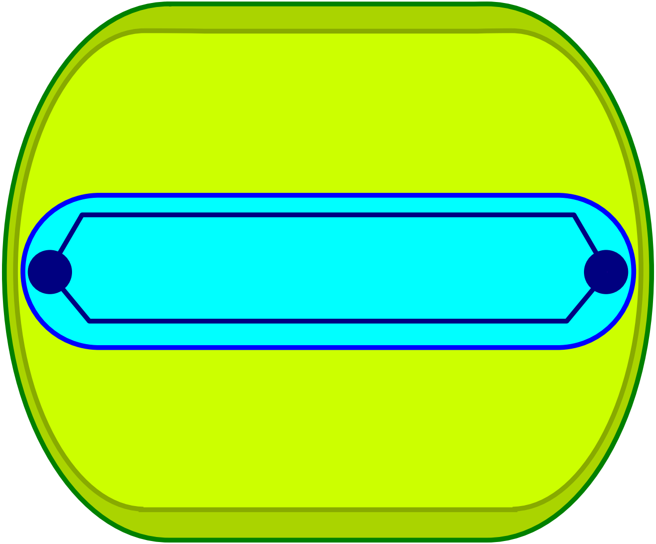
The circular prod. of four matrices! We start with two MM prod's

Scalar product of two tensors in HTD



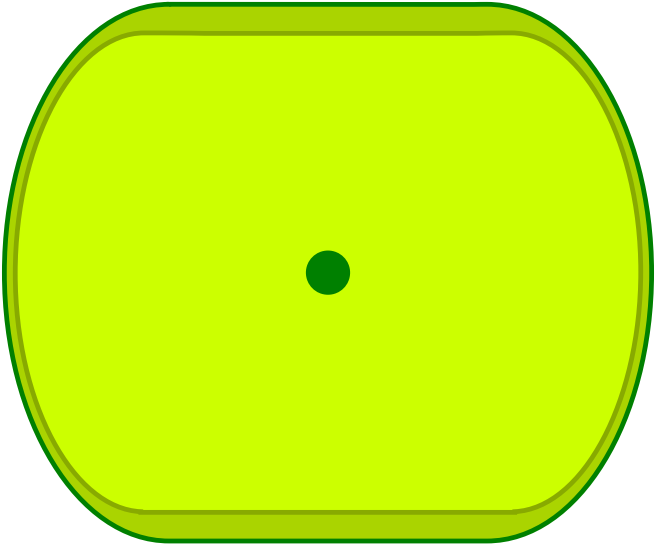
Thus we end up with two matrices

Scalar product of two tensors in HTD



We calculate their scalar product

Scalar product of two tensors in HTD



Done! ✔ ✔ ✔

Final notes on arithmetics of HTDs

For a linear combination and scallar product of two tensors

$$\varphi\mathcal{T} + \psi\mathcal{F}, \quad \langle\mathcal{T}, \mathcal{F}\rangle,$$

$\mathcal{T}$, $\mathcal{F}$ need to be of the same dimensions (and thus also the order).

It seems that requirement on the same tree-structure brings a new restriction, but it is possible do that also with tensors with different tree-structures.

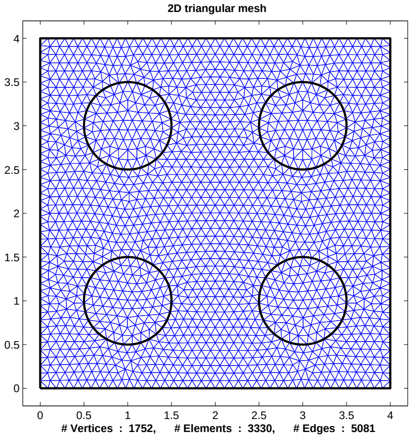
However, while doing that with tensors with different *binary* trees, there always appear tensors of higher orders than presented. *Typically* (i.e., if the *root is not* in the game), there appear at least one inner 'cube' of order *four* (no hihger orders are needed$^{(?!)}$).

While summation, it can employ some maximal (or the greates$^{(?!)}$) common sub-tree of both and recalculate the structure of one.

