THE FIRST SCHOOL-CONFERENCE ON TENSOR METHODS IN MATHEMATICS AND DATA SCIENCE
10-21 November 2024, Shenzhen

Low rank structures and efficient solving of the integral equations

Bulat Valiakhmetov ♠† Eugene Tyrtyshnikov ♢
♠ Lomonosov Moscow State University ♢ Marchuk Institute of Numerical Mathematics
†valiahmetovbulat@mail.ru

Introduction

The mosaic–skeleton (MSk) format is known to be used for approximating dense matrices generated by smooth enough functions. In this work we consider the application of such block low rank structures to different problems:

- solving the inverse problem of remote sensing;
- time-integrating Smoluchowski equation with complex kernels.

Main Objectives

- Apply the mosaic–skeleton method to precondition linear systems.
- Make the direct solver for the problem of remote sending.
- Accelerate the finite difference scheme for solving Smoluchowski equations with complex kernels.

1 Remote Sensing: Basic Setup

We consider the inverse problem of remote sensing:

$$\frac{\mu_0}{4\pi}\int\limits_P \left(\frac{3\,(\varphi(x), x-y)\,(x-y)}{\|x-y\|^5} - \frac{\varphi(x)}{\|x-y\|^3}\right) dx = \psi(y), \quad y \in Q.$$

Its descretization leads to the rectangular linear system $Ax = b$. The solution is considered in terms of least squares method with regularization:

$$\|A\varphi - \psi\|_2 \to \min_{\varphi} \quad \Rightarrow \quad (A^*A + \lambda^2 I)\varphi = A^*\psi.$$

The obtained symmetric system is then solved via the CGNR iterations.

2 Preconditioning and Direct Solver

If we consider $A = \left[A_1\, A_2 \dots A_K\right]^T$ where each $A_k = U_k V_k^*$ has rank r_k, then we can obtain the total skeleton decomposition:

$$A = \mathrm{diag}\,(U_1, U_2, \dots, U_K)\,\left[V_1 \dots V_K\right]^*.$$

This block factor structure is used to get the approximate SVD for A in $O\left((M+N)R^2\right)$ operations, where $R = \sum_k r_k$. Such decomposition is used to make the preconditioner for $A^*A + \lambda^2 I$:

$$\hat{A}^{-1} = V\left(\Sigma^2 + \lambda^2 I\right)^{-1} V^*, \quad \hat{A} = U\Sigma V^*.$$

If the approximate SVD is accurate enough, the solution can be calculated directly:

$$\varphi_\lambda = V\,\mathrm{diag}\left(\frac{\sigma_1}{\sigma_1^2 + \lambda^2}, \dots, \frac{\sigma_R}{\sigma_R^2 + \lambda^2}\right) U^*\psi.$$

Results of the comparative **experiments** are in the following Table.

Matrix	Error	Residual	Normal res.	Iter.	t_{solve}, ms	t_{prep}, ms
Full	$1.78 \cdot 10^{-2}$	$2.92 \cdot 10^{-4}$	$4.74 \cdot 10^{-5}$	32	2750	1026
MSk	$1.63 \cdot 10^{-2}$	$2.41 \cdot 10^{-4}$	$1.15 \cdot 10^{-5}$	35	149	1169
Full-Prec	$3.5 \cdot 10^{-3}$	$9.9 \cdot 10^{-6}$	$8.98 \cdot 10^{-9}$	1	64.9	1198
MSk-Prec	$3.78 \cdot 10^{-4}$	$8.68 \cdot 10^{-8}$	$7.93 \cdot 10^{-10}$	1	4.3	1341
MSk-SVD	$3.8 \cdot 10^{-4}$	$8.7 \cdot 10^{-8}$	$7.93 \cdot 10^{-10}$	0	0.41	1341

Table 1: Matrix A of size 15000×9600, approximation accuracy $\varepsilon = 10^{-6}$, residual tolerance $\tau = 10^{-4}$

3 Smoluchowski Equations and Convolution

Consider the discrete version of integro–differential Smoluchowski equation:

$$\frac{dn_s}{dt} = \frac{1}{2}\sum_{i+j=s} K_{ij} n_i n_j - \sum_{j=1}^{N} K_{sj} n_s n_j, \quad s \in \overline{1, N}.$$

For mosaic–skeleton kernel K the matrix–vector multiplication can be done in $O(N \log N)$ operations, and the convolution in $O(N \log^2 N)$ operations via the FFT. It allows to speed up the computations for a wide range of complex kernels of high rank.

