

# Fast Adaptive Blind MMSE Equalizer for Multichannel FIR Systems

Ibrahim Kacha,<sup>1,2</sup> Karim Abed-Meraim,<sup>2</sup> and Adel Belouchrani<sup>1</sup>

<sup>1</sup>Département d'Électronique, École Nationale Polytechnique (ENP), 10 avenue Hassen Badi El-Harrach, 16200 Algiers, Algeria

<sup>2</sup>Département Traitement du Signal et de l'Image, École Nationale Supérieure des Télécommunications (ENST), 37-39 rue Dareau, 75014 Paris, France

Received 30 December 2005; Revised 14 June 2006; Accepted 22 June 2006

We propose a new blind minimum mean square error (MMSE) equalization algorithm of noisy multichannel finite impulse response (FIR) systems, that relies only on second-order statistics. The proposed algorithm offers two important advantages: a low computational complexity and a relative robustness against channel order overestimation errors. Exploiting the fact that the columns of the equalizer matrix filter belong both to the signal subspace and to the kernel of truncated data covariance matrix, the proposed algorithm achieves blindly a direct estimation of the zero-delay MMSE equalizer parameters. We develop a two-step procedure to further improve the performance gain and control the equalization delay. An efficient fast adaptive implementation of our equalizer, based on the projection approximation and the shift invariance property of temporal data covariance matrix, is proposed for reducing the computational complexity from  $O(n^3)$  to  $O(qnd)$ , where  $q$  is the number of emitted signals,  $n$  the data vector length, and  $d$  the dimension of the signal subspace. We then derive a statistical performance analysis to compare the equalization performance with that of the optimal MMSE equalizer. Finally, simulation results are provided to illustrate the effectiveness of the proposed blind equalization algorithm.

Copyright © 2006 Ibrahim Kacha et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. INTRODUCTION

### 1.1. Blind equalization

An elementary problem in the area of digital communications is that of intersymbol interference (ISI). ISI results from linear amplitude and phase dispersion in the transmission channel, mainly due to multipath propagation. To achieve reliable communications, channel equalization is necessary to deal with ISI.

Conventional nonblind equalization algorithms require training sequence or a priori knowledge of the channel [1]. In the case of wireless communications these solutions are often inappropriate, since a training sequence is usually sent periodically, thus the effective channel throughput is considerably reduced. It follows that the blind and semiblind equalization of transmission channels represent a suitable alternative to traditional equalization, because they do not fully rely on training sequence or a priori channel knowledge.

In the first contributions [2, 3], blind identification/equalization (BIE) schemes were based, implicitly or explicitly on higher- (than second-) order statistics of the observation. However, the shortcoming of these methods is the high error variances often exhibited by higher-order statistical esti-

mates. This often translates into slow convergence for on-line methods or unreasonable data length requirements for off-line methods. In the pioneering work of Tong et al. [4], it has been shown that the second-order statistics contain sufficient information for BIE of multichannel FIR systems. Later, active research in BIE area has led to a variety of second-order statistics-based algorithms (see the survey paper [5], as well as the references therein). Many efficient solutions (e.g., [6]) suffer from the lack of robustness against channel order overestimation errors and are also computationally expensive. A lot of research effort has been done to either develop efficient techniques for channel order estimation (e.g., [7, 8]) or to develop BIE methods robust to channel order estimation errors. Several *robust* techniques have been proposed so far [9–13], but all of them depend explicitly or implicitly on the channel order and hence have only a limited robustness, in the sense that their performance degrades significantly when the channel overestimation error is large.

### 1.2. Contributions

In this work, we develop a blind adaptive equalization algorithm based on MMSE estimation, which presents a number of nice properties such as robustness to channel order

overestimation errors and low computational complexity. More precisely, this paper describes a new technique for direct design of MIMO blind adaptive MMSE equalizer, having  $O(qnd)$  complexity and relative robustness against channel order overestimation errors. We show that the columns of the zero-delay equalizer matrix filter belongs simultaneously to the signal subspace and to the kernel of truncated data covariance matrix. This property leads to a simple estimation method of the equalizer filter by minimizing a certain quadratic form subject to a properly chosen constraint. We present an efficient fast adaptive implementation of the novel algorithm, including a two-step estimation procedure, which allows us to compensate for the performance loss of the equalizer, compared to the nonblind one, and to choose a nonzero equalization delay. Also, we derive the asymptotic performance analysis of our method which leads to a closed form expression of the performance loss (compared to the optimal one) due to the considered blind processing.

