

LABORATOIRE



INFORMATIQUE, SIGNAUX ET SYSTÈMES  
DE SOPHIA ANTIPOLIS  
UMR 6070

# BLIND MIMO PARA-UNITARY EQUALIZATION

*Ludwig ROTA, Pierre COMON, Lieven DE LATHAUWER, Sylvie ICART*

*Projet ASTRE*

Rapport de recherche  
ISRN I3S/RR-2004-44-FR

Dec. 2004

---

RÉSUMÉ :

Cet article présente un égaliseur aveugle MIMO spatio-temporel de mélanges convolutifs. Une paramétrisation de l'égaliseur est proposée lorsque les observations ont été préalablement blanchies. Nos développements théoriques conduisent à un algorithme numérique qui balaye toutes les paires de sorties retardées. Cet algorithme implique la résolution d'un système polynomial de faible degré, dont les coefficients dépendent des cumulants des sorties. Un autre algorithme est proposé ensuite, et recherche la solution en un nombre fini d'opérations. Ce dernier algorithme peut être utilisé pour initialiser le premier. Les performances des algorithmes sont évaluées à l'aide de simulations.

MOTS CLÉS :

Egalisation Aveugle, Multi-entrees Multi-sorties, Communications numeriques

---

ABSTRACT:

This paper introduces a blind MIMO space-time equalizer for convolutive mixtures. A parametrization of the equalizer is proposed when observations have been pre-whitened. Theoretical developments lead to a numerical algorithm that sweeps all pairs of delayed outputs. This algorithm involves the solution of a low degree polynomial system, whose coefficients depend on the output cumulants. Next, another algorithm is derived that aims at finding the solution in a finite number of steps. The latter algorithm may be used to initialize the former. Simulations and performance of the numerical algorithms are also included.

KEY WORDS :

Blind Equalization, Multiple inputs Multiple Outputs, Digital communications

# Blind MIMO Paraunitary Equalization

Ludwig Rota, *Student Member, IEEE*, Pierre Comon, *Member, IEEE*,  
 Lieven De Lathauwer, *Member, IEEE*, and Sylvie Icart

## Abstract

This paper introduces a blind MIMO space-time equalizer for convolutive mixtures. A parametrization of the equalizer is proposed when observations have been pre-whitened. Theoretical developments lead to a numerical algorithm that sweeps all pairs of delayed outputs. This algorithm involves the solution of a polynomial system, the coefficients of which depend on the output cumulants. Next an algorithm is derived that aims at finding the solution in a finite number of steps. The latter algorithm may be used to initialize the former. Simulations and performance of the numerical algorithms are also included.

## Index Terms

Blind deconvolution; Blind equalizers; Multiple-Input/Multiple-Output (MIMO); High Order Statistics; Paraunitary parametrization; Off-line algorithm

## I. INTRODUCTION

In actual digital communication systems, the equalization problem is solved using learning sequences. These sequences, known by transmitter and receiver, permit to estimate channel parameters. Nevertheless, in the future, learning sequences may be seen as reducing the throughput, because they occupy a non negligible space in transmitted sequences. Hence less actual information is transmitted, *i.e.* useful data

Manuscript received February 5, 2004. Part of this research was supported by (1) the Research Council of the Katholieke Universiteit Leuven (GOA-MEFISTO-666), (2) the Flemish Government (F.W.O. project G.0240.99, the F.W.O. Research Communities ICCoS and ANMMM), (3) the Belgian Federal Government (IUAP V-22) and (4) the Excellence Network PASCAL (no. 2002-506778).

L. Rota, P. Comon and S. Icart are with Laboratoire I3S, CNRS UMR 6070, 2000 route des Lucioles, BP 121, 06903 Sophia-Antipolis Cedex, France (e-mail: {rota,comon,icart}@i3s.unice.fr).

L. De Lathauwer is with Laboratoire ETIS, CNRS UMR 8051, 6 avenue du Ponceau, BP 44, 95014 Cergy-Pontoise Cedex, France (e-mail: delathau@ensea.fr)

rate is lower than system data rate. Next, in some situations, learning sequences exist but are not known to the receiver.

Thus, blind separation methods for Multiple-Input Multiple-Output (MIMO) channels have raised an increasing interest for digital communications since they do not need learning sequences. Most blind MIMO equalization techniques use High Order Statistics (HOS) for separating signals [1], [2], [3], [4], [5], [6] this can be implicit through constant modulus [7], [8], [9], [10] or constant power [11] criteria. Algorithms resulting of such methods are also very important in military applications, such as interception.

Our main contribution consists of a block algorithm dedicated to blind MIMO equalization [12], [13], [14]. The particularity is that our method is based on a factorization of paraunitary filters like in [15], [16]. The paraunitary condition is not very restrictive since prewhitening can always be performed in a first stage (in a non unique manner) [17], [16]. Moreover, the algorithm derived from our method is very attractive since it can also be implemented "off-line". More precisely, our algorithm can run iterations on a single given block until convergence. On the contrary, "on-line" algorithms run a new iteration after each symbol arrival; consequently, their convergence is known to be longer in terms of number of symbols required (typically from 10,000 to 100,000 symbols).

Algorithms like PAJOD [18], [19] have already been proposed for MIMO channels. Unfortunately, the paraunitary constraint was not accurately verified in these techniques, especially at low SNR.

In the second section, model and hypotheses are presented. The third section introduces the parametrization of MIMO paraunitary filters, and states input-output cumulant relations. The fourth section proposes a contrast for finding equalizer parameters, in order to separate signals. The fifth section presents an iterative algorithm built from previous theoretical results. This algorithm is a Jacobi-type iteration, in which all possible rotation pairs in the parametrization of the paraunitary filter are swept until convergence. A different reasoning leads to a second technique in which the various matrices in the parametrization of the paraunitary filter are estimated one after the other in a non-iterative way. The output of this second algorithm may be used to initialize the first algorithm, consisting of a more exhaustive but computationally more expensive search. The second algorithm is presented in the sixth section. The performance for various Signal-to-Noise Ratios is illustrated in the last section.

## II. MODEL AND NOTATIONS

Throughout the paper,  $(^T)$  stands for transposition,  $(^H)$  for conjugate transposition,  $(^*)$  for complex conjugation, and  $j = \sqrt{-1}$ . Vectors and matrices are denoted with bold lowercase and bold uppercase letters respectively. Also denote by  $\mathbb{Z}$  the set of integers, and by  $\mathbb{N}$  the subset of positive integers.

Considering a digital communication system in a multipath environment, we define the MIMO model depicted in figure 1. Let  $\{\mathbf{G}(m), m \in \mathbb{Z}\}$  denote the matrix impulse response of the global system. Then, its transfer function is

$$\mathbf{G}[z] \stackrel{\text{def}}{=} \sum_m \mathbf{G}(m)z^{-m}. \quad (1)$$

The entries of matrix  $\mathbf{G}$  are denoted  $G_{ij}$ , where subscript  $ij$  denotes the  $i$ -th row and the  $j$ -th column of  $\mathbf{G}$ .

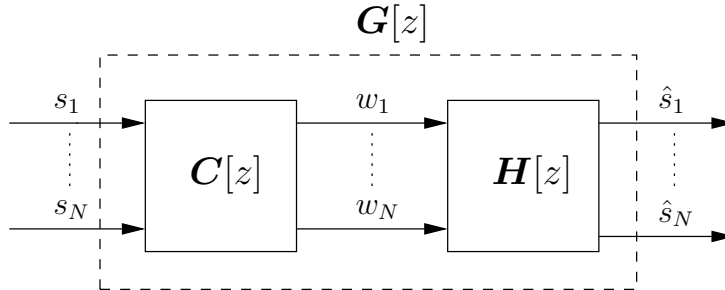


Fig. 1. Global system  $\mathbf{G}$ : sources  $s_i$  are filtered by channel  $\mathbf{C}[z]$  and observations  $w_i$  are equalized by  $\mathbf{H}[z]$ .

Now, consider the linear time-invariant (LTI) invertible system  $\mathbf{C}[z]$  of length  $K$ , mixing  $N$  white random processes ( $N$ : number of antennas at the transmitter and the receiver). This system, *i.e.* the *channel*, is described by

$$\mathbf{w}(n) = \sum_{k=0}^K \mathbf{C}(k)\mathbf{s}(n-k) \quad (2)$$

where  $\{\mathbf{C}(k), k = 0, \dots, K\}$  is a sequence of  $N \times N$  matrices denoting the complex Finite Impulse Response (FIR) of channel  $\mathbf{C}[z]$ ,  $\mathbf{s} = (s_1, \dots, s_N)^\top$  denotes the  $N$  – dimensional source vector of baseband complex signals,  $\mathbf{w} = (w_1, \dots, w_N)^\top$  the  $N$  – dimensional observation vector, and  $\hat{\mathbf{s}} = (\hat{s}_1, \dots, \hat{s}_N)^\top$  the  $N$  – dimensional estimated source vector. All these vectors are *spatially* and *temporally* white at *second order*.

**Remark 1:** Case  $K = 0$ , corresponding to an instantaneous mixture, is not considered here. Thus, throughout this paper, we assume most of the time that  $K > 0$ .

The multichannel blind deconvolution problem consists of finding a LTI filter  $\mathbf{H}[z]$ , *i.e.* the *equalizer*, in order to retrieve the  $N$  input signals  $s_i(n), i \in \{1, \dots, N\}, \forall n \in \mathbb{N}$ , solely from the observation of the outputs  $\mathbf{w}(n)$  of the unknown LTI channel  $\mathbf{C}[z]$ . The signals recovered can be reordered by a permutation matrix  $\mathbf{P}$ , and delayed by a diagonal filter  $\mathbf{\Lambda}[z]$ , so that  $\mathbf{C}[z]\mathbf{H}[z] = \mathbf{\Lambda}[z]\mathbf{P}$ . The estimated source vector

is defined by

$$\hat{\mathbf{s}}(n) = \sum_{l=0}^L \mathbf{H}(l) \mathbf{w}(n-l) \quad (3)$$

where  $L$  is the length of filter  $\mathbf{H}[z]$ . Hence the global LTI system  $\mathbf{G}[z]$  is defined according to:

$$\hat{\mathbf{s}}(n) = \sum_{m=0}^{K+L} \mathbf{G}(m) \mathbf{s}(n-m). \quad (4)$$

**Definition 1: Paraunitarity.** A polynomial matrix  $\mathbf{H}[z] \in \mathbb{C}^{N \times N}$  is said to be paraunitary [15] if

$$\mathbf{H}^{\text{H}}[1/z^*] \mathbf{H}[z] = \mathbf{I}_N \quad (5)$$

where  $\mathbf{I}_N \in \mathbb{R}^{N \times N}$  is the identity matrix. Because  $\mathbf{H}[z]$  is square, one can use the equivalent definition

$$\mathbf{H}[z] \mathbf{H}^{\text{H}}[1/z^*] = \mathbf{I}_N. \quad (6)$$

In this paper, we assume the following hypotheses:

- H1)** Inputs  $s_i(n), \forall i \in \{1, \dots, N\}, \forall n \in \mathbb{N}$ , are mutually independent and identically distributed (i.i.d.) zero-mean random processes, with unit variance.
- H2)** Vector  $\mathbf{s}(n)$  is stationary up to the considered order  $r, r \geq 3$ , *i.e.*  $\forall i \in \{1, \dots, N\}$ , the order- $r$  marginal cumulants,

$$C_p^q[s_i] = \text{Cum} \left[ \underbrace{s_i(n), \dots, s_i(n)}_{p \text{ terms}}, \underbrace{s_i^*(n), \dots, s_i^*(n)}_{q=r-p \text{ terms}} \right]$$

do not depend on  $n$ . For definitions of cumulants, refer to [20] and references therein.

