15 Lectures on Tensor Numerical Methods for Multi-dimensional PDEs

Lect. 6. Tucker tensor decomposition algorithm. Canonical-to-Tucker decomposition.

Tensor operations

Boris N. Khoromskij and Venera Khoromskaia
Institute of Natural Sciences, Shanghai Jiaotong University
April 2017

Max-Planck-Institute for Mathematics in the Sciences, Leipzig, Germany

Outline of the lecture

1. Why Tucker was important
   - Tucker tensor format
   - Higher order singular value decomposition
   - Tucker decomposition algorithm
   - Tucker decomposition algorithm in Matlab
   - Functional Tucker
   - Canonical to Tucker tensor approximation
   - Multigrid Tucker decomposition
   - Multigrid Tucker for canonical tensors
   - Basic tensor operations

Canonical tensor format

Rank-structured tensor formats:

1. The R-term canonical format is defined as

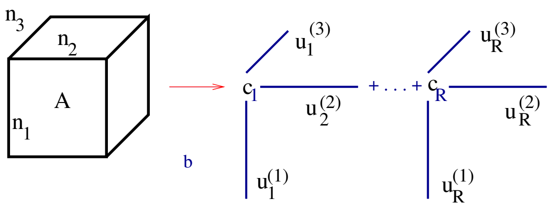
$$\mathbf{A} = \sum\nolimits_{k=1}^{R} c_k \mathbf{u}_k^{(1)} \otimes \ldots \otimes \mathbf{u}_k^{(d)}, \quad c_k \in \mathbb{R}$$

where $\mathbf{u}_k^{(\ell)} \in \mathbb{R}^{n_\ell}$ are normalized vectors. Storage: $dRn << n^d$.

Analogue: a separable function $f(x_1, x_2, \ldots, x_d) = \sum\limits_{k=1}^{R} f_{1,k}(x_1) f_{2,k}(x_2) \ldots f_{d,k}(x_2)$.

Alternative (contracted product) notation:

$\mathbf{A} = \mathbf{C} \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)}, \qquad \mathbf{C} = diag\{c_1, ..., c_d\} \in \mathbb{R}^{R^{\otimes d}}, \quad U^{(\ell)} = [\mathbf{u}_1^{(\ell)} \ldots \mathbf{u}_R^{(\ell)}] \in \mathbb{R}^{n_\ell \times R}.$



There are no stable algorithms for computation of the nearly optimal rank-R approximation of a full size tensor (it is an ill-posed problem).

Tucker tensor format

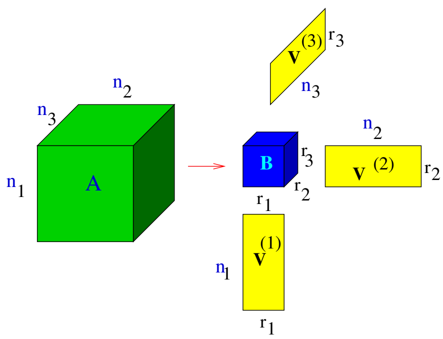
2. Given the rank parameter $\mathbf{r} = (r_1, \ldots, r_d)$ we define a tensor in the Tucker format [Tucker, 1966]

$$\mathbf{A}_{(\mathbf{r})} = \sum\nolimits_{k_1=1}^{r_1} \cdots \sum\nolimits_{k_d=1}^{r_d} \beta_{k_1,\ldots,k_d} \mathbf{v}_{k_1}^{(1)} \otimes \ldots \otimes \mathbf{v}_{k_d}^{(d)}, \tag{1}$$

with orthonormal $\mathbf{v}_{k_\ell}^{(\ell)} \in \mathbb{R}^{n_\ell}$ $(1 \le k_\ell \le r_\ell)$, and core tensor $\boldsymbol{\beta} = [\beta_{k_1,\ldots,k_d}] \in \mathbb{R}^{r_1 \times \cdots \times r_d}$.

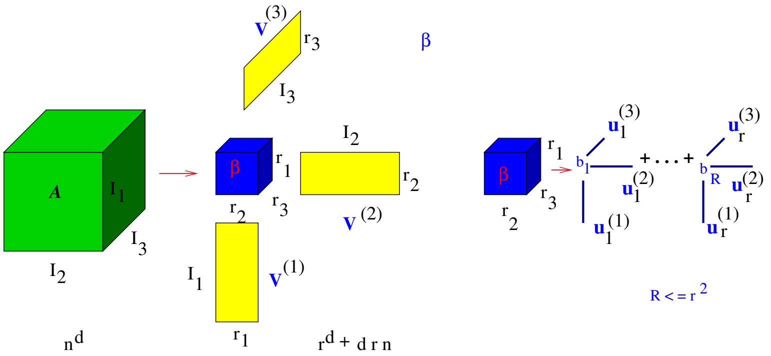
Alternative notation: using side matrices $V^{(\ell)} = [v_1^{(\ell)} \ldots v_{r_\ell}^{(\ell)}]$ and contracted products

$\mathbf{A}_{(\mathbf{r})} = \boldsymbol{\beta} \times_1 V^{(1)} \times_2 V^{(2)} \times_3 \ldots \times_d V^{(d)}.$



Storage: $r^d + drn \ll n^d$.

$$A_{(\mathbf{r})} = \left(\sum_{k=1}^{R} b_k u_k^{(1)} \otimes \ldots \otimes u_k^{(d)} \right) \times_1 V^{(1)} \times_2 V^{(2)} \times_3 \ldots \times_d V^{(d)}.$$



Advantages of the Tucker tensor decomposition:

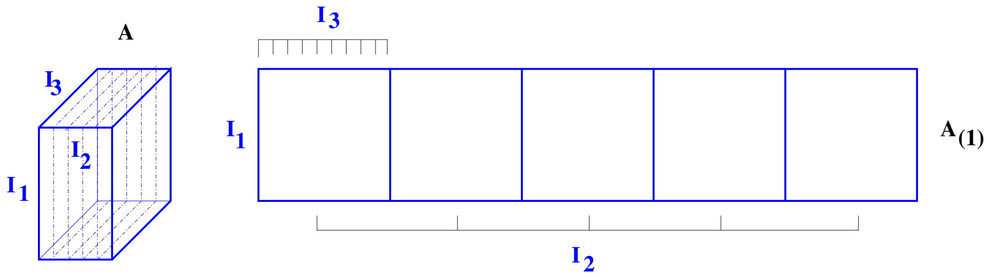
1. Robust algorithm for approximating full format tensors of size n^d.
2. Rank reduction of the rank-R canonical tensors with large R.
3. Efficient for 3D tensors since r^3 is small.

