MULTILINEAR GENERALIZED SINGULAR VALUE DECOMPOSITION (ML-GSVD) WITH APPLICATION TO COORDINATED BEAMFORMING IN MULTI-USER MIMO SYSTEMS

Liana Khamidullina[⋆]   *André L. F. de Almeida*[†]   *Martin Haardt*[⋆]

[⋆] Communication Research Laboratory, Ilmenau University of Technology, Ilmenau, Germany
[†] Department of Teleinformatics Engineering, Federal University of Ceará, Fortaleza, Brazil

ABSTRACT

In this paper, we propose a new Multilinear Generalized Singular Value Decomposition (ML-GSVD) which allows to jointly factorize a set of matrices with one common dimension. The ML-GSVD is an extension of the Generalized Singular Value Decomposition (GSVD) for more than two matrices. In comparison with other approaches that extend the GSVD, the proposed tensor decomposition preserves the essential properties of the original GSVD, such as orthogonality of the second mode factor matrices. In this work, we introduce two algorithms to compute the ML-GSVD. In addition, we present an application of the ML-GSVD to compute the beamforming matrices for the multi-user MIMO downlink channel with more than two users in wireless communications.

Index Terms— Tensor factorization, Generalized SVD, Multilinear GSVD, Beamforming

1. INTRODUCTION

The Generalized Singular Value Decomposition (GSVD) is a natural generalization of the SVD for two matrices, which is useful in various communication and biomedical applications, such as coordinated beamforming, MIMO relaying, physical layer security, and genomic signal processing [1–6].

In this paper, we present an extension of the GSVD [7], [8] to factorize a three-way tensor. The proposed multilinear generalized singular value decomposition (ML-GSVD) can be used for the joint analysis of a collection of more than two matrices which may have a varying number of columns and the same number of rows.

The authors in [9], [10] have already introduced a multidimensional decomposition to extend the GSVD to the tensor case. But the so-called higher-order GSVD (HO GSVD) in [9] does not preserve the orthogonality of the factor matrices as in the original GSVD. In [10] the authors have presented a Tensor GSVD to jointly decompose two data sets as a coupled decomposition of two tensors. Note that both papers [9] and [10] consider real-valued matrices and a biomedical data analysis. In contrast to the ML-GSVD, the HO GSVD and the Tensor GSVD do not inherit the properties of the original GSVD, such as the concept of common and private subspaces.

Due to the fact that the ML-GSVD provides orthogonal factor matrices for the individual slices, it is a valuable tool for coordinated downlink beamforming in a wireless multi-user MIMO system. More specifically, by applying the ML-GSVD to a set of channel matrices (associated with different users), we are able to identify common subspaces (CSs) to a group of users, as well as private subspaces (PSs) to individual users. Hence, by exploiting the structure of these subspaces, broadcast and multicast transmission can be simultaneously combined on the downlink. In [1], [2] and [3] the SVD-based beamforming is generalized to GSVD-based beamforming for only two users. The authors illustrate how the GSVD can be exploited for coordinated beamforming in a multi-user MIMO system with two users. The use of the ML-GSVD allows us to go further and increase the number of destinations. Depending on the number of transmit and receive antennas (tensor dimensions), the subspace structure of the ML-GSVD distinguishes between common and private subspaces. Common subspaces are used to transmit the same data to several users, while private subspaces allow to send confidential messages to different users simultaneously. Hence, the ML-GSVD allows to handle an arbitrary number of users that is less or equal than the number of transmit antennas on the downlink of a coordinated MIMO beamforming system.

The remainder of the paper is organized as follows. In Section 2, we review the GSVD for two matrices. Then, we introduce the ML-GSVD. Section 3 presents two algorithms for computing the ML-GSVD. In Section 4, we present an application of the ML-GSVD for coordinated beamforming in multi-user MIMO systems. Section 5 presents some numerical results and the paper is concluded in Section 6.

Notation: Matrices and vectors are denoted by upper-case and lower-case bold-faced letters, respectively. Bold faced calligraphic letters denote tensors. The superscripts $\{\cdot\}^{\mathrm{T}}$ and $\{\cdot\}^{\mathrm{H}}$ denote the transpose and Hermitian transpose, respectively, whereas $\mathrm{diag}\{\cdot\}$ is the operation of constructing a diagonal matrix with diagonal elements being the entries of the input vector, while $\mathrm{bdiag}\{\cdot\}$ is the operation of constructing a block diagonal matrix with the input matrices on the main diagonal. The i-th row and the j-th column of a matrix $\boldsymbol{A} \in \mathbb{C}^{I \times J}$ is represented by $\boldsymbol{A}(i,:) \in \mathbb{C}^J$ and $\boldsymbol{A}(:,j) \in \mathbb{C}^I$, respectively, where $i = 1, \ldots, I$ and $j = 1, \ldots, J$. The Kronecker and Khatri-Rao products are denoted as $\otimes$ and $\diamond$, respectively. Additionally, we denote the higher-order norm of a tensor $\boldsymbol{\mathcal{A}}$ by $\|\boldsymbol{\mathcal{A}}\|_{\mathrm{H}}$ and the norm of a vector $\boldsymbol{a}$ by $\|\boldsymbol{a}\|$. The r-mode unfolding of the tensor $\boldsymbol{\mathcal{A}}$ is denoted as $[\boldsymbol{\mathcal{A}}]_{(r)}$. $\boldsymbol{I}_d$ denotes the $d \times d$ identity matrix. $\{\boldsymbol{A}\}_{\mathcal{R}(\mathcal{Q})}$ and $\{\boldsymbol{A}\}_{\mathcal{C}(\mathcal{Q})}$ denote the sub-matrices consisting of the columns and rows of $\boldsymbol{A}$ with indices in the set $\mathcal{Q} \subseteq \{1, \ldots, I\}$, respectively.

