JOINT DATA AND CONNECTION TOPOLOGY RECOVERY IN COLLABORATIVE WIRELESS SENSOR NETWORKS

André L. F. de Almeida[⋆], *Alain Y. Kibangou*[†], *Sebastian Miron*[‡], *Daniel C. Araújo*[⋆]
[⋆]Department of Teleinformatics Engineering, Federal University of Ceará, Fortaleza, Brazil
[†]GIPSA-Lab, University Joseph Fourier, CNRS, Saint Martin d'Hères, France
[‡]CRAN, Université de Lorraine, CNRS, Vandœuvre-lès-Nancy, France
E-mails: {andre,araujo}@gtel.ufc.br, alain.kibangou@ujf-grenoble.fr, sebastian.miron@univ-lorraine.fr

ABSTRACT

This work considers a collaborative wireless sensor network where nodes locally exchange coded informative data before transmitting the combined data towards a remote fusion center equipped with an antenna array. For this communication scenario, a new blind estimation algorithm is developed for jointly recovering network transmitted data and connection topology at the fusion center. The proposed algorithm is based on a two-stage approach. The first stage is concerned with the estimation of the channel gains linking the nodes to the fusion center antennas. The second stage performs a joint estimation of network data and connection topology matrices by exploiting a constrained (PARALIND) tensor model for the collected data at the fusion center. Illustrative simulation results evaluate the performance of the proposed algorithm for some system configurations and network topologies.

***Index Terms*—** Blind estimation, wireless sensor networks, tensor modeling, PARALIND.

1. INTRODUCTION

A considerable attention and research has been devoted in recent years to the deployment of sensors for monitoring, collaborative information processing and control. In particular, wireless sensor networks that can operate autonomously, without a fusion center collecting and processing all measurements, exhibit desirable properties such as robustness against node failure [1]. The coordinated action of the different nodes distributed in the network requires local exchange of information between them according to a prescribed nodes connection topology. These local exchanges can possibly resort to distributed space-time coding protocols, [2, 3], where a collection of network nodes (users or sensors) work in a coordinated way to encode and decode the transmitted information by exploiting the spatial diversity.

A few recent works have developed tensor-based receivers in cooperative networks, [4, 5]. In [4], a supervised tensor-based receiver was proposed for two-way relaying cooperative systems with multiple antennas at the relay nodes. In [5], a tensor-based receiver was derived for distributed estimation in wireless sensor networks. In that work, the distributed estimation algorithm relies on an alternating least squares procedure combined with consensus averaging iterations. The work [6] proposed tensor-based collaborative space-time codes with distributed blind decoding based on a finite-time consensus algorithm. Both [5] and [6] capitalize on PARAFAC[1] analysis [7].

This work is partially supported by FUNCAP, INRIA and CNPq.

[1]PARAllel FACtors

In this paper, we propose a tensor-based approach for the joint recovery of network transmitted data and connection topology in a collaborative network. The network is composed by a set of nodes that locally exchange coded data according to a fixed connection topology, which is not known at the fusion center. After in-network coding and local collaboration, the sensor nodes communicate with a fusion center by simultaneously transmitting their coded signals during a time frame. Assuming that the fusion center is equipped with an antenna array, the proposed algorithm estimates network data and and nodes' connection topology in a two-stage approach. The first stage estimates the channel gains linking the nodes to the fusion center antennas. We consider two methods to accomplish this task: a conventional least-squares estimation using pilot data, and a PARAFAC-based blind estimation resorting to in-network coding instead. The second stage jointly estimates network data and connection topology matrices by exploiting a constrained (PARALIND)[2] tensor model [8] for the collected data at the fusion center. Our results show satisfactory performance of the proposed estimator and corroborate the benefits of in-network coding and nodes' collaboration for data recovery.

In the last decade, several blind receivers based on tensor models have been developed by exploiting multiple forms of diversity of the observed data (see e.g. [9]-[13] and references therein). All these works considered non-collaborative sources, which is generally the case in traditional communication systems. We propose an innovative tensor-based estimation approach, for a scenario where collaboration takes place among transmitting nodes in a network. Moreover, we consider a challenging problem where the nodes' connection topology is unknown at the fusion center.

