A Block Component Model-Based Blind DS-CDMA Receiver

Dimitri Nion and Lieven De Lathauwer, *Senior Member, IEEE*

***Abstract*—In this paper, we consider the problem of blind multiuser separation-equalization in the uplink of a wideband DS-CDMA system, in a multipath propagation environment with intersymbol-interference (ISI). To solve this problem, we propose a multilinear algebraic receiver that relies on a new third-order tensor decomposition and generalizes the parallel factor (PARAFAC) model. Our method is deterministic and exploits the temporal, spatial and spectral diversities to collect the received data in a third-order tensor. The specific algebraic structure of this tensor is then used to decompose it in a sum of user's contributions. The so-called Block Component Model (BCM) receiver does not require knowledge of the spreading codes, the propagation parameters, nor statistical independence of the sources but relies instead on a fundamental uniqueness condition of the decomposition that guarantees identifiability of every user's contribution. The development of fast and reliable techniques to calculate this decomposition is important. We propose a blind receiver based either on an alternating least squares (ALS) algorithm or on a Levenberg-Marquardt (LM) algorithm. Simulations illustrate the performance of the algorithms.**

***Index Terms*—Blind signal extraction, block component model (BCM), CDMA, higher-order tensor, PARAFAC.**

I. Introduction

In signal processing for telecommunications, the conception of highly efficient wireless systems that guarantee simultaneous multiuser access is essential. In training-based systems, the users periodically transmit a training sequence known by the receiver, which then estimates the parameters of the propagation channel, such as delays caused by multiple reflections of the radio waves on obstacles encountered. However, 20% of the bandwidth is devoted to training in GSM (up to 40% in UMTS). Blind methods are thus attractive so as to guarantee a high communication rate by eliminating (or reducing) the training sets. Moreover, in fast-time varying channels, the lack of stationarity makes training not efficient. Blind approaches can also be useful if severe multipath fading occurs during the training period. Besides, in some situations such as eavesdropping, the use of training sequences is not possible. It is the latter application that motivated this research.

Usually, blind techniques rely on temporal properties of the signals or spatial properties of the receiver. Temporal blind methods enforce known signal properties on their estimates such as finite alphabet (FA) or constant modulus (CM) of the signal constellation [1], [2]. Other algorithms use the constant symbol rate of a digital signal, that allows temporal oversampling so as to create "virtual" channels before equalization [3], [4]. Another common approach is based on higher-order statistics (HOS), which relies on probability distributions of the source signals [5], [6]. HOS methods usually require large amounts of data, i.e., the channel has to be stationary over a large number of symbol periods, which can limit the reliability of HOS-based algorithms.

Spatial blind methods estimate the signals impinging on an antenna array in two steps by considering the independent linear combinations of the source signals received. The directions of arrival (DOAs) are first estimated by direction-finding techniques such as MUSIC [7] or ESPRIT [8]. The array outputs are then optimally combined to extract each signal from interference and noise. However, the performance of these algorithms depends strongly on the reliability of prior spatial information (antenna calibration or special geometry). The temporal and spatial diversities may also be combined to build blind space-time equalizers [9].

Most of these blind methods in the literature are formulated in terms of second-order algebra and consist of solving a matrix decomposition problem of the form $\mathbf{X} = \mathbf{H}\cdot\mathbf{S}$, where $\mathbf{X}, \mathbf{H}$ and $\mathbf{S}$ are the observation, channel and source matrices, respectively. We refer to [10] and references therein for algebraic solutions to this problem. More recently, multilinear algebraic methods have received much attention in signal processing [11]–[17]. They revealed to be a powerful tool for (HOS-)based independent component analysis [13], array processing [14], and purely deterministic blind separation of CDMA signals [15].

In [15], the problem of blind multiuser separation-equalization of DS-CDMA signals impinging on an antenna array is considered. If the multipath reflectors are in the far field and in the absence of intersymbol-interference (ISI), the authors have shown that the received data can be arranged in a three-way array or third-order tensor that follows a so-called parallel factor

Manuscript received March 23, 2007; revised April 26, 2008. First published June 10, 2008; current version published October 15, 2008. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Kostas Berberidis. This work was supported in part by the French délégation générale pour l'armement (DGA), by the Research Council K.U. Leuven by Grant GOA-AMBioRICS, CoE EF/05/006 Optimization in Engineering, CIF1, by the Flemish Government under F.W.O. Project G.0321.06, and F.W.O. research communities ICCoS, ANMMM, by Belgian Federal Science Policy Office under IUAP P6/04, and by the E.U.: ERNSI. The major part of this work was carried out when the authors were with ETIS Lab, UMR 8051 (ENSEA, CNRS, Univ. Cergy-Pontoise), France.

D. Nion is with the Telecommunications Division of the Electronics and Computer Engineering Department, Technical University of Crete, Crete, Greece 731 00 (e-mail: nion@telecom.tuc.gr).

L. De Lathauwer is with the Research Group ESAT-SCD, Katholieke Universiteit Leuven (K.U. Leuven), B-3001 Leuven-Heverlee, Belgium. He is also with the K.U. Leuven Campus Kortrijk, Subfaculty Science and Technology, 8500 Kortrijk, Belgium (e-mail: delathau@esat.kuleuven.be).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2008.926982

(PARAFAC) model. In [16] and [17], only the far-field reflection assumption is taken into account and the authors propose algebraic methods more general than PARAFAC to deal with ISI caused by large delay spread.

In this paper, we address the problem of blind separation-equalization of DS-CDMA signals received by an antenna array for a more general propagation scenario: the multipath reflectors are not necessarily located in the far field of the antenna array and ISI may occur on several consecutive symbols. This propagation scenario is the same as the one studied in [18].

This paper shows how this problem can be solved by a new multilinear model, the so-called block component model (BCM) [19]–[21]. The BCM in this paper generalizes PARAFAC and the multilinear models of [15] and [16]. We also present *two* different algorithms to compute the decomposition of the BCM: an alternating least squares (ALS) algorithm and a Levenberg-Marquardt (LM) algorithm, which are themselves generalizations of their PARAFAC versions. These algorithms take the Toeplitz structure of the symbol matrix into account.

The system is described in terms of its discrete-time baseband-equivalent model where signals, codes, and channels are represented by samples of their complex envelopes taken at the chip rate. We start with the observation that the temporal, spatial and spectral diversities give a third-order tensor structure to the received data. We then exploit the algebraic structure of this tensor of observations, which follows a BCM. In this context, the BCM leads to a deterministic blind receiver that jointly performs *separation* and *equalization* from the convolutive mixtures received by the antennas. This receiver does not assume knowledge of the CDMA codes, channel impulse responses and antenna array responses neither statistical independence between the transmitted signals.

The *separation* of users' signals rather relies on the *uniqueness* of the BCM decomposition, which consists of writing the tensor of observations as sum of users' contributions. Each contribution consists of factors that represent the antenna array response, the transmitted symbols with interference, and the channel impulse response convolved with the CDMA code. The *equalization* (or echo canceling) is performed within each contribution by imposing a *Toeplitz* structure to the matrix that holds the transmitted symbols with interference.

Our working assumptions are the following:

1) The spreading codes are not known by the receiver and may not be orthogonal. The codes are assumed to be symbol-periodic and their length I is known or has been estimated (e.g., using cyclostationarity tests [22]).
2) We assume that the different user sequences have been synchronized at the symbol level.
3) Every multipath channel between user r and antenna k is supposed to be time-invariant over J symbols. Since the method we propose is deterministic, it can work for small frame sizes such that this assumption is valid.

The paper is organized as follows. In Section II, we recall the instantaneous and convolutive models that apply to DS-CDMA data received by an antenna array. In Section III, we introduce some multilinear algebra prerequisites. In Section IV, we discuss the PARAFAC decomposition. In Section V, we look at the convolutive data model from a multilinear algebraic perspective and we introduce the BCM blind receiver. In Section VI, we derive the ALS and LM algorithms to compute the decomposition of this model. Finally, Section VII illustrates the performance of these algorithms with six experiments. Section VIII is the conclusion.

Parts of this manuscript have appeared in the conference paper [23]. Some mathematical aspects are developed in more detail in [19]–[21].

Notation: A calligraphic letter $\mathcal{Y}$ denotes a third-order tensor and a bold-face capital letter $\mathbf{Y}$ denotes a matrix. Vectors are written in bold-face lower-case $\mathbf{y}$ and $\mathbf{y}_k$ indicates the kth column of matrix $\mathbf{Y}$. Scalars are lower-case y. The scalar y_i indicates the ith element of vector $\mathbf{y}$ and the scalar y_{ij} denotes the element on the ith row and the jth column of matrix $\mathbf{Y}$. Capitals are also used to denote index upper bounds $(i = 1, 2, \ldots, I)$ and matrix dimensions $\mathbf{Y}^{(I \times J)}$. The transpose, complex conjugate, complex conjugate transpose, and pseudoinverse are denoted by $\cdot^T$, $\cdot^*$, $\cdot^H$, and $\cdot^\dagger$, respectively. $\mathbf{1}_I$ and $\mathbf{0}_I$ denote column-vectors of ones and zeros, respectively. The Kronecker product is denoted by $\otimes$. The condition number of $\mathbf{Y}$, which is the ratio of its highest singular value to its smallest, is denoted by $\kappa(\mathbf{Y})$. The norm $\|\cdot\|_F$ is the Frobenius norm. Furthermore, the operator $\mathrm{vec}(\cdot)$ builds a vector from a matrix by stacking the columns of this matrix one above the other such that the element a_{ij} of the $(I \times J)$ matrix $\mathbf{A}$ becomes the element at position $i + (j-1)I$ of the vector $\mathrm{vec}(\mathbf{A})$. If a matrix $\mathbf{Y}$ is partitioned along one dimension, i.e., $\mathbf{Y}$ is of size $(I \times RJ)$ or $(RI \times J)$, then the rth $(I \times J)$ block is denoted by $\mathbf{Y}_r$.

II. Data Model: Analytic Form

A. Instantaneous Model

We consider R users transmitting, at the same time within the same bandwidth, frames of J symbols spread by DS-CDMA codes of length I towards an array of K antennas. In a direct-path only propagation scenario, the assumption that the channel is noiseless and memoryless leads to the following instantaneous model without interchip-interference (ICI):

$$y_{ijk} = \sum_{r=1}^{R} c_{ir} s_{jr} a_{kr} \tag{1}$$

where y_{ijk} is the output of the kth antenna for chip i and symbol j. For a given user r, the sequence $\{s_{jr}\}_{j=1}^{J}$ holds the transmitted symbols, $\{c_{ir}\}_{i=1}^{I}$ holds the CDMA code and $\{a_{kr}\}_{k=1}^{K}$ holds the fading factors multiplied by the antenna gains between this user and the K antennas.

In the case of multipath propagation, the channel for each user can be modeled as a finite impulse response (FIR) filter. If the delay spread is small (i.e., in the order of a few chips), ISI can be avoided by adopting a "guard chips" or "discard prefix" strategy [15]. Moreover, if multipath effects are in the far field, then the data have a similar structure as before. One just needs to replace in (1) c_{ir} by h_{ir}, the ith element of the convolution between the spreading code of user r and its channel impulse response

$$y_{ijk} = \sum_{r=1}^{R} h_{ir} s_{jr} a_{kr}. \tag{2}$$

In Section IV, we recall that this analytic model can algebraically be written as a PARAFAC model.

B. Convolutive Model

Let us focus on a more complex situation where multipath propagation with large delay spread leads to ISI, and where the multipath reflectors are not necessarily located in the far field. If T_s is the symbol period and $h_r(t)$ the spreading waveform of the rth user, then the baseband signal transmitted by this user is

$$e_r(t) = \sum_{j=1}^{J} s_{jr} h_r(t - jT_s).$$

Let $T_c = T_s/I$ be the chip period, and $g(t)$ the chip pulse (e.g., a raised cosine). Then the spreading waveform $h_r(t)$ is given by

$$h_r(t) = \sum_{i=1}^{I} c_{ir} g(t - iT_c).$$

The signal $e_r(t)$ is transmitted over a specular multipath channel with P discrete paths, each one being characterized by its own delay τ_{rp}, angle of arrival θ_{rp} and attenuation β_{rp}, where p denotes the path index. Let L denote the channel length at the symbol rate, meaning that interference is occurring over L symbols.

