



OPEN ACCESS

EDITED AND REVIEWED BY

Daniel Potts,
Chemnitz University of Technology,
Germany

*CORRESPONDENCE

Edoardo Di Napoli
e.di.napoli@fz-juelich.de

SPECIALTY SECTION

This article was submitted to
Mathematics of Computation and Data
Science,
a section of the journal
Frontiers in Applied Mathematics and
Statistics

RECEIVED 07 September 2022

ACCEPTED 08 September 2022

PUBLISHED 23 September 2022

CITATION

Di Napoli E, Bientinesi P, Li J and
Uschmajew A (2022) Editorial:
High-performance tensor
computations in scientific computing
and data science.
Front. Appl. Math. Stat. 8:1038885.
doi: 10.3389/fams.2022.1038885

COPYRIGHT

© 2022 Di Napoli, Bientinesi, Li and
Uschmajew. This is an open-access
article distributed under the terms of
the [Creative Commons Attribution
License \(CC BY\)](#). The use, distribution
or reproduction in other forums is
permitted, provided the original
author(s) and the copyright owner(s)
are credited and that the original
publication in this journal is cited, in
accordance with accepted academic
practice. No use, distribution or
reproduction is permitted which does
not comply with these terms.

Editorial: High-performance tensor computations in scientific computing and data science

Edoardo Di Napoli^{1*}, Paolo Bientinesi², Jiajia Li³ and
André Uschmajew⁴

¹Simulation and Data Lab Quantum Materials, Jülich Supercomputing Centre, Forschungszentrum Jülich, Jülich, Germany, ²Department of Computing Science, Umeå University, Umeå, Sweden,

³Department of Computer Science, North Carolina State University, Raleigh, NC, United States,

⁴Max Planck Institute for Mathematics in the Sciences, Leipzig, Germany

KEYWORDS

tensor operation, tensor decomposition, tensor network, multilinear algebra, high performance optimization, low-rank approximation, Deep Learning, tensor library

Editorial on the Research Topic

High-performance tensor computations in scientific computing and data science

Introduction

In the last two decades, tensor computations developed from a small and little known subject to a vast and heterogeneous field with many diverse topics ranging from high-order decomposition and low-rank approximation to optimization and multi-linear contractions. At the same time, several of these operations with tensors are progressively and diversely applied to many, rather distinct domains; from Quantum Chemistry to Deep Learning, and from Condensed Matter Physics to Remote Sensing. These domain-specific applications of tensor computations present a number of particular challenges originating from their high dimensionality, computational cost, and complexity. Usually, because these challenges could be quite diverse among application areas, there is not an homogeneous and uniform approach in the development of software programs tackling tensor operations. On the contrary, very often developers implement domain-specific libraries which compromise their use across disciplines. The end result is a fragmented community where efforts are often replicated and scattered [1].

This Research Topic represents an attempt in bringing together different communities, spearheading the latest cutting-edge results at the frontier of tensor computations, and sharing the lessons learned in domain-specific applications. The issue includes ten research articles written by experts in the field. For the sake of clarity, the articles can be somewhat artificially divided in four main areas: (i) decompositions, (ii) low-rank approximations, (iii) high-performance operations, and (iv) tensor networks. In practice, many of the works in this Research Topic spill over the boundaries of such

areas and are interdisciplinary in nature, thus demonstrating how cross-fertilizing the field of tensor computations is.

Decompositions

In multilinear algebra, the Canonical Polyadic decomposition (CP) is one of several generalization of the matrix Singular Value Decomposition (SVD) to tensors. The problem considered in Psarras et al. is the estimation of the uncertainty associated with the parameters of a Canonical Polyadic tensor decomposition. The authors demonstrate that it is possible to perform such an estimation (jackknife resampling) without altering the input tensor at the cost of a modest increase in floating point operations. This observation makes it possible to take advantage of a recent technique—Concurrent Alternating Least Squares (CALS, [2])—to accelerate the computation of jackknife resampling. The authors make the software generated publicly available.

Khoromskaia and Khoromskij present the reduced higher-order SVD (RHOSVD), which is an efficient version of the high-order SVD (HOSVD) applicable to tensors in CP format. The authors focus on the important step of rank truncation necessary in domain-specific computations with large scale tensors in scientific computing. Besides a survey, the article offers new error and stability results for the RHOSVD, as well as several applications to problems for computational physics, notably the rank-structured computations involving multi-particle interaction potentials by using range-separated tensor format.

While recovering the decomposition of a tensor can be seen as an a-posteriori operation on a given tensor, a specific tensor decomposition can be a-priori imposed as initial condition to the solution of a given problem. The work by Hendriks et al. studies problems that can be formulated as a block row Kronecker-structured (BRKS) linear system with a constrained tensor as the solution. The authors consider low-rank multilinear singular value decomposition (MLSVD), CP, and tensor train (TT) as the constrained tensors. Efficient algorithms to find these solutions are provided for large and high-order data tensors. This work also derives conditions under which the constrained tensors can be retrieved from a BRKS system. The experimental results demonstrate effectiveness of the proposed algorithms including an application to hyperspectral image reconstruction.