2. MULTILINEAR GENERALIZED SINGULAR VALUE DECOMPOSITION (ML-GSVD)

Before introducing the ML-GSVD, let us first review the GSVD of two matrices proposed in [7] and [8]. Let $\boldsymbol{H}_1 \in \mathbb{C}^{I \times J_1}$ and $\boldsymbol{H}_2 \in$

ICASSP 2020

$\mathbb{C}^{I \times J_2}$ be two matrices having the same number of rows, and an arbitrary number of columns J_1 and J_2. Then, the GSVD of $\boldsymbol{H_1}$ and $\boldsymbol{H_2}$ is defined as

$$
\begin{aligned}
\boldsymbol{H}_1 &= \boldsymbol{A} \cdot \boldsymbol{C}_1 \cdot \boldsymbol{B}_1^{\mathrm{T}}, \\
\boldsymbol{H}_2 &= \boldsymbol{A} \cdot \boldsymbol{C}_2 \cdot \boldsymbol{B}_2^{\mathrm{T}},
\end{aligned} \tag{1}
$$

where $\boldsymbol{B}_1 \in \mathbb{C}^{J_1 \times J_1}$ and $\boldsymbol{B}_2 \in \mathbb{C}^{J_2 \times J_2}$ have orthonormal columns, $\boldsymbol{A} \in \mathbb{C}^{I \times I}$ is nonsingular and common to both matrices. Moreover, $\boldsymbol{C}_1 \in \mathbb{R}^{I \times I}$ and $\boldsymbol{C}_2 \in \mathbb{R}^{I \times I}$ are non-negative diagonal matrices. The ratios of the corresponding entries of $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ are called generalized singular values of $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$. Let $q = \mathrm{rank}([\boldsymbol{H}_1 \ \boldsymbol{H}_2])$, $r = q - \mathrm{rank}(\boldsymbol{H}_2)$ and $s = \mathrm{rank}(\boldsymbol{H}_1) + \mathrm{rank}(\boldsymbol{H}_2) - q$, then $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ have the following structure:

$$
\boldsymbol{C}_1 = \begin{bmatrix} \boldsymbol{I}_r & & \\ & \hat{\boldsymbol{\Sigma}} & \\ & & \boldsymbol{0}_{(m-r-s) \times (t-r-s)} \end{bmatrix}, \tag{2}
$$

$$
\boldsymbol{C}_2 = \begin{bmatrix} \boldsymbol{0}_{(J_2-q+r) \times (r)} & & \\ & \hat{\boldsymbol{\Lambda}} & \\ & & \boldsymbol{I}_{(q-r-s) \times (q-r-s)} \end{bmatrix}, \tag{3}
$$

where $\boldsymbol{I}_r$ and $\boldsymbol{I}_{(q-r-s) \times (q-r-s)}$ are identity matrices, $\boldsymbol{0}_{(m-r-s) \times (t-r-s)}$ and $\boldsymbol{0}_{(J_2-q+r) \times (r)}$ are zero matrices possibly having no rows or no columns, $\hat{\boldsymbol{\Sigma}} = \mathrm{diag}(\sigma_1, \cdots, \sigma_s)$, $\hat{\boldsymbol{\Lambda}} = \mathrm{diag}(\lambda_1, \cdots, \lambda_s)$ are diagonal matrices, such that $0 < \sigma_n < 1, 0 < \lambda_n < 1$, and $\sigma_n^2 + \lambda_n^2 = 1$ for $n \in \{1, \cdots, s\}$. For more details on the GSVD we refer the reader to [7,8].

