Identification of Separable Systems Using Trilinear Filtering

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

***Abstract*—Linear filtering methods are well known and have been successfully applied in system identification and equalization problems. However, they become unpractical when the number of parameters to estimate is very large. The recently proposed assumption of system separability allows the development of computationally efficient alternatives to classic adaptive methods in this scenario. In this work, we show that system separability calls for multilinear system representation and filtering. Based on this parallel, the proposed filtering framework consists of a trilinear extension of the classical Wiener-Hopf (WH) solution that exploits the separability property to solve the supervised identification problem. Our numerical results shows the proposed algorithm can provide a better accuracy than the classical WH solution which ignores the multilinear system representation.**

I. INTRODUCTION

System identification is the problem of identifying parameters of an unknown system [1]. When the system input and output are available, supervised techniques such as the so-called Wiener-Hopf (WH) solution can be employed to identify the system response. Indeed, the WH solution is the minimum mean square error (MMSE) estimator for linear filters. However, it becomes inadequate for online system identification due to its relatively high computational cost. An alternative to this approach consists of using an adaptive filter. The coefficients of this device are updated according to an algorithm that minimizes the energy of the estimation error. The so-called least mean squares (LMS) algorithm is the canonical to adaptive filtering method due to its implementation simplicity and low complexity. Nevertheless, LMS-based algorithms suffer from slow convergence rate when the parameter space becomes too large [2].

Many ideas have been proposed to ameliorate the convergence rate of adaptive algorithms. For instance, step-size adaptation [3] and sparsity constraints [4] are known approaches for learning rate improvement. Recently, the authors in [2] proposed a low complexity LMS algorithm with fast learning rate, therein referred to as TensorLMS, which exploits the system separability assumption, meaning that the system impulse response vector can be decomposed as the Kronecker products of two vectors representing its components. Indeed, this assumption is plausible in Volterra systems [5] and array processing [6]. In [2], the existence and uniqueness of second-order separable systems was discussed.

This work was partially supported by CAPES, CNPq, and FUNCAP (Brazil).

Higher-order tensor filtering presents itself as the proper signal processing framework for exploiting multilinearly structured systems. Multilinear filtering was first introduced in the context of noise reduction in color images and multicomponent seismic data [7]. In this context, the higher-order tensor data is corrupted by a multidimensional noise and the original data is recovered by filtering the observed tensor data by matrix filters operating on each mode. Since reference signals were not available, the optima filters are obtained from subspace decomposition. In [8], a tensor (trilinear) filtering framework was proposed for equalization problems.

In this work, the multilinearity of separable systems is exploited for solving the identification problem using higher-order tensor modeling and filtering. More specifically, we assume that the overall system impulse response is third-order separable, i.e. it can be factored as the Kronecker product of three components. A tensor formalism is used to devise an algorithm for the identification of the system impulse response, leading to the trilinear extension of the WH solution. Furthermore, this tensor formalism provides proper notation and insight regarding the filtering operations. The proposed method employs an alternating minimization approach, whereas TensorLMS is based on the stochastic gradient descent method. According to our numerical simulations, the proposed algorithm overcomes the drawbacks of its classic counterpart and provides better system estimation quality.

Notation: Lowercase letters denote scalars, lowercase boldface letters denote vectors, uppercase boldface letters denote matrices and calligraphic letters denote higher-order tensors. The symbol $\otimes$ denotes the Kronecker product, $\diamond$ Khatri-Rao product, $\|\cdot\|_2^2$, $\times_n$ n-mode product, Euclidean norm, $\mathbb{E}[\cdot]$ statistical expectation, and $(\cdot)^\mathsf{T}$ transpose operator.

II. THE TRILINEAR FILTERING FRAMEWORK

Before presenting the trilinear filtering framework and its connection with the system identification problem, some useful tensor formalism is given for later use.

