TENSORS AND COMPUTATIONS

Eugene Tyrtyshnikov

Institute of Numerical Mathematics of Russian Academy of Sciences

eugene.tyrtyshnikov@gmail.com

11 September 2013

REPRESENTATION PROBLEM FOR MULTI-INDEX ARRAYS

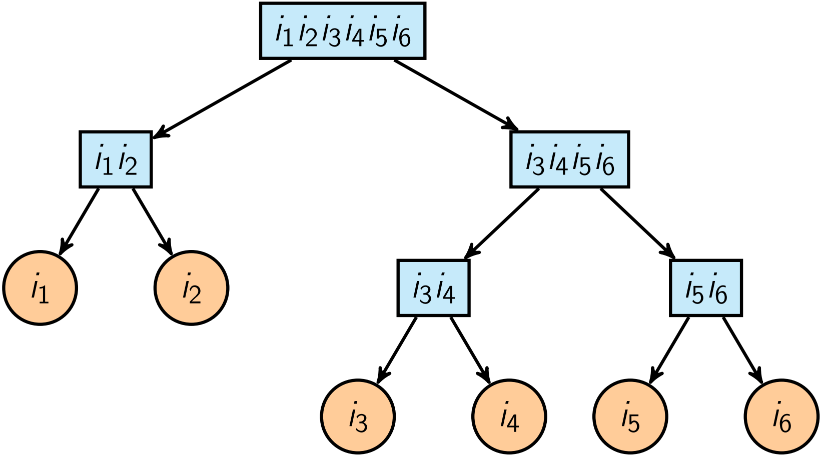
Going to consider an array $a(i_1, \ldots, i_d)$ of size

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

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

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

REDUCTION OF DIMENSIONALITY



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.

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.

GLOBAL SEARCH

A general heuristic scheme includes:

- Choose a reasonably small set M of optima suspects.
- Inflate M to a reasonably larger set M'
  - e.g. by mutation and crossover operations in the genetic or simulating annealing algorithms
- Assign some probabilities to the points of M' and deflate it to M'' of the same cardinality as M.
- Set $M := M''$ and repeat.

GLOBAL SEARCH IN A LOW-RANK MATRIX

- Find a low-rank skeleton representation or approximation
  - e.g. by the cross interpolation algorithm
- Find the maximal element using the skeletons
  - e.g. by reducing to the eigenvalue problem for a structured diagonal matrix

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.

DEFINITION

We call $r \times r$ submatrix $A_{\blacksquare}$ of rectangular $m \times n$ matrix A *maximum volume submatrix*, if it has maximum determinant in modulus among all possible $r \times r$ submatrices of A.

MAXIMIZATION VIA CROSS INTERPOLATION

DEFINITION

We call $r \times r$ submatrix $A_{\square}$ of rectangular $n \times r$ matrix A of full rank *dominant*, if all the entries of $AA_{\square}^{-1}$ are not greater than 1 in modulus.

DEFINITION

We call $r \times r$ submatrix $A_{\square}$ of rectangular $m \times n$ matrix A *dominant*, if it is dominant in the columns and rows it occupies.

THEOREM

If $A_\square$ is a dominant $r \times r$ submatrix of a $m \times n$ matrix A of rank r, then

$$|A_\square| \geq |A|/r^2.$$

MAXIMIZATION VIA CROSS INTERPOLATION

THEOREM

If $A_{\blacksquare}$ is maximum-volume $r \times r$ (nonsingular) submatrix of $m \times n$ matrix A, then

$$|A_{\blacksquare}| \geq |A|/(2r^2 + r).$$

S. Goreinov, I. Oseledets, D. Savostyanov, E. Tyrtyshnikov, N. Zamarashkin, How to find a good submatrix, *Matrix Methods: Theory, Algorithms and Applications. Devoted to the Memory of Gene Golub* (eds. V.Olshevsky and E.Tyrtyshnikov), World Scientific Publishers, Singapore, 2010, pp. 247–256.