Next, let us introduce the ML-GSVD for $K \geq 2$ complex-valued matrices. Let us consider a set of K complex-valued matrices $\boldsymbol{H}_k \in \mathbb{C}^{I \times J_k}$ with the same number I of rows and potentially a different number J_k of columns:

$$
\begin{aligned}
\boldsymbol{H}_1 &= \boldsymbol{A} \cdot \boldsymbol{C}_1 \cdot \boldsymbol{B}_1^{\mathrm{T}} \in \mathbb{C}^{I \times J_1}, \\
&\vdots \\
\boldsymbol{H}_K &= \boldsymbol{A} \cdot \boldsymbol{C}_K \cdot \boldsymbol{B}_K^{\mathrm{T}} \in \mathbb{C}^{I \times J_K}.
\end{aligned} \tag{4}
$$

The K matrices can be viewed as slices of the tensor $\boldsymbol{\mathcal{H}} \in \mathbb{C}^{I \times J \times K}$, $J = \max(J_1, \ldots, J_K, I)$ (zeros are added for those elements that are not defined in (4)), which allows us to use tensor based algorithms to compute the matrices $\boldsymbol{A}, \boldsymbol{C}$, and $\boldsymbol{B}_k$. The diagonal elements of $\boldsymbol{C}_k, k \in \{1, \cdots, K\}$, are stacked as rows of $\boldsymbol{C}$. Then, the ML-GSVD of the tensor $\boldsymbol{\mathcal{H}}$ (Fig. 1) can be defined in a slice-wise fashion as

$$
\boldsymbol{H}_k = \boldsymbol{A} \cdot \mathrm{diag}\{\boldsymbol{C}(k,:)\} \cdot \boldsymbol{B}_k^{\mathrm{T}} \in \mathbb{C}^{I \times J_k}, \tag{5}
$$

where $\boldsymbol{A} \in \mathbb{C}^{I \times I}$ is square, nonsingular, and common for all factorizations. The matrix $\boldsymbol{B}_k \in \mathbb{C}^{J_k \times I}$ corresponding to the k-th slice of $\boldsymbol{\mathcal{H}}$ has orthonormal columns. The elements of $\boldsymbol{C} \in \mathbb{R}^{K \times I}$ are non-negative, and the columns of $\boldsymbol{C}$ have unit norm. The values of the first row of $\boldsymbol{C}$ are sorted in descending order, such that the first row of $\boldsymbol{C}$ has the following structure:

$$
\boldsymbol{C}(1,:) = \begin{bmatrix} \boldsymbol{1}_{p_1}^{\mathrm{T}} & \boldsymbol{\sigma}_1^{\mathrm{T}} & \boldsymbol{0}_{p_2+\ldots+p_K}^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{1 \times I}, \tag{6}
$$

where $\boldsymbol{1}_{p_1}$ is a vector of ones, and $\boldsymbol{0}_{p_2+\ldots+p_K}$ is a vector of zeros, which might have no entries. The values of $\boldsymbol{\sigma}_k \in \mathbb{R}^{c_k}$ are in the range $(0,1)$, p_k and c_k are the dimensions of the so-called *common* and *private* subspaces, respectively. The remaining rows of $\boldsymbol{C}$ are
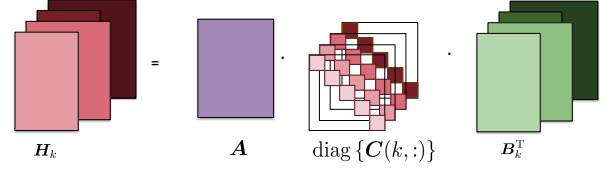

Fig. 1: 3-D Generalized Singular Value Decomposition.

sorted according to the first row. Whenever there are ambiguities, we sort the elements of the second row in descending order. Then we turn to the third row, whose elements are sorted according to the first and second rows. Whenever there are ambiguities, we sort the elements of the third row in descending order. After that, we switch to the fourth row, and so on.

Depending on the number of rows and columns in $\boldsymbol{H}$, we have the following cases:

1. $I \leq J_k$ for $\forall k$. The matrix $\boldsymbol{C}$ has the following structure:

$$
\boldsymbol{C}(k,:) = \begin{bmatrix} \boldsymbol{\sigma}_k^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{J_k}, \tag{7}
$$

where $\boldsymbol{\sigma}_k^{\mathrm{T}} = [\sigma_{k,1}, \cdots, \sigma_{k,I}] \in \mathbb{R}^{1 \times I}$, such that $1 > \sigma_{k,i} > 0$ for $i \in \{1, \cdots, I\}$. In this case, we have only a common subspace.

2. (a) $I > J_k$ for $\forall k$, $I < \sum_{k=1}^{K} J_k$

   (b) $I \leq J_k$ for some k, and $I > J_n$ for $n \neq k$

   The rows of $\boldsymbol{C}$ have a similar structures as in (6).

