Generalized Tensor Contractions for an Improved Receiver Design in MIMO-OFDM Systems

Kristina Naskovska *, *Martin Haardt* * and *André L. F. de Almeida* §
*Communications Research Laboratory, Ilmenau University of Technology, P. O. Box 100565, D-98684 Ilmenau, Germany
Email: kristina.naskovska, martin.haardt@tu-ilmenau.de
§Department of Teleinformatics Engineering, Federal University of Ceará (UFC), Fortaleza, Brazil

***Abstract*—Tensor contraction is a multilinear algebra operator that defines an inner product between two tensors with compatible dimensions. In this work, we show that the MIMO-OFDM (Multiple-Input Multiple-Output - Orthogonal Frequency Division Multiplexing) received signal can be modeled by means of the tensor contraction operator. This tensor model is obtained without requiring additional spreading and provides a new, compact, and flexible formulation of a MIMO-OFDM system. Moreover, exploiting it at the receiver side facilitates the design of several types of receivers based on iterative LS (Least Squares) or recursive LS. We compare the proposed iterative and recursive LS based receivers with and without enumeration and show their advantages over the traditional ZF-FFT (Zero Forcing - Fast Fourier Transform) receiver for MIMO OFDM. This structured tensor model also opens new research directions. Moreover, our generalized tensor contraction formulation can be extended to different multi-carrier MIMO systems.**

I. Introduction

OFDM (Orthogonal Frequency Division Multiplexing) is the most widely used multicarrier technique in current wireless communication systems. It is robust in multipath propagation environments and has a simple and efficient implementation [1], [2]. Using the FFT (Fast Fourier Transform) the complete frequency band is divided into smaller frequency subcarriers. The use of the cyclic prefix mitigates the ISI (Inter-Symbol Interference) and the ICI (Inter-Carrier Interference). Typically, the OFDM receiver is implemented in the frequency domain based on ZF (Zero Forcing) filter. More advanced solutions are proposed in [3]. Furthermore, optimal training and channel estimation for OFDM systems are proposed in [4], [5].

Tensor signal processing offers an improved identifiability, uniqueness and more efficient denoising compared to matrix based techniques. A good overview of multilinear algebra, tensor decompositions, and applications of tensor signal processing are provided in [6], [7], and [8]. Previous publications on tensor models for multicarrier communications systems [9], [10], [11], and [12] do not exploit the channel correlation between the adjacent subcarriers.

In [9], a MIMO (Multiple-Input Multiple-Output) multicarrier system is modeled using tensor algebra and the PARATUCK2 tensor decomposition resulting in a novel space, time, and frequency coding structure. Similarly in [10], trilinear coding in space, time, and frequency for MIMO-OFDM systems is proposed based on the CP (Canonical Polyadic) tensor decomposition. By exploiting tensor models, semi-blind receivers have been introduced for multicarrier communications systems in [12] and [11]. All these publications use additional spreading that leads to a significantly reduced spectral efficiency to create the tensor structure.

An inner product between two N-way tensors is also known as a tensor contraction [8]. Such a contraction operator has been introduced in [13]. Moreover, it is used in [14] to model two-hop MIMO relaying systems. Furthermore, in [15] an application of the contraction operator to MIMO-OFDM systems is presented. However, the transmitted signal tensor is assumed to be Khatri-Rao coded with a CP structure. In this work, we present the contraction between an uncoded signal tensor and a channel tensor for OFDM systems, yielding the same spectral efficiency as matrix based approaches (since no additional spreading is used).

Since we do not use the Khatri-Rao coding structure as in [15] we do not get the corresponding the CP structure for the transmit signal tensor. By exploiting this new tensor structure, we can reshape it into the factorization of a sum of Kharti-Rao products. This problem can be solved by means of generalized techniques based on iterative and recursive least squares originally proposed for blind source separation. Such techniques are presented in [16] and [17]. By extending these techniques, we propose a joint channel and symbol estimation algorithm for OFDM systems based on iterative and recursive least squares, with and without enumeration. Our computer simulations show the advantages of the proposed receives compared to the traditional ZF-FFT (Zero Forcing - Fast Fourier Transform) receiver.

The rest of this paper is organized as follows. In Section II we present the notation and some tensor algebra definitions used later in this paper. In Section III we introduce the proposed system model and the proposed receiver solutions. We evaluate the performance of the proposed receivers based on simulations presented in Section IV. In Section V we conclude this paper.

II. Tensor Algebra and Notation

We use the following notation. Scalars are denoted either as capital or lower-case italic letters, A, a. Vectors and matrices, are denoted as bold-face capital and lower-case letters, $\boldsymbol{a}, \boldsymbol{A}$, respectively. Tensors are represented by bold-face calligraphic letters $\boldsymbol{\mathcal{A}}$. The following superscripts, ^{T}, ^{H}, $^{-1}$, and $^+$ denote transposition, Hermitian transposition, matrix inversion and Moore-Penrose pseudo matrix inversion, respectively. The outer product, Kronecker product, and Khatri-Rao product are denoted as $\circ$, $\otimes$, and $\diamond$, respectively. The operators $||.||_{\mathrm{F}}$ and $||.||_{\mathrm{H}}$ denote the Frobenius norm and the higher order norm, respectively. Moreover, the n-mode product between a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{I_1 \times I_2 \ldots \times I_N}$ and a matrix $\boldsymbol{B} \in \mathbb{C}^{J \times I_n}$ is defined as $\boldsymbol{\mathcal{A}} \times_n \boldsymbol{B}$, for $n = 1, 2, \ldots N$ [7]. A super-diagonal or identity N-way tensor of dimension $R \times R \ldots \times R$ is denoted as $\boldsymbol{\mathcal{I}}_{N,R}$. Similarly, an identity matrix of dimension $R \times R$ is denoted as $\boldsymbol{I}_R$ and we denote a vector of ones of length R as $\boldsymbol{1}_R$. The n-th 3-mode slice of a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{I \times J \times N}$ is denoted as $\boldsymbol{\mathcal{A}}_{(.,.,n)}$ and accordingly one element of this tensor is denoted as $\boldsymbol{\mathcal{A}}_{(i,j,n)}$. The operator $\mathrm{diag}(.)$ transforms a vector into a diagonal matrix and the operator $\mathrm{vec}(.)$ transforms a matrix into a vector.