MINIMIZATION VIA MAXIMIZATION

$$\Phi_n(x) := \exp\{-n(f(x) - f_n)\}$$

Assume that

$$\Phi_n(x_{n+1}) \geqslant \frac{1}{C}\Phi(x_{\min}).$$

Then

$$\exp\{-n(f_{n+1} - f_n)\} \geqslant \frac{1}{C}\exp\{-n(f_{\min} - f_n)\} \quad \Rightarrow$$

$$f_{n+1} - f_{\min} \leqslant \frac{\log C}{n}$$

MATRIX CROSS ALGORITHM

- Given *initial* column indices $j_1, \dots, j_r$.
- Find *good* row indices $i_1, \dots, i_r$ in these columns.
- Find *good* column indices in the rows $i_1, \dots, i_r$.
- 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.

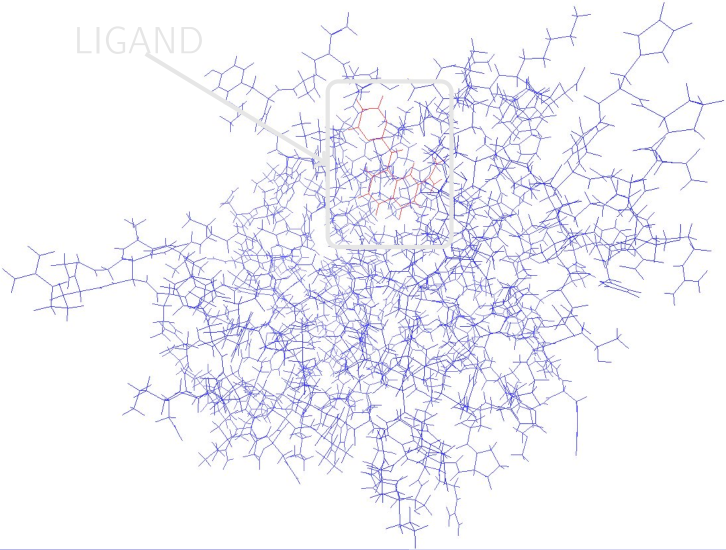
QUASIOPTIMALITY THEOREM FOR TENSOR TRAINS

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

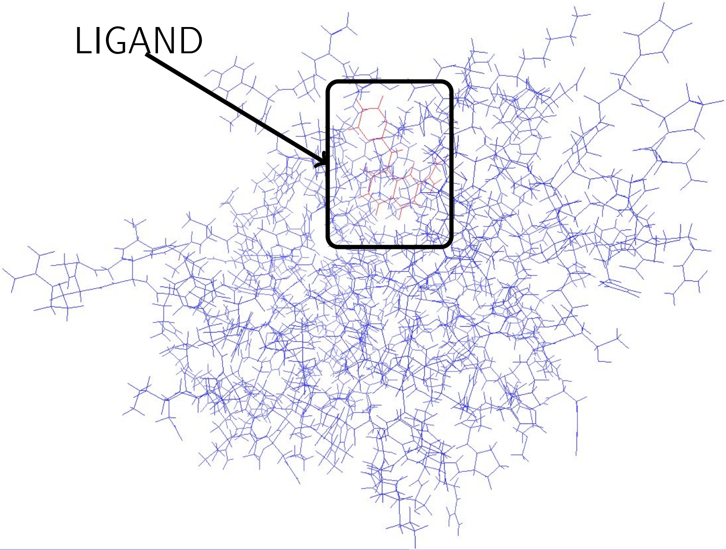
$$||A - \widetilde{A}||_C \leqslant 2dr\varepsilon ||A||_C.$$

ACCOMMODATION OF LIGAND INTO PROTEIN



DIRECT DOCKING IN THE DRUG DESIGN

ACCOMMODATION OF LIGAND INTO PROTEIN



MATHEMATICAL COMPONENTS OF THE DOCKING PROBLEM

- Define which degrees of freedom describe the ligand and the target protein and parametrize all possible interactions between them.
- Define the scoring function to be optimized.
- Find an efficient optimization algorithm over all selected degrees of freedom.