- H3)** At most one source has a zero marginal cumulant of order  $r$ .
- H4)**  $\mathbf{C}[z]$ ,  $\mathbf{H}[z]$ , and hence  $\mathbf{G}[z] = \mathbf{H}[z] \mathbf{C}[z]$  are all paraunitary. Thus we have the global relation:

$$\mathbf{H}[z] \mathbf{C}[z] \mathbf{C}^{\text{H}}[1/z^*] \mathbf{H}^{\text{H}}[1/z^*] = \mathbf{I}_N. \quad (7)$$

**Remark 2:** The constraint of hypothesis **H4** is not restrictive. Indeed, one can always whiten the observations by using a filter that factorizes the second-order power spectrum, *i.e.* a classical prewhitening of the observations [17], [16]. Thus, paraunitary filters can be obtained by space-time standardization of observations (second order white with unit covariance). This issue is not addressed in the paper.

### III. EQUALIZATION

We introduce in this section the global parametrization of the paraunitary equalizer. Then, we can implement it in terms of Givens rotations.

### A. Parametrization

Considering the previous hypotheses and models of section 2, we can make a first proposition:

**Proposition 1:** Let  $\mathbf{H}[z] \in \mathbb{C}^{N \times N}$  be a FIR paraunitary filter of length  $L \geq 0$ . The matrix  $\mathbf{H}[z]$  can be factorized (non uniquely) as the product depicted in figure 2

$$\mathbf{H}[z] = \mathbf{A}[z]\mathbf{Q}\mathbf{B}[z] \quad (8)$$

where  $\mathbf{A}[z]$  and  $\mathbf{B}[z]$  are FIR paraunitary filters of length  $\ell_A$  and  $\ell_B$  respectively, with  $0 \leq \ell_A \leq L$ ,  $0 \leq \ell_B \leq L$ ,  $\ell_A + \ell_B = L$ , and  $\mathbf{Q} \in \mathbb{C}^{N \times N}$  is unitary.

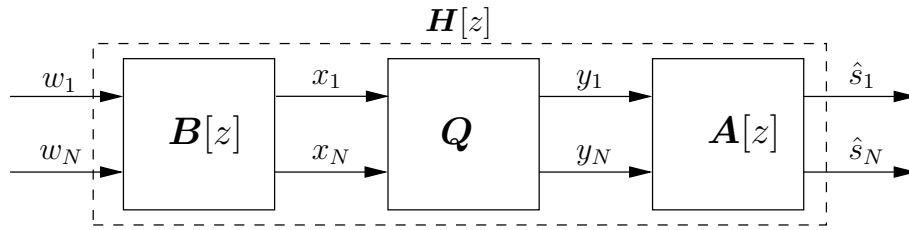


Fig. 2. Factorization of a paraunitary equalizer.

**PROOF 1:** The proof is trivial since we can extend the paraunitary factorization of [21] to the  $N$ -dimensional complex case:

$$\mathbf{H}[z] = \mathbf{Q}_{\lambda_h} \mathbf{Z}[z] \mathbf{Q}_{\lambda_h-1} \dots \mathbf{Z}[z] \mathbf{Q}_1 \mathbf{Z}[z] \mathbf{Q}_0 \quad (9)$$

where  $\mathbf{Q}_p \in \mathbb{C}^{N \times N}$  are unitary for  $p \in \mathbb{P} = \{0, \dots, \lambda_h\}$ ,  $\lambda_h$  is the McMillan degree<sup>1</sup> of  $\mathbf{H}[z]$ , and  $\mathbf{Z}[z] \in \mathbb{C}^{N \times N}$  is diagonal

$$\mathbf{Z}[z] = \begin{bmatrix} \mathbf{I}_{N-1} & 0 \\ 0 & z^{-1} \end{bmatrix}. \quad (10)$$

Thus, for  $0 < \ell_A \leq \lambda_h - p$ , we have

$$\mathbf{A}[z] = \mathbf{Q}_{\lambda_h} \mathbf{Z}[z] \dots \mathbf{Q}_{p+1} \mathbf{Z}[z] \quad (11)$$

and for  $0 < \ell_B \leq p$ , we have

$$\mathbf{B}[z] = \mathbf{Z}[z] \mathbf{Q}_{p-1} \dots \mathbf{Z}[z] \mathbf{Q}_0. \quad (12)$$

When  $\ell_A = 0$  (respectively  $\ell_B = 0$ ), we can replace  $\mathbf{A}[z]$  (respectively  $\mathbf{B}[z]$ ) by  $\mathbf{I}_N$ . Then, for a fixed  $p$ , any paraunitary filter of length  $L$  can be factorized like in (8).  $\diamond$

<sup>1</sup>The McMillan degree of a paraunitary filter is also the degree of its determinant [15, p. 737]. Thus, we have  $\lambda_h \geq L$ .

### B. MIMO relations

In the remaining, we assume the following notation for fourth order cumulants, e.g. cumulants of vector  $\mathbf{w}$ ,

$$\Gamma_{eg, fh}^{\mathbf{w}}(\boldsymbol{\nu}) = \text{Cum}[w_e(n - \nu_1), w_f^*(n - \nu_2), w_g(n - \nu_3), w_h^*(n - \nu_4)] \quad (13)$$

where  $e, f, g, h$  take their values in  $\{1, \dots, N\}$ , and  $\nu_i \in \mathbb{N}, \forall i \in \{1, \dots, 4\}$ .

Now, consider the following input-output relations for the convolutive model

$$\hat{s}_i(n) = \sum_{q,r,m} A_{iq}(m) Q_{qr} x_r(n - m), \quad (14)$$

and

$$x_r(n - m) = \sum_{s,l} B_{rs}(l) w_s(n - m - l). \quad (15)$$

From (14) and thanks to the multilinearity property of cumulants, we can express the input-output relations between cumulants of input  $\mathbf{x}$  and output  $\hat{\mathbf{s}}$ ,

$$\Gamma_{ik, jl}^{\hat{\mathbf{s}}} = \sum_{abcd} \sum_{\boldsymbol{\tau}} \sum_{qrst} A_{iq}(\tau_1) A_{jr}^*(\tau_2) A_{ks}(\tau_3) A_{lt}^*(\tau_4) Q_{qa} Q_{rb}^* Q_{sc} Q_{td}^* \Gamma_{ac, bd}^{\mathbf{x}}(\boldsymbol{\tau}) \quad (16)$$

with  $\boldsymbol{\tau} = (\tau_1, \tau_2, \tau_3, \tau_4)$ . Since  $\hat{\mathbf{s}}$  is the output of filter  $\mathbf{A}[z]$ , the range of each  $\tau_i$  is  $[0, \dots, \ell_A]$  and all indices  $a, b, c, d, i, j, k, l, q, r, s, t$  take their values in  $\{1, \dots, N\}$ . Then, we can write a similar relation between cumulants of input  $\mathbf{w}$  and output  $\mathbf{x}$ ,

$$\Gamma_{ac, bd}^{\mathbf{x}}(\boldsymbol{\tau}) = \sum_{\boldsymbol{\rho}} \sum_{efgh} B_{ae}(\rho_1) B_{bf}^*(\rho_2) B_{cg}(\rho_3) B_{dh}^*(\rho_4) \Gamma_{eg, fh}^{\mathbf{w}}(\boldsymbol{\tau} + \boldsymbol{\rho}) \quad (17)$$

with  $\boldsymbol{\rho} = (\rho_1, \rho_2, \rho_3, \rho_4)$ . Here, the range of each  $\rho_i$  is  $[0, \dots, \ell_B]$  since inputs are filtered by  $\mathbf{B}[z]$ . The global input-output relation of equalizer  $\mathbf{H}[z]$  can be easily deduced by combining (16) and (17) as shown in [22].

### C. Construction of $\mathbf{Q}_p$

Proposition 1 told us that all  $\mathbf{Q}_p$  have to be  $N \times N$  unitary in order to obtain a paraunitary equalizer. It is well known that any  $N \times N$  unitary matrix can be written as a product of  $M = N(N - 1)/2$  Givens rotations [23, p. 215]. The  $M$  rotations describe all possible pairs  $(i, j)$  with  $1 \leq i < j \leq N$ . Hence,

denote  $\mathbf{U}[i, j](\theta, \phi)$  the Givens rotation acting on pair  $(i, j)$  and defined by angles  $\theta$  and  $\phi$ :

$$\mathbf{U}[i, j](\theta, \phi) = \begin{bmatrix} \mathbf{I}_{n_1} & \vdots & \mathbf{0} & \vdots & \mathbf{0} \\ \cdots & \cos \theta[i, j] & \cdots & \sin \theta[i, j]e^{j\phi[i, j]} & \cdots \\ \mathbf{0} & \vdots & \mathbf{I}_{n_2} & \vdots & \mathbf{0} \\ \cdots & -\sin \theta[i, j]e^{-j\phi[i, j]} & \cdots & \cos \theta[i, j] & \cdots \\ \mathbf{0} & \vdots & \mathbf{0} & \vdots & \mathbf{I}_{n_3} \end{bmatrix} \begin{matrix} \\ i \\ \\ j \\ \end{matrix}$$

where  $n_1 + n_2 + n_3 = N - 2$ . These unitary rotations permit to develop a *Jacobi*-type iteration algorithm for equalizer (8).

Since we use Givens matrices, we can make 2 remarks:

**Remark 3:** For  $N = 2$ , unitary matrix  $\mathbf{Q}_p$  consists of a single rotation

$$\mathbf{Q}_p = \begin{bmatrix} \cos \theta_p & \sin \theta_p e^{j\phi_p} \\ -\sin \theta_p e^{-j\phi_p} & \cos \theta_p \end{bmatrix}. \quad (18)$$

Then, only one pair of angles  $(\theta_p, \phi_p)$  is needed for each  $\mathbf{Q}_p, p \in \mathbb{P}$ . Hence, when more than two signals are observed, it is also possible to use a *deflation*-type algorithm [24].