3. $I \geq \sum_{k=1}^{K} J_k$. In this case, the rows of $\boldsymbol{C}$ only contain ones and zeros, and their ordering is defined as follows:

$$
\boldsymbol{C}(k,:) = \begin{bmatrix} \boldsymbol{1}_{J_1}^{\mathrm{T}} & \boldsymbol{0}_{J_2}^{\mathrm{T}} & \cdots & \boldsymbol{0}_{J_K}^{\mathrm{T}} \\ \boldsymbol{0}_{J_1}^{\mathrm{T}} & \boldsymbol{1}_{J_2}^{\mathrm{T}} & \cdots & \boldsymbol{0}_{J_K}^{\mathrm{T}} \\ & & \vdots & \\ \boldsymbol{0}_{J_1}^{\mathrm{T}} & \boldsymbol{0}_{J_2}^{\mathrm{T}} & \cdots & \boldsymbol{1}_{J_K}^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{K \times I}, \tag{8}
$$

where $\boldsymbol{0}_{J_k}^{\mathrm{T}} \in \mathbb{R}^{1 \times J_k}$ and $\boldsymbol{1}_{J_k}^{\mathrm{T}} \in \mathbb{R}^{1 \times J_k}$ are row vectors of zeros and ones, respectively.

From equation (5), it can be observed that the ML-GSVD has some similarities with the PARAFAC2 decomposition [11]. It is known from [11] that the uniqueness of equation (5) (up to column permutation and scaling) is ensured by the Harshman constraint $\boldsymbol{B}_k^{\mathrm{T}} \cdot \boldsymbol{B}_k = \boldsymbol{F}^{\mathrm{T}} \cdot \boldsymbol{F}$, such that $\boldsymbol{B}_k^{\mathrm{T}} = \boldsymbol{F}^{\mathrm{T}} \cdot \boldsymbol{V}_k$, $\boldsymbol{V}_k \cdot \boldsymbol{V}_k^{\mathrm{T}} = \boldsymbol{I}_R$, and K is large enough. In the ML-GSVD, since the matrices $\boldsymbol{B}_k$ are orthogonal, we set $\boldsymbol{F}$ to the identity matrix, which implies $\boldsymbol{B}_k^{\mathrm{H}} \cdot \boldsymbol{B}_k = \boldsymbol{I}_I$. The similarity between PARAFAC2 and the ML-GSVD motivates us to extend efficient algorithms to compute PARAFAC2 for computing the ML-GSVD, as will be discussed in the next section.

3. COMPUTATION OF THE ML-GSVD

In this section, we propose two ALS-based direct fitting algorithms to calculate the ML-GSVD. The first one is based on the Direct Fitting algorithm in [12], while the second is based on the computation of PARAFAC2 via double contractions [13]. The steps of the algorithms are detailed in Algorithm 1 and Algorithm 2, respectively. Both algorithms are initialized with the values of $\boldsymbol{A}$ based on the

Algorithm 1 ALS based direct fitting algorithm for the computation of the ML-GSVD of the set of K matrices $\boldsymbol{H}_k \in \mathbb{C}^{I \times J_k}$.

Require: Tensor $\mathcal{H}$
1: Initialize $\boldsymbol{A}$ and $\boldsymbol{C}$.
2: **repeat**
3:     **for** $k = 1, 2 \ldots, K$ **do**
4:         Reconstruct $\widetilde{\mathcal{H}}$ with $\boldsymbol{A}$ and $\boldsymbol{C}$:
$$\widetilde{\boldsymbol{H}}_k = \boldsymbol{A} \cdot \operatorname{diag}\{\boldsymbol{C}(k,:)\}$$
5:         Update $\boldsymbol{B}_k$ $(k = 1, \ldots, K)$
$$\boldsymbol{T}_k = \boldsymbol{H}_k^{\mathrm{H}} \widetilde{\boldsymbol{H}}_k, \quad \boldsymbol{B}_k = (\boldsymbol{T}_k^{\mathrm{H}} \boldsymbol{T}_k)^{-\frac{1}{2}} \boldsymbol{T}_k^{\mathrm{H}}$$
6:         Update $\widetilde{\mathcal{H}}$: $\widetilde{\boldsymbol{H}}_k = \boldsymbol{H}_k \cdot \boldsymbol{B}_k^{\mathrm{H}}$
7:     **end for**
8:     Update $\boldsymbol{A}$ and $\boldsymbol{C}$:
$$\boldsymbol{A} = [\widetilde{\mathcal{H}}]_{(1)} \left(\operatorname{bdiag}\{\frac{\boldsymbol{C}(:,1)^{\mathrm{H}}}{\|\boldsymbol{C}(:,1)\|^2}, \ldots, \frac{\boldsymbol{C}(:,I)^{\mathrm{H}}}{\|\boldsymbol{C}(:,I)\|^2}\} \right)^{\mathrm{T}},$$
$$\boldsymbol{C} = [\widetilde{\mathcal{H}}]_{(3)} \left(\operatorname{diag}\{\frac{1}{\|\boldsymbol{A}(:,1)\|^2}, \ldots, \frac{1}{\|\boldsymbol{A}(:,I)\|^2}\} (\boldsymbol{A} \diamond \boldsymbol{I}_{\mathrm{I}})^{\mathrm{H}} \right)^{\mathrm{T}}$$
9:     **for** $k = 1, 2 \ldots, K$ **do**
10:         $\hat{\boldsymbol{H}}_k = \boldsymbol{A} \cdot \operatorname{diag}\{\boldsymbol{C}(k,:)\} \cdot \boldsymbol{B}_k^{\mathrm{T}} \in \mathbb{C}^{I \times J_k}$,
11:     **end for**
12:     $E_{\mathrm{R}} = \left(\left\|\hat{\mathcal{H}} - \mathcal{H}\right\|_{\mathrm{H}}^2 \right) / \|\mathcal{H}\|_{\mathrm{H}}^2$
13: **until** $\Delta E_{\mathrm{R}} = \left(E_{\mathrm{R}}^{\mathrm{old}} - E_{\mathrm{R}} \right) / E_{\mathrm{R}}^{\mathrm{old}}$ is smaller than a predefined threshold, $E_{\mathrm{R}}^{\mathrm{old}}$ is the residual in the previous iteration
14: Normalize and reorder the columns of $\boldsymbol{C}$, reorder the columns of $\boldsymbol{A}$ and $\boldsymbol{B}_k^{\mathrm{T}}$ accordingly

