Low-Rank Tensor MMSE Equalization

Lucas N. Ribeiro, André L. F. de Almeida, João C. M. Mota
Wireless Telecommunications Research Group (GTEL), Universidade Federal do Ceará, Fortaleza, Brazil
Emails: {nogueira, andre, mota}@gtel.ufc.br

***Abstract*—New-generation wireless communication systems will employ large-scale antenna arrays to satisfy the increasing capacity demand. This massive scenario brings new challenges to the channel equalization problem due to the increased signal processing complexity. We present a novel low-rank tensor equalizer to tackle the high computational demands of the classical linear approach. Specifically, we propose a method to design a canonical polyadic tensor filter to minimize the mean square error criterion. Our simulation results indicate that the proposed equalizer needs fewer calculations and is more robust to short training sequences than the benchmark.**

***Index Terms*—Tensor, Equalization, Beamforming, MIMO**

I. Introduction

Modern wireless communication systems rely on large-scale antenna arrays to enhance their performance [1], [2]. Such massive arrays yield high beamforming gain, improve interference suppression and ameliorate the spatial resolution capabilities of the system. However, the implementation of large-scale arrays raises some challenges, including computationally demanding signal processing, high energy consumption, among others. Tensor filtering has been investigated as a possible solution to the high computational complexity issue of large-scale systems [3]–[11].

In [3]–[5], we present beamforming methods for massive arrays considering a modest Kronecker separable system model. Therein, the high-dimensional beamforming vector is factorized into a Kronecker product of lower-dimension factors. Such a factorization allows us to optimize the beamformer for each low-dimensional factor, decreasing the number of calculations. We also observe that Kronecker beamformers may drastically reduce the number of calculations in the beamforming optimization with negligible signal recovery performance degradation. These filters, however, do not provide many degrees of freedom due to their rank-1 structure, limiting the performance and the applicability to more practical system models.

A general formulation of Kronecker filters is thus necessary to enhance their performance. In fact, Kronecker separable filters can be regarded as rank-1 tensor filters [12], [13]. One way to increase the filter's degrees of freedom consists of employing low-rank filters, i.e., we consider a finite sum of Kronecker-separable terms. In [6], [7], low-rank bilinear system identification methods are proposed to estimate acoustic impulse responses. It is shown that some sparse acoustic signatures nicely fits the low-rank bilinear model. In fact, sparsity is strongly linked to the low-rank system representation [14]. Some works [15], [16] also consider the identification of trilinear systems. These estimation methods exhibit better accuracy than their classical counterparts. In the context of wireless communications, [17] presents a low-rank bilinear filter for code division multiple access systems. The influence of the filter rank on the system performance is studied. It is shown that the rank parameter controls a complexity-equalization performance trade-off. Unfortunately, the analysis of [17] is restricted to the bilinear case and details on its computational complexity are lacking. Therefore the potential of multilinear filters is yet to be investigated.

The main contributions of the present paper can be summarized as: (i) We propose a novel low-rank multilinear equalizer for large-scale antenna array system based on the minimum mean square error (MMSE) filter. Our method extends those of [3]–[5] to deal with non-separable systems and it also generalizes those of [6], [16], [17] to the multidimensional case; (ii) We investigate the computational complexity and the equalization performance of the proposed method; (iii) With simulation results, we demonstrate the robustness of our method to short training sequences and its superior computational efficiency compared to the classical linear equalization approach.

A. Notation

$\boldsymbol{x}$ denotes vectors, $\boldsymbol{X}$ matrices and $\mathcal{X}$ tensors. $[\boldsymbol{X}]_{i,j}$ stands for the (i,j)-th entry of $\boldsymbol{X}$. The transpose, and the conjugate transpose (Hermitian) of $\boldsymbol{X}$ are denoted by $\boldsymbol{X}^{\mathsf{T}}$ and $\boldsymbol{X}^{\mathsf{H}}$, respectively. The $(M \times M)$-dimensional identity matrix is represented by $\boldsymbol{I}_M$ and the $(M \times N)$-dimensional null matrix by $\mathbf{0}_{M\times N}$. The ℓ_2 norm, the statistical expected value operator and the vectorization operator are respectively denoted as $\|\cdot\|_2$, $\mathbb{E}[\cdot]$, vec$(\cdot)$. The outer product, the Kronecker product, the n-mode product and the Big-O notation are referred to as $\circ$, $\otimes$, $\times_n$ and $O(\cdot)$, respectively.

