Low-rank tensor methods for high-dimensional eigenvalue problems

Daniel Kressner

CADMOS Chair of Numerical Algorithms and HPC
MATHICSE / SMA / SB
EPF Lausanne
http://anchp.epfl.ch

Based on joint work with
Ch. Tobler (MathWorks)
M. Steinlechner (EPFL), A. Uschmajew (EPFL)

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

cadmos
center for advanced modeling science

Low-rank tensor techniques

- Emerged during last five years in numerical analysis.
- Successfully applied to:
  - parameter-dependent / multi-dimensional integrals;
  - electronic structure calculations: Hartree-Fock / DFT;
  - stochastic and parametric PDEs;
  - high-dimensional Boltzmann / chemical master / Fokker-Planck / Schrödinger equations;
  - micromagnetism;
  - rational approximation problems;
  - computational homogenization;
  - computational finance;
  - stochastic automata networks;
  - multivariate regression and machine learning;
  - ...
- For references on these applications, see
  - L. Grasedyck, DK, Ch. Tobler (2013). A literature survey of low-rank tensor approximation techniques. GAMM-Mitteilungen, 36(1).
  - W. Hackbusch (2012). Tensor Spaces and Numerical Tensor Calculus, Springer.

High dimensionality

Continuous problem on d-dimensional domain with $d \gg 1$

$\Downarrow$ Straightforward discretization $\Downarrow$

Discretized problem of order $O(n^d)$

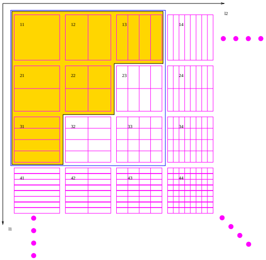
Other causes of high dimensionality:

- parameter dependencies
  - parametrized coefficients
  - parametrized topology
- stochastic coefficients
- systems describing joint probability distributions
- ...

Dealing with high dimensionality

Established techniques:

- Sparse grid collocation/Galerkin methods.
- Adaptive (wavelet) methods.
- Monte Carlo method.
- Reduced basis method.
- . . .



Common trait:

Smart discretization $\rightsquigarrow$ system *hopefully* of order $\ll O(N^D)$...

... but not in this talk!

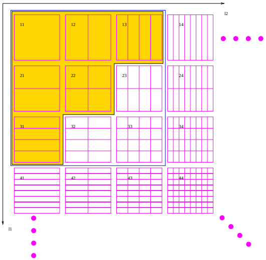
This talk:

Straightforward discretization $\rightsquigarrow$ system of order $O(N^D)$.

Dealing with high dimensionality

Established techniques:

- Sparse grid collocation/Galerkin methods.
- Adaptive (wavelet) methods.
- Monte Carlo method.
- Reduced basis method.
- ...



Common trait:

Smart discretization $\rightsquigarrow$ system *hopefully* of order $\ll O(N^D)$...

... but not in this talk!

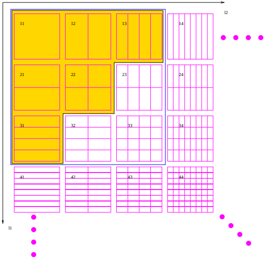
This talk:

Straightforward discretization $\rightsquigarrow$ system of order $O(N^D)$.

Dealing with high dimensionality

Established techniques:

- Sparse grid collocation/Galerkin methods.
- Adaptive (wavelet) methods.
- Monte Carlo method.
- Reduced basis method.
- ...



Common trait:

Smart discretization $\rightsquigarrow$ system *hopefully* of order $\ll O(N^D)$...

... but not in this talk!

This talk:

Straightforward discretization $\rightsquigarrow$ system of order $O(N^D)$.

Example: PDE-eigenvalue problem

Goal: Compute smallest eigenvalue for

$$\begin{array}{rcll} \Delta u(\xi) + V(\xi)u(\xi) & = & \lambda\, u(\xi) & \text{in } \Omega = [0,1]^d, \\ u(\xi) & = & 0 & \text{on } \partial\Omega. \end{array}$$

Assumption: Potential represented as

$$V(\xi) = \sum_{j=1}^{s} V_j^{(1)}(\xi_1)\, V_j^{(2)}(\xi_2) \cdots V_j^{(d)}(\xi_d).$$

$\rightsquigarrow$ finite difference discretization

$$\mathcal{A}\mathbf{u} = (\mathcal{A}_L + \mathcal{A}_V)\mathbf{u} = \lambda \mathbf{u},$$

with

$$\begin{array}{rcl} \mathcal{A}_L & = & \sum_{j=1}^{d} \underbrace{I \otimes \cdots \otimes I}_{d-j \text{ times}} \otimes A_L \otimes \underbrace{I \otimes \cdots \otimes I}_{j-1 \text{ times}}, \\ \mathcal{A}_V & = & \sum_{j=1}^{s} A_{V,j}^{(d)} \otimes \cdots \otimes A_{V,j}^{(2)} \otimes A_{V,j}^{(1)}. \end{array}$$

Example: Henon-Heiles potential

Consider $\Omega = [-10, 2]^d$ and potential ([Meyer et al. 1990; Raab et al. 2000; Faou et al. 2009])

$$V(\xi) = \frac{1}{2}\sum_{j=1}^{d} \sigma_j \xi_j^2 + \sum_{j=1}^{d-1}\left(\sigma_*(\xi_j\xi_{j+1}^2 - \frac{1}{3}\xi_j^3) + \frac{\sigma_*^2}{16}(\xi_j^2 + \xi_{j+1}^2)^2\right).$$

with $\sigma_j \equiv 1$, $\sigma_* = 0.2$.

Discretization with $n = 128$ dof/dimension for $d = 20$ dimensions.

- Eigenvector has $n^d \approx 10^{42}$ entries.
- Explicit storage of eigenvector would require 10^{25} exabyte![1]

[1] Global data storage in 2011 calculated at 295 exabyte, see http://www.bbc.co.uk/news/technology-12419672.