Tensor unfolding

The *unfolding of a tensor* along mode ℓ is a matrix

$$A_{(\ell)} = [a_{ij}] \in \mathbb{R}^{n_\ell \times (n_{\ell+1} \cdots n_d n_1 \cdots n_{\ell-1})}$$

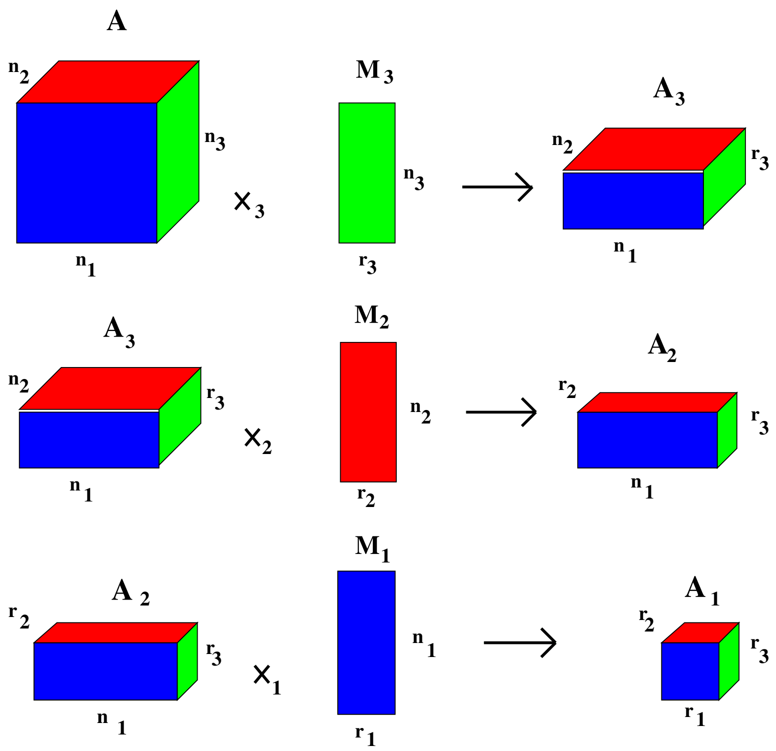
whose columns are the respective fibers along ℓ-mode.



Matlab commands: B= reshape(A,n1,n2*n3);

Or permute(A,[3,2,1]); B3= reshape(A,n3,n1*n2);

Contracted product of tensors



Tucker tensor approximation

Multilinear algebra for the problems of chemometrics, psychometrics, data processing, etc. **[Tucker '66, De Lathauwer et al, 2000]**

$$\mathbf{A}_0 \in \mathbb{V}_{\mathbf{n}}: \quad f(A) := \|\mathbf{A}_{(r)} - \mathbf{A}_0\|^2 \to min \quad \text{over} \;\; \mathbf{A} \in \boldsymbol{\mathcal{T}}_{\mathbf{r}}. \tag{2}$$

The minimisation problem is equivalent to the maximisation problem

$$g(V^{(1)}, ..., V^{(d)}) := \left\| \mathbf{A}_0 \times_1 {V^{(1)}}^T \times ... \times_d {V^{(d)}}^T \right\|^2 \to max \tag{3}$$

over the set of orthogonal matrices $V^{(\ell)} \in \mathbb{R}^{n_\ell \times r_\ell}$.

For given maximizer $V^{(\ell)}$, the core $\boldsymbol{\beta}$ that minimises (2) is

$$\boldsymbol{\beta} = \mathbf{A}_0 \times_1 {V^{(1)}}^T \times_2 \ldots \times_d {V^{(d)}}^T \in \mathbb{R}^{r_1 \times \ldots \times r_d}, \quad \mathbf{A}_{(r)} = \boldsymbol{\beta} \times_1 V^{(1)} \times_2 \ldots \times_d V^{(d)}. \tag{4}$$

Algorithm BTA ($\mathbb{V}_{\mathbf{n}} \to \boldsymbol{\mathcal{T}}_{\mathbf{r},\mathbf{n}}$). Complexity for full format tensors for $d = 3$:
$W_{F \to T} = O(n^4 + n^3 r + n^2 r^2 + n r^3)$.

For function related tensors, **[B. Khoromskij 2006]**:

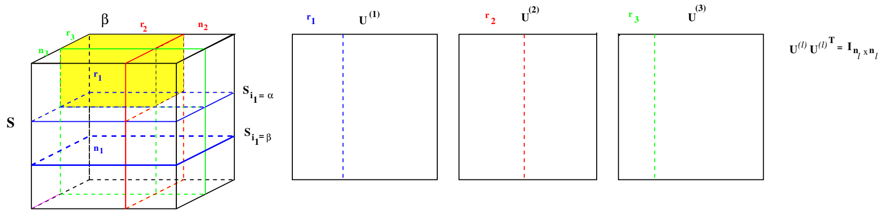
$$\|\mathbf{A}_{(r)} - \mathbf{A}_0\| \le C e^{-\alpha r}, \;\; \text{with } r = \max_\ell r_\ell.$$

Full format tensor to Tucker

Theorem 1. (dth order SVD, higher order SVD (HOSVD), **[De Lathauwer et al, 2000]**).
Every real $n_1 \times n_2 \times ... \times n_d$-tensor $\mathbf{A}$ can be written as the product

$$\mathbf{A} = \mathbf{S} \times_1 U^{(1)} \times_2 U^{(2)} ... \times_d U^{(d)},$$