The contraction $\boldsymbol{\mathcal{A}} \bullet_n^m \boldsymbol{\mathcal{C}}$ between two tensors $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{I_1 \times I_2 \ldots \times I_N}$ and $\boldsymbol{\mathcal{C}} \in \mathbb{C}^{J_1 \times J_2 \ldots \times J_N}$ represents an inner product of the n-th mode of $\boldsymbol{\mathcal{A}}$ with the m-th mode of $\boldsymbol{\mathcal{C}}$, provided that $I_n = J_m$ [8]. Contraction along several modes of compatible dimensions is also possible and accordingly the contraction along two modes is denoted as $\boldsymbol{\mathcal{A}} \bullet_{n,k}^{m,l} \boldsymbol{\mathcal{C}}$. The contraction along two modes between the tensors $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{I \times J \times M \times N}$ and $\boldsymbol{\mathcal{C}} \in \mathbb{C}^{M \times N \times K}$ is defined as [8],

$$(\boldsymbol{\mathcal{A}} \bullet_{3,4}^{1,2} \boldsymbol{\mathcal{C}})_{(i,j,k)} \triangleq \sum_{n=1}^{N} \sum_{m=1}^{M} \boldsymbol{\mathcal{A}}_{(i,j,m,n)} \cdot \boldsymbol{\mathcal{C}}_{(m,n,k)} = \boldsymbol{\mathcal{T}}_{(i,j,k)}.$$

This example represents a contraction of the 3-rd and 4-th mode of $\boldsymbol{\mathcal{A}}$ with the 1-st and 2-nd mode of $\boldsymbol{\mathcal{C}}$, respectively.

Using the concept of the generalized unfoldings [18], [15], it can be shown that the tensor contraction satisfies

$$[\boldsymbol{\mathcal{A}} \bullet_{3,4}^{1,2} \boldsymbol{\mathcal{C}}]_{([1,2],3)} = [\boldsymbol{\mathcal{A}}]_{([1,2],[3,4])} \cdot [\boldsymbol{\mathcal{C}}]_{([1,2],3)} =$$
$$[\boldsymbol{\mathcal{A}} \bullet_{4,3}^{2,1} \boldsymbol{\mathcal{C}}]_{([1,2],3)} = [\boldsymbol{\mathcal{A}}]_{([1,2],[4,3])} \cdot [\boldsymbol{\mathcal{C}}]_{([2,1],3)}.$$

In the generalized unfolding $[\boldsymbol{\mathcal{A}}]_{([1,2],[3,4])}$ the 1-st mode varies faster than the 2-nd mode between the rows and the 3-rd mode varies faster then the 4-th mode between the columns. A visualization of this generalized unfolding and the index ordering is depicted in Fig. 1.

The CP tensor decomposition decomposes a given tensor into the minimum number of rank one components. This CP decomposition

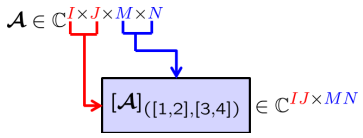

Fig. 1: Visualization of the generalized unfolding $[\mathcal{A}]_{([1,2],[3,4])}$.

of a 3-way, low rank noiseless tensor $\mathcal{A} \in \mathbb{C}^{I \times J \times M}$ into R rank one components is defined as

$$\mathcal{A} = \mathcal{I}_{3,R} \times_1 \boldsymbol{F}_1 \times_2 \boldsymbol{F}_2 \times_3 \boldsymbol{F}_3,$$

where $\boldsymbol{F}_1 \in \mathbb{C}^{I \times R}, \boldsymbol{F}_2 \in \mathbb{C}^{J \times R}$, and $\boldsymbol{F}_3 \in \mathbb{C}^{M \times R}$ are the factor matrices [7], [6]. For a tensor with a CP structure its unfoldings can be expressed in terms of factor matrices. For instance, the 3-mode unfolding of the tensor $\mathcal{A}$, i.e., $[\mathcal{A}]_{([3],[1,2])}$, satisfies

$$[\mathcal{A}]_{([3],[1,2])} = \boldsymbol{F}_3 \cdot (\boldsymbol{F}_2 \diamond \boldsymbol{F}_1)^T.$$

In a similar way, the rest of the tensor unfoldings can be defined.