**Remark 4:** Proposition 1 tells that  $\lambda_h + 1$  unitary matrices are used to obtain a complete factorization of the equalizer. For instance, the  $2 \times 2$  length-1 paraunitary filter defined by  $\text{diag}(\mathbf{H}[z]) = [-z^{-1}, z^{-1}]$  and 0 elsewhere, needs  $\lambda_h = \text{deg}[\det(\mathbf{H}[z])] = 2$  diagonal matrices  $\mathbf{Z}[z]$ . Indeed,  $\mathbf{H}[z]$  has the following factorization

$$\mathbf{H}[z] = \mathbf{Q}_2 \cdot \mathbf{Z}[z] \cdot \mathbf{Q}_1 \cdot \mathbf{Z}[z] \cdot e^{j\frac{\pi}{2}} \quad (19)$$

where  $\mathbf{Q}_2 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$  and  $\mathbf{Q}_1 = \begin{bmatrix} 0 & j \\ j & 0 \end{bmatrix}$ , i.e.  $(\theta_2 = \frac{\pi}{2}, \phi_2 = 0)$  and  $(\theta_1 = \frac{\pi}{2}, \phi_1 = \frac{\pi}{2})$ .

The exponential is part of the unknown matrix  $\mathbf{\Lambda}[z]\mathbf{P}$ , as explained after remark 1. It is also true for  $L \geq 1$  since we can group all exponentials in a single matrix  $\mathbf{\Lambda}[z]\mathbf{P}$ . Thus, reflections, e.g.  $\text{diag}(\mathbf{Q}) = [1, -1]$ , are covered by the parametrization of proposition 1, thanks to the inherent indeterminacy. The parametrization (8) is thus complete.

#### IV. OPTIMIZATION CRITERION

Let us start this section with two key definitions. The first definition deals with the set of filters that do not affect the independence of sources, a key point of blind source separation methods. The second definition describes the main properties of a *contrast*, i.e. an optimization criterion for separating signals.

**Definition 2: Trivial filters.** The set  $\mathcal{S}$  of source processes is characterized by assumptions, such as **H1**. One defines the set  $\mathcal{T}$  of *trivial filters*, as containing all filters that do not affect these assumptions. In other words,  $\mathcal{S}$  is stable by the operation of  $\mathcal{T}$ . For instance, filters of the form  $\mathbf{\Lambda}[z]\mathbf{P}$ , where  $\mathbf{P}$  is a permutation matrix, and  $\mathbf{\Lambda}[z]$  a diagonal filter, do not affect mutual independence between components of  $\mathbf{s}(n)$ . If in addition  $\mathbf{s}(n)$  is an i.i.d. non Gaussian process,  $\mathbf{\Lambda}[z]$  should contain only pure delays, integer multiples of the sampling period, and fixed complex factors; in other words, the entries of  $\mathbf{\Lambda}[z]$  are of the form  $\gamma z^k$ , with  $k \in \mathbb{Z}$  and  $|\gamma| = 1$ .

**Definition 3: Contrast.** Let  $\mathcal{H}$  be a set of filters, and denote  $\mathcal{H} \cdot \mathcal{S}$  the set of standardized linear processes obtained by operation of filters of  $\mathcal{H}$  on processes of  $\mathcal{S}$ . Denote  $\mathbf{I}$  the identity matrix. An optimization criterion,  $\Upsilon(\mathbf{H}; \mathbf{w})$ , will be referred to as a contrast defined on  $\mathbf{H} \in \mathcal{H}, \mathbf{w} \in \mathcal{H} \cdot \mathcal{S}$ , if it satisfies the three properties below [1]:

- **Invariance:** The contrast should not change within the set of acceptable solutions, which means that  $\forall \mathbf{w} \in \mathcal{H} \cdot \mathcal{S}, \forall \mathbf{H} \in \mathcal{T}$  then  $\Upsilon(\mathbf{H}; \mathbf{w}) = \Upsilon(\mathbf{I}; \mathbf{w})$ .
- **Domination:** If sources are already separated, any filter should decrease the contrast. In other words,  $\forall \mathbf{w} \in \mathcal{S}, \forall \mathbf{H} \in \mathcal{H}$ , then  $\Upsilon(\mathbf{H}; \mathbf{w}) \leq \Upsilon(\mathbf{I}; \mathbf{w})$ .
- **Discrimination:** The maximum contrast should be reached only for filters linked to each other via trivial filters:  $\forall \mathbf{w} \in \mathcal{S}, \Upsilon(\mathbf{H}; \mathbf{w}) = \Upsilon(\mathbf{I}; \mathbf{w}) \Rightarrow \mathbf{H} \in \mathcal{T}$ .

Now, let us focus on the contrast used to carry out the equalization of the system. We can separate signals blindly by using an approach based on *high order statistics*. Indeed, since fourth-order cross-cumulants are a good measure of statistical independence, we can build a contrast based on these values [1]:

**Proposition 2:** Let us consider sources with kurtosis of the same sign,  $\epsilon = \pm 1$ , the separation of of theses sources solely from outputs of the channel can be performed by maximizing the following contrast

$$\Upsilon_{1,4} = \epsilon \sum_{i=1}^N \Gamma_{ii,ii}^{\hat{\mathbf{s}}} \quad (20)$$

where cumulants of  $\hat{\mathbf{s}}$  are defined by

$$\Gamma_{ii,ii}^{\hat{\mathbf{s}}} = \sum_{abcd} \sum_{\boldsymbol{\tau}} \sum_{qrst} A_{iq}(\tau_1) A_{ir}^*(\tau_2) A_{is}(\tau_3) A_{it}^*(\tau_4) Q_{qa} Q_{rb}^* Q_{sc} Q_{td}^* \Gamma_{ac,bd}^{\mathbf{x}}(\boldsymbol{\tau}). \quad (21)$$

Hence, in order to estimate  $\mathbf{H}[z]$ , the criterion can be written

$$\mathbf{H} = \underset{\mathbb{Q}}{\text{Arg max}} \Upsilon_{1,4} \quad (22)$$

where  $\mathbb{Q}$  stands for the set of  $\mathbf{Q}_p$ ,  $p \in \mathbb{P}$ .

It has been proved in [1] that  $\Upsilon_{1,4}$  is a contrast. We maximize  $\Upsilon_{1,4}$  with respect to each pair  $(\theta, \phi)$  in turn. The sequence of values of  $\Upsilon_{1,4}$  obtained this way is monotonically increasing. Since it is also bounded above, it converges (see figure 3).

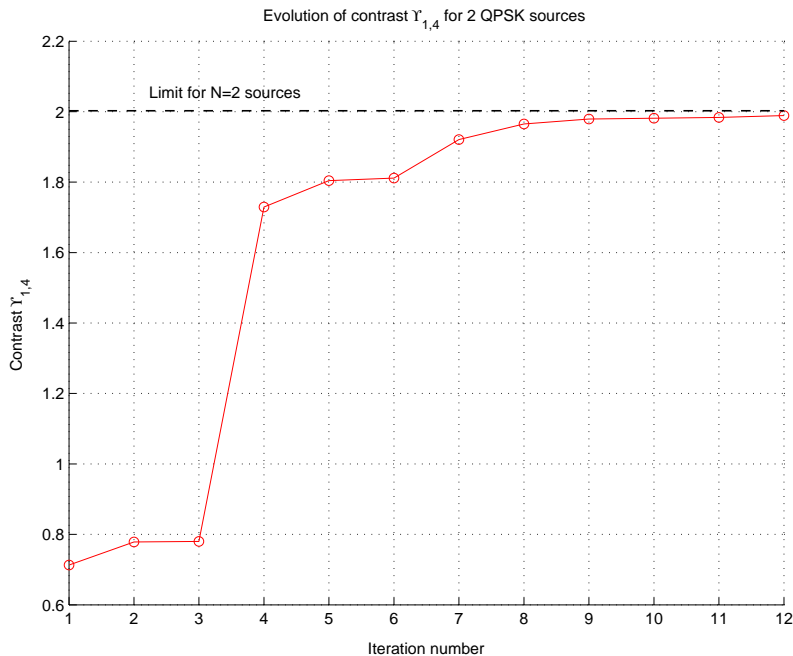


Fig. 3. Typical evolution of contrast  $\Upsilon_{1,4}$  for 25dB of SNR and length-3 equalizer.

## V. A JACOBI-TYPE ALGORITHM

In this section we detail an iteration of the algorithm for one pair  $(\theta, \phi)$  of Givens rotation. Then, as a *Jacobi*-type algorithm, we sweep on  $M = N(N - 1)/2$  pairs for each unitary matrix  $\mathbf{Q}_p$ .

### A. Processing a pair of outputs

For the sake of clarity, we drop index  $p$ . We have to find all pairs  $(\theta, \phi)$ , which maximize (20) independently from other pairs. To reach this goal, we have to simplify (21) firstly by expanding it, and

secondly by collecting terms involving  $\theta$  or  $\phi$ . In this manner, we obtain the output cumulants of  $\hat{\mathbf{s}}$ ,

$$\begin{aligned}
\Gamma_{ii,ii}^{\hat{\mathbf{s}}} &= \mathcal{K}_{(4)}^{(0)} \cos^4 \theta + \mathcal{K}_{(3)}^{(1)} \cos^3 \theta \sin \theta e^{j\phi} + \mathcal{K}_{(3)}^{(-1)} \cos^3 \theta \sin \theta e^{-j\phi} + \mathcal{K}_{(2)}^{(0)} \cos^2 \theta \sin^2 \theta \\
&+ \mathcal{K}_{(2)}^{(2)} \cos^2 \theta \sin^2 \theta e^{2j\phi} + \mathcal{K}_{(2)}^{(-2)} \cos^2 \theta \sin^2 \theta e^{-2j\phi} + \mathcal{K}_{(1)}^{(1)} \cos \theta \sin^3 \theta e^{j\phi} \\
&+ \mathcal{K}_{(1)}^{(-1)} \cos \theta \sin^3 \theta e^{-j\phi} + \mathcal{K}_{(1)}^{(3)} \cos \theta \sin^3 \theta e^{3j\phi} + \mathcal{K}_{(1)}^{(-3)} \cos \theta \sin^3 \theta e^{-3j\phi} \\
&+ \mathcal{K}_{(0)}^{(0)} \sin^4 \theta + \mathcal{K}_{(0)}^{(4)} \sin^4 \theta e^{4j\phi} + \mathcal{K}_{(0)}^{(-4)} \sin^4 \theta e^{-4j\phi} \\
&+ \mathcal{K}_{(0)}^{(2)} \sin^4 \theta e^{2j\phi} + \mathcal{K}_{(0)}^{(-2)} \sin^4 \theta e^{-2j\phi}.
\end{aligned} \tag{23}$$

where

$$\mathcal{K}_{(\alpha)}^{(\eta)} \cos^\alpha \theta \sin^{4-\alpha} \theta e^{j\eta\phi} = \sum_{\boldsymbol{\tau}} A_{iq}(\tau_1) A_{ir}^*(\tau_2) A_{is}(\tau_3) A_{it}^*(\tau_4) \Gamma_{ac,bd}^{\mathbf{x}}(\boldsymbol{\tau}) \cos^\alpha \theta \sin^{4-\alpha} \theta e^{j\eta\phi}. \tag{24}$$

Each  $\mathcal{K}_{(\alpha)}^{(\eta)}$  depends on indices  $a, b, c, d$  and  $q, r, s, t$  since the product  $Q_{qa} Q_{rb}^* Q_{sc} Q_{td}^*$  gives an expression of the form

$$Q_{qa} Q_{rb}^* Q_{sc} Q_{td}^* \rightarrow \cos^\alpha \theta \sin^{4-\alpha} \theta e^{j\eta\phi} \tag{25}$$

with  $\alpha \in \{0, \dots, 4\}$  and  $\eta \in \{\pm 1, \pm 2, \pm 3, \pm 4\}$ , as detailed in Appendix-A. Next, with  $\eta = 2\beta + \alpha - 4$ , we obtain a simpler expression

$$\Gamma_{ii,ii}^{\hat{\mathbf{s}}} = \sum_{\alpha=0}^4 \sum_{\beta=0}^{4-\alpha} \mathcal{K}_{(\alpha)}^{(2\beta+\alpha-4)} \cos^\alpha \theta \sin^{4-\alpha} \theta e^{j(2\beta+\alpha-4)\phi}. \tag{26}$$

**Remark 5:** Computing each  $\Gamma_{ii,ii}^{\hat{\mathbf{s}}}$  without simplifications require a total of  $(L+1)^4 N^8$  loops (equation (21)). Thus, readers can find an implementation trick in Appendix-B for reducing this number of loops to  $((L+1)N)^4$ . Moreover, the method detailed in Appendix-B uses less memory for storing matrices. Only the cumulant tensor  $\Gamma_{ac,bd}^{\mathbf{x}}(\boldsymbol{\tau})$ , which is computed at the beginning of each loop, needs a large memory, *i.e.* up to  $N^4(L+1)^4$  values to store.

