Abstract—We propose a blind FIR channel identification method based on the Parallel Factor (Parafac) analysis of a 3rd-order tensor composed of the 4-th order output cumulants. Our algorithm is based on a single-step least squares (LS) minimization procedure instead of using classical three-step Alternating Least Squares (ALS) methods. Using a Parafac-based decomposition, we avoid any kind of pre-processing such as the prewhitening operation, which is mandatory in most methods using higher-order statistics. Our method retrieves the channel vector without any permutation or scaling ambiguities. In addition, we establish a link between the cumulant tensor decomposition and the joint-diagonalization approach. Computer simulations illustrate the performance gains that our method provides with respect to other classical solutions. Initialization and convergence issues are also addressed.

I. INTRODUCTION

Blind channel identification and equalization consist in the retrieval of unknown information about the transmission channel and source signals from the knowledge of the received signal only. For several years, higher-order statistics (HOS) have been an important research topic in diverse fields including data communication, speech and image processing and geophysical data processing. The higher-order spectra have the ability to preserve both magnitude and (nonminimum-) phase information. Moreover, it is well-known that all cumulant spectra of order greater than 2 vanish for Gaussian signals, which makes HOS-based identification methods insensitive to an additive Gaussian noise.

Relationships connecting different higher-order cumulant slices and the parameters of a finite impulse response (FIR) model exist and it is well known that larger sets of output cumulants have the ability to improve identification performance [1]. Existing approaches to exploit the sample output cumulants include the joint-diagonalization of cumulant matrices [2]. However, such techniques involve a prewhitening transformation over the cumulant matrices, which is often a source of increased complexity and estimation errors. The factorization of third-order tensors containing sample 4th-order cumulants has the advantage of avoiding the prewhitening operation by fully exploiting the three-dimensional nature of the cumulant tensor.

The Parallel factor analysis (Parafac) of an m-dimensional tensor with rank $F$ consists in the decomposition of the tensor into a sum of $F$ factors where each factor can be written as the outer product of $m$ vectors [3]. The trilinear Parafac model ($m = 3$) has become very popular in the fields of Psychometrics and Chemometrics [4], [5] but it also has been widely used in Signal Processing applications [6], [7]. The key-point in the use of the Parafac is about its uniqueness property, which can be assured under simple conditions that are stated in the Kruskal Theorem [8]. Among many algorithms to fit the Parafac model, the Alternating Least Squares (ALS) algorithm is one of the most popular ones, despite its known problems concerning complexity, convergence speed and non-guaranteed convergence to the global minimum.

In this paper, we exploit the high symmetry of the 4th-order cumulants to propose a solution based on a single-step least squares (LS) minimization procedure. We recover the parameters of an FIR channel by means of the Parafac decomposition of a three-dimensional tensor containing the output 4th-order cumulants. The uniqueness issue is also addressed and we present computer simulations comparing our method with the joint-diagonalization based approach as well as with the (closed-form) Total Least Squares (TLS) solution. Initialization and convergence speed are discussed and we show that our method can be used to improve the TLS solution.

The paper is organized as follows: Section II introduces our signal model and some basic equations concerning the computation of 4th-order cumulants; in Section III we express the output cumulants as a three-dimensional tensor and show that it consists in a Parafac model; in Section IV we present the equations for the cumulant tensor decomposition and show that there is a direct link with the matrix diagonalization approach; in Section V we propose a simplified Parafac-based algorithm to estimate channel parameters; Section VI presents some illustrative computer simulation results and Section VII finally draws some conclusions and perspectives.

II. SIGNAL MODEL AND OUTPUT CUMULANTS

We start considering the baseband representation of a digital single-input single-output (SISO) communication channel where the output signal $y(n)$ after sampling at the symbol rate is written as follows:

$$y(n) = x(n) + v(n),$$

$$x(n) = \sum_{l=0}^{L} h_{l}s(n-l),$$

where $h_{0} = 1$, which is equivalent to a simple unit-norm constraint. Moreover, the following assumptions hold:
A1: The non-measurable, complex-valued, discrete input sequence \( s(n) \) is non-Gaussian, stationary, independent and identically distributed (i.i.d.) with symmetric distribution of zero-mean and variance \( \sigma_s^2 = 1 \).