---

Algorithm 2 ALS based algorithm for the computation of the ML-GSVD of the set of K matrices $\boldsymbol{H}_k \in \mathbb{C}^{I \times J_k}$

Require: Tensor $\mathcal{H}$
1: Initialize $\boldsymbol{A}$ and $\boldsymbol{C}$.
2: **repeat**
3:     **for** $k = 1, 2 \ldots, K$ **do**
4:         Reconstruct $\widetilde{\mathcal{H}}$ with $\boldsymbol{A}$ and $\boldsymbol{C}$: $\widetilde{\boldsymbol{H}}_k = \boldsymbol{A} \cdot \operatorname{diag}\{\boldsymbol{C}(k,:)\}$
5:         Update $\boldsymbol{B}_k$ $(k = 1, \ldots, K)$:
$$\boldsymbol{T}_k = \boldsymbol{H}_k^{\mathrm{H}} \widetilde{\boldsymbol{H}}_k, \quad \boldsymbol{B}_k = (\boldsymbol{T}_k^{\mathrm{H}} \boldsymbol{T}_k)^{-\frac{1}{2}} \boldsymbol{T}_k^{\mathrm{H}}$$
6:     **end for**
$$\bar{\boldsymbol{C}} = \left((\boldsymbol{I}_K \otimes \boldsymbol{1}_R^{\mathrm{T}}) \diamond \operatorname{vec}\left\{\boldsymbol{C}^{\mathrm{T}}\right\}^{\mathrm{T}} \right) \in \mathbb{R}^{K \times IK}$$
7:     Estimate $\boldsymbol{A}$: $\boldsymbol{A} = [\mathcal{H}]_{(1)} \cdot \left((\boldsymbol{1}_K^{\mathrm{T}} \otimes \boldsymbol{I}_I) \cdot [\widetilde{\boldsymbol{B}}]_{(2)} \diamond \bar{\boldsymbol{C}})^{\mathrm{T}} \right)^{+}$
$$\bar{\boldsymbol{A}} = \boldsymbol{A} \left(\boldsymbol{1}_K^{\mathrm{T}} \otimes \boldsymbol{I}_I \right) \in \mathbb{C}^{I \times IK}$$
8:     Estimate $\boldsymbol{C}$ by solving the non-negative least squares problem [14]:
$$\operatorname{vec}\{\mathcal{H}\} \approx [\widetilde{\boldsymbol{B}}]_{(2)} \diamond (\boldsymbol{I}_K \otimes \boldsymbol{1}_R^{\mathrm{T}}) \diamond \bar{\boldsymbol{A}} \cdot \operatorname{vec}\left\{\boldsymbol{C}^{\mathrm{T}}\right\}$$
9:     **for** $k = 1, 2 \ldots, K$ **do**
10:         $\hat{\boldsymbol{H}}_k = \boldsymbol{A} \cdot \operatorname{diag}\{\boldsymbol{C}(k,:)\} \cdot \boldsymbol{B}_k^{\mathrm{T}} \in \mathbb{C}^{I \times J_k}$
11:     **end for**
12:     $E_{\mathrm{R}} = \left(\left\|\hat{\mathcal{H}} - \mathcal{H}\right\|_{\mathrm{H}}^2 \right) / \|\mathcal{H}\|_{\mathrm{H}}^2$
13: **until** $\Delta E_{\mathrm{R}} = \left(E_{\mathrm{R}}^{\mathrm{old}} - E_{\mathrm{R}} \right) / E_{\mathrm{R}}^{\mathrm{old}}$ is smaller than a predefined threshold, $E_{\mathrm{R}}^{\mathrm{old}}$ is the residual in the previous iteration
14: Normalize and reorder the columns of $\boldsymbol{C}$, reorder the columns of $\boldsymbol{A}$ and $\boldsymbol{B}_k^{\mathrm{T}}$ accordingly

---

SVD of $\sum_{k=1}^{K} \boldsymbol{H}_k \boldsymbol{H}_k^{\mathrm{H}}$ and with a random non-negative matrix $\boldsymbol{C}$. The unitary matrix $\boldsymbol{B}_k$ is updated in step 5 via the extended generalized solution of the complex-valued Orthogonal Procrustes Problem (OPP) [15].