Next, we make the classical change of variables:  $\cos \phi = \frac{1-t^2}{1+t^2}$ ,  $\sin \phi = \frac{2t}{1+t^2}$  with  $t = \tan \frac{\phi}{2}$ , and  $\cos \theta = \frac{1}{\sqrt{1+u^2}}$ ,  $\sin \theta = \frac{u}{\sqrt{1+u^2}}$  with  $u = \tan \theta$ . Thus, the numerator and the denominator of the polynomial obtained are both of degree 8 in variable  $t$  and 4 in variable  $u$ . For maximizing contrast (20), we have to find roots of polynomial system (27), *i.e.* stationary points of  $\Upsilon_{1,4}$ , obtained thanks to derivatives

$$\left. \begin{aligned} \Phi_1(u, t) &\stackrel{\text{def}}{=} \frac{\partial \Upsilon_{1,4}}{\partial u} \\ \Phi_2(u, t) &\stackrel{\text{def}}{=} \frac{\partial \Upsilon_{1,4}}{\partial t} \end{aligned} \right\} \tag{27}$$

After some simplifications, *e.g.* obvious root  $u = 0$  in  $\Phi_2(u, t)$ , polynomials obtained have different global degrees: degree 12 for  $\Phi_1(u, t)$  (leading monomial is  $t^8 u^4$ ), and degree 11 for  $\Phi_2(u, t)$  (leading monomial is  $t^8 u^3$ ). Then we consider only variable  $u$  for  $\Phi_1(u, t)$  and  $\Phi_2(u, t)$ , and we collect terms of same degree in  $u$ . Hence, system (27) can be rewritten

$$\left. \begin{aligned} \Phi_1(u, t) &= \sum_{k=0}^4 \chi_{4-k}(t) u^k \\ \Phi_2(u, t) &= \sum_{k=0}^3 \xi_{3-k}(t) u^k \end{aligned} \right\} \quad (28)$$

The polynomial system (28) admits a solution if and only if the resultant, *i.e.* determinant of a Sylvester matrix, is null [25]. Thus, we obtain the determinant to solve

$$\begin{vmatrix} \xi_0(t) & 0 & 0 & 0 & \chi_0(t) & 0 & 0 \\ \xi_1(t) & \xi_0(t) & 0 & 0 & \chi_1(t) & \chi_0(t) & 0 \\ \xi_2(t) & \xi_1(t) & \xi_0(t) & 0 & \chi_2(t) & \chi_1(t) & \chi_0(t) \\ \xi_3(t) & \xi_2(t) & \xi_1(t) & \xi_0(t) & \chi_3(t) & \chi_2(t) & \chi_1(t) \\ 0 & \xi_3(t) & \xi_2(t) & \xi_1(t) & \chi_4(t) & \chi_3(t) & \chi_2(t) \\ 0 & 0 & \xi_3(t) & \xi_2(t) & 0 & \chi_4(t) & \chi_3(t) \\ 0 & 0 & 0 & \xi_3(t) & 0 & 0 & \chi_4(t) \end{vmatrix} = 0. \quad (29)$$

It turns out that this polynomial is of degree 56 and it generally admits no more than 16 real roots. When all real roots are found, we have to plug them back in (23) for selecting the best solution for (22), *i.e.* leading to the global maximum. It is not that hard to compute since powerful algorithms exist that can rapidly find roots of univariate polynomials.

Thus, we obtain:

Algorithm 1:

**compute**  $\Gamma_{eg, fh}^w(\tau + \rho)$   
**set**  $M := N(N - 1)/2$  and  $T := \lceil \sqrt{L} \rceil + 1$   
**set**  $t := 1..T$  and  $k := 0..M\lambda_h$   
**initialize**  $\forall k: \theta_k = 0, \phi_k = 0$   
**for each**  $t$  **do**  
  **for each**  $k$  **do**  
    **compute**  $\Gamma_{ac, bd}^x(\tau)$   
    **search**  $\theta_k$  and  $\phi_k$  maximizing  $\Upsilon_{1,4}$   
    **actualize** filters  $Q_p, A[z]$  and  $B[z]$

**return**  $\mathbf{H}[z]$

### B. Sweeping the pairs

Let us comment some points of this algorithm:

- the resulting tensor  $\Gamma_{ac,bd}^{\mathbf{x}}(\boldsymbol{\tau})$  is composed of  $N^2(L - \ell_B + 1)^2$  matrices each of size  $N(L - \ell_B + 1) \times N(L - \ell_B + 1)$ . Hence, the algorithm needs to store  $N^4(L - \ell_B + 1)^4$  elements in memory. Simplifications are generally impossible since we need exact values for computing roots of polynomial system (28),
- matrix  $\mathbf{U}_k$  corresponds to some matrix  $\mathbf{U}[i, j](\theta, \phi)$  with appropriate indices as presented in section III-C,
- In order to increase the precision of the angles, we suggest to execute  $T = \lceil \sqrt{L} \rceil + 1$  sweeps. Actually, the first angles computed are not well defined since all other angles are still null (set at stage **initialize**). Hence, when loop on each  $k$  is repeated several times, angles are better estimated. This procedure reminds the well known Jacobi sweeping widely used for eigenvalue computation.

**Remark 6:** In the case of a *slow* linear time variant channel (also considered as invariant in time for small periods), the initialization step **initialize** can be modified since angles previously found give a better approximation than null values.

## VI. A NON-ITERATIVE ALGORITHM

### A. Principle

The previous algorithm directly aimed at estimating the equalizer taps. In the current section we will first identify the channel and then invert it. Suppose the channel admits the paraunitary factorization

$$\mathbf{C}[z] = \mathbf{U}_{\lambda_c} \mathbf{Z}[z] \mathbf{U}_{\lambda_c-1} \dots \mathbf{Z}[z] \mathbf{U}_1 \mathbf{Z}[z] \mathbf{U}_0, \quad (30)$$

in which  $\mathbf{U}_r \in \mathbb{C}^{N \times N}$  are unitary for  $r \in \{0, \dots, \lambda_c\}$ , where  $\lambda_c$  is the McMillan degree of  $\mathbf{C}[z]$ , and  $\mathbf{Z}[z]$  is defined in Eq. (10). The inverse of the channel is given by

$$\mathbf{C}[z]^{-1} = \mathbf{U}_0^H \mathbf{Z}[z]^{-1} \mathbf{U}_1^H \mathbf{Z}[z]^{-1} \dots \mathbf{U}_{\lambda_c-1}^H \mathbf{Z}[z]^{-1} \mathbf{U}_{\lambda_c}^H.$$

Hence, once matrices  $\mathbf{U}_r$  have been determined, a causal equalizer is given by

$$z^{-\lambda_c} \mathbf{C}[z]^{-1} = \mathbf{U}_0^H z^{-1} \mathbf{Z}[z]^{-1} \mathbf{U}_1^H z^{-1} \mathbf{Z}[z]^{-1} \dots \mathbf{U}_{\lambda_c-1}^H z^{-1} \mathbf{Z}[z]^{-1} \mathbf{U}_{\lambda_c}^H. \quad (31)$$

The paraunitary factorization of  $z^{-1} \mathbf{Z}[z]^{-1}$  is straightforward, cf. Remark 4.

Before introducing a technique for the blind identification of the channel, let us have a closer look at the inherent indeterminacies of parametrization (30). It is clear that each matrix  $\mathbf{U}_r$ , for  $r \in \{1, \dots, \lambda_c\}$  may be multiplied from the right by a matrix

$$\mathbf{V}_r = \begin{bmatrix} \mathbf{W}_r & 0 \\ 0 & w_r \end{bmatrix}, \quad (32)$$

with  $\mathbf{W}_r \in \mathbb{C}^{(N-1) \times (N-1)}$  unitary and  $|w_r| = 1$ , provided  $\mathbf{U}_{r-1}$  is multiplied from the left by  $\mathbf{V}_r^H$ . Hence it is sufficient to subsequently estimate the last column of each  $\mathbf{U}_r$  (modulo a phase rotation), for  $r$  ranging from  $\lambda_c$  to 1, and finally estimate matrix  $\mathbf{U}_0$ .

First define some notation. Let the output of the system  $\mathbf{U}_k \dots \mathbf{Z}[z]\mathbf{U}_1\mathbf{Z}[z]\mathbf{U}_0$  be represented by  $\mathbf{s}^{(k)}$  and let  $\tilde{\mathbf{s}}^{(k)}$  denote the output of  $\mathbf{Z}[z]\mathbf{U}_k \dots \mathbf{Z}[z]\mathbf{U}_1\mathbf{Z}[z]\mathbf{U}_0$ . Due to the stationarity of the signals, we can write

$$\Gamma_{eg, fh}^{\mathbf{s}}(\boldsymbol{\nu}) = \text{Cum}[s_e(n - \nu_1), s_f^*(n - \nu_2), s_g(n - \nu_3), s_h^*(n - \nu_4)] \quad (33)$$

$$= \text{Cum}[s_e(n), s_f^*(n - \mu_1), s_h^*(n - \mu_2), s_g(n - \mu_3)] \quad (34)$$

$$= \Gamma_{eg, fh}^{\mathbf{s}}(\boldsymbol{\mu}), \quad (35)$$

in which  $\mu_1 = \nu_2 - \nu_1$ ,  $\mu_2 = \nu_4 - \nu_1$ ,  $\mu_3 = \nu_3 - \nu_1$ .