II. System Model

Let us consider a multiple-input multiple-output (MIMO) wireless communication system consisting of U user equipment (UE) and a single base-station (BS). We assume that each UE is equipped with a single omni-directional antenna and the BS employs a uniform linear array (ULA) of N omni-directional antennas whose axis is parallel to the ground plane. The spacing between the array antennas is considered to be $d = \lambda/2$, where λ denotes the carrier wavelength. This half-wavelength ULA setup is considered for simplicity purposes, however, our model can be easily adapted to different kinds

This work is partially supported by the Brazilian National Council for Scientific and Technological Development - CNPq, CAPES/PROBRAL Proc. numbers 88887.144009/2017-00, 308317/2018-1, and FUNCAP.

of antenna geometry. We consider the uplink scenario, where UE u emits an independent and identically distributed (i.i.d.) discrete-time digitally-modulated symbol sequence $s_u[k]$ with zero mean and variance σ_s^2, where k denotes the symbol period for $u \in \{1, \dots, U\}$. From our assumptions, it follows that

$$\mathbb{E}\left[s_i[k-p]s_j^*[k-q]\right] = \begin{cases} 0, & i \neq j \text{ or } p \neq q \\ \sigma_s^2, & i = j \text{ and } p = q \end{cases}.$$

We assume a frequency-selective wireless channel with Q delay taps and L multi-paths. Therefore, the discrete-time representation of the received signal at the n-th BS antenna can be expressed as [18]

$$x_n[k] = \sum_{u=1}^{U}\sum_{q=0}^{Q-1}\sum_{\ell=0}^{L-1} \alpha_{u,\ell}\, g(qT-\tau_{u,\ell})\, a_n(\theta_{u,\ell})\, s_u[k-q] + b_n[k],$$

where $\alpha_{u,\ell}$ denotes the complex channel gain, $g(\cdot)$ the effective pulse-shaping waveform, $\tau_{u,\ell}$ the propagation delay, T the symbol period, $a_n(\theta_{u,\ell})$ the channel spatial response and $b_n[k]$ the additive white Gaussian noise (AWGN) component for $n \in \{1, \dots, N\}$. The channel gains are modeled as i.i.d. Gaussian random variables with zero mean and unit variance, and the AWGN components are Gaussian random variables with zero mean and σ_n^2 variance. Since the BS employs a half-wavelength ULA, the channel spatial response term is given by $a_n(\theta_{u,\ell}) = e^{-\jmath\pi(n-1)\cos(\theta_{u,\ell})}$, where $\theta_{u,\ell}$ stands for the direction of arrival of the ℓ-th path associated with UE u. We also define the signal to noise ratio (SNR) as $\text{SNR} = \sigma_s^2/\sigma_n^2$.

Let $\boldsymbol{x}[k] = [x_1[k], \dots, x_N[k]]^\mathsf{T}$ denote the received signal vector at BS. It can be written as:

$$\boldsymbol{x}[k] = \sum_{u=1}^{U} \boldsymbol{H}_u \boldsymbol{s}_u[k] + \boldsymbol{b}[k], \tag{1}$$

$$\boldsymbol{s}_u[k] = [s_u[k], \dots, s_u[k-Q+1]]^\mathsf{T},$$

$$\boldsymbol{b}[k] = [b_1[k], \dots, b_N[k]]^\mathsf{T},$$

where

$$\boldsymbol{H}_u = \sum_{\ell=1}^{L} \alpha_{u,\ell}\, \boldsymbol{a}(\theta_{u,\ell})\, \boldsymbol{g}(\tau_{u,\ell})^\mathsf{T} \in \mathbb{C}^{N\times Q}, \tag{2}$$

$$\boldsymbol{a}(\theta_{u,\ell}) = \left[1, \dots, e^{-\jmath\pi(N-1)\cos(\theta_{u,\ell})}\right]^\mathsf{T} \in \mathbb{C}^N,$$

$$\boldsymbol{g}(\tau_{u,\ell}) = [g(-\tau_{u,\ell}), \dots, g((Q-1)T-\tau_{u,\ell})]^\mathsf{T} \in \mathbb{C}^Q$$

denote the BS-UE u uplink channel matrix, the array steering vector (spatial response), and the effective pulse-shaping vector (temporal response), respectively. Model (1) assumes that the channel is block-fading, i.e., $\boldsymbol{H}_u$ remains constant over a frame of K symbol periods.

Let us consider that the BS wishes to extract from $\boldsymbol{x}[k]$ the symbols of UE u while regarding the signals from the other $(U-1)$ users as interference. To emphasize this scenario, we rewrite (1) as

$$\boldsymbol{x}[k] = \boldsymbol{H}_u \boldsymbol{s}_u[k] + \sum_{j\neq u}^{U} \boldsymbol{H}_j \boldsymbol{s}_j[k] + \boldsymbol{b}[k] \tag{3}$$

where the first term corresponds to the desired signal and the other terms to interference and noise. The covariance matrix of (3) is defined as $\boldsymbol{R}_{xx} = \mathbb{E}\left[\boldsymbol{x}[k]\boldsymbol{x}^\mathsf{H}[k]\right]$. From our assumptions, the covariance matrix can be written as $\boldsymbol{R}_{xx} = \boldsymbol{R}_{dd} + \boldsymbol{R}_{ii} + \boldsymbol{R}_{bb}$, where $\boldsymbol{R}_{dd} = \boldsymbol{H}_u \boldsymbol{R}_{ss} \boldsymbol{H}_u^\mathsf{H}$, $\boldsymbol{R}_{ii} = \sum_{j\neq u}^{U} \boldsymbol{H}_j \boldsymbol{R}_{ss} \boldsymbol{H}_j^\mathsf{H}$, $\boldsymbol{R}_{ss} = \sigma_s^2 \boldsymbol{I}_Q$, $\boldsymbol{R}_{bb} = \sigma_n^2 \boldsymbol{I}_N$.