A. Tensor preliminaries

The $\{1,\dots,N\}$*-mode products* of $\mathcal{T}$ with N matrices $\{\mathbf{U}^{(n)}\}_{n=1}^N$ yield the tensor $\tilde{\mathcal{T}} = \mathcal{T} \times_1 \mathbf{U}^{(1)} \dots \times_N \mathbf{U}^{(N)} \in \mathbb{R}^{J_1\times\dots\times J_N}$ defined as [9]

$$[\tilde{\mathcal{T}}]_{j_1,\dots,j_N} = \sum_{i_1=1}^{I_1}\dots\sum_{i_N=1}^{I_N} x_{i_1,\dots,i_N} u_{j_1,i_1}^{(1)} \dots u_{j_N,i_N}^{(N)},$$

where $\mathbf{U}^{(n)} \in \mathbb{R}^{J_n \times I_n}$, $i_n \in \{1, \ldots, I_n\}$, and $j_n \in \{1, \ldots, J_n\}$, $n = 1, \ldots, N$. The n-mode unfolding of $\tilde{\mathcal{T}}$ is given by

$$\tilde{\mathbf{T}}_{(n)} = \mathbf{U}^{(n)} \mathbf{T}_{(n)} \mathbf{U}^{\otimes n^{\mathsf{T}}}, \quad (1)$$

where $\mathbf{T}_{(n)}$ denotes the n-mode unfolding of $\mathcal{T}$, and

$$\mathbf{U}^{\otimes n} = \mathbf{U}^{(N)} \otimes \ldots \otimes \mathbf{U}^{(n+1)} \otimes \mathbf{U}^{(n-1)} \otimes \ldots \otimes \mathbf{U}^{(1)}$$

denotes the Kronecker product of the matrices $\{\mathbf{U}^{(j)}\}_{j=1, j \neq n}^{N}$ in the decreasing order. Note that the $\{1, \ldots, N\}$-mode products of $\mathcal{T}$ with the N vectors $\{\mathbf{u}^{(n)}\}_{n=1}^{N}$ yields a scalar $t = \mathcal{T} \times_1 \mathbf{u}^{(1)^{\mathsf{T}}} \ldots \times_N \mathbf{u}^{(N)^{\mathsf{T}}}$ where $\mathbf{u}^{(n)} \in \mathbb{R}^{I_n \times 1}$, $n = 1, \ldots, N$.

Vector-Tensorization: Let us consider the linear map $\Theta : \mathbb{R}^{I_1 I_2 I_3} \to \mathbb{R}^{I_3 \times I_2 \times I_1}$, defined as

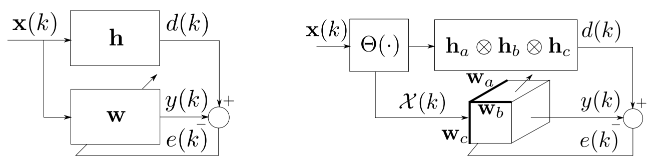
$$[\Theta(\mathbf{v})]_{i_3, i_2, i_1} = [\mathbf{v}]_{i_3 + I_3(i_2 - 1) + I_2 I_3 (i_1 - 1)}, \quad (2)$$

where $\mathbf{v} \in \mathbb{R}^{I_1 I_2 I_3}$ is the input vector, which can be partitioned into I_1 partitions of length $I_2 I_3$. These partitions can be further divided into I_2 subpartitions of length I_3. The input vector can be transformed into a third-order tensor $\mathcal{V} = \Theta(\mathbf{v}) \in \mathbb{R}^{I_3 \times I_2 \times I_1}$. The operation of transforming a vector (or matrix) into a tensor is referred to as "tensorization" [10], [11]. There are different forms of tensorization operations, and the specific transformation depends on the considered application. In our case, tensorization is deterministic and achieved by vector tri-partitioning, according to (2). The vector-tensorization is an isometric isomorphism for the l_2-norm of $\mathbf{v}$ on $\mathbb{R}^{I_1 I_2 I_3}$ and the Frobenius norm of $\mathcal{V}$ on $\mathbb{R}^{I_3 \times I_2 \times I_1}$.

B. Trilinear filtering

Consider an Mth order finite impulse response (FIR) filter with impulse response $\mathbf{z} \in \mathbb{R}^M$. The input regression vector and the output signal at instant k are represented by $\mathbf{x}(k) \in \mathbb{R}^M$ and $y(k) = \mathbf{z}^{\mathsf{T}} \mathbf{x}(k)$, respectively. Let us suppose this filter is third-order separable, i.e. it can be expressed as $\mathbf{z} = \mathbf{z}_a \otimes \mathbf{z}_b \otimes \mathbf{z}_c$, where $\mathbf{z}_a \in \mathbb{R}^{M_a}$, $\mathbf{z}_b \in \mathbb{R}^{M_b}$, and $\mathbf{z}_c \in \mathbb{R}^{M_c}$ are its component subfilters, with $M_a M_b M_c = M$. This corresponds to a rank-one third-order tensor decomposition problem [11]. In view of this, its existence and uniqueness in the least squares (LS) sense is guaranteed [12], [13]. Since there are no results ensuring the estimation error bound for higher-order tensor decompositions, devising the closed-form MMSE expression for a third-order tensor decomposition is a challenge.

