Algorithmic Optimization for Dense Tensor Decomposition

Yujie (Jeffrey) Jiang
Advisor: Grey Ballard | Co-Author: Koby Hayashi, Michael Tobia

WAKE FOREST UNIVERSITY
Department of Computer Science

Abstract

This research is inspired by neuroscientists who need to analyze huge sets of data, such as those in functional magnetic resonance imaging (fMRI) studies. This data is naturally multidimensional, as it tracks behavior in a set of regions over multiple time points and for multiple human subjects; we refer to a multidimensional data set as a tensor. Tensor decomposition is a method of reducing a tensor into a smaller set of more meaningful components, and this technique is widely used in many scientific fields such as signal processing, machine learning, and chemometrics. This research is focused on a specific tensor decomposition technique called CANDECOMP/PARAFAC Tensor Decomposition (CP decomposition). Our main goal is to make algorithmic optimizations in order to reduce the cost (both on time and space) of doing data analyses.

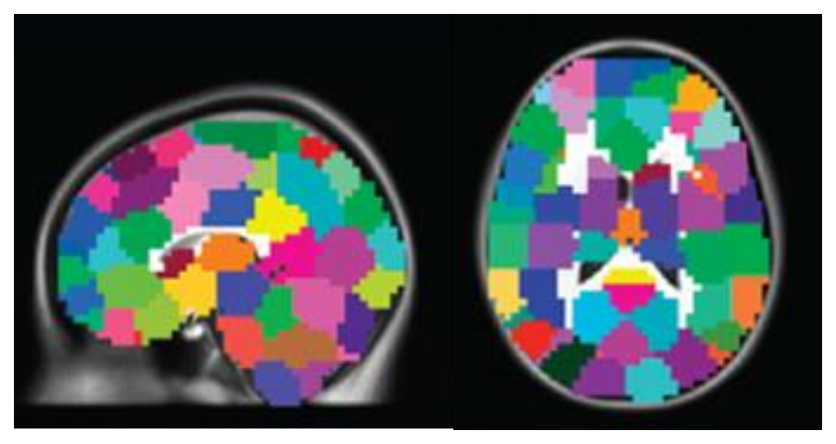
There is a specific computational step in CP decomposition known as Matricized Tensor Times Khatri-Rao Product (MTTKRP), which occupies almost all of the running time. We implemented two novel MTTKRP methods with shared-memory parallel algorithms. The benchmarking results showed that our parallel implementation to compute a CP decomposition of a neuroimaging dataset achieved a speedup of up to 7.4 over existing parallel solutions.

