TENSORS AND COMPUTATIONS

Eugene Tyrtyshnikov

Institute of Numerical Mathematics of Russian Academy of Sciences

eugene.tyrtyshnikov@gmail.com

20 November 2013

REPRESENTATION PROBLEM FOR MULTI-INDEX ARRAYS

Going to consider an array $a(i_1, \ldots, i_d)$ of size

$$\underbrace{n \times \ldots \times n}_{d \text{ times}}.$$

We have no hope to store all n^d elements.

For any practical computation we need *special structure* and *condensed representations* of d-arrays.

CANONICAL POLYADIC DECOMPOSITION

Hitchcock (1927)

$$A(i_1, \ldots, i_d) = \sum_{\alpha=1}^{R} U_1(i_1, \alpha) \ldots U_d(i_d, \alpha)$$

Minimal R is called *tensor rank* of A.

Instead of n^d elements of a tensor of shape $n \times \ldots \times n$ we store only dnR parameters.

CANONICAL POLYADIC DECOMPOSITION

- Used as a model for data. E.g. in the fluorescence spectrometry. Given n_1 samples of solutions, find the number of chemicals inside each and concentrations. The data is a $n_1 \times n_2 \times n_3$ array, where n_2 and n_3 match emission and excitation wavelengths. The canonical decomposition rank reveals the number of chemicals.
- Used in the complexity theory as a basic tool for computation of bilinear forms (Strassen, Pan, Bini and others).

УМНОЖЕНИЕ 2 $\times$ 2-МАТРИЦ

Правило "строка на столбец" дает 8 умножений:

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{bmatrix}$$

АЛГОРИТМ ШТРАССЕНА ДЛЯ БЫСТРОГО УМНОЖЕНИЯ МАТРИЦ

В 1969 Штрассен (Strassen) открыл алгоритм, в котором всего 7 умножений. При умножении блочных 2×2-матриц алгоритм Штрассена содержит 7 умножений блоков:

$$\begin{aligned}
\alpha_1 &= (a_{11} + a_{22})(b_{11} + b_{22}) \\
\alpha_2 &= (a_{21} + a_{22})b_{11} \\
\alpha_3 &= a_{11}(b_{12} - b_{22}) \\
\alpha_4 &= a_{22}(b_{21} - b_{11}) \\
\alpha_5 &= (a_{11} + a_{12})b_{22} \\
\alpha_6 &= (a_{21} - a_{11})(b_{11} + b_{12}) \\
\alpha_7 &= (a_{12} - a_{22})(b_{21} + b_{22})
\end{aligned}$$

$$\begin{aligned}
c_{11} &= \alpha_1 + \alpha_4 - \alpha_5 + \alpha_7 \\
c_{12} &= \alpha_3 + \alpha_5 \\
c_{21} &= \alpha_2 + \alpha_4 \\
c_{22} &= \alpha_1 + \alpha_3 - \alpha_2 + \alpha_6
\end{aligned}$$

ЧТО ДАЕТ КАНОНИЧЕСКОЕ ТЕНЗОРНОЕ

$$\begin{bmatrix} c_1 & c_2 \\ c_3 & c_4 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \begin{bmatrix} b_1 & b_2 \\ b_3 & b_4 \end{bmatrix} \qquad c_k = \sum_{i=1}^{n^2} \sum_{j=1}^{n^2} h_{ijk} \; a_i \; b_j$$

$$\boldsymbol{h_{ijk} = \sum_{\alpha=1}^{R} u_{i\alpha} \; v_{j\alpha} \; w_{k\alpha}}$$

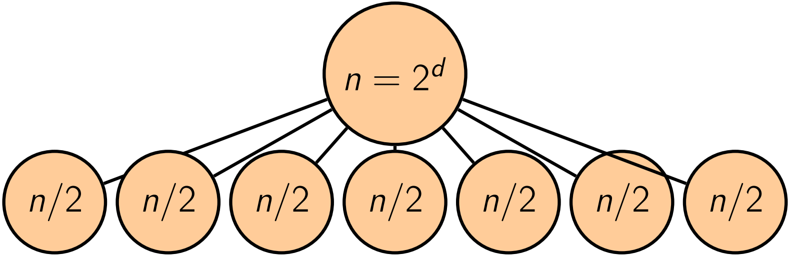
$$\Rightarrow \quad c_k = \sum_{\alpha=1}^{R} w_{k\alpha} \left(\sum_{i=1}^{n^2} u_{i\alpha} a_i \right) \left(\sum_{j=1}^{n^2} v_{j\alpha} b_j \right)$$

Теперь лишь R умножений блоков! Если $n = 2$, то $R = 7$ (Strassen, 1969).

Рекурсия $\Rightarrow$ $O(n^{\log_2 7})$ умножений для произвольного n.

РЕКУРСИЯ ДЛЯ БЫСТРОГО УМНОЖЕНИЯ МАТРИЦ

Рекурсия $\Rightarrow$ $O(n^{\log_2 7})$ умножений для произвольного n.