Notation: The following notation is used throughout the paper: Scalars are denoted by lower-case letters $(a, b, \ldots)$, vectors are written as lower-case boldface letters $(\mathbf{a}, \mathbf{b}, \ldots)$, matrices as upper-case boldface letters $(\mathbf{A}, \mathbf{B}, \ldots)$, and tensors as calligraphic letters $(\mathcal{A}, \mathcal{B}, \ldots)$. To retrieve the element (i, j) from a matrix $\mathbf{A}$, we use the notation $[\mathbf{A}]_{i,j}$. $\mathbf{A}^T$ and $\mathbf{A}^\dagger$ stand for transpose and pseudo-inverse of $\mathbf{A}$, respectively. The operator $\circ$ denotes the outer product between two vectors. The operator $vec(\cdot)$ forms a vector by stacking the columns of its matrix argument. $diag(\mathbf{a})$ forms a diagonal matrix out of its vector argument $\mathbf{a}$. The Khatri-Rao (columnwise Kronecker) product between two matrices $\mathbf{A} \in \mathbb{C}^{I\times R}$ and $\mathbf{B} \in \mathbb{C}^{J\times R}$ is symbolized by $\mathbf{A} \diamond \mathbf{B}$. We make use of the following property of the Khatri-Rao product:

$$vec(\mathbf{A}diag(\mathbf{x})\mathbf{B}) = (\mathbf{B}^T \diamond \mathbf{A})\mathbf{x} \tag{1}$$

The Khatri-Rao product between $\mathbf{A}$ and $\mathbf{B}$ can be written as:

[2]PARAllel profiles with LINear Dependencies

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO CEARA. Downloaded on November 18,2022 at 13:35:47 UTC from IEEE Xplore. Restrictions apply.

$$\mathbf{A} \diamond \mathbf{B} = \begin{bmatrix} \mathbf{B}D_1(\mathbf{A}) \\ \vdots \\ \mathbf{B}D_I(\mathbf{A}) \end{bmatrix}, \quad (2)$$

where $D_i(\mathbf{A})$ forms a diagonal matrix out of the i-th row of $\mathbf{A}$. The following property is also useful:

$$vec(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A})vec(\mathbf{B}), \quad (3)$$

where $\otimes$ is the Kronecker product operator, and $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ are matrices of compatible dimensions.

2. SYSTEM MODEL

Consider a network of M smart devices with sensing, computation, and communication capabilities. These nodes communicate with a fusion center by periodically transmitting informative sequences during a time frame of N samples. Sensors are assumed to be single-antenna devices while the fusion center is equipped with an array of K antennas (see Fig. 1). In order to robustify both transmission and data recovery at the fusion center end, the sensors perform an in-network coding through a collaborative step where redundancy is added to transmitted data. In addition to data recovery, monitoring the topology of the sensor network is also a task to be achieved by the fusion center. Such a task is to be carried out jointly with the informative data recovery one.

2.1. In-network coding

First, each node of the sensor network applies a temporal coding operation by spreading its scalar data sample over P discrete-time intervals. Then it sends the coded sequence to its single-hop neighbors. The matrix form of this in-network coding operation is given by

$$\mathbf{X}_n = diag(\mathbf{s}_n)\mathbf{C}^T \in \mathbb{C}^{M\times P}, \quad (4)$$

where $\mathbf{X}_n \in \mathbb{C}^{M\times P}$ gathers the space-time coded data of the collaborative network, $\mathbf{C} \in \mathbb{C}^{P\times M}$ is the coding matrix, whereas $\mathbf{s}_n \in \mathbb{C}^M$ stands for the vector containing the nth informative signals. The coding matrix $\mathbf{C}$ is chosen to be a Vandermonde matrix, with typical entry given by $c_{k,m} = e^{j2\pi(k-1)(m-1)/M}$. This choice is ideal from a coding and energy-efficiency viewpoints. Indeed, since its entries are complex exponentials, the amplitudes of the informative signals exchanged in the network are not altered by this coding operation.

The local exchanges between nodes are short distance communication assumed to be noise-free and unidirectional. They are modeled with a directed graph whose adjacency matrix is denoted by $\mathbf{\Phi} \in \mathbb{C}^{M\times M}$. Its (i,j)-th entry, denoted by $\phi_{i,j}$, equals 1 if there is an arc from j to i. Otherwise we have $\phi_{i,j} = 0$, $i,j = 1,\ldots,M$. After the collaboration step, the collected data, for the n-th informative signals, can be written as

$$\overline{\mathbf{X}}_n = \mathbf{\Phi}\mathbf{X}_n = \mathbf{\Phi}diag(\mathbf{s}_n)\mathbf{C}^T \in \mathbb{C}^{M\times P}. \quad (5)$$

The transmission power is the same for all the nodes, and satisfy $\sum_{n=1}^{N} Tr(\overline{\mathbf{X}}_n\overline{\mathbf{X}}_n^H) = \rho$, where ρ is the total transmitted power.

