

# Coherence Constrained Alternating Least Squares

Rodrigo Cabral Farias\*, José Henrique de Morais Goulart\*<sup>†</sup>, Pierre Comon<sup>†</sup>

\*Univ. Côte d'Azur, CNRS, I3S, 06900 Sophia-Antipolis, France,  
cabral@i3s.unice.fr

<sup>†</sup>Univ. of Grenoble Alpes, CNRS, Grenoble INP, GIPSA-Lab, F-38000 Grenoble, France,  
{henrique.goulart, pierre.comon}@gipsa-lab.fr

**Abstract**—In this paper we present a modification of alternating least squares (ALS) for tensor canonical polyadic approximation that takes into account mutual coherence constraints. The proposed algorithm can be used to ensure well-posedness of the tensor approximation problem during ALS iterates and so is an alternative to existing approaches. We conduct tests with the proposed approach by using it as initialization of unconstrained alternating least squares in difficult cases, when the underlying tensor model factors have nearly collinear columns and the unconstrained approach is prone to a degenerate behavior, failing to converge or converging slowly to an acceptable solution. The results of the tested cases indicate that by using such an initialization the unconstrained approach seems to avoid such a behavior.

## I. INTRODUCTION

The Canonical polyadic (CP) [1] tensor decomposition became a wide-spread data mining tool [2] recently, with applications in a variety of research areas, for example in neurosciences [3], chemometrics [4] and sensor arrays [5], to cite a few. One of the main reasons for its use is that it allows solving blind source separation problems with guarantees of uniqueness of the solution under mild conditions, and more importantly without statistical assumptions [6].

Although many uniqueness results are available in the exact decomposition case, in the approximation case, *i.e.* when data are noisy or when they do not exactly follow the CP model, the problem is in general ill-posed, since a solution may not exist [7]. One way to ensure existence is to impose an upper bound on the mutual coherence of the underlying CP factors [8]. This excludes solutions having nearly linearly dependent components which diverge whilst the overall model converges. In particular, nearly collinear components, which is the most usual manifestation of ill-posedness, are avoided. Furthermore, from a practical point of view, highly coherent tensor models are challenging to compute due to numerical conditioning issues. The standard algorithm for CP approximation, alternating least squares (ALS), has its convergence slowed down when the estimated factors are highly coherent, following a trajectory along solutions with quasi-linearly-dependent (typically collinear) components and seems to never converge to an adequate approximation. These trajectories, which we refer to as “degenerate,” are known in tensor community as “bottlenecks” and “swamps” [6].

To tackle the issue of ill-posedness, a penalized gradient descent approach has been proposed in [9] and applied to a problem of direction of arrival estimation with sensor arrays. In this paper, we propose to modify ALS so that it takes into account coherence constraints through its iterates. At each step of ALS we apply a suboptimal procedure relying on alternating

projections to modify the unconstrained solution such that it respects the coherence constraints. This alternating projection step is similar to the coherence constrained dictionary learning technique proposed in [10].

We present the CP approximation problem under coherence constraints and the unconstrained ALS approach in Section II. In Section III, our alternating approach with projections is proposed. Section IV presents a numerical example where the condition from [8] is used to ensure well-posedness. We also test the algorithm as an initialization for standard ALS in random tensor and sensor array examples. In both cases, we consider difficult instances with nearly collinear components where a randomly initialized ALS is prone to degenerate trajectories. In Section V we present our conclusions.

*a) Notation:* Tensors are denoted in upper-case bold calligraphic letters  $\mathcal{X}$ , matrices in upper-case bold letters  $\mathbf{X}$ , vectors in lower-case bold letters  $\mathbf{x}$  and scalars in lower-case letters  $x$ . Notations  $\|\cdot\|_F$  and  $\|\cdot\|_2$  stand for the Frobenius norm and  $\ell_2$  vector norm, respectively. The symbols  $\otimes$ ,  $\odot$  and  $\square$  are used for tensor, Khatri-Rao (column-wise Kronecker) and Hadamard products, respectively.  $\text{Diag}\{\mathbf{x}\}$  denotes a diagonal tensor or matrix with entries given by  $\mathbf{x}$ . The superscript  $\mathbf{X}^T$  denotes matrix transpose and  $\mathbf{X}^\dagger$  denotes the pseudo-inverse.

## II. TENSOR CP DECOMPOSITION WITH COHERENCE CONSTRAINTS AND ALTERNATING LEAST SQUARES

A tensor CP [11], [12] approximation of rank  $R$  and order 3 with constraints on the mutual coherences can be expressed as the following optimization problem:

$$\begin{aligned} \min_{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathcal{D}} \quad & \|\mathcal{Y} - (\mathbf{A} \otimes \mathbf{B} \otimes \mathbf{C}) \cdot \mathcal{D}\|_F^2 \\ \text{subject to} \quad & \mathbf{f}[\mu(\mathbf{A}), \mu(\mathbf{B}), \mu(\mathbf{C})] \leq \boldsymbol{\mu}_{\max}, \end{aligned} \quad (1)$$

where  $\mathcal{Y}$  is the tensor to be approximated,  $(\mathbf{A} \otimes \mathbf{B} \otimes \mathbf{C}) \cdot \mathcal{D}$  is the tensor CP model in the notation of [13],  $\mathbf{A} \in \mathbb{R}^{I \times R}$ ,  $\mathbf{B} \in \mathbb{R}^{J \times R}$  and  $\mathbf{C} \in \mathbb{R}^{K \times R}$  are the CP matrix factors with normalized components  $\|\mathbf{a}_r\|_2^2 = \|\mathbf{b}_r\|_2^2 = \|\mathbf{c}_r\|_2^2 = 1$ ,  $\forall r \in \{1, \dots, R\}$ ,  $\mathcal{D}$  is a diagonal tensor containing the weights of the rank-one components  $\mathcal{D} = \text{Diag}\{\lambda_1, \dots, \lambda_R\}$  and  $\mu(\cdot)$  is the mutual coherence function:

$$\mu(\mathbf{X}) = \max_{i \neq j} |G_{ij}^{\mathbf{X}}|, \quad (2)$$

where  $\mathbf{G}^{\mathbf{X}}$  is the Gram matrix  $\mathbf{G}^{\mathbf{X}} = \mathbf{X}^T \mathbf{X}$  of  $\mathbf{X}$ , which is assumed to have normalized columns. The (generally vector-valued) function  $\mathbf{f}$  defines the constraint function to be used on the coherences and  $\boldsymbol{\mu}_{\max}$  is a constant vector. This function can take one of the two following forms:

$$\mathbf{f}[\mu(\mathbf{A}), \mu(\mathbf{B}), \mu(\mathbf{C})] = \mu(\mathbf{A})\mu(\mathbf{B})\mu(\mathbf{C}) \leq \boldsymbol{\mu}_{\max} \quad (3)$$

This work was partially supported by the European Research Council under the European Programme FP7/2007-2013, Grant AdG-2013-320594 “DECODA.”

or

$$\begin{aligned} f[\mu(\mathbf{A}), \mu(\mathbf{B}), \mu(\mathbf{C})] &= [\mu(\mathbf{A}) \mu(\mathbf{B}) \mu(\mathbf{C})]^\top \\ &\leq [\mu_{\mathbf{A}}^{\max} \mu_{\mathbf{B}}^{\max} \mu_{\mathbf{C}}^{\max}]^\top. \end{aligned} \quad (4)$$

The first form of the constraint (3) can be used with  $\mu_{\max} = \frac{\nu}{R-1}$  if we want to respect the sufficient conditions in [8] so that a solution for the CP approximation problem (for  $R \geq 2$ ) always exists. If we simply want to bound the coherences, either because this is a prior information on the model or because we hope that by doing so we will prevent the underlying decomposition algorithm from falling into a degenerate trajectory, then we use the second form (4).

The minimization problem (1) has been solved in [9] using gradient descent. In what follows we propose to solve it in a suboptimal manner using an alternating approach, in a similar way to ALS but integrating coherence constraints. We will consider either one of the two constraints above since they lead to a similar problem in the alternating approach.

#### A. Alternating optimization

The standard ALS algorithm [14, pp. 61-62] for tensor CP approximation corresponds to solving the unconstrained optimization problem (1) using block coordinate descent where the blocks are the matrix factors. If we add one of the two types of coherence constraints above, the optimization problem to be solved to generate the  $k$ th update  $\hat{\mathbf{A}}_k$  is

$$\begin{aligned} \min_{\mathbf{A}} \quad & \|\mathcal{Y} - (\mathbf{A} \otimes \hat{\mathbf{B}}_{k-1} \otimes \hat{\mathbf{C}}_{k-1}) \cdot \hat{\mathcal{D}}_{k-1}\|_F^2 \\ \text{subj. to} \quad & \mu(\mathbf{A}) \leq \mu_{\mathbf{A}}^{k-1} \text{ and } \|\mathbf{a}_r\|_2^2 = 1, \forall r. \end{aligned} \quad (5)$$

where

$$\mu_{\mathbf{A}}^{k-1} = \begin{cases} \nu \left[ \mu(\hat{\mathbf{B}}_{k-1}) \mu(\hat{\mathbf{C}}_{k-1}) (R-1) \right]^{-1}, & \text{if (3) is used} \\ \mu_{\mathbf{A}}^{\max}, & \text{if (4) is used.} \end{cases}$$

Similar optimization problems are obtained for updates  $\hat{\mathbf{B}}_k$  and  $\hat{\mathbf{C}}_k$ , while the diagonal elements  $\hat{\lambda}_k$  of update  $\hat{\mathcal{D}}_k$  can be easily obtained with vectorization of the data in the least squares problem

$$\hat{\lambda}_k = \left[ \mathbf{M}_{CBA}^\top \mathbf{M}_{CBA} \right]^{-1} \left( \mathbf{M}_{CBA}^\top \text{vec}(\mathcal{Y}) \right), \quad (6)$$

where<sup>1</sup>  $\mathbf{M}_{CBA} = \hat{\mathbf{C}}_k \odot \hat{\mathbf{B}}_k \odot \hat{\mathbf{A}}_k$ .

Using the matrix unfoldings  $\mathbf{Y}_{(1)}$ ,  $\mathbf{Y}_{(2)}$  and  $\mathbf{Y}_{(3)}$  of  $\mathcal{Y}$  as defined in [12]<sup>2</sup>, the updates of standard ALS are given by

$$\begin{aligned} \hat{\mathbf{A}}_k &= \mathbf{Y}_{(1)} [(\hat{\mathbf{C}}_{k-1} \text{Diag}\{\hat{\lambda}_{k-1}\}) \odot \hat{\mathbf{B}}_{k-1}]^\dagger, \\ \hat{\mathbf{B}}_k &= \mathbf{Y}_{(2)} (\hat{\mathbf{C}}_{k-1} \odot \hat{\mathbf{A}}_k)^\dagger, \\ \hat{\mathbf{C}}_k &= \mathbf{Y}_{(3)} (\hat{\mathbf{B}}_k \odot \hat{\mathbf{A}}_k)^\dagger, \end{aligned} \quad (7)$$

just requiring normalization of the components after each update. Note, however, that the coherence-constrained version (5) of the problem is much harder: it does not have known analytical solution and the constraints define a non-convex set of allowable solutions, thus making the overall problem a non-convex one. This leads us to develop a specific update method which will give a suboptimal solution to the underlying least squares problem under coherence constraints.