III. MIMO-OFDM System

We assume a MIMO-OFDM system with M_T transmit and M_R receive antennas. All signals and equations are in the frequency domain. Moreover, N is the number of subcarriers and K denotes the number of transmitted frames. The received signal in the frequency domain $\tilde{\mathcal{Y}} \in \mathbb{C}^{N \times M_R \times K}$ after the removal of the cyclic prefix is defined by the means of the contraction operator

$$\tilde{\mathcal{Y}} = \tilde{\mathcal{H}} \bullet_{2,4}^{1,2} \tilde{\mathcal{S}} + \tilde{\mathcal{N}}.$$

The frequency selective propagation channel is represented by a channel tensor $\tilde{\mathcal{H}} \in \mathbb{C}^{N \times N \times M_R \times M_T}$ as in [15]. For each receive-transmit antenna pair the channel transfer matrix is a diagonal matrix and this represents the corresponding slice of the tensor $\tilde{\mathcal{H}}$, i.e., $\tilde{\mathcal{H}}_{(.,.,m_R,m_T)} = \operatorname{diag}\left(\tilde{\boldsymbol{h}}^{(m_R,m_T)}\right)$. The vector $\tilde{\boldsymbol{h}}^{(m_R,m_T)} \in \mathbb{C}^{N \times 1}$ contains the frequency domain channel coefficients. We assume that the channel stays constant during the K frames. The transmit signal tensor is denoted as $\tilde{\mathcal{S}} \in \mathbb{C}^{N \times M_T \times K}$ and $\tilde{\mathcal{N}} \in \mathbb{C}^{N \times M_R \times K}$ represents the additive white Gaussian noise in the frequency domain.

Using the generalized unfoldings for the received signal in the frequency domain we get

$$[\tilde{\mathcal{Y}}]_{([1,2],[3])} = [\tilde{\mathcal{H}}]_{([1,3],[2,4])} \tilde{\boldsymbol{\mathcal{S}}}_{([1,2],[3])} + [\tilde{\mathcal{N}}]_{([1,2],[3])} \in \mathbb{C}^{N \cdot M_R \times K}. \tag{1}$$

In [15] we have shown that the $([1,3],[2,4])$ generalized unfolding of the channel tensor can be expressed as

$$[\tilde{\mathcal{H}}]_{([1,3],[2,4])} = \bar{\boldsymbol{H}} \diamond (\boldsymbol{1}_{M_T}^T \otimes \boldsymbol{I}_N) \in \mathbb{C}^{N \cdot M_R \times N \cdot M_T},$$

where $\bar{\boldsymbol{H}} \in \mathbb{C}^{M_R \times N \cdot M_T}$ is a matrix containing all non zero elements of the tensor $\tilde{\mathcal{H}}$ and it is defined as,

$$\bar{\boldsymbol{H}} = \left[\begin{array}{c|c|c|c} \tilde{\boldsymbol{h}}^{(1,1)T} & \tilde{\boldsymbol{h}}^{(1,2)T} & \dots & \tilde{\boldsymbol{h}}^{(1,M_T)T} \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{\boldsymbol{h}}^{(M_R,1)T} & \tilde{\boldsymbol{h}}^{(M_R,2)T} & \dots & \tilde{\boldsymbol{h}}^{(M_R,M_T)T} \end{array}\right] \tag{2}$$

$$= \left[\begin{array}{cccc} \tilde{\boldsymbol{H}}_R^{(1)} & \tilde{\boldsymbol{H}}_R^{(2)} & \dots & \tilde{\boldsymbol{H}}_R^{(M_T)} \end{array}\right] \in \mathbb{C}^{M_R \times N \cdot M_T}. \tag{3}$$

The matrix $\tilde{\boldsymbol{\mathcal{S}}}_{([1,2],[3])} \in \mathbb{C}^{N \cdot M_T \times K}$ is the transpose of the 3-mode unfolding of $\tilde{\boldsymbol{\mathcal{S}}}$. For notational simplicity, we define the following block matrix $\bar{\boldsymbol{S}}$

$$\bar{\boldsymbol{S}} = \tilde{\boldsymbol{\mathcal{S}}}_{([1,2],[3])}^T = \left[\begin{array}{cccc} \tilde{\boldsymbol{S}}^{(1)} & \tilde{\boldsymbol{S}}^{(2)} & \dots & \tilde{\boldsymbol{S}}^{(M_T)} \end{array}\right] \in \mathbb{C}^{K \times N \cdot M_T}, \tag{4}$$

where $\tilde{\boldsymbol{S}}^{(m_T)} \in \mathbb{C}^{K \times N}$ contains the symbols transmitted via the m_T-th antenna. By substituting the corresponding tensor unfoldings in equation (1), we get

$$[\tilde{\mathcal{Y}}]_{([1,2],[3])} = \left(\bar{\boldsymbol{H}} \diamond (\boldsymbol{1}_{M_T}^T \otimes \boldsymbol{I}_N)\right) \cdot \bar{\boldsymbol{S}}^T + [\tilde{\mathcal{N}}]_{([1,2],[3])}. \tag{5}$$

The above equation satisfies the CP decomposition of a noisy tensor. By applying an inverse unfolding for the received signal in the frequency domain, we get after the removal of the cyclic prefix

$$\tilde{\mathcal{Y}} = \mathcal{I}_{3,N \cdot M_T} \times_1 (\boldsymbol{1}_{M_T}^T \otimes \boldsymbol{I}_N) \times_2 \bar{\boldsymbol{H}} \times_3 \bar{\boldsymbol{S}} + \tilde{\mathcal{N}}. \tag{6}$$

Our goal is to jointly estimate the channel and the symbols, i.e., $\bar{\boldsymbol{H}}$ and $\bar{\boldsymbol{S}}$ in equation (6). Note that all factor matrices are flat resulting in a degenerate CP model in all three modes. Therefore, it is difficult to estimate the channel and the symbols by simply fitting a CP model to the received signal tensor in (6).

