Accelerating Alternating Least Squares for Tensor Decomposition by Pairwise Perturbation

Linjian Ma[1] and Edgar Solomonik[1]

LPNA @CS@Illinois

[1]Department of Computer Science
University of Illinois at Urbana-Champaign

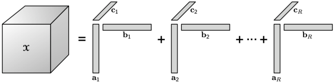
SIAM PP 2020
Seattle, WA

Outline

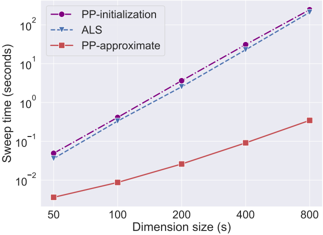
1. Overview
2. Alternating Least Squares for CP Decomposition
3. Pairwise Perturbation Algorithm
4. Performance Results
5. Conclusion

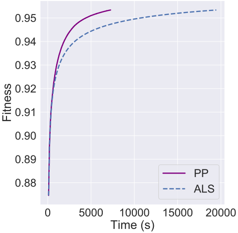
Overview

CP tensor decompositions[1]



- Pairwise perturbation approximates Alternating Least Squares (ALS)
  - rank R CP decomposition of tensor with order N and size s : reduces cost of sweep from $O(s^N R)$ to $O(s^2 R)$



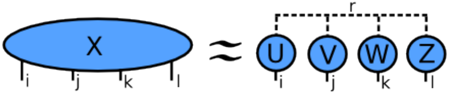


[1] Kolda and Bader, SIAM Review 2009

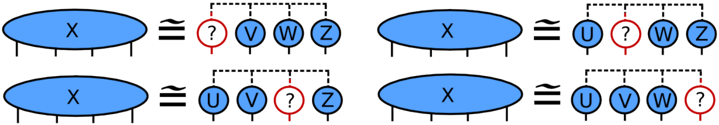
Alternating Least Squares for CP Decomposition

Consider rank R CP decomposition of an $s \times s \times s \times s$ tensor

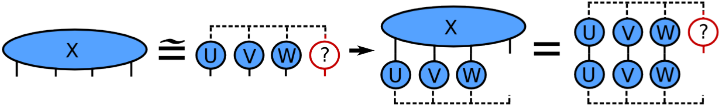
$$x_{ijkl} \approx \sum_{r=1}^{R} u_{ir} v_{jr} w_{kr} z_{lr},$$



ALS updates factor matrices in an alternating manner



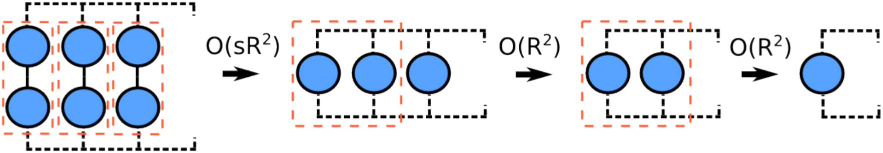
Each quadratic subproblem is typically solved via normal equations



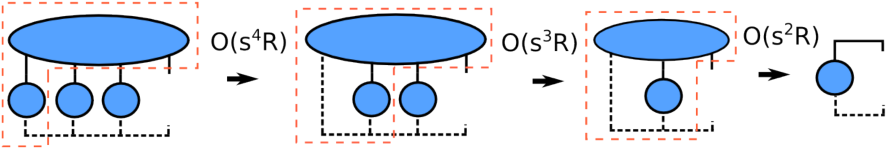
$$\boldsymbol{X} \cong \boldsymbol{\Gamma} \color{red}{\boldsymbol{Z}} \quad \rightarrow \quad \boldsymbol{M}^{(4)} = \boldsymbol{\Gamma}^T \boldsymbol{X} = \boldsymbol{\Gamma}^T \boldsymbol{\Gamma} \color{red}{\boldsymbol{Z}}$$

Tensor Contractions in CP ALS

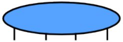
The normal equations ($\mathbf{\Gamma}^T\mathbf{\Gamma}$) are cheap to compute

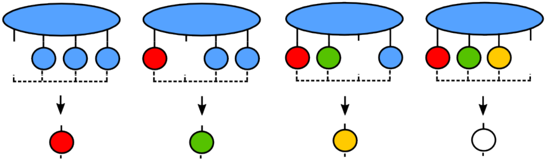


But forming the right-hand sides ($\boldsymbol{M}^{(4)}$) requires expensive MTTKRP (matricized tensor-times Khatri-Rao product)