The rest of the paper is organized as follows. In Section 2 the system model and problem statement are developed. Batch and adaptive implementations of the algorithm, using respectively, linear and quadratic constraints are introduced in Sections 3 and 4. Section 5 is devoted to the asymptotic performance analysis of the proposed blind MMSE filter. Simulation examples and performances evaluation are provided in Section 6. Finally, conclusions are drawn in Section 7.

### 1.3. Notations

Most notations are standard: vectors and matrices are represented by boldface small and capital letters, respectively. The matrix transpose, the complex conjugate, the hermitian, and the Moore-Penrose pseudo-inverse are denoted by  $(\cdot)^T$ ,  $(\cdot)^*$ ,  $(\cdot)^H$ , and  $(\cdot)^\#$ , respectively.  $\mathbf{I}_n$  is the  $n \times n$  identity matrix and  $\mathbf{0}$  (resp.,  $\mathbf{0}_{i \times k}$ ) denotes the zero matrix of appropriate dimension (resp., the zero matrix of dimension  $i \times k$ ). The symbol  $\otimes$  stands for the Kronecker product;  $\text{vec}(\cdot)$  and  $\text{vec}^{-1}(\cdot)$  denote the column vectorization operator and its inverse, respectively.  $E(\cdot)$  is the mathematical expectation. Also, we use some informal MATLAB notations, such as  $\mathbf{A}(k, :)$ ,  $\mathbf{A}(:, k)$ ,  $\mathbf{A}(i, k)$ ,  $\dots$ , for the  $k$ th row, the  $k$ th column, the  $(i, k)$ th entry of matrix  $\mathbf{A}$ , respectively.

## 2. DATA MODEL

Consider a discrete time MIMO system of  $q$  inputs,  $p$  outputs ( $p > q$ ) given by

$$\mathbf{x}(t) = \sum_{k=0}^L \mathbf{H}(k)\mathbf{s}(t-k) + \mathbf{b}(t), \quad (1)$$

where  $H(z) = \sum_{k=0}^L \mathbf{H}(k)z^{-k}$  is an unknown causal FIR  $p \times q$  transfer function. We assume (A1)  $H(z)$  is irreducible and column reduced, that is,  $\text{rank}(H(z)) = q$ , for all  $z$  and  $\mathbf{H}(L)$  is full column rank. (A2) The input (nonobservable) signal  $\mathbf{s}(t)$  is a  $q$ -dimensional random vector assumed to be an iid (independently and identically distributed) zero-mean unit power

complex circular process [14], with finite fourth-order moments, that is,  $E(\mathbf{s}(t+\tau)\mathbf{s}^H(t)) = \delta(\tau)\mathbf{I}_q$ ,  $E(\mathbf{s}(t+\tau)\mathbf{s}^T(t)) = \mathbf{0}$ ,  $E(|s_i(t)|^4) < \infty$ ,  $i = 1, \dots, q$ . (A3)  $\mathbf{b}(t)$  is an additive spatially and temporally white Gaussian noise of power  $\sigma_b^2\mathbf{I}_p$  and independent of the transmitted sequence  $\{\mathbf{s}(t)\}$ .<sup>1</sup>

By stacking  $N$  successive samples of the received signal  $\mathbf{x}(t)$  into a single vector, we obtain the  $n$ -dimensional ( $n = Np$ ) vector

$$\begin{aligned} \mathbf{x}_N(t) &= [\mathbf{x}^T(t) \ \mathbf{x}^T(t-1) \ \dots \ \mathbf{x}^T(t-N+1)]^T \\ &= \mathbf{H}_N \mathbf{s}_m(t) + \mathbf{b}_N(t), \end{aligned} \quad (2)$$

