Extrapolated Alternating Algorithms for Approximate Canonical Polyadic Decomposition

Andersen M.S. Ang[†], Jeremy, E. Cohen[#], Le T.K. Hien[†], Nicolas Gillis[†]

† Department of Mathematics and Operational Research, Université de Mons, Belgium
# CNRS, Université de Rennes, Inria, IRISA Campus de Beaulieu, Rennes, France

ICASSP 2020
May 4-8 2020
Lecture session WE3.L6: Optimization Algorithms II

Overview

1. Problem statement
2. Existing Alternating Algorithms
3. Proposed Approaches
4. Experiments
5. Summary and Perspective

Paper information

- Paper number: WE3.L6.1
- Paper preprint: https://bit.ly/3aRn1yw
- Slide avaliable: angms.science

aCPD : Approximate Canonical Polyadic Decomposition

- Given a order p tensor $T \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$ and a natural number r, find a tensor $\hat{T}$ s.t.

$$\text{aCPD :} \quad \hat{T} = \underset{\text{rank}(G) \leq r}{\operatorname{argmin}} \left\| T - G \right\|_F^2,$$

- Rank of a tensor G defined as

$$\min \left\{ r \in \mathbb{N} \;\middle|\; \exists a_i^{(j)} \in \mathbb{R}^{n_j}, G = \sum_{i=1}^{r} \bigotimes_{j=1}^{p} a_i^{(j)} \right\}.$$

- $\otimes$ Tensor product

$$[a^{(1)} \otimes \cdots \otimes a^{(p)}]_{i_1 \ldots i_p} = \prod_{j=1}^{p} a_{i_j}^{(j)}.$$

- Motivation : A challenging task in general
  - Nonconvex problem.
  - Degeneracy and swamps.
  - Escaping saddle points.

Existing Alternating Algorithms

- Update one block at a time, while keeping others fixed
- Two categories

	Exact Block-Coordinate Descent	Approximate Block-Coordinate Descent
Subproblem	solved optimally	solved approximately
Example	Alternating Least Squares (ALS) Hierarchical ALS (HALS)	Various alternating gradient methods

- ALS update on block $A^{(j)}$

$$A^{(j)}_{\text{New}} = \underbrace{g(T, A^{(l \neq j)})}_{\text{update function}} := T_{[j]} B^{(j)\dagger}, \quad B^{j^T} = \underbrace{\bigodot_{l \neq j} A^{(l)}}_{\text{Khatri-Rao product}} .$$

- HALS : column-wise version of ALS.
- Gradient update on block $A^{(j)}$

$$A^{(j)}_{\text{New}} = A^{(j)} - \frac{1}{L^{(j)}} \left(A^{(j)} B^{j^T} - T_{[j]} \right) B^{j} .$$

Proposed approaches

- Introduce 2 algorithms for computing aCPD that make use of extrapolation in 2 different ways

	Exact Block-Coordinate Descent	Approximate Block-Coordinate Descent
Proposed	Heuristic Extrapolation and Restart (HER)	Inertial Block Proximal Gradient (iBPG)
Convergence	Only empirical	With theoretical analysis

- Goal / contribution : show when computing aCPD, extrapolation can
  - enhance empirical convergence speed in difficult cases and
  - help escaping swamps.

- This observation is not new, can trace back to work by Harshman in 70s. We provide a fresh view on these issues by using more recent optimization techniques.