2.2. Transmission to the fusion center

The data collected in the network are then transmitted to the fusion center through a frequency-flat wireless channel $\mathbf{H} \in \mathbb{C}^{K\times M}$ assumed to be constant during the communication time-frame. For the n-th informative data, the corresponding received data are given by:

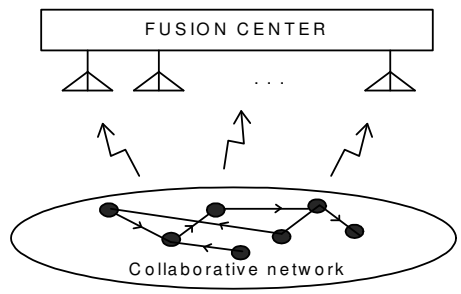

Fig. 1. Illustration of the considered communication scenario.

$$\mathbf{Y}_n = \mathbf{H}\overline{\mathbf{X}}_n + \mathbf{V}_n = \mathbf{H}\mathbf{\Phi}diag\big(\mathbf{s}_n\big)\mathbf{C}^T + \mathbf{V}_n \ \in \mathbb{C}^{K\times P}, \quad (6)$$

where $\mathbf{V}_n \in \mathbb{C}^{K\times P}$ is the additive white Gaussian noise (AWGN) matrix. Let $\mathbf{S} = [\mathbf{s}_1,\ldots,\mathbf{s}_N]^T \in \mathbb{C}^{N\times M}$ be a data matrix collecting the informative sequences of the M sensors along its columns. Given the received data $\mathbf{Y}_n$, $n = 1,\cdots,N$, the objective of the fusion center is to jointly recover $\mathbf{S}$ and the topology matrix $\mathbf{\Phi}$. Since the received data matrices can be organized in a three-way array, in the next section, we describe the multi-way analysis framework in which the problem will be solved.

3. PARALIND FORMULATION

3.1. Preliminaries

PARALIND models, introduced in [8], can be viewed as "constrained" PARAFAC models, where the columns of one of more matrix factors exhibit linear dependencies. The linear dependency profile associated with a given mode of the tensor is defined by a constraint (or interaction) matrix, generally composed of 1's and 0's. A special class of PARALIND models is represented by constrained factors (CONFAC) models [14] which restrict the columns of the constraint matrices to canonical vectors.

For a third-order tensor, the R-factor PARAFAC decomposition is the factorization in a sum of R *triple products*, each one being a rank-1 tensor. The PARAFAC decomposition of a tensor $\mathcal{X} \in \mathbb{C}^{I_1\times I_2\times I_3}$ is given by:

$$\mathcal{X} = \sum_{r=1}^{R} \mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)} \circ \mathbf{a}_r^{(3)} \leftrightarrow x_{i_1,i_2,i_3} = \sum_{r=1}^{R} a_{i_1,r}^{(1)} a_{i_2,r}^{(2)} a_{i_3,r}^{(3)}, \quad (7)$$

where $x_{i_1,i_2,i_3} = [\mathcal{X}]_{i_1,i_2,i_3}$, $a_{i_j,r}^{(j)} = [\mathbf{A}^{(j)}]_{i_j,r}$, are entries of matrix factors $\mathbf{A}^{(j)} \in \mathbb{C}^{I_j\times R}$, $j = 1,2,3$. R is the number of factors, also known as the *tensor rank*. A compact notation for the PARAFAC decomposition (7) is $\mathcal{X} = [\![\mathbf{A}^{(1)},\mathbf{A}^{(2)},\mathbf{A}^{(3)}]\!]$, where $\mathbf{A}^{(j)} = [\mathbf{a}_1^{(j)},\ldots,\mathbf{a}_R^{(j)}] \in \mathbb{C}^{I_j\times R}$, $j = 1,2,3$.