DOCKING AS A GLOBAL OPTIMIZATION PROBLEM

DIFFICULTIES:

- Degrees of freedom amount to 20-30 and higher.
- Many local minima.
- Singularities with large values of energy.
- High complexity of evaluation of the energy function.

OPTIMIZATION USING TT

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

OUTPUT: approximation to the global minimum.

IN THE LOOP:

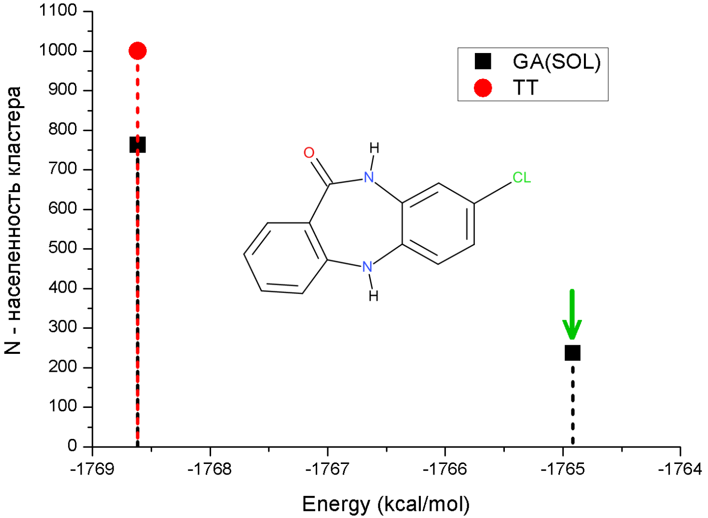
Step 1: Transformation of the functional s.t. $\arg\max g(x) = \arg\min f(x)$. E.g. $g(x) = \mathrm{arcctg}(f(x) - \tilde{f}_*)$.

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

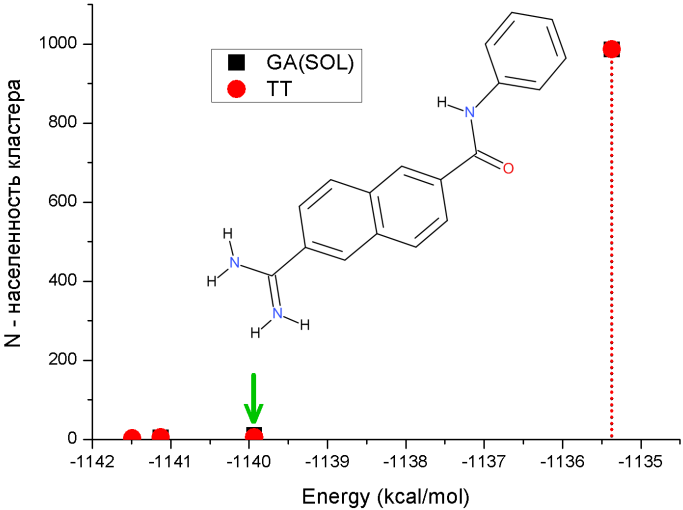
Step 3: Local optimizations of pseudo-max nodes.