<sup>1</sup>This implies  $\mathbf{M}_{CBA}^\top \mathbf{M}_{CBA} = (\hat{\mathbf{C}}_k^\top \hat{\mathbf{C}}_k) \square (\hat{\mathbf{B}}_k^\top \hat{\mathbf{B}}_k) \square (\hat{\mathbf{A}}_k^\top \hat{\mathbf{A}}_k)$ . Note that the right-hand side of this identity is cheaper to compute.

<sup>2</sup>The correspondence among elements of the tensor and of each unfolding is  $[\mathcal{X}]_{i,j,k} = [\mathbf{X}_{(1)}]_{i,m_1} = [\mathbf{X}_{(2)}]_{j,m_2} = [\mathbf{X}_{(3)}]_{k,m_3}$ , where  $m_1 = (k-1)J + j$ ,  $m_2 = (k-1)I + i$  and  $m_3 = (j-1)I + i$ .

### III. ALTERNATING LEAST SQUARES WITH COHERENCE CONSTRAINTS

Since the problem is similar for factors  $\mathbf{A}$ ,  $\mathbf{B}$  and  $\mathbf{C}$ , we present here only its version for  $\mathbf{A}$ . Using the first matrix unfolding  $\mathbf{Y}_{(1)}$  of  $\mathcal{Y}$  and making explicit the constraint with the Gram matrix  $\mathbf{G}^{\mathbf{A}}$  of  $\mathbf{A}$ , the update  $\hat{\mathbf{A}}_k$  is given as the solution of the following linear least squares problem under constraints:

$$\min_{\mathbf{A}} \|\mathbf{Y}_{(1)} - \mathbf{A} \mathbf{M}_{DCB}\|_F^2 \text{ subj. to } \begin{cases} |G_{ij}^{\mathbf{A}}| \leq \mu_{\mathbf{A}}^{k-1}, & i \neq j, \\ G_{ij}^{\mathbf{A}} = 1, & i = j, \end{cases} \quad (8)$$

where  $\mathbf{M}_{DCB} = (\hat{\mathbf{C}}_{k-1} \text{Diag}\{\hat{\lambda}_{k-1}\}) \odot \hat{\mathbf{B}}_{k-1}$ . Since the matrix  $\mathbf{A}$  can be written as a product  $\mathbf{A} = \mathbf{T} \mathbf{A}^c$  of an orthogonal basis  $\mathbf{T} \in \mathbb{R}^{I \times R}$  and coefficients on this basis  $\mathbf{A}^c \in \mathbb{R}^{R \times R}$  and the Gram matrix depends only on the coefficients  $\mathbf{G}^{\mathbf{A}} = (\mathbf{A}^c)^\top \mathbf{A}^c$ , we can rewrite problem (8) as a function of  $\mathbf{T}$  and  $\mathbf{A}^c$  as follows:

$$\min_{\mathbf{T}, \mathbf{A}^c} \|\mathbf{Y}_{(1)} - \mathbf{T} \mathbf{A}^c \mathbf{M}_{DCB}\|_F^2 \quad (9)$$

$$\text{subj. to } \mathbf{T}^\top \mathbf{T} = \mathbf{I}_R \text{ and } \begin{cases} |[(\mathbf{A}^c)^\top \mathbf{A}^c]_{ij}| \leq \mu_{\mathbf{A}}^{k-1}, & i \neq j, \\ [(\mathbf{A}^c)^\top \mathbf{A}^c]_{ij} = 1, & i = j. \end{cases}$$

To address this problem, we compute a suboptimal solution in a two-step fashion, as described below.

#### A. Suboptimal solution for $\mathbf{A}^c$ through alternating projections

Even if we are given an initial estimate  $\hat{\mathbf{T}}$  of  $\mathbf{T}$ , obtaining an estimate  $\hat{\mathbf{A}}^c$  of  $\mathbf{A}^c$  is difficult. Hence, we propose following a projection approach, similarly to [10]:

1) *Unconstrained solution*: we first retrieve the unconstrained least squares solution for  $\mathbf{A}$  using (7).

2) *Projection of the Gram matrix*: then, we construct the Gram matrix  $\mathbf{G}^{\mathbf{A}} = (\mathbf{A})^\top \mathbf{A}$  and project it onto the intersection of two sets:

- $\mathcal{C} = \{\mathbf{G} \mid G_{ii} = 1, |G_{ij}| \leq \mu_{\mathbf{A}}^{k-1}, i \neq j\}$ ;
- $\mathcal{S} = \{\mathbf{G} \mid \mathbf{G} \succeq 0\}$ ,

where  $\mathbf{X} \succeq 0$  means that  $\mathbf{X}$  is a positive semi-definite matrix. Note that these constraints ensure that the resulting matrix is a correlation matrix respecting the coherence constraints.