In the following Figure we show comparison of our approach with the efficient Majorant Monte Carlo simulations [1] method for the kernel K:

$$K_{ij} = \frac{(i+j)\left(i^{1/3} + j^{1/3}\right)^{2/3}}{(ij)^{5/9}\left|i^{2/3} - j^{2/3}\right|}.$$

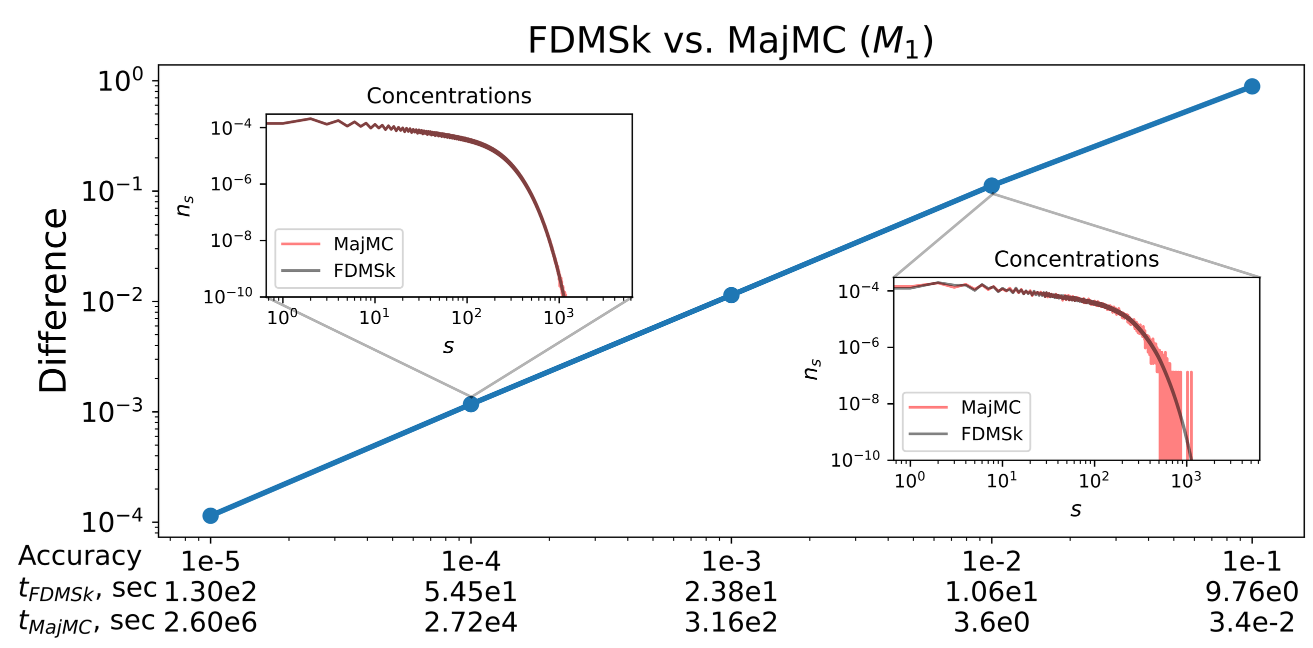

Figure 1: Comparison of Majorant Monte Carlo and Mosaic–skeleton + Runge–Kutta

4 Package `zaiMSk` [2]: Flexibility & Basics

Figure 2: Source at `gitlab.com/bulatral/mosaic-skeleton`

- Matrix approximation via Adaptive Cross or SVD.
- Matrix–vector multiplication and convolution.
- Krylov solvers with multiple RHS.
- Interface to C and Fortran.
- Threads and MPI parallelism.

Conclusions

- ▶ MSk helps preconditioning dense matrices while solving linear systems with block low rank structure.
- ▶ Large block low rank structures can be transformed into the direct solver.
- ▶ MSk format allows to solve a broad class of Smoluchowki coagulation equations and helps to speed up the finite difference ODE solvers.

References

[1] A. Osinsky. Low-rank Monte Carlo for Smoluchowski-class equations. *Journal of Computational Physics*, 506:112942, 2024.

[2] B. Valiakhmetov and E. Tyrtyshnikov. MSk - the package for a dense matrix approximation in the mosaic-skeleton format. In *Russian Supercomputing Days : Proceedings of the International Conference*, pages 20–27, Moscow, Russia, September 25–26 2023.

Acknowledgements

We thank S. Matveev and R. Dyachenko for the contribution to Smoluchowki equations solving, and D. Lukyanenko for important discussions on the remote sensing problem.