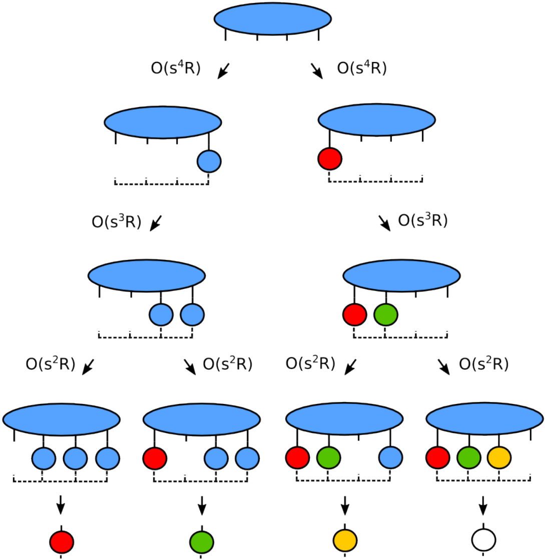
CP ALS Dimension Trees[2]





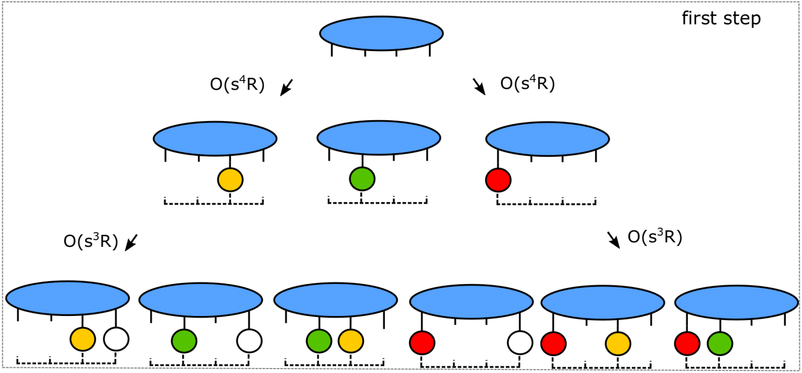
[2] Phan, Tichavskỳ, and Cichocki, IEEE Transactions on Signal Processing 2013

CP ALS Dimension Trees[3]



[3] Phan, Tichavskỳ, and Cichocki, IEEE Transactions on Signal Processing 2013

CP ALS with Pairwise Perturbation



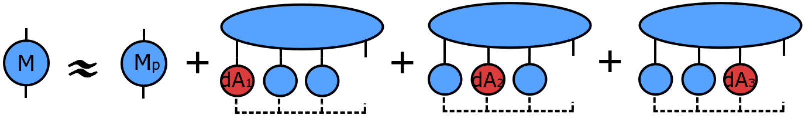
CP ALS with Pairwise Perturbation

Pairwise perturbation (PP) approximates $\boldsymbol{M}^{(n)} \approx \tilde{\boldsymbol{M}}^{(n)}$ using pairwise perturbation operators $\boldsymbol{\mathcal{M}}_p^{(i,n)}$

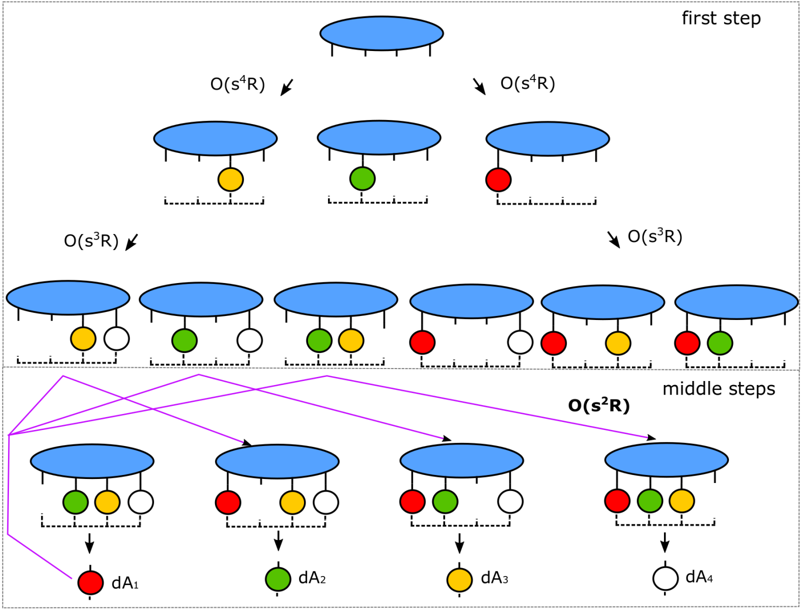
- Write $\boldsymbol{A}^{(n)} = \boldsymbol{A}_p^{(n)} + d\boldsymbol{A}^{(n)} \rightarrow \boldsymbol{M}^{(n)} = \boldsymbol{X}_{(n)} \bigodot_{i=1, i\neq n}^{N} (\boldsymbol{A}_p^{(i)} + d\boldsymbol{A}^{(i)})$
- Elementwise,