Example: Henon-Heiles potential

Consider $\Omega = [-10, 2]^d$ and potential ([Meyer et al. 1990; Raab et al. 2000; Faou et al. 2009])

$$V(\xi) = \frac{1}{2}\sum_{j=1}^{d} \sigma_j \xi_j^2 + \sum_{j=1}^{d-1} \Big(\sigma_* (\xi_j \xi_{j+1}^2 - \frac{1}{3}\xi_j^3) + \frac{\sigma_*^2}{16}(\xi_j^2 + \xi_{j+1}^2)^2 \Big).$$

with $\sigma_j \equiv 1$, $\sigma_* = 0.2$.

Discretization with $n = 128$ dof/dimension for $d = 20$ dimensions.

- Eigenvector has $n^d \approx 10^{42}$ entries.
- Explicit storage of eigenvector would require 10^{25} exabyte![1]

Solved with accuracy 10^{-12} in less than 1 hour on laptop.

[1] Global data storage in 2011 calculated at 295 exabyte, see `http://www.bbc.co.uk/news/technology-12419672`.

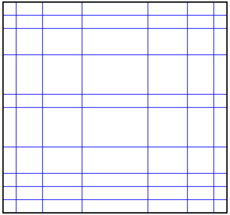
Contents

Low-rank matrices

Discretization of bivariate function

- Bivariate function: $f(x,y): \left[x_{\min}, x_{\max}\right] \times \left[y_{\min}, y_{\max}\right] \to \mathbb{R}$.
- Function values on tensor grid $[x_1, \ldots, x_n] \times [y_1, \ldots, y_m]$.
- Normally collected in looong vector:

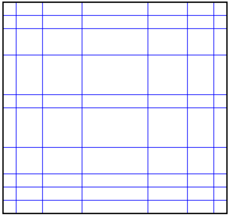
$$f = \begin{bmatrix} f(x_1, y_1) \\ f(x_2, y_1) \\ \vdots \\ f(x_m, y_1) \\ f(x_1, y_2) \\ f(x_2, y_2) \\ \vdots \\ f(x_m, y_2) \\ \vdots \\ \vdots \\ f(x_1, y_n) \\ f(x_2, y_n) \\ \vdots \\ f(x_m, y_n) \end{bmatrix}$$



Discretization of bivariate function

- Bivariate function: $f(x, y) : [x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}] \to \mathbb{R}$.
- Function values on tensor grid $[x_1, \dots, x_n] \times [y_1, \dots, y_m]$.
- Collected in a matrix:

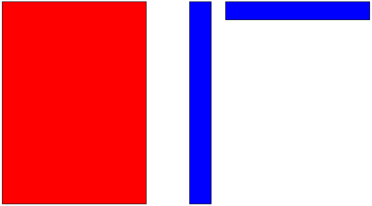
$$F = \begin{bmatrix} f(x_1, y_1) & f(x_1, y_2) & \cdots & f(x_1, y_n) \\ f(x_2, y_1) & f(x_2, y_2) & \cdots & f(x_2, y_n) \\ \vdots & \vdots & & \vdots \\ f(x_m, y_1) & f(x_m, y_2) & \cdots & f(x_m, y_n) \end{bmatrix}$$



Low-rank approximation

Setting: Matrix $X \in \mathbb{R}^{n \times m}$, m and n too large to compute/store X explicitly.

Idea: Replace X by RS^T with $R \in \mathbb{R}^{n \times r}$, $S \in \mathbb{R}^{m \times r}$ and $r \ll m, n$.



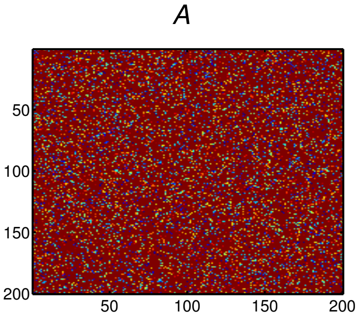
	X	RS^T
Memory	nm	$nr + rm$

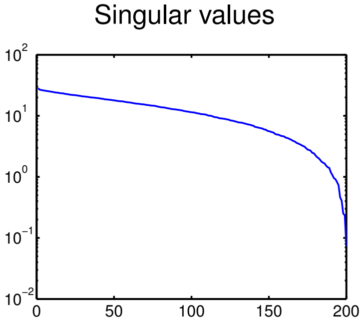
$$\min \left\{ \|X - RS^T\|_2 : \; R \in \mathbb{R}^{n \times r}, S \in \mathbb{R}^{m \times r} \right\} = \sigma_{r+1}.$$

with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\min\{m,n\}}$ of X.

Singular values of random matrices

```
A = rand(200);
semilogy(svd(A))
```



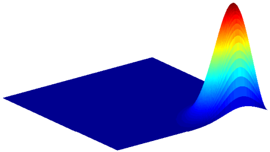


No reasonable low-rank approximation possible

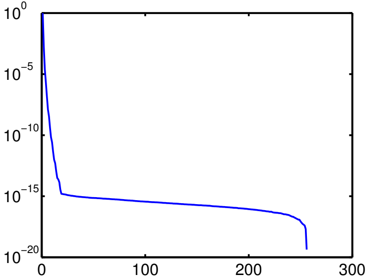
Singular values of ground state

- Computed ground state for Henon-Heiles potential for $d = 2$.
- Reshaped ground state into matrix.

Ground state



Singular values



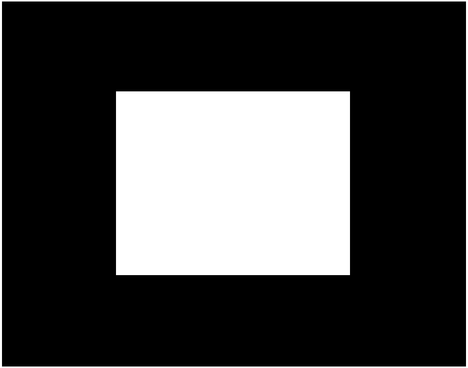
Excellent rank-10 approximation possible

When to expect good low-rank approximations

Rule of thumb: Smoothness helps.

When to expect good low-rank approximations

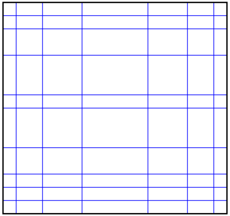
Rule of thumb: Smoothness helps, but is not always needed.



