Distributed Computation of Tensor Decompositions in Collaborative Networks

Andre L. F. de Almeida

Department of Teleinformatics Engineering

Federal University of Ceara

Fortaleza, Brazil

Email: andre@gtel.ufc.br

Alain Y. Kibangou

GIPSA-Lab

University Joseph Fourier/CNRS

Saint Martin d'Hères, France

Email: alain.kibangou@ujf-grenoble.fr

Abstract—In this paper, we consider the issue of distributed computation of tensor decompositions. A central unit observing a global data tensor assigns different data sub-tensors to several computing nodes grouped into clusters. The goal is to distribute the computation of a tensor decomposition across the different computing nodes of the network, which is particularly useful when dealing with large-scale data tensors. However, this is only possible when the data sub-tensors assigned to each computing node in a cluster satisfies minimum conditions for uniqueness. By allowing collaboration between computing nodes in a cluster, we show that average consensus based estimation is useful to yield unique estimates of the factor matrices of each data sub-tensor. Moreover, an essentially unique reconstruction of the global factor matrices at the central unit is possible by allowing the sub-tensors assigned to different clusters to overlap in one mode. The proposed approach may be useful to a number of distributed tensor-based estimation problems in signal processing.

I. Introduction

In many disciplines, data inherently has more than two axes of variation and can be arranged as tensors (i.e. multiway arrays). Computing tensor decompositions of multiway datasets is particularly useful to extract hidden patterns and structure in data analytics problems. In most of applications in the literature, batch processing is considered for computing tensor decompositions. Several algorithms have been proposed in the literature, which can be classified into three main categories: alternating algorithms, derivative based algorithms, and non-iterative algorithms (see e.g. [1], [2]). Considering that the data tensor can be serially acquired or the underlying process can be time-varying, adaptive algorithms have been proposed in [3]. These tensor decomposition algorithms are herein refereed to as "centralized", in the sense that the estimation of the factor matrices of the decomposition are accomplished in a unique central processor. However, when dealing with large-scale multiway datasets, centralized solutions may be too costly or even impractical due to the huge number of operations and memory storage requirements.

In [4], the authors proposed a distributed approach for the computation of the canonical polyadic decomposition (CPD) of a third-order tensor across a network of computing nodes. Distributed alternating least squares (D-ALS) and distributed Levenberg-Marquardt (D-LM) algorithms based on average consensus are proposed. A different approach for distributed computation of tensor decompositions is developed in [5]. Therein, the idea is to divide the large-scale data tensor into a grid of sub-tensors that are factorized independently at each computing node. However, a meaningful reconstruction of the global factor matrices by concatenation requires imposing that the partially estimated factor matrices obtained at each node are subject to the same permutation indeterminacy. Moreover, such a distributed decomposition approach requires that each sub-tensor have an essentially unique decomposition.

In this paper, we present a new framework that keeps uniqueness properties of the centralized scheme: The computation of a tensor decomposition is composed of two tiers. First, the original tensor is divided into different data sub-tensors along the third mode (assumed to be the mode with higher dimensionality) and assigned to different computing clusters. Within each cluster, the associated sub-tensor is further divided into a grid of smaller sub-tensors which are assigned to different computing nodes. Instead of working independently, the computing nodes are allowed to collaborate to estimate their factor matrices using a distributed average consensus algorithm [6]. This is particularly important when the sub-tensors assigned to the computing nodes does not admit an essentially unique CPD. In such a situation, distributed average consensus yields essentially unique estimates of the factor matrices at each computing node under mild conditions. Moreover, by allowing the sub-tensors assigned to different clusters to overlap along the third mode, the reconstruction of the global factor matrices at the central unit is possible.

Notations: Vectors are written as boldface lower-case letters $(\mathbf{a}, \mathbf{b}, \cdots)$, matrices as boldface upper-case letters $(\mathbf{A}, \mathbf{B}, \cdots)$, and tensor as calligraphic letters $(\mathcal{X}, \mathcal{Y}, \cdots)$. $\mathbf{A}^T$ stands for the transpose of $\mathbf{A}$ whereas $\mathbf{A}^H$ stands for its complex conjugate. The operator $\circ$ denotes the outer product while $\odot$ and $*$ stand for the Khatri-Rao (columnwise Kronecker) and Hadamard (elementwise) products, respectively.