III. Equalization Methods

We consider an equalizer filter to extract the desired signal from the observed signals and to compensate for channel distortions, multi-user interference, and noise. Let $\boldsymbol{w} \in \mathbb{C}^N$ represent the equalizer weights vector. The filter output can thus be written as $y[k] = \boldsymbol{w}^\mathsf{H}\boldsymbol{x}[k]$. We are interested in designing the equalizer weights to minimize the difference between filter output and the desired symbol sequence (lagged by some δ due to the propagation delay). We consider the mean square error (MSE) as an optimization metric, therefore the equalizer design problem can be stated as

$$\min_{\boldsymbol{w}} \mathbb{E}\left[|s_u[k-\delta] - y[k]|^2\right]. \tag{4}$$

In the following, we recall the classical MMSE filter, which is the optimal solution to (4). We discuss some of its properties and drawbacks. Next, we propose a novel equalizer design method based on tensor algebra that addresses some issues of the classical MMSE equalizer.

A. Linear MMSE Filter

Let $J(\boldsymbol{w}) = \mathbb{E}\left[|s_u[k-\delta] - y[k]|^2\right]$ denote the MSE objective function. It can be rewritten as

$$J(\boldsymbol{w}) = \sigma_s^2 - \boldsymbol{p}^\mathsf{H}\boldsymbol{w} - \boldsymbol{w}^\mathsf{H}\boldsymbol{p} + \boldsymbol{w}^\mathsf{H}\boldsymbol{R}_{xx}\boldsymbol{w}, \tag{5}$$

where $\boldsymbol{p} = \mathbb{E}\left[\boldsymbol{x}[k]s_u^*[k-\delta]\right] = \boldsymbol{H}_u\boldsymbol{R}_{ss}\boldsymbol{e}_\delta$ is the cross-covariance vector between the elements of $\boldsymbol{x}[k]$ and $s_u[k-\delta]$, and $\boldsymbol{e}_\delta = [0, \dots, 1, \dots, 0]^\mathsf{T}$ is a Q-dimensional vector with 1 at its δ-th entry and 0 elsewhere. Since $J(\boldsymbol{w})$ is convex, its global minimizer can be found by solving $\nabla J(\boldsymbol{w}) = \frac{\partial J}{\partial \boldsymbol{w}^*} = \boldsymbol{0}_{N\times 1}$. The MMSE filter is then given by

$$\boldsymbol{w}_{\text{MMSE}} = \boldsymbol{R}_{xx}^{-1}\boldsymbol{p}. \tag{6}$$

The minimum MSE is obtained by substituting (6) into (5):

$$\begin{aligned} J(\boldsymbol{w}_{\text{MMSE}}) &= \sigma_s^2 - \boldsymbol{p}^\mathsf{H}\boldsymbol{R}_{xx}^{-1}\boldsymbol{p} \\ &= \sigma_s^2 - \boldsymbol{e}_\delta^\mathsf{T}\boldsymbol{R}_{ss}\boldsymbol{H}_u^\mathsf{H}\boldsymbol{R}_{xx}^{-1}\boldsymbol{H}_u\boldsymbol{R}_{ss}\boldsymbol{e}_\delta. \end{aligned} \tag{7}$$

Equation (7) reveals that the choice of δ determines the minimum of the objective function. Therefore, to minimize (7) with respect to δ, one has to simply select δ as the index of the largest diagonal element of $\boldsymbol{R}_{ss}\boldsymbol{H}_u^\mathsf{H}\boldsymbol{R}_{xx}^{-1}\boldsymbol{H}_u\boldsymbol{R}_{ss}$.

The *a priori* knowledge of $\boldsymbol{R}_{xx}$ and $\boldsymbol{p}$ is hardly practical, and, thus, the statistics need to be estimated. In this case, one can consider a K-length training symbol sequence. Define $\boldsymbol{X} = [\boldsymbol{x}[0], \dots, \boldsymbol{x}[K-1]] \in \mathbb{C}^{N\times K}$ and $\boldsymbol{s}_u = [s_u[-\delta], \dots, s_u[K-1-\delta]]^\mathsf{T} \in \mathbb{C}^K$. The sample estimates are then given by

$$\boldsymbol{R}_{xx} \approx \frac{1}{K}\sum_{k=0}^{K-1} \boldsymbol{x}[k]\boldsymbol{x}^\mathsf{H}[k] = \frac{1}{K}\boldsymbol{X}\boldsymbol{X}^\mathsf{H}, \tag{8}$$

$$\boldsymbol{p} \approx \frac{1}{K}\sum_{k=0}^{K-1} \boldsymbol{x}[k]s_u^*[k-\delta] = \frac{1}{K}\boldsymbol{X}\boldsymbol{s}_u^*. \tag{9}$$

Unfortunately, the MMSE filter faces some issues when sample estimates are considered in the large-scale scenario. Long training symbol sequences are necessary to obtain sufficiently accurate statistics due to the large dimension of $\boldsymbol{R}_{xx}$. Moreover, the number of computations in (6) may be exceedingly large. To be more specific, one needs to carry out N^2K+NK products to estimate (8) and (9), and $O(N^3)+N^2$ products to calculate (6), which yields a total of

$$P_{\text{MMSE}}(N,K) = N^2K + NK + O(N^3) + N^2 \tag{10}$$

products. For large arrays, the cubic and quadratic terms in (10) yields a substantial number of products.