A2: The additive Gaussian noise sequence \( v(n) \) has zero-mean and is independent from \( s(n) \). Its autocorrelation function is unknown.

A3: The channel frequency-response is \( H(\omega) = \sum_l h_l e^{-j\omega l} \) with complex coefficients \( h_l \) representing the equivalent discrete impulse response, including the pulse shaping filter, the channel itself and the receiving filter.

A4: The system is causal with memory \( L \), i.e., \( h_l = 0 \), \( \forall l \notin [0, L] \). In addition, \( h_L \neq 0 \) and \( L \neq 0 \).

The output 4th-order cumulants will be defined as follows

\[
c_{4,y}(\tau_1, \tau_2, \tau_3) \triangleq \text{cum} \left\{ y^*(n), y(n+\tau_1), y^*(n+\tau_2), y(n+\tau_3) \right\}.
\]

Using the channel model (1), taking assumptions A1 and A2 into account, and making use of the multilinearity property of cumulants, it is easy to show that [9]:

\[
c_{4,y}(\tau_1, \tau_2, \tau_3) = \gamma_{4,s} \sum_{l=0}^{L} h_l^* h_{l+\tau_1} h_{l+\tau_2} h_{l+\tau_3}, \tag{2}
\]

where \( \gamma_{4,s} = c_{4,s}(0,0,0) \). Based on (2) and on assumption A4, it is easy to verify that

\[
c_{4,y}(\tau_1, \tau_2, \tau_3) = 0, \quad \forall \ |\tau_1|, \ |\tau_2|, \ |\tau_3| > L.
\]

Hence, making the time-lags \( \tau_1, \tau_2, \) and \( \tau_3 \) vary in the interval \([-L, L]\) we get all the nonzero values of \( c_{4,y}(\tau_1, \tau_2, \tau_3) \). Such a choice allows us to construct a maximal redundant information model, in which the 4th-order cumulants are taken for time-lags \( \tau_1, \tau_2, \tau_3 \) within the interval \([-L, L]\).

### III. TENSOR FORMULATION

Let us define the three-dimensional tensor \( C^{(4,y)} \in \mathbb{C}^{(2L+1) \times (2L+1) \times (2L+1)} \), in which the element in position \((i, j, k)\) corresponds to \( c_{4,y}(i, j, k), \) with \( i = \tau_1 + L + 1, \ j = \tau_2 + L + 1 \) and \( k = \tau_3 + L + 1 \), as shown in Figure 1a. Therefore, \( C^{(4,y)} \) is clearly a 3rd-order tensor represented as a cube of dimensions \((2L+1) \times (2L+1) \times (2L+1)\) and it can be written as:

\[
C^{(4,y)} = \sum_{i=1}^{2L+1} \sum_{j=1}^{2L+1} \sum_{k=1}^{2L+1} C_{ijk} e_i^{(2L+1)} \circ e_j^{(2L+1)} \circ e_k^{(2L+1)}, \tag{4}
\]

where \( C_{ijk} = c_{4,y}(i - L - 1, j - L - 1, k - L - 1) \), the symbol \( \circ \) stands for the outer product and \( e_p^{(P)} \) denotes the \( p \)-th canonical basis vector with dimension \( P \) [10]. Replacing (2) in (4) we can easily write the tensor \( C^{(4,y)} \) as a sum of \( L + 1 \) outer products, each one involving exactly 3 vectors, as follows:

\[
C^{(4,y)} = \gamma_{4,s} \sum_{l=0}^{L} h_l^* h_l \circ h_l^* \circ h_l \tag{5}
\]

where \( h_l = \sum_{p=1}^{2L+1} h_{l+p-L-1} e_p^{(2L+1)} \). The above notation leads us to define the channel coefficient matrix \( H \in \mathbb{C}^{(2L+1) \times (L+1)} \) as follows:

\[
H \triangleq \mathcal{H}(h) = \begin{bmatrix} h_0 & h_1 & \cdots & h_L \\ 0 & 0 & \cdots & h_0 \\ \vdots & \vdots & \ddots & \vdots \\ h_0 & h_1 & \cdots & h_L \\ \vdots & \vdots & \ddots & \vdots \\ h_L & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \end{bmatrix} \tag{6}
\]

where \( \mathcal{H}(\cdot) \) is an operator that builds a special Hankel matrix from the vector argument as shown above and the channel coefficients vector is:

\[
h = \begin{bmatrix} h_0 \ldots h_L \end{bmatrix}^T \in \mathbb{C}^{(L+1)} \tag{7}
\]

### IV. CUMULANT TENSOR DECOMPOSITION

Slicing tensor \( C^{(4,y)} \) along the three possible directions (horizontal, vertical and frontal) yields in each case \( 2L + 1 \) matrices of dimensions \((2L+1) \times (2L+1)\). Figure 1b illustrates frontal slicing for \( k \in [1, 2L+1] \). The two-dimensional slices obtained from each direction will be denoted \( C_{i,j,k}^{(4,y)} \) and \( C_{i,j,k}^{(4,y)} \) with \( i, j, k \in [1, 2L+1] \), respectively.

Slicing \( C^{(4,y)} \) along the frontal direction we get:

\[
C_{i,j,k}^{(4,y)} = \sum_{l=1}^{2L+1} \sum_{j=1}^{2L+1} C_{ijk} e_i^{(2L+1)} e_j^{(2L+1)} e_k^{(2L+1)}
\]

and then

\[
C_{i,j,k}^{(4,y)} = \gamma_{4,s} \sum_{l=0}^{L} h_l^* h_{l+k-L-1} h_l h_l^T
\]

for all \( k \in [1, 2L+1] \), where \( D_p(\cdot) \) is an operator that forms a diagonal matrix with the main diagonal constituted by the elements of the \( p \)-th row of the matrix argument and:

\[
\Sigma = HD_{k}(\Sigma) H^T, \tag{8}
\]

where the \( \text{Diag}(\cdot) \) operator builds a diagonal matrix using the entries of the vector argument, so \( D_k(\Sigma) \in \mathbb{C}^{(L+1) \times (L+1)} \). Slicing \( C^{(4,y)} \) along the vertical direction, we get:

\[
C_{i,j,k}^{(4,y)} = \sum_{l=1}^{2L+1} \sum_{j=1}^{2L+1} C_{ijk} e_i^{(2L+1)} e_j^{(2L+1)} e_k^{(2L+1)}
\]

and then

\[
C_{i,j,k}^{(4,y)} = \gamma_{4,s} \text{Diag}(h^*) D_j(\Sigma) H^T
\]

for all \( j \in [1, 2L+1] \), where

\[
\Sigma = HDiag(h^*) D_{j}(\Sigma) H^T, \tag{9}
\]

where

\[
\Sigma = HDiag(h^*) D_{j}(\Sigma) H^T, \tag{10}
\]

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the ICC 2007 proceedings.
for all $j \in [1, 2L + 1]$. Finally, taking the horizontal direction and following the same reasoning, we can easily write:

$$C_{k, i}^{(4,y)} = \sum_{j=1}^{2L+1} \sum_{k=1}^{2L+1} C_{ijk}^{(2L+1)} e_k^{(2L+1)^T}$$

$$= \gamma_{4,s} \sum_{l=0}^{L} h_l^{*} h_{l-i-L-1} h_i^{*} h_i^{T}$$

$$= \gamma_{4,s} H^{*} D_i (H) \text{Diag}(h^*) H^{T}$$

$$= \gamma_{4,s} H^{*} D_i (H) \Sigma^T,$$  \hspace{1cm} (11)

for all $i \in [1, 2L + 1]$.