In order to express the output signal $y(k) = \mathbf{z}^{\mathsf{T}} \mathbf{x}(k)$ as the outcome of a trilinear filtering, let us partition the input signal into $\mathbf{x}(k) = [\mathbf{x}_1(k), \mathbf{x}_2(k), \ldots, \mathbf{x}_{M_a}(k)]^{\mathsf{T}}$ where $\mathbf{x}_{m_a}(k) \in \mathbb{R}^{M_b M_c}$ for $m_a = 1, \ldots, M_a$. Now, let us further divide each partition $\mathbf{x}_{m_a}(k)$ into $\mathbf{x}_{m_a}(k) = [\mathbf{x}_{1,m_a}(k), \mathbf{x}_{2,m_a}(k), \ldots, \mathbf{x}_{M_b,m_a}(k)]^{\mathsf{T}}$, where $\mathbf{x}_{m_b,m_a}(k) = [x_{1,m_b,m_a}(k), x_{2,m_b,m_a}(k), \ldots, x_{M_c,m_b,m_a}(k)]^{\mathsf{T}} \in \mathbb{R}^{M_c}$ for $m_b = 1, \ldots, M_b$ and $x_{m_c,m_b,m_a}(k) \in \mathbb{R}$ is an element of the third-order tensor $\mathcal{X}(k) = \Theta[\mathbf{x}(k)] \in \mathbb{R}^{M_c \times M_b \times M_a}$. The output signal can be written as:



(a) System identification using classical adaptive filtering.

(b) System identification using trilinear filtering.

Fig. 1. Comparison between classical and trilinear WH approaches.

$$\begin{aligned} y(k) &= (\mathbf{z}_a \otimes \mathbf{z}_b \otimes \mathbf{z}_c)^{\mathsf{T}} \mathbf{x}(k) \\ &= \sum_{m_a=1}^{M_a} [\mathbf{z}_a]_{m_a} (\mathbf{z}_b \otimes \mathbf{z}_c)^{\mathsf{T}} \mathbf{x}_{m_a}(k) \\ &= \sum_{m_a=1}^{M_a} \sum_{m_b=1}^{M_b} [\mathbf{z}_a]_{m_a} [\mathbf{z}_b]_{m_b} \mathbf{z}_c^{\mathsf{T}} \mathbf{x}_{m_b,m_a}(k) \\ &= \sum_{m_a=1}^{M_a} \sum_{m_b=1}^{M_b} \sum_{m_c=1}^{M_c} [\mathbf{z}_a]_{m_a} [\mathbf{z}_b]_{m_b} [\mathbf{z}_c]_{m_c} x_{m_c,m_b,m_a}(k). \quad (3) \end{aligned}$$

Note that $y(k)$ is trilinear with respect to the elements of the subfilters as seen on (3). This operation is identified as the $\{1, 2, 3\}$-mode product (1):

$$\begin{aligned} y(k) &= \mathcal{X}(k) \times_1 \mathbf{z}_c^{\mathsf{T}} \times_2 \mathbf{z}_b^{\mathsf{T}} \times_3 \mathbf{z}_a^{\mathsf{T}}, && (4) \\ &= \mathbf{z}_c^{\mathsf{T}} \mathbf{u}_c(k) = \mathbf{z}_c^{T} \mathbf{X}_{(1)}(k) (\mathbf{z}_a \otimes \mathbf{z}_b) && (5) \\ &= \mathbf{z}_b^{\mathsf{T}} \mathbf{u}_b(k) = \mathbf{z}_b^{T} \mathbf{X}_{(2)}(k) (\mathbf{z}_a \otimes \mathbf{z}_c) && (6) \\ &= \mathbf{z}_a^{\mathsf{T}} \mathbf{u}_a(k) = \mathbf{z}_a^{T} \mathbf{X}_{(3)}(k) (\mathbf{z}_b \otimes \mathbf{z}_c) && (7) \end{aligned}$$