where  $\mathbf{s}_m(t) = [\mathbf{s}^T(t) \ \dots \ \mathbf{s}^T(t-m+1)]^T$ ,  $\mathbf{b}_N(t) = [\mathbf{b}^T(t) \ \dots \ \mathbf{b}^T(t-N+1)]^T$ ,  $m = N+L$  and  $\mathbf{H}_N$  is the channel convolution matrix of dimension  $n \times d$ , ( $d = qm$ ), given by

$$\mathbf{H}_N = \begin{bmatrix} \mathbf{H}(0) & \dots & \mathbf{H}(L) & \mathbf{0} \\ & \ddots & & \ddots \\ \mathbf{0} & & \mathbf{H}(0) & \dots & \mathbf{H}(L) \end{bmatrix}. \quad (3)$$

It is shown in [15] that if  $N$  is large enough and under assumption (A1), matrix  $\mathbf{H}_N$  is full column rank.

## 3. ALGORITHM DERIVATION

### 3.1. MMSE equalizer

Consider a  $\tau$ -delay MMSE equalizer ( $\tau \in \{0, 1, \dots, m-1\}$ ). Under the above data model, one can easily show that the equalizer matrix  $\mathbf{V}_\tau$  corresponding to the desired solution is given by

$$\mathbf{V}_\tau = \arg \min_{\mathbf{V}} E(\|\mathbf{s}(t-\tau) - \mathbf{V}^H \mathbf{x}_N(t)\|^2) = \mathbf{C}^{-1} \mathbf{G}_\tau, \quad (4)$$

where

$$\mathbf{C} \stackrel{\text{def}}{=} E(\mathbf{x}_N(t)\mathbf{x}_N^H(t)) = \mathbf{H}_N \mathbf{H}_N^H + \sigma_b^2 \mathbf{I}_n \quad (5)$$

is the data covariance matrix and  $\mathbf{G}_\tau$  is an  $n \times q$  matrix given by

$$\mathbf{G}_\tau \stackrel{\text{def}}{=} E(\mathbf{x}_N(t)\mathbf{s}^H(t-\tau)) = \mathbf{H}_N \mathbf{J}_{q\tau, q, q(m-\tau-1)}, \quad (6)$$

$\mathbf{J}_{j,k,l}$  is a truncation matrix defined as follow:

$$\mathbf{J}_{j,k,l} \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{0}_{j \times k} \\ \mathbf{I}_k \\ \mathbf{0}_{l \times k} \end{bmatrix}. \quad (7)$$

Note that  $\mathbf{H}_N \mathbf{J}_{q\tau, q, q(m-\tau-1)}$  denotes the submatrix of  $\mathbf{H}_N$  given by the column vectors of indices varying in the range  $[\tau q +$

<sup>1</sup> Note that the column reduced condition in assumption (A1) can be relaxed, but that would lead to more complex notations. Similarly, the circularity and the finite value of the fourth-order moments of the input signal in assumption (A2) and the Gaussianity of additive noise in assumption (A3) are not necessary for the derivation of our algorithm, but used only for the asymptotic performance analysis.

$1, \dots, (\tau+1)q]$ . From (4), (5), (6) and using matrix inversion lemma, matrix  $\mathbf{V}_\tau$  is also expressed as  $\mathbf{V}_\tau = \mathbf{H}_N \bar{\mathbf{V}}_\tau$ , where  $\bar{\mathbf{V}}_\tau$  is a  $d \times q$ -dimensional matrix given by

$$\bar{\mathbf{V}}_\tau = \frac{1}{\sigma_b^2} \left( \mathbf{I}_d - \frac{1}{\sigma_b^2} (\sigma_b^2 \mathbf{I}_d + \mathbf{H}_N^H \mathbf{H}_N)^{-1} \mathbf{H}_N^H \mathbf{H}_N \right) \mathbf{J}_{q\tau, q, q(m-\tau-1)}. \quad (8)$$

Clearly, the columns of MMSE matrix filter  $\mathbf{V}_\tau$  belong to the *signal subspace* (i.e.,  $\text{range}(\mathbf{H}_N)$ ) and thus one can write

$$\mathbf{V}_\tau = \mathbf{W} \tilde{\mathbf{V}}_\tau, \quad (9)$$

where  $\mathbf{W}$  is an  $n \times d$  matrix whose column vectors form an orthonormal basis of the signal subspace (there exist a nonsingular  $d \times d$  matrix  $\mathbf{P}$  such that  $\mathbf{W} = \mathbf{H}_N \mathbf{P}$ ) and  $\tilde{\mathbf{V}}_\tau$  is a  $d \times q$ -dimensional matrix.