$$\boldsymbol{M}^{(n)}(y,k) = \boldsymbol{M}_p^{(n)}(y,k) + \sum_{i=1, i\neq n}^{N} \sum_{x=1}^{s_i} \boldsymbol{\mathcal{M}}_p^{(i,n)}(x,y,k) d\boldsymbol{A}^{(i)}(x,k) +$$

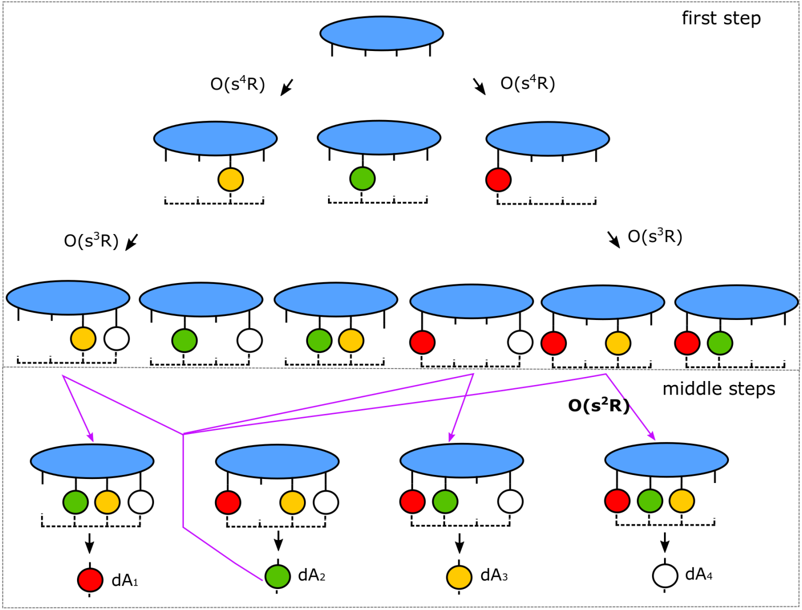
$$\sum_{i=1, i\neq n}^{N} \sum_{j=i+1, j\neq n}^{N} \sum_{x=1}^{s_i} \sum_{z=1}^{s_j} \boldsymbol{\mathcal{M}}_p^{(i,j,n)}(x,z,y,k) d\boldsymbol{A}^{(i)}(x,k) d\boldsymbol{A}^{(j)}(z,k) + \cdots$$