Algorithm 1 iBPG for CPD

Initialization: Set $\delta_w = 0.99$, $\beta = 1.01$, $t_0 = 1$, 2 sets of initial factor matrices $(A_{-1}^{(1)}, \dots, A_{-1}^{(p)})$ and $(A_0^{(1)}, \dots, A_0^{(p)})$. Set $k = 1$.
Set $A_{\text{prev}}^{(j)} = A_{-1}^{(j)}, j = 1, \dots, p$.
Set $A_{\text{cur}}^{(j)} = A_0^{(j)}, j = 1, \dots, p$.
repeat
  for $j = 1, \dots, p$ **do**
    $t_k = \frac{1}{2}(1 + \sqrt{1 + 4t_{k-1}^2})$, $\hat{w}_{k-1} = \frac{t_{k-1} - 1}{t_k}$
    $w_{k-1}^{(j)} = \min\left(\hat{w}_{k-1}, \delta_w \sqrt{\frac{L_{k-2}^{(j)}}{L_{k-1}^{(j)}}}\right)$
    $L_k^{(j)} = \left\| (B_k^{(j)})^T B_k^{(j)} \right\|$
    repeat
      Compute two extrapolation points
      $$\hat{A}^{(j,1)} = A_{\text{cur}}^{(j)} + w_{k-1}^{(j)} \left(A_{\text{cur}}^{(j)} - A_{\text{prev}}^{(j)}\right),$$
      $$\hat{A}^{(j,2)} = A_{\text{cur}}^{(j)} + \beta w_{k-1}^{(j)} \left(A_{\text{cur}}^{(j)} - A_{\text{prev}}^{(j)}\right)$$
      Set $A_{\text{prev}}^{(j)} = A_{\text{cur}}^{(j)}$.
      Update $A_{\text{cur}}^{(j)}$ by gradient step:
      $$A_{\text{cur}}^{(j)} = \hat{A}^{(j,2)} - \frac{\left(\hat{A}^{(j,1)} (B_{k-1}^{(j)})^T - \mathcal{T}_{[j]}\right) B_{k-1}^{(j)}}{L_{k-1}^{(j)}}.$$
    until some criteria is satisfied
    Set $A_k^{(j)} = A_{\text{cur}}^{(j)}$.
  end for
  Set $k = k + 1$.
until some criteria is satisfied

- An Alternating (proximal) grad. descent algo. to solve a general noncvx. nonsmooth block separable composite optimization problem.
  - Use 2 different extrapolation pts to compute gradient and to add inertial force.
  - No restarts.
  - Flexible in the choice of the order in which the blocks are updated.
  - Theory : iBPG for aCPD satisfies the condition for sub-sequential convergence. (Details in: arxiv:1903.01818)

Algorithm 2 herALS for CPD

Initialization: Choose $\beta_0 \in (0,1)$, $\eta \geq \gamma \geq \bar{\gamma} \geq 1$, 2 sets of initial factor matrices $(A_0^{(1)}, \dots, A_0^{(p)})$ and $(Z_0^{(1)}, \dots, Z_0^{(p)})$
Set $\bar{\beta}_0 = 1$ and $k = 1$.

repeat
  for $j = 1, \dots, p$ **do**
    Update:
    $A_k^{(j)} = g\left(T, \left[Z_k^{(l<j)}, Z_{k-1}^{(l>j)}\right]\right)$
    Extrapolate:
    $Z_k^{(j)} = A_k^{(j)} + \beta_k \left(A_k^{(j)} - A_{k-1}^{(j)}\right)$
  end for
  Compute $\hat{F}_k = F(A_k^{(p)}; Z_k^{(l \neq p)})$.
  if $\hat{F}_k > \hat{F}_{k-1}$ for $k \geq 2$ **then**
    Set $Z_k^{(j)} = A_k^{(j)}$ for $j = 1, \dots, p$
    Set $\bar{\beta}_k = \beta_{k-1}$,
      $\beta_k = \beta_{k-1}/\eta$
  else
    Set $A_k^{(j)} = Z_k^{(j)}$ for $j = 1, \dots, p$
    Set $\bar{\beta}_k = \max\{1, \bar{\beta}_{k-1}\bar{\gamma}\}$,
      $\beta_k = \max\{\bar{\beta}_{k-1}, \beta_{k-1}\gamma\}$,
  end if
  Set $k = k + 1$.
until some criteria is satisfied