The received signal impinging on the array of K antennas is then sampled at the chip rate. If we denote by $y_{rp}(i,j,k)$ the ith chip within the jth symbol period of the rth user's signal received by the kth antenna from the pth path, we get

$$y_{rp}(i,j,k) = \beta_{rp} a_k(\theta_{rp}) \sum_{l=1}^{L} h_{rp}\left(i + (l-1)I\right) s_{j-l+1,r}. \tag{3}$$

In (3), $h_{rp}(i + (l-1)I)$ is the sample of $h_r(t - \tau_{rp})$ at instant $(i + (l-1)I)T_c$ and $a_k(\theta_{rp})$ is the response of the kth antenna to the signal of the rth user coming from the pth path with an angle of arrival θ_{rp}.

Finally, by summing the contributions of the P paths for each of the R users, we get the expression for one sample y_{ijk} of the overall received signal, which stands for the convolutive model

$$y_{ijk} = \sum_{r=1}^{R} \sum_{p=1}^{P} \beta_{rp} a_k(\theta_{rp}) \sum_{l=1}^{L} h_{rp}\left(i + (l-1)I\right) s_{j-l+1,r} \tag{4}$$

where r, p, and l are the user, path, and interfering symbol index, respectively. In Section V, we will show that this convolutive model can be algebraically written as a new model, the so-called BCM. Our approach generalizes the results of [16], where we assumed that all multipath reflectors are in the far field.

Remark: Note that we have taken the same value of P and L for all users. This was done for notational convenience. Our methods still work if P and L are user-dependent. However, in the latter case the formal derivation of the maximum number of users for which uniqueness of the model is guaranteed, becomes a harder problem than the one in [20], of which the results are reported in Section V-B.

III. Multilinear Algebra Prerequisites

Definition 1. (Mode-n Product): The mode-1 product of a third-order tensor $\mathcal{Y} \in \mathbb{C}^{L\times M\times N}$ by a matrix $\mathbf{A} \in \mathbb{C}^{I\times L}$, denoted by $\mathcal{Y} \bullet_1 \mathbf{A}$, is an $(I \times M \times N)$-tensor with elements defined, for all index values, by

$$(\mathcal{Y} \bullet_1 \mathbf{A})_{imn} = \sum_{l=1}^{L} y_{lmn} a_{il}.$$

Similarly, the mode-2 product by a matrix $\mathbf{B} \in \mathbb{C}^{J\times M}$ and the mode-3 product by $\mathbf{C} \in \mathbb{C}^{K\times N}$ are the $(L \times J \times N)$ and $(L \times M \times K)$ tensors, respectively, with elements defined by

$$(\mathcal{Y} \bullet_2 \mathbf{B})_{ljn} = \sum_{m=1}^{M} y_{lmn} b_{jm},$$

$$(\mathcal{Y} \bullet_3 \mathbf{C})_{lmk} = \sum_{n=1}^{N} y_{lmn} c_{kn}.$$

In this notation, the matrix product $\mathbf{Y} = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^T$ takes the form of $\mathbf{Y} = \mathbf{S} \bullet_1 \mathbf{U} \bullet_2 \mathbf{V}$.

Definition 2. (Rank-1 Tensor): The third-order tensor $\mathcal{Y} \in \mathbb{C}^{I\times J\times K}$ is rank-1 if its elements can be written as $y_{ijk} = a_i b_j c_k$, where $\mathbf{a} \in \mathbb{C}^{I\times 1}$, $\mathbf{b} \in \mathbb{C}^{J\times 1}$ and $\mathbf{c} \in \mathbb{C}^{K\times 1}$.

This definition generalizes the definition of a rank-1 matrix: $\mathbf{A} \in \mathbb{C}^{I\times J}$ has rank 1 if $\mathbf{A} = \mathbf{a} \cdot \mathbf{b}^T$.

Definition 3. (Tensor Rank): The rank of $\mathcal{Y}$ is defined as the minimum number of rank-1 tensors yielding $\mathcal{Y}$ in a linear combination.

Definition 4. (Frobenius Norm): The Frobenius norm of the tensor $\mathcal{Y} \in \mathbb{C}^{I\times J\times K}$ is defined by

$$\|\mathcal{Y}\|_F = \sqrt{\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K} |y_{ijk}|^2}.$$

Definition 5. (Matrix Representation of a Tensor): The three standard matrix representations of a third-order tensor $\mathcal{Y} \in \mathbb{C}^{I\times J\times K}$, denoted by $\mathbf{Y}^{(IK\times J)}$, $\mathbf{Y}^{(JI\times K)}$ and $\mathbf{Y}^{(KJ\times I)}$ are defined by

$$\begin{cases} \left[\mathbf{Y}^{(IK\times J)}\right]_{(i-1)K+k,j} = y_{ijk} \\ \left[\mathbf{Y}^{(JI\times K)}\right]_{(j-1)I+i,k} = y_{ijk} \\ \left[\mathbf{Y}^{(KJ\times I)}\right]_{(k-1)J+j,i} = y_{ijk}. \end{cases}$$

$\mathbf{Y}^{(IK\times J)}$ results from the row-wise concatenation of the I slices $\mathbf{Y}_i$ of size $(K \times J)$ in Fig. 1. In the same way, $\mathbf{Y}^{(JI\times K)}$ and $\mathbf{Y}^{(KJ\times I)}$ result from the concatenation of the slices $\mathbf{Y}_j$ and $\mathbf{Y}_k$, respectively. By convention, when a partitioned matrix is built, indices to the right vary more rapidly than indices to the left. In this notation, $\mathbf{Y}^{(IK\times J)}$ results from concatenation of I blocks of size $K \times J$ while $\mathbf{Y}^{(KI\times J)}$ results from that of K blocks of size $I \times J$.

IV. Parallel Factor Analysis

A *Canonical* or a *Parallel Factor Decomposition* of a tensor $\mathcal{Y} \in \mathbb{C}^{I\times J\times K}$ with elements denoted by y_{ijk} is a decomposition

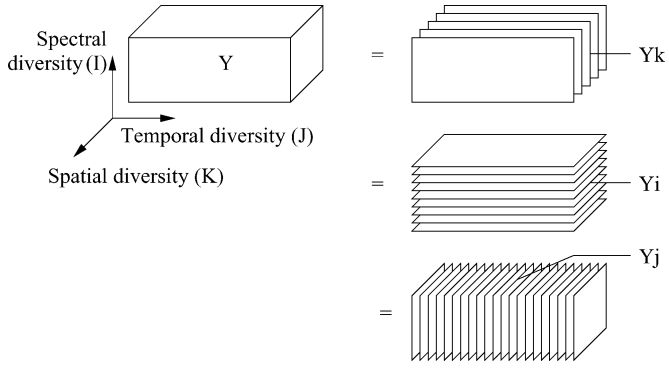


Fig. 1. The three ways of matricizing a third-order tensor.

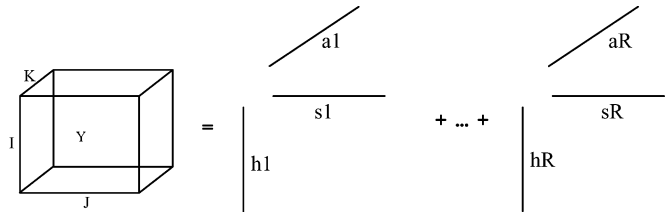


Fig. 2. Schematic representation of the PARAFAC model.

of $\mathcal{Y}$ as a linear combination of a minimal number of rank-1 tensors

$$y_{ijk} = \sum_{r=1}^{R} \mathbf{h}_r(i)\mathbf{s}_r(j)\mathbf{a}_r(k) \tag{5}$$

where $\mathbf{h}_r$, $\mathbf{s}_r$, $\mathbf{a}_r$, are the rth columns of matrices $\mathbf{H} \in \mathbb{C}^{I\times R}$, $\mathbf{S} \in \mathbb{C}^{J\times R}$, and $\mathbf{A} \in \mathbb{C}^{K\times R}$, respectively, and where i, j, and k denote the row index. This trilinear model, represented in Fig. 2, was introduced in [24] and [25]. Around 1970, it was independently reintroduced in Psychometrics [26] and Phonetics [27]. Later on, the decomposition was also applied in Chemometrics and food industry [28]. PARAFAC has also found its way to data analysis [29] and wireless communications [14], [15], [30].

From (2) it is clear that PARAFAC can only be unique up to some trivial indeterminacies: any permutation of the rank-1 terms in (5) leads to the same tensor $\mathcal{Y}$ and any scaling and counterscaling of the factors of the rank-1 terms also lead to the same tensor $\mathcal{Y}$. PARAFAC uniqueness is studied in [31]–[33]. A specific case is where $\mathbf{A}$ and $\mathbf{S}$ are both full column rank R, and $\mathbf{H}$ does not contain colinear columns. In this case, PARAFAC is unique up to its trivial indeterminacies [34].

In the next section, we show that the problem of *blind separation-equalization* of *convolutive* mixtures (4) can be solved by a new tensor decomposition that is more general than PARAFAC.

V. BCM

A. Convolutive Model: Algebraic Form

We start with (3), in which $y_{rp}(i,j,k)$, for fixed indices r and p, and varying indices i, j and k, can be considered as an element of an $(I\times J\times K)$ tensor $\mathcal{Y}_{rp}$. The samples $h_{rp}(i+(l-1)I)$ are the entries of an $(I\times L)$ matrix $\mathbf{H}_{rp}$, the symbols $s_{j-l+1,r}$ are the entries of an $(L\times J)$ Toeplitz matrix $\mathbf{S}_r^T$ and the coefficients $\beta_{rp}a_k(\theta_{rp})$ are the entries of a $(K\times 1)$ vector $\mathbf{a}_{rp}$. The tensor $\mathcal{Y}_{rp}$ can, thus, be represented by

$$\mathcal{Y}_{rp} = \mathbf{H}_{rp} \bullet_2 \mathbf{S}_r \bullet_3 \mathbf{a}_{rp}, \tag{6}$$

This tensor represents the contribution of the pth path associated to the rth user. Considering all P paths, the overall contribution of the rth user is given by

$$\mathcal{Y}_r = \sum_{p=1}^{P} \mathcal{Y}_{rp} = \sum_{p=1}^{P} \mathbf{H}_{rp} \bullet_2 \mathbf{S}_r \bullet_3 \mathbf{a}_{rp}. \tag{7}$$

This equation can be rewritten in a more compact way as

$$\mathcal{Y}_r = \mathcal{H}_r \bullet_2 \mathbf{S}_r \bullet_3 \mathbf{A}_r \tag{8}$$

where $\mathcal{H}_r$ is an $(I\times L\times P)$ tensor with each $(I\times L)$ frontal slice equal to one of the $\mathbf{H}_{rp}$ matrices: $[\mathcal{H}_r]_{:,:,p} = \mathbf{H}_{rp}$. The $(J\times L)$ Toeplitz matrix $\mathbf{S}_r$ holds the interfering symbols and the $(K\times P)$ matrix $\mathbf{A}_r$ contains the set of vectors $\mathbf{a}_{rp}$. Finally, we consider R users transmitting at the same time, each along P paths, and we obtain the following tensor equivalent of (4) after summing the R contributions

$$\mathcal{Y} = \sum_{r=1}^{R} \mathcal{H}_r \bullet_2 \mathbf{S}_r \bullet_3 \mathbf{A}_r. \tag{9}$$

Thus, the tensor of observations $\mathcal{Y} \in \mathbb{C}^{I\times J\times K}$ follows a BCM, represented in Fig. 3 [20], [23]. An equivalent but formally different model was proposed in [18].