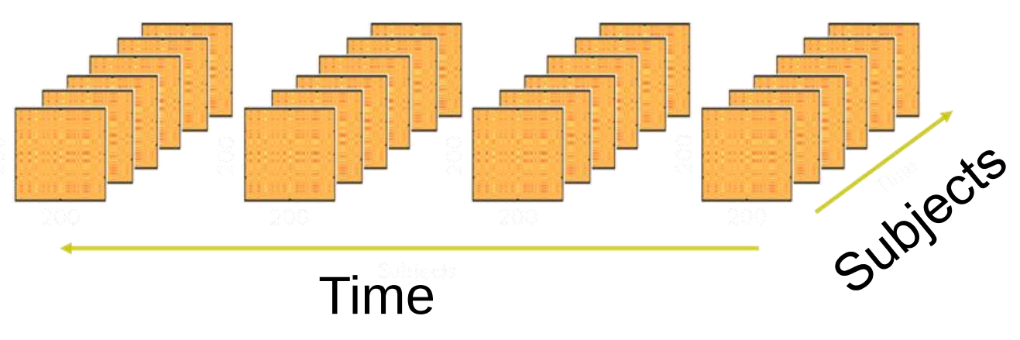
Problem Significance

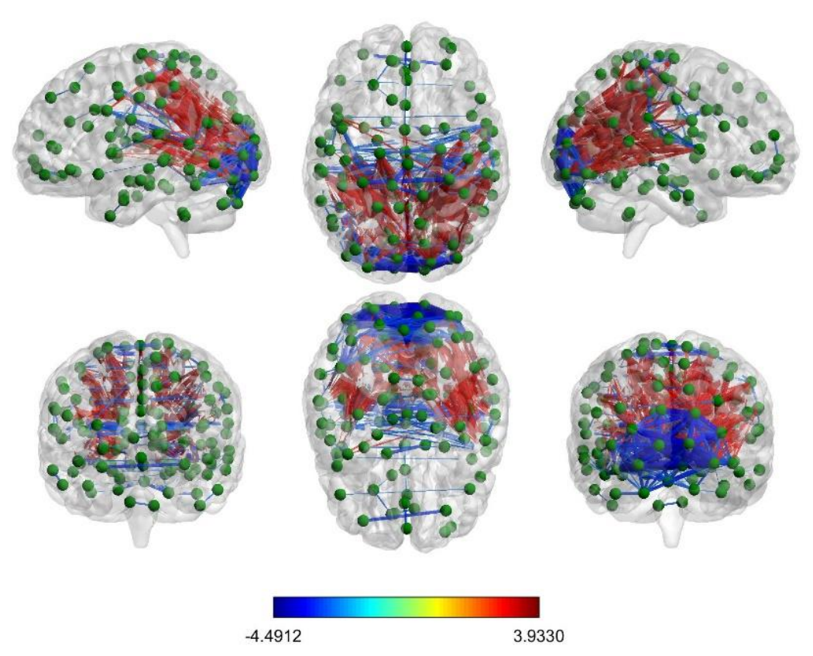
Data analysis: a process of inspecting, transforming, and modeling data.

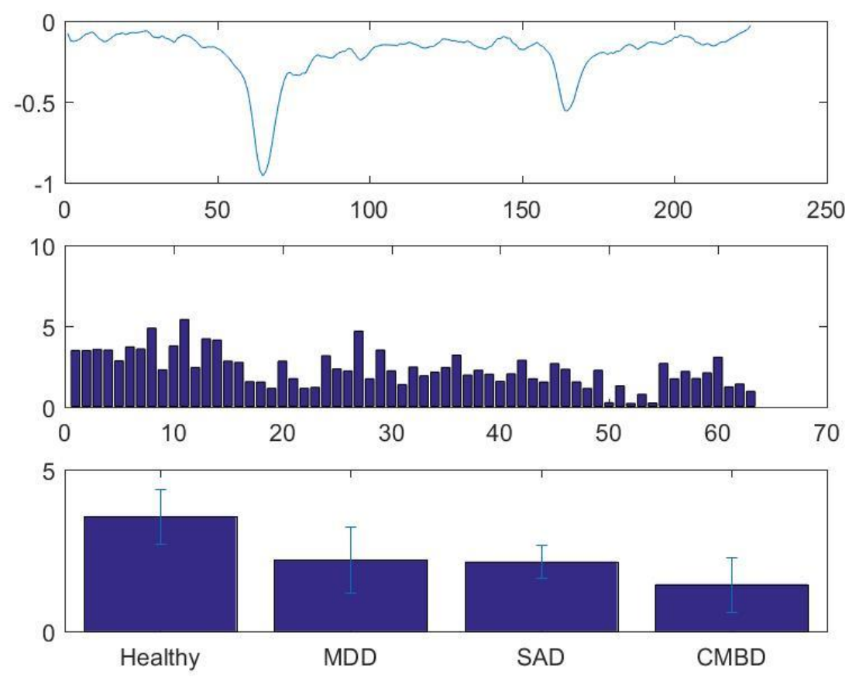
- Discover useful information from huge and messy data and then support decision-making.
- Multidimensional data analysis has a broad application in many areas, including but not limited to
  - Machine learning
  - Chemometrics
  - Signal processing
  - Neuroscience.

We especially focused on is **Functional magnetic resonance imaging (fMRI)**—a medical technology which measures brain activity by detecting changes associated with blood flow.









Terminology

CP Decomposition:

Also known as “CANDECOMP/PARAFAC Tensor Decomposition”, is a broadly used method for tensor factorization. It factorizes a tensor into a sum of outer products of vectors. For example, for a 3-way tensor $\mathcal{X}$, the CP decomposition can be written as