### 3.2. Blind equalization

Our objective here is to derive a blind estimate of the zero-delay MMSE equalizer  $\mathbf{V}_0$ . From (4), (6), (7), and (9), one can write  $\mathbf{V}_0 = \mathbf{W} \tilde{\mathbf{V}}_0$ , with

$$\mathbf{C} \mathbf{W} \tilde{\mathbf{V}}_0 = \begin{bmatrix} \mathbf{H}(0) \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}. \quad (10)$$

If we truncate the first  $p$  rows of system (10), we obtain

$$\mathbf{T} \tilde{\mathbf{V}}_0 = \mathbf{0}, \quad (11)$$

where  $\mathbf{T}$  is an  $(n-p) \times d$  matrix given by

$$\mathbf{T} \stackrel{\text{def}}{=} \bar{\mathbf{C}} \mathbf{W}, \quad (12)$$

$$\bar{\mathbf{C}} = \mathbf{C}(p+1:n, :) = \mathbf{J}_{p, n-p, 0}^T \mathbf{C}. \quad (13)$$

Matrix  $\bar{\mathbf{C}}$  is a submatrix of  $\mathbf{C}$  given by its  $n-p$  rows. Equation (11) shows that the columns of  $\tilde{\mathbf{V}}_0$  belong to the right null space of  $\mathbf{T}$  ( $\text{null}_r(\mathbf{T}) = \{\mathbf{z} \in \mathbb{C}^d : \mathbf{T}\mathbf{z} = \mathbf{0}\}$ ). Reversely, we can establish that (11) characterizes uniquely the zero-delay MMSE equalizer. We have the following result.

**Theorem 1.** *Under the above data assumptions and for  $N > qL + 1$  the solution of*

$$\mathbf{T} \tilde{\mathbf{V}} = \mathbf{0}, \quad (14)$$

*subject to the constraint*

$$\text{rank}(\tilde{\mathbf{V}}) = q, \quad (15)$$

*is unique (up to a constant  $q \times q$  nonsingular matrix) and corresponds to the desired MMSE equalizer, that is,*