The indeterminacies of the BCM defined in (9) are characterized as follows:

$$\mathcal{Y} = \sum_{r=1}^{R} \left(\gamma_r^{-1}\mathcal{H}_r \bullet_3 \mathbf{U}_r\right) \bullet_2 (\gamma_r \mathbf{S}_r) \bullet_3 \left(\mathbf{U}_r^{-1}\mathbf{A}_r\right) \tag{10}$$

where the scalar γ_r and the nonsingular matrix $\mathbf{U}_r$ represent the indeterminacy in modes two and three, respectively. Note that the indeterminacy in the second mode involves a scalar rather than a matrix due to the Toeplitz structure of $\mathbf{S}_r$. In fact, replacing γ_r by a $(L\times L)$ nonsingular matrix $\mathbf{V}_r$ would destroy the Toeplitz structure of $\mathbf{S}_r$, unless $\mathbf{V}_r = \gamma_r\mathbf{I}_L$. The algorithms we propose in Section VI for the computation of the decomposition enforce this structure at every iteration by updating the generator vectors of the Toeplitz matrices instead of the matrices themselves.

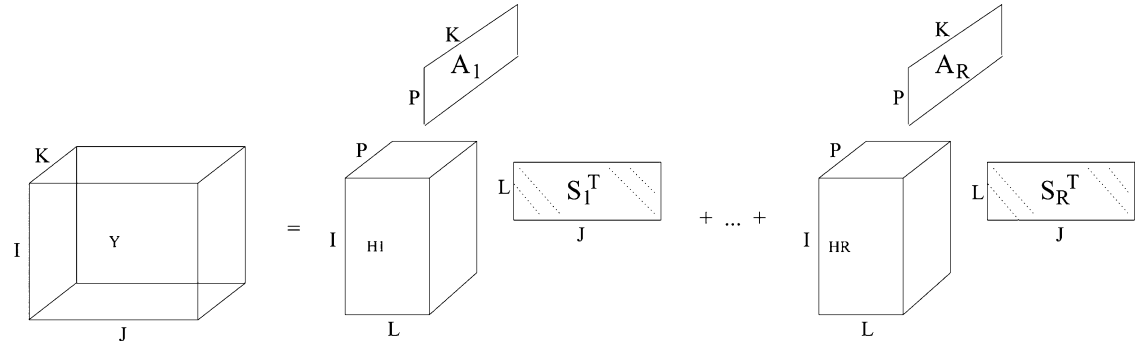


Fig. 3. Schematic representation of the BCM.

B. Uniqueness of the BCM

If the BCM (9) is unique (up to the indeterminacies mentioned in Section V-A) then its computation allows for the separation of the different users' signals and the estimation of the transmitted sequences. Let us denote by $\mathbf{A}$ and $\mathbf{S}$, the $(K \times RP)$ and $(J \times RL)$ matrices that result from the concatenation of the R blocks $\mathbf{A}_r$ and $\mathbf{S}_r$, respectively. The following theorem has been derived in [20, Theorem 6.1]. We call a tensor generic when its entries can be considered drawn from continuous probability density functions.

Theorem 1: Suppose that $rank(\mathbf{A}) = RP$, $rank(\mathbf{S}) = RL$, $I \geq 3$, and that $\mathcal{H}_r$ are generic, then the BCM decomposition is essentially unique.

The condition in Theorem 1 is similar to the PARAFAC condition studied in [34] (see also Section IV). The main difference is that for $\mathcal{H}_r$ at least three instead of two slices are required. Practically speaking, uniqueness is guaranteed if $I \geq 3$ and if the number of users R is bounded by $min(\lfloor K/P \rfloor, \lfloor J/L \rfloor)$. In some cases that do not satisfy the conditions on the ranks of $\mathbf{A}$ and $\mathbf{S}$, uniqueness may nevertheless be proved in analogy with [20, Ex. 3].

C. Matrix Representation of the BCM

Let us denote by $\mathbf{H}_r^{(PL \times I)}$ the matrix resulting from the row-wise concatenation of the P slices of size $(L \times I)$ of $\mathcal{H}_r$. We then build the matrix $\mathbf{H}$ of size $(RPL \times I)$ by concatenation of all $\mathbf{H}_r^{(PL \times I)}$, $r = 1, \ldots, R$. The decomposition of $\mathcal{Y}$ following the BCM then consists of the estimation of the three matrices $\mathbf{A} \in \mathbb{C}^{K \times RP}$, $\mathbf{S} \in \mathbb{C}^{J \times RL}$, and $\mathbf{H} \in \mathbb{C}^{RPL \times I}$. The matrix $\mathbf{S}$ has a block-Toeplitz structure. We now write each matrix representation of $\mathcal{Y}$ in terms of the unknowns.

Let us consider the tensor $\mathcal{Q}_r = \mathcal{H}_r \bullet_2 \mathbf{S}_r$ of size $(I \times J \times P)$ and its matrix representation $\mathbf{Q}_r^{(JI \times P)}$. Equation (9) can then be rewritten as

$$\mathbf{Y}^{(JI \times K)} = \mathbf{Q} \cdot \mathbf{A}^T \tag{11}$$

where $\mathbf{Q}$ is the $(JI \times RP)$ matrix resulting from the concatenation of the matrices $\mathbf{Q}_r^{(JI \times P)}$. Let us now consider the tensor $\mathcal{G}_r = \mathcal{H}_r \bullet_3 \mathbf{A}_r$ of size $(I \times L \times K)$ and its matrix representation $\mathbf{G}_r^{(IK \times L)}$. Equation (9) can then be rewritten as

$$\mathbf{Y}^{(IK \times J)} = \mathbf{G} \cdot \mathbf{S}^T \tag{12}$$

where $\mathbf{G}$ is the $(IK \times RL)$ matrix resulting from the concatenation of the matrices $\mathbf{G}_r^{(IK \times L)}$. Finally, the third matrix representation of $\mathcal{Y}$ is given by

$$\mathbf{Y}^{(KJ \times I)} = \mathbf{Z} \cdot \mathbf{H} \tag{13}$$

where $\mathbf{Z} = [\mathbf{A}_1 \otimes \mathbf{S}_1, \ldots, \mathbf{A}_R \otimes \mathbf{S}_R]$ is a matrix of size $(KJ \times RPL)$.

D. Known User Signatures

Several blind CDMA receivers proposed in the literature capitalize on the known signature of a user of interest in order to remove multiuser interference (MUI) either in a single path [35] or multipath scenario [36], [37]. A CM criterion [1], [38] can also be included, which leads to constrained optimization techniques [39]–[41]. Even though users' signatures are known by the base station in civil applications, they are not available in applications such as eavesdropping. As for the PARAFAC receiver of [15], the algebraic structure of the BCM is rich enough to solve the problem without this prior knowledge. Our techniques can however be adapted to take the spreading codes into account. It suffices to realize that the tensors $\mathcal{H}_r$ contain the convolution of each user's spreading code with the corresponding channel. If the spreading codes are known, then the model is still multilinear, in the transmitted symbols (matrices $\mathbf{S}_r$), in the antenna gains (matrices $\mathbf{A}_r$), and in the channel coefficients, respectively. This structure can directly be exploited in the ALS algorithm proposed in the next section. The LM algorithm can be adapted as well: if knowledge of the spreading codes is assumed, the parameter vector $\mathbf{p_H}$ in Section VI-B has to be replaced by a vector that contains the channel coefficients. These modifications are left as future research.

VI. Computation of the BCM

In this section, we present two different algorithms to compute the decomposition of the BCM given in Section V, i.e., we want to estimate $\mathcal{H}_r$, $\mathbf{S}_r$ and $\mathbf{A}_r$ for each user, given only $\mathcal{Y}$.

We suppose that R (number of active users), P (maximum number of paths per user) and L (maximum channel length at the symbol rate over all users) are known. Methods for the estimation of the rank R in PARAFAC are available from the literature [42]. In addition, methods for the estimation of the dimensions of the so-called Tucker model [43]–[45] are available from

[46]–[48]. See also [28], [29] for a survey. These methods can possibly be generalized to the estimation of the BCM dimensions. Such a generalization is outside the scope of this paper. In Section VII we give some guidelines on how to estimate L and P. Note that the product RL (supposed to be less than J and KI) can be estimated as the number of significant singular values of the matrix $\mathbf{Y}^{(IK\times J)}$ in (12), provided that the noise level is low enough. If needed, the procedures to be presented in this paper could be repeated for a limited set of parameters and the most plausible value retained.

First, an ALS algorithm is developed. ALS is a well-known technique for the calculation of the PARAFAC decomposition [28] and has been extended to the decomposition of a tensor in Block Terms in [21]. In Section VI-A, we show how the general ALS scheme can be modified so as to take the Toeplitz structure in the second mode into account and thus perform equalization within each contribution.

The second algorithm we propose is of the LM type. In the case of PARAFAC, this method outperforms ALS [49] and is especially useful for "difficult" problems [50]. In Section VI-B, we generalize the LM scheme to decompose the BCM and the Toeplitz structure is also taken into account.

Let $\hat{\mathcal{Y}}$ denote an estimation of the tensor of observations $\mathcal{Y}$ given in (9). The algorithms derived in this section have been designed to minimize the following cost function:

$$\phi = \frac{1}{2}\|\mathcal{Y} - \hat{\mathcal{Y}}\|_F^2. \tag{14}$$

A. ALS Algorithm

The ALS algorithm exploits the multilinearity of the algebraic model to alternate between conditional least-squares updates of the unknowns $\mathbf{A}$, $\mathbf{S}$ and $\mathbf{H}$ in each iteration. Given previous estimates of two unknowns, the update rule of the third unknown follows from the matrix representations of $\mathcal{Y}$ derived in Section V-C:

$$\begin{cases} \hat{\mathbf{S}} = \left(\mathbf{Y}^{(IK\times J)}\right)^T \cdot (\mathbf{G}^T)^\dagger \\ \hat{\mathbf{A}} = \left(\mathbf{Y}^{(JI\times K)}\right)^T \cdot (\mathbf{Q}^T)^\dagger \\ \hat{\mathbf{H}} = \mathbf{Z}^\dagger \cdot \mathbf{Y}^{(KJ\times I)}. \end{cases} \tag{15}$$

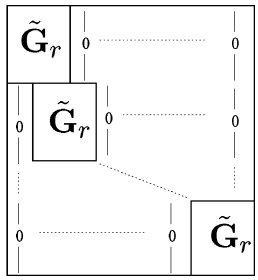
However, the block-Toeplitz structure is not preserved in this update procedure of $\mathbf{S}$. This structure could to some extent be imposed afterwards, as explained in [3], [17], and [18].

In the following, we propose to replace this two-steps update of $\mathbf{S}$ by a single one that consists of the least-squares estimation of the generator vectors of the Toeplitz matrices. Let $\mathbf{s}_r$ be the $(J + L - 1 \times 1)$ generator vector of the Toeplitz matrix $\mathbf{S}_r$ corresponding to the r^{th} user: $\mathbf{s}_r = [s_{2-L,r} \dots s_{1,r} \dots s_{J,r}]^T$. We denote by $\mathbf{s}$ the $R(J + L - 1) \times 1$ vector resulting from the concatenation of all vectors $\mathbf{s}_r$. We will now rewrite (12) in terms of $\mathbf{s}$. We start with the following expression:

$$\mathbf{Y}^{(IK\times J)} = \sum_{r=1}^{R} \mathbf{G}_r \cdot \mathbf{S}_r^T \tag{16}$$

where $\mathbf{G}_r \in \mathbb{C}^{IK\times L}$. Let $\tilde{\mathbf{G}}_r$ be the $(IK \times L)$ matrix obtained by stacking the columns of $\mathbf{G}_r$ in the reverse order, from the last to the first (in Matlab notation we have $\tilde{\mathbf{G}}_r = \mathbf{G}_r(:, L : -1 : 1)$). We then build the following $(JIK \times J+L-1)$ matrix $\mathbf{M}_r$:

$\mathbf{M}_r =$



The matrix $\mathbf{M}_r$ consists of J submatrices of dimension $(IK \times J + L - 1)$. In subsequent submatrices, the matrix $\tilde{\mathbf{G}}_r$ is shifted one position to the right. We now have $\mathrm{vec}(\mathbf{G}_r \cdot \mathbf{S}_r^T) = \mathbf{M}_r \cdot \mathbf{s}_r$. Finally, (16) can be written as

$$Y^{(JIK\times 1)} = \mathbf{M} \cdot \mathbf{s}, \tag{17}$$

where $Y^{(JIK\times 1)} = \mathrm{vec}(\mathbf{Y}^{(IK\times J)})$ and $\mathbf{M}$ is the $JIK \times R(J+L-1)$ matrix obtained by column wise concatenation of all $\mathbf{M}_r$. The least squares update of $\mathbf{s}$ is then given by

$$\hat{\mathbf{s}} = \mathbf{M}^\dagger \cdot Y^{(JIK\times 1)}. \tag{18}$$

As discussed in Section V-B, enforcing the Toeplitz structure in each ALS iteration with this technique reduces the ambiguity on each $\mathbf{S}_r$ to a single complex scalar γ_r.