[*Kressner, Tobler*, htucker—Matlab toolbox, 2012]
http://anchp.epfl.ch/htucker

A (simple) example of practical application

Heat conductivity problem



Poisson (steady-state heat) equation:

$$\begin{aligned} -\nabla(\sigma(\xi)\nabla u) &= f && \text{in } \Omega \\ u &= u_\Gamma && \text{on } \Gamma = \partial\Omega \end{aligned}$$

and $\sigma(\xi)$ with piecewise constant heat conductivity

$$\sigma(\xi) = \begin{cases} 1+\theta_\ell & \text{for } \xi \in \mathrm{Disc}_\ell \\ 1 & \text{for } \xi \notin \mathrm{Disc}_\ell \end{cases} ;$$

f denotes the heat-flux density of sources.

FEM discretization with piecewise linear elments then gives us two SPD matrices: The *stiffness* $A \in \mathbb{R}^{n\times n}$ and *mass* $M \in \mathbb{R}^{n\times n}$ *matrix*.

Heat conductivity problem

We are interested in controllability of the dynamical system (DS)

$$\begin{aligned} M\frac{d}{dt}\boldsymbol{u}(t) &= A\boldsymbol{u}(t) + B\boldsymbol{f}(t) \\ \boldsymbol{y}(t) &= C\boldsymbol{u}(t) + D\boldsymbol{f}(t) \end{aligned}$$

where

- $M \in \mathbb{R}^{m\times m}$ and $A \in \mathbb{R}^{m\times m}$ are the (SPD) mass and stiffness matrices; $\boldsymbol{u}(t) \in \mathbb{R}^m \times \boxed{\text{TIME}}$ denotes the inner state of DS;
- $B \in \mathbb{R}^{m\times p}$ localizes the p ($p \ll m$) inputs of the control signal $\boldsymbol{f}(t) \in \mathbb{R}^d \times \boxed{\text{TIME}}$;
- and the rest defines the output signal $\boldsymbol{y}(t) \in \mathbb{R}^q \times \boxed{\text{TIME}}$.

The so-called controllability Gramian then solves the generalized Lyapunov equation (LE)

$$AXM^\mathsf{T} + MXA^\mathsf{T} = -BB^\mathsf{T}.$$

Heat conductivity problem

Since M is SPD, $M = LL^{\mathsf{T}}$ (Cholesky fact.), the generalized LE

$$AXM^{\mathsf{T}} + MXA^{\mathsf{T}} = -BB^{\mathsf{T}}$$

is congruent to a standard LE

$$L^{-1} \cdot \left\backslash \quad AXLL^{\mathsf{T}} + LL^{\mathsf{T}}XA^{\mathsf{T}} = -BB^{\mathsf{T}} \quad \right/ \cdot L^{-\mathsf{T}}$$

$$L^{-1}\underbrace{A\,X}_{I=L^{-\mathsf{T}}L^{\mathsf{T}}}L + L^{\mathsf{T}}\underbrace{X\,A^{\mathsf{T}}}_{I=LL^{-1}}L^{-\mathsf{T}} = -L^{-1}BB^{\mathsf{T}}L^{-\mathsf{T}}$$

$$\left(L^{-1}AL^{-\mathsf{T}}\right)\left(L^{\mathsf{T}}XL\right) + \left(L^{\mathsf{T}}XL\right)\left(L^{-1}AL^{-\mathsf{T}}\right)^{\mathsf{T}} = -\left(L^{-1}B\right)\left(L^{-1}B\right)^{\mathsf{T}}$$

$$\widetilde{A}\widetilde{X} + \widetilde{X}\widetilde{A}^{\mathsf{T}} = -\widetilde{B}\widetilde{B}^{\mathsf{T}}$$

with SPD $\widetilde{A}$. Note that for $B \in \mathbb{R}^{m\times p}$, $p = 1$, singular values of

$$\widetilde{X} = \widetilde{X}^{\mathsf{T}} = \int_{\boxed{\text{TIME}}} \left(e^{\widetilde{A}t}\widetilde{B}\right)\left(e^{\widetilde{A}t}\,\widetilde{B}\right)^{\mathsf{T}} dt$$

decay exponentially; it is well approximable by a low-rank matrix.