II. Problem Statement

A central unit observes a global data tensor $\mathcal{X} \in \mathbb{C}^{I \times J \times K}$ following a CPD given by [7], [8]:

$$\mathcal{X} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r, \tag{1}$$

where $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_R] \in \mathbb{C}^{I \times R}$, $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_R] \in \mathbb{C}^{J \times R}$ and $\mathbf{C} = [\mathbf{c}_1, \ldots, \mathbf{c}_R] \in \mathbb{C}^{K \times R}$ are the factor matrices associated with the CPD of $\mathcal{X}$. We work under the assumption that essential uniqueness of this CPD holds, i.e. according to

[9], $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ can be uniquely estimated (up to column permutation and scaling) if $k_{\mathbf{A}} + k_{\mathbf{B}} + k_{\mathbf{C}} \geq 2R + 2$, where $k_{(\cdot)}$ denotes the Kruskal-rank of its matrix argument. The global tensor $\mathcal{X}$ is subdivided by the central unit in sub-tensors $\mathcal{X}^{(q)} \in \mathbb{C}^{I \times J \times K^{(q)}}$, $q = 1, \ldots, Q$, which are assigned to Q computing clusters, where $K^{(q)}$ denotes the third-mode dimensionality of the q-th sub-tensor[1]. The CPD of sub-tensor $\mathcal{X}^{(q)}$ is then given by

$$\mathcal{X}^{(q)} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r^{(q)}, \tag{2}$$

where $\mathbf{C}^{(q)} = [\mathbf{c}_1^{(q)}, \ldots, \mathbf{c}_R^{(q)}] \in \mathbb{C}^{K^{(q)} \times R}$.

In practice, the central unit may be a server that gathers the global (possibly large-scale data tensor, while the computing nodes can be co-located in the same chip, or may be represented by remote processors that are geographically distributed. Assume that the q-th cluster is composed by L computing nodes driven by a "master node" that communicates directly with the central unit[2]. Upon reception of the data sub-tensor $\mathcal{X}^{(q)}$, the q-th master node generates a grid of sub-tensors by partitioning $\mathcal{X}^{(q)}$ into multiple sub-tensors $\mathcal{X}^{(\ell_1, \ell_2, q)} \in \mathbb{C}^{I_{\ell_1} \times J_{\ell_2} \times K^{(q)}}$, where $\sum_{\ell_1=1}^{L_1} I_{\ell_1} = I$, $\sum_{\ell_2=1}^{L_2} J_{\ell_2} = J$, L_1 and L_2 denote the number of sub-tensors along the first and second modes, respectively, and $\ell_1 = 1, \ldots, L_1$, $\ell_2 = 1, \ldots, L_2$ are the indices of the sub-tensors in the first and second modes, respectively. Each sub-tensor $\mathcal{X}^{(\ell_1, \ell_2, q)}$ is allocated to a different computing node, so that $L = L_1 L_2$ corresponds to the number of computing nodes per cluster. The CPD of sub-tensor $\mathcal{X}^{(\ell_1, \ell_2, q)}$ is then given by:

$$\mathcal{X}^{(\ell_1, \ell_2, q)} = \sum_{r=1}^{R} \mathbf{a}_r^{(\ell_1)} \circ \mathbf{b}_r^{(\ell_2)} \circ \mathbf{c}_r^{(q)}, \tag{3}$$

where $\mathbf{A}^{(\ell_1)} = [\mathbf{a}_1^{(\ell_1)}, \ldots, \mathbf{a}_R^{(\ell_1)}] \in \mathbb{C}^{I_{\ell_1} \times R}$, $\mathbf{B}^{(\ell_2)} = [\mathbf{b}_1^{(\ell_2)}, \ldots, \mathbf{b}_R^{(\ell_2)}] \in \mathbb{C}^{J_{\ell_2} \times R}$ in addition to $\mathbf{C}^{(q)}$ are the factor matrices associated with this CPD. Although the CPD of each sub-tensor $\mathcal{X}^{(q)}$ is assumed to be essentially unique, i.e. $k_{\mathbf{A}} + k_{\mathbf{B}} + k_{\mathbf{C}^{(q)}} \geq 2R + 2$, $q = 1, \ldots, Q$, we are interested in a situation where the identifiability conditions are not locally fulfilled at each computing node.