CAN WE FIND TENSOR RANK IN FINITELY MANY OPERATIONS?

Consider equations

$$A(i_1 \dots i_d) = \sum_{\alpha=1}^{R} U_1(i_1, \alpha) \dots U_d(i_d, \alpha)$$

$$1 \leqslant i_1 \leqslant n_1, \quad \dots, \quad 1 \leqslant i_d \leqslant n_d$$

$$\Leftrightarrow \quad f_1 = 0, \ \dots, \ f_m = 0$$

$f_1, \dots, f_m$ – polynomials of p variables

$$p := (n_1 + \dots + n_d)R, \qquad m := n_1 \dots n_d.$$

REMEMBER NULLSTELLENSATZ

The system

$$f_1 = 0, \quad \dots \quad f_m = 0 \tag{1}$$

is inconsistent if there exist polynomials $g_1, \dots, g_m$ such that

$$f_1\, g_1 + \dots + f_m\, g_m = 1. \tag{2}$$

Nullstellensatz: (2) $\Leftrightarrow$ inconsistensy of (1).

Note that (2) is a system of linear algebraic equations!

COMPUTATION OF TENSOR RANK

1. R := 1
2. Check inconsistensy.
   Quit if consistent.
3. Set R := R+1 and repeat.

On exit $R =$ tensor rank.

But the degrees of $g_1, \ldots, g_m$ may be too high!

MINIMALITY CONDITIONS

DEFINITION.
Kruskal rank $k(A)$ of the columns of A is the maximal number k s.t. *any* k columns of A are linearly independent.

Canonical decomposition is determined by matrices

$$U_1 := [U_1(i_1,\, \alpha)], \quad \dots, \quad U_d := [U_d(i_d,\, \alpha)].$$

Each has R columns.

SUFFICIENT MINIMALITY CONDITIONS

KRUSKAL THEOREM.
Assume that

$$k(U_1) + \ldots + k(U_d) \geqslant 2R + d - 1.$$

Then R is the tensor rank and the decomposition is unique.

WEAKER CONDITIONS

THEOREM (Domanov & De Lathauwer, 2013).
Let $d = 3$ and assume that

$$k(U_1) + r(U_2) + r(U_3) \geqslant 2R + 2,$$

$$r(U_1) + k(U_2) + r(U_3) \geqslant 2R + 2,$$

$$r(U_1) + r(U_2) + k(U_3) \geqslant 2R + 2.$$

Then R is the tensor rank and the decomposition is unique.

EVEN WEAKER CONDITIONS

THEOREM (Domanov & De Lathauwer, 2013).
Let $d = 3$ and assume that

$$r(U_1) + k(U_2) \geqslant R + m, \quad k(U_1) \geqslant m, \quad k(U_3) \geqslant 1,$$

where $m := R - r(U_3) + 2$. Then R is the tensor rank.

RANK OF d-DIMENSIONAL LAPALCIAN

Let $a, b \in \mathbb{C}^2$ be linearly independent.

$$A := A_1 + \ldots + A_d$$

$$A_i = \underbrace{a \otimes \ldots \otimes a}_{i-1 \text{ terms}} \otimes b \otimes \underbrace{a \otimes \ldots \otimes a}_{d-i \text{ terms}}$$

Kruskal does not work!

CASES OF DIMENSIONALITY 2 AND 3

We do not lose generality by taking

$$a = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \cdots \\ 0 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \cdots \\ 0 \end{bmatrix}.$$

The task reduces to the case $2 \times \ldots \times 2$.

- $d = 2$: tensor rank = matrix rank.
- $d = 3$: simultaneous diagonalization of slices.

GENERAL CASE

TEOPEMA.
Tensor rank of the d-dimensional Laplacian is equal to d.

GENERIC RANK FOR COMPLEX TENSORS

NOTATION.

$X_k :=$ the set of $m \times n \times q$-tensors of rank $\leqslant k$.

The closure $\overline{X}_k$ is an algebraic variety. There exists $g = \mathrm{grank}(m, n, q)$ s.t.

$$\overline{X}_1 \subsetneq \ldots \subsetneq \overline{X}_{g-1} \subsetneq \overline{X}_g = \mathbb{C}^{m \times n \times q}$$

This number g is called generic rank of tensors of shape $m \times n \times q$.

FORMULA FOR GENERIC RANK

CONJECTURE (Friedland). Under the conditions

$$3 \leqslant m \leqslant n \leqslant q \leqslant (m-1)(n-1),$$

$$(m,n,q) \neq (3, 2p+1, 2p+1)$$

we have

$$\mathrm{grank}(m,n,q) = \left\lceil \frac{mnq}{m+n+q-2} \right\rceil$$

Strassen (1983):
$\mathrm{grank}(3, 2p+1, 2p+1) = \left\lceil \frac{3(2p+1)^2}{4p+3} \right\rceil + 1.$

UNCLOSENESS OF THE SETS X_k

$$\underbrace{b \otimes a \otimes a + a \otimes b \otimes a + a \otimes a \otimes b}_{\text{rank} = 3} =$$

$$\underbrace{\frac{1}{\varepsilon}(a + \varepsilon b) \otimes (a + \varepsilon b) \otimes (a + \varepsilon b) \; - \; \frac{1}{\varepsilon} a \otimes a \otimes a}_{\text{rank} = 2} + O(\varepsilon)$$

Hence $X_2 \neq \overline{X}_2$.