Heat conductivity problem

We look for the (symmetric) low-rank matrix solution $X \in \mathbb{R}^{m\times m}$ of

$$AXM^\mathsf{T} + MXA^\mathsf{T} = -BB^\mathsf{T}.$$

But recall that

$$\sigma(\xi) = \begin{cases} 1+\theta_\ell & \text{for } \xi \in \mathrm{Disc}_\ell \\ 1 & \text{for } \xi \notin \mathrm{Disc}_\ell \end{cases},$$

thus in this case

$$A = A(\theta_1,\theta_2,\theta_3,\theta_4) = A_0 + \textstyle\sum_{\ell=1}^{4} \theta_\ell A_\ell \qquad \text{with} \qquad \theta_\ell \in \mathbb{R}^+$$

after discretization $\theta_\ell \in \{\theta_{\ell,1}, \theta_{\ell,2}, \ldots, \theta_{\ell,d_\ell}\}$. Then also

$$X = X(\theta_1,\theta_2,\theta_3,\theta_4) = \mathcal{X} \in \mathbb{R}^{m\times m\times d_1\times d_2\times d_3\times d_4}$$

and we look for a 6-way tensor, symmetric in the first two modes.

Heat conductivity problem

Since the opeator is SPD, we use the CG method.

Questions:

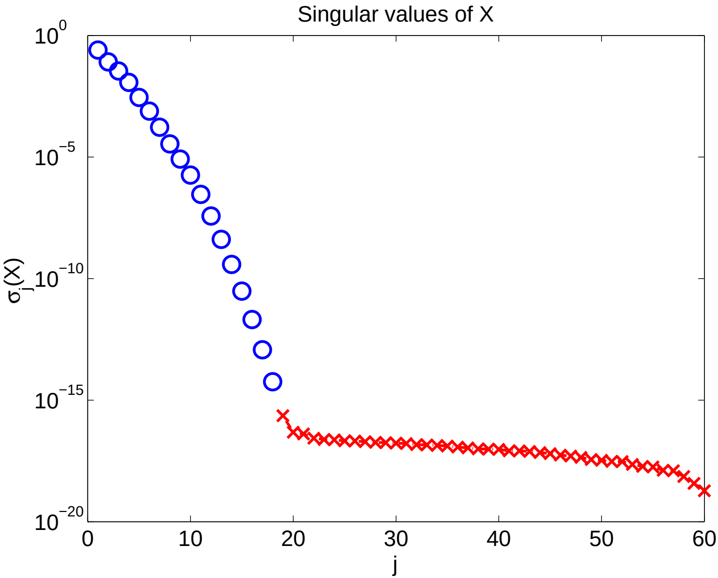
- Is $\mathcal{X} \in \mathbb{R}^{m \times m \times d_1 \times d_2 \times d_3 \times d_4}$ approximable by a low-rank tensor?
- We do not have any ranks. How to define numerical ranks of individual objects (residuals, direction vectors, ...)?
- The generalized vs. standard LE; the congruence

$$AXM^{\mathsf{T}} + MXA^{\mathsf{T}} = -BB^{\mathsf{T}} \quad \longleftrightarrow \quad \widetilde{A}\widetilde{X} + \widetilde{X}\widetilde{A}^{\mathsf{T}} = -\widetilde{B}\widetilde{B}^{\mathsf{T}}$$

  change behavior of CG.
- Preconditioner should preserve the structure of the problem.
- Usually, the goal of preconditioning is to speed-up the convergence in terms of iterations. Here, the cost of iteration strongly depends on ranks (dimensions of the Tucker core and its inner cubes). But preconditioner involves these dimension. But how?