Let $\mathbf{X}_1^{(\ell_1, \ell_2, q)} \in \mathbb{C}^{J_{\ell_2} K^{(q)} \times I_{\ell_1}}$, $\mathbf{X}_2^{(\ell_1, \ell_2, q)} \in \mathbb{C}^{I_{\ell_1} K^{(q)} \times J_{\ell_2}}$, and $\mathbf{X}_3^{(\ell_1, \ell_2, q)} \in \mathbb{C}^{I_{\ell_1} J_{\ell_2} \times K^{(q)}}$ be matrix unfoldings of the sub-tensor $\mathcal{X}^{(\ell_1, \ell_2, q)}$. The matrix unfoldings of the sub-tensor $\mathcal{X}^{(q)} \in \mathbb{C}^{I \times J \times K^{(q)}}$ assigned to q-th cluster can be defined by concatenating the $L = L_1 L_2$ matrix unfoldings of the sub-tensors $\mathcal{X}^{(\ell_1, \ell_2, q)} \in \mathbb{C}^{I_{\ell_1} \times J_{\ell_2} \times K^{(q)}}$, as:

$$\mathbf{X}_1^{(q)} \doteq [\mathbf{X}_1^{(1,1,q)T}, \ldots, \mathbf{X}_1^{(1,L_2,q)T}, \ldots, \mathbf{X}_1^{(L_1,L_2,q)T}]^T \in \mathbb{C}^{JK^{(q)} \times I}$$

$$\mathbf{X}_2^{(q)} \doteq [\mathbf{X}_2^{(1,1,q)T}, \ldots, \mathbf{X}_2^{(1,L_2,q)T}, \ldots, \mathbf{X}_2^{(L_1,L_2,q)T}]^T \in \mathbb{C}^{K^{(q)}I \times J}$$

$$\mathbf{X}_3^{(q)} \doteq [\mathbf{X}_3^{(1,q)T}, \ldots, \mathbf{X}_3^{(1,L_2,q)T}, \ldots, \mathbf{X}_3^{(L_1,L_2,q)T}]^T \in \mathbb{C}^{IJ \times K^{(q)}}$$

---

[1]Although we have assumed that the sub-tensors are generated by partitioning along the third mode, partitions along the first or second modes would be equally possible. The mode having higher dimensionality can be choosen for this purpose. Herein, we assume that $K >> I$ and $K >> J$.

[2]We have assumed the same number L of computing nodes for all clusters to aleviate notation. The following developments are equally valid when the clusters have different number of nodes.

If the q-th master node were able to concentrate the whole computational load of the $L = L_1 L_2$ computing nodes available in the q-th cluster, the factor matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}^{(q)}$ could be estimated directly from the sub-tensor $\mathcal{X}^{(q)}$. Various algorithms could be used for this purpose, the alternating least squares (ALS) being the most popular one. By exploiting the three matrix unfoldings of $\mathcal{X}^{(q)}$, the ALS algorithm acts by alternately minimizing the following cost functions

$$\mathbb{J}_1 = \left\| \mathbf{X}_1^{(q)} - \left(\mathbf{B} \odot \mathbf{C}^{(q)} \right) \mathbf{A}^T \right\|_F^2, \tag{4}$$

$$\mathbb{J}_2 = \left\| \mathbf{X}_2^{(q)} - \left(\mathbf{C}^{(q)} \odot \mathbf{A} \right) \mathbf{B}^T \right\|_F^2, \tag{5}$$

$$\mathbb{J}_3 = \left\| \mathbf{X}_3^{(q)} - (\mathbf{A} \odot \mathbf{B}) \mathbf{C}^{(q)T} \right\|_F^2. \tag{6}$$

In our context, however, instead of directly estimating the triplet $(\mathbf{A}, \mathbf{B}, \mathbf{C}^{(q)})$, each node should carry out the estimation of its own triplet $(\mathbf{A}^{(\ell_1)}, \mathbf{B}^{(\ell_2)}, \mathbf{C}^{(q)})$. However, since essential uniqueness is not necessary fulfilled at each computing node, running independent ALS do not guarantee to preserve the overall CPD uniqueness property. By allowing collaboration between computing nodes in a cluster, we propose the use of average consensus based estimation to ensure essentially uniqueness at each computing node. The consensus-based approach also ensures the L computing nodes reach an agreement on the estimation of the third mode factor matrix $\mathbf{C}^{(q)}$, which is a common parameter of the CPD of all sub-tensors $\mathcal{X}^{(\ell_1, \ell_2, q)}$, $\ell_1 = 1, \ldots, L_1$, $\ell_2 = 1, \ldots, L_2$, as shown in (3).