What about other values $1 < k < \mathrm{grank}$?

Solution is eveident if we knew that for any tensor A of rank $< \mathrm{grank}(m, n, q)$ there is a tensor B of rank 1 s.t.

$$\mathrm{rank}(A + B) = \mathrm{rank}\, A + 1.$$

CANONICAL DECOMPOSITION IS DIFFICULT TO COMPUTE

No special techniques like those in the matrix case.
Fall back to general optimization methods.

What can replace the canonical decompostion in computations?

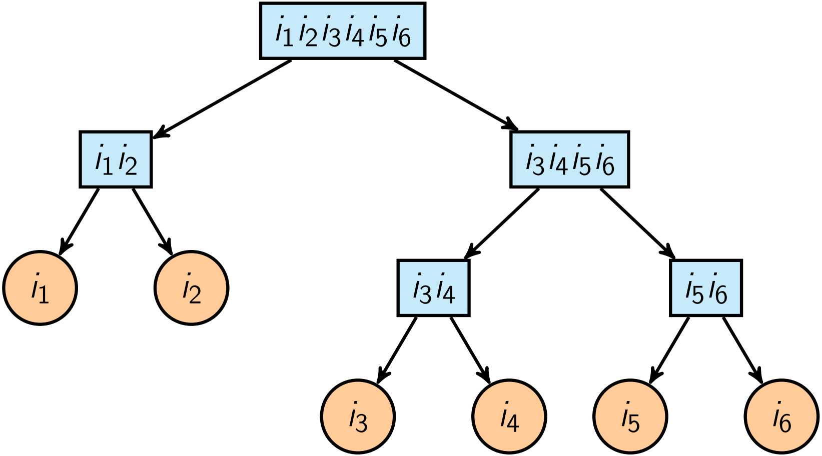
NEW REPRESENTATION FORMATS

New decompositions `plus` algorithms:

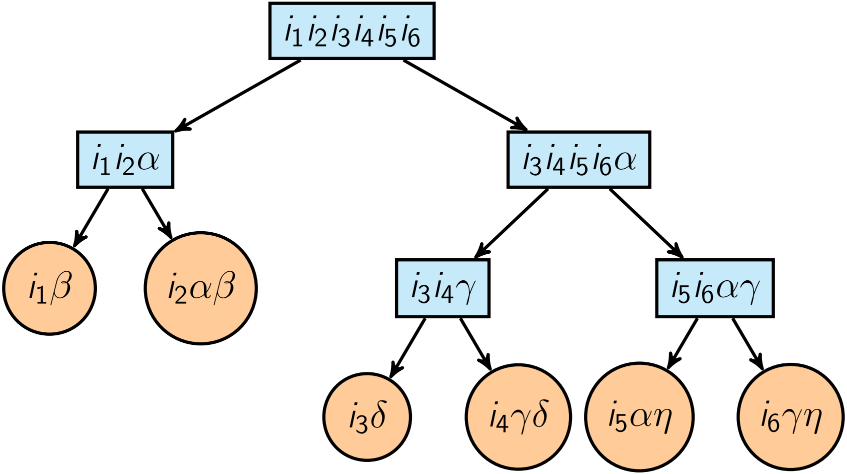
- TT (Tensor Train)
  Moscow: Oseledets, Tyrtyshnikov
- HT (Hierarchical Tucker)
  Leipzig: Hackbusch, Grasedyck

Nothing new is new ...
(tensor networks in spin systems)