In the last step the normalization of the columns of $\boldsymbol{C}$ is performed to ensure that they have unit norm. To ensure that the elements of $\boldsymbol{C}$ are real, we multiply $\operatorname{diag}\{\boldsymbol{C}(k,:)\}$ by its complex conjugate, and compensate it in the columns of $\boldsymbol{B}_k^{\mathrm{T}}$ (included in step 14). Next, the elements of $\boldsymbol{C}$ are ordered as in (6) and in the description below this equation, while the columns of $\boldsymbol{A}$ and the rows of $\boldsymbol{B}_k$ are reordered accordingly. Note that in the case $I > \max(J_1, \ldots, J_K)$, zero columns are appended to the matrices $\boldsymbol{H}_k \in \mathbb{C}^{I \times J_k}$ in Alg. 2, so that the original tensor $\mathcal{H}$ is of size $I \times I \times K$.

Given an arbitrary tensor, our simulations show that the proposed ML-GSVD is almost exact in case 3 (yielding an error E_{R} on the order of 10^{-7}). In cases 1 and 2 a good approximation is obtained in the least squares sense (the error E_{R} is on the order of 10^{-2}).

4. APPLICATION TO COORDINATED BEAMFORMING IN MULTIUSER MIMO SYSTEMS

An interesting application of the ML-GSVD is the design of transmit and receive beamforming matrices in a coordinated multi-user MIMO system combining broadcast and multicast transmissions. In [1] and [3], the authors proposed a GSVD-based coordinated beamforming, which was limited to 2 destinations (users). The proposed ML-GSVD provides a generalization to any number K of users that is less or equal than the number of transmit antennas I.

Fig. 2 illustrates a coordinated downlink multi-user MIMO system, where the source (base station) serves $K = 3$ destinations (users). The matrix $\boldsymbol{H}_k \in \mathbb{C}^{I \times J_k}$ is a slice of the multi-user MIMO channel tensor $\mathcal{H}$, representing the channel between the access point and the kth user. The received signal at the kth user is given by:

$$\boldsymbol{y}_k = \boldsymbol{H}_k^{\mathrm{T}} \{\boldsymbol{A}^{\mathrm{T}}\}_{\mathcal{C}(\mathcal{Q})}^{-1} \{\boldsymbol{x}\}_{\mathcal{R}(\mathcal{Q})} + \boldsymbol{n}_k, \tag{9}$$

and at the detector we get:

$$\hat{\boldsymbol{y}}_k = \{\boldsymbol{B}_k^{\mathrm{H}}\}_{\mathcal{R}(\mathcal{Q})} \boldsymbol{y}_k, \tag{10}$$

where $\boldsymbol{x}$ is the transmitted data vector, $\{\boldsymbol{A}^{\mathrm{T}}\}_{\mathcal{C}(\mathcal{Q})}^{-1}$ is the transmit beamforming matrix, and $\{\boldsymbol{B}_k^{\mathrm{H}}\}_{\mathcal{R}(\mathcal{Q})}$ is the receive beamforming matrix, which jointly diagonalize the channel matrices represented by $\boldsymbol{H}_k$ to get virtual channels (VCs) that enable a simultaneous point-to-multipoint connection with private and common messages. $\mathcal{C}(\mathcal{Q})$ and $\mathcal{R}(\mathcal{Q})$ denote the columns and rows of the matrix with indices in the set $\mathcal{Q} \subseteq \{1, \ldots, I\}$. The required subset of VCs (private or common subspaces) can be chosen by appropriate selection of the columns of the transmit precoding matrix, and the corresponding rows of the receive beamforming matrices. For instance, if the ith and $(i+1)$th columns of $\boldsymbol{C}$ lie in a common subspace, for broadcasting we choose $\mathcal{Q} \in \{i, i+1\}$, and select the ith and $(i+1)$th columns and rows of the transmit and receive beamforming matrices, respectively. The elements of $\operatorname{diag}\{\boldsymbol{C}(k,:)\}$ contain the normalized gains of the corresponding virtual channels (private or common subspaces). The condition $I > J_k$ is the requirement to have private subspaces, while if $I < J_k$ only broadcasting is possible (see Section 2). The private subspaces (ones and zeros) in (6) are used by the source S to send confidential messages to the user U_k,
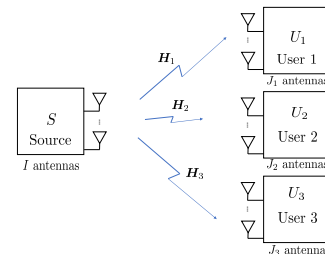

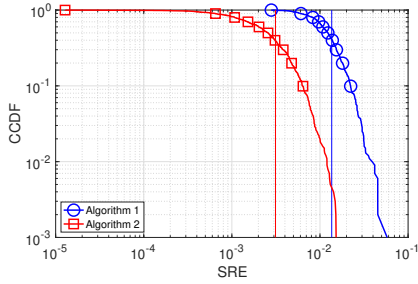**Fig. 2**: Source-to-3 destinations downlink multi-user MIMO.