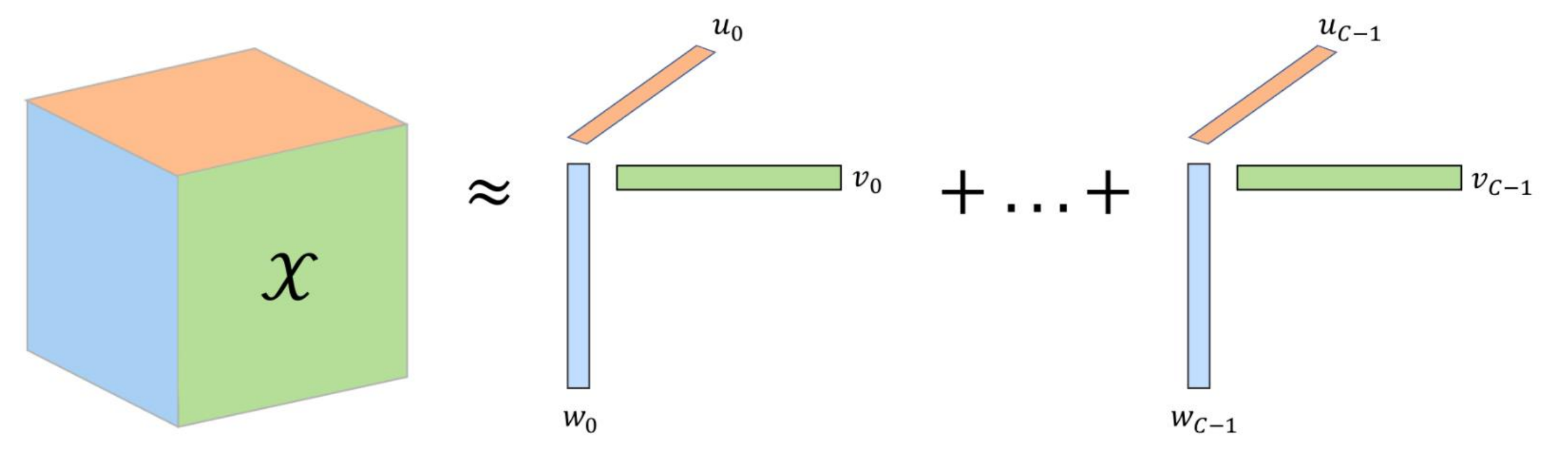
$$\mathcal{X} \approx \sum_{c=0}^{C-1} u_c \circ v_c \circ w_c$$

where $R > 0$ and u_c, v_c, w_c are vectors of appropriate dimensions. The notation " ∘ " denotes the outer product for tensors.

Approach

Our Contributions:

- Developed a row-wise, shared memory algorithm for computing a Khatri-Rao product of multiple matrices.
- Implemented a new 1-step and an existing 2-step MTTKRP algorithm.
- Parallelized both algorithms using a combination of OpenMP and multithreaded BLAS.