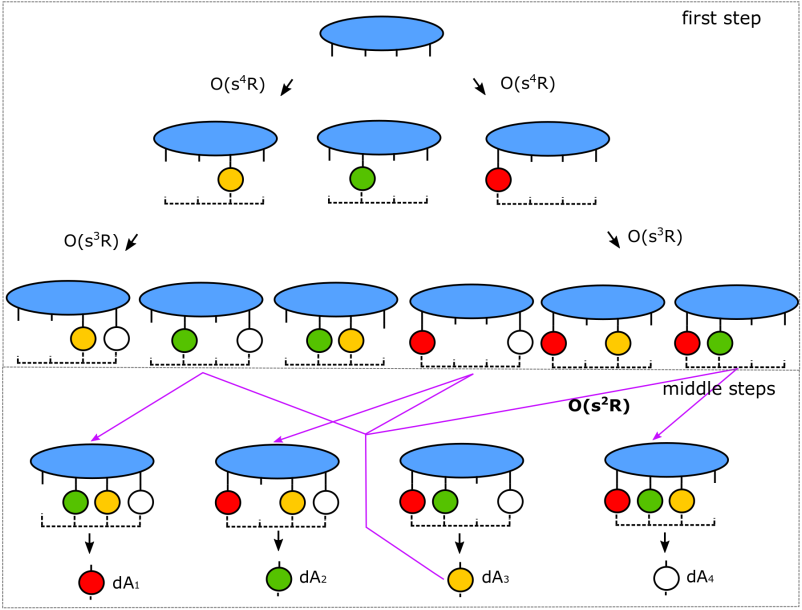
CP ALS with Pairwise Perturbation



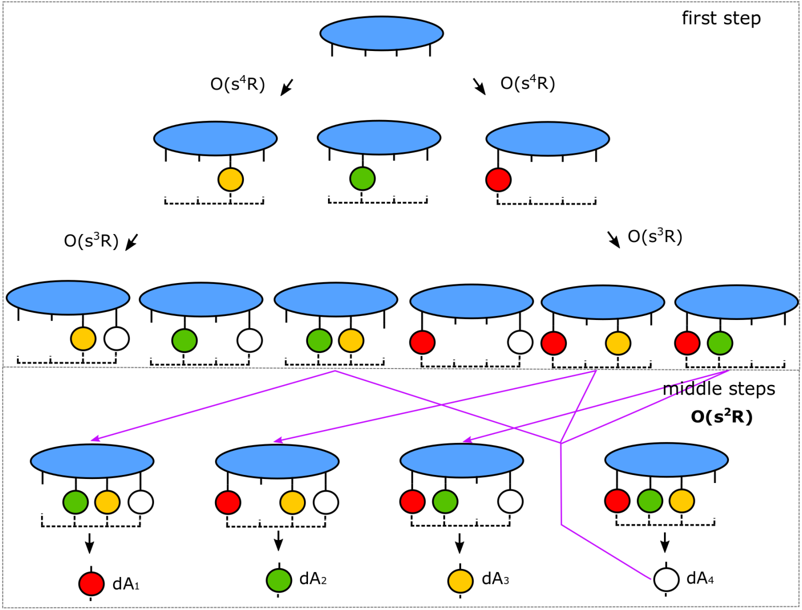
CP ALS with Pairwise Perturbation



CP ALS with Pairwise Perturbation



CP ALS with Pairwise Perturbation



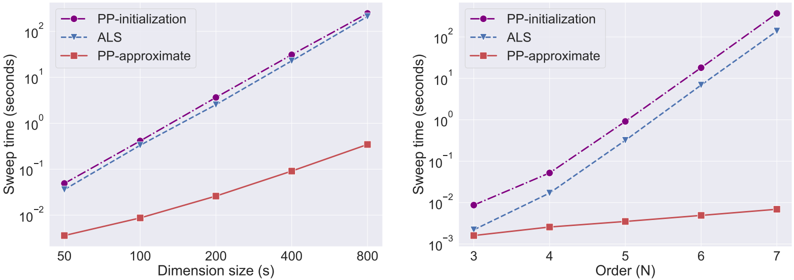
Implementation

We used NumPy and Cyclops Tensor Framework[4] to implement standard dimension tree ALS and pairwise perturbation

- Cyclops is a C++ library with Python interface that distributes each tensor over MPI
- Used in chemistry (PySCF, QChem), quantum circuit simulation (IBM/LLNL), and graph analysis (betweenness centrality)
- Sparse tensors supported but unused here
- Used interface to ScaLAPACK SVD to solve linear systems

[4] https://github.com/cyclops-community/ctf

Sequential Benchmark results

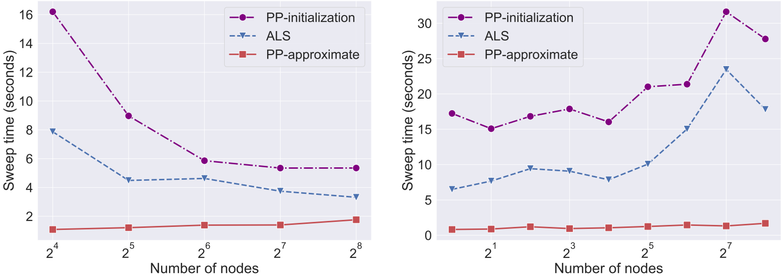


(a) Benchmark 1

(b) Benchmark 2

- Experiments performed on Blue Waters supercomputer, implemented with NumPy
- Benchmark 1: order $N = 3$, rank $R = 50$
- Benchmark 2: dimension size $s = 20$, rank $R = 20$
- First step of PP (setup) costs slightly more than ALS sweep
- Middle steps (subsequent approximations) up to 10^4X faster