Discretization of bivariate function

- Bivariate function: $f(x,y) : [x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}] \to \mathbb{R}$.
- Function values on tensor grid $[x_1, \dots, x_n] \times [y_1, \dots, y_m]$:

$$F = \begin{bmatrix} f(x_1,y_1) & f(x_1,y_2) & \cdots & f(x_1,y_n) \\ f(x_2,y_1) & f(x_2,y_2) & \cdots & f(x_2,y_n) \\ \vdots & \vdots & & \vdots \\ f(x_m,y_1) & f(x_m,y_2) & \cdots & f(x_m,y_n) \end{bmatrix}$$



Basic but crucial observation: $f(x,y) = g(x)h(y) \rightsquigarrow$

$$F = \begin{bmatrix} g(x_1)h(y_1) & \cdots & g(x_1)h(y_n) \\ \vdots & & \vdots \\ g(x_m)h(y_1) & \cdots & g(x_m)h(y_n) \end{bmatrix} = \begin{bmatrix} g(x_1) \\ \vdots \\ g(x_m) \end{bmatrix} \begin{bmatrix} h(y_1) & \cdots & h(y_n) \end{bmatrix}$$

Separability implies rank 1.

Separability and low rank

Approximation by sum of separable functions

$$f(x,y) = \underbrace{g_1(x)h_1(y) + \cdots + g_r(x)h_r(y)}_{=: f_r(x,y)} + \text{error}.$$

Define

$$F_r = \begin{bmatrix} f_r(x_1,y_1) & \cdots & f_r(x_1,y_n) \\ \vdots & & \vdots \\ f_r(x_m,y_1) & \cdots & f_r(x_m,y_n) \end{bmatrix}.$$

Then F_r has rank $\leq r$ and $\|F - F_r\|_F \leq \sqrt{mn} \times \text{error}$.

$\rightsquigarrow$

$$\sigma_{r+1}(F) \leq \|F - F_r\|_2 \leq \|F - F_r\|_F \leq \sqrt{mn} \times \text{error}.$$

Semi-separable approximation implies low-rank approximation.

Semi-separable approximation by polynomials

Solution of approximation problem

$$f(x,y) = g_1(x)h_1(y) + \cdots + g_r(x)h_r(y) + \text{error}.$$

not trivial; g_j, h_j can be chosen arbitrarily!

General construction by polynomial interpolation:

1. Lagrange interpolation of $f(x,y)$ in y-coordinate:

$$I_y[f](x,y) = \sum_{j=1}^{r} f(x,\theta_j) L_j(y)$$

   with Lagrange polynomials L_j of degree $r-1$ on $[x_{\min}, x_{\max}]$.

2. Interpolation of $I_y[f]$ in x-coordinate:

$$I_x[I_y[f]](x,y) = \sum_{i,j=1}^{r} f(\xi_i,\theta_j) L_i(x) L_j(y) \mathrel{\hat{=}} \sum_{i=1}^{r} L_{i,x}(x) L_{j,y}(y),$$

   where $f[f(\xi_i,\theta_j)]_{i,j}$ is “diagonalized” by SVD.

Semi-separable approximation by polynomials

$$\begin{aligned} \text{error} &\le \|f - I_x[I_y[f]]\|_\infty \\ &= \|f - I_x[f] + I_x[f] - I_x[I_y[f]]\|_\infty \\ &\le \|f - I_x[f]\|_\infty + \|I_x\|_\infty \|f - I_y[f]\|_\infty \end{aligned}$$

with Lebesgue constant $\|I_x\|_\infty \sim \log r$ when using Chebyshev interpolation nodes.

[Temlyakov'1992, Uschmajew/Schneider'2013]:

$$\sup_{f \in B^s} \inf \left\| f(x,y) - \sum_{k=1}^{r} g_k(x) h_k(y) \right\|_{L^2} \sim r^{-s},$$

with Sobolev space B^s of periodic functions with partial derivatives up to order s.

Semi-separable approximation by polynomials

$$\begin{aligned} \text{error} &\leq \|f - I_x[I_y[f]]\|_\infty \\ &= \|f - I_x[f] + I_x[f] - I_x[I_y[f]]\|_\infty \\ &\leq \|f - I_x[f]\|_\infty + \|I_x\|_\infty \|f - I_y[f]\|_\infty \end{aligned}$$

with Lebesgue constant $\|I_x\|_\infty \sim \log r$ when using Chebyshev interpolation nodes.

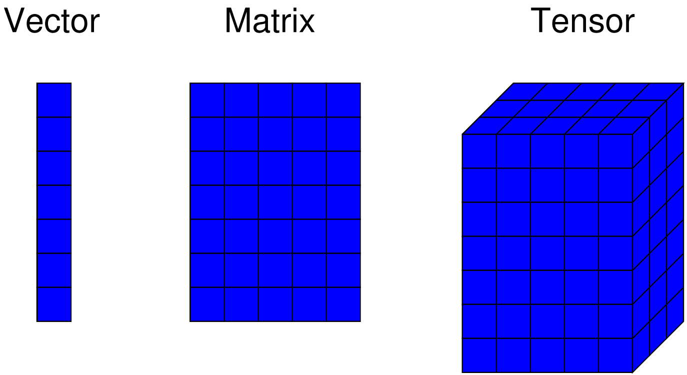
[Temlyakov'1992, Uschmajew/Schneider'2013]:

$$\sup_{f \in B^s} \inf \left\| f(x,y) - \sum_{k=1}^{r} g_k(x) h_k(y) \right\|_{L^2} \sim r^{-s},$$

with Sobolev space B^s of periodic functions with partial derivatives up to order s.