where $\mathbf{u}_c(k) = \mathbf{X}_{(1)}(k)(\mathbf{z}_a \otimes \mathbf{z}_b) \in \mathbb{R}^{M_c}$, $\mathbf{u}_b(k) = \mathbf{X}_{(2)}(k)(\mathbf{z}_a \otimes \mathbf{z}_c) \in \mathbb{R}^{M_b}$, and $\mathbf{u}_a(k) = \mathbf{X}_{(3)}(k)(\mathbf{z}_b \otimes \mathbf{z}_c) \in \mathbb{R}^{M_a}$ are the input of the subfilters $\mathbf{z}_c$, $\mathbf{z}_b$, and $\mathbf{z}_a$, respectively. The matrices $\mathbf{X}_{(1)}(k) \in \mathbb{R}^{M_c \times M_a M_b}$, $\mathbf{X}_{(2)}(k) \in \mathbb{R}^{M_b \times M_a M_c}$, and $\mathbf{X}_{(3)}(k) \in \mathbb{R}^{M_a \times M_b M_c}$ denote the $\{1, 2, 3\}$-mode unfoldings of $\mathcal{X}(k)$, respectively. Equations (5), (6), and (7) indicate that the product (4) can be equivalently represented by the output of the three linear subfilters

C. Optimum trilinear filtering

Consider an Mth order *unknown* trilinearly separable FIR system whose impulse response is $\mathbf{h} = \mathbf{h}_a \otimes \mathbf{h}_b \otimes \mathbf{h}_c$, where $\mathbf{h}_a \in \mathbb{R}^{M_a}$, $\mathbf{h}_b \in \mathbb{R}^{M_b}$, $\mathbf{h}_c \in \mathbb{R}^{M_c}$ and $M_a M_b M_c = M$. Now consider a trilinearly separable filter $\mathbf{w} = \mathbf{w}_a \otimes \mathbf{w}_b \otimes \mathbf{w}_c$, where $\mathbf{w}_a \in \mathbb{R}^{M_a}$, $\mathbf{w}_b \in \mathbb{R}^{M_b}$ and $\mathbf{w}_c \in \mathbb{R}^{M_c}$ are its subfilters. The input regression vector $\mathbf{x}(k) \in \mathbb{R}^M$ is tensorized, resulting in the input tensor signal $\mathcal{X}(k) = \Theta(\mathbf{x}(k)) \in \mathbb{R}^{M_c \times M_b \times M_a}$. Both the unknown system and the trilinear filter are driven by this tensor signal. The output of the unknown system is the desired signal $d(k) = \mathbf{h}^{\mathsf{T}} \mathbf{x}(k)$ of the trilinear filter, as depicted in Fig. 1(a). At the filter output, the estimation error $e(k) = d(k) - y(k)$ is calculated, where $y(k) = \mathbf{w}^{\mathsf{T}} \mathbf{x}(k) = \mathcal{X}(k) \times_1 \mathbf{w}_c^{\mathsf{T}} \times_2 \mathbf{w}_b^{\mathsf{T}} \times_3 \mathbf{w}_a^{\mathsf{T}}$ is the filter output. The mean square value of the estimation error is chosen to design the filter, leading to the following optimization problem:

$$\min_{\mathbf{w}_a, \mathbf{w}_b, \mathbf{w}_c} \mathbb{E}\left[|d(k) - \mathcal{X}(k) \times_1 \mathbf{w}_c^{\mathsf{T}} \times_2 \mathbf{w}_b^{\mathsf{T}} \times_3 \mathbf{w}_a^{\mathsf{T}}|^2 \right]. \quad (8)$$