Moreover, we assume that the symbol matrix consists of data and pilot symbols, $\bar{\boldsymbol{S}} = \bar{\boldsymbol{S}}_{\mathrm{d}} + \bar{\boldsymbol{S}}_{\mathrm{p}}$. The matrix $\bar{\boldsymbol{S}}_{\mathrm{d}}$ and $\bar{\boldsymbol{S}}_{\mathrm{p}}$ represents the data symbols and the pilot symbols, respectively. The matrix $\bar{\boldsymbol{S}}_{\mathrm{d}}$ contains zeros at the positions of the pilot symbols. Accordingly, the matrix $\bar{\boldsymbol{S}}_{\mathrm{p}}$ contains non-zero elements only at the pilot positions. Typically, there are three ways of arranging the pilot symbol within the OFDM blocks (block, comb, and lattice-type) [1]. As we assume that the channel stays constant during the K frames, we send pilots only within in the first frame with subcarrier spacing of ΔF between two pilot symbols. This results in $\lfloor \frac{N}{\Delta F} \rfloor$ pilot symbols per OFDM block. In comparison, other publications such as [9], [10], [11], and [12] use N pilot symbols per OFDM block. A reduced number of pilot symbols can be used only if the channel correlation among adjacent subcarriers is exploited for channel estimation.

Using the prior knowledge of the pilot symbols and their positions, the channel in the frequency domain can be estimated. Naturally, the channel is estimated only at those subcarrier positions where the pilot symbols are located. Afterwards, an interpolation is applied to get the complete channel estimate. Moreover, as shown in [4], [5] the channel can be first estimated in the time domain and then transformed into the frequency domain. Either way, this leads to a pilot based channel estimate that we denote as $\hat{\bar{\boldsymbol{H}}}_{\mathrm{p}}$, or $\hat{\tilde{\mathcal{H}}}_{\mathrm{p}}$. The pilot based channel estimate is then used to estimate the data symbols. In the rest of this section, we discuss different ways to estimate the symbols.

Traditionally, the estimate of the symbols is obtained in the frequency domain with a ZF receiver. In this case, the symbols are calculated by inverting the channel matrix for each subcarrier individually. This ZF receiver using the above defined tensor notation is summarized in Algorithm 1.

Algorithm 1: ZF receiver

initialization $\hat{\tilde{\mathcal{H}}}_{\mathrm{p}}$;
for $n = 1 : N$ **do**
  $\hat{\tilde{\boldsymbol{\mathcal{S}}}}_{(n,.,.)} \approx \hat{\tilde{\mathcal{H}}}_{\mathrm{p}(n,n,.,.)}^{+} \tilde{\mathcal{Y}}_{(n,.,.)}$;
end

Alternatively, if we compute the 1-mode unfolding ([1],[2,3]) of the tensor $\tilde{\mathcal{Y}}$ in equation (6), we get

$$[\tilde{\mathcal{Y}}]_{([1],[2,3])} = (\boldsymbol{1}_{M_T}^T \otimes \boldsymbol{I}_N) \cdot \left(\bar{\boldsymbol{S}} \diamond \bar{\boldsymbol{H}}\right)^T + [\tilde{\mathcal{N}}]_{([1],[3,2])} \in \mathbb{C}^{N \times M_R \cdot K}.$$

Tanking into account the structure of the matrices $(\boldsymbol{1}_{M_T}^T \otimes \boldsymbol{I}_N) \in \mathbb{R}^{N \times N \cdot M_T}$, $\bar{\boldsymbol{H}}$ in (3), and $\bar{\boldsymbol{S}}$ in (4), the above unfolding is

$$[\tilde{\mathcal{Y}}]_{([1],[2,3])} = \sum_{m_T=1}^{M_T} \left(\tilde{\boldsymbol{S}}^{(m_T)} \diamond \tilde{\boldsymbol{H}}_R^{(m_T)}\right)^T + [\tilde{\mathcal{N}}]_{([1],[3,2])}.$$

After transposition and omitting the noise term, we get

$$[\tilde{\mathcal{Y}}]_{([2,3],[1])} \approx \sum_{m_T=1}^{M_T} \left(\tilde{\boldsymbol{S}}^{(m_T)} \diamond \tilde{\boldsymbol{H}}_R^{(m_T)}\right) \in \mathbb{C}^{M_R \cdot K \times N}.$$

This sum of Khatri-Rao products can be resolved in a column-wise fashion. Let $\tilde{\boldsymbol{y}}_n \in \mathbb{C}^{M_R \cdot K \times 1}$ denote the n-th column of $[\tilde{\boldsymbol{\mathcal{Y}}}]_{([2,3],[1])} \in \mathbb{C}^{M_R \cdot K \times N}$. After reshaping this vector into matrix $\tilde{\boldsymbol{Y}}_n \in \mathbb{C}^{M_R \times K}$, such that $\tilde{\boldsymbol{y}}_n = \mathrm{vec}(\tilde{\boldsymbol{Y}}_n)$, it is easy to see that this matrix satisfies

$$\tilde{\boldsymbol{Y}}_n \approx \tilde{\boldsymbol{H}}_n \cdot \tilde{\boldsymbol{S}}_n, \tag{7}$$

where $\tilde{\boldsymbol{H}}_n$ and $\tilde{\boldsymbol{S}}_n$ are the n-th slice of $\tilde{\boldsymbol{\mathcal{H}}}_{(n,n,:,:)} \in \mathbb{C}^{M_R \times M_T}$ and $\tilde{\boldsymbol{\mathcal{S}}}_{(n,:,:)} \in \mathbb{C}^{M_T \times K}$, respectively. Note that $\tilde{\boldsymbol{Y}}_n$ is the n-th slice of $\tilde{\boldsymbol{\mathcal{Y}}}_{(n,:,:)}$. Using the pseudo inverse of the channel we get the traditional ZF receiver as summarized in Algorithm 1.