III. DISTRIBUTED COMPUTATION

The communication links between computing nodes of each cluster are modeled by means of an undirected graph $\mathrm{G}(\mathrm{N}, \mathrm{E})$. The node set and the edge set are denoted by $\mathrm{N} = \{1, \cdots, L\}$ and E, respectively. Each edge is an unordered pair of distinct nodes. We assume that G is a connected graph, meaning that it exists a path between any two pair of computing nodes within a cluster. In other words, information can be exchanged by any pair of nodes of the same cluster through direct or multi-hop links.

A. Consensus-based estimation of $\mathbf{C}^{(q)}$

Define the ordered pair $\ell \doteq (\ell_1, \ell_2)$, where ℓ is a super-index satisfying $\ell = (\ell_1 - 1)L_2 + \ell_2$, $\ell_1 = 1, \ldots, L_1$, $\ell_2 = 1, \ldots, L_2$, $\ell = 1, \ldots, L$. By concatenating $L = L_1 L_2$ matrix unfoldings of each type according to this convention, we can define the matrix

$$\mathbf{Z}^{(q)} = \mathbf{A} \odot \mathbf{B} = \begin{bmatrix} \mathbf{Z}^{(1,q)T} & \cdots & \mathbf{Z}^{(\ell,q)T} & \cdots & \mathbf{Z}^{(L,q)T} \end{bmatrix}^T.$$

From the definition of $\mathbf{X}_3^{(q)}$, we have:

$$\mathbb{J}_3 = \left\| \begin{bmatrix} \mathbf{X}_3^{(1,q)} \\ \vdots \\ \mathbf{X}_3^{(\ell,q)} \\ \vdots \\ \mathbf{X}_3^{(L,q)} \end{bmatrix} - \begin{bmatrix} \mathbf{Z}^{(1,q)} \\ \vdots \\ \mathbf{Z}^{(\ell,q)} \\ \vdots \\ \mathbf{Z}^{(L,q)} \end{bmatrix} \mathbf{C}^{(q)T} \right\|_F^2. \tag{7}$$

The solution for $\mathbf{C}^{(q)}$ that minimizes $\mathbb{J}_3$ can be written as a sum of L contributions

$$\widehat{\mathbf{C}}^{(q)T} = \left(\frac{1}{L} \sum_{\ell=1}^{L} \mathbf{\Gamma}^{(\ell,q)} \right)^{-1} \left(\frac{1}{L} \sum_{\ell=1}^{L} \mathbf{\Psi}^{(\ell,q)} \right) = (\overline{\mathbf{\Gamma}}^{(q)})^{-1} \overline{\mathbf{\Psi}}^{(q)}$$
(8)

where

$$\mathbf{\Gamma}^{(\ell,q)} = \mathbf{Z}^{(\ell,q)H} \mathbf{Z}^{(\ell,q)} = (\mathbf{A}^{(\ell_1)H} \mathbf{A}^{(\ell_1)}) * (\mathbf{B}^{(\ell_2)H} \mathbf{B}^{(\ell_2)}), \quad (9)$$
$$\mathbf{\Psi}^{(\ell,q)} = \mathbf{Z}^{(\ell,q)H} \mathbf{X}_3^{(\ell,q)} = (\mathbf{A}^{(\ell_1)} \odot \mathbf{B}^{(\ell_2)})^H \mathbf{X}_3^{(\ell,q)}. \quad (10)$$

Note that the computation of (8) results on averaging the $L = L_1 L_2$ matrices $\mathbf{\Gamma}^{(\ell,q)}$ and $\mathbf{\Psi}^{(\ell,q)}$ associated with the different nodes. Such an averaging can be computed locally at every node via local communication and computation within the cluster by means of the consensus algorithm, which is a well-known solution to distributed computing problems [6], [10]. By examining (8), one can note that averaging over a sufficient number L of independent terms so that $\overline{\mathbf{\Gamma}}^{(q)}$ is nonsingular ensures the identifiability of $\mathbf{C}^{(q)}$ in the least squares sense. This could not be possible if nodes do not collaborate. Without nodes' collaboration, the solution (8) reduces to $\widehat{\mathbf{C}}^{(q)T} = (\mathbf{\Gamma}^{(\ell,q)})^{-1} \mathbf{\Psi}^{(\ell,q)}$. A limiting situation is the one where $k_{\mathbf{A}^{(\ell_1)}} = 1$ and/or $k_{\mathbf{B}^{(\ell_2)}} = 1$ for a particular node. In this case, $\overline{\mathbf{\Gamma}}^{(q)}$ is singular, and essential uniqueness is not locally fulfilled. Such a lack of uniqueness will affect a proper reconstruction of the global factor matrices at the central unit, even if uniqueness hold for the remaining nodes.