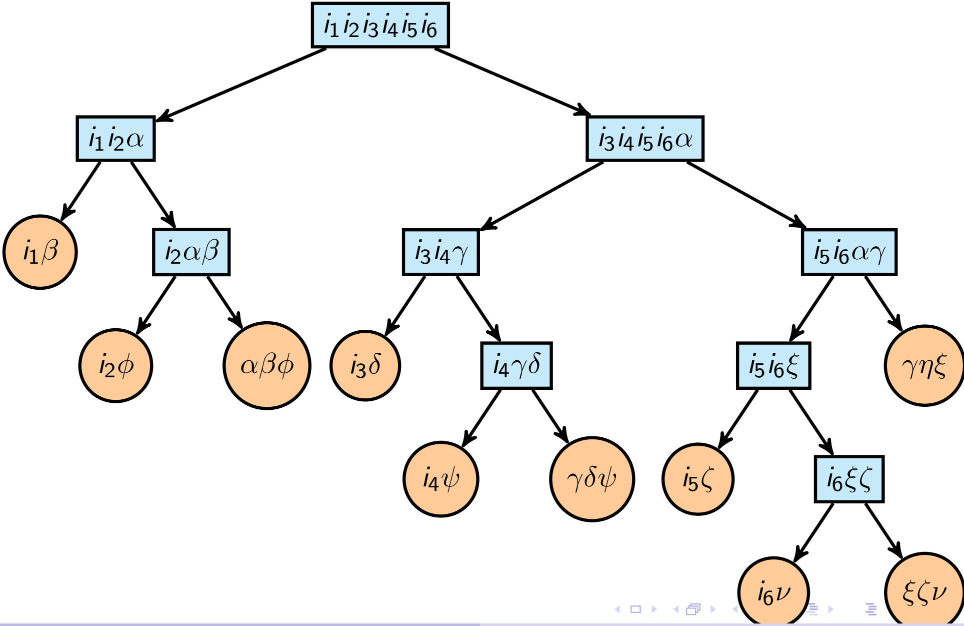
REDUCTION OF DIMENSIONALITY



SCHEME FOR TT



SCHEME FOR HT



HOW RANK-ONE DECOMPOSITION BECOMES TENSOR TRAIN

Consider a rank-one separation of variables

$$a(i_1, i_2, i_3) = g_1(i_1)\; g_2(i_2)\; g_3(i_3).$$

Now, consider $g_1(i_1)$, $g_2(i_2)$, $g_3(i_3)$ as matrices of agreed sizes so that the product is a scalar. Then

$$a(i_1, i_2, i_3) = \sum_{\alpha_1=1}^{r_1} \sum_{\alpha_2=1}^{r_2} g_1(i_1, \alpha_1)\, g_2(\alpha_1, i_2, \alpha_2)\, g_3(\alpha_2, i_3).$$

TENSOR TRAIN (TT) DECOMPOSITION

$$a(i_1, \ldots, i_d) = \sum \prod_{k=1}^{d} g_k(\alpha_{k-1}, i_k, \alpha_k)$$

Assume summation over repeated indices.

$1 \leqslant i_k \leqslant n_k$ for $1 \leqslant k \leqslant d$

$1 \leqslant \alpha_k \leqslant r_k$ for $0 \leqslant k \leqslant d$ and $r_0 = r_d = 1$

r_k are called TT ranks

2D TENSOR TRAIN EXAMPLE

$$a(i_1,\ i_2) = \sum g_1(i_1,\ \alpha_1)\, g_2(\alpha_1,\ i_2)$$

This is the skeleton (dyadic) decomposition of a matrix!

$$A = G_1\, G_2$$

$$A \ \text{ is } \ n_1 \times n_2, \quad G_1 \ \text{ is } \ n_1 \times r_1, \quad G_2 \ \text{ is } \ r_1 \times n_2$$

$$r_1 \geqslant \operatorname{rank} A$$

3D TENSOR TRAIN EXAMPLE

$$a(i_1,\ i_2,\ i_3) = \sum g_1(i_1,\ \alpha_1)\, g_2(\alpha_1,\ i_2,\ \alpha_2)\, g_3(\alpha_2,\ i_3)$$

TENSOR TRAIN IS EASY TO GET

For a 3-tensor we need two skeleton (dyadic) decompositions for associated unfolding matrices:

- $a(i_1,\ i_2 i_3) = \sum g_1(i_1,\ \alpha_1)\, a_1(\alpha_1,\ i_2 i_3)$
- $a_1(\alpha_1 i_2\ ,\ i_3) = \sum g_2(\alpha_1 i_2\ ,\ \alpha_2)\, g_3(\alpha_2,\ i_3)$

For a d-tensor we need $d-1$ skeleton (dyadic) decompositions.

IF WE APPROXIMATE USING SVD THEN LOCAL ERROR IN EACH SKELETON DECOMPOSITION DOES NOT BLOW UP

THEOREM.

If the Frobenius-norm error for kth skeleton decompostion is ε_k, then the overall error E is upper bounded by

$$E \leqslant \sqrt{\sum_{k=1}^{d-1} \varepsilon_k^2}.$$

I. Oseledets, E. Tyrtyshnikov, TT-cross approximation for multidimensional arrays, Linear Algebra Appl., 432 (2010), pp. 70–88.

EXAMPLES OF QUANTIZATION

$f(x)$ is a function on $[0, 1]$

$$a(i_1, \dots, i_d) = f(ih), \quad i = \frac{i_1}{2} + \frac{i_2}{2^2} + \dots + \frac{i_d}{2^d}$$

The array of values of f is viewed as a tensor of size $2 \times \dots \times 2$.

`EXAMPLE 1.` $f(x) = e^x + e^{2x} + e^{3x}$

`ttrank= 2.7      ERROR=1.5e-14`

`EXAMPLE 2.` $f(x) = 1 + x + x^2 + x^3$

`ttrank= 3.4      ERROR=2.4e-14`

`EXAMPLE 3.` $f(x) = 1/(x - 0.1)$

`ttrank= 10.1     ERROR=5.4e-14`

THEOREMS

If there is an ε-approximation with separated variables

$$f(x+y) \approx \sum_{k=1}^{r} u_k(x) v_k(y), \quad r = r(\varepsilon),$$

then a TT exists with error ε and TT-ranks $\leqslant r$.