We now build the following ALS algorithm for the computation of the BCM, where the superscript $\cdot^{(n)}$ denotes an estimation at the nth iteration.

ALS algorithm with Toeplitz structure preserved

Initialize $\mathbf{H}^{(0)}$ and $\mathbf{A}^{(0)}$, set $n = 1$

while $\|\mathcal{Y}^{(n)} - \mathcal{Y}^{(n-1)}\|_F \geq \epsilon$ (e.g. $\epsilon = 10^{-5}$) **do**
1. Build $\mathbf{M}$ from $\mathbf{A}^{(n-1)}$ and $\mathbf{H}^{(n-1)}$
2. Estimate $\mathbf{s}^{(n)}$ from (18) and build $\mathbf{S}^{(n)}$
3. Build $\mathbf{Q}$ from $\mathbf{H}^{(n-1)}$ and $\mathbf{S}^{(n)}$
4. Estimate $\mathbf{A}^{(n)}$ from (15)
5. Build $\mathbf{Z}$ from $\mathbf{A}^{(n)}$ and $\mathbf{S}^{(n)}$
6. Update $\mathbf{H}$ from (15)
7. $n \leftarrow n + 1$

end

Since each substep of the ALS algorithm monotonically decreases the cost function, the algorithm converges in principle to at least a local minimum.

B. LM Algorithm

The ALS algorithm for PARAFAC is known to converge slowly when applied to problems with high collinearity of factors across one or more modes that cause the dataset to be ill-conditioned [50]. Particularly, ill-conditioned factor matrices also occur when the ALS goes through a so-called "swamp," a region with convergence speed almost null after which convergence resumes [51]. In [49], a LM algorithm is proposed for PARAFAC, providing quadratic convergence, and

adapted to ill-conditioned problems. In contrast to ALS, the factors in the three modes are updated at the same time. In this section, we generalize the LM scheme to the decomposition of the BCM. The Toeplitz structure of the matrices $\mathbf{S}_r$ is also taken into account.

1) Derivation of the LM Update Equations: Let $\hat{\mathcal{Y}}$ be an estimation of the tensor of observations $\mathcal{Y}$. Each element of this tensor can be written as

$$\hat{y}_{ijk} = \sum_{r=1}^{R}\sum_{l=1}^{L}\sum_{p=1}^{P} \hat{\mathcal{H}}_r(i,l,p)\hat{\mathbf{S}}_r(j,l)\hat{\mathbf{A}}_r(k,p), \tag{19}$$

where $\hat{\mathbf{A}}_r$, $\hat{\mathbf{S}}_r$ and $\hat{\mathcal{H}}_r$ denote an estimation of $\mathbf{A}_r$, $\mathbf{S}_r$ and $\mathcal{H}_r$, respectively.

Let us stack all the unknowns in a vector $\mathbf{p}$ that contains the elements of the R matrices $\hat{\mathbf{A}}_r$, the R tensors $\hat{\mathcal{H}}_r$ and the R generator vectors of the Toeplitz matrices $\hat{\mathbf{S}}_r$. This vector is of length $F = R(PK+ILP+(J+L-1))$, where F represents the number of unknowns. If we denote by $\mathbf{p}_{\hat{\mathbf{A}}}$ and $\mathbf{p}_{\hat{\mathbf{H}}}$ the vectors of size $(RPK \times 1)$ and $(RILP \times 1)$ such that $\mathbf{p}_{\hat{\mathbf{A}}} = \mathrm{vec}(\hat{\mathbf{A}}^T)$ and $\mathbf{p}_{\hat{\mathbf{H}}} = \mathrm{vec}(\hat{\mathbf{H}})$, then $\mathbf{p}$ is written as

$$\mathbf{p} = \left[\mathbf{p}_{\hat{\mathbf{A}}}^T, \mathbf{p}_{\hat{\mathbf{H}}}^T, \hat{\mathbf{s}}^T\right]^T.$$

We denote by $\hat{Y}^{(KJI\times 1)}$ and $Y^{(KJI\times 1)}$ the vectors of size $(KJI \times 1)$ that contain the entries of $\hat{\mathcal{Y}}$ (built from $\mathbf{p}$) and $\mathcal{Y}$, respectively, stacked such that the left index $(k = 1, \ldots, K)$ is the slowest and the right index $(i = 1, \ldots, I)$ the fastest.

The scalars $\hat{y}_m(\mathbf{p})$ and y_m represent the mth element of $\hat{Y}^{(KJI\times 1)}$ and $Y^{(KJI\times 1)}$, respectively, where $m = (k-1)JI + (j-1)I + i$ is a superindex that combines i, j and k, with $m = 1, \ldots, M = KJI$.

The problem now consists of the minimization of the following cost function with respect to the vector $\mathbf{p}$:

$$\phi(\mathbf{p}) = \frac{1}{2}\sum_{m=1}^{M} |\hat{y}_m(\mathbf{p}) - y_m|^2 = \frac{1}{2}\mathbf{r}(\mathbf{p})^H\mathbf{r}(\mathbf{p}), \tag{20}$$

where $\mathbf{r}(\mathbf{p}) = \hat{Y}^{(KJI\times 1)} - Y^{(KJI\times 1)}$ is the vector of residuals.

Let the $(M \times F)$ matrix $\mathbf{J}$ be the Jacobian matrix of $\mathbf{r}$ with respect to $\mathbf{p}$. Further denote by $\mathbf{g} \in \mathbb{C}^{F\times 1}$ the gradient of ϕ with respect to $\mathbf{p}$

$$\mathbf{g} \stackrel{\text{def}}{=} \phi'(\mathbf{p}) = \mathbf{J}^H\mathbf{r}. \tag{21}$$

One solution to minimize ϕ has been proposed by Levenberg and Marquardt [52] and is also known as "damped Gauss-Newton Method." It consists of updating the step $\Delta\mathbf{p}$ from the following modified normal equations:

$$(\mathbf{J}^H\mathbf{J} + \lambda\mathbf{I}_F)\Delta\mathbf{p} = -\mathbf{g}. \tag{22}$$

The damping parameter $\lambda > 0$ makes the matrix $(\mathbf{J}^H\mathbf{J} + \lambda\mathbf{I}_F)$ nonsingular and positive definite which ensures that $\Delta\mathbf{p}$ is a descent direction. This parameter has several effects:

- For large values of λ, (22) gives $\Delta\mathbf{p} \simeq -(1/\lambda)\mathbf{g}$, i.e., a short step in the steepest descent direction.
- For small values of λ, (22) reduces to a Gauss-Newton update, with quadratic convergence.

The value of λ is adapted after each iteration in a way that is thoroughly described in [53]. The LM algorithm for the computation of the BCM is summarized as follows.

LM algorithm with Toeplitz structure preserved

Initialize $\mathbf{p}$, set $n = 1$

while $\|\mathcal{Y}^{(n)} - \mathcal{Y}^{(n-1)}\|_F \geq \epsilon$ (e.g. $\epsilon = 10^{-5}$) **do**

1. Calculate $\mathbf{J}^H\mathbf{J}$ and $\mathbf{g}$
2. Solve $(\mathbf{J}^H\mathbf{J} + \lambda\mathbf{I})\Delta\mathbf{p} = -\mathbf{g}$ to find $\Delta\mathbf{p}$
3. Update $\mathbf{p}$: $\mathbf{p}^{(n+1)} = \mathbf{p}^{(n)} + \Delta\mathbf{p}$
4. Update λ as in [53].
5. $n \leftarrow n + 1$

end

2) Calculation of the Jacobian and Gradient: The LM method has a high convergence speed, but the calculation of the Jacobian and gradient in every iteration may become time consuming when the size of the data increases. In [49], the specific structure of the Jacobian and gradient for PARAFAC are exploited so that they can be computed in a relatively efficient way. We generalize these results to the BCM. From Section VI-A, given the estimated factors $\hat{\mathbf{A}}$, $\hat{\mathbf{s}}$ and $\hat{\mathbf{H}}$, we can write the estimated tensor model $\hat{\mathcal{Y}}$ under the following matrix and vector forms:

$$\begin{cases} \hat{\mathbf{Y}}^{(JI\times K)} = \mathbf{Q}\cdot\hat{\mathbf{A}}^T \\ \hat{Y}^{(JIK\times 1)} = \mathbf{M}\cdot\hat{\mathbf{s}} \\ \hat{\mathbf{Y}}^{(KJ\times I)} = \mathbf{Z}\cdot\hat{\mathbf{H}} \end{cases} \Leftrightarrow \begin{cases} \hat{Y}^{(KJI\times 1)} = (\mathbf{I}_K \otimes \mathbf{Q})\cdot\mathbf{p}_{\hat{\mathbf{A}}} \\ \hat{Y}^{(JIK\times 1)} = \mathbf{M}\cdot\hat{\mathbf{s}} \\ \hat{Y}^{(IKJ\times 1)} = (\mathbf{I}_I \otimes \mathbf{Z})\cdot\mathbf{p}_{\hat{\mathbf{H}}} \end{cases} \tag{23}$$

where we recall from Section V-C that the $(JI \times RP)$ matrix $\mathbf{Q}$ is built from $\hat{\mathbf{s}}$ and $\hat{\mathbf{H}}$ only, the $(JIK \times R(J+L-1))$ matrix $\mathbf{M}$ is from $\hat{\mathbf{A}}$ and $\hat{\mathbf{H}}$ and the $(KJ \times RPL)$ matrix $\mathbf{Z}$ is from $\hat{\mathbf{A}}$ and $\hat{\mathbf{s}}$.