Parallel scaling results

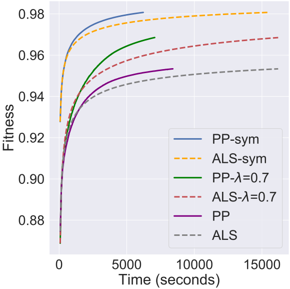


(c) Strong scaling of CP decomposition

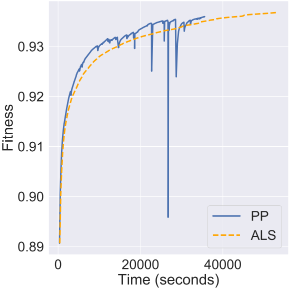
(d) Weak scaling of CP decomposition

- Experiments performed on Stampede2 with Cyclops
- Strong scaling: order $N = 6$, dimension $s = 50$, and rank $R = 6$.
- Weak scaling: on p processors, order $N = 6$ with dimension $s = \lfloor 32p^{1/6} \rfloor$, and rank $R = \lfloor 4p^{1/6} \rfloor$.
- Middle steps (subsequent approximations) up to 10X faster

Performance for Quantum Chemistry Tensor



(e) Sequential results with NumPy



(f) Parallel results with Cyclops

- Order 3 density fitting intermediate arising in quantum chemistry
- Sequential experiments: tensor size $904 \times 56 \times 56$, $R = 1000$
- Parallel experiments: tensor size $4520 \times 280 \times 280$, $R = 1800$
- $1.9 - 2.5$X speedup for sequential experiments, 1.24X speedup for parallel experiments

Summary and Conclusion

- Introduced new pairwise perturbation algorithm to approximate ALS in CP and Tucker decomposition
- Approximate sweep faster for CP by factor of $O(s^{N-2})$ and for Tucker by factor of $O(s^{N-2}/R^{N-2})$
- Both CP and Tucker ALS with dimension trees and with PP implemented using both Numpy and Cyclops[5]
- Speed-ups of about 2.5X for a range of problems on Stampede2 and Blue Waters.
- Error analysis of the algorithm will be presented at PP1 Poster Session
- For pseudocodes, analysis, and results, see arXiv:1811.10573

[5] https://github.com/LinjianMa/pairwise-perturbation

Back-up Slides

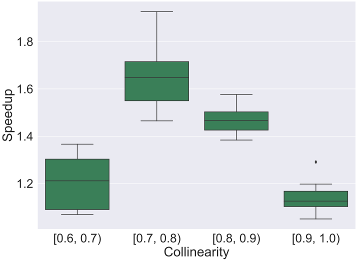
Quantum chemistry tensors

- Density fitting intermediate tensor arising in quantum chemistry: the Cholesky factor of the two-electron integral tensor.
- For an order 4 two-electron integral tensor $\boldsymbol{\mathcal{T}}$, its Cholesky factor is an order 3 tensor $\boldsymbol{\mathcal{D}}$, with their relations shown as follows:

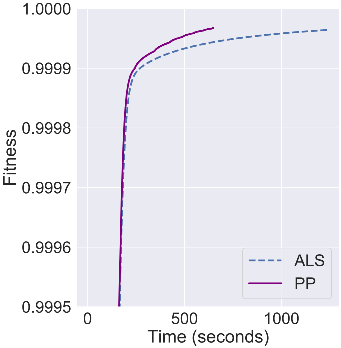
$$\boldsymbol{\mathcal{T}}(a,b,c,d) = \sum_{s=1}^{P} \boldsymbol{\mathcal{D}}(a,b,s)\boldsymbol{\mathcal{D}}(i,j,s),$$

where P is the third mode dimension size of $\boldsymbol{\mathcal{D}}$.

Results for Synthetic Tensors



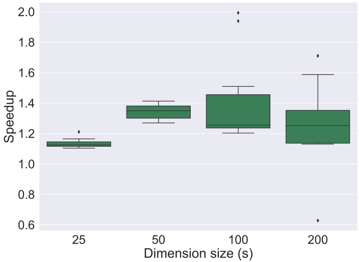
(e) Speedup-collinearity relation



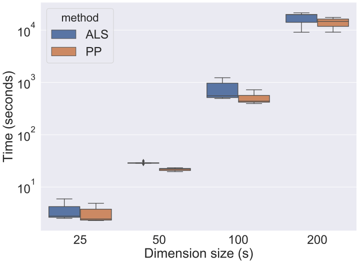
(f) Fitness-time relation for the input tensor with collinearity $[0.7, 0.8)$