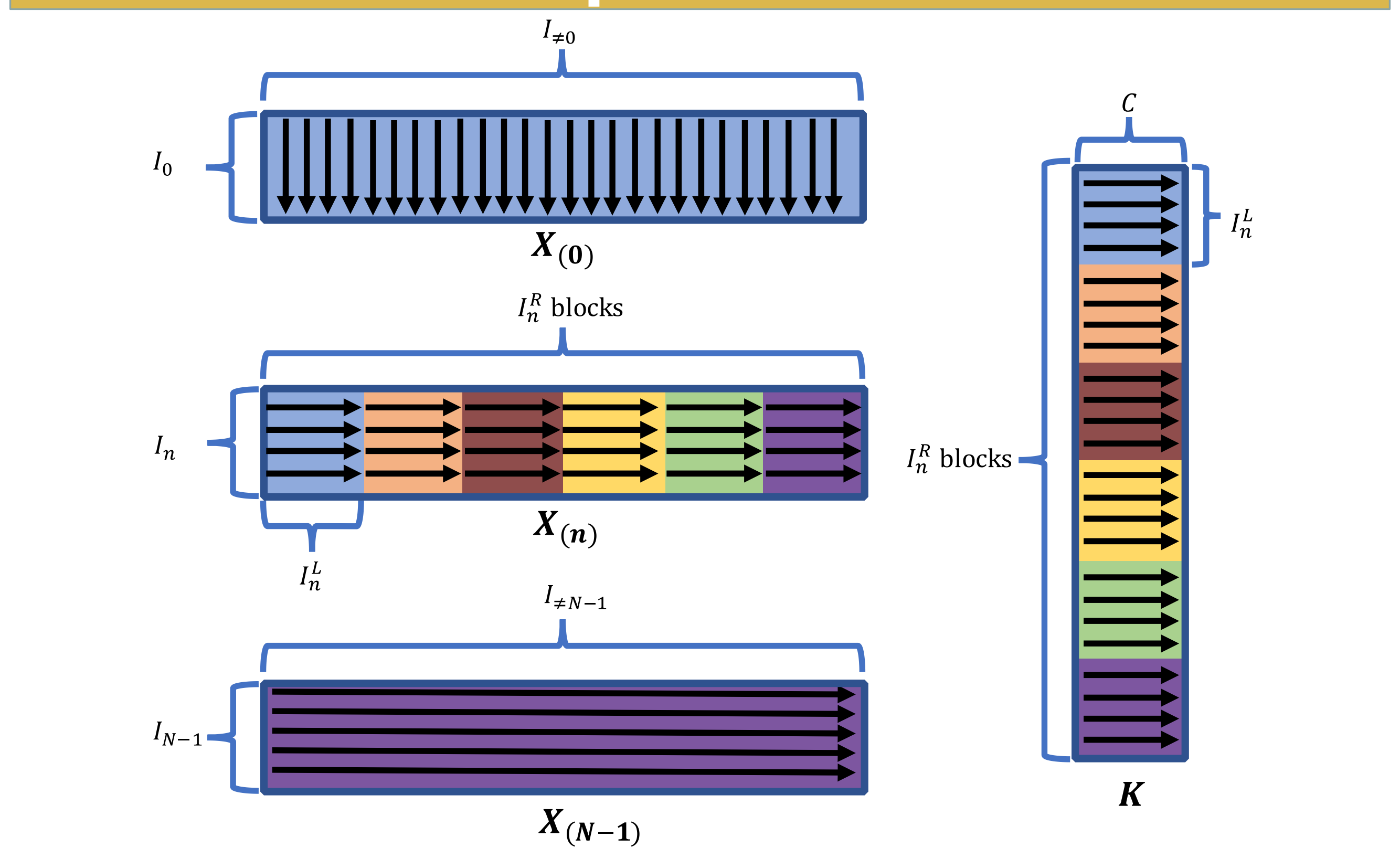
Algorithm 5 CP_ALS

```
Require: $\mathfrak{X}$ is an $N$-way tensor with dimensions $I_0 \times I_1 \times \cdots \times I_{N-1}$, $n \in [N]$, $\mathbf{U}_{(n)}$ is the $n^{th}$
  factor matrix, and a rank $C$
  function $\mathfrak{Y} = \text{CP\_ALS}(\mathfrak{X}, C)$
    while stopping conditions not met do
      for $n \in [N]$ do
        $\mathbf{H} = \mathbf{U}_{(0)}^\mathsf{T}\mathbf{U}_{(0)} * \cdots * \mathbf{U}_{(n-1)}^\mathsf{T}\mathbf{U}_{(n-1)} * \mathbf{U}_{(n+1)}^\mathsf{T}\mathbf{U}_{(n+1)} * \cdots * \mathbf{U}_{(N-1)}^\mathsf{T}\mathbf{U}_{(N-1)}$
        $\mathbf{M} = \mathbf{X}_{(n)}(\mathbf{U}_{(N-1)} \odot \cdots \odot \mathbf{U}_{(n+1)} \odot \mathbf{U}_{n-1} \odot \cdots \odot \mathbf{U}_{(0)})$
        solve $\mathbf{U}_n = \mathbf{M}\mathbf{H}^\dagger$
      end for
    end while
  end function
Ensure: $\mathfrak{Y}$ is a rank $C$ CP Model
```