We can use alternating projections as proposed in [10], but since both sets are convex we can use Dykstra's projection algorithm [15] as presented in [16], which guarantees an asymptotic convergence to the optimal projection. The steps of the Dykstra's projection are described in Algorithm 1 where the first inner projection  $P_{\mathcal{C}}^\mu(\mathbf{G})$  acts element-wise on  $\mathbf{G}$  as follows:

$$[P_{\mathcal{C}}^\mu(\mathbf{G})]_{ij} = \begin{cases} 0, & |G_{ij}| = 0 \text{ and } i \neq j, \\ \frac{G_{ij}}{|G_{ij}|} \min\{\mu, |G_{ij}|\}, & |G_{ij}| \neq 0 \text{ and } i \neq j, \\ 1, & i = j, \end{cases} \quad (10)$$

and the second is a spectral projection

$$P_{\mathcal{S}}(\mathbf{G}) = \mathbf{E} P_+(\mathbf{\Lambda}) \mathbf{E}^\top, \quad (11)$$

where  $\mathbf{E}$  is the matrix of eigenvectors of  $\mathbf{G}$ ,  $\mathbf{\Lambda}$  is the diagonal matrix with its eigenvalues and  $P_+(\mathbf{X})$  is the entry-wise projection onto the nonnegative orthant

$$[P_+(\mathbf{X})]_{ij} = \begin{cases} X_{ij}, & X_{ij} \geq 0, \\ 0, & \text{otherwise.} \end{cases} \quad (12)$$

**Algorithm 1** Dykstra's projection on the set of correlation matrices with upper-bound constraint -  $\overline{\mathbf{G}} = P_{\text{coh}}(\mathbf{G}, \mu_{\text{max}}, n_{\text{proj}})$

**Require:**  $\mathbf{G}, \mu_{\text{max}}$   
Initialize:  $\mathbf{R} \leftarrow \mathbf{0}_R, \mathbf{S} \leftarrow \mathbf{0}_R, \mathbf{G}_{\mathcal{S}} \leftarrow \mathbf{G}, n \leftarrow 1$ ,  
**while**  $n < n_{\text{proj}}$  **do**  
1:  $\overline{\mathbf{G}} \leftarrow \mathbf{G}_{\mathcal{S}} - \mathbf{R}$   
2:  $\mathbf{G}_{\mathcal{C}} \leftarrow P_{\mathcal{C}}^{\mu_{\text{max}}}(\overline{\mathbf{G}})$  with  $P_{\mathcal{C}}^{\mu_{\text{max}}}(\cdot)$  given by (10)  
3:  $\mathbf{R} \leftarrow \mathbf{G}_{\mathcal{C}} - \overline{\mathbf{G}}$   
4:  $\overline{\mathbf{G}} \leftarrow \mathbf{G}_{\mathcal{C}} - \mathbf{S}$   
5:  $\mathbf{G}_{\mathcal{S}} \leftarrow P_{\mathcal{S}}(\overline{\mathbf{G}})$  with  $P_{\mathcal{S}}(\cdot)$  given by (11)  
6:  $\mathbf{S} \leftarrow \mathbf{G}_{\mathcal{S}} - \overline{\mathbf{G}}$   
7:  $n \leftarrow n + 1$   
**end while**

3) Retrieval of the square root matrix from Gram matrix: after obtaining the projection of  $\mathbf{G}^A$  onto  $\mathcal{C} \cap \mathcal{S}$ , denoted by  $\overline{\mathbf{G}}^A$ , an estimate of  $\mathbf{A}^c$  respecting the constraints is obtained by taking the matrix square root of  $\overline{\mathbf{G}}^A$

$$\hat{\mathbf{A}}^c = \mathbf{E}_G \mathbf{\Lambda}^{1/2} \mathbf{E}_G^T, \quad (13)$$

where  $\mathbf{E}_G$  is a matrix with the eigenvectors of  $\overline{\mathbf{G}}^A$  and  $\mathbf{\Lambda}^{1/2}$  is a diagonal matrix with the square roots of its eigenvalues.

For the case where  $R > I$ , one can use instead a Cholesky factorization of  $\overline{\mathbf{G}}^A$  with total pivoting to obtain an upper-triangular matrix  $\hat{\mathbf{A}}^c$  whose last  $R - I$  rows are discarded. (In this case,  $\mathbf{T}$  is  $I \times I$  and  $\hat{\mathbf{A}}^c$  is  $I \times R$ .) For  $I \leq R$ , this alternative is more costly than taking the matrix square root, since matrices  $\mathbf{E}_G$  and  $\mathbf{\Lambda}$  are already evaluated at the alternating-projections step.

### B. Solution for $\mathbf{T}$

Given  $\hat{\mathbf{A}}^c$ , the optimization problem to be solved for  $\mathbf{T}$  is known as the orthogonal Procrustes problem and has the following closed-form solution [17]:

$$\hat{\mathbf{T}} = \mathbf{U}\mathbf{V}^T, \quad (14)$$

where  $\mathbf{U}$  and  $\mathbf{V}$  are the matrices of left and right singular vectors obtained from the SVD of  $\mathbf{Y}_{(1)} \mathbf{M}_{DCB}^T \hat{\mathbf{A}}^c$ .

### C. Approximate least squares with coherence constraints

This approximate two-step solution could be iterated in an alternating manner in order to refine  $\hat{\mathbf{A}}^c$  and  $\hat{\mathbf{T}}$ . This is done by solving for  $\mathbf{A}^c$  the unconstrained version of (9) with the last estimate of  $\mathbf{T}$  and resuming the procedure from step 2) of Subsection III-A, now computing the Gramian as  $\mathbf{G}^A = (\hat{\mathbf{A}}^c)^T \hat{\mathbf{A}}^c$ . However, in practice we have observed that the described procedure without refinement seems to give reasonable estimates. The resulting procedure, called coherence-constrained least squares (CC-LS), is described in Algorithm 2.