In particular, equation (8) establishes a direct link between the tensor decomposition and the joint-diagonalization approach [2]. However, diagonalization procedures of the type $A = UDU^{H}$ require $U$ to be unitary, i.e. $UU^{H} = I$. In our case, since $H$ is not unitary, the diagonalization of matrices in (8) for $k \in [1, 2L + 1]$ is not possible without a previous orthonormalization step in which a new set of modified cumulant matrices is constructed as $C_{k,i}^{(4,y)} = WC_{k,i}^{(4,y)} H^{H}$, with $WH$ unitary. Matrix $W$ is often referred to as the pre-whitening factor and its computation usually requires resorting to second-order statistics (SOS). This additional step, very common in most HOS-based methods, is time-consuming and often responsible for increased estimation errors.

It is important to express the cumulant tensor in an unfolded tensor representation, obtained by stacking up the 2D slices. Three unfolded representations of $C^{(4,y)}$ are possible. Stacking up matrices $C_{k,i}^{(4,y)}$ for $k \in [1, 2L + 1]$, we get:

$$C_{[3]} = \begin{pmatrix}
C_{[3]}^{(4,y)} \\
\vdots \\
C_{[3]}^{(4,y)} \end{pmatrix}
\begin{pmatrix}
H D_1 (H) \\
\vdots \\
H D_{2L+1} (H)
\end{pmatrix}
H^{H},$$

hence

$$C_{[3]} = \gamma_{4,s} \left( \Sigma \otimes H \right) H^{H},$$  \hspace{1cm} (12)

where $\otimes$ stands for the Khatri-Rao product. We also get:

$$C_{[2]} = \gamma_{4,s} \left( H^{*} \otimes \Sigma \right) H^{T}$$

from (10) and finally

$$C_{[1]} = \gamma_{4,s} \left( H \otimes H^{*} \right) \Sigma^{T}$$  \hspace{1cm} (14)

from (11). Table I resumes the information about the Parafac decomposition of tensor $C^{(4,y)}$ highlighting the formulae developed above. Equations (12)–(14) can be used to estimate matrices $H$ and $\Sigma$ and thus the channel parameters. However, we can simplify the estimation of $H$ by taking the relationships between the channel coefficient vector and the matrices $H$ and $\Sigma$ into account, thus eliminating the ambiguities.

<table>
<thead>
<tr>
<th>Direction</th>
<th>Tensor 2D slices</th>
<th>Unfolded representations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Horizontal</td>
<td>$C_{[4,y]}^{(3,y)} = \gamma_{4,s} H^{*} D_i (H) \Sigma^{T}$</td>
<td>$C_{[1]} = \gamma_{4,s} \left( H \otimes H^{*} \right) \Sigma^{T}$</td>
</tr>
<tr>
<td>Vertical</td>
<td>$C_{[4,y]}^{(3,y)} = \gamma_{4,s} \Sigma D_i (H^{*}) H^{T}$</td>
<td>$C_{[2]} = \gamma_{4,s} \left( H^{*} \otimes \Sigma \right) H^{T}$</td>
</tr>
<tr>
<td>Frontal</td>
<td>$C_{[4,y]}^{(3,y)} = \gamma_{4,s} H D_k (\Sigma) H^{H}$</td>
<td>$C_{[3]} = \gamma_{4,s} \left( \Sigma \otimes H \right) H^{H}$</td>
</tr>
</tbody>
</table>

The uniqueness issue

The great importance of Parafac is mostly due to the simple conditions that assure its uniqueness, for tensors of order higher than 2. A sufficient uniqueness condition is stated by the Kruskal Theorem [8]. For the case of the 3rd-order tensor $C^{(4,y)}$, the Kruskal Theorem states that the Parafac decomposition is unique up to column scaling and permutation ambiguities if:

$$2k_{H} + k_{\Sigma} \geq 2(L + 1),$$

where $k_{H}$ is the $k$-rank of the channel matrix $H$. The $k$-rank of an $n \times m$ matrix $X$, denoted by $k_{X}$, stands for the largest integer $k_{X}$ for which every set containing $k_{X}$ columns of $X$ is independent. From this definition, note that $k_{X} \leq r_{X} \leq \min(n, m)$. Several authors have addressed the Parafac uniqueness problem and different proofs have been given to the Kruskal Theorem [8], [5].