Let $\mathbf{\Gamma}_{(0)}^{(\ell,q)}$ and $\mathbf{\Psi}_{(0)}^{(\ell,q)}$ be matrices initially computed at node $\ell = (\ell_1, \ell_2)$ at time $t = 0$. By allowing these matrices to be exchanged across the network of nodes of the q-th cluster, at each iteration t, each computing node updates the values of $\mathbf{\Gamma}_{(t)}^{(\ell,q)}$ and $\mathbf{\Psi}_{(t)}^{(\ell,q)}$ by adding a weighted sum of local discrepancies, i.e. differences between its own value and those obtained from its neighbors $\ell' \in \mathrm{N}_\ell = \left\{ \ell' | (\ell, \ell') \in \mathrm{E} \right\}$. The updating equation for $\mathbf{\Psi}_{(t)}^{(\ell,q)}$ is given by [4]:

$$\mathbf{\Psi}_{(t+1)}^{(\ell,q)} = \mathbf{\Psi}_{(t)}^{(\ell,q)} + \sum_{\ell' \in \mathrm{N}_\ell} w_{\ell,\ell'} \left(\mathbf{\Psi}_{(t)}^{(\ell',q)} - \mathbf{\Psi}_{(t)}^{(\ell,q)} \right). \quad (11)$$

where $w_{\ell,\ell'}$ is a weight associated with the edge $\{\ell, \ell'\}$, which is a typical element of the weight matrix $\mathbf{W} \in \mathbb{C}^{L \times L}$. This matrix is doubly stochastic, i.e. $\mathbf{1}^T \mathbf{W} = \mathbf{1}^T$ and $\mathbf{W1} = \mathbf{1}$, where $\mathbf{1} \in \mathbb{C}^{L \times 1}$ is a vector of ones. Several weights fulfilling the previous conditions have been proposed in the literature. For further details, see [11], [12] and references therein. The update equation for $\mathbf{\Gamma}_{(t)}^{(\ell,q)}$ is similar.

B. Collaborative estimation of $\mathbf{A}^{(\ell_1)}$ and $\mathbf{B}^{(\ell_2)}$

The factor matrices $\mathbf{A}^{(\ell_1)}$ and $\mathbf{B}^{(\ell_2)}$ associated with the CPD of $\mathcal{X}^{(\ell,q)}$ are common to computing nodes sharing the same indices l_1 or l_2. A collaboration is needed for estimating these matrices. From the definition of the matrix unfoldings $\mathbf{X}_1^{(q)}$ and $\mathbf{X}_2^{(q)}$, we have:

$$\mathbb{J}_1 = \sum_{\ell_1=1}^{L_1} \sum_{\ell_2=1}^{L_2} \left\| \mathbf{X}_1^{(\ell_1,\ell_2,q)} - (\mathbf{B}^{(\ell_2)} \odot \mathbf{C}^{(q)}) \mathbf{A}^{(\ell_1)T} \right\|_F^2,$$

$$\mathbb{J}_2 = \sum_{\ell_1=1}^{L_1} \sum_{\ell_2=1}^{L_2} \left\| \mathbf{X}_2^{(\ell_1,\ell_2,q)} - (\mathbf{C}^{(q)} \odot \mathbf{A}^{(\ell_1)}) \mathbf{B}^{(\ell_2)T} \right\|_F^2,$$