PARALIND can be viewed as a constrained PARAFAC decomposition, where some columns of one (or more) matrix factor(s) are linearly dependent [8]. In this paper, we consider a PARALIND model with linear dependencies in the first mode, i.e. $\mathbf{A}^{(1)} = \bar{\mathbf{A}}^{(1)}\mathbf{\Phi}$, where $\bar{\mathbf{A}}^{(1)} \in \mathbb{C}^{I_1\times R_1}$ is a reduced matrix whose columns correspond to the subset of R_1 columns of $\mathbf{A}^{(1)}$ that are linearly independent, and $\mathbf{\Phi} \in \mathbb{C}^{R_1\times R}$ is a constraint matrix that generate the linear dependency pattern of the columns of $\mathbf{A}^{(1)}$.

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO CEARA. Downloaded on November 18,2022 at 13:35:47 UTC from IEEE Xplore. Restrictions apply.

3.2. PARALIND data model

Let us define $y_{k,n,p} = [\mathbf{Y}_n]_{k,p}$ as the (k,n,p)-th entry of tensor $\mathcal{Y} \in \mathbb{C}^{K\times N\times P}$ collecting all the data received at the fusion center. Then, by ignoring the additive noise term for convenience, in scalar form, the end-to-end signal model (6) yields

$$y_{k,n,p} \quad = \quad \sum_{r=1}^{M}\left(\sum_{m=1}^{M} h_{k,m}\phi_{m,r}\right) s_{n,r}c_{p,r}, \tag{8}$$

which corresponds to the PARALIND decomposition. By analogy with (7), we deduce the correspondences

$$(\bar{\mathbf{A}}^{(1)}\mathbf{\Phi}, \mathbf{A}^{(2)}, \mathbf{A}^{(3)}) \longleftrightarrow (\mathbf{H\Phi}, \mathbf{S}, \mathbf{C})$$
$$(I_1, I_2, I_3, R_1, R) \longleftrightarrow (K, N, P, M, M).$$

3.3. Model identifiability issues

In what follows, we provide a succinct analysis of the identifiability of the model described by (8). It is now well known that PARAFAC possesses desirable uniqueness properties under mild conditions, the most well-known of all being stated by the *Kruskal's condition* [15]. Applied to model (8), this condition states that $\mathbf{H\Phi}, \mathbf{S}$ and $\mathbf{C}$ can be estimated up to column permutation and scaling from $\mathcal{Y}$ if $k_{(\mathbf{H\Phi})} + k_{\mathbf{S}} + k_{\mathbf{C}} \geq 2M+2$, where $k_{(.)}$ denotes the Kruskal-rank[3] of a matrix. In the case of uni-directional transmissions during the collaboration step, owing to self-loops involved in the underlined directed graph, $\mathbf{\Phi} \in \mathbb{C}^{M\times M}$ has full column rank. Also, assuming that $K \geq M$ and the wireless links between sensors and receive antennas at the fusion center are independent, $\mathbf{H}$ has full column rank, implying $rank(\mathbf{H}) = k_{(\mathbf{H\Phi})} = M$. The coding matrix $\mathbf{C} \in \mathbb{C}^{P\times M}$ has full rank by definition due to its Vandermonde structure. Given these considerations, along with the reasonable assumptions that the M nodes transmit independent information and that the length N of the transmitted data block exceeds the number of collaborating nodes, Kruskal's condition becomes $min(P, M) \geq 2$. This implies that, in practice, under the considered assumptions, the model is always identifiable.

Remark 1: If linear dependencies (and more particularly, collinearities) are present between columns of the matrices of model (8), Kruskal's condition is no longer satisfied. This could be the case of two nodes sending the same information to the fusion center or reusing the same code. For these scenarios, partial uniqueness conditions exist guaranteeing unique recovery of only part of the parameters [16].Due to space limitation, a more detailed model identifiability analysis will be developed in a forthcoming paper.

4. PROPOSED ESTIMATOR

Our goal is to derive a blind estimation algorithm to be used at the fusion center for jointly recovering the informative data matrix $\mathbf{S}$ and the network topology matrix $\mathbf{\Phi}$. The in-network coding matrix $\mathbf{C}$ is assumed to be known at the fusion center, while the channel matrix $\mathbf{H}$ is unknown and must be estimated. We propose a two-stage approach for estimating network data and nodes' connection topology.

[3]The Kruskal-rank of a matrix $\mathbf{A}$ is the maximum number κ such that every κ columns of $\mathbf{A}$ are linearly independent.