Low-rank tensors

Vectors, matrices, and tensors



- scalar $=$ tensor of order 0
- (column) vector $=$ tensor of order 1
- matrix $=$ tensor of order 2
- tensor of order 3
  $= n_1 n_2 n_3$ numbers arranged in $n_1 \times n_2 \times n_3$ array

Tensors of arbitrary order

A d-th order tensor $\mathcal{X}$ of size $n_1 \times n_2 \times \cdots \times n_d$ is a d-dimensional array with entries

$$\mathcal{X}_{i_1,i_2,\ldots,i_d}, \qquad i_\mu \in \{1,\ldots,n_\mu\} \text{ for } \mu = 1,\ldots,d.$$

In the following, entries of $\mathcal{X}$ are real (for simplicity) $\rightsquigarrow$

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

Multi-index notation:

$$\mathfrak{I} = \{1,\ldots,n_1\} \times \{1,\ldots,n_2\} \times \cdots \times \{1,\ldots,n_d\}.$$

Then $i \in \mathfrak{I}$ is a tuple of d indices:

$$i = (i_1, i_2, \ldots, i_d).$$

Allows to write entries of $\mathcal{X}$ as $\mathcal{X}_i$ for $i \in \mathfrak{I}$.

Functions and tensors

Consider a function $f(x_1, \ldots, x_d) \in \mathbb{R}$ in d variables $x_1, \ldots, x_d$.
Tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ represents discretization of u:

- $\mathcal{X}$ contains function values of f evaluated on a grid; **or**
- $\mathcal{X}$ contains coefficients of truncated expansion in tensorized basis functions:

$$f(x_1, \ldots, x_d) \approx \sum_{i \in \mathfrak{I}} \mathcal{X}_i \, \phi_{i_1}(x_1)\phi_{i_2}(x_2) \cdots \phi_{i_d}(x_d).$$

Functions and tensors

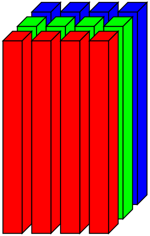
One of many ways to separate variables of f:

$$f(x_1, x_2, \ldots, x_d) \approx \sum_{k=1}^{r} g_k(x_1) h_k(x_2, \ldots, x_d)$$

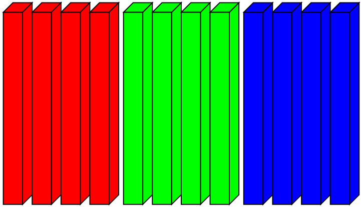
Corresponding matrix/tensor decomposition?

Matricization

Stack 1-fibers into an $n_1 \times (n_2 \cdots n_d)$ matrix:



$\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$



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

Separation wrt $\{x_1\}$ corresponds to low-rank matrix approximation

$$X^{(1)} \approx U_1 V_1^T.$$

Tucker decomposition

- Consider low-rank decompositions corresponding to $\{x_1\}, \{x_2\}, \{x_3\}$:

$$
\begin{aligned}
X^{(1)} &\approx U_1 V_1^T \\
X^{(2)} &\approx U_2 V_2^T \\
X^{(3)} &\approx U_3 V_3^T
\end{aligned}
$$

- Form $r_1 \times r_2 \times r_3$ core tensor $\mathcal{C}$:

$$\text{vec}(\mathcal{C}) := \left(U_3^T \otimes U_2^T \otimes U_1^T\right) \cdot \text{vec}(\mathcal{X}).$$

- Yields Tucker decomposition:

$$\text{vec}(\mathcal{X}) \approx \left(U_3 \otimes U_2 \otimes U_1\right) \cdot \text{vec}(\mathcal{C}).$$

Approximation error governed by truncated singular values of $X^{(1)}, X^{(2)}, X^{(3)}$.

Tucker decomposition

- Consider low-rank decompositions corresponding to $\{x_1\}, \{x_2\}, \{x_3\}$:

$$X^{(1)} \approx U_1 V_1^T$$
$$X^{(2)} \approx U_2 V_2^T$$
$$X^{(3)} \approx U_3 V_3^T$$

- Form $r_1 \times r_2 \times r_3$ core tensor $\mathcal{C}$:

$$\mathrm{vec}(\mathcal{C}) := \left(U_3^T \otimes U_2^T \otimes U_1^T\right) \cdot \mathrm{vec}(\mathcal{X}).$$

- Yields Tucker decomposition:

$$\mathrm{vec}(\mathcal{X}) \approx \left(U_3 \otimes U_2 \otimes U_1\right) \cdot \mathrm{vec}(\mathcal{C}).$$

Approximation error governed by truncated singular values of $X^{(1)}, X^{(2)}, X^{(3)}$.

Need for storing $r_1 \times r_2 \times \cdots \times r_d$ core tensor hurts in high dimensions.

Functions and tensors

Another one of many ways to separate variables of f:

$$f(x_1, x_2, x_3, \ldots, x_d) \approx \sum_{k=1}^{r} g_k(x_1, x_2) h_k(x_2, \ldots, x_d)$$

Corresponding matrix/tensor decomposition?

More general matricizations

Separation wrt $\{x_1, x_2\}$ corresponds to low-rank matrix approximation

$$X^{(1,2)} \approx U_{12} V_{12}^T$$

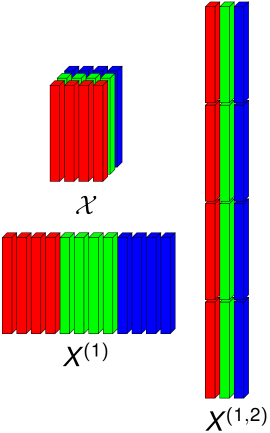
for $(1,2)$ matricization of $\mathcal{X}$.

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

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

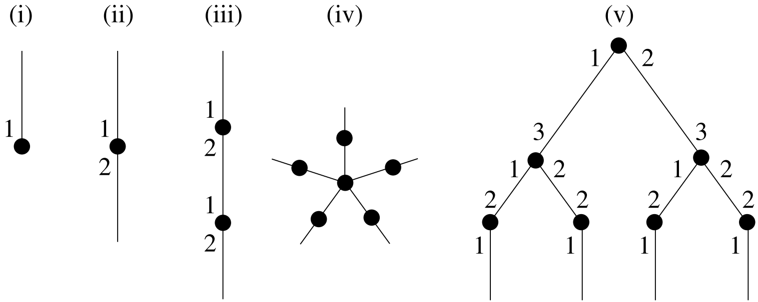
with

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



Tensor network diagrams

Examples:



(i) vector;

(ii) matrix;

(iii) matrix-matrix multiplication;

(iv) Tucker decomposition;

(v) hierarchical Tucker decomposition.