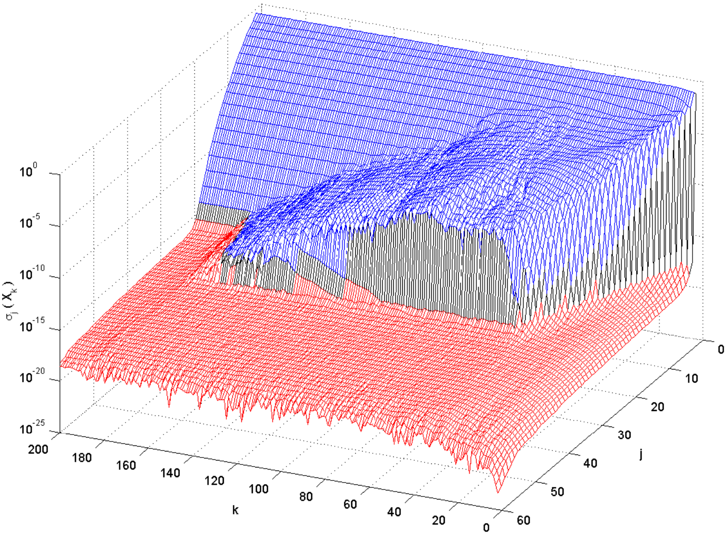
Heat conductivity problem—No parameters

Singular values of $X \in \mathbb{R}^{m \times m}$



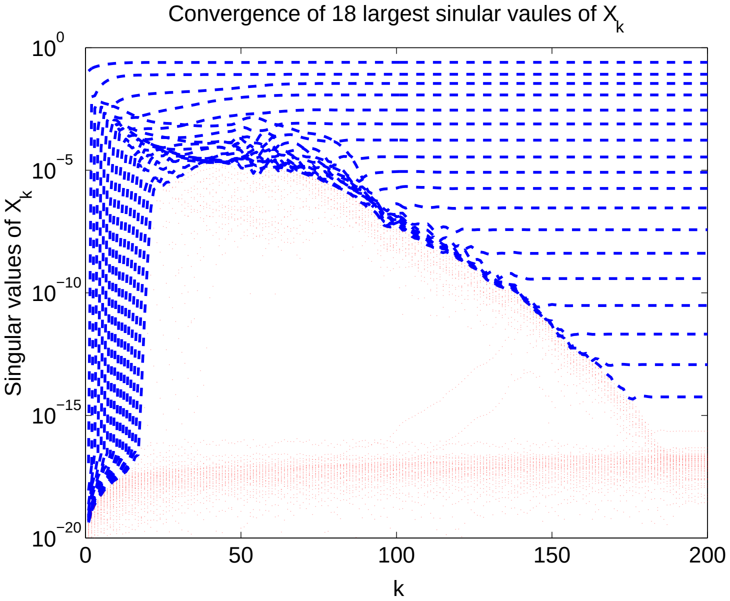
Heat conductivity problem—No parameters

Singular values of CG approximations X_k



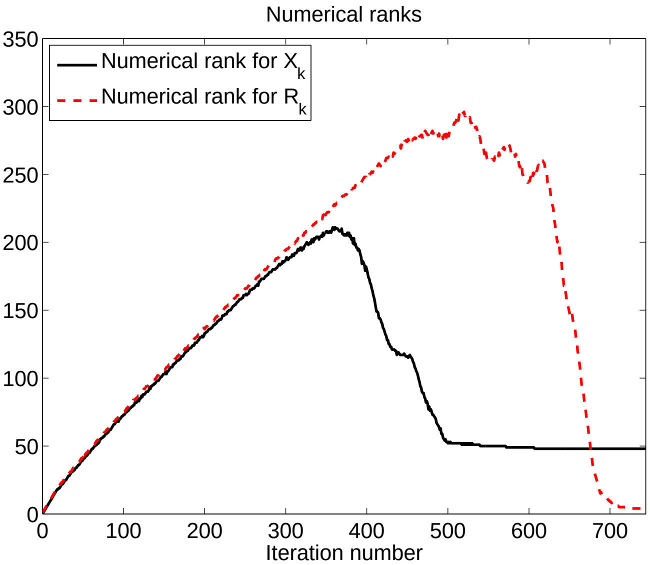
Heat conductivity problem—No parameters

Convergence of sing'vals of CG approximations X_k



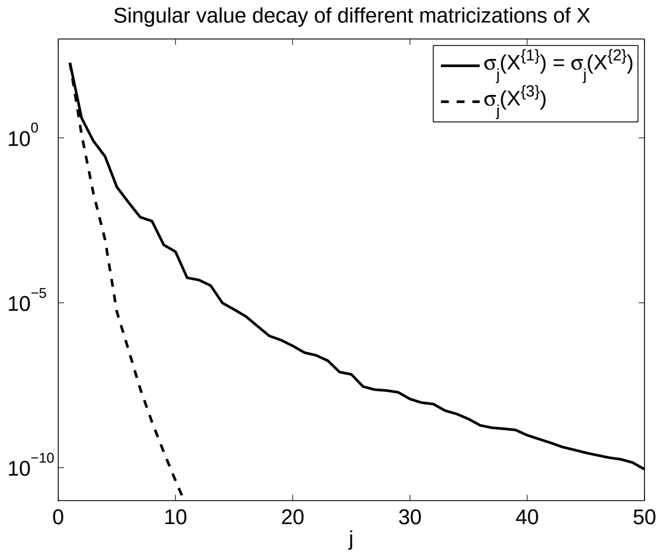
Heat conductivity problem—No parameters

Ranks of CG approximations X_k and residuals R_k



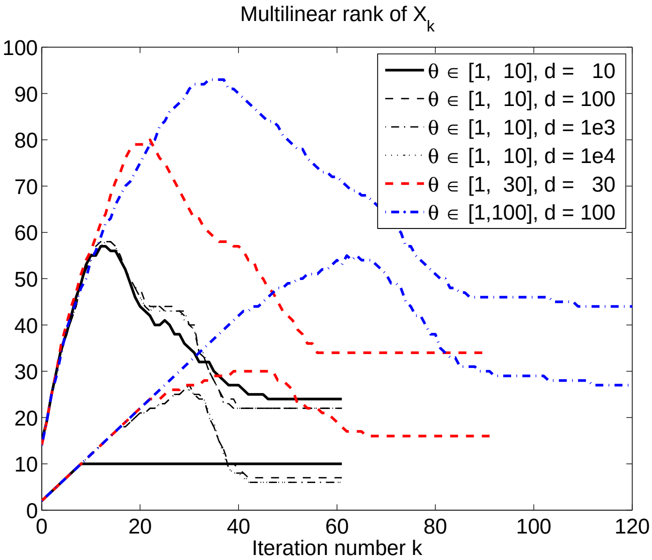
Heat conductivity problem—One parameter

Singular values of $\mathcal{X} \in \mathbb{R}^{m \times m \times d}$



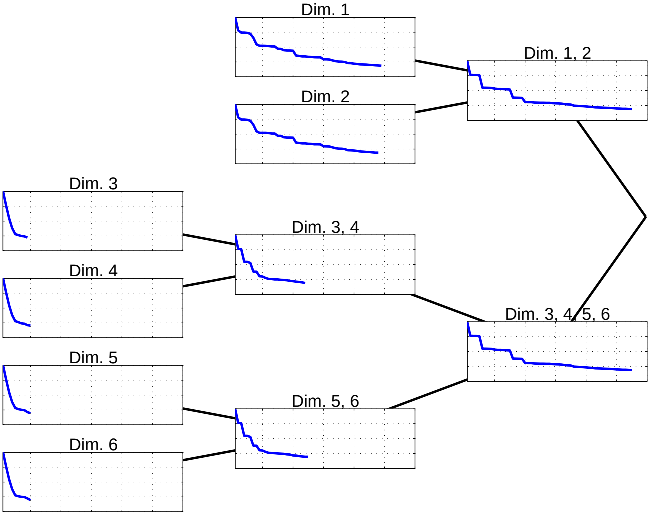
Heat conductivity problem—One parameter

Ranks of CG approximations $\mathcal{X}_k$



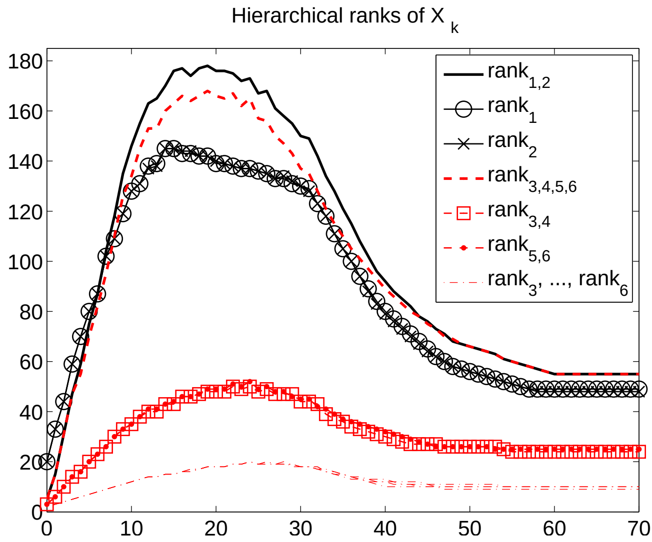
Heat conductivity problem—All four parameters

The tree structure and singular values of $\mathcal{X} \in \mathbb{R}^{m \times m \times d_1 \times d_2 \times d_3 \times d_4}$



Heat conductivity problem—All four parameters

Ranks of CG approximations $\mathcal{X}_k$