B. Low-Rank Tensor MMSE (LR-TMMSE) Filter

In this section, we introduce a tensor equalizer which solves the computational complexity issues of the classical MMSE filter. First, assume the number N of antennas at the BS can be factorized as $N = \prod_{d=1}^{D} N_d$. Then, we reshape the column vector $\boldsymbol{w}$ into a D-th order tensor $\mathcal{W} \in \mathbb{C}^{N_1\times\cdots\times N_D}$. The entry w_n of $\boldsymbol{w}$ relates to the entry $w_{n_1,\ldots,n_D}$ of $\mathcal{W}$ as $n = n_1 + (n_2-1)N_1 + \cdots + (n_D-1)\prod_{m=1}^{D-1} N_m$ for $n \in \{1,\ldots,N\}$ and $n_d \in \{1,\ldots,N_d\}$, $\forall d \in \{1,\ldots,D\}$. With such a reshaping operation, the equalizer output is rewritten as

$$\begin{aligned} y[k] &= \boldsymbol{w}^{\mathsf{H}}\boldsymbol{x}[k] = \sum_{n=1}^{N} w_n^* x_n[k] \\ &= \sum_{n_1,\ldots,n_D=1}^{N_1,\ldots,N_D} w^*_{n_1,\ldots,n_D} x_{n_1,\ldots,n_D}[k], \end{aligned}$$

where $\mathcal{X}[k] \in \mathbb{C}^{N_1\times\cdots\times N_D}$ denotes the D-dimensional reshape of $\boldsymbol{x}[k]$.

In previous works [3], [4], we consider rank-1 separable tensor filters, i.e., $\mathcal{W}$ is written as an outer product of vectors. However, such a structure is too strict for some applications. For example, the channel matrix (2) cannot be separated as a Kronecker product, and, thus, rank-1 tensor filters would exhibit poor equalization performance. To overcome this limitation, let us decompose $\mathcal{W}$ as a sum of R rank-1 terms:

$$\mathcal{W} = \sum_{r=1}^{R} \boldsymbol{w}_{1,r} \circ \ldots \circ \boldsymbol{w}_{D,r}. \tag{11}$$

where $\boldsymbol{w}_{d,r} \in \mathbb{C}^{N_d\times 1}$ for $d \in \{1,\ldots,D\}$, $r \in \{1,\ldots,R\}$ and R denotes the filter rank. Note that (11) is known as the canonical polyadic (CP) decomposition in tensor literature [12], [13]. It is interesting to mention the relationship between the outer and Kronecker product notations. Specifically, by vectorizing (11), we obtain $\text{vec}(\mathcal{W}) = \sum_{r=1}^{R} \boldsymbol{w}_{D,r} \otimes \ldots \otimes \boldsymbol{w}_{1,r}$ [12]. The extra degrees of freedom brought by the R separable components allow the CP tensor to better equalize non-separable systems such as (2). Note that R is a parameter to be chosen by the filter designer. As it grows, the filter has more degrees of freedom, but also more parameters to estimate. Therefore a judicious choice of R which balances the performance-complexity trade-off is preferred.

Algorithm 1 Low-Rank Tensor MMSE Equalizer

Require: Received signals $\boldsymbol{X} \in \mathbb{C}^{N\times K}$, training sequence $\boldsymbol{s}_u \in \mathbb{C}^{K}$, filter rank R, filter order D, filter dimensions N_d for $d \in \{1,\ldots,D\}$.

Initialize $\boldsymbol{w}_{d,r}$ as $[1,0,\ldots,0]^{\mathsf{T}}$, $d \in \{1,\ldots D\}$, $r \in \{1,\ldots,R\}$
repeat
  for $d = 1,\ldots,D$ **do**
    Build $\boldsymbol{U}_{d,r}$ for $r = 1,\ldots,R$ by (18)
    Form $\boldsymbol{U}_d = \left[\boldsymbol{U}_{d,1}^{\mathsf{T}},\ldots,\boldsymbol{U}_{d,R}^{\mathsf{T}}\right]^{\mathsf{T}}$
    Estimate $\boldsymbol{R}_{u_d u_d}$ and $\boldsymbol{p}_{u_d}$ by (19)
    Update $\boldsymbol{w}_d$ by (15)
  end for
until convergence criterion triggers
Form tensor filter $\mathcal{W}$ using (11) and (13)
$\boldsymbol{w} \leftarrow \text{vec}(\mathcal{W})$

Assuming structure (11), the filter coefficients can be written as

$$w_{n_1,\ldots,n_D} = \sum_{r=1}^{R}\prod_{d=1}^{D} [\boldsymbol{w}_{d,r}]_{n_d},$$

which allows us to recast the equalizer output $y[k]$ as follows

$$\begin{aligned} y[k] &= \sum_{n_1,\ldots,n_D=1}^{N_1,\ldots,N_D} w^*_{n_1,\ldots,n_D} x_{n_1,\ldots,n_D}[k] \\ &= \sum_{n_1,\ldots,n_D=1}^{N_1,\ldots,N_D} \left(\sum_{r=1}^{R} [\boldsymbol{w}_{1,r}]^*_{n_1} \ldots [\boldsymbol{w}_{D,r}]^*_{n_D}\right) x_{n_1,\ldots,n_D}[k]. \end{aligned}$$