Alternatively, the channel and the symbols on each subcarrier can be estimated by means of iterative or recursive LS algorithms. Similar algorithms were proposed in [16] and [17] for blind source separation. We have extended the algorithms presented in [17] to our application.

Algorithm 2: Iterative Least Squares with Projection

initialization $\tilde{\boldsymbol{\mathcal{H}}}_{\mathrm{p}}$, maxIteration, minErr;
for $n = 1 : N$ **do**
  set $i = 1$, $e = \infty$;
  while $i <$ *maxIteration or* $e >$ *minErr* **do**
    $\tilde{\bar{\boldsymbol{S}}}_n^{(i)} = (\tilde{\boldsymbol{H}}_n^{(i-1)H} \tilde{\boldsymbol{H}}_n^{(i-1)})^{-1} \tilde{\boldsymbol{H}}_n^{(i-1)H} \tilde{\boldsymbol{Y}}_n$;
    $\tilde{\boldsymbol{S}}_n^{(i)} = \mathrm{proj}\left(\tilde{\bar{\boldsymbol{S}}}_n^{(i)}\right)$;
    if $\mathrm{rank}\left(\tilde{\boldsymbol{S}}_n^{(i)}\right) = M_T$ **then**
      $\tilde{\boldsymbol{H}}_n^{(i)} = \tilde{\boldsymbol{Y}}_n \tilde{\boldsymbol{S}}_n^{(i)H} (\tilde{\boldsymbol{S}}_n^{(i)H} \tilde{\boldsymbol{S}}_n^{(i)})^{-1}$;
    else
      $\tilde{\boldsymbol{H}}_n^{(i)} = \tilde{\boldsymbol{H}}_n^{(i-1)}$;
    end
    $i = i + 1$, $e = \|\tilde{\boldsymbol{H}}_n^{(i-1)} - \tilde{\boldsymbol{H}}_n^{(i)}\|_{\mathrm{F}}^2$;
  end
end

Algorithm 3: Iterative Least Squares with Enumeration

initialization $\tilde{\boldsymbol{\mathcal{H}}}_{\mathrm{p}}$, maxIteration, minErr;
for $n = 1 : N$ **do**
  set $i = 1$, $e = \infty$;
  while $i <$ *maxIteration or* $e >$ *minErr* **do**
    for $k = 1 : K$ **do**
      $\hat{\boldsymbol{s}} = \arg \min_{\boldsymbol{s}^{(j)} \in \Omega} \|\tilde{\boldsymbol{Y}}_{n(.,k)} - \tilde{\boldsymbol{H}}_n^{(i-1)} \boldsymbol{s}^{(j)}\|$;
      $j = 1, \ldots M^{M_T}$;
      $\tilde{\boldsymbol{S}}_{n(.,k)}^{(i)} = \hat{\boldsymbol{s}}$;
    end
    if $\mathrm{rank}\left(\tilde{\boldsymbol{S}}_n^{(i)}\right) = M_T$ **then**
      $\tilde{\boldsymbol{H}}_n^{(i)} = \tilde{\boldsymbol{Y}}_n \tilde{\boldsymbol{S}}_n^{(i)H} (\tilde{\boldsymbol{S}}_n^{(i)H} \tilde{\boldsymbol{S}}_n^{(i)})^{-1}$;
    else
      $\tilde{\boldsymbol{H}}_n^{(i)} = \tilde{\boldsymbol{H}}_n^{(i-1)}$;
    end
    $i = i + 1$, $e = \|\tilde{\boldsymbol{H}}_n^{(i-1)} - \tilde{\boldsymbol{H}}_n^{(i)}\|_{\mathrm{F}}^2$;
  end
end

The first two algorithms, namely ILSP (Iterative Least Squares with Projection) and ILSE (Iterative Least Squares with Enumeration) summarized in Algorithm 2 and Algorithm 3, respectively, are iterative algorithms based on LS. Both algorithms are initialized with the pilot based channel estimate, the maximum number of iterations (maxIteration), and the minimum error difference between to consecutive updates (minErr). The ILSP algorithm is essentially an iterative version of the ZF algorithm, where in each iteration the estimated symbols are projected onto the finite alphabet Ω of the transmitted symbols. This finite alphabet depends on the modulation type and the modulation order M. Details regarding the convergence for different finite alphabets are discussed in [17]. To estimate the symbols, we compute a pseudo inverse of the channel which leads to the condition $M_R \geq M_T$. On the other hand, the ILSE algorithm does not require this condition as it estimates the symbols based on enumeration. Equation (8) represents the enumeration or the search over the final alphabet of symbols.

$$\hat{\boldsymbol{s}} = \arg \min_{\boldsymbol{s}^{(j)} \in \Omega} \|\tilde{\boldsymbol{Y}}_{n(.,k)} - \tilde{\boldsymbol{H}}_n \boldsymbol{s}^{(j)}\|, j = 1, \ldots M^{M_T} \tag{8}$$

Both algorithms update the channel only if it is possible, i.e., if the rank of the symbol matrix $\tilde{\boldsymbol{S}}_n \in \mathbb{C}^{M_T \times K}$ is M_T. Note that this is not possible for all values of M_T, K, and for all patterns of random data symbols.