If $f(x)$ is a sum of r exponents, then an exact TT exists with the ranks r.

For a polynomial of degree m an exact TT exists with the ranks $r = m+1$.

If $f(x) = 1/(x-\delta)$ then $r = \log \varepsilon^{-1} + \log \delta^{-1}$.

COLUMN-AND-ROW INTERPOLATION OF MATRICES

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \qquad A_{11} \text{ is } r \times r$$

A can be interpolated on the first r columns and rows by

$$\begin{bmatrix} A_{11} \\ A_{21} \end{bmatrix} A_{11}^{-1} \begin{bmatrix} A_{11} & A_{12} \end{bmatrix}$$

COLUMN-AND-ROW INTERPOLATION OF MATRICES

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} - \begin{bmatrix} A_{11} \\ A_{21} \end{bmatrix} A_{11}^{-1} \begin{bmatrix} A_{11} & A_{12} \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 \\ 0 & A_{22} - A_{21}A_{11}^{-1}A_{12} \end{bmatrix}$$

MAXIMAL VOLUME PRINCIPLE

THEOREM (Goreinov, Tyrtyshnikov) *Let*

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix},$$

where A_{11} *is a* $r \times r$ *block with maximal determinant in modulus (volume) among all* $r \times r$ *blocks in* A*. Then the rank-*r *matrix*

$$A_r = \begin{bmatrix} A_{11} \\ A_{21} \end{bmatrix} A_{11}^{-1} \begin{bmatrix} A_{11} & A_{12} \end{bmatrix}$$

approximates A *with the Chebyshev-norm error at most in* $(r+1)^2$ *times larger than the error of best approximation of rank* r*.*

MATRIX CROSS ALGORITHM

- Given *initial* column indices $j_1, \ldots, j_r$.
- Find *good* row indices $i_1, \ldots, i_r$ in these columns.
- Find *good* column indices in the rows $i_1, \ldots, i_r$.
- Proceed choosing good columns and rows until the skeleton cross approximations stabilize.

E.E.Tyrtyshnikov, Incomplete cross approximation in the mosaic-skeleton method, *Computing* 64, no. 4 (2000), 367–380.

TENSOR-TRAIN CROSS ALGORITHM

Let $a_1 = a(i_1, i_2, i_3, i_4)$. Seek crosses in the unfolding matrices.
On input: r initial columns in each. Select *good* rows.

$$A_1 = [a(i_1\,;\; i_2, i_3, i_4)], \quad J_1 = \{i_2^{(\beta_1)} i_3^{(\beta_1)} i_4^{(\beta_1)}\}$$

$$A_2 = [a(i_1, i_2\,;\; i_3, i_4)], \quad J_2 = \{i_3^{(\beta_2)} i_4^{(\beta_2)}\}$$

$$A_3 = [a(i_1, i_2, i_3\,;\; i_4)], \quad J_3 = \{i_4^{(\beta_3)}\}$$

rows	matrix	skeleton decomposition
$I_1 = \{i_1^{(\alpha_1)}\}$	$a_1(i_1\,;\; i_2, i_3, i_4)$	$a_1 = \sum_{\alpha_1} g_1(i_1; \alpha_1)\; a_2(\alpha_1;\; i_2, i_3, i_4)$
$I_2 = \{i_1^{(\alpha_2)} i_2^{(\alpha_2)}\}$	$a_2(\alpha_1, i_2\,;\; i_3, i_4)$	$a_2 = \sum_{\alpha_2} g_2(\alpha_1, i_2;\; \alpha_2)\; a_3(\alpha_2, i_3;\; i_4)$
$I_3 = \{i_1^{(\alpha_3)} i_2^{(\alpha_3)} i_3^{(\alpha_3)}\}$	$a_3(\alpha_2, i_3\,;\; i_4)$	$a_3 = \sum_{\alpha_3} g_3(\alpha_2, i_3;\; \alpha_3)\; g_4(\alpha_3;\; i_4)$

Finally

$$a = \sum_{\alpha_1, \alpha_2, \alpha_3, \alpha_4} g_1(i_1, \alpha_1)\; g_2(\alpha_1, i_2, \alpha_2)\; g_3(\alpha_2, i_3, \alpha_3)\; g_4(\alpha_3, i_4)$$

TENSOR TRAIN FROM CROSSES IN UNFOLDING MATRICES

$$A(i_1 \dots i_d) = \prod_{k=1}^{d} A(J_{\leqslant k-1},\ i_k,\ J_{>k})\,[A(J_{\leqslant k},\ J_{>k})]^{-1}$$

I. Oseledets, E. Tyrtyshnikov, TT-cross approximation for multidimensional arrays, Linear Algebra Appl., 432 (2010), pp. 70–88.