We apply Algorithm 2 in an alternating way (with respect to  $\mathbf{A}, \mathbf{B}, \mathbf{C}$ ) to obtain a coherence-constrained alternating least squares (CC-ALS) scheme, as described in Algorithm 3. Note that this algorithm can be easily modified to work with complex valued CP decompositions: it is only required to replace all matrix transposes with Hermitian matrix transposes.

**Algorithm 2** Coherence constrained least squares approximate solution:  $\hat{\mathbf{X}} = \text{CC-LS}(\mathbf{Y}, \mathbf{M}, \mu_{\text{max}}, n_{\text{proj}})$

**Require:**  $\mathbf{Y}, \mathbf{M}, \mu_{\text{max}}, n_{\text{proj}}$   
1: Obtain the unconstrained solution:  $\hat{\mathbf{X}} \leftarrow \mathbf{M}^\dagger \mathbf{Y}$ .  
2: Evaluate the Gram matrix:  $\mathbf{G} \leftarrow \hat{\mathbf{X}}^T \hat{\mathbf{X}}$   
3: Apply approximate projection:  $\overline{\mathbf{G}} \leftarrow P_{\text{coh}}(\mathbf{G}, \mu_{\text{max}}, n_{\text{proj}})$ .  
5: Evaluate square root matrix (13):  $\hat{\mathbf{A}}^c \leftarrow \mathbf{E}_G \mathbf{\Lambda}^{1/2} \mathbf{E}_G^T$   
6: Compute SVD:  $\mathbf{Y}_{(1)} \mathbf{M}_{DCB}^T \hat{\mathbf{A}}^c = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$   
7: Obtain  $\hat{\mathbf{T}}$  with (14):  $\hat{\mathbf{T}} \leftarrow \mathbf{U}\mathbf{V}^T$   
8: Evaluate  $\hat{\mathbf{X}} \leftarrow \hat{\mathbf{T}} \hat{\mathbf{A}}^c$

**Algorithm 3** Coherence constrained suboptimal ALS -  $[\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}, \hat{\mathbf{D}}] = \text{CC-ALS}(\mathcal{Y}, \mu_{\text{max}}, n_{\text{iter}}, n_{\text{proj}})$

**Require:**  $\mathcal{Y}, \mu_{\text{max}}, n_{\text{iter}}, n_{\text{proj}}$   
Initialize:  $\hat{\mathbf{A}}_0, \hat{\mathbf{B}}_0$  and  $\hat{\mathbf{C}}_0$  with normalized columns and  $\hat{\mathbf{D}}_0$  a diagonal tensor,  $k \leftarrow 1$ .  
Unfold: create unfoldings  $\mathbf{Y}_{(1)}, \mathbf{Y}_{(2)}$  and  $\mathbf{Y}_{(3)}$  of  $\mathcal{Y}$ .  
**while**  $k < n_{\text{iter}}$  or criteria on cost function is fulfilled: **do**  
1: Write  $\mathbf{M}_{DCB}$  and update  $\mu_{\mathbf{A}}^k$  if required.  
2: Update  $\hat{\mathbf{A}}_k = \text{CC-LS}(\mathbf{Y}_{(1)}, \mathbf{M}_{DCB}, \mu_{\mathbf{A}}^k, n_{\text{proj}})$   
3-6: Update  $\hat{\mathbf{B}}_k$  and  $\hat{\mathbf{C}}_k$  similarly.  
7: Update  $\hat{\mathbf{D}}_k$  with (6).  
8:  $k \leftarrow k + 1$   
**end while**

## IV. SIMULATIONS

In this section we apply CC-ALS (Algorithm 3) to a few examples. To compare CPD estimates, we evaluate the congruence [9]

$$\phi = \text{Tr} \left[ (\hat{\mathbf{A}}\mathbf{A}^T)(\hat{\mathbf{B}}\mathbf{B}^T)(\hat{\mathbf{C}}\mathbf{C}^T) \right] / R \quad (15)$$

between each estimated CP model and the ground truth. When  $\mathbf{A}, \mathbf{B}$  and  $\mathbf{C}$  have normalized columns, this measure of approximation quality ranges from 0 (worst) to 1 (best). Also, the numerical sensitivity of a CPD with respect to its rank-one components is measured with the condition number  $\kappa$  of [18].

### A. Resulting factors with random correlated components

1) *Estimation of ill-conditioned CPD:* we first test CC-ALS in a simple case with random factors to illustrate the effect of the constraint on the resulting factors. We consider here the coherence constraint (3) which ensures the existence of a tensor approximation solution, i.e.,  $\mu_{\text{max}} = 1/(R - 1)$ .

a) *CP model generation:* we construct a  $6 \times 6 \times 6$  rank-4 CP model  $\mathcal{X}$ , and then add a noise tensor  $\mathcal{N}$  with independent and identically distributed (i.i.d.) zero-mean Gaussian elements of small standard deviation  $\sigma = 10^{-4}$ . Elements of matrix factor  $\mathbf{C}$  are i.i.d. standard Gaussian. Factors  $\mathbf{A}$  and  $\mathbf{B}$  have three columns drawn likewise, while the fourth one is a copy of the third plus a centered Gaussian i.i.d. perturbation with  $\sigma_w = 0.1$ .  $\mathcal{D}$  receives the product of the norms of the components for different factors and the factors are then normalized. Hence, both  $\mathbf{A}$  and  $\mathbf{B}$  have two nearly collinear columns.

b) *Initialization and CC-ALS parameters:* CC-ALS is initialized with random factors  $\mathbf{A}, \mathbf{B}$  and  $\mathbf{C}$  with i.i.d. standard Gaussian elements and they are further normalized as the CP model. We execute 2000 iterations and the number of Dykstra's projection iterations  $n_{\text{proj}}$  is set to 5.