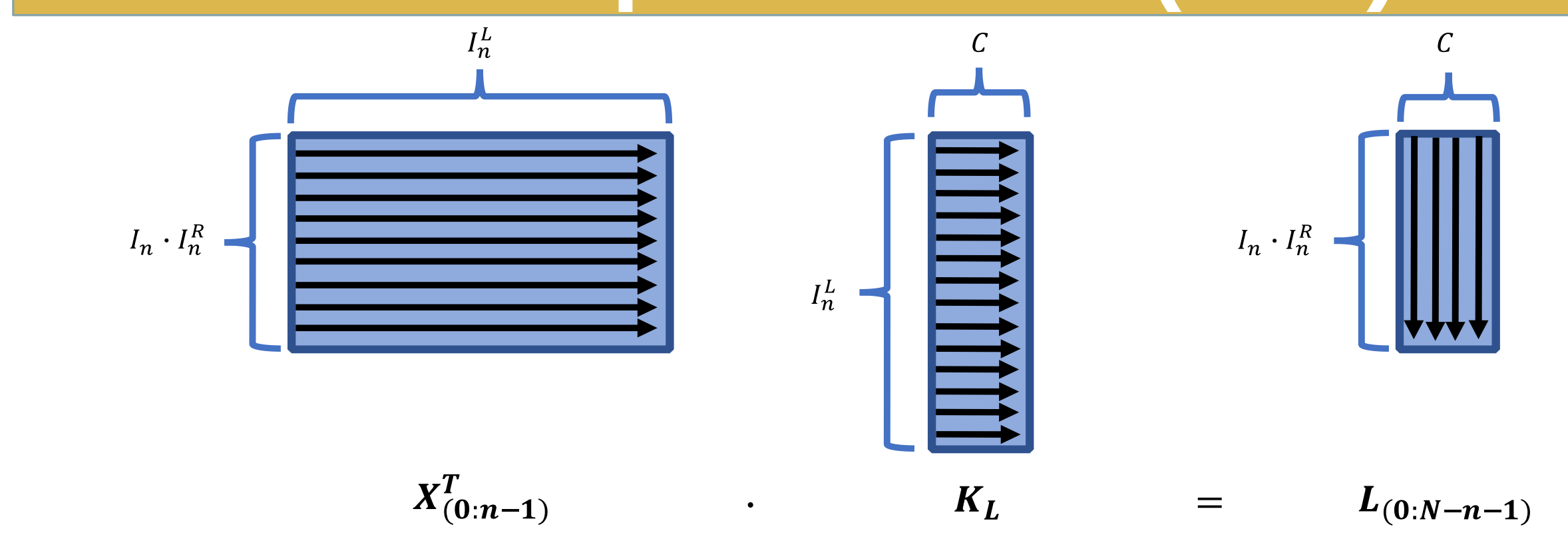
The most expensive step—Matricized Tensor Times Khatri Rao Product

$$\mathbf{M} = \boldsymbol{X}_{(\boldsymbol{n})}(\boldsymbol{U}_{\boldsymbol{N-1}} \odot \ldots \odot \boldsymbol{U}_{\boldsymbol{n+1}} \odot \boldsymbol{U}_{\boldsymbol{n-1}} \odot \ldots \odot \boldsymbol{U}_{\boldsymbol{0}})$$

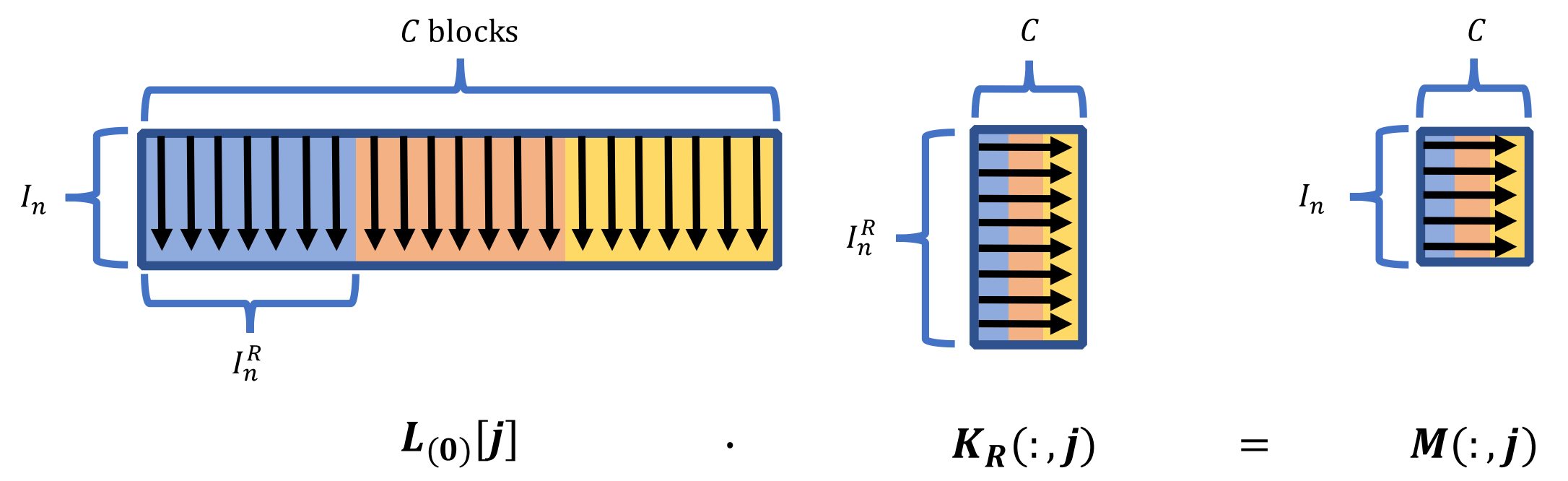
1-step MMTKRP



2-step MTTKRP (Left)