QUASIOPTIMALITY THEOREM FOR TENSOR TRAINS

THEOREM (Savostyanov'2013)

Assume that a d-tensor A is approximated by $\widetilde{A}$ on the maximal volume crosses in the unfolding matrices, and let the error is upper bounded by $\varepsilon\,||A||_C$ in each matrix. Then for sufficiently small ε we have

$$||A - \widetilde{A}||_C \leqslant 2dr\varepsilon||A||_C.$$

TT-QUADRATURE

$$I(d) = \int \sin(x_1 + x_2 + \ldots + x_d)\; dx_1 dx_2 \ldots dx_d =$$

$$\mathrm{Im} \int_{[0,1]^d} e^{i(x_1+x_2+\ldots+x_d)}\; dx_1 dx_2 \ldots dx_d = \mathrm{Im}\big(\big(\frac{e^i - 1}{i}\big)^d\big).$$

$n = 11$ on each dimension $\Rightarrow$ in total n^d values!

Only a small portion of them is still computed.

d	$I(d)$	Rel.error	Time
100	-3.926795e-03	2.915654e-13	0.77
1000	-2.637513e-19	3.482065e-11	11.60
2000	2.628834e-37	8.905594e-12	33.05
4000	9.400335e-74	2.284085e-10	105.49

TT-QUADRATURE

Quasi-Monte-Carlo: 2^{30} quasirandom points (Niederreiter).
TT: 4 Chebyshev nodes on each axis.

Function: $f(x_1, x_2, \dots, x_d) = 3^d \prod_{i=1}^{d} (1 - 2x_i)^2$

Dim.	Error QMC	Time QMC	Error TT	Time TT
10	$3.7 * 10^{-5}$	205.6	0	0.0064
20	$1.2 * 10^{-2}$	403.4	0	0.0087
30	$3.9 * 10^{1}$	601.9	0	0.013
500	$*$	$*$	$2.0 * 10^{-15}$	0.104

ТЕНЗОРНЫЕ ПОЕЗДА В КВАНТОВОЙ ХИМИИ

Большое число измерений естественно возникает в задачах квантовой молекулярной динамики:

$$H\Psi = (-\frac{1}{2}\Delta + V(R_1, \dots, R_f))\Psi = E\Psi,$$

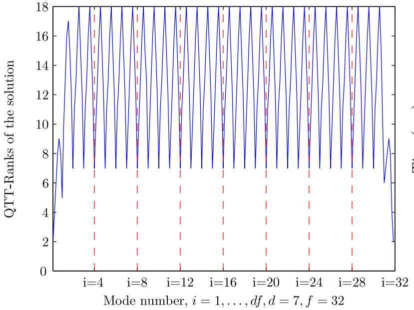
V – заданная поверхность потенциальной энергии (PES).

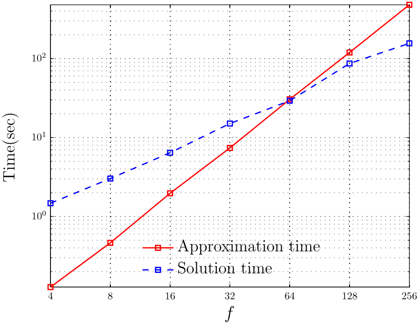
Вычисление V требует решения уравнения Шредингера при разных значениях координат атомов $R_1, \dots, R_f$. Метод ТТ-интерполяции позволяет найти тензорный поезд для V по относительному малому числу значений V.

ТЕНЗОРНЫЕ ПОЕЗДА В КВАНТОВОЙ ХИМИИ

Потенциал Эно-Хайлеса:

$$V(q_1, \ldots, q_f) = \frac{1}{2} \sum_{k=1}^{f} q_k^2 + \lambda \sum_{k=1}^{f-1} \left(q_k^2 q_{k+1} - \frac{1}{3} q_k^3 \right).$$





ТТ-ранги и время решения задачи Эно-Хайлеса (Оселедец–Хоромский).

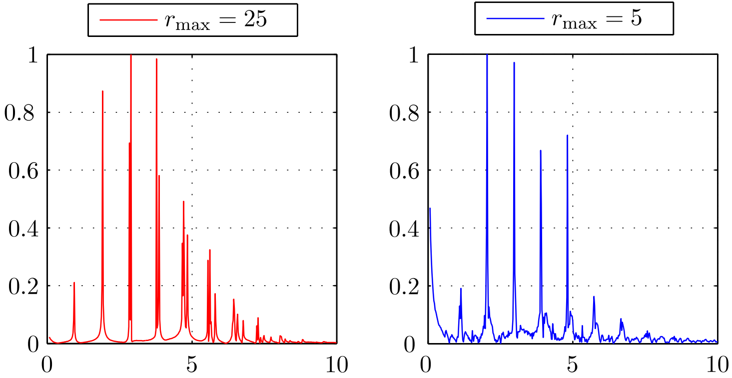
ВЫЧИСЛЕНИЕ СПЕКТРА В ЦЕЛОМ

Вычисление спектра (многих собств. значений) – с помощью эволюции по времени:

$$\frac{\partial \Psi}{\partial t} = iH\Psi, \quad \Psi(0) = \Psi_0.$$

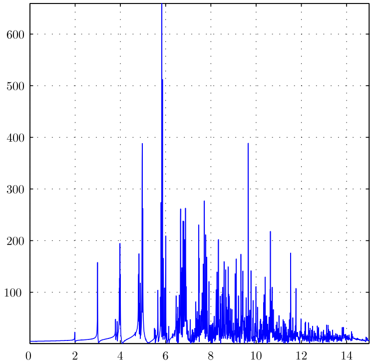
“Физическая схема” имеет вид $\Psi(t) = e^{iHt}\Psi_0$, вычисляется автокорреляционная функция $a(t) = (\Psi(t), \Psi_0)$, а затем спектр – с помощью Фурье-преобразования от $a(t)$.