1. $U^{(\ell)} = [bfu_1^{(\ell)} bfu_2^{(\ell)} ... bfu_{n_\ell}^{(\ell)}]$, ${U^{(\ell)}}^T U^{(\ell)} = I_{n_\ell \times n_\ell}$,
2. Subtensors of $\mathbf{S}$, $\mathbf{S}_{i_\ell = \alpha}$, for fixed ℓth index, have the properties of
(i) all-orthogonality: $\langle \mathbf{S}_{i_\ell=\alpha}, \mathbf{S}_{i_\ell=\beta} \rangle = 0 \quad \text{when} \quad \alpha \neq \beta$,
(ii) ordering: $\|\mathbf{S}_{i_\ell=1}\| \geq \|\mathbf{S}_{i_\ell=2}\| \geq ... \geq \|\mathbf{S}_{i_\ell=n_\ell}\| \geq 0$ for all positive values of ℓ.
The Frobenius norms $\|\mathbf{S}_{i_\ell=i}\|$, $\sigma_i^{(\ell)}$, are ℓ-mode singular values of $\mathbf{A}_{(\ell)}$ and the vector $bfu_i^{(\ell)}$ is an ith ℓ-mode left singular vector of $\mathbf{A}_{(\ell)}$.



Truncated HOSVD

Theorem 2.
(Approximation by HOSVD,**[De Lathauwer et al, 2000]**).
Let the HOSVD of $\mathbf{A}$ be given as in Theorem 1 and let the ℓ-mode rank of $\mathbf{A}$, rank$(A_{(\ell)})$, be equal to R_ℓ ($\ell = 1, ..., d$).
Define a tensor $\tilde{\mathbf{A}}$ by discarding the smallest ℓ-mode singular values $\sigma_{r_\ell+1}^{(\ell)}, \sigma_{r_\ell+2}^{(\ell)}, ..., \sigma_{R_\ell}^{(\ell)}$ for given values of r_ℓ ($\ell = 1, ..., d$), i.e., set the corresponding parts of $\mathbf{S}$ equal to zero. Then we have

$$\|A - \tilde{A}\|^2 \leq \sum_{i_1=r_1+1}^{R_1} {\sigma_{i_1}^{(1)}}^2 + \sum_{i_2=r_2+1}^{R_2} {\sigma_{i_2}^{(2)}}^2 + \cdots + \sum_{i_d=r_d+1}^{R_d} {\sigma_{i_d}^{(d)}}^2.$$

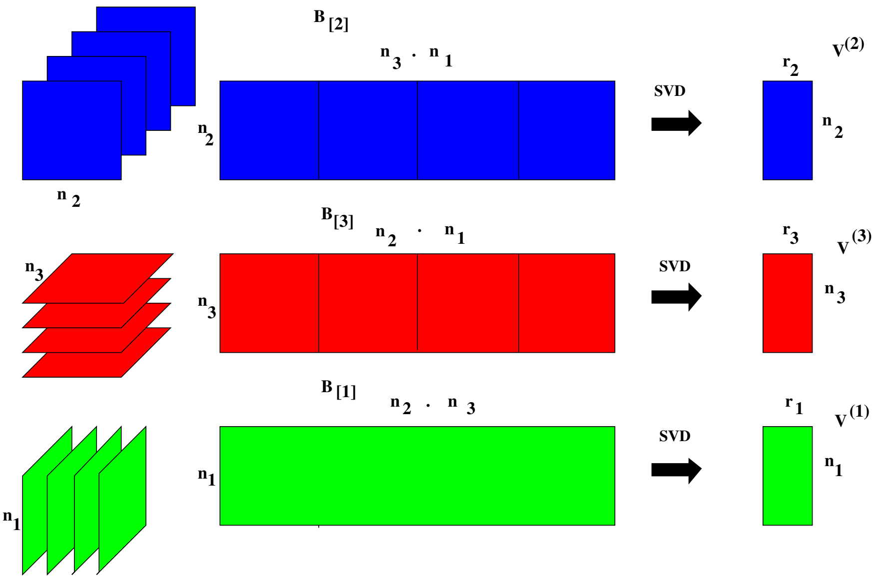
Full-size tensor to Tucker

Algorithm ALS Tucker ($\mathbb{V}_{\mathbf{n}} \to \mathcal{T}_{\mathbf{r},\mathbf{n}}$). Given the input tensor $\mathbf{A} \in \mathbb{V}_{\mathbf{n}}$.

1. Compute an initial guess $V_0^{(\ell)}$ ($\ell = 1, ..., d$) for the ℓ-mode side-matrices by truncated SVD applied to matrix unfolding $A_{(\ell)}$ (cost $O(n^{d+1})$).
2. For $k = 1 : k_{max}$ do: for each $q = 1, ..., d$, and with fixed side-matrices $V^{(\ell)} \in \mathbb{R}^{n \times r_\ell}$, $\ell \neq q$, optimise the side matrix $V^{(q)}$ via computing the dominating r_q-dimensional subspace (truncated SVD) for the respective matrix unfolding $B_{(q)} \in \mathbb{R}^{n \times \overline{r}_q}$, $\overline{r}_q = r_1 ... r_{q-1} r_{q+1} ... r_d$, corresponding to the q-mode contracted product
$$B_{(q)} = \mathbf{A} \times_1 {V^{(1)}}^T \times ... \times_{q-1} {V^{(q-1)}}^T \times_{q+1} {V^{(q+1)}}^T ... \times_d {V^{(d)}}^T.$$
Each iteration costs $O(dr^{d-1} n \min\{r^{d-1}, n\})$, since $\overline{r}_q = O(r^{d-1})$.
3. Compute the core $\boldsymbol{\beta}$ as the representation coefficients of the orthogonal projection of $\mathbf{A}$ onto $\mathbb{T}_{\mathbf{n}} = \otimes_{\ell=1}^{d} \mathbb{T}_\ell$ with $\mathbb{T}_\ell = \mathrm{span}\{v_\ell^k\}_{k=1}^{r_\ell}$,
$$\boldsymbol{\beta} = \mathbf{A} \times_1 {V^{(1)}}^T \times ... \times_d {V^{(d)}}^T \in \mathbb{R}^{r_1 \times \cdots \times r_d},$$
at the cost $O(r^d n)$.