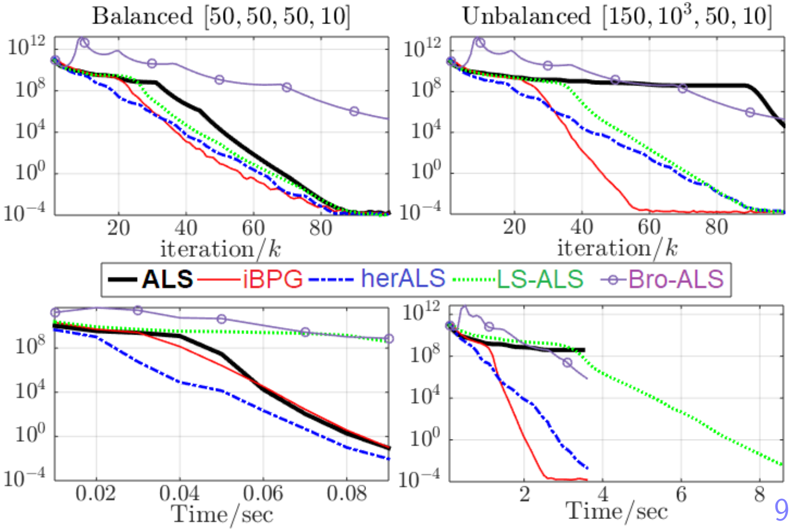
- An extrapolation of the factor estimates **between** each block update.
  - Parameters $(\eta, \gamma, \bar{\gamma})$.
  - Restarts based on criterion $\hat{F}$ which is cheap to compute.
- Restart mechanism and β_k update
  - If $\hat{F}$ decrease : we grow β.
  - If $\hat{F}$ increase : we decrease β.
  - Similar update on $\bar{\beta}$.
- Not limited to ALS, HER can accelerate any BCD algo.
- No additional computational cost : cost of one iteration of herALS is the same as one iteration of ALS, because of the use of $\hat{F}$.
- No intensive parameter tuning is needed on $(\eta, \gamma, \bar{\gamma})$.
- On Nonnegative CPD: arXiv:2001.04321

Exeriments setup

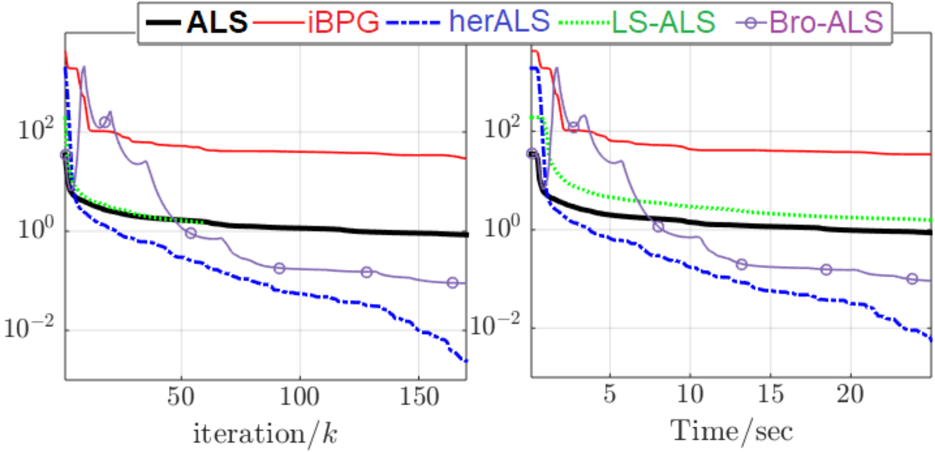
- Algorithms to compare : iBPG and herALS, and
  - **ALS** : Original un-accelerated ALS
  - Bro-ALS : accelerated ALS using Bro's acceleration, which uses a different heuristic approach to perform extrapolation.
  - LS-ALS : accelerated ALS where extrapolation sequence is computed by line search.
- Data sets : order 3 tensors, synthetic and real data from fluorescence spectroscopy and remote sensing.
- Data preprocessing : no preprocessing other than replacing NaN as 0.
- All experiments are run over 20 random initializations.
- Plotting : median of cost value over these 20 trials.