Algorithm 4: Recursive Least Squares with Projection

initialization $\tilde{\boldsymbol{\mathcal{H}}}_{\mathrm{p}}$, $0 \leq \alpha \leq 1$;
for $n = 1 : N$ **do**
  $\tilde{\bar{\boldsymbol{S}}}_n = (\tilde{\boldsymbol{H}}_n^H \tilde{\boldsymbol{H}}_n)^{-1} \tilde{\boldsymbol{H}}_n^H \tilde{\boldsymbol{Y}}_n$;
  $\tilde{\boldsymbol{S}}_n = \mathrm{proj}\left(\tilde{\bar{\boldsymbol{S}}}_n\right)$;
  set $\boldsymbol{P}^{(0)} = \boldsymbol{I}_{M_T}$, $\tilde{\boldsymbol{H}}_n^{(0)} = \tilde{\boldsymbol{H}}_n$;
  for $k = 1 : K$ **do**
    $\boldsymbol{s} = \tilde{\boldsymbol{S}}_{n(.,k)}$;
    $\tilde{\boldsymbol{H}}_n^{(k)} = \tilde{\boldsymbol{H}}_n^{(k-1)} + \frac{\left(\tilde{\boldsymbol{Y}}_{n(.,k)} - \tilde{\boldsymbol{H}}_n^{(k-1)} \boldsymbol{s}\right)}{\alpha + \boldsymbol{s}^H \boldsymbol{P}^{(k-1)} \boldsymbol{s}} \boldsymbol{s}^H \boldsymbol{P}^{(k-1)}$;
    $\boldsymbol{P}^{(k)} = \frac{1}{\alpha}\left(\boldsymbol{P}^{(k-1)} - \frac{\boldsymbol{P}^{(k-1)} \boldsymbol{s} \boldsymbol{s}^H \boldsymbol{P}^{(k-1)}}{\alpha + \boldsymbol{s}^H \boldsymbol{P}^{(k-1)} \boldsymbol{s}}\right)$;
  end
end

Algorithm 5: Recursive Least Squares with Enumeration

initialization $\tilde{\boldsymbol{\mathcal{H}}}_{\mathrm{p}}$, $0 \leq \alpha \leq 1$;
for $n = 1 : N$ **do**
  set $\boldsymbol{P}^{(0)} = \boldsymbol{I}_{M_T}$, $\tilde{\boldsymbol{H}}_n^{(0)} = \tilde{\boldsymbol{H}}_n$;
  for $k = 1 : K$ **do**
    $\hat{\boldsymbol{s}} = \arg \min_{\boldsymbol{s}^{(j)} \in \Omega} \|\tilde{\boldsymbol{Y}}_{n(.,k)} - \tilde{\boldsymbol{H}}_n^{(k-1)} \boldsymbol{s}^{(j)}\|$;
    $j = 1, \ldots M^{M_T}$;
    $\tilde{\boldsymbol{S}}_{n(.,k)} = \hat{\boldsymbol{s}}$;
    $\tilde{\boldsymbol{H}}_n^{(k)} = \tilde{\boldsymbol{H}}_n^{(k-1)} + \frac{\left(\tilde{\boldsymbol{Y}}_{n(.,k)} - \tilde{\boldsymbol{H}}_n^{(k-1)} \hat{\boldsymbol{s}}\right)}{\alpha + \hat{\boldsymbol{s}}^H \boldsymbol{P}^{(k-1)} \hat{\boldsymbol{s}}} \hat{\boldsymbol{s}}^H \boldsymbol{P}^{(k-1)}$;
    $\boldsymbol{P}^{(k)} = \frac{1}{\alpha}\left(\boldsymbol{P}^{(k-1)} - \frac{\boldsymbol{P}^{(k-1)} \hat{\boldsymbol{s}} \hat{\boldsymbol{s}}^H \boldsymbol{P}^{(k-1)}}{\alpha + \hat{\boldsymbol{s}}^H \boldsymbol{P}^{(k-1)} \hat{\boldsymbol{s}}}\right)$;
  end
end

The remaining two algorithms, namely RLSP (Recursive Least Squares with Projections) and RSLE (Recursive Least Squares with Enumeration) are recursive implementations of ILSP and ISLE, respectively. In both algorithms the channel is estimated based on RLS (Recursive Least Squares), where α is the weighting coefficient and $\boldsymbol{P}$ denotes the inverse correlation matrix. Due to the computation of the pseudo-inverse of the channel matrix, $M_R \geq M_T$ should hold. On the other hand, the RLSE algorithm is suitable for any values of M_T, M_R, and K.

ILSP has the same computational complexity as traditional ZF receivers with the added complexity of the additional iterations if the symbol matrix has full rank. The ILSE algorithm does not compute a pseudo-inverse of the channel matrix. However its computational complexity comes from the enumeration and it depends on the number of antennas and the modulation order. The recursive algorithms, RLSP and RLSE require a finite number of iterations that is equal to $N \cdot K$. The RLSP algorithm still computes the pseudo-inverse of the

channel, however the RLSE does not compute any matrix pseudo-inverse. Therefore, the RLSE algorithm has a smaller computational complexity.