By isolating $\boldsymbol{w}_{d,r}$ from the other $(D-1)$ factors, we get:

$$\begin{aligned} y[k] &= \sum_{r=1}^{R}\sum_{n_d=1}^{N_d} [\boldsymbol{w}_{d,r}]^*_{n_d} \left(\sum_{n_q=1}^{N_q}\prod_{q\neq d}^{D} [\boldsymbol{w}_{q,r}]^*_{n_q} x_{n_1,\ldots,n_D}[k]\right) \\ &= \sum_{r=1}^{R}\sum_{n_d=1}^{N_d} [\boldsymbol{w}_{d,r}]^*_{n_d} [\boldsymbol{u}_{d,r}[k]]_{n_d} = \boldsymbol{w}_d^{\mathsf{H}} \boldsymbol{u}_d[k] \end{aligned} \tag{12}$$

where we define

$$\begin{gathered} \boldsymbol{u}_{d,r}[k] = \boldsymbol{X}_{(d)}[k]\bar{\boldsymbol{w}}^*_{d,r} \in \mathbb{C}^{N_d}, \\ \boldsymbol{u}_d[k] = \left[\boldsymbol{u}_{d,1}^{\mathsf{T}}[k],\ldots,\boldsymbol{u}_{d,R}^{\mathsf{T}}[k]\right]^{\mathsf{T}} \in \mathbb{C}^{RN_d}, \\ \boldsymbol{w}_d = \left[\boldsymbol{w}_{d,1}^{\mathsf{T}},\ldots,\boldsymbol{w}_{d,R}^{\mathsf{T}}\right]^{\mathsf{T}} \in \mathbb{C}^{RN_d}, \\ \bar{\boldsymbol{w}}_{d,r} = \bigotimes_{q\neq d}^{D} \boldsymbol{w}_{q,r} \in \mathbb{C}^{\bar{N}_d\times 1},\ \bar{N}_d = \prod_{q\neq d}^{D} N_q. \end{gathered} \tag{13}$$

Matrix $\boldsymbol{X}_{(d)}[k] \in \mathbb{C}^{N_d\times\bar{N}_d}$ denotes the d-mode matrix unfolding of $\mathcal{X}[k]$. The tensor element $(n_1,\ldots,n_D)$ maps to the $\boldsymbol{X}_{(d)}[k]$ element (n_d, j) as $j = 1 + \sum_{j\neq d}^{D}(n_j - 1)\prod_{m\neq d}^{j-1} N_m$ [13]. For more information on tensor notation, the reader is kindly referred to [12], [13]. Equation (12) explicits the multilinear property of our tensor filter since $y[k]$ is linear with respect to $\boldsymbol{w}_d$ given that $\bar{\boldsymbol{w}}_{d,r}$ is fixed for all $d \in \{1,\ldots,D\}$.

The multilinear filter output (12) allows us to reformulate the linear equalization problem (4) as

$$\min_{\boldsymbol{w}_d} \mathbb{E}\left[|s_u[k-\delta] - \boldsymbol{w}_d^{\mathsf{H}}\boldsymbol{u}_d[k]|^2\right], \quad d \in \{1,\dots,D\}. \quad (14)$$

There are several ways to solve (14). We propose an alternating minimization approach in which we solve for each $\boldsymbol{w}_d$ sequentially until a convergence criterion is satisfied. For a given d, (14) can be seen as a low-dimensional MMSE problem. Hence, we have that

$$\boldsymbol{w}_{d,\text{MMSE}} = \boldsymbol{R}_{u_d,u_d}^{-1}\boldsymbol{p}_{u_d} \in \mathbb{C}^{RN_d}, \quad (15)$$

$$\boldsymbol{R}_{u_d,u_d} = \mathbb{E}\left[\boldsymbol{u}_d[k]\boldsymbol{u}_d^{\mathsf{H}}[k]\right] \in \mathbb{C}^{RN_d \times RN_d}, \quad (16)$$

$$\boldsymbol{p}_{u_d} = \mathbb{E}\left[\boldsymbol{u}_d[k]s_u^*[k-\delta]\right] \in \mathbb{C}^{RN_d} \quad (17)$$

for $d \in \{1,\dots,D\}$.