Low-rank approximations

One important application of low-rank tensors is the representation of high-dimensional functions. In their respective papers, Ayvaz et al. and Götte et al. demonstrate how low-rank tensor decomposition can be used for representing and optimizing certain classes of multivariate polynomials,

essentially by using a low-rank model for their coefficient tensors. This approach provides practical access to quite a rich set of nonlinear classes of multivariate polynomials in low-parametric format that can be used as models in several tasks of data science and machine learning. These applications are amply demonstrated in the papers and used as a confirmation of the efficacy and correctness of the methods. While in Ayvaz et al. the authors focus on the CP format and efficient optimization based on Gauss-Newton-type algorithms, the work presented in Götte et al. proposes a block sparse TT format in combination with alternating least squares optimization.

Cohen introduces a framework for structured low-rank approximations of matrices and tensors in which the columns of one of the factor matrices are known or required to be sparse with respect to a fixed dictionary. Such a model subsumes several special cases with important applications in signal processing and data science. The focus of the work is on efficient optimization algorithms, especially on the sparse-coding sub-problem that appears when applying an alternating optimization strategy, which is of interest in itself. Several approaches, both convex and non-convex, are considered for handling this important problem and their performance is compared. The paper therefore also serves as a valuable overview on the subject.

HPC operations

In their rather comprehensive paper Georganas et al. present a programming abstraction (the Tensor Processing Primitives or TPP for short) striving for efficient and portable implementation of tensor operations, with a special focus on Deep Learning (DL) workloads. The aim of these primitives is to provide a 'middle way' between the monolithic and inflexible operators offered by DL libraries and the high level of abstraction provided by Tensor Compilers. The TPP attempt to strike a balance between these two extremes by providing relatively low-level 2D tensor primitives that act as building blocks for more complex and high-level DL operators. In other words, the TPP specification are platform agnostic while their implementation is platform specific. The article provides numerous practical examples where TPP are used in the realm of DL workloads as well as HPC tasks not specific to data science.

On a completely different direction, Bassoy presents a technique to implement basic tensor operations in C++ avoiding pointer arithmetic and instead relying on iterators. The technique is incorporated into the uBlas extension of Boost, and is demonstrated on element-wise tensor operations (e.g., tensor addition), as well as tensor multiplications (tensor-times-vector, tensor-times-matrix, and tensor-times-tensor). The aim is a modular design to deal with tensors

and sub-tensors of arbitrary dimension, abstracting from storage formats.

Tensor Networks

Tensor Networks methods originated from Condensed Matter Physics but their application nowadays can span diverse fields such as Quantum Computing and Artificial Intelligence and has emerged as a mainstream field in tensor computations [3]. This Research Topic includes two publications which are at the crossroad between HPC and Tensor Networks. The paper by Lyakh et al. considers the processing of tensor networks. Specifically, it introduces a high-performance library to build, transform, and numerically evaluate tensor networks with arbitrary graph structures and complexity. The library is designed to run on laptops, workstations, as well as HPC platforms, including shared-memory, distributed-memory, and GPU-accelerated systems.

While Lyakh et al. focus on the specifics of tensor networks operations, the work by Evenbly maintains an high-level approach and is aimed at researchers already familiar with the theoretical setup of Tensor Networks that want to code their own software programs. It provides a practical description of how such programs need to be designed and implemented if they are going to ripe the benefits of High-Performance low-level numerical libraries and parallel architectures. The content is organized in sections, each covering a specific building block appearing in Tensor Network algorithms, such as contractions, decompositions, and gauge transformations. At the end of each section a useful summary is provided as a sort of recipe to realize in practice the specific Tensor Network operation in terms of the building blocks.

References

1. Psarras C, Karlsson L, Li J, Bientinesi P. The landscape of software for tensor computations. *CoRR. abs/2103.13756* (2021) doi: 10.48550/arXiv.2103.13756
2. Psarras C, Karlsson L, Bro R, Bientinesi P. Algorithm XXX: concurrent alternating least squares for multiple simultaneous canonical polyadic decompositions. *ACM Trans Math Softw.* (2022) 48:1–20. doi: 10.1145/3519383

Author contributions

EDN wrote the introduction and finalized the manuscript. All authors contributed to the manuscript a short summary for the papers they edited, and approved the submitted version.

Acknowledgments

We would like to thank the effort and contribution of all review editors whose meticulous and time-consuming work made this Research Topic possible.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

3. Silvi P, Tschirsich F, Gerster M, Jünemann J, Jaschke D, Rizzi M, et al. The Tensor Networks Anthology: Simulation techniques for many-body quantum lattice systems. *SciPost Phys Lect Notes.* (2019) 8:8. doi: 10.21468/SciPostPhysLectNotes.8