Hierarchical construction

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

Column spaces are nested $\rightsquigarrow$

$$
\begin{aligned}
t = t_1 \cup t_2 \quad &\Rightarrow \quad \operatorname{span}(U_t) \subset \operatorname{span}(U_{t_2} \otimes U_{t_1}) \\
&\Rightarrow \quad \exists B_t : \; U_t = (U_{t_2} \otimes U_{t_1}) B_t.
\end{aligned}
$$

Size of U_t:

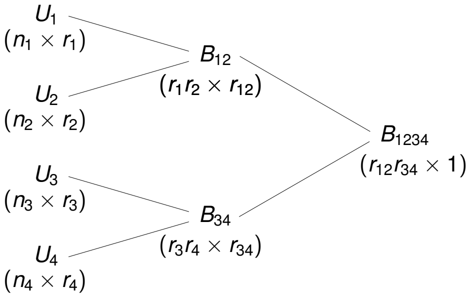
$$
U_t \in \mathbb{R}^{n_{t_1} \cdots n_{t_k} \times r_t} \quad \text{with} \quad r_t = \operatorname{rank}\big(X^{(t)}\big).
$$

For $d = 4$:

$$
\begin{aligned}
U_{12} &= (U_2 \otimes U_1) B_{12} \\
U_{34} &= (U_4 \otimes U_3) B_{34} \\
\operatorname{vec}(\mathcal{X}) = X^{(1234)} &= (U_{34} \otimes U_{12}) B_{1234} \\
\Rightarrow \quad \operatorname{vec}(\mathcal{X}) &= (U_4 \otimes U_3 \otimes U_2 \otimes U_1)(B_{34} \otimes B_{12}) B_{1234}.
\end{aligned}
$$

Dimension tree

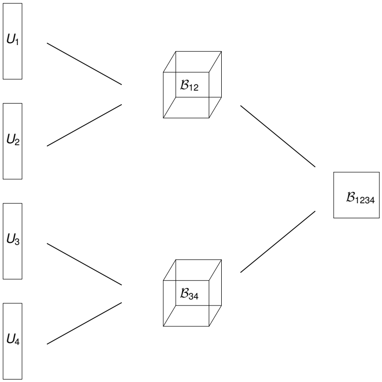
Tree structure for $d = 4$:



Reshape:

$$\begin{aligned}
B_{12} \in \mathbb{R}^{r_1 r_2 \times r_{12}} \quad &\Rightarrow \quad \mathcal{B}_{12} \in \mathbb{R}^{r_1 \times r_2 \times r_{12}} \\
B_{34} \in \mathbb{R}^{r_3 r_4 \times r_{34}} \quad &\Rightarrow \quad \mathcal{B}_{34} \in \mathbb{R}^{r_3 \times r_4 \times r_{34}} \\
B_{1234} \in \mathbb{R}^{r_{12} r_{34} \times 1} \quad &\Rightarrow \quad \mathcal{B}_{1234} \in \mathbb{R}^{r_{12} \times r_{34}}
\end{aligned}$$

Dimension tree



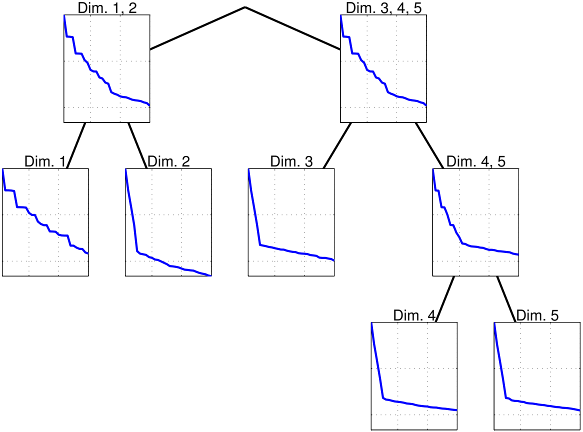
- Often, U_1, U_2, U_3, U_4 are orthonormal. This is advantageous but not required.
- Storage requirements for general d:

$$\mathcal{O}(dnr) + \mathcal{O}(dr^3),$$

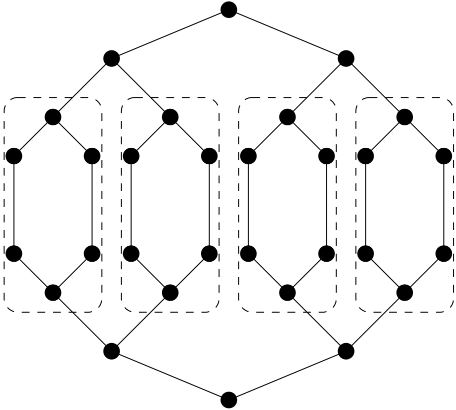
where $r = max\{r_t\}$, $n = \max\{n_\mu\}$.

Singular value tree

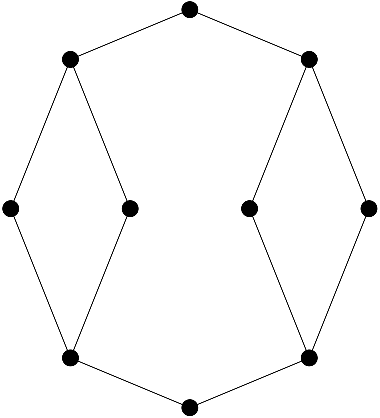
Example: Singular value tree of solution to elliptic PDE with 4 parameters.