Let us calculate the sample estimates of $\boldsymbol{R}_{u_d u_d}$ and $\boldsymbol{p}_{u_d}$. To this end, define $\mathcal{X} \in \mathbb{C}^{N_1 \times \cdots \times N_D \times K}$ as the tensor reshaping of $\boldsymbol{X}$. Now, consider the following tensor-vector product [13] evaluated at the tensor modes $\mathbb{J}_d = \{j = 1,\dots,D \,|\, j \neq d\}$

$$\boldsymbol{U}_{d,r} = \mathcal{X} \bigtimes_{j \in \mathbb{J}_d} \boldsymbol{w}_{j,r}^{\mathsf{H}} \in \mathbb{C}^{N_d \times K}. \quad (18)$$

The elements of (18) are given by

$$[\boldsymbol{U}_{d,r}]_{n_d,k} = \sum_{n_1=1}^{N_1} \cdots \sum_{n_{d-1}=1}^{N_{d-1}} \sum_{n_{d+1}=1}^{N_{d+1}} \cdots \sum_{n_D=1}^{N_D} [\mathcal{X}]_{n_1,\dots,n_D,k} \prod_{j \in \mathbb{J}_d} [\boldsymbol{w}_{d,r}]_{n_j}^*.$$

Define $\boldsymbol{U}_d = \left[\boldsymbol{U}_{d,1}^{\mathsf{T}},\dots,\boldsymbol{U}_{d,R}^{\mathsf{T}}\right]^{\mathsf{T}} \in \mathbb{C}^{RN_d \times K}$. The statistics (16) and (17) may be estimated as

$$\boldsymbol{R}_{u_d u_d} \approx \frac{1}{K}\boldsymbol{U}_d\boldsymbol{U}_d^{\mathsf{H}}, \quad \boldsymbol{p}_{u_d} \approx \frac{1}{K}\boldsymbol{U}_d\boldsymbol{s}_u^* \quad (19)$$

for $d \in \{1,\dots,D\}$. Henceforth, this method is referred to as low-rank tensor MMSE (LR-TMMSE) filter and it is summarized in Algorithm 1.

The LR-TMMSE equalizer is an iterative method. Let us assume that it converges within I iterations. Each iteration carries out $R(D-1)NK$ products to compute $\boldsymbol{U}_d$, $N_d^2K + N_dK$ to estimate the statistics and $O(N_d^3) + N_d^2$ for each $\boldsymbol{w}_d$. Therefore, LR-TMMSE carries out a total of

$$P_{\text{LR-TMMSE}}(\{N_d\}, D, I, K) = I\left[\sum_{d=1}^{D} R(D-1)NK + N_d^2K + N_dK + O(N_d^3) + N_d\right] \quad (20)$$

products.

IV. Simulation Results

In this section, we present simulation results conducted to analyze the performance of the proposed LR-TMMSE equalizer. In all simulations, the sample-based MMSE is considered as the benchmark equalizer. We consider two figures of merit: (i) number of products to obtain the filter weights vector according to (10) and (20), (ii) the signal to noise and interference ratio (SINR) after equalization. For a given equalizer $\boldsymbol{w}$,

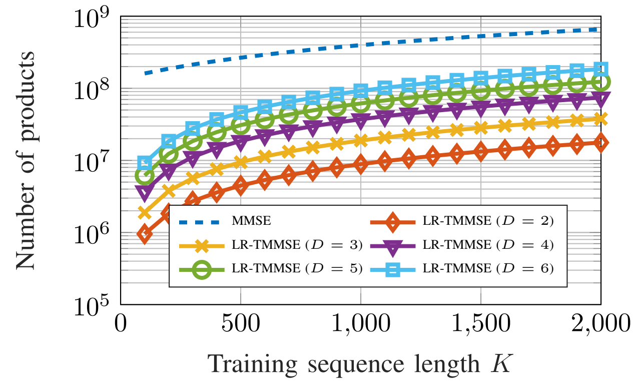

Fig. 1. $N = 512$ antennas, $I = 2$ iterations, filter rank $R = 3$.

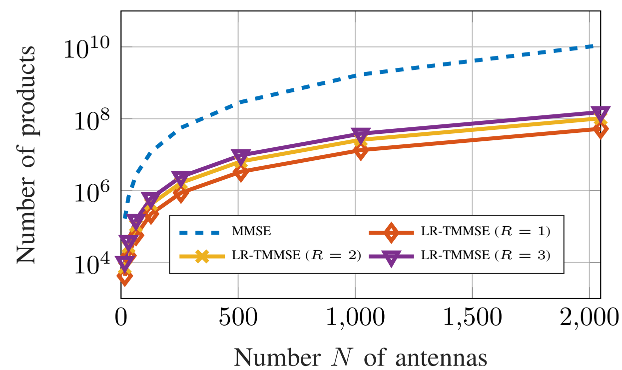

Fig. 2. $K = 600$ symbols, $I = 2$ iterations, filter order $D = 3$.

we have $\text{SINR}(\boldsymbol{w}) = (\boldsymbol{w}^{\mathsf{H}}\boldsymbol{R}_{xx}\boldsymbol{w})/(\boldsymbol{w}^{\mathsf{H}}(\boldsymbol{R}_{ii} + \boldsymbol{R}_{bb})\boldsymbol{w})$. The figures presented in this section were obtained by averaging the results from 1000 independent experiments. Each experiment consists of generating a K-length QPSK-modulated symbol sequence for all U UE, building a channel realization, forming the observed signals and finally applying the equalizers. The desired signal delay δ is optimized for all equalizers as explained in Sec. III-A. We consider the following parameter setup in our simulations: $U = 4$ users, $\sigma_s^2 = 1$, $L = 5$ channel paths, the directions of arrival $\theta_{u,\ell}$ are drawn from a random variable uniformly distributed in $[-90°, 90°]$, the sinc function is set as the effective pulse-shaping waveform $g(t)$. LR-TMMSE achieves convergence when $\|\boldsymbol{w}_{i+1} - \boldsymbol{w}_i\|_2^2 < \epsilon$, where i denotes the iteration number and ϵ is a small positive threshold. We set $\epsilon = 0.1$ and, according to a preliminary simulation, the algorithm typically converges within $I = 2$ iterations.