- Order 3 tensor, dimension $s = 100$ and rank $R = 100$
- Each box based on 10 runs
- Overall convergence up to 1.7X faster

Results for Synthetic Tensors



(e) Speed-up with random tensors



(f) Time for experiments with random tensors

- Order 3 tensor, $s = R$
- Each box based on 10 runs
- Overall convergence up to 2.0X faster

Error Analysis: First Attempt

Consider order $N = 3$ tensor $\boldsymbol{\mathcal{X}}$, let $\boldsymbol{M}^{(3)}$ be the right-hand-sides needed to form the third factor matrix $\boldsymbol{A}^{(3)}$

- Bound columnwise error of $\tilde{\boldsymbol{M}}^{(3)}$ computed by PP middle step
- The ith factor matrix changed by $d\boldsymbol{A}^{(i)}$ since the first step of PP
- Error bound based on conditioning bound of $\boldsymbol{f}_{\boldsymbol{\mathcal{X}}} \in \mathbb{R}^s \times \mathbb{R}^s \to \mathbb{R}^s$,

$$\boldsymbol{z} = \boldsymbol{f}_{\boldsymbol{\mathcal{X}}}(\boldsymbol{u}, \boldsymbol{v}) \Rightarrow z_k = \sum_{i,j} x_{ijk} u_i v_j$$

Theorem (Columnwise Error Bound from Tensor Conditioning)

If $||d\boldsymbol{a}_k^{(l)}||_2 / ||\boldsymbol{a}_k^{(l)}||_2 \leq \epsilon$ *for* $l \in \{1, 2, 3\}$,

$$\frac{||\tilde{\boldsymbol{m}}_k^{(3)} - \boldsymbol{m}_k^{(3)}||_2}{||\boldsymbol{m}_k^{(3)}||_2} \leq \frac{\max_{\boldsymbol{u},\boldsymbol{v} \in \mathbb{S}^{s-1}} ||\boldsymbol{f}_{\boldsymbol{\mathcal{X}}}(\boldsymbol{u}, \boldsymbol{v})||_2}{\min_{\boldsymbol{y},\boldsymbol{z} \in \mathbb{S}^{s-1}} ||\boldsymbol{f}_{\boldsymbol{\mathcal{X}}}(\boldsymbol{y}, \boldsymbol{z})||_2} O(\epsilon^2).$$

Error Analysis: Second Attempt

Again, consider order $N = 3$ tensor $\boldsymbol{\mathcal{X}}$, let $\boldsymbol{M}^{(3)}$ be the right-hand-sides needed to form the third factor matrix $\boldsymbol{A}^{(3)}$

- Define $\boldsymbol{M}_{new}^{(3)} - \boldsymbol{M}^{(3)} = \boldsymbol{H}^{(1,3)} + \boldsymbol{H}^{(2,3)}$
- Define $\boldsymbol{A}_{new}^{(i)} - \boldsymbol{A}^{(i)} = \delta \boldsymbol{A}^{(i)}$
- Bound columnwise error of *approximate update* $\tilde{\boldsymbol{H}}^{(1,3)}$ to $\tilde{\boldsymbol{M}}^{(3)}$ computed by PP middle step due to change in $\boldsymbol{A}^{(1)}$

Theorem (Columnwise Error Bound from Matricization Conditioning)

For $\epsilon_k = ||\delta \boldsymbol{a}_k^{(2)}||_2 / ||\boldsymbol{a}_k^{(2)}||_2 < 1$ *and* $\hat{\boldsymbol{T}} = \boldsymbol{\mathcal{X}} \times_1 \delta \boldsymbol{a}_k^{(1)}$,

$$\frac{||\tilde{\boldsymbol{h}}_k^{(1,3)} - \boldsymbol{h}_k^{(1,3)}||_2}{||\boldsymbol{h}_k^{(1,3)}||_2} \leq \kappa(\hat{\boldsymbol{T}})\epsilon_k, \ \textit{where } \kappa(\hat{\boldsymbol{T}}) = \frac{\sigma_{\max}(\hat{\boldsymbol{T}})}{\sigma_{\min}(\hat{\boldsymbol{T}})}$$

- For $N > 3$: higher-order absolute error terms scale as $O(\epsilon_k \epsilon_l)$, but can dominate, so have no relative error bound