Full-size tensor to Tucker

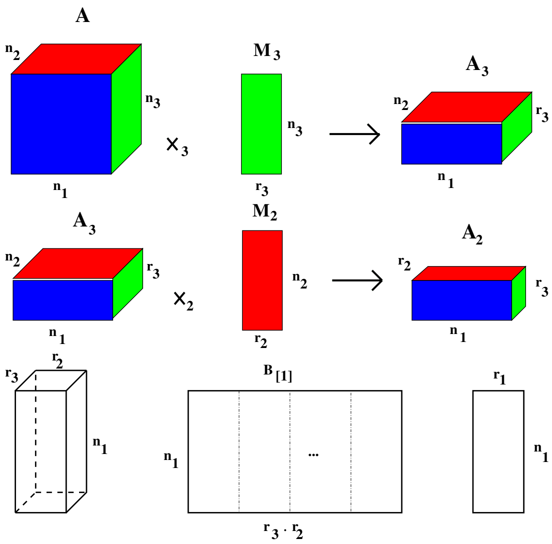
First step in the Tucker decomposition algorithm :
1. Matrix unfoldings of a tensor and calculation of the initial guess for Tucker side-matrices,



Matlab commands: B= reshape(A,n1,n2*n3);
Or permute(A,[3,2,1]); B3= reshape(A,n3,n1*n2);

"Single-hole" tensor

Second step in the Tucker decomposition algorithm for full size tensors:
2. Compute "single-hole" tensors for ALS by using contracted product of the initial 3D full-size tensor and orthogonal matrices $V^{(\ell)}, \ell = 1, 2, 3$ for all dimensions except one.



Update $V^{(\ell)}$ by SVD of unfolding matrix of the "single hole tensor".

Tucker in matlab

```
clear; nstep=12; b=10.0;
T_error=zeros(1,nstep); nd=3; kmax =3; gam=1; R=5.0;
n= 32; b2=b/2; Hunif = b/n;
xcol=-b2:Hunif:b2; [~,n1]=size(xcol); disp(n1); n2= n1; n3= n1;
A3=zeros(n1,n2,n3); A3F=zeros(n1,n2,n3); ycol=xcol; zcol=xcol;

for i=1:n1
    for j=1:n2
        for k=1:n3
          A3(i,j,k)=EXP_GAM_1204(xcol(1,i),ycol(1,j),zcol(1,k),R,gam);
       end
    end
end

for ir=1:nstep
NR=[ir ir ir]; disp(ir);
[U1,U2,U3,LAM3F] = Tucker_full_3D(A3,NR,kmax);
A3F=Tuck_2_F(LAM3F,U1,U2,U3);
err=Tnorm(A3F - A3)/Tnorm(A3); disp(err);
T_error(1,ir)=abs(err);
end
figure(1); semilogy(T_error);
```

```
function [U1,U2,U3,LAM3F] = Tucker_full_3D(A3,NR,kmax)
[n1 n2 n3]=size(A3);
R1=NR(1); R2=NR(2); R3=NR(3); nd=3;
%_______________ Fase I - Initial Guess _________________________

D= permute(A3,[1,3,2]); B1= reshape(D,n1,n2*n3);
[Us, ~, Vs]= svd(double(B1),0); U1=Us(:,1:R1);
D= permute(A3,[2,1,3]); B2= reshape(D,n2,n1*n3);
[Us, ~, Vs]= svd(double(B2),0); U2=Us(:,1:R2);
D= permute(A3,[3,2,1]); B3= reshape(D,n3,n1*n2);
[Us, ~, Vs]= svd(double(B3),0); U3=Us(:,1:R3);
%_______________ Fase II - ALS Iteration __________________________
for k1=1:kmax
 Y1= B1*kron(U2,U3); C1=reshape(Y1,n1,R2*R3);
 [W, S1, V] = svd(double(C1), 0); U1= W(:,1:R1);
 Y2= B2*kron(U3,U1); C2=reshape(Y2,n2,R1*R3);
 [W, S1, V] = svd(double(C2), 0); U2= W(:,1:R2);
 Y3= B3*kron(U1,U2); C3=reshape(Y3,n3,R2*R1);
 [W, S1, V] = svd(double(C3), 0); U3= W(:,1:R3);
end
Y1= B1*kron(U2,U3); LAM3 = U1'*Y1 ; LLL=reshape(LAM3,R1,R3,R2);
LAM3F=permute(LLL,[1,3,2]);
```

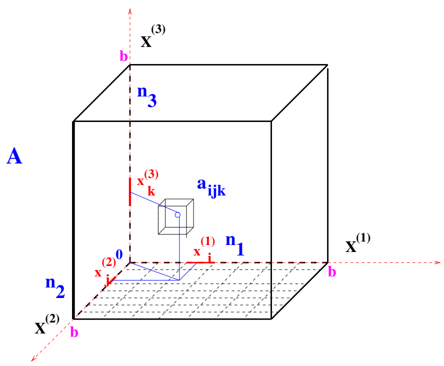

Functional Tucker approximation [B.N. Khoromskij, V. Khoromskaia '07]

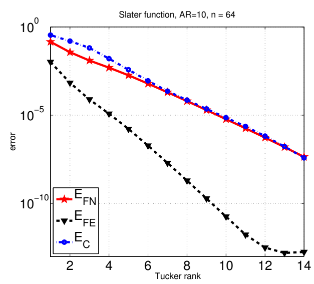
$f(x^{(1)}, x^{(2)}, x^{(3)}) \in \mathbb{R}^3$ is discretized in $[a,b]^3$. Sampling points: $x^{(\ell)}_{i_\ell} = a_\ell + (i_\ell - \frac{1}{2})(\frac{b-a}{n_\ell}), i_\ell = 1, 2, \dots n_\ell$.

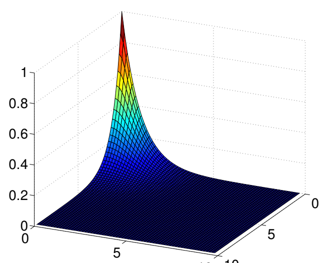
$\Rightarrow$ We generate a tensor $A \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ with entries $a_{ijk} = f(x^{(1)}_i, x^{(2)}_j, x^{(3)}_k)$.

