Tensor approach for blind FIR channel identification using 4th-order cumulants

Carlos Estêvão R. Fernandes
Gérard Favier and João Cesar M. Mota

contact: cfernand@i3s.unice.fr

June 8, 2006
Outline

1. HOS and blind system identification
2. FIR channel model
3. Third-order tensor of 4th-order cumulants
4. Canonical tensor decomposition
5. Decomposing the cumulant tensor
6. Parafac-based Blind FIR Channel Identification
7. Illustrative computer simulations
8. Conclusions and perspectives
1. HOS and blind system identification

- Higher-order statistics (HOS): phase retrieval
- Moments $\times$ cumulants: Gaussian noise elimination
- Relationships connecting cumulant slices of different orders
- The use of a larger set of output cumulants improve identification performance: Total Least Squares (TLS) solution of an overdetermined system [Comon92].
- Optimality is defined in terms of asymptotic performance of cumulant estimators, which is not taken into account by these methods.
- Combination of the statistical data arises as a solution. Existing approaches include:
  - Slice weighting: [Fonollosa93], [DingLiang00], [DingLiang01]
  - Joint diagonalization: [Cardoso93], [Cardoso96], [Belouchrani00]
  - Tensor-based techniques: DeLathauwer, Comon and ???
2. FIR channel model

The following assumptions hold:

A1 : Input signal $s(n)$ is non-Gaussian, stationary, i.i.d. with zero-mean and variance $\sigma_s^2 = 1$.

A2 : Additive Gaussian noise sequence $v(n)$ is zero-mean and independent from $s(n)$.

A3 : Channel coefficients $h(l)$ represent the equivalent i.r. including pulse shaping filters.

A4 : $h(l) = 0 \forall l \notin [0, L]$. In addition, $h(L) \neq 0$ and $L \neq 0$.

$$y(n) = \sum_{l=0}^{L} h(l) s(n - l) + v(n), \quad h(0) = 1.$$ (1)

$$s(n) \quad h(t) \quad x(t) \quad + \quad y(t) \quad w(t)$$
3. Third-order tensor of 4th-order cumulants

- Fourth-order cumulants:

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

$$c_{4,y}(\tau_1, \tau_2, \tau_3) = \gamma_{4,s} \sum_{l=0}^L h^*(l)h(l+\tau_1)h^*(l+\tau_2)h(l+\tau_3)$$

where \(\gamma_{4,s} = C_{4,s}(0, 0, 0)\).

$$C^{(4,y)} = \sum_{i=0}^{2L} \sum_{j=0}^{2L} \sum_{k=0}^{2L} c_{4,y}(i-L, j-L, k-L)e_{i+1} \circ e_{j+1} \circ e_{k+1}$$
4. Canonical tensor decomposition

Revisiting Parafac:

\[ x_{i,j,k} = \sum_{f=1}^{F} a_{if} b_{jf} c_{kf} \]  

(2)

where \( i \in [1, I], j \in [1, J] \) and \( k \in [1, K] \).

Defining \([A]_{if} = a_{if}, [B]_{jf} = b_{jf}\) and \([C]_{kf} = c_{kf}\),

\[
\begin{align*}
A &= [a_1 \cdots a_F] \\
B &= [b_1 \cdots b_F] \\
C &= [c_1 \cdots c_F] 
\end{align*}
\]
• Slicing along horizontal direction:

\[ X_{i,.} = BD_i(A)C^T \in \mathbb{C}^{J \times K} \implies X_{[1]} = (A \odot B)C^T \in \mathbb{C}^{I \times J \times K} \]

• Slicing along vertical direction:

\[ X_{.j} = CD_j(B)A^T \in \mathbb{C}^{K \times I} \implies X_{[2]} = (B \odot C)A^T \in \mathbb{C}^{K \times J \times I} \]

• Slicing along frontal direction:

\[ X_{..k} = AD_k(C)B^T \in \mathbb{C}^{I \times J} \implies X_{[3]} = (C \odot A)B^T \in \mathbb{C}^{K \times I \times J} \]
The Kruskal rank and uniqueness condition

Parafac is shown to be essentially unique up to trivial ambiguities if

\[ k_A + k_B + k_C \geq 2(F + 1), \quad F > 1. \quad (3) \]

where \( k_A \leq \text{rank}(A) \).

Remarks:

- The rank of a three-dimensional tensor is defined as the minimum number \( F \) of (3-way) factors needed to decompose the tensor in the form of (2).

- Great advantage of Parafac over bilinear decompositions, where some rotation is always possible without changing the fit of the model.

- The rank is not bounded by the tensor dimensions: blind identification of systems with more inputs than outputs.
5. Decomposing the cumulant tensor

\[ C^{(4,y)} = \sum_{i=0}^{2L} \sum_{j=0}^{2L} \sum_{k=0}^{2L} c_{4,y}(i - L, j - L, k - L) e_{i+1} \circ e_{j+1} \circ e_{k+1}, \quad \in \mathbb{C}^{(2L+1) \times (2L+1) \times (2L+1)} \]

- Slicing \( C^{(4,y)} \) along horizontal direction, we get

\[ C^{(4,y)}_{:,k} = \gamma_{4,s} H_{k}(\Sigma) H_{k}^{H}, \quad \in \mathbb{C}^{(2L+1) \times (2L+1)} \]
• Using the unfolded tensor notation:

\[ X_{..k} = AD_k(C)B^T \in \mathbb{C}^{I \times J} \implies X_{[3]} = (C \diamond A)B^T \in \mathbb{C}^{KI \times J} \]

we get

\[ C_{(4,y)}^{(4,y)} = \gamma_{4,s}HD_k(\Sigma)H^H \implies C_{[3]} = \gamma_{4,s}(\Sigma \diamond H)H^H \in \mathbb{C}^{(2L+1)^2 \times (2L+1)} \]

where \( D_k(\cdot) \) is a diagonal matrix built from the \( k \)th row of the matrix argument.