By minimizing $\mathbb{J}_1$ and $\mathbb{J}_2$ with respect to $\mathbf{A}^{(\ell_1)}$ and $\mathbf{B}^{(\ell_2)}$, respectively, each cluster yields

$$\widehat{\mathbf{A}}^{(\ell_1)T} = \left(\frac{1}{L_2} \sum_{\ell_2=1}^{L_2} \mathbf{\Gamma}_A^{(\ell_2,q)} \right)^{-1} \left(\frac{1}{L_2} \sum_{\ell_2=1}^{L_2} \mathbf{\Psi}_A^{(\ell_2,q)} \right) \quad (12)$$

where
$$\mathbf{\Gamma}_A^{(\ell_2,q)} = (\mathbf{B}^{(\ell_2)H} \mathbf{B}^{(\ell_2)}) * (\mathbf{C}^{(q)H} \mathbf{C}^{(q)}), \quad (13)$$
$$\mathbf{\Psi}_A^{(\ell_2,q)} = (\mathbf{B}^{(\ell_2)} \odot \mathbf{C}^{(q)})^H \mathbf{X}_1^{(\ell_1,\ell_2,q)}, \quad (14)$$

and

$$\widehat{\mathbf{B}}^{(\ell_2)T} = \left(\frac{1}{L_1} \sum_{\ell_1=1}^{L_1} \mathbf{\Gamma}_B^{(\ell_1,q)} \right)^{-1} \left(\frac{1}{L_1} \sum_{\ell_1=1}^{L_1} \mathbf{\Psi}_B^{(\ell_1,q)} \right) \quad (15)$$

where
$$\mathbf{\Gamma}_B^{(\ell_1,q)} = (\mathbf{C}^{(q)H} \mathbf{C}^{(q)}) * (\mathbf{A}^{(\ell_1)H} \mathbf{A}^{(\ell_1)}), \quad (16)$$
$$\mathbf{\Psi}_B^{(\ell_1,q)} = (\mathbf{C}^{(q)} \odot \mathbf{A}^{(\ell_1)})^H \mathbf{X}_2^{(\ell_1,\ell_2,q)}. \quad (17)$$

As for the $\mathbf{C}^{(q)}$ matrix, the two other matrices can also be estimated through a consensus scheme. Therefore, the proposed algorithm consists of interlacing average consensus steps with three ALS estimation steps. The detailed description of the algorithm is not provided here due to space limitation.

C. Reconstruction of the global factor matrices

Let $\widehat{\mathbf{A}}^{(\ell_1,q)}$, $\widehat{\mathbf{B}}^{(\ell_2,q)}$ and $\widehat{\mathbf{C}}^{(q)}$ be the factor matrices estimated at the node (ℓ_1, ℓ_2) of the q-th cluster. Define $\widehat{\mathbf{A}}^{(q)} = [\widehat{\mathbf{A}}^{(1,q)}, \dots, \widehat{\mathbf{A}}^{(L_1,q)}]$ and $\widehat{\mathbf{B}}^{(q)} = [\widehat{\mathbf{B}}^{(1,q)}, \dots, \widehat{\mathbf{B}}^{(L_2,q)}]$. Due to the parallel processing performed independently at the Q computing clusters, the estimated triplets $(\widehat{\mathbf{A}}^{(q)}, \widehat{\mathbf{B}}^{(q)}, \widehat{\mathbf{C}}^{(q)})$ are subject to unknown and different permutation and scaling indeterminacies, i.e. $\widehat{\mathbf{A}}^{(q)} = \mathbf{A}^{(q)} \mathbf{\Delta}_1^{(q)} \mathbf{\Pi}^{(q)}$, $\widehat{\mathbf{B}}^{(q)} = \mathbf{B}^{(q)} \mathbf{\Delta}_2^{(q)} \mathbf{\Pi}^{(q)}$, $\widehat{\mathbf{C}}^{(q)} = \mathbf{C}^{(q)} \mathbf{\Delta}_3^{(q)} \mathbf{\Pi}^{(q)}$. An essentially unique reconstruction of the global factor matrices at the central unit is possible by allowing the sub-tensors assigned to different clusters to overlap in the third mode. We assume that $\mathcal{X}^{(q)}$ and $\mathcal{X}^{(q+1)}$, $q = 1, \dots, Q-1$, share a set of M third mode slices, such that the following relation holds

$$\mathbf{X}_{\cdot\cdot(K^{(q)}-M+m)}^{(q)} = \mathbf{X}_{\cdot\cdot m}^{(q+1)}, \quad m = 1, \dots, M, \quad (18)$$