In Fig. 1, we show the resulting estimates of the nearly collinear columns of  $\mathbf{A}$  and  $\mathbf{B}$ , and also the estimates of the

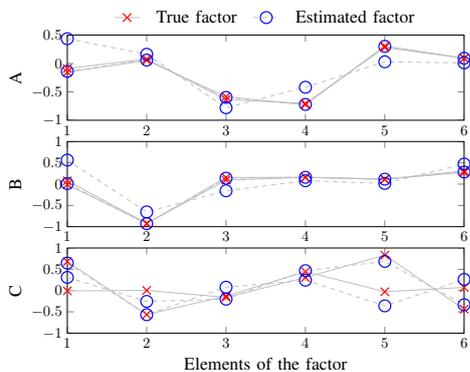


Fig. 1. An illustration of CC-ALS for the CP approximation of a rank-4  $6 \times 6 \times 6$  tensor. The original CP factors are random, with two nearly collinear columns in  $\mathbf{A}$  and  $\mathbf{B}$ . Initial CC ALS factors are also generated randomly. The coherence is constrained by condition (3).

corresponding columns of  $\mathbf{C}$ . The other columns are nearly perfectly estimated, and are thus omitted. The coherence of the original CP model is 0.610, while the bound imposes a much lower coherence of  $1/3$ . The congruence and coherence of the CC-ALS estimate are  $\phi \approx 0.86$  and  $\mu = 1/3$ , respectively, and thus the constraint is met. So, as expected, CC-ALS yields perturbed versions of the nearly collinear components in  $\mathbf{A}$  and  $\mathbf{B}$ . These perturbations “spread” the components so that the constraint is satisfied. We also see that they induce errors on the corresponding components of  $\mathbf{C}$ , even though they are not nearly collinear.

2) *Ensuring existence via coherence constraint:* In the second example, we generate a random  $4 \times 4 \times 2$  rank-4 tensor  $\mathcal{X}$  by independently drawing the entries of its factors from a standard Gaussian distribution. We add a noise tensor  $\mathcal{N}$  generated as in the previous example, but this time with  $\sigma = 10^{-1}$ . We pick a joint realization of  $\mathcal{X}$  and  $\mathcal{N}$  such that  $\mathbf{Y}_1$  is nonsingular and  $\mathbf{Y}_2\mathbf{Y}_1^{-1}$  has complex eigenvalues, where  $\mathbf{Y}_i$  is the  $i$ th frontal slab of  $\mathcal{Y} = \mathcal{X} + \sigma\mathcal{N}$ . This implies  $\mathcal{Y}$  has no best rank-4 approximation [19]. The condition number of the “true” CPD of  $\mathcal{X}$  is  $\kappa \approx 37$ , while its coherence is  $\mu \approx 0.49$ . Running ALS for 4000 iterations with a random initialization yields a solution with condition number  $\kappa_{\text{ALS}} \approx 617$  and coherence  $\mu_{\text{ALS}} \approx 0.78$ , while the measured congruence was  $\phi_{\text{ALS}} \approx 0.61$ . If we perform more iterations, then  $\kappa_{\text{ALS}} \rightarrow \infty$  and  $\mu_{\text{ALS}} \rightarrow 1$ , because no solution exists. By contrast, CC-ALS behaves stably, reaching after 4000 iterations a CPD with condition number  $\kappa_{\text{CC-ALS}} \approx 28$ , (imposed) coherence  $\mu_{\text{CC-ALS}} \approx 1/3$  and congruence  $\phi_{\text{CC-ALS}} \approx 0.88$ , thus significantly outperforming ALS.

### B. Randomly generated factors with correlated columns

In this example, we use CC-ALS as an initialization of ALS to avoid convergence issues (“bottlenecks” and “swamps”). Here, a solution exists by construction since no noise is added, but the CPD is ill-conditioned due to nearly collinear columns on its factors. Thus, we impose a fixed coherence bound.

a) *Rank 4:* factors of a  $6 \times 6 \times 6$  rank-4 CPD are generated as in Sec. IV-A1, but now forcing  $\mathbf{A}$  and  $\mathbf{B}$  to have three nearly collinear columns. 100 realizations are generated and, for each one of them, ALS is run for 2000 iterations, with 50 random initializations. For the same generated models and initializations, we run also 100 iterations of CC-ALS (with  $\mu_{\text{max}} = [0.93 \ 0.93 \ 0.93]^T$ ) followed by 1900 iterations of ALS. Finally, one run of ALS (2000 iterations) initialized with an

algebraic solution via a generalized eigenvalue decomposition (GEVD) [20] is also performed. This algebraic solution is computed using Tensorlab [21]. Fig. 2(a) shows an histogram of the resulting congruences. Visibly, ALS is much more likely to yield good estimates being initialized by CC-ALS or GEVD than using a random initial point. Thus, CC-ALS can often “steer” the algorithm into the vicinity of a global minimum, almost as effectively as the algebraic initialization, whose good performance is due to the absence of noise. Fig. 2(b) shows the empirical cumulative distribution function (ECDF) of  $1/\kappa$  measured for the ground truth and for the algorithms’ estimates. Clearly, the outcomes of randomly initialized ALS are much less reliable numerically than the others, whose distributions of  $1/\kappa$  are close to the true one.