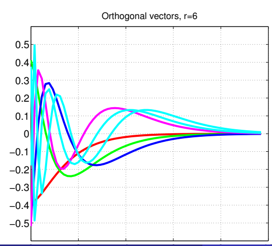
Example: Slater function $g(x) = \exp(-\alpha\|x\|)$ with $x = (x^{(1)}, x^{(2)}, x^{(3)})^T \in \mathbb{R}^3$

$E_{FN} = \frac{\|A_0 - A_{(r)}\|}{\|A_0\|} \quad E_{FE} = \frac{\|A_0\| - \|A_{(r)}\|}{\|A_0\|}, \quad E_C := \frac{\max_{i\in\mathcal{I}} |a_{0,i} - a_{r,i}|}{\max_{i\in\mathcal{I}} |a_{0,i}|}.$



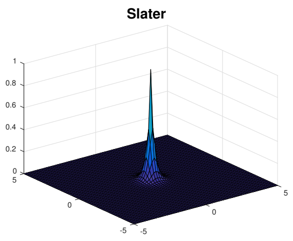


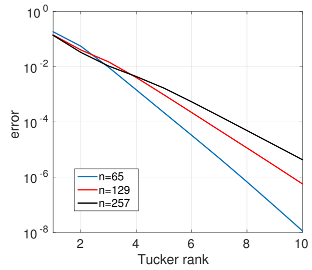


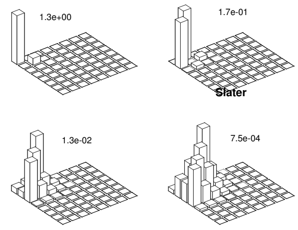


[B.N. Khoromskij, '06],[B.N. Khoromskij, V. Khoromskaia '07]

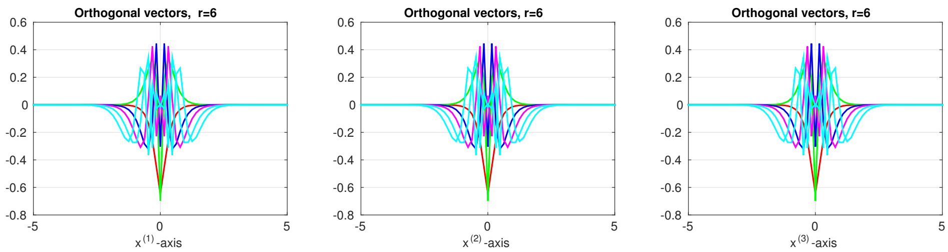
Slater function $f(x) = e^{-\alpha \|x\|)}$, $\quad E_{FN} = \frac{||A_0 - A_{(r)}||}{||A_0||}$





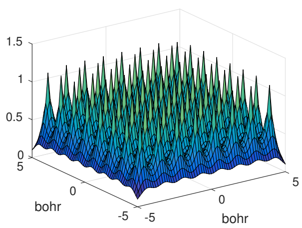


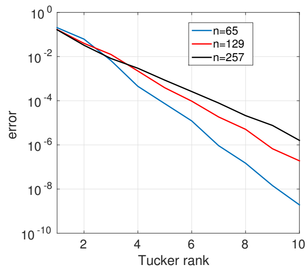
Slater potential $f = e^{-\alpha\sqrt{x^2+y^2}}$, $\alpha = 5$. Decay of Tucker approximation error E_{FN}. Decay of entries of the Tucker core β.



Vectors of orthogonal side matrices of the Tucker decomposition of the tensor corresponding to Slater function

[B.N. Khoromskij, V. Khoromskaia '07],[V. Khoromskaia '10]



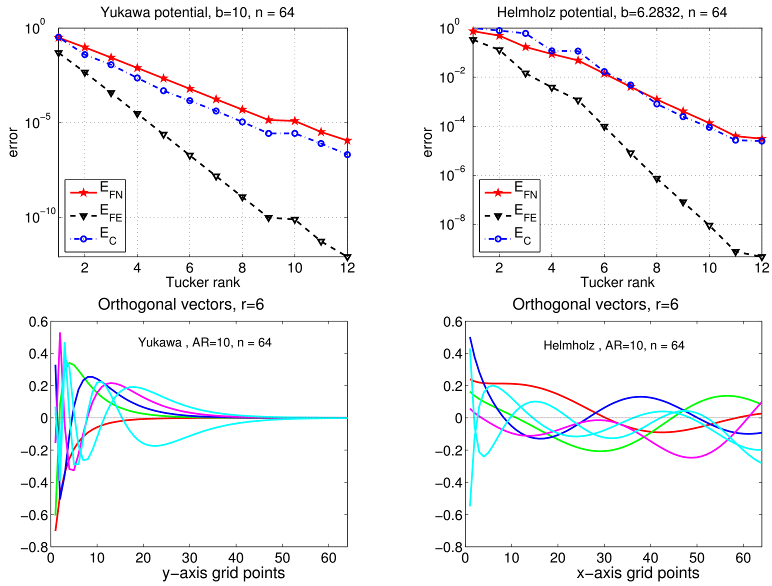


$8 \times 8 \times 8$ Slater.

The "multi-centered Slater potential" obtained by displacing a single Slater potential with respect to the $m \times m \times m$ spatial grid of size $H > 0$, (here $m = 10$)

$$g(x) = \sum_{i=1}^{m} \sum_{j=1}^{m} \sum_{k=1}^{m} e^{-\alpha\sqrt{(x_1 - iH)^2 + (x_2 - jH)^2 + (x_3 - kH)^2}}.$$

$$g(x) = \frac{e^{-\|x\|}}{\|x\|} \qquad g(x) = \frac{\cos \|x\|}{\|x\|} \qquad x = (x_1, x_2, x_3) \in \mathbb{R}^3$$



[V. Khoromskaia, 2010]

Disadvantages of the standard Tucker algorithm

Disadvantages of the Tucker algorithm for full size tensors :

1. The standard Tucker algorithm is nontractable for large n!
2. Iif we have a canonical tensor with large R and n.
3. How to reduce the rank of a canonical tensor with large R and n ?