Due to its Hankel structure and assumption A4, matrix $H$ is full-rank and then $k_{H} = r_{H} = L + 1$. From (9) and
assumption A4, we conclude that \( \Sigma \) is also full-rank so that 
\( k_\Sigma = r_\Sigma = L + 1 \) and hence \( 2k_H + k_\Sigma = 3L + 3 \). So, from (15) we conclude that the Kruskal uniqueness condition is always satisfied under the considered assumptions. Therefore, the Parafac-based Blind channel identification (PBCI) algorithm, which is based on a joint diagonalization approach. Exploiting the Hankel structure of \( H \) and the system constraint \( h_0 = 1 \), no permutation nor scaling ambiguities remain in the model. Making use of the symmetries of the 4th-order cumulants and of some Khatri-Rao product properties, we eliminate the need for estimating \( H \) prior to \( h \).

V. PARAFAC-BASED BLIND CHANNEL IDENTIFICATION ALGORITHM

In order to estimate the channel coefficients without having to compute \( H \) and \( \Sigma \) we make use of relationships (6) and (9) and, based on the following property of the Khatri-Rao product [11], we propose an iterative Least Squares (LS) procedure to estimate vector \( h \).

Property 1 If matrices \( A \in \mathbb{C}^{m \times n} \) and \( B \in \mathbb{C}^{n \times m} \) and vector \( d \in \mathbb{C}^n \) are such that \( X = ADAG \), then it holds
\[
\text{vec}(X) = (B^T \circ A) d
\] (16)
where \( \text{vec}(\cdot) \) stands for the vectorizing operator.

Note from Table I that, among the three unfolded matrices, \( C_{[1]} \) is the most suitable representation to apply Property 1. That is because we can rewrite (14) as follows:
\[
C_{[1]} = \gamma_{4,s} (H \circ H^*) \text{Diag}(h^*) H^T
\]
and thus, using (16), it is straightforward to write:
\[
\text{vec}(C_{[1]}) = \gamma_{4,s} (H \circ H \circ H^*) h^*.
\] (17)

The channel coefficients vector \( h \) can be obtained by minimizing the following least squares (LS) cost function:
\[
J(h^*) = \left\| \text{vec}(C_{[1]}) - \hat{G}^{(r-1)} h^* \right\|^2_F
\] (18)
by means of an iterative procedure, where:
\[
\hat{G}^{(r-1)} = \gamma_{4,s} (\hat{H}^{(r-1)} \circ \hat{H}^{(r-1)} \circ \hat{H}^{(r-1)*}).
\] (19)

The algorithm is initialized with a Hankel matrix \( \hat{H}^{(0)} \) in which the first column is \( [0^T_L \ h_0^{(0)T}]^T \) and the last row is \( [\hat{h}_L^{(0)} \ 0^T_L]^T \), where \( h_0^{(0)} = [1 \ v]^T \) and \( v \in C^{(L)} \) is a Gaussian random vector. The algorithm iterates until convergence of the parametric estimator, i.e., until \( \| h^{(r)} - \hat{h}^{(r-1)} \| / \| h^{(r)} \| \leq \varepsilon \), where \( r \) is the iteration number and \( \varepsilon \) is an arbitrary small positive constant. At each iteration \( r \), we use the estimate \( h^{(r-1)} \) to update \( H^{(r-1)} \).

Our strategy ensures the Hankel structure of \( H \) at each iteration, taking advantage of its full-rank property to make the tensor decomposition essentially unique and free from permutation ambiguities. The constraint \( h_0 = 1 \) is taken into account in the construction of \( \hat{h}^{(r)} \) in order to avoid any kind of scaling ambiguities. Furthermore, a single-step LS minimization procedure is used instead of the classical three-step alternating least squares (ALS) algorithm. For this reason, our method should be expected to increase convergence speed. Finally, this approach is shown to provide an improved solution at each new iteration and if it converges to the global minimum, the LS solution to the model is found [3].