The computational complexity of LR-TMMSE is studied in Figures 1 and 2. In the former figure, we investigate the influence of the training sequence length K on the number of products for tensor equalizers with different order D. We fix the number of antennas to $N = 512$ and we consider D-order tensor filters such that their dimensions $\{N_d\}$ satisfy $\prod_{d=1}^{D} N_d = 512$. Figure 1 reveals that LR-TMMSE computes fewer products than the benchmark for the considered parameters. Moreover, it shows that the complexity of LR-TMMSE increases with D. Although the cubic term in (20) tends to become less important as D grows, we observe a significant

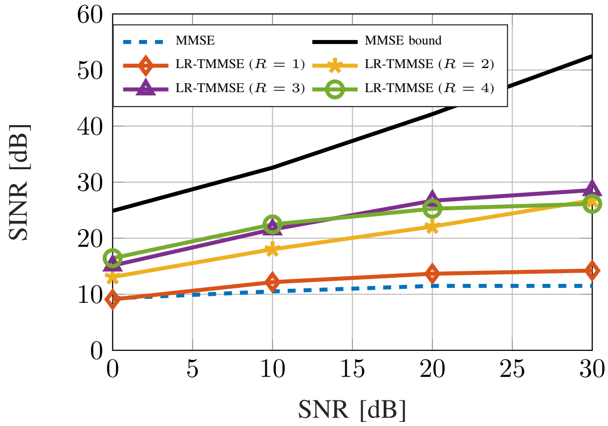

Fig. 3. $N = 512$ antennas, $K = 600$ symbols, filter order $D = 3$.

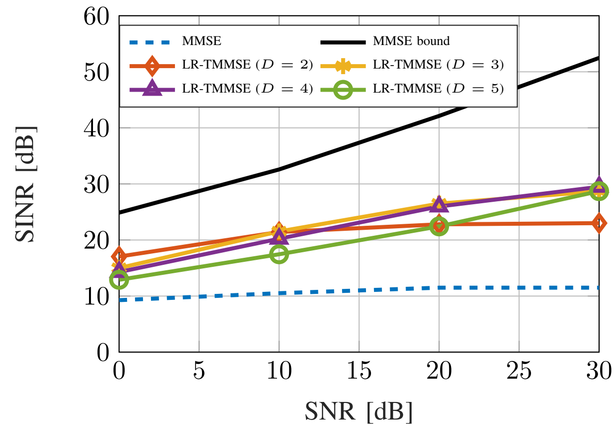

Fig. 4. $N = 512$ antennas, $K = 600$ symbols, filter rank $R = 3$.

overhead associated with the computation of the $\boldsymbol{U}_d$ matrices. We plot the computational complexity as a function of the BS array size N for different ranks R in Figure 2. We observe that R does not influence much on the complexity. Besides, LR-TMMSE is more computationally efficient than the benchmark even for very large array sizes.

We investigate the SINR performance for different ranks R and orders D for only $K = 600$ training symbols in Figures 3 and 4. We consider the theoretical MMSE as an SINR upper bound. Figure 3 shows that $R = 1$ performs poorly, as expected. This is because the signal model (1) is not separable. Therefore, it is necessary to increase the filter rank R to better equalize the system. However, we note that SINR is rather similar for $R = 3$ and $R = 4$, indicating that the LR-TMMSE performance is bounded with respect to this parameter. We plot the SINR for $R = 3$ and different D in Figure 4. This result suggests that the tensor order has a limited effect on the SINR. Still, the difference between $D = 2$ and $D \in \{3, 4, 5\}$ at 30 dB SNR is 5 dB.

We finally analyze the effects of the training sequence length K on the SINR for $N = 512$ antennas in Figures 5 and 6. LR-TMMSE yields its worst performance when $R = 1$. In this case, even the benchmark performs better for $K \geq 700$. However, when we increase the filter rank, we notice an SINR gain, which becomes bounded at $R = 4$, confirming the discussion in the previous paragraph. In Figure 6, we notice that SINR slightly varies for different D. Yet, LR-TMMSE provides the worst SINR for short sequences with $D = 2$.

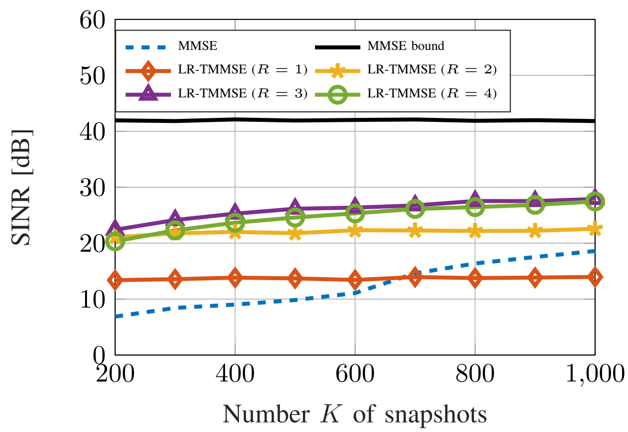

Fig. 5. $N = 512$ antennas, SNR = 20 dB, filter order $D = 3$.