4.1. First stage: channel estimation

In this stage, only dedicated to channel estimation, the nodes avoid an effective collaboration (i.e. the exchange of their data measurements) in order to minimize energy consumption. In our model, this means that $\mathbf{\Phi} = \mathbf{I}_M$. We consider a least-squares (LS) estimator using pilot data. Each node uses a short pilot sequence of N_0 samples that will be exploited at the fusion center for estimating the channel matrix $\mathbf{H}$. We define $\mathbf{S}_0 \in \mathbb{C}^{N_0\times M}$ as the pilot sequence matrix. This matrix is chosen to be a DFT matrix satisfying $\mathbf{S}_0\mathbf{S}_0^H = \rho_0\mathbf{I}_M$ (i.e. $N_0 = M$), where ρ_0 denotes the total transmitted power during the training phase. In-network coding is not applied, meaning that $P = 1$ and $\mathbf{C} = \mathbf{1}_M^T$. Defining $\mathbf{Y}_0 \in \mathbb{C}^{K\times N_0}$ as the data matrix collected during the training phase, we have $\mathbf{Y}_0 = \mathbf{HS}_0^T + \mathbf{V}_0$, where $\mathbf{V}_0 \in \mathbb{C}^{K\times N_0}$ is the additive noise matrix. The LS estimate of the channel matrix is then given by $\widehat{\mathbf{H}} = \mathbf{Y}_0\mathbf{S}_0^*$.

Remark 2: As an alternative to the pilot-based LS estimator, we can also use a blind ALS estimator. The idea is to trade off pilots for coding, contrary to the previous method. The use of coding allows one to build a PARAFAC model $\mathcal{Y}_0 = [\![\mathbf{H}, \mathbf{S}, \mathbf{C}]\!]$ for the data collected at the fusion center during the first stage, where $\mathbf{S}$ now contains useful network data (instead of pilots). We have omitted further details on this approach due to space limitation.

4.2. Second stage: joint data and topology estimation

Once the channel matrix is estimated, the second stage of the proposed estimator is concerned with the joint estimation of informative data and nodes' connection topology. The data matrix $\mathbf{Y}_n \in \mathbb{C}^{K\times P}$ defined in (6) can be seen as the n-th slice of the tensor $\mathcal{Y} \in \mathbb{C}^{K\times N\times P}$, obtained by fixing its second mode to index n. Defining $\mathbf{y}_n = vec(\mathbf{Y}_n) \in \mathbb{C}^{PK\times 1}$, and applying property (1), we obtain $\mathbf{y}_n = \big(\mathbf{C}\diamond(\mathbf{H\Phi})\big)\mathbf{s}_n$. By stacking columnwise the N vectors $\mathbf{y}_1, \ldots, \mathbf{y}_N$, we get

$$\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N] = \big(\mathbf{C}\diamond(\mathbf{H\Phi})\big)\mathbf{S}^T \ \in \mathbb{C}^{PK\times N}. \tag{9}$$

Now, let $\mathbf{Z} = [\mathbf{Y}_1, \ldots, \mathbf{Y}_N] \in \mathbb{C}^{K\times NP}$ be a matrix stacking columnwise N second-mode slices of the data tensor $\mathcal{Y}$. Using property (2) yields

$$\begin{aligned}\mathbf{Z}^T \quad &= \quad \begin{bmatrix}\mathbf{Y}_1^T\\ \vdots \\ \mathbf{Y}_N^T\end{bmatrix} = \begin{bmatrix}\mathbf{C}D_1(\mathbf{S})\\ \vdots \\ \mathbf{C}D_N(\mathbf{S})\end{bmatrix}\mathbf{\Phi}^T\mathbf{H}^T \\ &= \quad (\mathbf{S}\diamond\mathbf{C})\mathbf{\Phi}^T\mathbf{H}^T \ \in \mathbb{C}^{NP\times K}.\end{aligned} \tag{10}$$

Defining $\mathbf{z} = vec(\mathbf{Z}^T) \in \mathbb{C}^{KNP\times 1}$ and applying property (3) to (10) yields

$$\mathbf{z} = \big(\mathbf{H}\otimes(\mathbf{S}\diamond\mathbf{C})\big)\boldsymbol{\phi}, \tag{11}$$

where $\boldsymbol{\phi} = vec(\mathbf{\Phi}^T)$.