$$\tilde{\mathbf{V}} = \tilde{\mathbf{V}}_0 \mathbf{R}, \quad (16)$$

*for a given constant  $q \times q$  invertible matrix  $\mathbf{R}$ .*

*Proof.* Let  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$  denote the eigenvalues of  $\mathbf{C}$ . Since  $\mathbf{H}_N$  is full column rank, the *signal part* of the covariance matrix  $\mathbf{C}$ , that is,  $\mathbf{H}_N \mathbf{H}_N^H$  has rank  $d$ , hence  $\lambda_k > \sigma_b^2$ ,  $k = 1, \dots, d$  and  $\lambda_k = \sigma_b^2$ ,  $k = d+1, \dots, n$ . Denote the unit-norm eigenvectors associated with the eigenvalues  $\lambda_1, \dots, \lambda_d$  by  $\mathbf{u}_s(1), \dots, \mathbf{u}_s(d)$ , and those corresponding to  $\lambda_{d+1}, \dots, \lambda_n$  by  $\mathbf{u}_b(1), \dots, \mathbf{u}_b(n-d)$ . Also define  $\mathbf{U}_s = [\mathbf{u}_s(1) \dots \mathbf{u}_s(d)]$  and  $\mathbf{U}_b = [\mathbf{u}_b(1) \dots \mathbf{u}_b(n-d)]$ . The covariance matrix is thus also expressed as  $\mathbf{C} = \mathbf{U}_s \text{diag}(\lambda_1, \dots, \lambda_d) \mathbf{U}_s^H + \sigma_b^2 \mathbf{U}_b \mathbf{U}_b^H$ . The columns of matrix  $\mathbf{U}_s$  span the signal subspace, that is,  $\text{range}(\mathbf{H}_N \mathbf{H}_N^H) = \text{range}(\mathbf{H}_N)$ , there exist a nonsingular  $d \times d$  matrix  $\mathbf{P}'$  such that  $\mathbf{U}_s = \mathbf{H}_N \mathbf{P}'$ , while the columns of  $\mathbf{U}_b$  span its orthogonal complement, the *noise subspace*, that is,  $\mathbf{U}_b^H \mathbf{U}_s = \mathbf{0}$ . As  $\mathbf{W}$  is an orthonormal basis of the signal subspace, there exists nonsingular  $d \times d$  matrices  $\mathbf{P}$  and  $\mathbf{P}''$  such that  $\mathbf{W} = \mathbf{H}_N \mathbf{P} = \mathbf{U}_s \mathbf{P}''$ , hence  $\mathbf{C} \mathbf{W} = (\mathbf{H}_N \mathbf{P}' \text{diag}(\lambda_1, \dots, \lambda_d) \mathbf{U}_s^H + \sigma_b^2 \mathbf{U}_b \mathbf{U}_b^H) \mathbf{U}_s \mathbf{P}'' = \mathbf{H}_N \mathbf{S}$ , where  $\mathbf{S} = \mathbf{P}' \text{diag}(\lambda_1, \dots, \lambda_d) \mathbf{P}''$  is nonsingular. Consequently,  $\mathbf{T} = \mathbf{C}(p+1:n, :) \mathbf{W} = \mathbf{H}_N(p+1:n, :) \mathbf{S}$ . Since  $\mathbf{H}_N$  is block-Toeplitz matrix (see equation (3)),  $\mathbf{H}_N(p+1:n, :) = [\mathbf{0}_{(n-p) \times q} \ \mathbf{H}_{N-1}]$ . As  $\mathbf{H}_{N-1}$  is full column rank, it implies that  $\dim(\text{null}_r(\mathbf{T})) = \dim(\text{null}_r([\mathbf{0}_{(n-p) \times q} \ \mathbf{H}_{N-1}])) = q$ . It follows that any full column rank  $d \times q$  matrix  $\tilde{\mathbf{V}}$ , solution of (14), can be considered as a basis of the right null space of matrix  $\mathbf{T}$ . According to (11) the columns of matrix  $\tilde{\mathbf{V}}_0$ , which characterize the MMSE filter given by (10), belong to  $\text{null}_r(\mathbf{T})$  and are linearly independent, it follows that  $\tilde{\mathbf{V}} = \tilde{\mathbf{V}}_0 \mathbf{R}$ , where  $\mathbf{R}$  is a nonsingular  $q \times q$  matrix.  $\square$

### 3.3. Implementation

#### 3.3.1. The SIMO case

In the SIMO case ( $q = 1$ ) matrix  $\tilde{\mathbf{V}}$  is replaced by the  $d$ -dimensional vector  $\tilde{\mathbf{v}}$  and (14) can be solved, simply, in the least squares sense subject to the unit norm constraint:

$$\tilde{\mathbf{v}} = \arg \min_{\|\mathbf{z}\|=1} (\mathbf{z}^H \mathbf{Q} \mathbf{z}), \quad (17)$$

where  $\mathbf{Q}$  is a  $(d \times d)$  matrix defined by

$$\mathbf{Q} \stackrel{\text{def}}{=} \mathbf{T}^H \mathbf{T}. \quad (18)$$

Then, according to (9) and (16), we obtain the MMSE equalizer vector  $\mathbf{v}_0 = r \mathbf{v}$ , where  $r$  is a given nonzero scalar and  $\mathbf{v}$  is the  $n$ -dimensional vector given by

$$\mathbf{v} = \mathbf{W} \tilde{\mathbf{v}}. \quad (19)$$

A batch-processing implementation of the SIMO blind MMSE equalization algorithm is summarized in Algorithm 1.