From the partitioned structure of $\mathbf{p}$, we can write $\mathbf{J} = [\mathbf{J}_\mathbf{A}|\mathbf{J}_\mathbf{s}|\mathbf{J}_\mathbf{H}]$ where $\mathbf{J}$, $\mathbf{J}_\mathbf{A}$, $\mathbf{J}_\mathbf{s}$ and $\mathbf{J}_\mathbf{H}$ stand for $\mathbf{J}(\mathbf{p})$, $\mathbf{J}(\mathbf{p}_{\hat{\mathbf{A}}})$, $\mathbf{J}(\hat{\mathbf{s}})$, and $\mathbf{J}(\mathbf{p}_{\hat{\mathbf{H}}})$, respectively. Thus, the $(KJI \times KRP)$ matrix $\mathbf{J}_\mathbf{A}$ is directly given by (23)

$$\mathbf{J}_\mathbf{A} = \mathbf{I}_K \otimes \mathbf{Q}. \tag{24}$$

In (23b) and (23c), the order the entries are stacked in $\hat{Y}$ is different. The matrix $\mathbf{J}_\mathbf{s}$ is calculated by remapping $\mathbf{M}$ from $JIK \times R(J+L-1))$ to $(KJI \times R(J+L-1))$, i.e.

$$\mathbf{J}_\mathbf{s} = \mathbf{W}_1 \cdot \mathbf{M}. \tag{25}$$

The permutation matrix $\mathbf{W}_1 \in \mathbb{R}^{KJI\times JIK}$ is defined as follows:

$$\mathbf{W}_1 = \left[\mathbf{I}_K \otimes \mathbf{e}_{JI}^{(1)}, \mathbf{I}_K \otimes \mathbf{e}_{JI}^{(2)}, \cdots, \mathbf{I}_K \otimes \mathbf{e}_{JI}^{(JI)}\right] \tag{26}$$

where $\mathbf{e}_{JI}^{(f)}$, $f = 1 \ldots JI$, is the f^{th} column vector of the identity matrix $\mathbf{I}_{JI}$. In the same way, $\mathbf{J}_\mathbf{H}$ is calculated by remapping $(\mathbf{I}_I \otimes \mathbf{Z})$ from $(IKJ \times IRLP)$ to $(KJI \times IRLP)$

$$\mathbf{J}_\mathbf{H} = \mathbf{W}_2 \cdot (\mathbf{I}_I \otimes \mathbf{Z})$$

where the permutation matrix $\mathbf{W}_2 \in \mathbb{C}^{KJI \times IKJ}$ is defined by

$$\mathbf{W}_2 = \left[\mathbf{I}_{KJ} \otimes \mathbf{e}_I^{(1)}, \mathbf{I}_{KJ} \otimes \mathbf{e}_I^{(2)}, \cdots, \mathbf{I}_{KJ} \otimes \mathbf{e}_I^{(I)}\right]. \quad (27)$$

We can then directly build the blocks of $\mathbf{J}^H\mathbf{J}$ in (22)

$$\mathbf{J}^H\mathbf{J} = \begin{bmatrix} \mathbf{J}_{\mathbf{A}}^H \cdot \mathbf{J}_{\mathbf{A}} & \mathbf{J}_{\mathbf{A}}^H \cdot \mathbf{J}_{\mathbf{s}} & \mathbf{J}_{\mathbf{A}}^H \cdot \mathbf{J}_{\mathbf{H}} \\ \mathbf{J}_{\mathbf{s}}^H \cdot \mathbf{J}_{\mathbf{A}} & \mathbf{J}_{\mathbf{s}}^H \cdot \mathbf{J}_{\mathbf{s}} & \mathbf{J}_{\mathbf{s}}^H \cdot \mathbf{J}_{\mathbf{H}} \\ \mathbf{J}_{\mathbf{H}}^H \cdot \mathbf{J}_{\mathbf{A}} & \mathbf{J}_{\mathbf{H}}^H \cdot \mathbf{J}_{\mathbf{s}} & \mathbf{J}_{\mathbf{H}}^H \cdot \mathbf{J}_{\mathbf{H}} \end{bmatrix}. \quad (28)$$

In the computation of the three diagonal blocks one can use properties of the Kronecker product. We obtain

$$\begin{cases} \mathbf{J}_{\mathbf{A}}^H \cdot \mathbf{J}_{\mathbf{A}} = \mathbf{I}_K \otimes (\mathbf{Q}^H \cdot \mathbf{Q}) \\ \mathbf{J}_{\mathbf{H}}^H \cdot \mathbf{J}_{\mathbf{H}} = \mathbf{I}_I \otimes (\mathbf{Z}^H \cdot \mathbf{Z}) \\ \mathbf{J}_{\mathbf{s}}^H \cdot \mathbf{J}_{\mathbf{s}} = \mathbf{M}^H \cdot \mathbf{M} \end{cases}$$

while the off-diagonal blocks can be written as

$$\begin{cases} \mathbf{J}_{\mathbf{A}}^H \cdot \mathbf{J}_{\mathbf{s}} = (\mathbf{I}_K \otimes \mathbf{Q})^H \cdot \mathbf{W}_1 \cdot \mathbf{M} \\ \mathbf{J}_{\mathbf{s}}^H \cdot \mathbf{J}_{\mathbf{H}} = \mathbf{M}^H \cdot \mathbf{W}_1^T \cdot \mathbf{W}_2 \cdot (\mathbf{I}_I \otimes \mathbf{Z}) \\ \mathbf{J}_{\mathbf{A}}^H \cdot \mathbf{J}_{\mathbf{H}} = (\mathbf{I}_K \otimes \mathbf{Q})^H \cdot \mathbf{W}_2 \cdot (\mathbf{I}_I \otimes \mathbf{Z}) \end{cases}.$$

The three remaining off-diagonal blocks are then given by Hermitian symmetry. In order to save significant storage space, the sparsity can also be used. We now explain how the gradient of ϕ in (21) can be calculated. Due to the structure of $\mathbf{p}$, we can write $\mathbf{g}$ as the concatenation of the following three gradients:

$$\mathbf{g} = \begin{bmatrix} \phi'(\mathbf{p}_{\hat{\mathbf{A}}}) \\ \phi'(\hat{\mathbf{s}}) \\ \phi'(\mathbf{p}_{\hat{\mathbf{H}}}) \end{bmatrix} = \begin{bmatrix} \mathbf{g}_{\mathbf{A}} \\ \mathbf{g}_{\mathbf{s}} \\ \mathbf{g}_{\mathbf{H}} \end{bmatrix} \quad (29)$$

where $\mathbf{g}_{\mathbf{A}}$, $\mathbf{g}_{\mathbf{s}}$, and $\mathbf{g}_{\mathbf{H}}$ have length KRP, $R(J+L-1)$, and $IRPL$, respectively. From (23), we can rewrite the loss function defined in (14) and (20) under the three equivalent following forms:

$$\begin{cases} \phi = \frac{1}{2}\left\|Y^{(KJI\times 1)} - (\mathbf{I}_K \otimes \mathbf{Q}) \cdot \mathbf{p}_{\hat{\mathbf{A}}}\right\|^2 \\ \phi = \frac{1}{2}\left\|Y^{(JIK\times 1)} - \mathbf{M} \cdot \hat{\mathbf{s}}\right\|^2 \\ \phi = \frac{1}{2}\left\|Y^{(IKJ\times 1)} - (\mathbf{I}_I \otimes \mathbf{Z}) \cdot \mathbf{p}_{\hat{\mathbf{H}}}\right\|^2 \end{cases} \quad (30)$$

which implies

$$\begin{cases} \mathbf{g}_{\mathbf{A}} = \left(\mathbf{I}_K \otimes (\mathbf{Q}^H \cdot \mathbf{Q})\right) \cdot \mathbf{p}_{\hat{\mathbf{A}}} - (\mathbf{I}_K \otimes \mathbf{Q})^H \cdot Y^{(KJI\times 1)} \\ \mathbf{g}_{\mathbf{s}} = (\mathbf{M}^H \cdot \mathbf{M}) \cdot \hat{\mathbf{s}} - \mathbf{M}^H \cdot Y^{(JIK\times 1)} \\ \mathbf{g}_{\mathbf{H}} = \left(\mathbf{I}_I \otimes (\mathbf{Z}^H \cdot \mathbf{Z})\right) \cdot \mathbf{p}_{\hat{\mathbf{H}}} - (\mathbf{I}_I \otimes \mathbf{Z})^H \cdot Y^{(IKJ\times 1)}. \end{cases}$$

Finally, since λ is updated such that the matrix $(\mathbf{J}^H\mathbf{J} + \lambda\mathbf{I}_F)$ is nonsingular and positive-definite, (22) can be solved by Cholesky decomposition and backsubstitution, an efficient way to solve this set of normal equations.

VII. Simulation Results

In this section, we illustrate and compare the performance of the ALS and LM algorithms. Consider the matrix $\mathbf{S}_t \in \mathbb{C}^{J+L-1\times R}$ that holds the R generator vectors of the Toeplitz matrices $\mathbf{S}_r$. We denote by $\hat{\mathbf{S}}_t$ an estimate of this matrix. In case of perfect estimation, these two matrices are equal, up to permutation and scaling of the columns. For the purpose of performance evaluation in presence of noise, the permutation ambiguity is first solved by using a greedy least squares column matching algorithm [15]. Then, each column of the reordered matrix $\hat{\mathbf{S}}_t$ has to be divided by a scaling factor γ_r, as mentioned in Section V. These scaling factors are taken as the diagonal elements of the matrix $\mathbf{S}_t^\dagger \cdot \hat{\mathbf{S}}_t$. Before calculation of the bit error rate (BER), the elements of the reordered and rescaled matrix $\hat{\mathbf{S}}_t$ are projected to the closest points of the finite symbol alphabet.

The results of the first two experiments of this section have been obtained with spreading codes of length $I = 16$, short frames of $J = 30$ QPSK-symbols, $K = 4$ antennas, $L = 3$ interfering symbols, $P = 2$ major paths per user and $R = 5$ users. The antenna gains, the spreading sequences and the channel fading coefficients are drawn from an independent identically distributed (i.i.d.) Gaussian generator with zero mean and unit variance. Since $K < RP$, the rank of $\mathbf{A}$ is at most K so that Theorem 1 can not directly be applied. In this particular case, uniqueness of the BCM can nevertheless be proved in analogy with [20, Ex. 3].

In a first experiment (Fig. 4), we illustrate the impact of ill-conditioned noise-free data on the convergence speed. We have fixed the tensors $\mathcal{H}_r$ and the matrices $\mathbf{S}_r$ and for each value of the set $\kappa(\mathbf{A}) = \{1, 5, 10, 30, 60, 100, 200\}$, we test the ALS and LM algorithms with 10 different random initializations. The condition number is imposed from an SVD of a randomly drawn matrix $\mathbf{A}$, $\mathbf{A} = \mathbf{U} \cdot \Sigma \cdot \mathbf{V}^H$, after which $\mathbf{U}$ and $\mathbf{V}$ are kept fixed while $\mathbf{\Sigma}$ is changed so as to enforce the desired value of $\kappa(\mathbf{A})$. Since the model is noise-free, we have that $\phi = 0$ if the global minimum is reached. We select the best initialization as the one that leads to the global minimum (the threshold has been fixed to $\phi < 10^{-6}$) with the fewest iterations. Fig. 4(a) shows the number of iterations N needed to converge as a function of the values of $\kappa(\mathbf{A})$. The ALS algorithm is very sensitive to the value of $\kappa(\mathbf{A})$ since N varies from 80 to 10^5 when $\kappa(\mathbf{A})$ takes values from 1 to 200. Under the same conditions, the LM algorithm requires only 10 to 40 iterations. This behavior can be explained by Fig. 4(b), which shows the evolution of ϕ with respect to the iteration index for $\kappa(\mathbf{A}) = \{1, 10, 30, 200\}$. It turns out that the convergence speed of the ALS algorithm drastically decreases for increasing values of $\kappa(\mathbf{A})$. After several small steps in the steepest direction, i.e., large values of λ in (22), the convergence speed of the LM algorithm always increases, which is the result of quadratic convergence of Gauss-Newton steps.

In a second experiment (Fig. 5) we assume the presence of additive white Gaussian noise (AWGN) so that the observed tensor is given by $\mathcal{Y}_{\text{obs}} = \mathcal{Y} + \mathcal{N} = \sum_{r=1}^R \mathcal{Y}_r + \mathcal{N}$, where $\mathcal{Y}$ is the tensor that contains the noise-free data resulting from the sum of

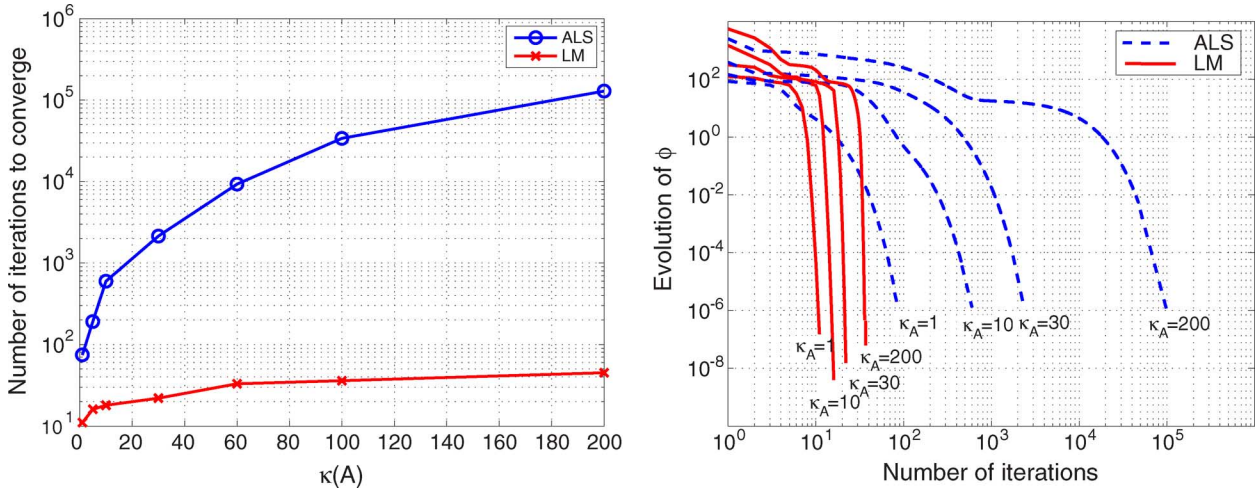


Fig. 4. Impact of $\kappa(\mathbf{A})$ on the performance of ALS and LM algorithms for the decomposition of the noise-free tensor $\mathcal{Y}$. (a) Number of iterations to converge with respect to $\kappa(\mathbf{A})$. (b) Evolution of ϕ with respect to iteration index.