The PARALIND core equations (9) and (11) are the bases for this stage. Let us introduce the noise contribution in these core equations, by defining $\tilde{\mathbf{Y}} = \mathbf{Y} + \mathbf{V_Y}$ and $\tilde{\mathbf{z}} = \mathbf{z} + \mathbf{v_z}$, where $\mathbf{V_Y} \in \mathbb{C}^{PK\times N}$ and $\mathbf{v_z} \in \mathbb{C}^{KNP\times 1}$ are the corresponding additive noise terms. The estimates of $\mathbf{S}$ and $\mathbf{\Phi}$ can be found by optimizing, respectively, the following least squares (LS) criteria:

$$\widehat{\mathbf{S}} = \underset{\mathbf{S}}{argmin}\left\|\tilde{\mathbf{Y}} - \big(\mathbf{C}\diamond(\widehat{\mathbf{H}}\mathbf{\Phi})\big)\mathbf{S}^T\right\|_F^2, \tag{12}$$

$$\widehat{\boldsymbol{\phi}} = \underset{\boldsymbol{\phi}}{argmin}\left\|\tilde{\mathbf{z}} - \big(\widehat{\mathbf{H}}\otimes(\mathbf{S}\diamond\mathbf{C})\big)\boldsymbol{\phi}\right\|_F^2, \tag{13}$$

where $\widehat{\mathbf{H}}$ is the estimate of the channel matrix obtained in the first stage. Both problems are solved iteratively using the well-known

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO CEARA. Downloaded on November 18,2022 at 13:35:47 UTC from IEEE Xplore. Restrictions apply.

alternating least squares (ALS) algorithm [17]. Below is a summary of the proposed algorithm.

Two-Stage Estimation Algorithm

First Stage: Assuming $\mathbf{\Phi} = \mathbf{I}_M$ and $\mathbf{C} = \mathbf{1}_M^T$ $(P = 1)$, obtain an initial channel estimate $\widehat{\mathbf{H}}$ using pilot sequences: $\widehat{\mathbf{H}} = \tilde{\mathbf{Y}}_0 \mathbf{S}_0^*$;

Second Stage: From the previously obtained channel estimate $\widehat{\mathbf{H}}$, apply the ALS algorithm to obtain the estimates $\widehat{\mathbf{S}}$ and $\widehat{\mathbf{\Phi}}$ of the data and connection topology matrices, respectively.

Initialization: Set $t = 0$; Initialize $\widehat{\mathbf{S}}(t = 0)$;

(2.1) $t = t + 1$;

(2.2) Compute $\widehat{\boldsymbol{\phi}}(t) = \left(\widehat{\mathbf{H}} \otimes (\widehat{\mathbf{S}}(t-1) \diamond \mathbf{C})\right)^{\dagger} \tilde{\mathbf{z}}$;

(2.3) Reshape $\widehat{\phi}(t)$ to obtain $\widehat{\mathbf{\Phi}}(t)$;

(2.4) Compute $\widehat{\mathbf{S}}^T(t) = \left(\mathbf{C} \diamond (\widehat{\mathbf{H}}\widehat{\mathbf{\Phi}}(t))\right)^{\dagger} \tilde{\mathbf{Y}}$;

(2.5) Repeat steps (2.1)-(2.4) until convergence.

Define $e(t) = \|\tilde{\mathbf{Y}} - (\mathbf{C} \diamond (\widehat{\mathbf{H}}\widehat{\mathbf{\Phi}}(t))\widehat{\mathbf{S}}^T(t)\|_F^2$ as the model reconstruction error calculated at the t-th iteration. We declare the convergence when $|e(t) - e(t-1)| < 10^{-6}$. Convergence to the global minimum is always achieved within a few iterations (usually between 10 and 30) due to the knowledge of the code matrix at the fusion center. Such a knowledge also avoids permutation ambiguity in the estimated $\widehat{\mathbf{H}}$, $\widehat{\mathbf{S}}$, and $\widehat{\mathbf{\Phi}}$. The column scaling ambiguity affecting $\widehat{\mathbf{S}}$ and $\widehat{\mathbf{\Phi}}$ are solved through normalization.