IV. Simulation Results

We compare the performance of the five algorithms, ZF, ILSP, ILSE, RLSP, and RLSE using Monte Carlo simulations. First, we consider a 2×2 OFDM system, with K frames, and $N = 128$ subcarriers. The pilot symbols are transmitted on every third subcarrier, $\Delta F = 3$ and only during the first frame. Using these pilots we obtain the pilot based channel estimate with which we initialize all of the algorithms. The transmitted data symbols are independent and modulated using 4-QAM modulation. The frequency selective propagation channel is modeled accoring to the 3GPP Pedestrian A channel (Ped A). The duration of the cyclic prefix is 32 samples and the weighting factor $\alpha = 1$. In Fig. 2 and 3 we depict the SER (Symbol Error Rate) as a function of the E_b/N_0 (energy per bit/ noise power spectral density) in dB for $K = 2$ and $K = 8$, respectively. Both algorithms based on enumeration, ILSE and RLSE

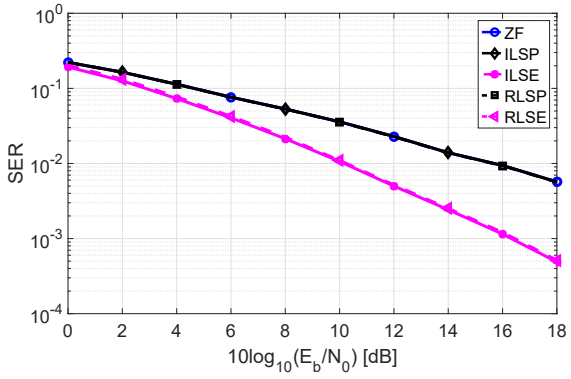

Fig. 2: SER for a 2×2 OFDM system, $\Delta F = 3, N = 128, K = 2$.

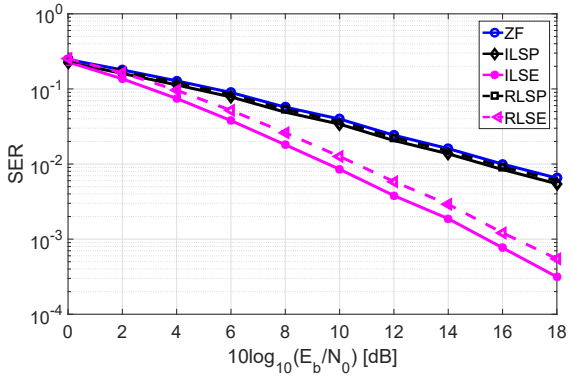

Fig. 3: SER for a 2×2 OFDM system, $\Delta F = 3, N = 128, K = 8$.

outperform the rest of the algorithms. The performance of ILSP and RLSP is similar to the ZF performance and it depends on the number of frames. As shown in Fig. 3, increasing the number of frames leads to a slightly better SER as compared to ZF. Note that the transmitted data symbols are independent and randomly drawn with no guarantee that the matrices $\tilde{\boldsymbol{S}}_n$ are of rank M_T. Therefore, in many cases the number of iterations is equal to one. In all of the other simulated cases the algorithms converge after 3 iterations. As in [17] we also observe that the iterative algorithms have a better performance than the recursive ones for an increased number of frames. However, the recursive algorithms, RSLP and RLSE, require less computational complexity than the iterative ones, ILSP and ILSE. Moreover, for the same simulation parameters as in Fig. 3 but taking into account only 100 realizations and $E_\mathrm{b}/N_0 = 10$ dB we depict the computational time required for each algorithm in Table I. The ILSP algorithm requires the smallest amount of time, because additional iterations will not be computed when the symbol matrix does not have full rank. The RLSP algorithm requires the longest time as it performs iterations and computes a pseudo-inverse of the channel matrix. The RLSE algorithm has the smallest computational complexity and requires a moderate amount of time.

Algorithm	ILSP	ILSE	RLSP	RLSE
Total Time [s]	2.815	6.910	7.962	4.810

TABLE I: Computational time required for each algorithm.

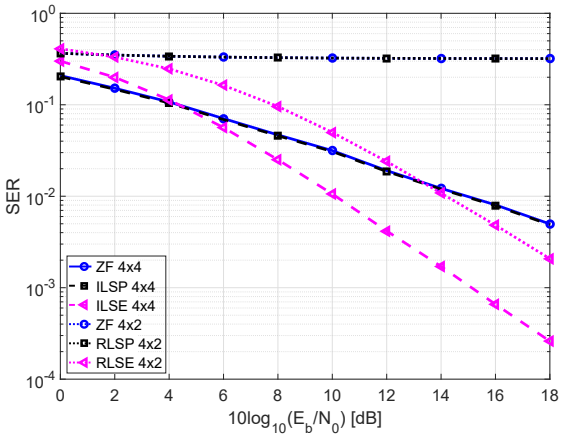

Fig. 4: SER for an OFDM system, $\Delta F = 4, N = 512, K = 2$ and $M_T \times M_R$ antennas depicted in the legend.

Furthermore, in Fig. 4 we show the SER as a function of E_b/N_0 in dB for MIMO OFDM systems of dimension 4×4 and 4×2, respectively. Here, we compare only the recursive algorithms with respect to the ZF receiver, as they are less complex and have a comparable performance as the respective iterative versions. The RLSE outperforms the rest of the algorithms and it is capable of estimating the data symbols even if $M_T > M_R$ without additional spreading as in [15].