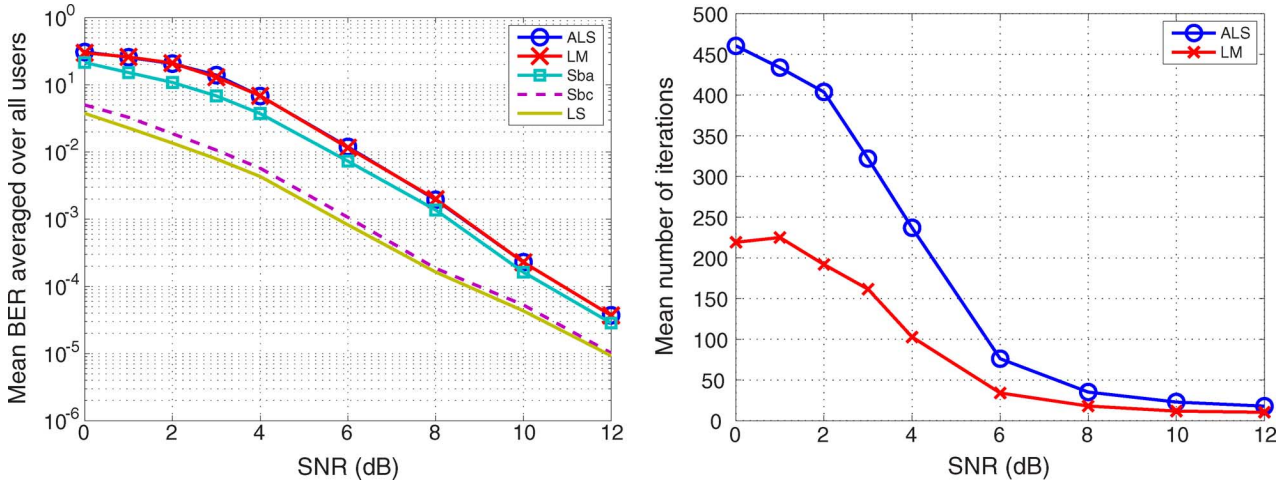


Fig. 5. Average performance of the BCM-based blind receiver in presence of AWGN, by means of ALS or LM. (a) Mean BER versus SNR. (b) Mean number of iterations versus SNR.

R contributions $\mathcal{Y}_r$ and $\mathcal{N}$ contains noise with variable variance. The signal-to-noise ratio (SNR) at the input of the multiuser receiver is defined as

$$\mathrm{SNR} = 10\log_{10}\left(\frac{\|\mathcal{Y}\|_F^2}{\|\mathcal{N}\|_F^2}\right)\ [\mathrm{dB}].$$

We illustrate the performance of the BCM blind receiver based either on ALS or LM by means of 1000 Monte Carlo runs. For each run, the unknown matrices are redrawn and the decomposition of the BCM is calculated from 10 different random starting points. The best starting point is then selected as the one that leads to the smallest value of ϕ. The stop criterion is $\|\mathcal{Y}^{(n)} - \mathcal{Y}^{(n-1)}\|_F < 10^{-7}$ and no control on $\kappa(\mathbf{A})$ is assumed. In Fig. 5(a) the BER is averaged over all users. We compare to the performance of the nonblind least-squares (LS) receiver of which the performance is often used as a benchmark for blind algorithms. In contrast to our algorithms, the LS receiver assumes knowledge of the channel coefficients, spreading codes and antenna array response, i.e., matrices $\mathbf{A}$ and $\mathbf{H}$ are known. The LS solution for the symbol estimates is then given by (18), where perfect knowledge of $\mathbf{M}$ is assumed. We also plot the performance of two semiblind techniques: either $\mathcal{H}_r$ known (Sbc curve) or $\mathbf{A}_r$ known (Sba curve). The ALS and LM algorithms give the same BER curves which means that, on the average, they converge to the same point. Moreover, the performance of the blind receiver based on BCM is close to that of the LS receiver (the gap between the two curves is 2 dB for $\mathrm{BER} = 10^{-4}$). Note that assuming the antenna array response known ($KRP = 40$ known scalars out of 680 unknowns) does not affect the estimation of the remaining unknowns much. In fact, there is enough structure in the BCM itself so that knowledge of the factors in one dimension does not significantly improve the estimation of the factors in the other dimensions. This fact was noticed in [54] for the PARAFAC model. Of course, assuming the tensors $\mathcal{H}_r$ known has more impact on the performance ($RILP = 480$ known scalars out of 680). In Fig. 5, we compare the mean number of iterations required by ALS

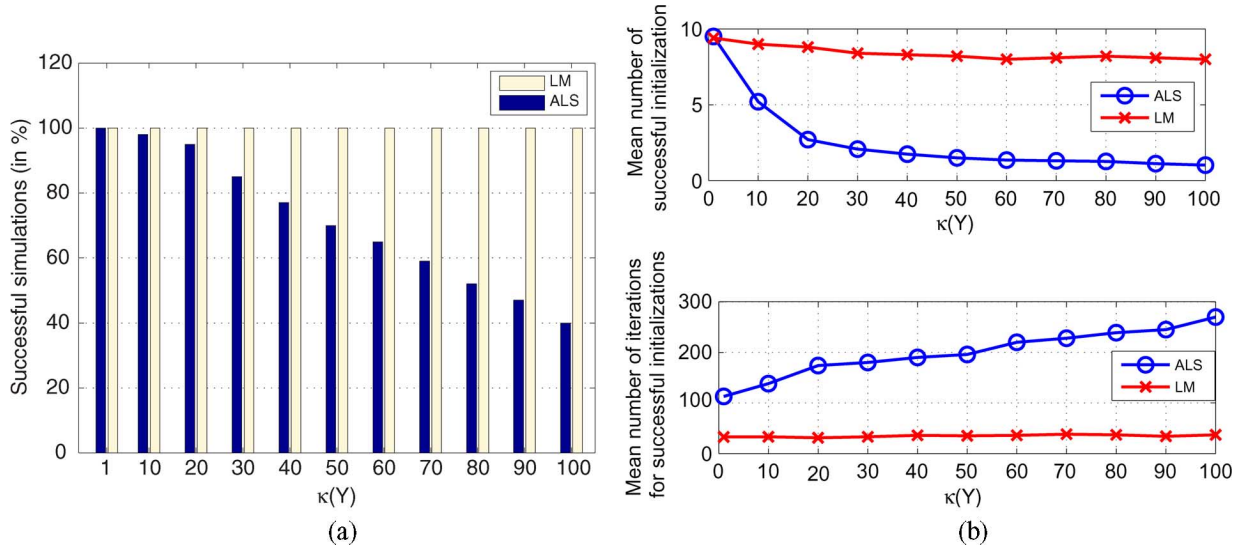


Fig. 6. Impact of $\kappa(\mathcal{Y})$ on the performance of ALS and LM algorithms for the decomposition of the noise-free tensor $\mathcal{Y}$ of (31). (a) Percentage of successful simulations versus $\kappa(\mathcal{Y})$. (b) Mean number of successful initializations among successful simulations, and mean number of iterations for these initializations.

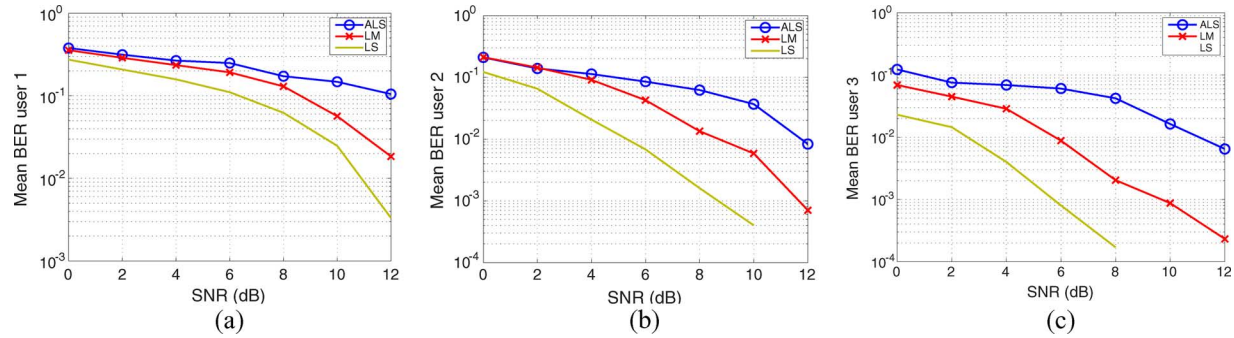


Fig. 7. Near-far effect on the performance of ALS and LM algorithms with $\kappa(\mathcal{Y}) = 10$, in presence of AWGN. (a) BER of user 1 ($\alpha_1 = 1$). (b) BER of user 2 ($\alpha_2 = 5.5$). (c) BER of user 3 ($\alpha_3 = 10$).

and LM algorithms for the 1000 runs. It is clear that the LM algorithm requires fewer iterations than ALS. For instance, the relative gain in number of iterations is 60% for $\mathrm{SNR} = 4$ dB. Although one LM iteration is (reasonably) more costly than one ALS iteration, LM is to be preferred for the dimensions in this test (the gain in time we measured for $\mathrm{SNR} = 4$ dB was 40%).