A multiplication with  $\mathbf{Z}[z]$  makes the region of support of the fourth-order cumulants grow by one delay. For instance, the boundary  $-k \leq \mu_1 \leq k$  for  $\Gamma^{\mathbf{s}^{(k)}}(\boldsymbol{\mu})$  is replaced, for  $\Gamma^{\tilde{\mathbf{s}}^{(k)}}(\boldsymbol{\mu})$ , by the boundary  $-k - 1 \leq \mu_1 \leq k + 1$ . A crucial observation is that the new cumulants that are formed in the borders of the  $\boldsymbol{\mu}$ -domain, are not full tensors; instead, most of their entries are zero. Let us, by way of example, consider the region  $0 \leq \mu_1 \leq \mu_2 \leq \mu_3$  (which is the principal region for real-valued data). (For complex-valued data, and considering the cumulant symmetry (34), the principal domain consists of the union of the regions defined by  $0 \leq \mu_1 \leq \mu_2 \leq \mu_3$ ,  $0 \leq \mu_1 \leq \mu_3 \leq \mu_2$  and  $0 \leq \mu_3 \leq \mu_2 \leq \mu_1$ .) Multiplication with  $\mathbf{Z}[z]$  results in the following substitutions:

$$\Gamma_{\overline{NN}, \overline{NN}}^{\tilde{\mathbf{s}}^{(k)}}(k + 1, k + 1, k + 1) \leftarrow \Gamma_{\overline{NN}, \overline{NN}}^{\mathbf{s}^{(k)}}(k, k, k), \quad (36)$$

$$\Gamma_{\overline{NN}, \overline{NN}}^{\tilde{\mathbf{s}}^{(k)}}(\mu_1, k + 1, k + 1) \leftarrow \Gamma_{\overline{NN}, \overline{NN}}^{\mathbf{s}^{(k)}}(\mu_1 - 1, k, k) \quad (0 < \mu_1 < k + 1), \quad (37)$$

$$\Gamma_{\overline{NN}, \overline{NN}}^{\tilde{\mathbf{s}}^{(k)}}(\mu_1, k + 1, k + 1) \leftarrow \Gamma_{\overline{NN}, \overline{NN}}^{\mathbf{s}^{(k)}}(\mu_1, k, k) \quad (0 < \mu_1 < k + 1), \quad (38)$$

$$\Gamma_{\overline{NN}, \overline{NN}}^{\tilde{\mathbf{s}}^{(k)}}(0, k + 1, k + 1) \leftarrow \Gamma_{\overline{NN}, \overline{NN}}^{\mathbf{s}^{(k)}}(0, k, k), \quad (39)$$

$$\Gamma_{\overline{NN},\overline{NN}}^{\tilde{\mathbf{s}}^{(k)}}(\mu_1, \mu_2, k+1) \leftarrow \Gamma_{\overline{NN},\overline{NN}}^{\mathbf{s}^{(k)}}(\mu_1 - 1, \mu_2 - 1, k) \quad (0 < \mu_1 \leq \mu_2 < k+1), \quad (40)$$

$$\Gamma_{\overline{NN},\overline{NN}}^{\tilde{\mathbf{s}}^{(k)}}(\mu_1, \mu_2, k+1) \leftarrow \Gamma_{\overline{NN},\overline{NN}}^{\mathbf{s}^{(k)}}(\mu_1, \mu_2 - 1, k) \quad (0 < \mu_1 \leq \mu_2 < k+1), \quad (41)$$

$$\Gamma_{\overline{NN},\overline{NN}}^{\tilde{\mathbf{s}}^{(k)}}(\mu_1, \mu_2, k+1) \leftarrow \Gamma_{\overline{NN},\overline{NN}}^{\mathbf{s}^{(k)}}(\mu_1 - 1, \mu_2, k) \quad (0 < \mu_1 \leq \mu_2 < k+1), \quad (42)$$

$$\Gamma_{\overline{NN},\overline{NN}}^{\tilde{\mathbf{s}}^{(k)}}(\mu_1, \mu_2, k+1) \leftarrow \Gamma_{\overline{NN},\overline{NN}}^{\mathbf{s}^{(k)}}(\mu_1, \mu_2, k) \quad (0 < \mu_1 \leq \mu_2 < k+1), \quad (43)$$

$$\Gamma_{\overline{NN},\overline{NN}}^{\tilde{\mathbf{s}}^{(k)}}(0, \mu_2, k+1) \leftarrow \Gamma_{\overline{NN},\overline{NN}}^{\mathbf{s}^{(k)}}(0, \mu_2 - 1, k) \quad (0 < \mu_2 < k+1), \quad (44)$$

$$\Gamma_{\overline{NN},\overline{NN}}^{\tilde{\mathbf{s}}^{(k)}}(0, \mu_2, k+1) \leftarrow \Gamma_{\overline{NN},\overline{NN}}^{\mathbf{s}^{(k)}}(0, \mu_2, k) \quad (0 < \mu_2 < k+1), \quad (45)$$

$$\Gamma_{\overline{NN},\overline{NN}}^{\tilde{\mathbf{s}}^{(k)}}(0, 0, k+1) \leftarrow \Gamma_{\overline{NN},\overline{NN}}^{\mathbf{s}^{(k)}}(0, 0, k), \quad (46)$$

in which  $\overline{N} \in \{1, \dots, N-1\}$ .

Due to the multilinearity property, the cumulants of  $\tilde{\mathbf{s}}^{(k)}$  and  $\mathbf{s}^{(k+1)}$  are related by

$$\Gamma_{ef,gh}^{\mathbf{s}^{(k+1)}}(\boldsymbol{\mu}) = \sum_{abcd} (\mathbf{U}_{k+1})_{ea} (\mathbf{U}_{k+1})_{gb}^* (\mathbf{U}_{k+1})_{fc} (\mathbf{U}_{k+1})_{hd}^* \Gamma_{ac,bd}^{\tilde{\mathbf{s}}^{(k)}}(\boldsymbol{\mu}). \quad (47)$$

It turns out that the algebraic structure following from Eqs. (36-46) and (47) allows to estimate the last column of  $\mathbf{U}_{k+1}$ . An algorithm will be described in detail in the next subsection.

The overall procedure for the blind identification of  $\mathbf{C}[z]$  starts then with the estimation of  $\mathbf{U}_{\lambda_c}$ , followed by an inverse time shift, after which  $\mathbf{U}_{\lambda_c-1}$  is estimated, again followed by an inverse time shift, etc. In other words, in each step a layer of sequence (30) is peeled off, until finally only the estimation of  $\mathbf{U}_0$  remains, which is a classical Blind Source Separation (BSS) problem.

### B. Algorithm

Let  $\mathbf{q} \in \mathbb{C}^N$  denote the last column of  $\mathbf{U}_{k+1}$ . Eqs. (36-46) and (47) imply that, in the absence of noise, some subvectors of the cumulant tensors of  $\mathbf{s}^{(k+1)}$  are proportional to  $\mathbf{q}$  (or  $\mathbf{q}^*$ ). Other vectors are known to be linear combinations of the first  $N-1$  columns of  $\mathbf{U}_{k+1}$  (or their complex conjugates) and are therefore orthogonal to  $\mathbf{q}$  (or  $\mathbf{q}^*$ ). We have, for arbitrary  $\alpha, \beta, \gamma$ :

$$\Gamma_{:, \alpha, \beta, \gamma}^{\mathbf{S}^{(k+1)}}(k+1, k+1, k+1) \perp \mathbf{q}, \quad (48)$$

$$\Gamma_{\alpha, \beta, :, \gamma}^{\mathbf{S}^{(k+1)}}(k+1, k+1, k+1) \sim \mathbf{q}^*, \quad (49)$$

$$\Gamma_{\alpha, :, \beta, \gamma}^{\mathbf{S}^{(k+1)}}(k+1, k+1, k+1) \sim \mathbf{q}, \quad (50)$$

$$\Gamma_{:, \alpha, \beta, \gamma}^{\mathbf{S}^{(k+1)}}(\mu_1, k+1, k+1) \perp \mathbf{q} \quad (0 < \mu_1 < k+1), \quad (51)$$

$$\Gamma_{\alpha, \beta, \gamma, :}^{\mathbf{S}^{(k+1)}}(\mu_1, k+1, k+1) \sim \mathbf{q}^* \quad (0 < \mu_1 < k+1), \quad (52)$$

$$\Gamma_{\alpha, :, \beta, \gamma}^{\mathbf{S}^{(k+1)}}(\mu_1, k+1, k+1) \sim \mathbf{q} \quad (0 < \mu_1 < k+1), \quad (53)$$

$$\Gamma_{:, \alpha, \beta, \gamma}^{\mathbf{S}^{(k+1)}}(0, k+1, k+1) \perp \mathbf{q}, \quad (54)$$

$$\Gamma_{\alpha, \beta, :, \gamma}^{\mathbf{S}^{(k+1)}}(0, k+1, k+1) \perp \mathbf{q}^*, \quad (55)$$

$$\Gamma_{\alpha, \beta, \gamma, :}^{\mathbf{S}^{(k+1)}}(0, k+1, k+1) \sim \mathbf{q}^*, \quad (56)$$

$$\Gamma_{\alpha, :, \beta, \gamma}^{\mathbf{S}^{(k+1)}}(0, k+1, k+1) \sim \mathbf{q}, \quad (57)$$

$$\Gamma_{:, \alpha, \beta, \gamma}^{\mathbf{S}^{(k+1)}}(\mu_1, \mu_2, k+1) \perp \mathbf{q} \quad (0 < \mu_1 \leq \mu_2 < k+1), \quad (58)$$

$$\Gamma_{\alpha, :, \beta, \gamma}^{\mathbf{S}^{(k+1)}}(\mu_1, \mu_2, k+1) \sim \mathbf{q} \quad (0 < \mu_1 \leq \mu_2 < k+1), \quad (59)$$

$$\Gamma_{:, \alpha, \beta, \gamma}^{\mathbf{S}^{(k+1)}}(0, \mu_2, k+1) \perp \mathbf{q} \quad (0 < \mu_2 < k+1), \quad (60)$$

$$\Gamma_{\alpha, \beta, :, \gamma}^{\mathbf{S}^{(k+1)}}(0, \mu_2, k+1) \perp \mathbf{q}^* \quad (0 < \mu_2 < k+1), \quad (61)$$

$$\Gamma_{\alpha, :, \beta, \gamma}^{\mathbf{S}^{(k+1)}}(0, \mu_2, k+1) \sim \mathbf{q} \quad (0 < \mu_2 < k+1), \quad (62)$$

$$\Gamma_{:, \alpha, \beta, \gamma}^{\mathbf{S}^{(k+1)}}(0, 0, k+1) \perp \mathbf{q}, \quad (63)$$

$$\Gamma_{\alpha, \beta, :, \gamma}^{\mathbf{S}^{(k+1)}}(0, 0, k+1) \perp \mathbf{q}^*, \quad (64)$$

$$\Gamma_{\alpha, :, \beta, \gamma}^{\mathbf{S}^{(k+1)}}(0, 0, k+1) \sim \mathbf{q}, \quad (65)$$

in which we used the MATLAB colon notation.