where $M < \min(K^{(1)}, \dots, K^{(Q)})$ is assumed. Let $\mathbf{C}_{\mathrm{last}}^{(q)} \in \mathbb{C}^{M \times R}$ be a sub-matrix formed from the last M rows of $\mathbf{C}^{(q)}$ and $\mathbf{C}_{\mathrm{first}}^{(q+1)} \in \mathbb{C}^{M \times R}$ be a sub-matrix formed from the first M rows of $\mathbf{C}^{(q+1)}$, $q = 1, \dots, Q-1$. Condition (18) ensures that if the CPD's of $\mathcal{X}^{(q)}$ and $\mathcal{X}^{(q+1)}$ are essentially unique, then there exists a scaling $\overline{\mathbf{\Delta}}^{(q)}$ and a permutation matrix $\overline{\mathbf{\Pi}}^{(q)}$ such that

$$\widehat{\mathbf{C}}_{\mathrm{first}}^{(q+1)} = \widehat{\mathbf{C}}_{\mathrm{last}}^{(q)} \overline{\mathbf{\Delta}}^{(q)} \overline{\mathbf{\Pi}}^{(q)}, \quad q = 1, \dots, Q-1.$$

The permutation matrix $\overline{\mathbf{\Pi}}^{(q)}$ can be determined by a greedy pairwise column matching procedure applied on the matrix pair $(\mathbf{C}_{\mathrm{last}}^{(q)}, \mathbf{C}_{\mathrm{first}}^{(q+1)})$, $q = 1, \dots, Q-1$, after proper normalization of their columns. By increasing the overlapping factor M, this permutation matrix can be determined more efficiently.

After "aligning" the columns of the $\widehat{\mathbf{C}}^{(1)}, \dots, \widehat{\mathbf{C}}^{(Q)}$ (resp. $\widehat{\mathbf{A}}^{(1)}, \dots, \widehat{\mathbf{A}}^{(Q)}$ and $\widehat{\mathbf{B}}^{(1)}, \dots, \widehat{\mathbf{B}}^{(Q)}$), the central unit can build
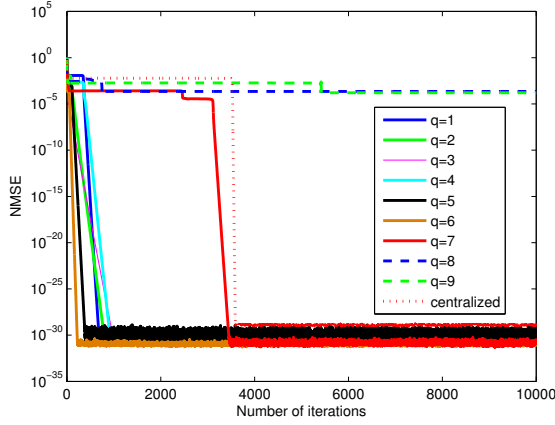
Fig. 1. NMSE for each cluster compared to the standard ALS algorithm (dotted lines)

Fig. 2. Sources reconstruction: Actual (left), Standard PARAFAC (middle), Proposed method (right)

a global estimate of the third factor matrix by concatenation, i.e. $\widehat{\mathbf{C}} = [\widehat{\mathbf{C}}^{(1)T}, \dots, \widehat{\mathbf{C}}^{(Q)T}]^T$ satisfying the essential uniqueness property. Since the first and second mode factor matrices should be common to all clusters (c.f. (3)), the central unit can obtain final estimates of these matrices by taking weighted averages $\widehat{\mathbf{A}} = \sum_{q=1}^{Q} w_q \widehat{\mathbf{A}}^{(q)}$, and $\widehat{\mathbf{B}} = \sum_{q=1}^{Q} w_q \widehat{\mathbf{B}}^{(q)}$, with properly chosen weights. Th normalized mean square error can be used for computing such weights.

IV. SIMULATION RESULTS

In this section we illustrate the efficiency of the proposed method by simulating a $20 \times 20 \times 100$ tensor admitting a CPD with rank 6. The 6 sources describing the third matrix $\mathbf{C}$ are given in Fig. 2 (left). The central node partitions the tensor in 9 sub-tensors of dimensions $20 \times 20 \times 20$, meaning that we have 9 clusters. The number of overlapped slices is $M = 10$. In each cluster, the two remaining modes are subdivided so that each computing node processes a $5 \times 5 \times 20$ tensor. The computing nodes in a cluster collaborate through a communication policy represented by a Hamming graph $H(2, L_1)$ [13]. Using the finite-time average consensus protocol in [12], two consensus iterations are required for obtaining the exact averages while exact summation for estimating $\mathbf{A}$ and $\mathbf{B}$ is obtained in a single step. Fig. 1 depicts the averaged Normalized Mean square error (NMSE) for each cluster. We can note that two clusters ($q = 8$ and $q = 9$) require much more iterations to converge. This is due to collinearity occurring in the corresponding sub-factor matrices. As a consequence the last rows of $\mathbf{C}$ are not well estimated. The central node can fix this problem by using the factor matrices $\mathbf{A}$ and $\mathbf{B}$ reconstructed with the 7 other clusters to estimate the last rows of $\mathbf{C}$. In Fig. 2, we can note that the sources in $\mathbf{C}$ are correctly reconstructed.