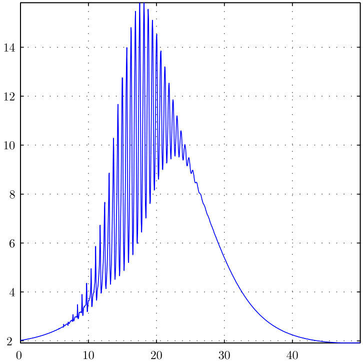
ВЫЧИСЛЕНИЕ СПЕКТРА В ЦЕЛОМ



Спектры для потенциала Эно-Хайлеса при $f = 2$ и различных ТТ-рангах.

ВЫЧИСЛЕНИЕ СПЕКТРА В ЦЕЛОМ





Спектры для потенциала Эно-Хайлеса при $f = 4$ и $f = 10$.

ТЕНЗОРНЫЕ ПОЕЗДА ДЛЯ УРАВНЕНИЙ С ПАРАМЕТРАМИ

Уравнение диффузии на квадрате $[0,1]^2$. Область разбита на $p \times p$ квадратов с постоянными коэффициентами диффузии, p^2 параметров со значениями от 0.1 до 1.

256 точек по каждому из параметров,
пространственная сетка 256×256.
Решение *при всех значениях параметров* приближается тензорным поездом с относительной точностью 10^{-5}:

Число параметров	Память на хранение решения
4	8 Mb
16	24 Mb
64	78 Mb

ТЕНЗОРНЫЕ ПОЕЗДА В ДИСКРЕТНОЙ ОПТИМИЗАЦИИ

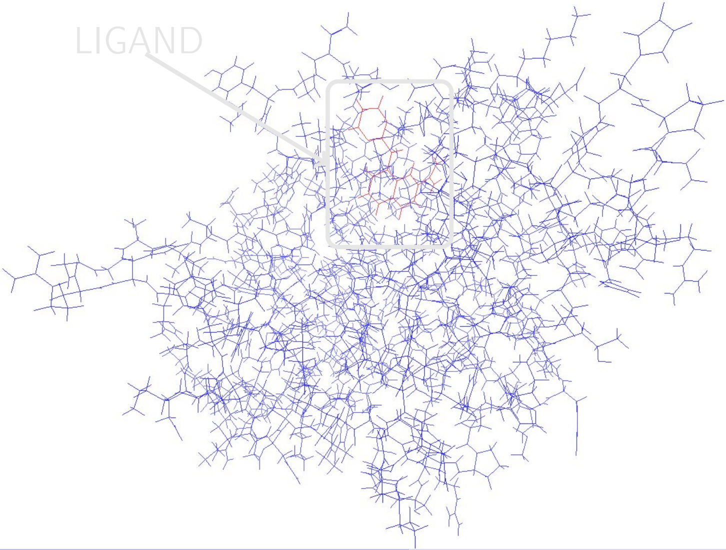
Среди элементов ТТ-интерполяции тензора найти минимум или максимум. Задача дискретной оптимизации решается как задача на собственные значения для диагональных матриц. Блочная минимизация отношения Релея в ТТ-формате, блоки размера 5, ТТ-ранги $\leqslant 5$ (О. С. Лебедева).

Функция	Область	Размер	Число итер.	(Ax, x)	(Ae_i, e_i) $e_i \approx x$	Точный max
$\prod_{i=1}^{3}(1+0.1\,x_i+\sin\, x_i)$	$[1, 50]^3$	2^{15}	30	428.2342	429.2342	429.2342
то же	$[1, 50]^3$	2^{30}	50	430.7838	430.7845	
$\prod_{i=1}^{3}(x+\sin\, x_i)$	$[1, 20]^3$	2^{15}	30	8181.2	8181.2	8181.2
то же	$[1, 20]^3$	2^{30}	50	8181.2	8181.2	