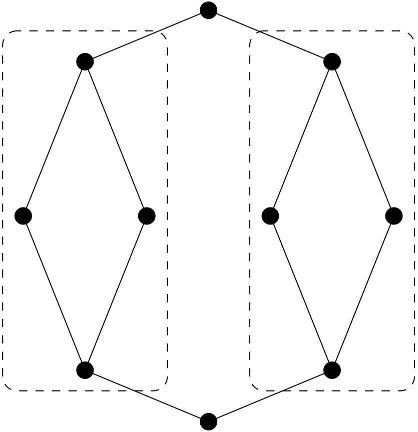
Computation of inner products



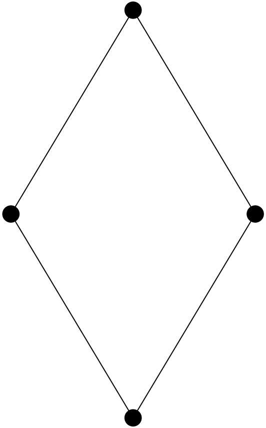
Computation of inner products



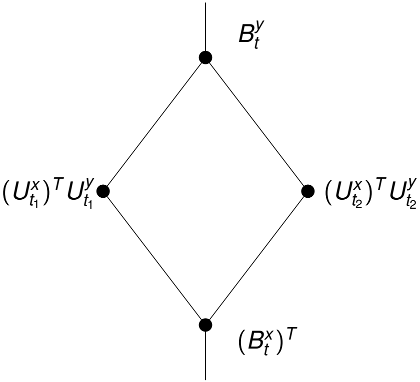
Computation of inner products



Computation of inner products



Computation of inner products – contraction step



$$(U_t^x)^T U_t^y = (B_t^x)^T \big((U_{t_2}^x)^T U_{t_2}^y \otimes (U_{t_1}^x)^T U_{t_1}^y\big) B_t^y .$$

- `htucker` command: `innerprod(x,y)`
- Overall cost: $\mathcal{O}(dnr^2) + \mathcal{O}(dr^4)$.

Hierarchical Tucker decomposition

- Simulation of quantum many-body systems: tree tensor networks [Shi/Duan/Vidal'2006].
- Numerical analysis: [Hackbusch/Kühn'2009], [Grasedyck'2010].
- MATLAB toolbox from `http://anchp.epfl.ch/htucker`
- For fixed r: Storage linear in d!
- When to expect good low-rank approximations?

Hierarchical Tucker decomposition

- When to expect good low-rank approximations?
- Approximation error from separation wrt to $\{x_1, \dots, x_a\}$:

$$f(x_1, \dots, x_a, x_{a+1}, \dots, x_d) \approx \sum_{k=1}^{r} g_k(x_1, \dots, x_a) h_k(x_{a+1}, \dots, x_d)$$

  for $a = 1, \dots, d-1$.
- [DK/Tobler'2011]: For analytic functions

$$\text{error} \lesssim \exp(-r^{\max\{1/a, 1/(d-a)\}}).$$

- [Temlyakov'1992, Uschmajew/Schneider'2013]: For $f \in B^{s,\text{mix}}$

$$\text{error} \lesssim r^{-2s} (\log r)^{2s(\max\{a, d-a\}-1)}.$$

Smoothness is neither sufficient nor necessary for high dimensions!

Hierarchical Tucker decomposition

- When to expect good low-rank approximations?
- Approximation error from separation wrt to $\{x_1, \ldots, x_a\}$:

$$f(x_1, \ldots, x_a, x_{a+1}, \ldots, x_d) \approx \sum_{k=1}^{r} g_k(x_1, \ldots, x_a) h_k(x_{a+1}, \ldots, x_d)$$

  for $a = 1, \ldots, d-1$.
- [DK/Tobler'2011]: For analytic functions

$$\text{error} \lesssim \exp(-r^{\max\{1/a, 1/(d-a)\}}).$$

- [Temlyakov'1992, Uschmajew/Schneider'2013]: For $f \in B^{s,\text{mix}}$

$$\text{error} \lesssim r^{-2s} (\log r)^{2s(\max\{a, d-a\}-1)}.$$

Smoothness is neither sufficient nor necessary for high dimensions!

Low-rank methods for 2D

Rayleigh quotients wrt low-rank matrices

Consider symmetric $n^2 \times n^2$ matrix $\mathcal{A}$. Then

$$\lambda_{\min}(\mathcal{A}) = \min_{x \neq 0} \frac{\langle x, \mathcal{A}x \rangle}{\langle x, x \rangle}.$$

We now...

- reshape vector x into $n \times n$ matrix X;
- reinterpret $\mathcal{A}x$ as linear operator $\mathcal{A} : X \mapsto \mathcal{A}(X)$;
- for example if $\mathcal{A} = \sum_{k=1}^{s} B_k \otimes A_k$ then

$$\mathcal{A}(X) = \sum_{k=1}^{s} B_k X A_k^T.$$

Rayleigh quotients wrt low-rank matrices

Consider symmetric $n^2 \times n^2$ matrix $\mathcal{A}$. Then

$$\lambda_{\min}(\mathcal{A}) = \min_{X \neq 0} \frac{\langle X, \mathcal{A}(X) \rangle}{\langle X, X \rangle}$$

with matrix inner product $\langle \cdot, \cdot \rangle$. We now...

- restrict X to low-rank matrices.

Rayleigh quotients wrt low-rank matrices

Consider symmetric $n^2 \times n^2$ matrix $\mathcal{A}$. Then

$$\lambda_{\min}(\mathcal{A}) \approx \min_{X = UV^T \neq 0} \frac{\langle X, \mathcal{A}(X) \rangle}{\langle X, X \rangle}.$$

- Approximation error governed by low-rank approximability of X.
- Solved by Riemannian optimization techniques or ALS.

ALS

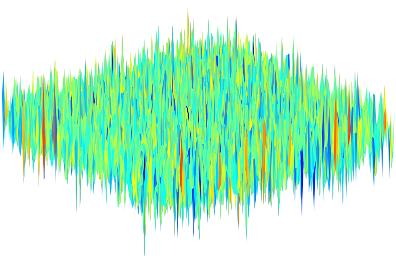
ALS for solving

$$\lambda_{\min}(\mathcal{A}) \approx \min_{X=UV^T \neq 0} \frac{\langle X, \mathcal{A}(X) \rangle}{\langle X, X \rangle}.$$

Initially:

- fix target rank r
- $U \in \mathbb{R}^{m \times r}$, $V^{n \times r}$ randomly, such that V is ONB

$\tilde{\lambda} - \lambda = 6 \times 10^3$
residual $= 3 \times 10^3$



ALS

ALS for solving

$$\lambda_{\min}(\mathcal{A}) \approx \min_{X=UV^T \neq 0} \frac{\langle X, \mathcal{A}(X)\rangle}{\langle X, X\rangle}.$$

Fix V, optimize for U.

$$\begin{aligned} \langle X, \mathcal{A}(X)\rangle &= \text{vec}(UV^T)^T \mathcal{A}\, \text{vec}(UV^T) \\ &= \text{vec}(U)^T (V \otimes I)^T \mathcal{A} (V \otimes I) \text{vec}(U) \end{aligned}$$

$\rightsquigarrow$ Compute smallest eigenvalue of reduced matrix ($rn \times rn$) matrix

$$(V \otimes I)^T \mathcal{A} (V \otimes I).$$

Note: Computation of reduced matrix benefits from Kronecker structure of $\mathcal{A}$.