Fig. 3: CCDF of SRE for the three-way complex-valued tensor with slices of size 10×3, 10×4, and 10×5, respectively. The vertical lines represent the mean value.

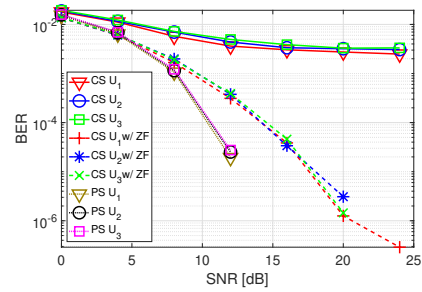**Fig. 4**: BER vs. SNR. 1200 QPSK symbols, $I = 12$, $J_1 = 6$, $J_2 = 7$, $J_3 = 8$. The results are averaged over 5000 channel realizations.

while the common subspace ($\boldsymbol{\sigma}_k$) is used for broadcasting common messages to all users. The private subspaces always have unit normalized gains, and the normalized gains of the common subspaces are less than one. Note that the resulting number of private and common subspaces depends both on the dimensionality and realization of the channel tensor $\mathcal{H} \in \mathbb{C}^{I \times J \times K}$.

5. SIMULATION RESULTS

In this section, we show some simulation results to evaluate the performance of the proposed ML-GSVD. First, we assess the performance of the two algorithms proposed in Section 3. In our simulations, the complex-valued tensor $\mathcal{H}$ is generated randomly from a zero mean unit variance complex Gaussian distribution. As an accuracy measure, we use the $\mathrm{SRE} = \left\| \hat{\mathcal{H}} - \mathcal{H} \right\|_{\mathrm{H}}^2 / \|\mathcal{H}\|_{\mathrm{H}}^2$ (Squared Reconstruction Error), where $\hat{\mathcal{H}}$ is the reconstructed channel tensor using the estimated factor matrices. The maximum number of iterations for both algorithms is set to 200. The algorithms are stopped if they reach the maximum number of iterations or if they reach the minimum error of the cost function, which is fixed to 10^{-7}.

In Fig. 3, we depict the CCDF (Complementary Cumulative Distribution Function) of the SRE for a complex-valued tensor with the slices of size 10×3, 10×4, and 10×5. The SREs presented in Fig. 3 corresponds to 1000 realizations of $\mathcal{H}$. The vertical lines represent the mean values for the each curve. The Algorithm 2 has a better accuracy, while the Algorithm 1 is significantly faster.

In the next experiment, we evaluate the performance of the proposed ML-GSVD based beamforming for a coordinated downlink MIMO system with three users. We plot the average BER as a function of the signal-to-noise ratio (SNR), which is defined as $\mathrm{SNR} = 1/\sigma_{\mathrm{n}}^2$, where σ_n^2 is the noise variance. We choose an asymmetrical scenario with $I = 12$ transmit antennas at the source, and, respectively, $J_1 = 6$, $J_2 = 7$, and $J_3 = 8$ receive antennas at the users. The channels $\boldsymbol{H}_1$, $\boldsymbol{H}_2$, and $\boldsymbol{H}_3$ are drawn from a zero mean i.i.d. circularly symmetric complex Gaussian distribution. At each run, 1200 QPSK symbols are transmitted. The simulation results are depicted in Fig. 4. As it can be observed, the PSs have an identical BER performance, while the performance of the CSs is worse. This is explained by the subspace structure of $\boldsymbol{C}$: In this case, the PSs have a higher gain than CSs. These gains are obtained as the product of the normalized gain in $\boldsymbol{C}$ and the norm of the corresponding column in $\boldsymbol{A}$. To improve the BER for the CSs, we employ the ML-GSVD-based zero-forcing detection. Furthermore, additional
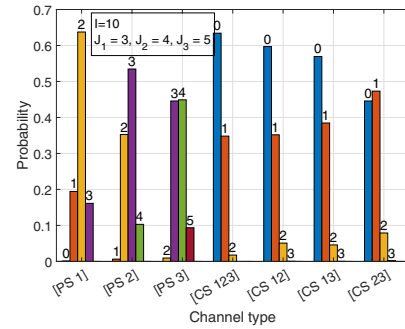

Fig. 5: Probability histogram. The numbers above the bars indicate the number of PSs and CSs. The results are averaged over 5000 channel realizations.

power allocation could be used to improve the overall BER performance. That should be done by allocating more power to the CSs as compare to the PSs. This study will be further investigated in the extended version of this paper. Figure 5 illustrates the histogram of the probability of having private subspaces (PSs) and common subspaces (CSs) with $I = 10$ transmit antennas at the source, and $J_1 = 3$, $J_2 = 4$ and $J_3 = 5$ antennas at users, respectively. As it can be seen, the proposed ML-GSVD beamforming also provides CSs between 2 of 3 users.