5. NUMERICAL RESULTS

Some computer simulations have been carried out for evaluating the performance of the proposed tensor-based estimator. Our experiments consider a network of $M = 4$ collaborating nodes and two possible connection topologies, as follows:

$$\mathbf{\Phi}_1 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}, \quad \mathbf{\Phi}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The first one represents a connected digraph ("grid" topology) while the second corresponds to a unconnected digraph. The remaining system parameters are $P = P_0 = 4$, $N_0 = 4$, and $N = 10$. We consider $K = 4, 8$ or 16 antennas at the fusion center. For evaluating the accuracy of network data recovery and connection topology estimation, we calculate the following normalized error metrics:

$$E_{\mathbf{S}} = \frac{1}{L}\sum_{l=1}^{L} \frac{\|\widehat{\mathbf{S}}_{(l)} - \mathbf{S}_{(l)}\|_F^2}{\|\mathbf{S}_{(l)}\|_F^2}, \quad E_{\mathbf{\Phi}} = \frac{1}{LM^2}\sum_{l=1}^{L} |\widehat{\mathbf{\Phi}}_{(l)} - \mathbf{\Phi}|_0,$$

where L denotes the number of Monte Carlo runs, while $\widehat{\mathbf{S}}_{(l)}$ and $\widehat{\mathbf{\Phi}}_{(l)}$ are the final estimates of the informative data and topology matrices at the l-th run, and $|\cdot|_0$ denotes the ℓ_0-norm. In each run, the total transmitted powers ρ_0 and ρ for the training and collaboration phases are chosen to satisfy the signal to noise ratio measures SNR$_0$[dB]= $10log_{10}(\|\mathbf{Y}_0\|_F^2/\|\mathbf{V}_{\mathbf{Y}_0}\|_F^2)$ and SNR[dB]= $10log_{10}(\|\mathbf{Y}\|_F^2/\|\mathbf{V}_{\mathbf{Y}}\|_F^2)$, respectively.

Figure 2 depicts the $E_{\mathbf{S}}$ vs. SNR curves for $K = 4$ and 16. The results are satisfactory and the performance of network data estimation is nearly the same for both topologies. As expected, more accurate estimates are obtained as more antennas are used at the fusion center. In Figure 3, we turn our attention to connection topology estimation. The left-hand and right-hand sub-figures are associated with Topologies 1 and 2, respectively. Considerable

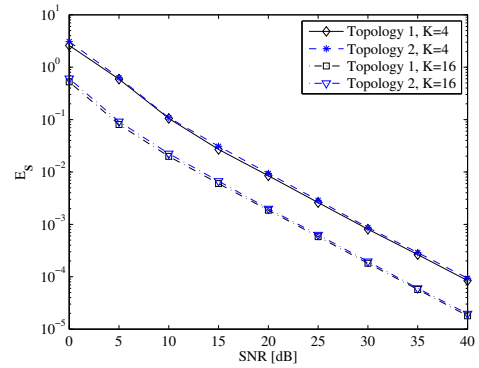

Fig. 2. $E_{\mathbf{S}}$ vs. SNR performance.

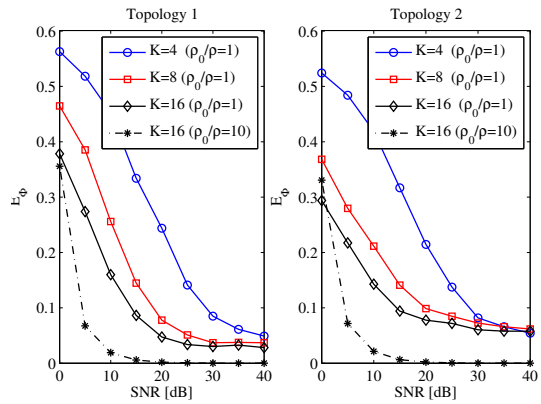

Fig. 3. $E_{\mathbf{\Phi}}$ vs. SNR performance.

performance improvements are obtained by increasing K, especially in noise-limited scenarios. Note that we consider $\rho_0/\rho = 1$ and $\rho_0/\rho = 10$. In this first situation, equal transmission powers are allocated to the first (channel estimation) and second (data recovery) stages. In the second one, the system dedicates more energy resources to channel estimation. It can be seen that the network topology is more accurately estimated in the second situation. For both topologies, $K = 16$ and SNR=10dB, the average rate of identifying erroneously the ("on/off") status of a node connection is around 2% when $\rho_0/\rho = 10$ and grows to 15% when $\rho_0/\rho = 1$.