In a third experiment (Fig. 6), we illustrate the impact of the near-far effect on the performance of the ALS and LM algorithms for the separation of noise-free data. For this experiment 10 different initializations are redrawn for each of the 1000 runs. The parameters have been chosen as follows: codes of length $I = 6$, $J = 30$ QPSK symbols, $K = 4$ antennas, $L = 2$ interfering symbols, $P = 2$ paths, and $R = 3$ users. The stop criterion is the same as in the previous experiment. The observed tensor is generated as follows:

$$\mathcal{Y} = \sum_{r=1}^{3} \alpha_r \frac{\mathcal{Y}_r}{||\mathcal{Y}_r||_F} \tag{31}$$

where the weight α_r is used to control the power of the r^{th} user's signal at the input of the multiuser receiver. We denote by $\kappa(\mathcal{Y})$ the ratio $\max(\alpha_r)/\min(\alpha_r)$. In all the following results,

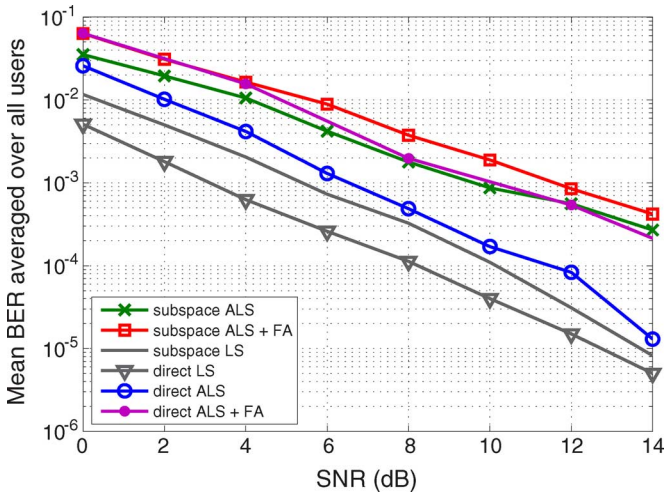


Fig. 8. Impact of Toeplitz structure preservation and FA stage.

the set of amplitudes used to weight the three users' contributions is $\{\alpha_1 = 1, \alpha_2 = (\alpha_1 + \alpha_3)/2, \alpha_3 = \kappa(\mathcal{Y})\}$. In Fig. 6(a), we calculate the percentage of successful simulations, i.e., simulations for which at least one initialization leads to the global

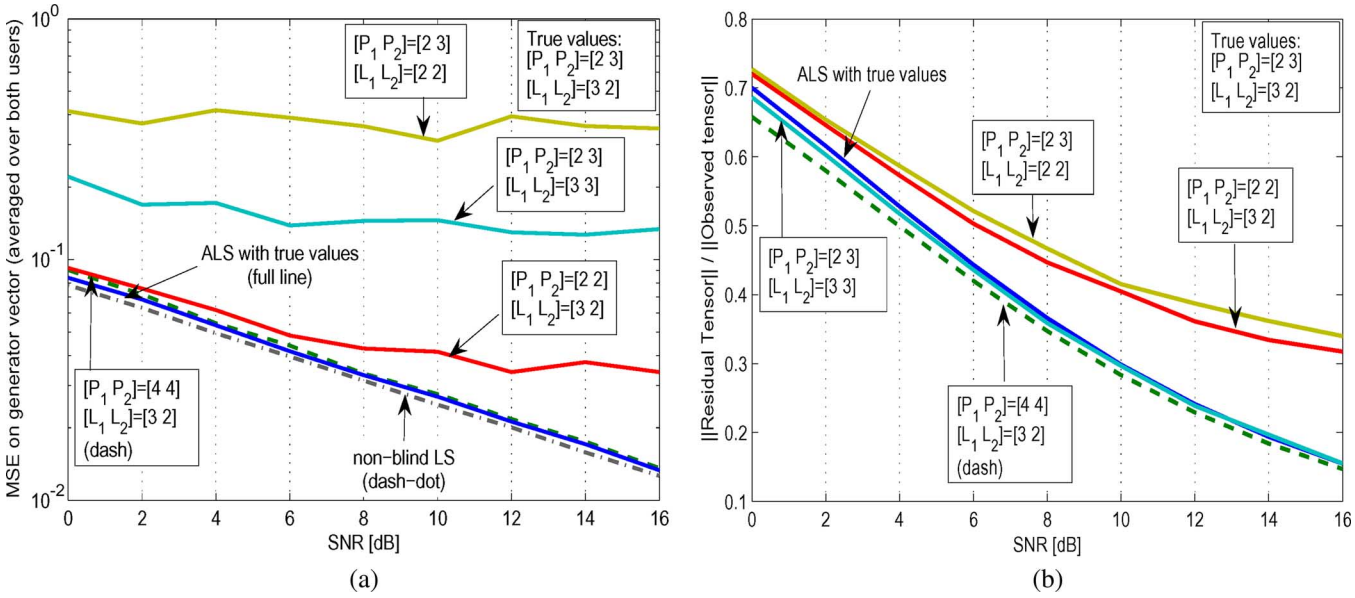


Fig. 9. Impact of over- and underestimation of P_r and L_r. (a) MSE of generator vectors versus SNR. (b) MSE of residual tensor.

minimum, where the $\mathrm{BER} = 0$ for each user. In Fig. 6(b), we have only selected the successful simulations. We calculate the average number of successful initializations for these simulations (top figure) and the average number of iterations for these successful initializations (bottom figure). We conclude that the LM algorithm is less sensitive to the value of $\kappa(\mathcal{Y})$ than ALS. Similarly to Fig. 4(b), the poor performance of ALS is due to swamps, of which occurrence and length are linked to the value of $\kappa(\mathcal{Y})$.

In a fourth experiment (Fig. 7), we illustrate the near-far effect in presence of AWGN. We use the same values of parameters as in the previous experiment and the same conditions for Monte Carlo runs as in the second experiment. The evolution of the mean value of BER for user 1 ($\alpha_1 = 1$), user 2 ($\alpha_2 = 5.5$) and user 3 ($\alpha_3 = 10$) is illustrated by Fig. 7(a)-(c), respectively. Compared to the performance of ALS, the performance of the LM algorithm is closer to that of the LS estimator for all three users, which is due to the higher probability of ALS to stop in a swamp. Under the same simulation conditions with $\alpha_1 = \alpha_2 = \alpha_3 = 1$, we obtained the same curves for ALS and LM.

In a fifth experiment (Fig. 8), we illustrate the impact of the Toeplitz structure preservation strategy and the FA projection strategy. The tensor of observations is built with $I = 16$, $J = 50$ BPSK symbols, $K = 6$, $L_r = 2$, and $P_r = 3$ for each of the $R = 3$ users, and $\kappa(\mathcal{Y}) = 1$. The methods indexed by "direct" in the legend update the generator vectors of the Toeplitz matrices in one step, as proposed in this paper. The methods indexed by "subspace" estimate these vectors in two steps as follows. First, the matrices $\mathbf{S}_r$ are estimated in the least squares sense while ignoring the Toeplitz structure. Then, the subspace-based method of [3] is used for each $\mathbf{S}_r$ to recover its structure, as mentioned in Section VI-A. Apart from this Toeplitz structure update strategy, we also compare to the performance obtained with the same algorithms when a FA projection step is added within the loop after the update of $\mathbf{s}$, as proposed in [18]. In the FA step, we assume that the first symbol of each user sequence is known and equal to 1. This allows one to eliminate the scaling ambiguity by normalizing each estimated symbol sequence $\hat{\mathbf{s}}_r$ by its first element, after which the remaining symbols are projected to the closest FA point. As a conclusion of this experiment, the Toeplitz structure preservation strategy is more accurate than the two-steps subspace strategy. As in [18], we noticed that convergence is accelerated by the use of FA property within the ALS loop (reduction by a factor 6 in this experiment). However, rescaling each estimated sequence according to a potentially erroneous first symbol may affect the estimation of the other symbols, which explains the degradation of the performance.

In a last experiment (Figs. 9 and 10), we assume that the number of paths P_r and the channel length L_r are user-dependent. We illustrate the impact of under- and overestimation of these parameters. We have considered $R = 2$ users, with $\{L_1, P_1\} = \{3, 2\}$, $\{L_2, P_2\} = \{2, 3\}$. The other parameters are $I = 8$, $J = 50$ QPSK symbols, $K = 8$ and $\kappa(\mathcal{Y}) = 1$. In Fig. 9(a), we illustrate the evolution of the mean square error (MSE) of the generator vector of the symbol matrix as a function of the SNR, for different choices of $\{L_r\}$ and $\{P_r\}$. The performance of the LS estimator is plotted as a benchmark. The figure shows that overestimation of P_r did not significantly affect the performance. On the other hand, underestimation of P_r or L_r did degrade the performance, the reason being that part of the effective signal was considered as noise. Overestimation of L_r degraded the performance too.

Fig. 9(b) shows the evolution of $e = ||\mathcal{Y}_{\mathrm{obs}} - \hat{\mathcal{Y}}||_F / ||\mathcal{Y}_{\mathrm{obs}}||_F$. We see that the fit improves for increasing P_r and L_r, until these parameters reach their true values. After this point, the fit does not significantly improve anymore. This can be used to estimate P_r and L_r: the parameter estimates are increased to the point where the effect on the fit becomes negligible. Moreover, over-

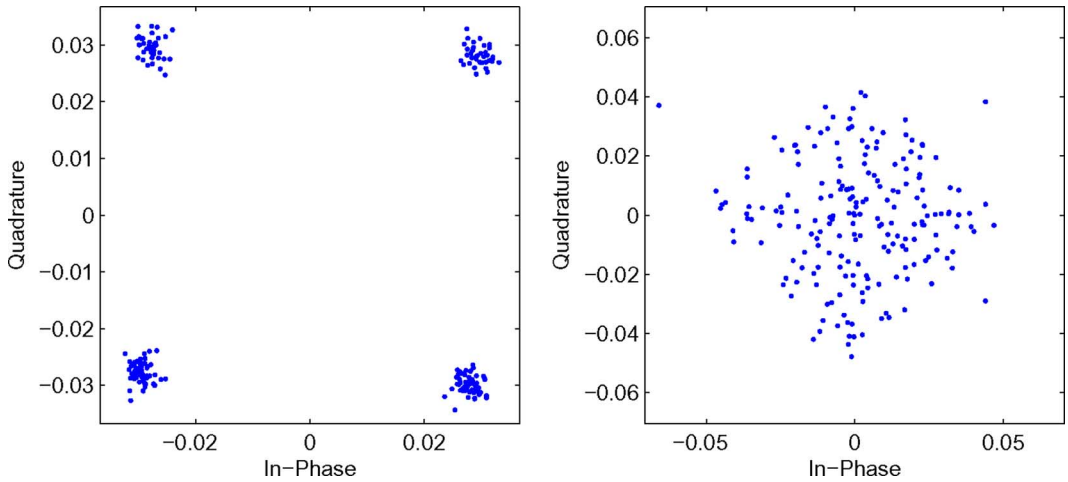


Fig. 10. Impact of overestimation of L_r, SNR $= 10$ dB, $J = 200$. User 1 (left): $P_1 = 2$, $L_1 = 3$, and $\hat{P}_1 = 3$, $\hat{L}_1 = 3$. User 2 (right): $P_2 = 3$, $L_2 = 2$, and $\hat{P}_2 = 3$, $\hat{L}_2 = 3$.

estimation of L_r is indicated by the fact that the source constellation pattern is lost. When the true value L_r is strictly smaller than its estimate $\hat{L}_r$, then $\mathbf{S}_r$ can be very well approximated by $\hat{\mathbf{S}}_r \cdot \mathbf{T}_r$, with $\mathbf{T}_r$ an $(\hat{L}_r \times L_r)$ Toeplitz matrix. In other words, the true generator vector $\mathbf{s}_r$ can be obtained from a linear combination of the columns of $\hat{\mathbf{S}}_r$. This implies in turn that the estimated generator vector $\hat{\mathbf{s}}_r$ does not show the typical constellation pattern anymore. This is illustrated in Fig. 10. The left panel shows the estimated symbols for user 1, with P_1 overestimated and L_1 correctly estimated. The right panel shows the estimated symbols for user 2, with P_2 correctly estimated and L_2 overestimated. Finally, we mention that we have observed that overestimation of L_r and/or P_r slowed down the convergence.

VIII. Conclusion

In this paper, we have shown how Block Component Analysis of a third-order tensor leads to a powerful blind receiver in DS-CDMA systems where convolutive mixtures are received by an array of antennas. The tensor model takes both ISI and multipath propagation aspects into account, which was not the case for the blind PARAFAC receiver in [15]. The *separation* of each user's contribution relies on the uniqueness property of the BCM decomposition. The *equalization* step is performed within each contribution by imposing a Toeplitz structure to the symbol matrix. This method does not require the channel to be stationary over many symbol periods since it is deterministic and, thus, works for very short data sequences.

The BCM approach involves joint estimation of the factors in the three modes, meaning that the method performs blind channel identification, blind symbol estimation and blind DOA estimation for each user. In a semiblind situation where the antenna array geometry is known, the particular structure of the matrices $\mathbf{A}_r$ [10] could be exploited. Block Component Analysis can also be applied to other systems where at least three diversities are available. For instance, the code diversity could be replaced by a temporal oversampling diversity. The computation strategy for the calculation of the BCM decomposition is an important issue. It turns out that an algorithm of the LM type offers faster convergence than an ALS algorithm and is better suited for blind extraction of ill-conditioned data and small-power contributions.

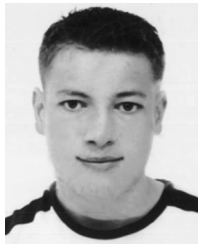
Acknowledgment

The authors wish to thank C. Navasca of the Rochester Institute of Technology, NY, for proofreading an early version of the manuscript.