• By association from the above relations:

\[
\begin{align*}
A &= H \\
B &= H^* \\
C &= \gamma_{4,s}\Sigma, \quad \text{where} \quad \Sigma = HD_1(h^H),
\end{align*}
\]

where

\[
h \triangleq \begin{bmatrix} h(0) & h(1) & \ldots & h(L) \end{bmatrix}^T \quad \text{and} \quad H \triangleq \mathcal{H}(h)
\]
where the operator $\mathcal{H}(\cdot)$ builds a special Hankel matrix from the vector argument so that

$$
\mathbf{H} = 
\begin{pmatrix}
0 & 0 & \cdots & h(0) \\
\vdots & \vdots & \ddots & \vdots \\
0 & h(0) & \cdots & h(L - 1) \\
h(0) & h(1) & \cdots & h(L) \\
\vdots & \vdots & \ddots & \vdots \\
h(L - 1) & h(L) & \cdots & 0 \\
h(L) & 0 & \cdots & 0
\end{pmatrix} 
\in \mathbb{C}^{(2L + 1) \times (L + 1)}
$$

Remarks:

- $\mathbf{C}^{(4,y)}_{j,k}$ establishes a direct link between the tensor decomposition and the joint-diagonalization approach.

- To diagonalize $\mathbf{C}^{(4,y)}_{j,k}$, since $\mathbf{H}$ is not unitary, a prewhitening step is needed in order to make its columns orthonormal.
6. **Parafac-based Blind Channel Identification**

- Algorithm based on iterative Least Squares (LS) procedure:
  1. For iteration \( r = 0 \), initialize \( \hat{h}^{(r)} \) as a Gaussian random vector;
  2. Build the channel matrix as \( H^{(0)} = \mathcal{H}(\hat{h}^{(0)}) \) and initialize \( A_0, B_0 \) and \( C_0 \);
  3. For \( r \geq 1 \), minimize the cost function \( J(\hat{h}^{(r)}) = \|C_3 - (C_{r-1} \odot A_{r-1})B_{r-1}^T\|_F^2 \) in the LS sense to estimate \( \hat{h}^{(r)} \) so that
    \[
    \hat{h}^{(r)} = \left[H^{(r-1)} \odot (H^{(r-1)} \odot H^{(r-1)*})\right] \# vec(C_3)
    \]
    (4)
  4. Compute \( H^{(r)} \) from \( \hat{h}^{(r)} \) and update \( A_r, B_r \) and \( C_r \);
  5. Reiterate until convergence of the parametric error, i.e. \( \|\hat{h}^{(r)} - \hat{h}^{(r-1)}\| / \|\hat{h}^{(r)}\| < \varepsilon \).
Remarks:

- This strategy ensures an improved solution at each iteration and if it converges to the global minimum, then the LS solution to the model is found.

- The full-rank property of the channel matrix $\mathbf{H}$ ensures that $k_H = \text{rank}(\mathbf{H}) = L + 1$ and, due to assumption A4, we have $\text{rank}(\Sigma) \geq 2$.

- The Hankel structure of $\mathbf{H}$ must be guaranteed in order to the identifiability condition to be satisfied.

- Exploiting the Hankel structure, no permutation and/or scaling ambiguities remain.
7. Illustrative computer simulations

- Simulation scenario:
  \[ h = [1.0, 1.37 - 1.12i, 0.94 - 0.75i]^T \]

- Sample data length: 5000 symbols

- Noise range: signal-to-noise ratio varies between 5 and 35dB

- Monte Carlo runs: \( M = 50 \)

- Evaluation criterion:
  \[
  NMSE = \frac{1}{M} \sum_{m=1}^{M} \frac{\|h - \hat{h}_m\|^2}{\|h\|^2}
  \]
Performances of channel identification algorithms: Parafac-based approach × FOSI × TLS.

QPSK modulation ($L = 2$)

16-PSK modulation ($L = 2$)
Parametric error convergence of PBCI algorithm using random and TLS initializations

QPSK modulation ($L = 2$)
8. Conclusions and perspectives

• We have established a new blind identification method based on the Parafac decomposition of a 3rd-order tensor.

• This technique avoids a non-optimal pre-processing step used in classical diagonalization-based algorithms ⇒ performance gains!

• Convergence speed can be increased by initializing the algorithm with the TLS solutions, thus improving its performance.

• The method is robust to noise, as it should be expected for HOS-based techniques.

• Next step is to extend the idea to the problem of blind MIMO channel identification and sources separation as opposed to the tensor diagonalization techniques:
  – Overdetermined × Underdetermined mixtures;
  – Static × dynamic cases;