Step 1: Compute a *Partial MTTKRP*

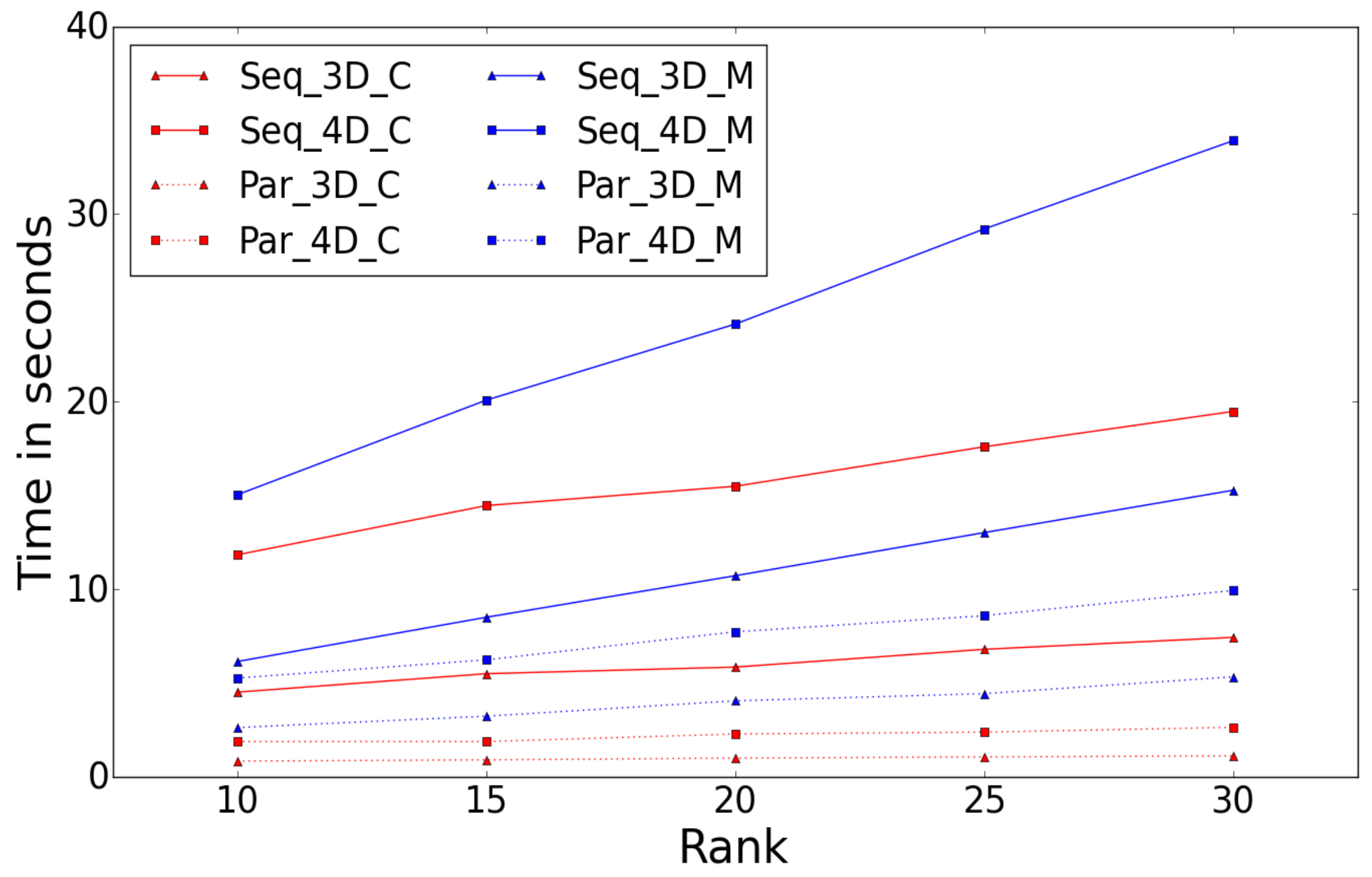


Step 2: Compute a series of TTVs (tensor times vector)