Eqs. (48-65) lead to a classical least squares problem in  $\mathbf{q}$ . A condition of the type  $\mathbf{v} \perp \mathbf{q}$  can be expressed as  $\mathbf{v}^H \mathbf{q} = 0$ . Equivalently,  $\mathbf{v} \perp \mathbf{q}^*$  can be expressed as  $\mathbf{v}^T \mathbf{q} = 0$ . By stacking all the orthogonality constraints, we obtain

$$\mathbf{M}_1 \mathbf{q} = \mathbf{0}, \quad (66)$$

in which  $\mathbf{M}_1$  has more rows than columns. The least squares solution of this equation, *i.e.* the vector  $\mathbf{q}$  minimizing

$$f_1(\mathbf{q}) = \|\mathbf{M}_1 \mathbf{q}\|^2 = \mathbf{q}^H \mathbf{M}_1^H \mathbf{M}_1 \mathbf{q}, \quad (67)$$

is given by the right singular vector of  $M_1$  corresponding to the smallest singular value.

Next, we build in the same way a matrix  $M_2$  from the proportionality conditions. For each condition  $v \sim q$  a row vector  $v^H$  is added to  $M_2$ , while each condition  $v \sim q^*$  yields a row  $v^T$ . The proportionality can be imposed by maximizing

$$f_2(q) = \|M_2 q\|^2 = q^H M_2^H M_2 q. \quad (68)$$

The solution that is optimal in least-squares sense, is given by the right singular vector of  $M_2$  corresponding to the largest singular value.

The orthogonality and proportionality constraints can be combined by minimizing  $f_1 - f_2$ . The optimal vector  $q$  is the eigenvector corresponding to the smallest eigenvalue of the positive (semi)definite matrix  $M_1^H M_1 - M_2^H M_2 + \|M_2\|^2 \mathbf{I}$ .

**Example 1** For  $N = 2$ , conditions of the type

$$\begin{pmatrix} a \\ b \end{pmatrix} \perp q^*$$

can be written as

$$\begin{pmatrix} a & b \end{pmatrix} \cdot q = 0;$$

proportionalities of the form

$$\begin{pmatrix} a \\ b \end{pmatrix} \sim q,$$

can be rewritten as

$$\begin{pmatrix} -b & a \end{pmatrix} \cdot q = 0.$$

By stacking such equations, one obtains a set of  $P \gg 2$  homogeneous linear equations, which has to be solved in least squares sense under the constraint  $\|q\| = 1$ . The optimal solution is the second right singular vector of the coefficient matrix.

**Remark 7:** Note that theoretically, *i.e.* in the absence of noise, the vector  $q$  already follows from one of the proportionalities in (48-65). So we have proved that, in principle, a paraunitary filter can be blindly identified in a finite number of steps, just consisting of the normalization of a vector and the computation of its orthogonal complement.

This also means that in the absence of noise the set of equations derived from (48-65) have a unique solution, which corresponds to the exact  $q$ . The goal of stacking constraints in  $M_1$  and  $M_2$  is to make

the approach more robust in the presence of noise. For instance, Eq. (66) simply corresponds to an overdetermined set of noisy homogeneous linear equations. In the presence of noise, such a set can then be solved in least-squares sense.

The non-iterative technique can be outlined as follows:

Algorithm 2:

**compute**  $\Gamma^w(\boldsymbol{\mu})$   
**set**  $k := \lambda_c - 1..0$   
**for each**  $k$  **do**  
    **compute**  $\Gamma^{s^{(k+1)}}(\boldsymbol{\mu})$  on the edges of the domain of support  
    **compute** the last column of  $\mathbf{U}_{k+1}$   
    **complete** with an orthogonal complement  
**determine**  $\mathbf{U}_0$  by means of instantaneous BSS  
**return**  $\mathbf{C}[z]$   
**return**  $\mathbf{H}[z]$  by inverting  $\mathbf{C}[z]$

Alg. 2 has the advantage over Alg. 1 of being much less computationally expensive. Partially, this is due to the fact that it does not contain an iteration over  $t$ . A disadvantage of this is that estimation errors, for a given  $k$ , are not reduced in subsequent steps. A second drawback of the approach is that the estimation of a unitary factor  $\mathbf{U}_{k+1}$  is based on the cumulants of the outer slices of the  $\boldsymbol{\mu}$ -domain, which form a decreasing part of the domain of support as  $k$  increases. Note that the sum of the squared Frobenius-norms of all the cumulant tensors in the domain of support is constant, as we consider only unitary transformations and time shifts, so that the outer slices usually contain less information as  $k$  increases. However, Alg. 2 may yield a first estimate of the equalizer, which is to be fine-tuned by Alg. 1.

## VII. COMPUTER RESULTS

Alg. 2 has been tested on FIR complex mixtures of length  $K = 2$ , with  $N = 3$  unit variance QPSK white processes. Channels, constructed as explained in section 3.3, are paraunitary in order to preserve second-order whiteness. For each randomly generated channel, a block of noisy observations is drawn according to:

$$\mathbf{w}(n) = \sum_{k=0}^K \mathbf{C}(k)\mathbf{s}(n-k) + \rho\mathbf{v}(n)$$

where  $v(n)$  is a white circular complex Gaussian noise with identity covariance matrix. Parameter  $\rho$  is introduced in order to control the Signal to Noise Ratio (SNR) per symbol, and is defined as:

$$SNR_{dB} = -20 \log_{10} \rho.$$

The experiment consisted of 100 Monte Carlo runs. We considered data blocks of 1024 and 2048 symbols. For the estimation of  $\mathbf{U}_0$ , we used the algorithm [26]. For each trial, the Minimum Mean Square Error solution is computed. In case of MIMO convolutive mixture, the MMSE equalizer is defined as:

$$\mathbf{H}_{MMSE}[z] = \mathbf{C}_{ss}[z] \mathbf{C}^\dagger[z] \mathbf{C}_{ww}^{-1}[z] \quad (69)$$

where  $\mathbf{C}_{ss}[z]$  is the correlation matrix of the  $N$  input signals  $s_i(n)$ , and  $\mathbf{C}_{ww}[z]$  the correlation matrix of the  $N$  observations  $w_i(n)$ .

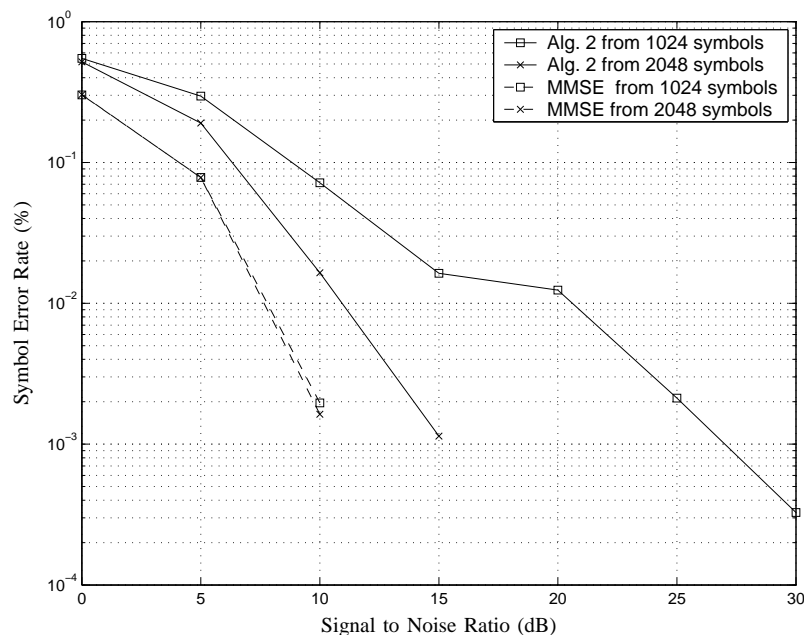


Fig. 4. Symbol Error Rate as a function of SNR in the first experiment (Alg. 2).

In figure 4 we plotted median curves for the obtained Symbol Error Rate (SER), *i.e.* the number of errors divided by the total number of source values.

For large data blocks, the performance of Alg. 2 comes close to that of the MMSE equalizer. If the block length decreases, the precision decreases. Nevertheless, the performance remains reasonable taking into account the very low computational cost of the algorithm. In this simulation, the solution is obtained by means of the Eigenvalue Decomposition of 2 ( $3 \times 3$ ) matrices and the BSS of an instantaneous ( $3 \times 3$ )

mixture. Alg. 2 can be used to compute a “quick and dirty” first estimate of the paraunitary filter, with which a more expensive algorithm can subsequently be initialized.

Alg. 1 has been tested on mixtures of length  $K = 3$  with  $N = 2$ . We have generated 100 random channels with data block lengths of 256, and 512 symbols. Equalizers returned by the algorithm are then tested with two different white processes of 5000 symbols each in order to compute the SER. Thus, the minimal resolution is  $(2 * 5000 * 100)^{-1} = 10^{-6}$ . Figure 5 shows median results of the 100 trials for QPSK signals, *i.e.* median is an estimate of average SER. This figure proves that Alg. 1 works well on short data length since from only 256 symbols and with a noise of 16dB, the median SER is below the minimal resolution.

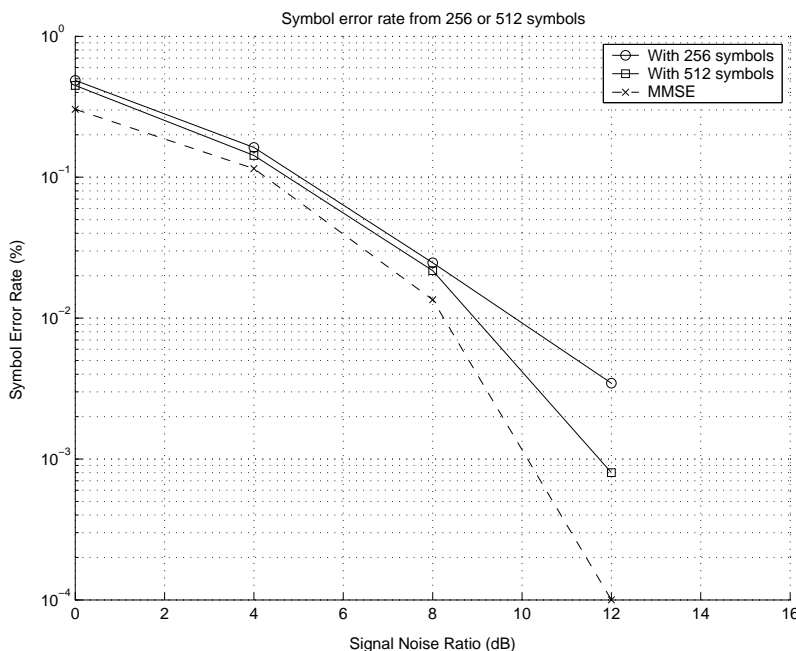


Fig. 5. Symbol Error Rate obtained by Alg. 1 when a length-3 equalizer is built from blocks of 256 or 512 symbols. For both lengths, the median of the SER is below  $10^{-6}$  after 12dB.