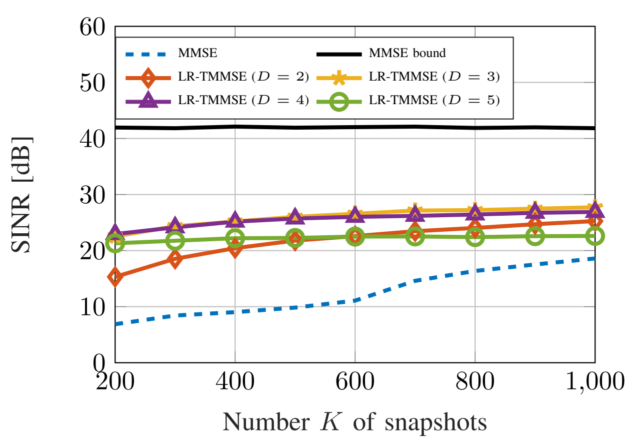

Fig. 6. $N = 512$ antennas, SNR = 20 dB, filter rank $R = 3$.

V. Conclusion

In this paper, we introduced a novel low-rank tensor MMSE equalizer based on the MMSE filter for non-separable (low-rank) MIMO channels. Our simulation results indicate that the proposed method is more robust to short training symbol sequences and more computationally efficient than the benchmark (linear MMSE) solution. The obtained results also show that the tensor equalizer performance increases with the filter rank up to a certain level. The tensor filter order has shown to be less relevant to the equalizer SINR, although third-order filters yield the best performance.

REFERENCES

[1] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 186–195, Feb. 2014.

[2] S. Schwarz and M. Rupp, "Society in motion: challenges for LTE and beyond mobile communications," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 76–83, May 2016.

[3] L. N. Ribeiro, A. L. F. De Almeida, and J. C. M. Mota, "Tensor beamforming for multilinear translation invariant arrays," in *Proc. 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 2966–2970.

[4] L. N. Ribeiro, A. L. de Almeida, J. A. Nossek, and J. C. M. Mota, "Low-complexity separable beamformers for massive antenna array systems," *IET Signal Processing*, 2019.

[5] L. N. Ribeiro, A. L. de Almeida, and J. C. M. Mota, "Separable linearly constrained minimum variance beamformers," *Signal Processing*, vol. 158, pp. 15–25, 2019.

[6] C. Paleologu, J. Benesty, and S. Ciochină, "Linear system identification based on a Kronecker product decomposition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1793–1808, Oct 2018.

[7] C. Elisei-Iliescu, C. Stanciu, C. Paleologu, J. Benesty, C. Anghel, and S. Ciochină, "Efficient recursive least-squares algorithms for the identification of bilinear forms," *Digital Signal Processing*, vol. 83, pp. 280–296, 2018.

[8] I. Cohen, J. Benesty, and J. Chen, "Differential Kronecker product beamforming," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 5, pp. 892–902, May 2019.

[9] M. Rupp and S. Schwarz, "A tensor LMS algorithm," in *Proc. 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3347–3351.

[10] ——, "Gradient-based approaches to learn tensor products," in *Proc. 2015 23rd European Signal Processing Conference (EUSIPCO)*, pp. 2486–2490.

[11] L. Liu, J. Xie, L. Wang, Z. Zhang, and Y. Zhu, "Robust tensor beamforming for polarization sensitive arrays," *Multidimensional Systems and Signal Processing*, pp. 1–22, 2018.

[12] P. Comon, "Tensors: a brief introduction," *IEEE Signal Processing Magazine*, vol. 31, no. 3, pp. 44–53, 2014.

[13] T. Kolda and B. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, Aug. 2009.

[14] G. Wunder, H. Boche, T. Strohmer, and P. Jung, "Sparse signal processing concepts for efficient 5G system design," *IEEE Access*, vol. 3, pp. 195–208, 2015.

[15] L. N. Ribeiro, A. L. F. de Almeida, and J. C. M. Mota, "Identification of separable systems using trilinear filtering," in *Proc. 2015 IEEE 6th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 2015, pp. 189–192.

[16] L.-M. Dogariu, S. Ciochină, J. Benesty, and C. Paleologu, "System identification based on tensor decompositions: A trilinear approach," *Symmetry*, vol. 11, no. 4, p. 556, 2019.

[17] O. Filiz and A. Yener, "Rank constrained temporal-spatial matrix filters for CDMA systems," *IEEE Transactions on Wireless Communications*, vol. 3, no. 6, pp. 1974–1979, Nov. 2004.

[18] P. Almers, E. Bonek, A. Burr, N. Czink, M. Debbah, V. Degli-Esposti, H. Hofstetter, P. Kyösti, D. Laurenson, G. Matz, A. Molisch, C. Oestges, and H. Özcelik, "Survey of channel and radio propagation models for wireless MIMO systems," *EURASIP Journal on Wireless Communications and Networking*, vol. 2007, no. 1, p. 019070, Feb 2007.