Synthetic data sets $T = \sum_{q=1}^{r} a_q^{(1)} \otimes a_q^{(2)} \otimes a_q^{(3)} + \sigma N$

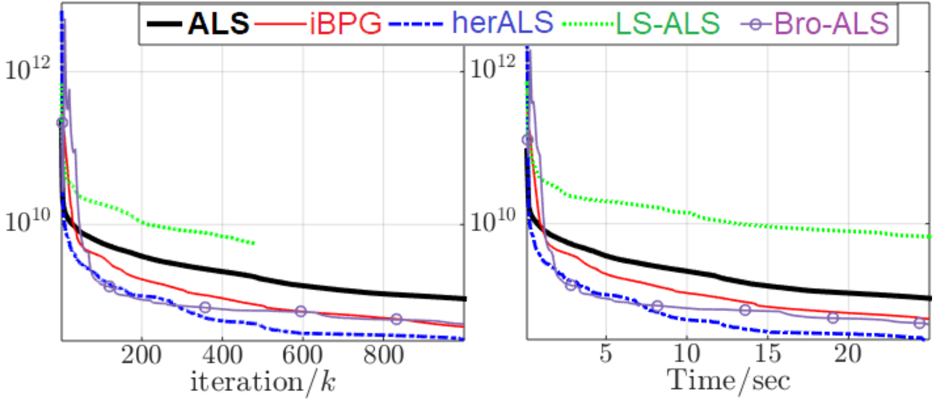
- N : zero mean unitary variance Gaussian noise
- Condition number of $A^{(j)}$ adjusted to 100 (ill-condition)
- Notation: $[I, J, K, r]$ is the tensor size and the factorization rank

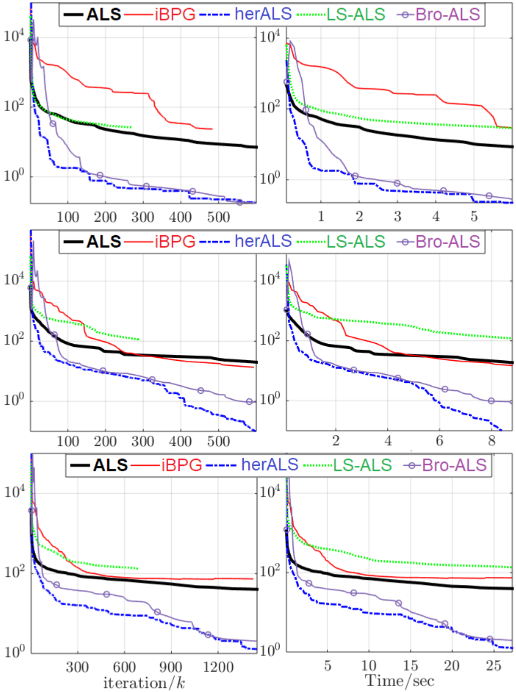


Wine data $[44, 2700, 200, 15]$



Indian Pines data $[145, 145, 200, 16]$





- Blood data $[289, 301, 41, r]$ $r \in \{3, 6, 10\}$ (top,middle,bottom)

On overall experiment results

- On synthetic data with ill-conditioned tensors, iBPG outperforms workhorse algo. **ALS**, and helps escaping swamps.
- On real data, herALS outperforms all tested methods while iBPG shows mitigated results.

Last page - summary and perspective

- Two extrapolation alternating algo. for solving aCPD : iBPG, HER-BCD.
- This work provides further practical evidence that extrapolation helps escaping swamps when computing aCPD.
- On constrained CPD, HER approach works even better for nonnegative CPD, see arXiv:2001.04321.
- Open problem : theoretical convergence analysis of HER.

Paper preprint: https://bit.ly/3aRn1yw
Slide: angms.science
End of presentation