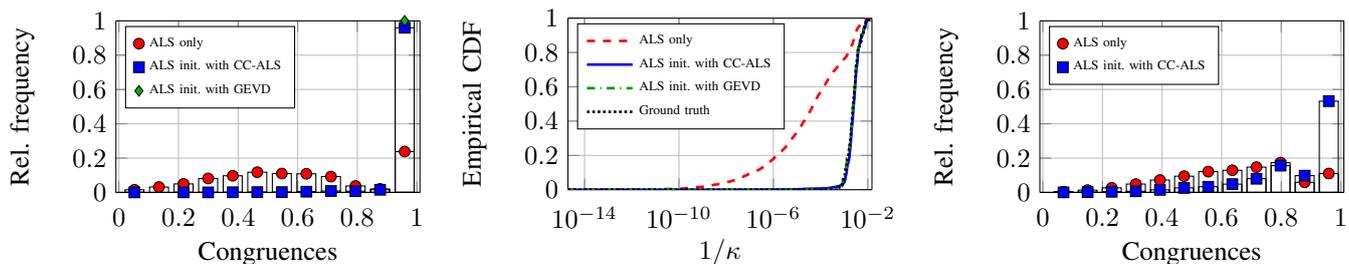
b) *Rank 7:* the same procedure is followed for random rank-7 tensors, still with  $\mathbf{A}$  and  $\mathbf{B}$  having three nearly collinear columns. Here, CC-ALS imposes the bound  $\mu_{\text{max}} = [0.98 \ 0.98 \ 0.90]^T$ , and 10 iterations of the procedure described in Subsections III-A–III-B are performed. An histogram of the resulting congruences is shown in Fig. 2(c). As expected, performance now degrades, since this is a much more challenging scenario. Nevertheless, initializing with CC-ALS still brings a significant improvement, whereas algebraic solutions typically do not apply because the rank exceeds all tensor dimensions.

### C. Sensor array

A more practical example of application is to sensor arrays. By forming a sensor array with subarrays which are simply translated versions of each other, the measurements can be stored as a third-order tensor where the element  $\mathcal{Y}_{ijk}$  corresponds to the  $k$ th time sample of the signal measured at the  $i$ th sensor of the  $j$ th subarray. Under the assumption of far-field propagation and narrowband signals,  $\mathcal{Y}$  follows a CP model where  $R$  is the number of signal sources. The columns of  $\mathbf{A}$  and  $\mathbf{B}$  carry information on the array structure and on the directions of arrival of the source signals, while the columns of  $\mathbf{C}$  are time samples of the sources, see [5], [9] for details.

c) *CP model:* here we use a sensor array simulation setting similar to [9, Section VII]. Data generated in this setting follows a rank-3 CP model of dimensions  $4 \times 3 \times 6$ . It has a very high coherence (around 0.99), while its condition number is  $\kappa \approx 5.46 \times 10^{15}$ . In [9], a regularized formulation is used to impose the bound  $\mu_{\text{max}} = 1/(R-1)$ , which guarantees well-posedness of the problem. A gradient descent (GD) algorithm is proposed to address the resulting optimization problem. Differently from [9], we consider that all three sources come from close directions  $\boldsymbol{\theta} = [83^\circ \ 80^\circ \ 75^\circ]^T$  (not only two) so that we have a matrix factor  $\mathbf{A}$  with 3 nearly collinear columns. In practice, such a quasi-collinearity causes ALS to fall very often on degenerate trajectories.

d) *Algorithms parameters:* because no noise is added, existence is guaranteed, and hence we do not impose  $\mu_{\text{max}} = 1/(R-1)$ . Rather, we use the more relaxed constraint  $\mu_{\text{max}} = [0.85 \ 0.99 \ 0.99]^T$ , whose interpretation is as follows. We impose a certain degree of separation of source directions (constraint on  $\mathbf{A}$ ), while we allow the translation vectors to be nearly collinear (constraint on  $\mathbf{B}$ ) and the sources to be highly correlated (constraint on  $\mathbf{C}$ ). We now compare a randomly initialized ALS solution with the version initialized using CC-ALS, but also with another version initialized with the GD algorithm of [9]. Both ALS and CC-ALS are modified to work with complex values, and perform a total number of 15000



(a) Normalized histogram of congruences ( $R = 4$ ) (b) ECDF of reciprocal of cond. number ( $R = 4$ ) (c) Normalized histogram of congruences ( $R = 7$ )

Fig. 2. Results for 100 different realizations of an ill-conditioned  $6 \times 6 \times 6$  rank- $R$  CP model in which  $\mathbf{A}$  and  $\mathbf{B}$  have three nearly collinear columns.

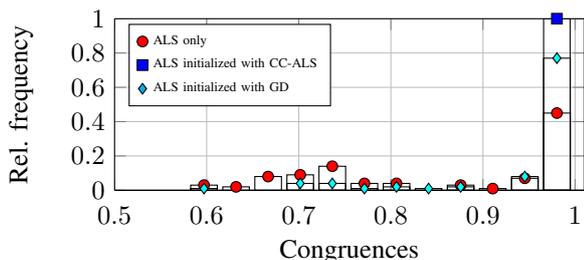


Fig. 3. Histogram of congruences for 100 runs of ALS initialized randomly (circles), by CC-ALS (squares) and GD (diamond). CC-ALS and GD use the same initial solution as the randomly initialized ALS. These results concern a rank-3 approximation of a  $4 \times 3 \times 6$  tensor arising from a simulated antenna array which receives signals from far-field sources at  $83^\circ$ ,  $80^\circ$  and  $75^\circ$  from the reference axis.

iterations, of which 6000 are performed by CC-ALS when it is used. The maximum number of iterations set for GD was 2000. Initializations are defined as in the previous example but with complex Gaussian samples, and are the same for (randomly initialized) ALS, CC-ALS and GD.