After initializing \( \hat{h}^{(0)} \) as a Gaussian random vector, the algorithm to estimate the channel coefficients vector can be summarized as follows, for \( r \geq 1 \):

1) Use (6) to build \( \hat{H}^{(r-1)} = H(\hat{h}^{(r-1)}) \);
2) Using (19), compute \( \hat{G}^{(r-1)} \);
3) Minimize the cost function (18) so that
\[
\hat{h}^{(r)*} = \gamma_{4,s}^{-1} \hat{G}^{(r-1)*} \text{vec}(C_{[1]})
\] (20)
where \((\cdot)^\#\) stands for the Moore-Penrose pseudoinverse, computed from the singular value decomposition (SVD) of its matrix argument.
4) Reiterate until convergence of the parametric error, i.e., \( \| h^{(r)} - \hat{h}^{(r-1)} \| / \| h^{(r)} \| \leq \varepsilon \).

Next, we will present some computer simulations illustrating the applicability of the proposed Parafac-based Blind Channel Identification (PBCI) method.

VI. COMPUTER SIMULATIONS

In this section, our results are compared with those obtained from the well-known Fourth-Order System Identification (FOSI) algorithm, which is based on a joint diagonalization technique. As suggested by the authors of this latter method, the FOSI algorithm performance is obtained by averaging the results of the two solutions proposed in [2]. In addition, we also compare PBCI results with the optimal solution in the total least squares (TLS) sense, proposed in [1]. In order to further assess the performance of our methods, other algorithms proposed in the literature should be used in the future, such as the FIR system identification proposed in [12].

Parametric channel estimation performance is evaluated by means of the normalized mean squared error (NMSE) of the estimator, which is computed through the following formula:
\[
\text{NMSE} = \frac{1}{P} \sum_{p=1}^{P} \frac{\| \hat{h}_p^{(\infty)} - h \|^2}{\| h \|^2},
\] (21)
where \( P \) is the number of Monte Carlo runs and \( \hat{h}_p^{(\infty)} \) is the channel estimate obtained after convergence of the experiment \( p \in [1, P] \).

Computer simulations indicate that our approach outperforms both the FOSI algorithm and the TLS solution.
That can be observed in Figure 2, where the NMSE is plotted against signal-to-noise ratio (SNR) for PBCI, FOSI and TLS algorithms. These curves represent the average of $P = 50$ Monte Carlo runs and channel parameter vector equals $h = [1, 1.37 - 1.12i, 0.94 - 0.75i]^T$ ($L = 2$). Fourth-order cumulants were estimated from $N = 10000$ output data samples assuming perfect knowledge of the channel memory $L$.

We also show that the number of iterations required for convergence of our Parafac-based algorithm can be reduced by initializing PCBI with the TLS solution. Figure 3 illustrates this situation, showing that this new initialization condition yields the same steady state results as before, but requires less iterations. This can be viewed as a way to improve the TLS solution over a limited number of iteration steps.

VII. CONCLUSIONS AND PERSPECTIVES

This paper has presented a new blind FIR channel identification method based on the Parafac decomposition of a 3rd-order tensor composed of 4th-order output cumulants. Our method fully exploits the three-dimensional nature of the cumulant tensor and has the advantage of avoiding any kind of preprocessing. Moreover, our tensor decomposition algorithm is based on a single-step LS procedure instead of the classical three-step alternating least squares (ALS) algorithm. Uniqueness and convergence issues have been addressed. Computer simulations show that the Parafac-based approach provides better estimation performance than both the (closed-form) TLS solution and the joint-diagonalization based algorithm. Furthermore, the convergence of the PBCI algorithm can be accelerated when it is initialized with the TLS solution.

REFERENCES


Fig. 2. Identification performances of PBCI, FOSI and TLS methods with QPSK modulation.

Fig. 3. Evolution of the parametric error: PBCI initialized with TLS solution vs. random initialization.