For example, if we want to represent this multidimensional function on fine grid and with much lower rank:

A separable function

$$f(x_1, x_2, \ldots, x_d) = \sum_{k=1}^{R} f_{1,k}(x_1) f_{2,k}(x_2) \ldots f_{d,k}(x_2).$$

Canonical to Tucker Approximation

Theorem **[B.N. Khoromskij, V. Khoromskaia(SISC, 34 (2009))]**
Let $A \in \mathcal{C}_{R,\mathbf{n}}$. Then the minimisation problem

$$A \in \mathcal{C}_{R,\mathbf{n}} \subset \mathbb{V}_{\mathbf{n}} : \quad A_{(\mathbf{r})} = \operatorname{argmin}_{V \in \mathcal{T}_{\mathbf{r},\mathbf{n}}} \|A - V\|_{\mathbb{V}_{\mathbf{r}}}, \tag{5}$$

is equivalent to

$$[Z^{(1)}, ..., Z^{(d)}] = \operatorname{argmax}_{V^{(\ell)} \in \mathcal{M}_\ell} \left\| \sum_{\nu=1}^{R} \xi_\nu \left({V^{(1)}}^T u_\nu^{(1)} \right) \otimes ... \otimes \left({V^{(d)}}^T u_\nu^{(d)} \right) \right\|^2_{\mathbb{V}_{\mathbf{r}}},$$

$V^{(\ell)} \in \mathbb{R}^{n_\ell \times r_\ell}$ – orthogonal.
Init. guess: RHOSVD $Z_0^{(\ell)} \Rightarrow$ the truncated SVD of side matrices
$U^{(\ell)} = [u_1^{(\ell)} ... u_R^{(\ell)}]$, under the compatibility condition $r_\ell \leq rank(U^{(\ell)})$
Error bounds for RHOSVD

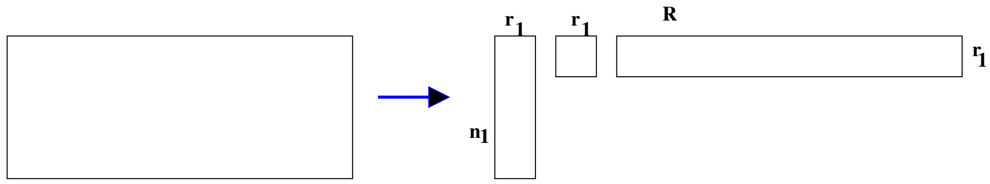
$$\|A - A^0_{(\mathbf{r})}\| \leq \|\boldsymbol{\xi}\| \sum_{\ell=1}^{d} (\sum_{k=r_\ell+1}^{\min(n,R)} \sigma^2_{\ell,k})^{1/2}, \quad \text{where } \|\boldsymbol{\xi}\|^2 = \sum_{\nu=1}^{R} \xi_\nu^2.$$

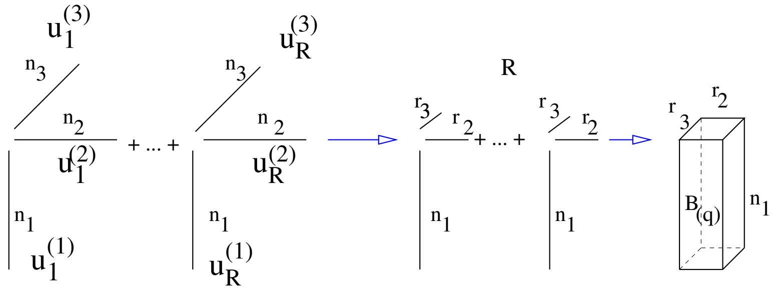
Canonical-to-Tucker approximation

[B.N. Khoromskij, V. Khoromskaia'07],[B.N. Khoromskij, V. Khoromskaia'08]
Canonical-to-Tucker (3D). Given a tensor in the canonical format,

$$\mathbf{A} = \mathbf{C} \times_1 U^{(1)} \times_2 U^{(2)} \ldots \times_d U^{(d)}, \qquad \mathbf{C} \in \mathbb{R}^{R \times \ldots \times R}, \qquad U^{(\ell)} = [\mathbf{u}_1^{(\ell)} \ldots \mathbf{u}_R^{(\ell)}] \in \mathbb{R}^{n_\ell \times R}.$$

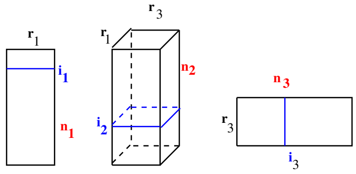
Main point: RHOSVD of side matrices: $U^{(\ell)} \approx Z_r^{(\ell)} \Sigma_r^{(\ell)} W_r^{(\ell)}$, where $U_r^{(\ell)} \in \mathbb{R}^{n_\ell \times r}$, $r \ll R$.





Application of the C2T decomposition

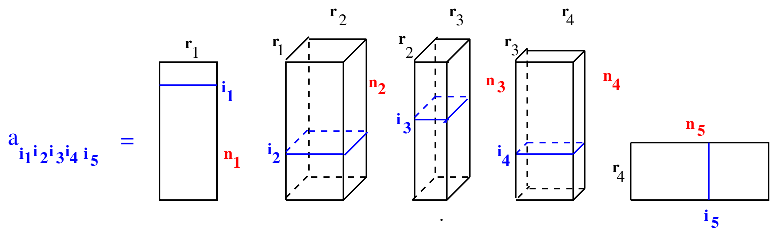
Canonical-to-Tucker decomposition (3D) **[B.N. Khoromskij, V.K. '08]**
For $d = 3$ the non-contracted C2T tensor looks as:



MPS (Matrix Product States) format was introduced by **[White '92]**,
Tensor train **[Oseledets, Tyrtyshnikov '09]** (for canonical function related tensors),

$a_{i_1 \ldots i_d} = \sum_\alpha G^{(1)}_{\alpha_1}[i_1] G^{(2)}_{\alpha_1 \alpha_2}[i_2] \cdots G^{(d)}_{\alpha_{d-1}}[i_d] \equiv G^{(1)}[i_1] G^{(2)}[i_2] \ldots G^{(d)}[i_d]$, where $G^{(\ell)}[i_\ell] \in \mathbb{R}^{r_{\ell-1} \times r_\ell}$.