6. CONCLUSIONS

We have presented a new Multilinear Generalized Singular Value Decomposition (ML-GSVD) as an extension of GSVD to jointly factorize a set of an arbitrary number ($K \geq 2$) of matrices with a common number of rows or columns. In comparison with existing GSVD generalization methods, our ML-GSVD preserves the properties of the original GSVD, such as orthogonality of the second mode factor matrices. Two ALS-based algorithms to compute the ML-GSVD have been formulated. Moreover, an application of the ML-GSVD to coordinated beamforming in the downlink of a multi-user MIMO system was examined. Our numerical results have shown that a base station can transmit both common and confidential messages to $K \geq 2$ users simultaneously, by exploiting the subspace structure of the common and private virtual channels via the ML-GSVD.

As an interesting perspective, we shall consider a physical layer security application [3], where the ML-GSVD can be used to extend the existing results to more than two users.

7. REFERENCES

[1] D. Senaratne and C. Tellambura, "Generalized Singular Value Decomposition for Coordinated Beamforming in MIMO Systems," in *Proc. 2010 IEEE Global Telecommunications Conference GLOBECOM 2010*, Dec 2010, pp. 1–6.

[2] D. Senaratne and C. Tellambura, "GSVD Beamforming for Two-User MIMO Downlink Channel," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 6, pp. 2596–2606, July 2013.

[3] A. Khisti, G. Wornell, A. Wiesel, and Y. Eldar, "On the Gaussian MIMO Wiretap Channel," in *Proc. 2007 IEEE International Symposium on Information Theory*, June 2007, pp. 2471–2475.

[4] K. A. Aiello and O. Alter, "Platform-Independent Genome-wide Pattern of DNA Copy-Number Alterations Predicting Astrocytoma Survival and Response to Treatment Revealed by the GSVD Formulated as a Comparative Spectral Decomposition," *PLOS ONE*, vol. 11, no. 10, pp. 1–22, 10 2016.

[5] O. Alter, P. O. Brown, and D. Botstein, "Generalized singular value decomposition for comparative analysis of genome-scale expression data sets of two different organisms," *Proceedings of the National Academy of Sciences*, vol. 100, no. 6, pp. 3351–3356, 2003.

[6] C. H. Lee, B. O. Alpert, P. Sankaranarayanan, and O. Alter, "GSVD Comparison of Patient-Matched Normal and Tumor aCGH Profiles Reveals Global Copy-Number Alterations Predicting Glioblastoma Multiforme Survival," *PLOS ONE*, vol. 7, no. 1, pp. 1–11, 01 2012.

[7] C. Paige and M. Saunders, "Towards a Generalized Singular Value Decomposition," *SIAM Journal on Numerical Analysis*, vol. 18, no. 3, pp. 398–405, 1981.

[8] C. F. Van Loan, "Generalizing the Singular Value Decomposition," *SIAM Journal on Numerical Analysis*, vol. 13, pp. 76–83, Mar. 1976.

[9] S. P. Ponnapalli, M. A. Saunders, C. F. Van Loan, and O. Alter, "A Higher-Order Generalized Singular Value Decomposition for Comparison of Global mRNA Expression from Multiple Organisms," *PLOS ONE*, vol. 6, no. 12, pp. 1–11, 12 2011.

[10] P. Sankaranarayanan, T. E. Schomay, K. A. Aiello, and O. Alter, "Tensor GSVD of Patient- and Platform-Matched Tumor and Normal DNA Copy-Number Profiles Uncovers Chromosome Arm-Wide Patterns of Tumor-Exclusive Platform-Consistent Alterations Encoding for Cell Transformation and Predicting Ovarian Cancer Survival," *PLOS ONE*, vol. 10, no. 4, pp. 1–21, 04 2015.

[11] R. A. Harshman, "PARAFAC2: Mathematical and technical notes," *UCLA Working Papers in Phonetics*, vol. 22, pp. 30 – 44, 1972.

[12] H. A. L. Kiers, J. M. F. ten Berge, and R. Bro, "PARAFAC2 - part I. A direct fitting algorithm for the PARAFAC2 model," *Journal of Chemometrics*, 1999.

[13] K. Naskovska, Y. Cheng, A. L. F. de Almeida, and M. Haardt, "Efficient Computation of the PARAFAC2 Decomposition via Generalized Tensor Contractions," in *Proc. 2018 52nd Asilomar Conference on Signals, Systems, and Computers*, Oct. 2018, pp. 323–327.

[14] R. Bro and S. De Jong, "A Fast Non-negativity-constrained Least Squares Algorithm," *Journal of Chemometrics*, vol. 11, pp. 393–401, 09 1997.

[15] P. H. Schoenemann, "A generalized solution of the Orthogonal Procrustes Problem," *Psychometrika*, vol. 31, no. 1, 1966.