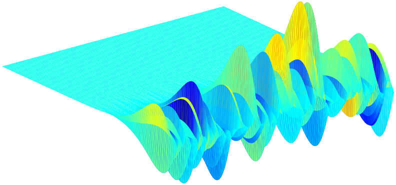
ALS

ALS for solving

$$\lambda_{\min}(\mathcal{A}) \approx \min_{X=UV^T \neq 0} \frac{\langle X, \mathcal{A}(X) \rangle}{\langle X, X \rangle}.$$

Fix V, optimize for U.

$\tilde{\lambda} - \lambda = 2 \times 10^3$
residual $= 2 \times 10^3$



ALS

ALS for solving

$$\lambda_{\min}(\mathcal{A}) \approx \min_{X=UV^T\neq 0} \frac{\langle X, \mathcal{A}(X)\rangle}{\langle X, X\rangle}.$$

Orthonormalize U, fix U, optimize for V.

$$\begin{aligned}\langle X, \mathcal{A}(X)\rangle &= \mathrm{vec}(UV^T)^T \mathcal{A}\, \mathrm{vec}(UV^T) \\ &= \mathrm{vec}(V^T)(I \otimes U)^T \mathcal{A}(I \otimes U)\mathrm{vec}(V^T)\end{aligned}$$

$\rightsquigarrow$ Compute smallest eigenvalue of reduced matrix ($rn \times rn$) matrix

$$(I \otimes U)^T \mathcal{A}(I \otimes U).$$

Note: Computation of reduced matrix benefits from Kronecker structure of $\mathcal{A}$.

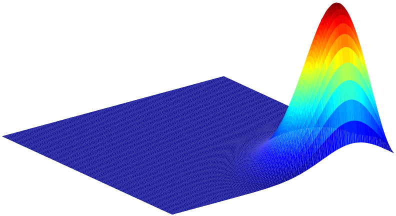
ALS

ALS for solving

$$\lambda_{\min}(\mathcal{A}) \approx \min_{X=UV^T \neq 0} \frac{\langle X, \mathcal{A}(X) \rangle}{\langle X, X \rangle}.$$

Orthonormalize U, fix U, optimize for V.

$\tilde{\lambda} - \lambda = 1.5 \times 10^{-7}$
residual $= 7.7 \times 10^{-3}$



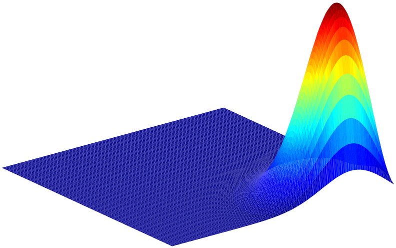
ALS

ALS for solving

$$\lambda_{\min}(\mathcal{A}) \approx \min_{X=UV^T \neq 0} \frac{\langle X, \mathcal{A}(X) \rangle}{\langle X, X \rangle}.$$

Orthonormalize V, fix V, optimize for U.

$\tilde{\lambda} - \lambda = 1 \times 10^{-12}$
residual $= 6 \times 10^{-7}$



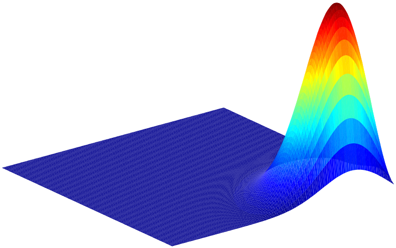
ALS

ALS for solving

$$\lambda_{\min}(\mathcal{A}) \approx \min_{X=UV^T\neq 0} \frac{\langle X, \mathcal{A}(X)\rangle}{\langle X, X\rangle}.$$

Orthonormalize U, fix U, optimize for V.

$\tilde{\lambda} - \lambda = 7.6 \times 10^{-13}$
residual $= 7.2 \times 10^{-8}$



Low-rank methods for arbitrary dimensions

ALS

Originally from computational quantum physics [Schollwöck 2011] for matrix product states.

Goal:

$$\min\left\{\frac{\langle \mathcal{X}, \mathcal{A}(\mathcal{X})\rangle}{\langle \mathcal{X}, \mathcal{X}\rangle} : \mathcal{X} \in \mathcal{H}\text{-Tucker}\big((r_t)_{t\in\mathcal{T}}\big),\ \mathcal{X} \neq 0\right\}$$

Method: Choose one node t, fix all other nodes, set new tensor at node t to minimize Rayleigh quotient $\frac{\langle \mathcal{X}, \mathcal{A}(\mathcal{X})\rangle}{\langle \mathcal{X}, \mathcal{X}\rangle}$. This is done for all nodes (a sweep), and sweeps are continued until convergence.