Example for $d = 5$:



$$a_{i_1, i_2, i_3, i_4, i_5} = \prod_{k=1}^{5} B_k(i_k), \quad A \in \mathbb{R}^{n1 \times n_2 \times n3 \times n_4 \times n_5}, \quad B_k \in \mathbb{R}^{r_{k-1} \times r_k}, \quad r_0 = r_5 = 1.$$

(See also hierarchical tensor format (HT) **[Hackbusch, Kuehn 2009]**).

Multigrid Tucker decomposition

In electronic structure calculations we need large $n \times n \times n$ 3D Cartesian grids, and robust rank reduction for large R.

The conventional BTA algorithm ($d = 3$) $\Rightarrow$ small n, moderate R:
(a) For full format tensors

$$W_{F2T} = O(n^{d+1}) \quad \Rightarrow \quad W_{F2T} = O(n^4)$$

(b) for the canonical rank-R input tensors

$$W_{C2T} = O(Rn \min\{R, n\} + r^{d-1} n \min\{r^{d-1}, n\}),$$
$$\Rightarrow W_{C2T} = O(nR^2 + nr^4).$$

$\Rightarrow$ SVD of side matrices of the sizes $\geq 10^4 \times 10^4$ is computationally unfeasible.

The answer to the problem:

Multigrid accelerated best Tucker approximation:

$$W_{F2T} = O(n^3), \quad \text{for full size tensors}$$
$$W_{C2T} = O(rRn), \quad \text{for canonical rank-R tensors.}$$

Multigrid Tucker decomposition

1. Sequence of nonlinear appr. problems for $A = A_n$,$n = n_m := n_0 2^m$, $m = 0, 1, \dots, M$, on a sequence of refined grids $\omega_{\mathbf{3},n_m}$.
2. $Z_0^{(q)}$ on grid $\omega_{\mathbf{3},n_m}$ $\to$ by linear interp. of $Z^{(q)} \in \mathbb{R}^{n_{m-1} \times r_q}$ from $\omega_{\mathbf{3},n_{m-1}}$.
3. The *restricted ALS iteration*, is based on "most important fibers" (MIFs) of unfolding matrices.
   Positions of MIFs are extracted at the coarsest grid

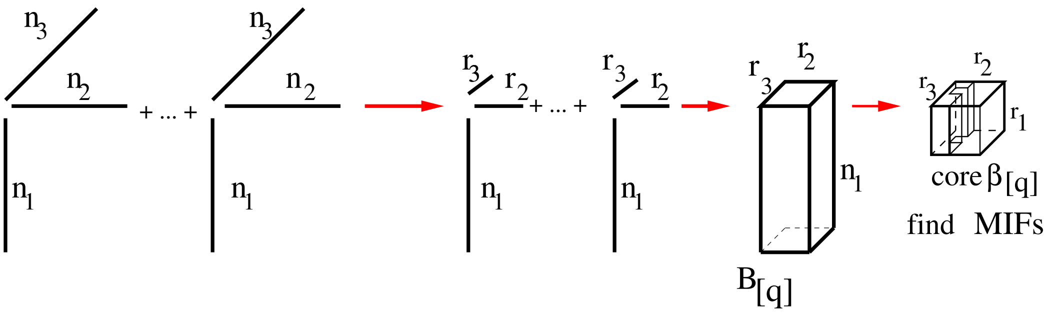
$$\boldsymbol{\beta}_{(q)} = {Z^{(q)}}^T B_{(q)} \in \mathbb{R}^{r_q \times \bar{r}_q}. \tag{6}$$

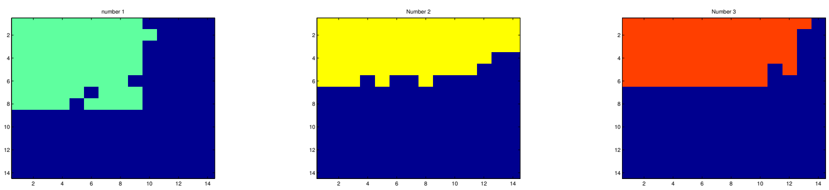
   Location of MIFs corresponds to pr columns in $\boldsymbol{\beta}_{(q)}$, $p \ll r$ with maximal Euclidean norms (maximal energy principle).
4. Complexity for $d = 3$: $O(Rrn_M + p^2 r^2 n_M)$.
   (Linear w.r.t. n and R !)

MGA C2T ($\mathcal{C}_{R,\mathbf{n}}$)

[B.N. Khoromskij, V.Kh. (SISC, 34 (2009))]

$B_{(q)} \to$ unfolding of $(B_{[q]})$, $\quad \beta_{(q)} = {Z^{(q)}}^T B_{(q)}$; $n_0 = 64$, $n_M = 16384$





MIFs for H_2O, $p = 4$.

C2T + T2C rank reduction

[B.N. Khoromskij, V. Khoromskaia '09]
Next step: Tucker-to-canonical transform.

$$\mathbf{A} = \boldsymbol{\beta} \times_1 V^{(1)} \times_2 V^{(2)} \times_3 V^{(3)}$$

where $\boldsymbol{\beta} \in \mathbb{R}^{r \times r \times r}$, $V^{(\ell)} \in \mathbb{R}^{n \times r}$.

Represent the (small) core tensor in the canonical tensor format,

$$\boldsymbol{\beta} = \mathbf{C} \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)}, \quad U^{(1)} \in \mathbb{R}^{r \times R_r},$$

where $R_r \leq r^2$.
Then

$$\mathbf{A} = \left(\mathbf{C} \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)}\right) \times_1 V^{(1)} \times_2 V^{(2)} \times_3 V^{(3)}$$

which reduces to

$$\mathbf{A} = \mathbf{C} \times_1 W^{(1)} \times_2 W^{(2)} \times_3 W^{(3)}, \quad W^{(3)} \in \mathbb{R}^{n \times R_r}$$

where $W^{(\ell)} = V^{(\ell)} U^{(\ell)}$.