V. Conclusion

In this paper, we have presented a tensor model for a MIMO-OFDM systems based on the generalized contraction operator. The derivation of this model facilitates the design of several types of receivers based on iterative and recursive LS algorithms. We have compared these algorithms with and without enumeration with the traditional ZF receiver. ILSP and RLSP show a similar performance as the ZF algorithm. The other two algorithms, ISLE and RLSE based on enumeration, outperform the rest of the algorithms at the cost of increased complexity. Both recursive algorithms, RLSE and RLSP have less computational complexity as compared to their iterative versions. The RLSE algorithm does not perform matrix inversion, being suitable for any configuration setup. It is capable of estimating the data symbols even for $M_T \geq M_R$ without additional spreading. In the future, recursive algorithms can be used to exploit the correlation of the channel tensor, especially in time varying scenarios. Moreover, multiple unfoldings can be exploited sequentially to capture the tensor structure in the receiver design. Furthermore, the system can be modified such that only specific codewords leading to rank M_T data matrices are used. This transmit strategy would guarantee that each symbol matrix is invertible to improve the channel estimates. Finally, it is worth mentioning that our generalized tensor contraction formalism presented here is very general and can be extended to any other multicarrier system, such as GFDM (Generalizes Frequency Devision Multiplexing) or FBMC (Filter Bank Multi-Carrier) leading to tensor based improvement of these multicarrier systems.

REFERENCES

[1] T. Hwang, C. Yang, G. Wu, S. Li, and G. Ye Li, "OFDM and Its Wireless Applications: A Survey," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 1673–1694, 2009.

[2] B. Farhang-Boroujeny, "OFDM versus filter bank multicarrier," *IEEE Signal Processing Magazine*, vol. 28, no. 3, pp. 92 – 112, 2011.

[3] M. Speth, S. A. Fechtel, G. Fock, and H. Meyr, "Optimum receiver design for wireless broad-band systems using OFDM. I," *IEEE Transactions on Communications*, vol. 47, no. 11, pp. 1668–1677, 1999.

[4] I. Barhumi, G. Leus, and M. Moonen, "Optimal training design for MIMO OFDM systems in mobile wireless channels," *IEEE Trans. Signal Process.*, vol. 5, pp. 1615–1624, 2003.

[5] D. Hu, L. Yang, Y. Shi, and L. He, "Optimal pilot sequence design for channel estimation in MIMO OFDM systems," *IEEE Communications Letters*, vol. 10, pp. 1–3, 2006.

[6] A. Cichocki, D. Mandic, A. Phan, C. Caiafa, G. Zhou, Q. Zhao, and L. de Lathauwer, "Tensor decompositions for signal processing applications: From two-way to multiway," *IEEE Signal Processing Magazine*, vol. 32, pp. 145–163, 2015.

[7] T. Kolda and B. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, pp. 455–500, 2009.

[8] A. Cichocki, "Era of big data processing: A new approach via tensor networks and tensor decompositions," *arXiv:1403.2048 [cs.ET]*, 2014.

[9] A. L. F. de Almeida, G. Favier, and L. R. Ximenes, "Space-time-frequency (STF) MIMO communication systems with blind receiver based on a generalized PARATUCK2 model," *IEEE Trans. Signal Process.*, vol. 61, no. 8, pp. 1895–1909, 2013.

[10] A. L. F. de Almeida and G. Favier, "Unified tensor model for space-frequency spreading-multiplexing (SFSM) MIMO communication systems," *EURASIP Journal on Advances in Signal Processing*, vol. 48, 2013.

[11] K. Liu, J. P. C. L. da Costa, H. C. So, and A. L. F. de Almeida, "Semi-blind receivers for joint symbol and channel estimation in space-time-frequency MIMO-OFDM systems," *in IEEE Trans. Signal Process.*, vol. 61, no. 21, pp. 5444–5457, 2013.

[12] G. Favier and A. L. F. de Almeida, "Tensor space-time-frequency coding with semi-blind receivers for MIMO wireless communication systems," *IEEE Trans. Signal Process.*, vol. 62, no. 22, pp. 5987–6002, 2014.

[13] M. Haardt, F. Roemer, and G. Del Galdo, "Higher-order SVD based subspace estimation to improve the parameter estimation accuracy in multi-dimensional harmonic retrieval problems," *IEEE Trans. Signal Process.*, vol. 56, pp. 3198–3213, 2008.

[14] B. Sokal, A. L. F. de Almeida, and M. Haardt, "Rank-One Tensor Modeling Approach to Joint Channel and Symbol Estimation in Two-Hop MIMO Relaying Systems," *in Proc. XXXV Simpsio Brasileiro de Telecomunicacoes e Prosessamento de Sinais (SBrT 2017)*, 2017.

[15] K. Naskovska, M. Haardt, and A. L. F. de Almeida, "Generalized tensor contraction with application to Khatri-Rao coded MIMO OFDM systems," *in Proc. IEEE 7th Int. Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pp. 286 – 290, 2017.

[16] S. Talwar, M. Viberg, and A. Paulraj, "Blind estimation of multiple co-channel digital signals using an antenna array," *IEEE Signal Processing Letters*, vol. 1, no. 2, pp. 29–31, 1994.

[17] ——, "Blind separation of synchronous co-channel digital signals using an antenna array. I. Algorithms," *IEEE Trans. Signal Process.*, vol. 44, no. 5, pp. 1184–1197, 1996.

[18] F. Roemer, C. Schroeter, and M. Haardt, "A semi-algebraic framework for approximate CP decompositions via joint matrix diagonalization and generalized unfoldings," *in Proc. of the 46th Asilomar Conference on Signals, Systems, and Computers*, pp. 2023–2027, November 2012.