The objective function is clearly nonlinear with respect to the subfilters. Recalling that $y(k)$ can be represented in terms of its subfilters, the problem (8) can be divided in three subproblems:

$$\min_{\mathbf{w}_a} \; \mathbb{E}\left[\left|d(k) - \mathbf{w}_a^{\mathsf{T}}\mathbf{u}_a(k)\right|^2\right], \tag{9}$$

$$\min_{\mathbf{w}_b} \; \mathbb{E}\left[\left|d(k) - \mathbf{w}_b^{\mathsf{T}}\mathbf{u}_b(k)\right|^2\right], \tag{10}$$

$$\min_{\mathbf{w}_c} \; \mathbb{E}\left[\left|d(k) - \mathbf{w}_c^{\mathsf{T}}\mathbf{u}_c(k)\right|^2\right], \tag{11}$$

where $\mathbf{u}_a(k)$, $\mathbf{u}_b(k)$, and $\mathbf{u}_c(k)$ are defined as in Section II-B. Note that these vectors can be alternatively interpreted as weighted versions of the $\mathcal{X}(k)$ unfoldings. There is clearly an interdependency between the modes of $y(k)$ which hinders the optimization of (8). In view of this, the alternating least-squares (ALS) method can be used to solve (8). In this case, the subproblems (9), (10), and (11) are solved in a alternating manner. This method converges at least to a local minimum, and convergence to the global minimum cannot be guaranteed. Each subproblem corresponds to a classical LS estimation problem [1] (conditioned on the solutions provided by the other two subproblems). The solution of these subproblems is given by the WH equations:

$$\hat{\mathbf{w}}_a = \mathbf{R}_a^{-1}\mathbf{p}_a, \tag{12}$$

$$\hat{\mathbf{w}}_b = \mathbf{R}_b^{-1}\mathbf{p}_b, \tag{13}$$

$$\hat{\mathbf{w}}_c = \mathbf{R}_c^{-1}\mathbf{p}_c,, \tag{14}$$

where $\mathbf{R}_\lambda = \mathbb{E}\left[\mathbf{u}_\lambda(k)\mathbf{u}_\lambda(k)^{\mathsf{T}}\right] \in \mathbb{R}^{M_\lambda \times M_\lambda}$ is the autocorrelation matrix of $\mathbf{u}_\lambda(k)$, and $\mathbf{p}_\lambda = \mathbb{E}\left[d(k)\mathbf{u}_\lambda(k)\right] \in \mathbb{R}^{M_\lambda}$ is the crosscorrelation vector between $\mathbf{u}_\lambda(k)$ and $d(k)$ for $\lambda = a,\, b,\, c$. Note that these statistics can be expressed in terms of adaptive weighting matrices. For instance, consider the autocorrelation matrix of $\mathbf{u}_a(k)$:

$$\begin{aligned}\mathbf{R}_a &= \mathbb{E}\left[\mathbf{u}_a(k)\mathbf{u}_a(k)^{\mathsf{T}}\right] \\ &= \mathbb{E}\left[\mathbf{X}_{(3)}(k)(\mathbf{w}_b \otimes \mathbf{w}_c)(\mathbf{w}_b \otimes \mathbf{w}_c)^{\mathsf{T}}\mathbf{X}_{(3)}(k)^{\mathsf{T}}\right] \\ &= \mathbb{E}\left[\mathbf{X}_{(3)}(k)\mathbf{Q}_a\mathbf{X}_{(3)}(k)^{\mathsf{T}}\right].\end{aligned} \tag{15}$$

Now $\mathbf{R}_a$ is interpreted as the autocorrelation matrix of $\mathbf{X}_{(3)}(k)$ weighted by $\mathbf{Q}_a = (\mathbf{w}_b \otimes \mathbf{w}_c)(\mathbf{w}_b \otimes \mathbf{w}_c)^{\mathsf{T}}$. The crosscorrelation vector $\mathbf{p}_a$ can be interpreted as the weighted crosscorrelation between $\mathbf{X}_{(3)}(k)$ and $d(k)$:

$$\begin{aligned}\mathbf{p}_a &= \mathbb{E}\left[d(k)\mathbf{u}_a(k)\right] \\ &= \mathbb{E}\left[d(k)\mathbf{X}_{(3)}(k)(\mathbf{w}_b \otimes \mathbf{w}_c)\right] \\ &= \mathbb{E}\left[d(k)\mathbf{X}_{(3)}(k)\mathbf{q}_a\right],\end{aligned} \tag{16}$$

where $\mathbf{q}_a = (\mathbf{w}_b \otimes \mathbf{w}_c)$ and $\mathbf{Q}_a = \mathbf{q}_a\mathbf{q}_a^{\mathsf{T}}$.