The histogram of resulting congruences is given in Fig. 3. Once again, CC-ALS initialization markedly improves ALS performance. So does GD initialization, but to a smaller extent and at a larger cost: the mean times of the CC-ALS-initialized and GD-initialized versions are, respectively, 17.9 and 26.9 seconds.

## V. CONCLUSIONS

In this paper, we have proposed a modified version of ALS for CP decomposition under coherence constraints, termed CC-ALS. As shown in the literature, the use of coherence constraints guarantees that the tensor approximation is a well-posed problem. The algorithm proposed here is an alternative to existing gradient descent algorithms which typically require careful tuning of step-size parameters and converge slowly. We have found that in difficult CP approximation examples, where standard ALS is prone to degenerate trajectories, its initialization with the result of CC-ALS allows avoiding them most of the time. In practice for such examples, we have observed that fixing the coherence constraints to high enough values seem to work better in general than using the sufficient conditions on well-posedness such as those in [8]. However if we increase the coherence constraint to a value too close to one, then the CC-ALS inherits the degenerate behavior of ALS and an interaction between this behavior and the inner alternating projections of CC-ALS makes its trajectory completely erratic. As a consequence, one track for possible

further investigation is to study how high this coherence constraint should be set. As another issue, the convergence of the algorithm should be studied, however, the nesting of an alternating outer loop and a suboptimal alternating projections inner loop probably makes this a quite difficult task.

## REFERENCES

- [1] F. L. Hitchcock, "The expression of a tensor or a polyadic as a sum of products," *Studies in Applied Mathematics*, vol. 6, no. 1-4, pp. 164–189, 1927.
- [2] M. Mørup, "Applications of tensor (multiway array) factorizations and decompositions in data mining," *Data Min. Knowl. Discov.*, vol. 1, no. 1, pp. 24–40, 2011.
- [3] H. Becker, L. Albera, P. Comon, M. Haardt, G. Birot, F. Wendling, M. Gavaret, C.-G. Bénar, and I. Merlet, "EEG extended source localization: tensor-based vs. conventional methods," *NeuroImage*, vol. 96, pp. 143–157, 2014.
- [4] A. Smilde, R. Bro, and P. Geladi, *Multi-way analysis: applications in the chemical sciences*. John Wiley & Sons, 2005.
- [5] N. D. Sidiropoulos, R. Bro, and G. B. Giannakis, "Parallel factor analysis in sensor array processing," *IEEE Trans. Signal Process.*, vol. 48, no. 8, pp. 2377–2388, 2000.
- [6] P. Comon, X. Luciani, and A. L. F. De Almeida, "Tensor decompositions, alternating least squares and other tales," *J. Chemometr.*, vol. 23, no. 7-8, pp. 393–405, 2009.
- [7] V. De Silva and L.-H. Lim, "Tensor rank and the ill-posedness of the best low-rank approximation problem," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 1084–1127, 2008.
- [8] L.-H. Lim and P. Comon, "Blind multilinear identification," *IEEE Trans. Inf. Theory*, vol. 60, no. 2, pp. 1260–1280, 2014.
- [9] S. Sahnoun and P. Comon, "Joint source estimation and localization," *IEEE Trans. Signal Process.*, vol. 63, no. 10, pp. 2485–2495, 2015.
- [10] D. Barchiesi and M. D. Plumbley, "Learning incoherent dictionaries for sparse approximation using iterative projections and rotations," *IEEE Trans. Signal Process.*, vol. 61, no. 8, pp. 2055–2065, 2013.
- [11] P. Comon, "Tensors: a brief introduction," *IEEE Signal Process. Mag.*, vol. 31, no. 3, pp. 44–53, 2014.
- [12] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.
- [13] J. E. Cohen, "About notations in multiway array processing," *arXiv preprint arXiv:1511.01306*, 2015.
- [14] R. Bro, "Multi-way analysis in the food industry: models, algorithms, and applications," Ph.D. dissertation, Univ. of Amsterdam, 1998.
- [15] R. L. Dykstra, "An algorithm for restricted least squares regression," *J. Am. Stat. Assoc.*, vol. 78, no. 384, pp. 837–842, 1983.
- [16] N. J. Higham, "Computing the nearest correlation matrix - a problem from finance," *IMA J. Numer. Anal.*, vol. 22, no. 3, pp. 329–343, 2002.
- [17] P. H. Schönemann, "A generalized solution of the orthogonal Procrustes problem," *Psychometrika*, vol. 31, no. 1, pp. 1–10, 1966.
- [18] P. Breiding and N. Vannieuwenhoven, "The condition number of joint decompositions," *SIAM J. Matrix Anal. Appl.*, vol. 39, no. 1, pp. 287–309, 2018.
- [19] A. Stegeman, "Degeneracy in Candecomp/Parafac explained for  $p \times p \times 2$  arrays of rank  $p + 1$  or higher," *Psychometrika*, vol. 71, no. 3, pp. 483–501, Sep 2006.
- [20] E. Sanchez and B. R. Kowalski, "Tensorial resolution: a direct trilinear decomposition," *J. Chemometr.*, vol. 4, no. 1, pp. 29–45, 1990.
- [21] N. Vervliet, O. Debals, L. Sorber, M. Van Barel, and L. De Lathauwer, "Tensorlab 3.0, mar. 2016," URL <http://www.tensorlab.net>.