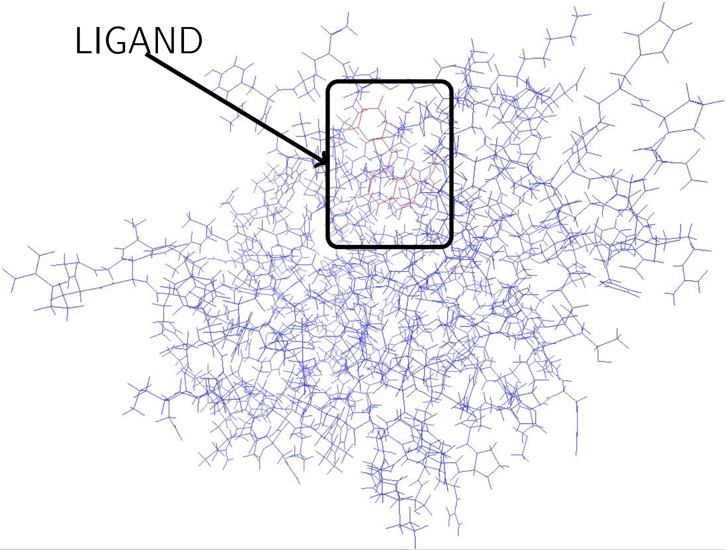
TWO TYPES OF OPTIMIZATION PROBLEMS

- Given a functional $f(x)$, find its approximate minimizer in the tensor train format.
  - DMRG algorithm (White'1993)
  - AMEn algorithm (Dolgov-Savostyanov'2013)
- Given a functional $f(x)$, chase its global minimum using tensor trains.
  - Application to the docking problem as an alternative to genetic algorithms.

ACCOMMODATION OF LIGAND INTO PROTEIN



ACCOMMODATION OF LIGAND INTO PROTEIN



OPTIMIZATION USING TT

INPUT: $f(x_1, \dots, x_d)$ and $n \times \cdots \times n$ grid.

OUTPUT: approximation to the global minimum.

IN THE LOOP:

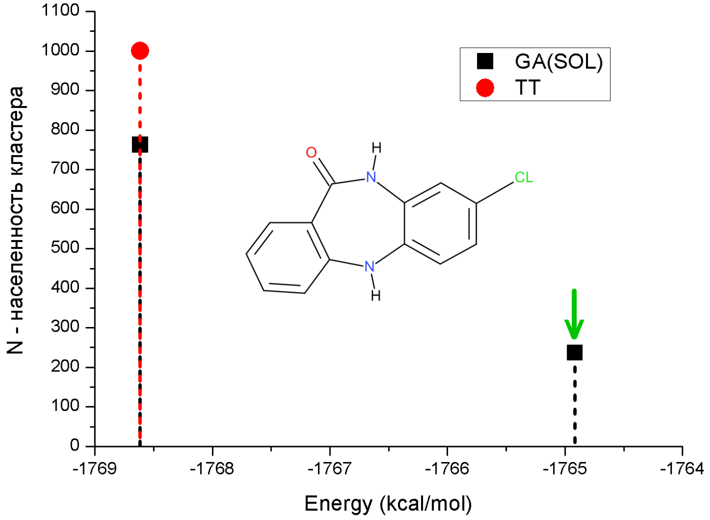
Step 1: Transformation of the functional s.t. $\arg\max g(x) = \arg\min f(x)$. E.g. $g(x) = \mathrm{arcctg}(f(x) - \tilde{f}_*)$.

Step 2: TT-CROSS interpolation with the adaptive choice of pseudo-max nodes.

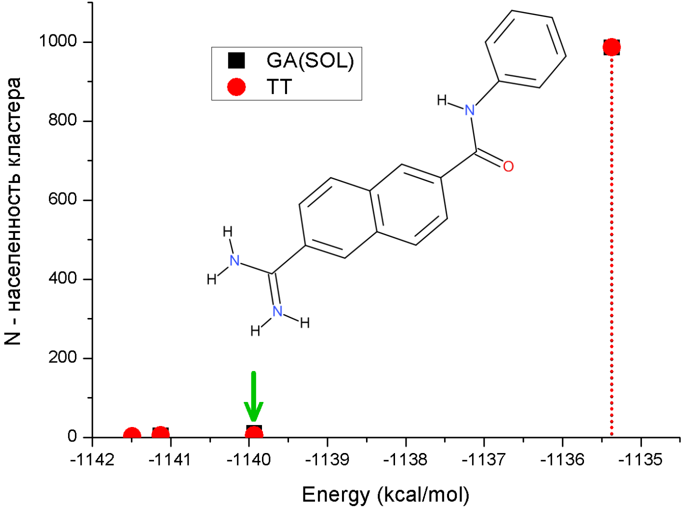
Step 3: Local optimizations of pseudo-max nodes.

Step 4: Renewal of $\tilde{f}_*$.

TTDock vs SOL: chk1_8



TTDock vs SOL: urokinase_7



TENSOR TRAIN DOCKING (TTdock)

- Tensor Train Decomposition opens new prospects in Global Minimum Search
- TTdock more than 10 times faster than SOL
- Direct docking: direct calculaion of all interactions between ligand and protein atoms
- Tensor Train Mining Minima: Global + Local Minima

D.Zheltkov, E.T. in collaboration with V.Sulimov and DIMONTA

NEW APPLICATIONS OF TENSOR TRAINS

- Fokker-Planck, Smoluchovski equations
- Differential equations with parameters
- Green functions in integral equations
- Spin dynamics
- Global optimization algorithms
- Many others