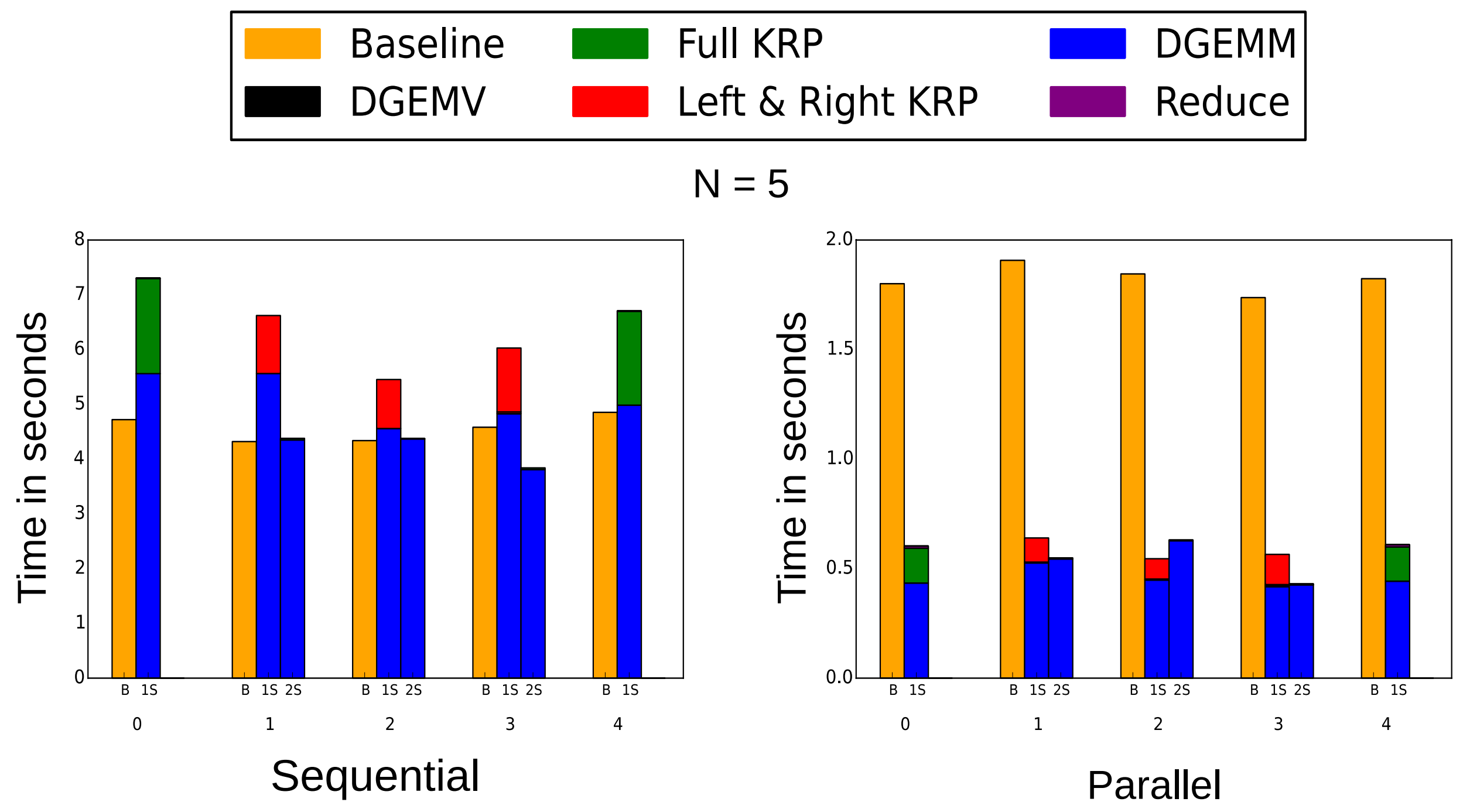
Results (selected)

An essential part of this work is to speed up the alternating least squares algorithm for CP decomposition (aka CP-ALS) in order to analyze neuroscience (fMRI) data.