Another basis for comparison is the distance (Frobenius norm) between the equalizer found and the Zero-Forcing solution. In fact, the ZF solution for MIMO convolutive mixtures is simply the inverse of the channel. Then, when  $K = L$ , the global transfer function with the ZF equalizer  $\mathbf{H}_{ZF}[z]$  is

$$\mathbf{G}[z] = \sum_n \sum_m \mathbf{C}(m) \mathbf{H}_{ZF}(n) z^{-(n-m)} = \mathbf{I} z^{-L}. \quad (70)$$

So, we just have to compare the global transfer function  $\mathbf{G}[z]$  obtained with the paraunitary equalizer, to

the identity matrix up to a multiplicative matrix  $\mathbf{D}[z] = \mathbf{\Lambda}[z]\mathbf{P}$  (definition 1). Once  $\mathbf{G}[z]$  is estimated, we store in an  $N \times N(2L - 1)$  array  $\mathbb{G}$  by merely stacking the matrices one after the other. Next, we search for columns containing the entry of largest modulus in each row of  $\mathbb{G}$ . By comparing columns indices, which have to be different modulo  $N$ , we find matrix  $\mathbf{P}$  and  $\mathbf{\Lambda}[z]$ . Next, we normalize each row by its entry of largest modulus. The matrix obtained this way is denoted  $\bar{\mathbb{G}}$ . Then, we replace the  $N$  previous entries of  $\bar{\mathbb{G}}$  by zeros. This is the same as subtracting  $\mathbb{D}$  to  $\bar{\mathbb{G}}$ , where  $\mathbb{D}$  is made from  $\mathbf{D}[z]$  with the same size as  $\mathbb{G}$ . Finally, we compute the Frobenius norm of the resulting matrix, *i.e.*  $\|\bar{\mathbb{G}} - \mathbb{D}\|$ . This value is the distance between the paraunitary equalizer found and the ZF solution. Medians of distances are reported in figure 6. We note that between 0 and 8dB and for 512 symbols, the distance decreases rapidly from 1 to 0.25. Then, after 8dB, the distance stays approximately constant.

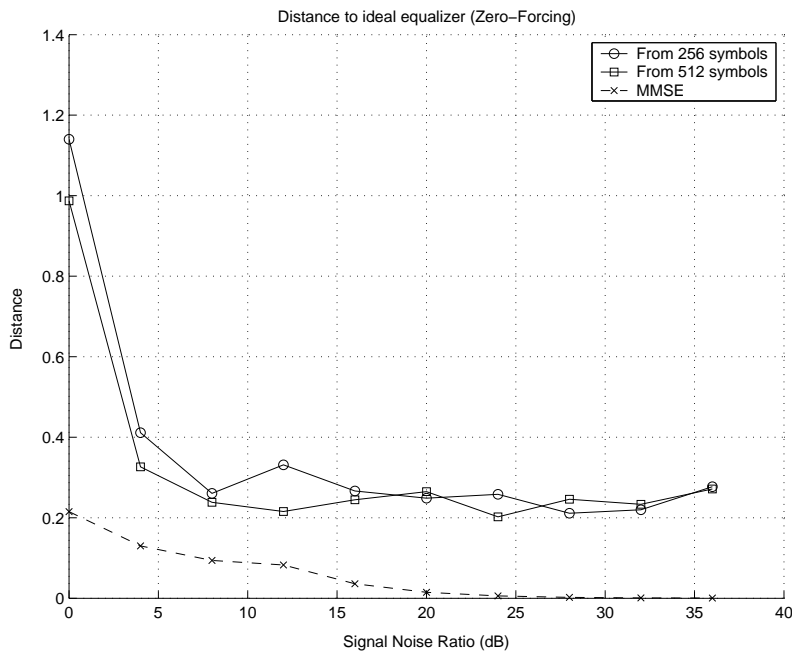


Fig. 6. Distance to the Zero-Forcing equalizer when a length-3 equalizer is built from blocks of 256 or 512 symbols (Alg. 1). Dashed line: distance between ZF and MSE.

## VIII. CONCLUDING REMARKS

The Blind MIMO Equalization problem of digital communication systems can be reduced to a simpler model, composed of a paraunitary channel and a paraunitary equalizer. Thanks to this statement, we have presented a parametrization of the equalizer, in order to carry out blind source separation, *i.e.* without

learning sequence. Based on some theoretical developments involving a parametrization of paraunitary filters, two algorithms have been devised. The first iteratively maximizes a contrast through successive sweeps and eventually equalizes the channel. The second implements a blind channel identification algebraically, and relies on equations in the noiseless case. The latter is not iterative and hence less computationally costly, but less accurate in the presence of noise. Moreover, we have demonstrated that the computation of the blind equalizer can be completed within a polynomial complexity, useful in burst-mode transmissions. Performances presented in the last section report an average SER of 0.35% at 12dB of SNR, with data length of 256 symbols only. Open issues currently being addressed include robustness to channel length misadjustment, and space-time whitening.

#### ACKNOWLEDGEMENT

The authors wish to thank the anonymous reviewers for their helpful remarks, which allowed to improve the clarity of the paper.

#### APPENDIX

##### *Appendix-A*

Table I is used for implementing the algorithm. It gives definition of each  $\mathcal{K}$  for any delays  $\tau_i$ . For instance, for any delays  $\tau_i$ ,  $\mathcal{K}_{(4)}^{(0)}$  corresponds to the sum

$$\mathcal{K}_{(4)}^{(0)} = \sum_{qrst} \sum_{\boldsymbol{\tau}} A_{iq}(\tau_1) A_{ir}^*(\tau_2) A_{is}(\tau_3) A_{it}^*(\tau_4) \Gamma_{qs,rt}^{\boldsymbol{\tau}}(\boldsymbol{\tau}). \quad (71)$$

##### *Appendix-B*

In order to reduce the complexity of the algorithm, it is suggested to store each  $\mathcal{K}$  in a matrix  $\mathbf{T} \in \mathbb{C}^{N^4 \times N^4}$ . This matrix is defined by

$$\mathbf{T} \stackrel{\text{def}}{=} \sum_{\boldsymbol{\tau}, i} \mathbf{f}_i(\boldsymbol{\tau}) \mathbf{c}(\boldsymbol{\tau})^\top. \quad (72)$$

where, for  $N = 2$ ,

$$\mathbf{f}_i(\boldsymbol{\tau}) = \begin{bmatrix} f_i^{1111}(\tau_1, \tau_2, \tau_3, \tau_4) \\ f_i^{1112}(\tau_1, \tau_2, \tau_3, \tau_4) \\ f_i^{1121}(\tau_1, \tau_2, \tau_3, \tau_4) \\ \vdots \\ f_i^{2222}(\tau_1, \tau_2, \tau_3, \tau_4) \end{bmatrix} \quad (73)$$

$\mathcal{K}_{(\alpha)}^{(2\beta+\alpha-4)} \Rightarrow \sum_{\mathcal{T}} A_{iq}(\tau_1)A_{ir}^*(\tau_2)A_{is}(\tau_3)A_{it}^*(\tau_4)\Gamma_{ac,bd}^{\mathcal{X}}(\boldsymbol{\tau})$	
$\mathcal{K}_{(4)}^{(0)}$	$q = a, r = b, s = c, t = d$
$\mathcal{K}_{(3)}^{(1)}$	$q = a, r = b, s = c, t = 2, d = 1$
	$q = a, r = b, s = 1, c = 2, t = d$
	$q = a, r = 2, b = 1, s = c, t = d$
	$q = 1, a = 2, r = b, s = c, t = d$
$\mathcal{K}_{(3)}^{(-1)}$	$q = a, r = b, s = c, t = 1, d = 2$
	$q = a, r = b, s = 2, c = 1, t = d$
	$q = a, r = 1, b = 2, s = c, t = d$
	$q = 2, a = 1, r = b, s = c, t = d$
$\mathcal{K}_{(2)}^{(0)}$	$q = a, r = b, s = 1, c = 2, t = 1, d = 2$
	$q = a, r = b, s = 2, c = 1, t = 2, d = 1$
	$q = a, r = 2, b = 1, s = c, t = 1, d = 2$
	$q = a, r = 1, b = 2, s = c, t = 2, d = 1$
	$q = a, r = 2, b = 1, s = 2, c = 1, t = d$
	$q = a, r = 1, b = 2, s = 1, c = 2, t = d$
	$q = 1, a = 2, r = 1, b = 2, s = c, t = d$
	$q = 2, a = 1, r = 2, b = 1, s = c, t = d$
	$q = 1, a = 2, r = b, s = c, t = 1, d = 2$
	$q = 2, a = 1, r = b, s = c, t = 2, d = 1$
$\mathcal{K}_{(2)}^{(2)}$	$q = a, r = b, s = 1, c = 2, t = 2, d = 1$
	$q = a, r = 2, b = 1, s = c, t = 2, d = 1$
	$q = a, r = 2, b = 1, s = 1, c = 2, t = d$
	$q = 1, a = 2, r = b, s = c, t = 2, d = 1$
$\mathcal{K}_{(2)}^{(-2)}$	$q = a, r = 1, b = 2, s = c, t = 1, d = 2$
	$q = a, r = 1, b = 2, s = 2, c = 1, t = d$
	$q = 2, a = 1, r = b, s = c, t = 1, d = 2$
	$q = 2, a = 1, r = 1, b = 2, s = c, t = d$
$\mathcal{K}_{(1)}^{(3)}$	$q = a, r = 2, b = 1, s = 1, c = 2, t = 2, d = 1$
	$q = 1, a = 2, r = b, s = 1, c = 2, t = 2, d = 1$
	$q = 1, a = 2, r = 2, b = 1, s = c, t = 2, d = 1$
	$q = 1, a = 2, r = 2, b = 1, s = 1, c = 2, t = d$
$\mathcal{K}_{(1)}^{(-3)}$	$q = a, r = 1, b = 2, s = 2, c = 1, t = 1, d = 2$
	$q = 2, a = 1, r = b, s = 2, c = 1, t = 1, d = 2$
	$q = 2, a = 1, r = 1, b = 2, s = c, t = 1, d = 2$
	$q = 2, a = 1, r = 1, b = 2, s = 2, c = 1, t = d$
$\mathcal{K}_{(1)}^{(1)}$	$q = a, r = 2, b = 1, s = 2, c = 1, t = 2, d = 1$
	$q = a, r = 1, b = 2, s = 1, c = 2, t = 2, d = 1$
	$q = a, r = 2, b = 1, s = 1, c = 2, t = 1, d = 2$
	$q = 1, a = 2, r = b, s = 2, c = 1, t = 2, d = 1$
	$q = 1, a = 2, r = b, s = 1, c = 2, t = 1, d = 2$
	$q = 2, a = 1, r = b, s = 1, c = 2, t = 2, d = 1$
	$q = 1, a = 2, r = 1, b = 2, s = c, t = 2, d = 1$
	$q = 1, a = 2, r = 2, b = 1, s = c, t = 1, d = 2$
	$q = 2, a = 1, r = 2, b = 1, s = c, t = 2, d = 1$
	$q = 1, a = 2, r = 2, b = 1, s = 2, c = 1, t = d$
$\mathcal{K}_{(1)}^{(-1)}$	$q = a, r = 2, b = 1, s = 2, c = 1, t = 1, d = 2$
	$q = a, r = 1, b = 2, s = 1, c = 2, t = 1, d = 2$
	$q = a, r = 1, b = 2, s = 2, c = 1, t = 2, d = 1$
	$q = 1, a = 2, r = b, s = 2, c = 1, t = 1, d = 2$
	$q = 2, a = 1, r = b, s = 2, c = 1, t = 2, d = 1$
	$q = 2, a = 1, r = b, s = 1, c = 2, t = 1, d = 2$
	$q = 1, a = 2, r = 1, b = 2, s = c, t = 1, d = 2$
	$q = 2, a = 1, r = 1, b = 2, s = c, t = 2, d = 1$
	$q = 2, a = 1, r = 2, b = 1, s = c, t = 1, d = 2$
	$q = 2, a = 1, r = 2, b = 1, s = 2, c = 1, t = d$
$\mathcal{K}_{(0)}^{(2)}$	$q = 1, a = 2, r = 2, b = 1, s = 1, c = 2, t = 1, d = 2$
	$q = 1, a = 2, r = 2, b = 1, s = 2, c = 1, t = 2, d = 1$
	$q = 1, a = 2, r = 1, b = 2, s = 1, c = 2, t = 2, d = 1$
	$q = 2, a = 1, r = 2, b = 1, s = 1, c = 2, t = 2, d = 1$
$\mathcal{K}_{(0)}^{(-2)}$	$q = 2, a = 1, r = 1, b = 2, s = 2, c = 1, t = 2, d = 1$
	$q = 2, a = 1, r = 1, b = 2, s = 2, c = 1, t = 1, d = 2$
	$q = 2, a = 1, r = 2, b = 1, s = 2, c = 1, t = 1, d = 2$
	$q = 1, a = 2, r = 1, b = 2, s = 2, c = 1, t = 1, d = 2$
$\mathcal{K}_{(0)}^{(0)}$	$q = 1, a = 2, r = 1, b = 2, s = 1, c = 2, t = 1, d = 2$
	$q = 1, a = 2, r = 2, b = 1, s = 2, c = 1, t = 1, d = 2$
	$q = 1, a = 2, r = 1, b = 2, s = 2, c = 1, t = 2, d = 1$
	$q = 2, a = 1, r = 2, b = 1, s = 1, c = 2, t = 1, d = 2$
$\mathcal{K}_{(0)}^{(-4)}$	$q = 2, a = 1, r = 1, b = 2, s = 1, c = 2, t = 2, d = 1$
	$q = 2, a = 1, r = 2, b = 1, s = 2, c = 1, t = 2, d = 1$
	$q = 1, a = 2, r = 2, b = 1, s = 1, c = 2, t = 2, d = 1$
	$q = 2, a = 1, r = 1, b = 2, s = 2, c = 1, t = 1, d = 2$