6. CONCLUSION

We have resorted to PARAFAC/PARALIND modeling to solve the problem of information recovery and network topology estimation in collaborative networks with the help of a fusion center equipped with an antenna array. The proposed tensor-based solution assumes an in-network coding strategy and is based on a two-stage estimation approach that may use pilot data or operate in a completely blind setting. In particular, our results indicate that more accurate estimates of the connection topology matrix are obtained as more power is dedicated to the training phase, which is the bottleneck of the overall estimator performance. Although not shown here, we have also observed that, for lower SNRs, the performance of our estimator improves for networks with higher degrees of connectivity. However, even for weakly connected networks, we believe that the accuracy of the topology estimation can be improved by exploiting the sparsity of $\mathbf{\Phi}$. This issue will be addressed in a future work.

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO CEARA. Downloaded on November 18,2022 at 13:35:47 UTC from IEEE Xplore. Restrictions apply.

7. REFERENCES

[1] R. Olfati-Saber, A. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. of the IEEE*, vol. 95, no. 1, pp. 215–233, January 2007.

[2] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: efficient protocols and outage behavior," *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 3062–3080, Dec. 2004.

[3] K. J. Ray Liu, A. K. Sadek, W. Su, and A. Kwasinski, "Cooperative communications and networking, Cambridge University Press, 2009."

[4] F. Roemer and M. Haardt, "Tensor-based channel estimation and iterative refinements for two-way relaying with multiple antennas and spatial reuse," *IEEE Trans. Signal Process.*, vol. 58, no. 11, pp. 5720–5735, Nov. 2010.

[5] A. Y. Kibangou and A. L F. de Almeida, "Distributed PARAFAC based DS-CDMA blind receiver for wireless sensor networks," *Proc. SPAWC 2010*, Marrakech, Morocco, Jun. 2010.

[6] A. Y. Kibangou and A. L F. de Almeida, "Distributed Khatri-Rao space-time coding and decoding for cooperative networks," *Proc. EUSIPCO 2011*, Barcelona, Spain, Aug. 2011.

[7] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis," *UCLA Working Papers in Phonetics*, 16th edition, Dec. 1970.

[8] R. Bro, R. A. Harshman, N. D. Sidiropoulos, and M. E. Lundy, "Modeling multi-way data with linearly dependent loadings," *J. Chemometrics*, vol. 23, no. 7-8, pp. 324-340, July-Aug. 2009.

[9] N.D. Sidiropoulos, G.B. Giannakis, and R. Bro, "Blind PARAFAC receivers for DS-CDMA systems," *IEEE Trans. Signal Process.*, vol. 48, no. 3, pp. 810–823, March 2000.

[10] A.L.F. de Almeida, G. Favier, and J.-C.-M. Mota, "PARAFAC-based unified tensor modeling for wireless communication systems with application to blind multiuser equalization," *Signal Processing*, vol. 87, no. 2, pp. 337–351, February 2007.

[11] D. Nion and L. De Lathauwer, "A block component model-based blind DS-CDMA receiver," *IEEE Trans. Signal Process.*, vol. 56, no. 11, pp. 5567 –5579, November 2008.

[12] A.Y. Kibangou and G. Favier, "Blind equalization of nonlinear channels using tensor decompositions with code/space/time diversities," *Signal Processing*, vol. 89, no. 2, pp. 133–143, February 2009.

[13] C. A. R Fernandes, A. L. F. de Almeida, and D. B. da Costa, " Unified tensor modeling for blind receivers in multiuser uplink cooperative systems," *IEEE Signal Process. Lett.*, vol. 19, no. 5, pp. 247-250, May 2012.

[14] A. L. F. de Almeida, G. Favier, and J. C. M. Mota, "A constrained factor decomposition with application to MIMO antenna systems," *IEEE Trans. Signal Process.*, vol. 56, n. 6, pp. 2429–2442, June 2008.

[15] J.B. 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.

[16] X. Guo, S. Miron, D. Brie, and A. Stegeman, "Uni-mode and partial uniqueness conditions for CANDECOMP/PARAFAC of three-way arrays with linearly dependent loadings," *SIAM J. Matrix Anal. Appl.*, vol. 33, no. 1, pp. 111–129, 2012.

[17] A. Smilde and R. Bro and P. Geladi "Multi-way analysis: application in chemical sciences" *Wiley*, 2004.

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO CEARA. Downloaded on November 18,2022 at 13:35:47 UTC from IEEE Xplore. Restrictions apply.