Tensor operations

$$A_1 = \boldsymbol{\beta} \times_1 U^{(1)} \times_2 U^{(2)} \ldots \times_d U^{(d)}, \quad A_2 = \boldsymbol{\zeta} \times_1 V^{(1)} \times_2 V^{(2)} \ldots \times_d V^{(d)}, \tag{7}$$

the *scalar product* is computed by

$$\langle A_1, A_2 \rangle := \sum_{\mathbf{k}=1}^{\mathbf{r}_1} \sum_{\mathbf{m}=1}^{\mathbf{r}_2} \beta_{k_1 \ldots k_d} \zeta_{m_1 \ldots m_d} \prod_{\ell=1}^{d} \left\langle u_{k_\ell}^{(\ell)}, v_{m_\ell}^{(\ell)} \right\rangle. \tag{8}$$

In fact, applying the definition of the scalar product to the rank-1 tensors (with $R = \mathbf{r} = 1$), we have

$$\langle A_1, A_2 \rangle := \sum_{\mathbf{i} \in \mathcal{I}} u_{i_1}^{(1)} \cdots u_{i_d}^{(d)} v_{i_1}^{(1)} \cdots v_{i_d}^{(d)}$$

$$= \sum_{i_1=1}^{n_1} u_{i_1}^{(1)} v_{i_1}^{(1)} \cdots \sum_{i_d=1}^{n_d} u_{i_d}^{(d)} v_{i_d}^{(d)} = \prod_{\ell=1}^{d} \left\langle u^{(\ell)}, v^{(\ell)} \right\rangle. \tag{9}$$

Tensor operations

For given tensors $A, B \in \mathbb{R}^{\mathcal{I}}$, the *Hadamard product* $A \odot B \in \mathbb{R}^{\mathcal{I}}$ of two tensors of the same size $\mathcal{I}$ is defined by the componentwise product,

$$(A \odot B)_{\mathbf{i}} = a_{\mathbf{i}} \cdot b_{\mathbf{i}}, \quad \mathbf{i} \in \mathcal{I}.$$

Hence, for $A_1, A_2 \in \boldsymbol{\mathcal{T}}_{\mathbf{r}}$, as in (7), we tensorize the Hadamard product by

$$A_1 \odot A_2 := \sum_{k_1, m_1 = 1}^{r} \cdots \sum_{k_d, m_d = 1}^{r} \beta_{k_1 \ldots k_d} \zeta_{m_1 \ldots m_d} \left(u_{k_1}^{(1)} \odot v_{m_1}^{(1)} \right) \otimes \ldots \otimes \left(u_{k_d}^{(d)} \odot v_{m_d}^{(d)} \right). \qquad (10)$$

For the rank-1 tensors (with $\boldsymbol{\beta} = \boldsymbol{\zeta} = 1$), we have

$$\begin{aligned} (A_1 \odot \; A_2)_{\mathbf{i}} =& (u_{i_1}^{(1)} v_{i_1}^{(1)}) \cdots (u_{i_d}^{(d)} v_{i_d}^{(d)}) \\ =& \left(u^{(1)} \odot v^{(1)} \right) \otimes \cdots \otimes \left(u^{(d)} \odot v^{(d)} \right). \end{aligned} \qquad (11)$$

Then (10) follows by summation over all rank-1 terms in $A_1 \odot A_2$. Relation (10) leads to the storage requirements

$$\mathcal{N}_{st(A \odot B)} = O(dr^2 n + r^{2d}),$$

that includes the memory size for d modes $n \times r \times r$ Tucker vectors, and for the new Tucker core of size $(r^2)^d$.

Tensor operations

For given $A_1, A_2 \in \boldsymbol{\mathcal{T}}_{\mathbf{r}}$, see (7), we now tensorize the convolution product via

$$A_1 * A_2 := \sum_{\mathbf{k}=1}^{\mathbf{r}} \sum_{\mathbf{m}=1}^{\mathbf{r}} \beta_{k_1 \ldots k_d} \zeta_{m_1 \ldots m_d} \left(u_{k_1}^{(1)} * u_{m_1}^{(1)} \right) \otimes \ldots \otimes \left(u_{k_d}^{(d)} * v_{m_d}^{(d)} \right). \qquad (12)$$

This relation again follows from the analysis for the case of rank-1 convolving tensors F and G, similar to the discussion for scalar product of tensors,

$$\begin{aligned} (F * G)_{\mathbf{j}} :=& \sum_{\mathbf{i} \in \mathcal{I}} f_{i_1}^{(1)} \cdots f_{i_d}^{(d)} g_{j_1 - i_1 + 1}^{(1)} \cdots g_{j_d - i_d + 1}^{(d)} \\ =& \sum_{i_1 = 1}^{n_1} f_{i_1}^{(1)} g_{j_1 - i_1 + 1}^{(1)} \cdots \sum_{i_d = 1}^{n_d} f_{i_d}^{(d)} g_{j_d - i_d + 1}^{(d)} = \prod_{\ell = 1}^{d} \left(f^{(\ell)} * g^{(\ell)} \right)_{j_\ell}. \end{aligned} \qquad (13)$$

Assuming that "one-dimensional" convolutions of n-vectors, $u_{k_\ell}^{(\ell)} * v_{m_\ell}^{(\ell)} \in \mathbb{R}^{2n-1}$, can be computed in $O(n \log n)$ operations, we arrive at the overall complexity estimate

$$\mathcal{N}_{\cdot * \cdot} = O(dr^2 n \log n + r^{2d}).$$

In our particular case of equidistant grids we obtain (by setting $a = u_{k_\ell}^{(\ell)}, b = v_{m_\ell}^{(\ell)} \in \mathbb{R}^n$)

$$(u_{k_\ell}^{(\ell)} * v_{m_\ell}^{(\ell)})_j = \sum_{i=1}^{n} a_i b_{j-i+1}, \quad j = 1, \ldots, 2n-1.$$

Hence, the 1D convolution can be performed by FFT in $O(n \log n)$ operations.