References

Selected original works:

- *F. L. Hitchcock*, The expression of a tensor or a polyadic as a sum of products, Journal of Mathematics and Physics 6(1) (1927), pp. 164–189.
- *F. L. Hitchcock*, Multilple invariants and generalized rank of a *p*-way matrix or tensor, Journal of Math. and Physics 7(1) (1927), pp. 39–79.
- *L. R. Tucker*, Implications of factor analysis of three-way matrices for measurement of change, in Problems in Measuring Change, C. W. Harris, ed., University of Wisconsin Press, 1963, pp. 122–137.
- *L. R. Tucker*, The extension of factor analysis to three-dimensional matrices, in Contributions to Mathematical Psychology, H. Gulliksen and N. Frederiksen, eds., Holt, Rinehardt & Winston, NY, 1964, pp. 110–127.
- *L. R. Tucker*, Some mathematical notes on three-mode factor analysis, Psychometrika, 31 (1966), pp. 279–311.
- *L. Grasedyck*, Hierarchical singular value decomposition of tensors, SIAM Journal on Matrix Analysis and Appl. 31(4) (2010), pp. 2029–2054.
- *I. V. Oseledets*, Tensor-train decomposition, SIAM Journal on Scientific Computing, 33 (2011), pp. 2295–2317.

References

Survey papers & lectures:

• *T. G. Kolda, B. W. Bader*, Tensor decompositions and applications, SIAM Review 51(3) (2009), pp. 455–500.

• *L. Grasedyck, D. Kressner, C. Tobler*, A literature survey of low-rank tensor approximation techniques, GAMM Mitt. 36(1) (2013), pp. 53–78.

• *B. N. Khoromskij*, Introduction to tensor numerical methods in scientific computing (ETH Zürich, slides), 2010.

• *E. Tyrtyshnikov*, Num. meth. with tensor represent. of data (Summers school slides), 2012. On-line: `http://academy2012.hpc-russia.ru/files/lectures/algebra/0703_1_tyrtyshnikov.pdf`.

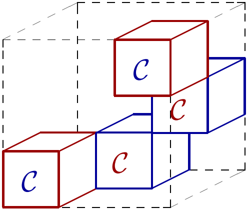
Toolboxes:

• *B. W. Bader, T. G. Kolda*, MATLAB Tensor Toolbox, version 2.6., 2015. On-line: `http://www.sandia.gov/~tgkolda/TensorToolbox`.

• *D. Kressner, C. Tobler*, `htucker`—A MATLAB toolbox for tensors in hierarchical Tucker format, TR 2012-02, SAM ETH Zurich, 2012. On-line: `http://anchp.epfl.ch/htucker`.

• *L. Sorber, M. Van Barel, L. De Lathauwer*, Tensorlab 3.0, 2016. On-line: `http://www.tensorlab.net`.

That's All Volks!



Thank You for Your Attention