#### 3.3.2. The MIMO case

In this situation, the quadratic constraint on  $\tilde{\mathbf{V}}$  does not guarantee condition (15) in Theorem 1. One possible solution is to choose a linear constraint (instead of the quadratic one)

$$\begin{aligned}
\mathbf{C} &= \frac{1}{K} \sum_{t=0}^{K-1} \mathbf{x}_N(t) \mathbf{x}_N^H(t), \quad (K: \text{sample size}) \\
(\mathbf{W}, \Lambda_1) &= \text{eigs}(\mathbf{C}, d), \quad (\text{extracts the } d \text{ principal eigenvectors of } \mathbf{C}) \\
\mathbf{T} &= \mathbf{C}(p+1:n, :)\mathbf{W} \\
\mathbf{Q} &= \mathbf{T}^H \mathbf{T} \\
\tilde{\mathbf{v}} &= \text{the least eigenvector of } \mathbf{Q} \\
\mathbf{v} &= \mathbf{W}\tilde{\mathbf{v}}
\end{aligned}$$

ALGORITHM 1: SIMO blind MMSE equalization algorithm.

such as the  $q \times q$  first block of matrix  $\tilde{\mathbf{V}}$  is lower triangular

$$\tilde{\mathbf{V}}(1:q, 1:q) = \begin{bmatrix} 1 & \cdots & 0 \\ \times & \ddots & \vdots \\ \times & \times & 1 \end{bmatrix}, \quad (20)$$

which will guarantee that matrix  $\tilde{\mathbf{V}}$  has a full column rank  $q$ .

It is clear that (14) is equivalent to (see [16] for more details)

$$(\mathbf{I}_q \otimes \mathbf{T}) \text{vec}(\tilde{\mathbf{V}}) = \mathbf{0}. \quad (21)$$

Taking into account the lower triangular constraint in (20), (21) becomes

$$\mathbf{a} + \mathbf{A}\bar{\mathbf{v}} = \mathbf{0}, \quad (22)$$

where

$$\begin{aligned}
\bar{\mathbf{v}} &= \mathbf{J}^T \text{vec}(\tilde{\mathbf{V}}), \\
\mathbf{a} &= \text{vec}(\mathbf{T}\mathbf{J}_{0,q,d-q}), \\
\mathbf{A} &= (\mathbf{I}_q \otimes \mathbf{T})\mathbf{J}, \\
\mathbf{J} &= \text{diag}(\mathbf{J}_1, \mathbf{J}_2, \dots, \mathbf{J}_q), \\
\mathbf{J}_k &= \mathbf{J}_{k,d-k,0}, \quad k = 1, \dots, q.
\end{aligned} \quad (23)$$

The solution of (22) is given by

$$\bar{\mathbf{v}} = -\mathbf{A}^\# \mathbf{a}. \quad (24)$$

Matrix  $\tilde{\mathbf{V}}$ , solution of (14), is then given by  $\tilde{\mathbf{V}} = \text{vec}^{-1}(\tilde{\mathbf{v}})$  where  $\tilde{\mathbf{v}}$  is obtained from  $\bar{\mathbf{v}}$  by adding ones and zeros at the appropriate entries according to

$$\tilde{\mathbf{v}} = \mathbf{J}\bar{\mathbf{v}} + \text{vec}(\mathbf{J}_{0,q,d-q}). \quad (25)$$

From (9) and (16), we obtain the MMSE equalizer matrix  $\mathbf{V}_0 = \mathbf{V}\mathbf{R}^{-1}$ , where  $\mathbf{R}$  is a constant invertible  $q \times q$  matrix and  $\mathbf{V}$  is an  $(n \times q)$  matrix given by

$$\mathbf{V} = \mathbf{W}\tilde{\mathbf{v}}. \quad (26)$$

Thus, we obtain a block-processing implementation of the MIMO blind MMSE equalization algorithm that is summarized in Algorithm 2. Note that the  $q \times q$  constant matrix

$\mathbf{R}$  comes from the inherent indeterminacies of MIMO blind identification systems using second-order statistics [15]. Usually, this indeterminacy is solved by applying some blind source separation algorithms.