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

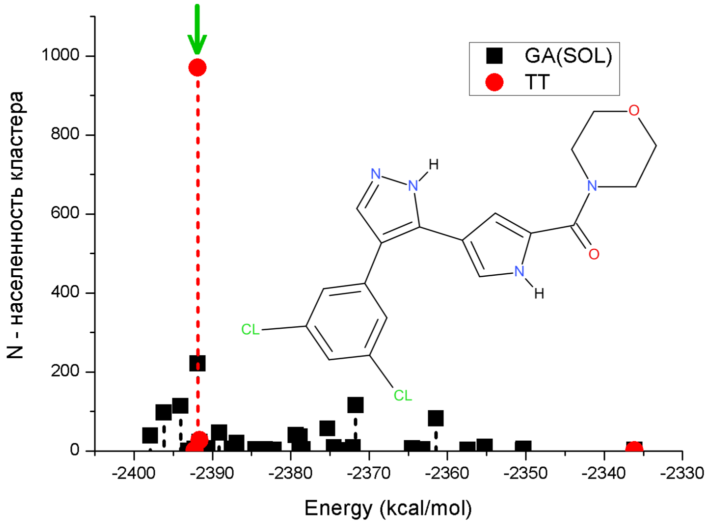
TTDock vs SOL: chk1_8



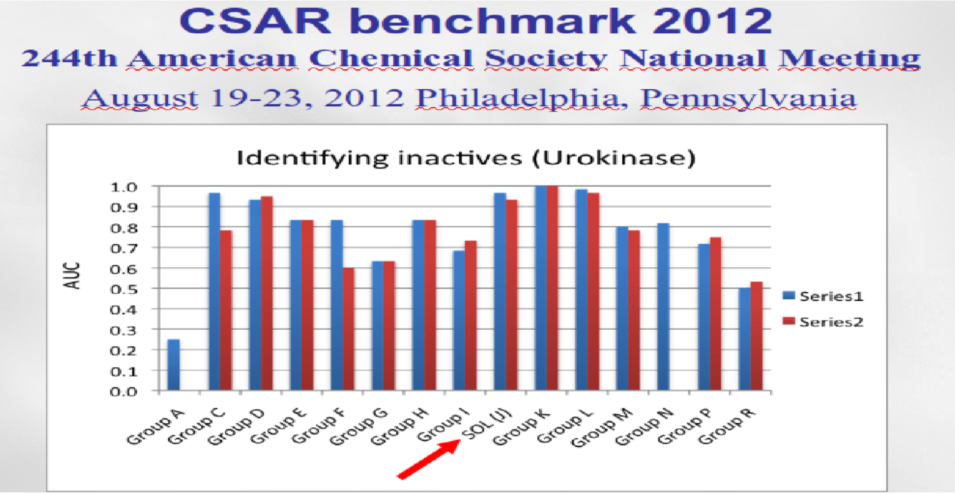
TTDock vs SOL: urokinase_7



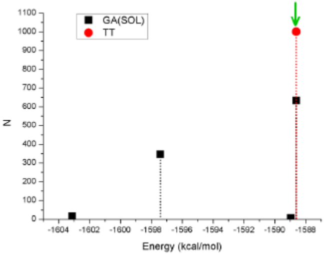
TTDock vs SOL: erk2_000124

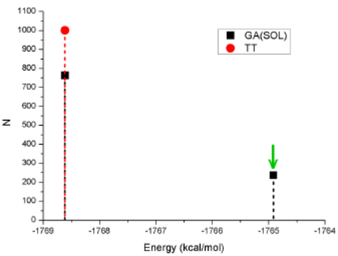


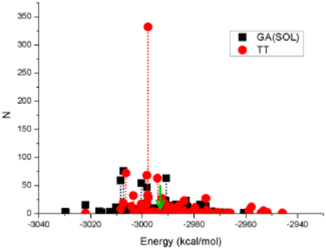
DOCKING PROGRAM SOL (DIMONTA)

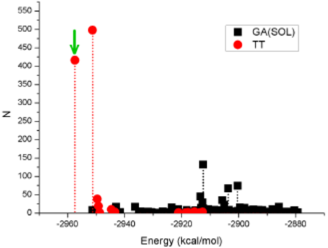


COMPARISON OF SOL AND TT-DOCK









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

WHY SHOULD WE USE TENSOR TRAINS

... **Surely every medicine is an innovation; and he that will not apply new remedies, must expect new evils ...**

Francis Bacon

(1561-1626)

OF INNOVATIONS

Thank you!