Matrix Equations and Tensor Techniques XI
Leuven, January 7–9, 2026

Nyström Low-Rank Approximation: Communication Lower Bounds and Algorithms

Hussam Al Daas[1], Grey Ballard[2], Laura Grigori[3], Md Taufique Hussain[2], *Suraj Kumar*[*,4], Mohammad Marufur Rahman[2], and Kathryn Rouse[5]

[1]STFC, Rutherford Appleton Laboratory, UK
[2]Wake Forest University, USA
[3]École Polytechnique Fédérale de Lausanne, Switzerland
[4] Institut national de recherche en sciences et technologies du numérique, France
[5]Inmar Intelligence, USA

The general form of a Nyström approximation of a positive semidefinite matrix $\mathbf{A}$ is given as $\widetilde{\mathbf{A}} = (\mathbf{A\Omega})(\mathbf{\Omega}^T\mathbf{A\Omega})^\dagger(\mathbf{A\Omega})^T$ [4, 5, 1], for some random matrix $\mathbf{\Omega}$. Theoretical guarantees of the quality of the approximation have been presented for several types of random matrices including Gaussian random matrices and random matrices based on Fourier-like transforms [2, 3].

Many low-rank approximation methods compute the product of a matrix with a random matrix. Because each processor can generate its portion of the random matrix, these products can be computed without communicating the random matrix, decreasing the amount of communication required from that of dense matrix multiplication. The goal of this work is to design efficient parallel algorithms for computing a Nyström approximation with a particular focus of avoiding unnecessary communication of random matrices. We address this question theoretically, establishing communication lower bounds and analyzing asymptotic algorithmic costs, as well as practically, implementing the most communication efficient algorithms and applying them to large data sets on

[*]Corresponding author: e-mail suraj.kumar@inria.fr

10s of nodes (100s of GPUs or 1000s of CPU cores) of a supercomputer. Our lower bound approach uses a geometric inequality that relates computation to data to build a constrained optimization problem whose analytical solution yields communication lower bounds. We consider both a single matrix multiplication involving a random input as well as the entire Nyström computation. We develop implementations using both Python and C++ and adapt them for both CPU-only and GPU platforms and explore the performance tradeoffs among the various configurations. Our results show the benefits of GPUs for performing dense matrix multiplication to accelerate Nyström approximation and the effectiveness of direct communication among GPUs to scale to large problems, and we obtain low-rank approximations of symmetric matrices of dimension 50,000 in fractions of a second.

We focus on $\mathbf{A\Omega}$ and $\mathbf{\Omega}^T\mathbf{A\Omega}$ computations. The main contributions of our work are to

1. establish communication lower bounds for the parallel computation of $\mathbf{B} = \mathbf{A\Omega}$ where $\mathbf{\Omega}$ is a random matrix and present parallel algorithms whose communication costs are optimal in all ranges;

2. establish communication lower bounds for the parallel computation of the sequence of computations $\mathbf{B} = \mathbf{A\Omega}$ followed by $\mathbf{C} = \mathbf{\Omega}^T\mathbf{B}$ where $\mathbf{\Omega}$ is a random matrix, and present parallel algorithms whose communication costs are close to the lower bounds;

3. implement the most efficient algorithms using both Python and C++ for CPUs and GPUs and benchmark the implementations to demonstrate the communication efficiency of the algorithms and their parallel scaling.

References

[1] A. Gittens and M. W. Mahoney. “Revisiting the Nyström method for improved large-scale machine learning”. In: *The Journal of Machine Learning Research* 17.1 (2016), pp. 3977–4041.

[2] N. Halko, P.-G. Martinsson, and J. A. Tropp. “Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions”. In: *SIAM review* 53.2 (2011), pp. 217–288.

[3] P.-G. Martinsson and J. A. Tropp. “Randomized numerical linear algebra: Foundations and algorithms”. In: *Acta Numerica* 29 (2020), pp. 403–572.

[4] E. J. Nyström. “Über die praktisch Auflölung of integral equations with applications to boundary value problems”. In: (1930).

[5] C. Williams and M. Seeger. “Using the Nyström method to speed up kernel machines”. In: *Advances in neural information processing systems* 13 (2000).