### 3.4. Selection of the equalizer delay

It is known that the choice of the equalizer delay may affect significantly the equalization performance in SIMO and MIMO systems. In particular, nonzero-delay equalizers can have much improved performance compared to the zero-delay ones [10]. Indeed, one can write the spatiotemporal vector in (2) as follows:

$$\mathbf{x}_N(t) = \sum_{k=0}^{m-1} \mathbf{G}_k \mathbf{s}(t-k) + \mathbf{b}_N(t), \quad (27)$$

where  $\mathbf{G}_k$  is defined in (6) and represents a submatrix of  $\mathbf{H}_N$  given by the column vectors of indices varying in the range  $[kq+1, \dots, (k+1)q]$ . One can observe that  $\|\mathbf{G}_0\| \leq \|\mathbf{G}_1\| \leq \dots \leq \|\mathbf{G}_L\| = \|\mathbf{G}_{L+1}\| = \dots = \|\mathbf{G}_{N-1}\|$  and  $\|\mathbf{G}_{N-1}\| \geq \|\mathbf{G}_N\| \geq \dots \geq \|\mathbf{G}_{d-1}\|$ . In other words, the input symbols with delays  $\tau$ ,  $L \leq \tau \leq N-1$  are multiplied in (27) by (matrix) factors of maximum norm. Consequently, the best equalizer delay belongs, in general, to the range  $[L, \dots, N-1]$ . One can observe also that, the performance gain of the nonzero equalizer with delay in the range  $[L, \dots, N-1]$  can be large compared to that of equalizers with extreme delays, that is,  $\tau = 0$  or  $\tau = d-1$ . The gain difference becomes, in general, negligible when we consider two equalizers with delays belonging to the interval  $[L, \dots, N-1]$  (see [10]). Hence, in practice, the search for the optimal equalizer delay is computationally expensive and worthless and it is often sufficient to choose a *good* delay in the range  $[L, \dots, N-1]$ , for example,  $\tau = L$  as we did in this paper.

Moreover, it is shown in Section 5 that the blind estimation of the MMSE filter results in a performance loss compared to the nonblind one. To compensate for this performance loss and also to have a controlled nonzero equalization delay which helps to improve performance of the equalizer, we propose here a two-step approach to estimate the blind MMSE equalizer. In the first step, we estimate  $\mathbf{V}_0$  according to the previous algorithms, while, in the second step, we refine this estimation by exploiting the a priori knowledge of the finite alphabet to which belongs the symbols  $\mathbf{s}(t)$ . This

$$\begin{aligned}
\mathbf{C} &= \frac{1}{K} \sum_{t=0}^{K-1} \mathbf{x}_N(t) \mathbf{x}_N^H(t), \text{ (} K \text{: sample size)} \\
(\mathbf{W}, \Lambda) &= \text{eigs}(\mathbf{C}, d), \text{ (extracts the } d \text{ principal eigenvectors of } \mathbf{C}) \\
\mathbf{T} &= \mathbf{C}(p+1:n, :)\mathbf{W} \\
\mathbf{a} &= \text{vec}(\mathbf{T}(:, 1:q)) \\
\mathbf{A} &= (\mathbf{I}_q \otimes \mathbf{T})\mathbf{J} \\
\bar{\mathbf{v}} &= -\mathbf{A}^{\#}\mathbf{a} \\
\tilde{\mathbf{V}} &= \text{vec}^{-1}(\mathbf{J}\bar{\mathbf{v}}) + \mathbf{J}_{0,q,d-q} \\
\mathbf{V} &= \mathbf{W}\tilde{\mathbf{V}}
\end{aligned}$$

ALGORITHM 2: MIMO blind MMSE equalization algorithm.

Estimate  $\hat{\mathbf{s}}(t)$ ,  $t = 0 \dots K - 1$ , using  $\mathbf{V}$  given by Algorithm 1 or Algorithm 2 followed by BSS (e.g., ACMA in [17]).

$$\begin{aligned}
\mathbf{G}_\tau &= \frac{1}{K} \sum_{t=\tau}^{K+\tau-1} \mathbf{x}_N(t) \hat{\mathbf{s}}^H(t - \tau) \\
\mathbf{V}_\tau &= \mathbf{C}^{-1} \mathbf{G}_\tau
\end{aligned}$$

ALGORITHM 3: Two-step equalization procedure.