V. CONCLUSION AND PERSPECTIVES

We have proposed a distributed approach to compute the CPD of a third-order tensor across a network of collaborative nodes. Our solution allows to parallelize the computation of the decomposition across independent computing clusters while benefiting from some level of collaboration and coordinat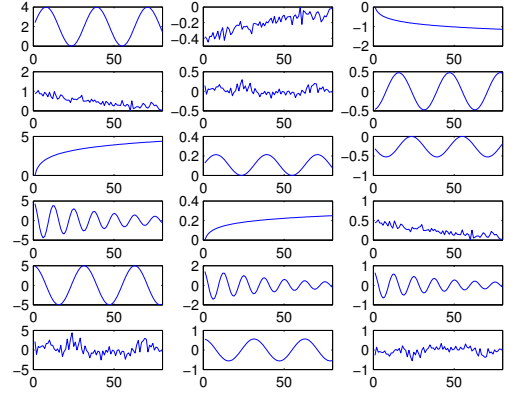ion between neighboring computing nodes to ensure essential uniqueness of the factor matrices. Although we have restricted ourselves to third-order tensors with a particular partitioning, the proposed framework can be generalized to N-th order tensors and to different sub-tensor partitioning schemes. The case of sub-tensors that share multiple factor matrices and overlap in more than one mode is currently under investigation. The proposed approach may be useful to a number of distributed estimation problems in signal processing.

REFERENCES

[1] G. Tomasi and R. Bro, "A comparison of algorithms for fitting the PARAFAC model," *Comp. Stat. Data Anal.*, vol. 50, no. 7, pp. 1700–1734, 2006.

[2] P. Comon, X. Luciani, and A. L. F. de Almeida, "Tensor decompositions, alternating least squares and other tales," *J. Chemom.*, vol. 23, pp. 393–405, 2009.

[3] D. Nion and N. Sidiropoulos, "Adaptive algorithms to track the PARAFAC decomposition of a third-order tensor," *IEEE Trans. on Signal Proc.*, vol. 57, no. 6, pp. 2299–2310, June 2009.

[4] A. Y. Kibangou and A. L. F. de Almeida, "Consensus-based in-network computation of the PARAFAC decomposition," *submitted*, 2013.

[5] A. H. Phan and A. Cichocki, "PARAFAC algorithms for large-scale problems," *Neurocomputing*, vol. 74, no. 11, pp. 1970 – 1984, 2011.

[6] L. Xiao, S. Boyd, and S.-J. Kim, "Distributed average consensus with least-mean-square deviation," *J. Parallel Distrib. Comput.*, vol. 67, pp. 33–46, 2007.

[7] R. Harshman, "Foundation of the PARAFAC procedure: models and conditions for an "explanatory" multimodal factor analysis," *UCLA working papers in phonetics*, vol. 16, pp. 1–84, 1970.

[8] J. Caroll and J. Chang, "Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition," *Psychometrika*, vol. 35, pp. 283–319, 1970.

[9] J. Kruskal, "Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics," *Linear Algebra Applicat.*, vol. 18, pp. 95–138, 1977.

[10] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems Control Lett.*, vol. 53, pp. 65–78, 2004.

[11] A. Kibangou, "Finite-time average consensus based protocol for distributed estimation over awgn channels," in *Proc. of the 50th IEEE Conference on Decision and Control (CDC)*, Orlando, Fl, USA, 2011.

[12] ——, "Graph Laplacian based matrix design for finite-time distributed average consensus," in *Proc. of the American Conference on Control(ACC)*, Montréal, Canada, 2012.

[13] A. Brouwer, A. Cohen, and A. Neumaier, *Distance-regular graphs*. Springer, 1989.