D. Trilinear Wiener-Hopf algorithm

To implement the ALS optimization, the batch Trilinear Wiener-Hopf (TriWH) algorithm is proposed. It consists of sequentially calculating (12), (13), and (14) using the sample estimate of the autocorrelation matrices and crosscorrelation vectors, as described in Algorithm 1. The convergence is attained when the normalized square error (NSE) between the true filter $\mathbf{h}$ and the estimated filter $\mathbf{w}$ is smaller than a threshold ε.

While the classical WH solution presents the standard complexity of $O(M^2)$ due to the inversion of a $(M \times M)$-dimensional autocorrelation matrix, TriWH presents a complexity of $O(Q(M_a^2 + M_b^2 + M_c^2))$, where Q is the number of iterations necessary to attain the convergence. This computational complexity considerably smaller than its classic counterpart depending on how the input vector partitioning is done [2].

Algorithm 1 TriWH

procedure TRIWH($\mathbf{x}(k)$, $d(k)$, $\mathbf{h}$, ε)
- $q \leftarrow 0$
- Initialize NSE(q), $\mathbf{w}_a(q)$, $\mathbf{w}_b(q)$ and $\mathbf{w}_c(q)$.
- **repeat**
  - Initialize $\mathbf{R}_\lambda(k)$ and $\mathbf{p}_\lambda(k)$ for $\lambda = a, b, c$
  - $\mathbf{q}_c(q) \leftarrow (\mathbf{w}_a(q) \otimes \mathbf{w}_b(q))$
  - $\mathbf{Q}_c(q) \leftarrow \mathbf{q}_c\mathbf{q}_c^{\mathsf{T}}$
  - **for** $k = 0, \ldots, K-1$ **do**
    - $\mathcal{X}(k) \leftarrow \mathbf{\Theta}(\mathbf{x}(k))$
    - $\mathbf{R}_c(k+1) \leftarrow \mathbf{R}_c(k) + (1/K)\mathbf{X}_{(1)}(k)\mathbf{Q}_c(q)\mathbf{X}_{(1)}(k)^{\mathsf{T}}$
    - $\mathbf{p}_c(k+1) \leftarrow \mathbf{p}_c(k) + (1/K)d(k)\mathbf{X}_{(1)}(k)\mathbf{q}_c(q)$
  - **end for**
  - $\mathbf{w}_c(q+1) \leftarrow \mathbf{R}_c(k+1)^{-1}\mathbf{p}_c(k+1)$
  - $\mathbf{q}_b(q) \leftarrow (\mathbf{w}_a(q) \otimes \mathbf{w}_c(q+1))$
  - $\mathbf{Q}_b(q) \leftarrow \mathbf{q}_b\mathbf{q}_b^{\mathsf{T}}$
  - **for** $k = 0, \ldots, K-1$ **do**
    - $\mathcal{X}(k) \leftarrow \mathbf{\Theta}(\mathbf{x}(k))$
    - $\mathbf{R}_b(k+1) \leftarrow \mathbf{R}_b(k) + (1/K)\mathbf{X}_{(2)}(k)\mathbf{Q}_b(q)\mathbf{X}_{(1)}(k)^{\mathsf{T}}$
    - $\mathbf{p}_b(k+1) \leftarrow \mathbf{p}_b(k) + (1/K)d(k)\mathbf{X}_{(2)}(k)\mathbf{q}_b(q)$
  - **end for**
  - $\mathbf{w}_b(q+1) \leftarrow \mathbf{R}_b(k+1)^{-1}\mathbf{p}_b(k+1)$
  - $\mathbf{q}_a(q) \leftarrow (\mathbf{w}_b(q+1) \otimes \mathbf{w}_c(q+1))$
  - $\mathbf{Q}_a(q) \leftarrow \mathbf{q}_a\mathbf{q}_a^{\mathsf{T}}$
  - **for** $k = 0, \ldots, K-1$ **do**
    - $\mathcal{X}(k) \leftarrow \mathbf{\Theta}(\mathbf{x}(k))$
    - $\mathbf{R}_a(k+1) \leftarrow \mathbf{R}_a(k) + (1/K)\mathbf{X}_{(3)}(k)\mathbf{Q}_a(q)\mathbf{X}_{(3)}(k)^{\mathsf{T}}$
    - $\mathbf{p}_a(k+1) \leftarrow \mathbf{p}_a(k) + (1/K)d(k)\mathbf{X}_{(3)}(k)\mathbf{q}_a(q)$
  - **end for**
  - $\mathbf{w}_a(q+1) \leftarrow \mathbf{R}_a(q+1)^{-1}\mathbf{p}_a(q+1)$
  - $q \leftarrow q+1$
  - $\mathbf{w}(q) \leftarrow \mathbf{w}_a(q) \otimes \mathbf{w}_b(q) \otimes \mathbf{w}_c(q)$
  - NSE(q) $= \|\mathbf{h} - \mathbf{w}(q)\|_2^2 \,/\, \|\mathbf{h}\|_2^2$
- **until** $|\text{NSE}(q) - \text{NSE}(q-1)| < \varepsilon$