TABLE I

DEFINITION OF EACH PRODUCT  $\mathcal{K}_{(\alpha)}^{(2\beta+\alpha-4)}$ .with  $f_i^{qrst}(\tau_1, \tau_2, \tau_3, \tau_4) = A_{iq}(\tau_1)A_{ir}^*(\tau_2)A_{is}(\tau_3)A_{it}^*(\tau_4)$ , and

$$\mathbf{c}(\boldsymbol{\tau}) = \begin{bmatrix} \Gamma_{11,11}^{\mathcal{X}}(\tau_1, \tau_2, \tau_3, \tau_4) \\ \Gamma_{11,12}^{\mathcal{X}}(\tau_1, \tau_2, \tau_3, \tau_4) \\ \Gamma_{12,11}^{\mathcal{X}}(\tau_1, \tau_2, \tau_3, \tau_4) \\ \vdots \\ \Gamma_{22,22}^{\mathcal{X}}(\tau_1, \tau_2, \tau_3, \tau_4) \end{bmatrix}. \quad (74)$$

For only 2 observations, we have  $\mathbf{T} \in \mathbb{C}^{16 \times 16}$ . Hence we obtain the  $N^8 = 256$  possible values of (21) by summing  $L^4$  matrices of size  $16 \times 16$ . Then, in accordance with table I, we just have to define each  $\mathcal{K}$  by summing entries of  $\mathbf{T}$ . Moreover, we see that elements of  $\mathbf{f}_i(\boldsymbol{\tau})$  and  $\mathbf{c}(\boldsymbol{\tau})$  have the same ordering, *i.e.*  $q \equiv a, r \equiv b, s \equiv c, t \equiv d$ . Hence, both vectors can be constructed thanks to the same loops. So, we reduce the number of loops by  $N^4$ , and hence the computation time.

We immediately see another simplification if we include indices  $\tau_1, \tau_2, \tau_3, \tau_4$  in vectors  $\mathbf{f}_i$  and  $\mathbf{c}$ , *e.g.* by Kronecker product. In this case, both vectors are of length  $(N(L+1))^4$  and then  $\mathbf{T}$  is  $(N(L+1))^4 \times (N(L+1))^4$ . So, we only have  $(N(L+1))^4$  loops for building vectors  $\mathbf{f}_i$  and  $\mathbf{c}$ , and then we have to make only one product  $\mathbf{f}_i \mathbf{c}^T$  for matrix  $\mathbf{T}$ . Nevertheless, the memory capacity is then the drawback of this method since more memory is used for storing matrix  $\mathbf{T}$ . As a consequence, this last simplification is probably interesting for small values of  $L$ .

#### REFERENCES

- [1] P. COMON, "Contrasts for Multichannel Blind Deconvolution," *IEEE Signal Processing Letters*, vol. 3, no. 7, pp. 209–211, July 1996.
- [2] D. DONOHO, "On minimum entropy deconvolution," in *Applied time series analysis II*, A. Press, Ed., 1981, pp. 565–609.
- [3] J. K. TUGNAIT, "Identification and Deconvolution of Multichannel NonGaussian Processes using Higher Order Statistics and Inverse Filter Criteria," *IEEE Trans. on Signal Processing*, vol. 45, pp. 658–672, Mar. 1997.
- [4] —, "On Blind Separation of Convolutional Mixtures of Independent Linear Signals in Unknown Additive Noise," *IEEE Trans. on Signal Processing*, vol. 46, no. 11, pp. 3117–3123, 1998.
- [5] Y. INOUE and R. LIU, "Direct Blind Deconvolution of Multiuser-Multichannel System Driven by Temporally White Source Signals," in *Proc. of ICASSP*, vol. 5, Phoenix, Arizona, USA, 1999, pp. 2889–2892.
- [6] Z. DING and T. NGUYEN, "Stationary Points of a Kurtosis Maximization Algorithm for Blind Signal Separation and Antenna Beamforming," *IEEE Trans. on Signal Processing*, vol. 48, no. 6, pp. 1587–1596, June 2000.
- [7] I. FIJALKOW, A. TOUZNI, and J. R. TREICHLER, "Fractionally spaced equalization using CMA: Robustness to channel noise and lack of disparity," *IEEE Trans. on Signal Processing*, vol. 45, no. 1, pp. 56–66, Jan. 1997.
- [8] A. J. V. der VEEN and A. PAULRAJ, "An Analytical Constant Modulus Algorithm," *IEEE Trans. on Signal Processing*, vol. 44, no. 5, pp. 1136–1155, May 1996.
- [9] C. B. PAPADIAS, "Globally Convergent Blind Source Separation Based on a Multiuser Kurtosis Maximization Criterion," *IEEE Trans. on Signal Processing*, vol. 48, no. 12, pp. 3508–3519, December 2000.
- [10] A. TOUZNI, I. FIJALKOW, M. G. LARIMORE, and J. R. TREICHLER, "A Globally Convergent Approach for Blind MIMO Adaptive Deconvolution," *IEEE Trans. on Signal Processing*, vol. 49, no. 6, pp. 1166–1178, June 2001.
- [11] O. GRELLIER, P. COMON, B. MOURRAIN, and P. TREBUCHET, "Analytical Blind Channel Identification," *IEEE Trans. on Signal Processing*, vol. 50, no. 9, Sept. 2002.
- [12] L. ROTA, P. COMON, and S. ICART, "Blind Equalization of MIMO Channels," in *Proc. of Signal Processing Advances in Wireless Communications*, Rome, Italy, June 15-18 2003.

- [13] L. D. LATHAUWER, B. D. MOOR, and J. VANDEWALLE, "An Algebraic Approach to Blind MIMO Identification," in *Proc. 2nd Int. Workshop on Independent Component Analysis and Blind Signal Separation (ICA 2000)*, Helsinki, Finland,, June 19-22 2000.
- [14] —, "An Algebraic Approach to the Blind Identification of Paraunitary Filters," in *Proc. of IEEE Wireless Communications and Networking Conference*, Chicago, USA, September 23-28 2000.
- [15] P. P. VAIDYANATHAN, *Multirate Systems and Filter Banks*. London: Prentice-Hall, 1993.
- [16] Y. INOUE and R. LIU, "A System-Theoretic Foundation for Blind Equalization of an FIR MIMO Channel System," *IEEE Trans. on Circuits and Systems*, vol. 49, no. 4, pp. 425–436, April 2002.
- [17] A. H. SAYED and T. KAILATH, "A Survey of Spectral Factorization Methods," *Numer. Linear Algebra Appl.*, vol. 8, pp. 467–496, 2001.
- [18] P. COMON and L. ROTA, "Blind Separation of Independent Sources from Convolutional Mixtures," *IEICE Trans. on Fundamentals of Elec. Com. Comput. Sciences*, vol. E86-A, no. 3, pp. 550–563, March 2003.
- [19] P. COMON, E. MOREAU, and L. ROTA, "Blind Separation of Convolutional Mixtures: a Contrast-Based Joint Diagonalization Approach," in *Proc. of Int. Conf. Independent Component Analysis*, San Diego, Dec. 9-13 2001, pp. 686–691.
- [20] P. McCULLAGH, *Tensor Methods in Statistics*, ser. Monographs on Statistics and Applied Probability. Chapman and Hall, 1987.
- [21] X. GAO, T. Q. NGUYEN, and G. STRANG, "On Factorization of  $M$ -Channel Paraunitary Filterbanks," *IEEE Trans. on Signal Processing*, vol. 49, no. 7, pp. 1433–1446, July 2001.
- [22] L. ROTA, P. COMON, and S. ICART, "Blind MIMO Equalization under paraunitary constraint," Laboratoire I3S, I3S Report RR-2002-35-FR, Sept. 2002.
- [23] G. H. GOLUB and C. F. V. LOAN, *Matrix Computation*, 3rd ed. The John Hopkins University Press, 1996.
- [24] N. DELFOSSE and P. LOUBATON, "Adaptive blind separation of independent Sources: a deflation approach," *Signal Processing*, vol. 45, pp. 59–83, 1995.
- [25] D. COX, J. LITTLE, and D. O'SHEA, *Using Algebraic Geometry*. Springer Verlag, New York, 1998.
- [26] L. D. LATHAUWER, B. D. MOOR, and J. VANDEWALLE, "Independent Component Analysis and (Simultaneous) Third-Order Tensor Diagonalization," *IEEE Trans. on Signal Processing*, vol. 49, no. 10, pp. 2262–2271, October 2001.