Sketch:

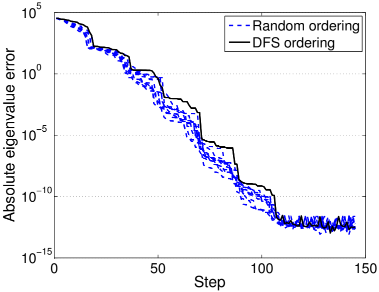
$$X^{(t)} = U_t V_t^T = \big(U_{t_r} \otimes U_{t_l}\big) B_t V_t^T,$$

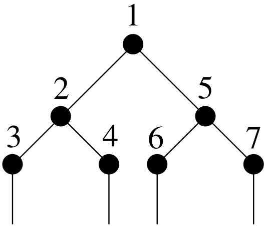
$$\mathrm{vec}(\mathcal{X}) = \big(V_t \otimes U_{t_r} \otimes U_{t_l}\big)\mathrm{vec}(B_t) = \mathcal{U}_t\ \mathrm{vec}(B_t).$$

$$\Rightarrow \min\left\{\frac{y^T(\mathcal{U}_t^T \mathcal{A}\,\mathcal{U}_t)y}{y^T(\mathcal{U}_t^T\mathcal{U}_t)y} : y \in \mathbb{R}^{r_{t_l} r_{t_r} r_t},\ y \neq 0\right\}.$$

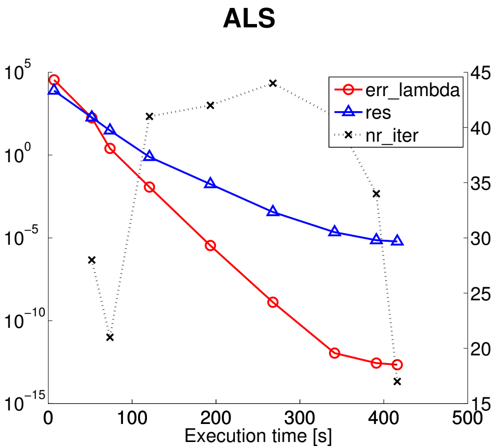
ALS - Comments

Ordering of a sweep In principle, nodes of tensor can be traversed in any ordering. Experimentally, makes little difference. Depth-first-search ordering allows data reuse in the computation of the reduced eigenvalue problems.



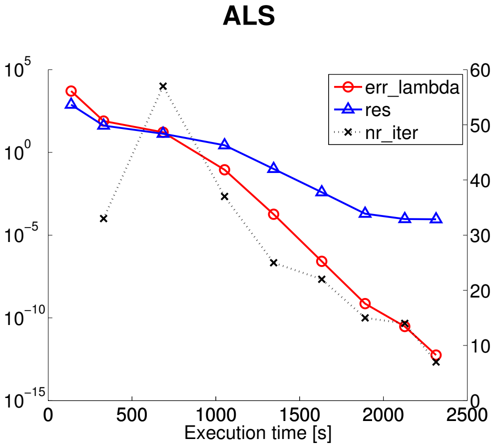


Numerical Experiments - Sine potential, $d = 10$



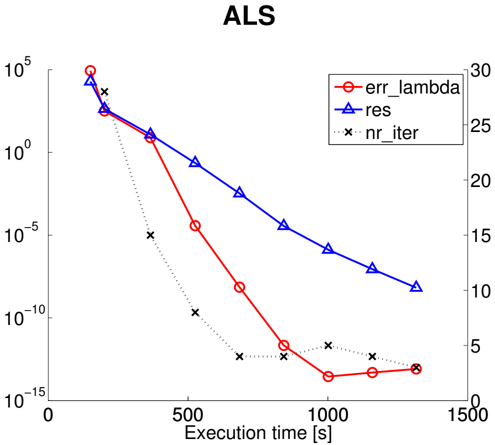
Hierarchical ranks 40.

Numerical Experiments - Henon-Heiles, $d = 20$



Hierarchical rank 40.

Numerical Experiments - $1/\|\xi\|_2$ potential, $d = 20$



Hierarchical rank 30.

Outlook

Outlook: Low-rank tensor completion

Setting:

- Consider tensor $\mathcal{X}$ with very few entries known, described by linear projection P_Ω.
- Assume low (multilinear) rank model for $\mathcal{X}$.

Low-rank reconstruction:

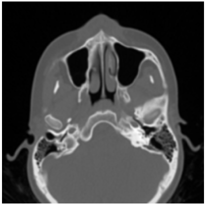
$$\begin{aligned} \min_{\mathcal{X}} \quad & \frac{1}{2}\|\mathrm{P}_\Omega \mathcal{X} - \text{known entries}\|^2 \\ \text{subject to} \quad & \mathcal{X} \in \mathcal{M}_k := \{\mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d} : \mathrm{rank}(\mathcal{X}) = \mathbf{k}) \end{aligned}$$

- $\mathcal{M}_k$ is a smooth manifold.
- Becomes Riemannian with metric induced by standard inner product.
- Allows to apply general Riemannian optimization techniques [Absil, Mahony and Sepulchre'05].
- Adaption of nonlinear conjugate gradient method in [DK/Steinlechner/Vandereycken'12].

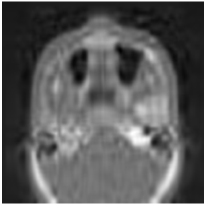
Outlook: Reconstruction of CT Scan

$199 \times 199 \times 150$ tensor from MRI/CT data base “INCISIX”.

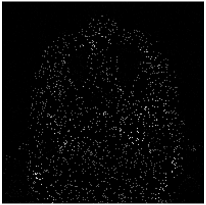
Slice of original tensor



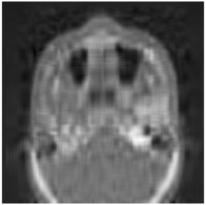
HOSVD approx. of rank 21



Sampled tensor (6.7%)



Low-rank completion of rank 21



Compares very well with existing results wrt low-rank recovery and speed, e.g., Gandy/Recht/Yamada/’2011: Tensor completion and low-n-rank tensor recovery via convex optimization.

Conclusions

Conclusions and Outlook

- Scientific computing with low-rank tensors rapidly evolving field and highly technical.
- Low-rank tensors capable of solving certain high-dimensional eigenvalue problems.
- Precise scope of applications far from clear; many applications remain to be explored. More analysis needed!

Some current trends:

- Tensorization of vectors $+$ low rank (discrete Chebfun?) by Hackbusch, Khoromskij, Oseledets, Tyrtishnikov, ...
- Computational differential geometry on low-rank tensor manifolds by Koch, Lubich, Schneider, Uschmajew, Vandereycken, ...
- Robust low rank (Candes et al.) for tensors $\rightsquigarrow$ suitable way of dealing with singularities?
- ...

Workshop on Matrix Equations and Tensor Techniques
EPF Lausanne, October 10th - 11th 2013

©Alain Herzog

http://anchp.epfl.ch/MatrixEquations

Organizers: P. Benner, H. Faßbender, L. Grasedyck, D. Kressner.