References

[1] D. N. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Trans. Commun.*, vol. 28, pp. 1867–1875, 1980.
[2] A.-J. van der Veen and A. Paulraj, "An analytical constant modulus algorithm," *IEEE Trans. Signal Process.*, vol. 44, pp. 1136–1155, 1996.
[3] E. Moulines, P. Duhamel, J.-F. Cardoso, and S. Mayrargue, "Subspace methods for the blind identification of multichannel FIR filters," *IEEE Trans. Signal Process.*, vol. 43, pp. 516–525, 1995.
[4] A.-J. van der Veen, S. Talwar, and A. Paulraj, "Blind estimation of multiple digital signals transmitted over FIR channels," *IEEE Signal Process. Lett.*, vol. 2, pp. 99–102, 1995.
[5] A. Swami, G. B. Giannakis, and G. Zhou, "Bibliography on higher-order statistics," *Signal Process.*, vol. 60, no. 1, pp. 65–126, 1997.
[6] A. Touzni, I. Fijalkow, M. G. Larimore, and J. R. Treichler, "A globally convergent approach for blind MIMO adaptive deconvolution," *IEEE Trans. Signal Process.*, vol. 49, no. 6, pp. 1166–1178, 2001.
[7] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. 34, pp. 276–280, 1986.
[8] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, pp. 984–995, 1989.
[9] A.-J. van der Veen, S. Talwar, and A. Paulraj, "A Subspace approach to blind space-time signal processing for wireless communication systems," *IEEE Trans. Signal Process.*, vol. 45, pp. 173–190, 1997.
[10] A.-J. van der Veen, "Algebraic methods for deterministic blind beamforming," *Proc. IEEE*, vol. 86, pp. 1987–2008, 1998.
[11] L. De Lathauwer, "Signal processing based on multilinear algebra," Ph.D. dissertation, Faculty of Eng., K.U., Leuven, Belgium, 1997.
[12] P. Comon, "Tensor decompositions," in *Mathematics in Signal Proc. V*, J. McWhirter and I. Proudler, Eds. New York: Clarendon, 2002, pp. 1–24.
[13] P. Comon, "Independent component analysis, a new concept?," *Signal Process. Elsevier*, vol. 36, no. 3, pp. 287–314, Apr. 1994, Special Issue on Higher-Order Statistics.
[14] N. D. Sidiropoulos, R. Bro, and G. B. Giannakis, "Parallel factor analysis in sensor array processing," *IEEE Trans. Signal Process.*, vol. 48, pp. 2377–2388, 2000.
[15] N. D. Sidiropoulos, G. B. Giannakis, and R. Bro, "Blind PARAFAC receivers for DS-CDMA systems," *IEEE Trans. Signal Process.*, vol. 48, pp. 810–823, 2000.

[16] L. De Lathauwer and A. de Baynast, "Blind deconvolution of DS-CDMA signals by means of decomposition in rank-(1,L,L) terms," *IEEE Trans. Signal Process.*, vol. 56, no. 4, pp. 1562–1571, Apr. 2008.
[17] N. D. Sidiropoulos and G. Z. Dimic, "Blind multiuser detection in W-CDMA systems with large delay spread," *IEEE Signal Process. Lett.*, vol. 8, no. 3, pp. 87–89, Mar. 2001.
[18] 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 Process.*, vol. 87, pp. 337–351, 2007.
[19] L. De Lathauwer, "Decompositions of a higher-order tensor in block terms—Part I: Lemmas for partitioned matrices," *SIAM J. Matrix Anal. Appl. (Special Issue on Tensor Decompositions and Applications)*, vol. 30, no. 3, pp. 1022–1032, 2008.
[20] L. De Lathauwer, "Decompositions of a higher-order tensor in block terms—Part II: Definitions and uniqueness," *SIAM J. Matrix Anal. Appl. (Special Issue on Tensor Decompositions and Applications)*, vol. 30, no. 3, pp. 1033–1066, 2008.
[21] L. De Lathauwer and D. Nion, "Decompositions of a higher-order tensor in block terms—Part III: Alternating least squares algorithms," *SIAM J. Matrix Anal. Appl. (Special Issue on Tensor Decompositions and Applications)*, vol. 30, no. 3, pp. 1067–1083, 2008.
[22] G. B. Giannakis, "Cyclostationary signal analysis," in *Digital Signal Processing Handbook*, V. K. Madisetti and D. Williams, Eds. Boca Raton, FL: CRC, 1998.
[23] D. Nion and L. De Lathauwer, "A block factor analysis based receiver for blind multiuser access in wireless communications," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Toulouse, France, 2006, pp. 825–828.
[24] F. L. Hitchcock, "The expression of a tensor or a polyadic as a sum of products," *J. Math. Phys.*, vol. 6, no. 1, pp. 164–189, 1927.
[25] F. L. Hitchcock, "Multiple invariants and generalized rank of a p-way matrix or tensor," *J. Math. Phys.*, vol. 7, no. 1, pp. 39–79, 1927.
[26] J. D. Carroll and J. Chang, "Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition," *Psychometrika*, vol. 35, no. 3, pp. 283–319, 1970.
[27] R. A. Harshman, "Foundations of the PARAFAC procedure: Model and conditions for an 'explanatory' multimode factor analysis," in *UCLA Working Papers in Phonet.*, 1970, vol. 16, pp. 1–84.
[28] A. Smilde, R. Bro, and P. Geladi, *Multi-Way Analysis. Applications in the Chemical Sciences*. Chichester, U.K.: Wiley, 2004.
[29] P. M. Kroonenberg, *Applied Multiway Data Analysis*. : Wiley Series in Probabil. Statist., 2008.
[30] L. De Lathauwer and J. Castaing, "Tensor-based techniques for the blind separation of DS-CDMA signals," *Signal Process., Special Issue Tensor Signal Process.*, vol. 87, no. 2, pp. 322–336, Feb. 2007.
[31] L. De Lathauwer, "A link between the canonical decomposition in multilinear algebra and simultaneous matrix diagonalization," *SIAM J. Matrix Anal. Appl.*, vol. 28, no. 3, pp. 642–666, 2006.
[32] J. B. Kruskal, "Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics," *Linear Algebra Appl.*, vol. 18, pp. 95–138, 1977.
[33] A. Stegeman and N. D. Sidiropoulos, "On Kruskal's uniqueness condition for the candecomp/parafac decomposition," *Linear Alg. Appl.*, vol. 420, pp. 540–552, 2007.
[34] S. E. Leurgans, R. T. Ross, and R. B. Abel, "A decomposition for three-way arrays," *SIAM J. Matrix Anal. Appl.*, vol. 14, no. 4, pp. 1064–1083, 1993.
[35] M. Honig, U. Madhow, and S. Verdu, "Blind adaptive multiuser detection," *IEEE Trans. Inf. Theory*, vol. 41, pp. 944–960, 1995.
[36] M. K. Tsatsanis and Z. Xu, "Performance analysis of minimum variance CDMA receivers," *IEEE Trans. Signal Process.*, vol. 46, no. 11, pp. 3014–3022, Nov. 1998.
[37] Z. Xu and M. K. Tsatsanis, "Blind adaptive algorithms for minimum variance CDMA receivers," *IEEE Trans. Commun.*, vol. 49, no. 1, pp. 3014–3022, Jan. 2001.
[38] C. Xu, G. Feng, and K. S. Kwak, "A modified constrained constant modulus approach to blind adaptive multiuser detection," *IEEE Trans. Commun.*, vol. 49, no. 9, pp. 1642–1648, 2001.
[39] Z. Xu and P. Liu, "Code-constrained blind detection of CDMA signals in multipath channels," *IEEE Signal Process. Lett.*, vol. 9, no. 12, pp. 389–392, 2002.
[40] R. C. de Lamare and R. Sampaio Neto, "Blind adaptive code-constrained constant modulus algorithms for CDMA interference suppression in multipath channels," *IEEE Comm. Lett.*, vol. 9, no. 4, pp. 334–336, 2005.
[41] P. Daehyun, R. M. Golden, M. Torlak, and E. M. Dowling, "Blind adaptive CDMA processing for smart antennas using the block shanno constant modulus algorithm," *IEEE Trans. Signal Process.*, vol. 54, no. 5, pp. 1956–1959, 2006.
[42] R. Bro and H. A. L. Kiers, "A new efficient method for determining the numbers of components in PARAFAC models," *J. Chemom.*, vol. 17, pp. 274–286, 2003.
[43] L. R. Tucker, "The extension of factor analysis to three-dimensional matrices," in *Contributions to Mathematical Psychology*, R. Holt and Winston, Eds. New York: H. Gulliksen and N. Frederiksen, 1964, pp. 109–127.
[44] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, pp. 279–311, 1966.
[45] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, 2000.
[46] E. Ceulemans and H. A. L. Kiers, "Selecting among three-mode principal component models of different types and complexities: A numerical convex hull based method," *Br. J. Math. Statist. Psychol.*, vol. 59, pp. 133–150, 2006.
[47] N. Renard, S. Bourennane, and J. Blanc-Talon, "Multiway filtering applied on hyperspectral images," in *ACIVS*, 2006, pp. 127–137.
[48] M. E. Timmerman and H. A. L. Kiers, "Three-mode principal components analysis: Choosing the numbers of components and sensitivity to local optima," *Br. J. Math. Statist. Psychol.*, vol. 53, pp. 1–16, 2000.
[49] G. Tomasi and R. Bro, "A comparison of algorithms for fitting the PARAFAC model," *Comp. Stat. Data Anal.*, vol. 50, pp. 1700–1734, 2006.
[50] P. K. Hopke, P. Paatero, H. Jia, R. T. Ross, and R. A. Harshman, "Three-way (PARAFAC) factor analysis: Examination and comparison of alternative computational methods as applied to ill-conditioned data," *Chemom. Intell. Lab. Syst.*, vol. 43, pp. 25–42, 1998.
[51] W. Rayens and W. Mitchell, "Two-factor degeneracies and a stabilization of PARAFAC," *Chemom. Intell. Lab. Syst.*, vol. 38, pp. 173–181, 1997.
[52] D. Marquardt, "An algorithm for least-squares estimation of non-linear parameters," *SIAM J. Appl. Math.*, vol. 11, pp. 431–441, 1963.
[53] K. Madsen, H. B. Nielsen, and O. Tingleff, *Methods for Non-Linear Least Squares Problems*, 2nd ed. Copenhagen: Tech. Univ. Denmark, 2004.
[54] X. Liu and N. D. Sidiropoulos, "Cramer-Rao lower bounds for low-rank decomposition of multidimensional arrays," *IEEE Trans. Signal Process.*, vol. 49, no. 9, pp. 2074–2086, Sep. 2001.



Dimitri Nion was born in Lille, France, on September 6, 1980. He received the electronic engineering degree from ISEN, Lille, France, in 2003, the M.S. degree from Queen Mary University, London, U.K., in 2003, and the Ph.D. degree in signal processing from the University of Cergy-Pontoise, France, in 2007.

He is currently a Postdoctoral Fellow with the Telecommunications Division of the Electronics and Computer Engineering Department, Technical University of Crete, Kounoupidiana Campus, Chania, Crete, Greece. His research interests include linear and multilinear algebra, blind source separation, array processing, optimization, and adaptive signal processing.

Lieven De Lathauwer (M'04–SM'06) was born in Aalst, Belgium, on November 10, 1969. He received the Master's degree in electromechanical engineering and the Ph.D. degree in applied sciences from the Katholieke Universiteit Leuven (K.U. Leuven), Leuven, Belgium, in 1992 and 1997, respectively. His Ph.D. thesis concerned signal processing based on multilinear algebra.

From 2000 to 2007, he was with the Centre National de la Recherche Scientifique (C.N.R.S.), Cergy-Pontoise, France. He is currently with the K.U. Leuven. His research interests include linear and multilinear algebra, statistical signal and array processing, higher-order statistics, independent component analysis, identification, blind identification, and equalization.

Dr. De Lathauwer is an Associate Editor of the *SIAM Journal on Matrix Analysis and Applications*.