end procedure

III. NUMERICAL RESULTS

A separable FIR filter $\mathbf{h} = \mathbf{h}_a \otimes \mathbf{h}_b \otimes \mathbf{h}_c \in \mathbb{R}^M$ with $M = 1024$ parameters was estimated in the carried out experiments. The subfilters were defined similar to [2]: $\mathbf{h}_a \in \mathbb{R}^8$ is a vector whose m_ath element is given by $[\mathbf{h}_a]_{m_a} = 0.9^{m_a - 1}$ for $m_a = 1, 2, \ldots, 8$, $\mathbf{h}_b = [0, 0, \ldots, 0, 1, 0]^{\mathsf{T}} \in \mathbb{R}^{32}$, and $\mathbf{h}_c \in \mathbb{R}^4$ is a vector whose elements are Gaussian random variables with zero mean and unitary variance. This setup, depicted in Fig. 2, approximates a channel with echoes [2]. The input signal $x(k)$ was taken from a zero mean and unit variance Gaussian random process. An additive white Gaussian noise (AWGN) term with zero mean and variance 10^{-2} was added to the desired signal $d(k)$ for $k = 0, \ldots, K-1$. The sample size was set to $K = 15000$ in all experiments.

Monte Carlo simulations with N independent realizations were performed to assess the performance of NLMS, TensorLMS, TriWH, and the WH solution. The step-size of the LMS-based algorithms was set to 0.5. Independent impulse responses $\mathbf{h}^{(n)}$ were generated for each nth realization. The normalized mean square error (NMSE) between the

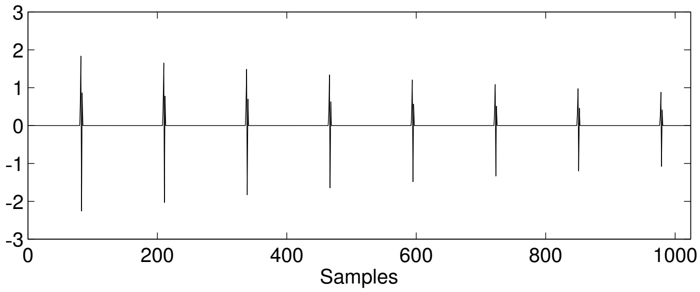


Fig. 2. Example of separable impulse response considered in this work.

actual and estimated filter at the qth iteration, defined as $\frac{1}{N}\sum_{n=1}^{N}\|\mathbf{h}^{(n)}-\mathbf{w}^{(n)}(q)\|_2^2/\|\mathbf{h}^{(n)}\|_2^2$, was used to measure the system mismatch. In the first experiment, the system to be identified was assumed to be perfectly separable. The performance of the studied methods is depicted in the left plot in Fig. 3. In the second experiment, their performances were assessed when the system was not perfectly separable. To reproduce this scenario, an AWGN component with zero mean and variance 10^{-4} was added to the true system impulse response. The right plot in Fig. 3 depicts their performance in this scenario.

When the system separability assumption holds, the TriWH solution presents the smallest system mismatch. It is an expected result since TriWH is a nonlinear method that was designed to properly explore the system separability, which is ignored by the classic methods. Furthermore, the adaptive weighting present in the autocorrelation and crosscorrelation matrices and vectors (c.f. (15) and (16)) contribute as well to the performance gain, playing the role of a weighted LS estimation with adjustable weights. It is important to recall that TriWH is less computationally expensive than its classical counterpart that do not exploit system separability. Regarding the iterative solutions, it can be seen that TriWH presented the best performance. In our simulations, the algorithm converged in about 3 iterations and presented a relative system mismatch much lower than the other algorithms. However, its computational complexity is greater than the iterative alternatives due to the calculation of matrix inverses. We also note that when the unknown system was not perfectly separable, TensorLMS and TriWH do not perform very well, since they could only identify the separable components. Since TensorLMS exploits the system separability, its performance could be similar to that of TriWH by setting a sufficiently small step-size, which would considerably decrease its convergence rate.