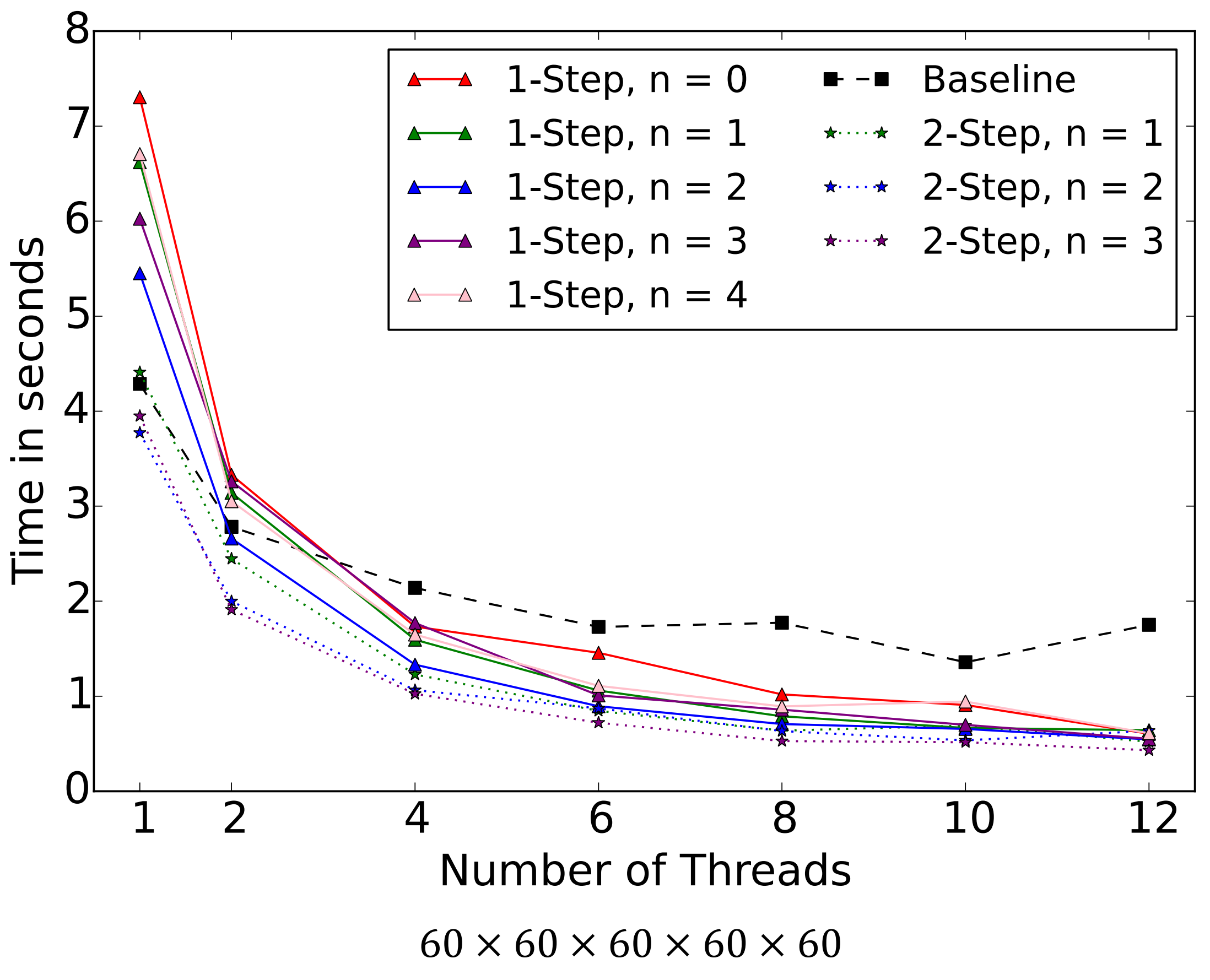
- Data size: 3-way tensor with dimension $225 \times 59 \times 19900$.
  - Refined by linearization from a 4-way tensor of size $225 \times 59 \times 200 \times 200$.
  - Representing for 225 time steps and for 59 subjects the correlation between fMRI signals measured at 200 different brain regions.



Per iteration time of a CP decomposition via ALS. Matlab used the Tensor Toolbox cp_als function, version 2.6.



Time breakdown of 1-step and 2-step MTTKRP (and baseline DGEMM) across modes for varying numbers of modes.



Time comparison of 1-step and 2-step MTTKRP algorithms for different modes over varying numbers of threads. The baseline DGEMM benchmark is the time to multiply column-major matrices of the same dimensions as the MTTKRP.