IV. Conclusion and Perspectives

The tensor filtering framework was introduced in the supervised separable system estimation problem. Such type of system is useful to model multidimensional array of sensors. The tensor formalism present in this framework provided proper notation and interpretation for this trilinear problem. A nonlinear problem was considered to perform the identification. This problem was solved by exploiting its trilinear structure, leading to three linear subproblems. Based on this idea, the TriWH algorithm was proposed. According to our numerical experiments, it performed better than alternative

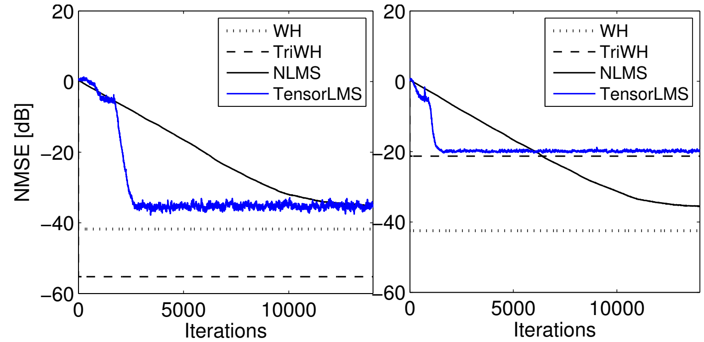


Fig. 3. Performance evaluation when the system is perfectly separable (left) and not perfectly separable (right).

solutions. A convergence analysis and the extension to the multichannel case will be provided in an extended version of this paper.

Acknowledgment

The authors would like to thank Dr. Markus Rupp and Dr. Stefan Schwarz for publicly providing the MATLAB implementation of TensorLMS.

References

[1] S. S. Haykin, *Adaptive filter theory*, 4th ed. Pearson Hall, 2002.

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

[3] R. H. Kwong and E. W. Johnston, "A variable step size LMS algorithm," *IEEE Transactions on Signal Processing*, vol. 40, no. 7, pp. 1633–1642, 1992.

[4] D. L. Duttweiler, "Proportionate normalized least-mean-squares adaptation in echo cancelers," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 5, pp. 508–518, 2000.

[5] R. D. Nowak and B. D. Van Veen, "Random and pseudorandom inputs for Volterra filter identification," *IEEE Transactions on Signal Processing*, vol. 42, no. 8, pp. 2124–2135, 1994.

[6] L.-H. Lim and P. Comon, "Blind multilinear identification," *IEEE Transactions on Information Theory*, vol. 60, no. 2, pp. 1260–1280, 2014.

[7] D. Muti and S. Bourennane, "Multidimensional filtering based on a tensor approach," *Signal Processing*, vol. 85, no. 12, pp. 2338–2353, 2005.

[8] L. N. Ribeiro, J. C. M. Mota, and A. L. F. de Almeida, "Trilinear Wiener filtering: Application to equalization problems," in *XXXI Simpósio Brasileiro de Telecomunicações, Fortaleza, Brazil*. SBrT, 2013.

[9] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[10] L. Grasedyck, "Polynomial approximation in hierarchical tucker format by vector-tensorization," Aachen, Germany, April 2010.

[11] L. Grasedyck, D. Kressner, and C. Tobler, "A literature survey of low-rank tensor approximation techniques," *GAMM-Mitteilungen*, vol. 36, no. 1, pp. 53–78, 2013.

[12] L. De Lathauwer, B. De Moor, and J. Vandewalle, "On the best rank-1 and rank-$(R_1, R_2, \ldots, R_n)$ approximation of higher-order tensors," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1324–1342, 2000.

[13] T. Zhang and G. H. Golub, "Rank-one approximation to high order tensors," *SIAM Journal on Matrix Analysis and Applications*, vol. 23, no. 2, pp. 534–550, 2001.