is done by performing a hard decision on the symbols that are then used to reestimate  $\mathbf{V}_\tau$  according to (4) and (6).<sup>2</sup>

More precisely, operating with equalizer filter  $\mathbf{V}$  in (26) (or in (19) for the SIMO case) on the received data vector  $\mathbf{x}_N(t)$  in (2), we obtain, according to (9) and (16), an estimation of the emitted signal  $\tilde{\mathbf{s}}(t) = \mathbf{V}^H \mathbf{x}_N(t) = \mathbf{R}^H \mathbf{V}_0^H \mathbf{x}_N(t)$ , as  $\mathbf{V}_0^H \mathbf{x}_N(t) = \mathbf{s}(t) + \boldsymbol{\epsilon}(t)$ , where  $\boldsymbol{\epsilon}(t)$  represents the residual estimation error (of minimum variance) of  $\mathbf{s}(t)$ , it follows that

$$\tilde{\mathbf{s}}(t) = \mathbf{R}^H \mathbf{s}(t) + \tilde{\boldsymbol{\epsilon}}(t), \quad (28)$$

where  $\tilde{\boldsymbol{\epsilon}}(t) = \mathbf{R}^H \boldsymbol{\epsilon}(t)$ . It is clear from (28), that the estimated signal  $\tilde{\mathbf{s}}(t)$  is an instantaneous mixture of the emitted signal  $\mathbf{s}(t)$  corrupted by an additive colored noise  $\tilde{\boldsymbol{\epsilon}}(t)$ . Thus, an identification of  $\mathbf{R}$  (i.e., resolving the ambiguity) is then necessary to extract the original signal and to decrease the mean square error (MSE) towards zero. This is achieved by applying (in batch or adaptive way) a blind source separation (BSS) algorithm to the equalizer output (28), followed by a hard decision on the symbols. In this paper, we have used the ACMA algorithm (analytical constant modulus algorithm) in [17] for batch processing implementation and the A-CMS algorithm (adaptive constant modulus separation) in [18] for adaptive implementation. Indeed, constant modulus algorithms (CMA)-like algorithms (ACMA and A-CMS) have relatively low cost and are very efficient in separating (finite alphabet) communication signals. The two-step

blind MMSE equalization algorithms are summarized in Algorithms 1, 2, and 3.

### 3.5. Robustness

We study here the robustness of the proposed blind MMSE equalizer against channel order overestimation errors. Let us consider, for simplicity, the SIMO case where the channel order is used to determine the column dimension equal to  $d = L + N$  of matrix  $\mathbf{W}$  (which corresponds, in practice, to the size of the dominant subspace of  $\mathbf{C}$ ). Let  $L' > L$  be the over-estimated channel order and hence  $d' = L' + N$  is the column dimension of  $\mathbf{W}$ , that is, we consider the subspace spanned by the  $d'$  dominant eigenvector of  $\mathbf{C}$ . We argue here that, as long as the number of sensors  $p$  plus the overestimation error order  $L' - L$  is smaller than the noise subspace dimension, that is,  $p + L' - L < n - d$ , the least squares solution of (14) provides a consistent estimate of the MMSE equalizer. This observation comes from the following.

Note that, using (5), matrix  $\bar{\mathbf{C}}$  defined in (13) is expressed as  $\bar{\mathbf{C}} = [\mathbf{H}' \ \mathbf{C}']$ , where  $\mathbf{H}'$  is an  $(n - p) \times p$ -dimensional matrix and  $\mathbf{C}' = \mathbf{H}_{N-1} \mathbf{H}_{N-1}^H + \sigma_b^2 \mathbf{I}_{n-p}$  an  $(n - p) \times (n - p)$  full-rank matrix. It follows that the right null space of  $\bar{\mathbf{C}}$ ,  $\text{null}_r(\bar{\mathbf{C}}) = \{\mathbf{z} \in \mathbb{C}^n : \bar{\mathbf{C}}\mathbf{z} = \mathbf{0}\}$ , is a  $p$ -dimensional subspace. Now, one can observe that only one direction of  $\text{null}_r(\bar{\mathbf{C}})$  belongs to the signal subspace since  $\text{null}_r(\bar{\mathbf{C}}) \cap \text{range}(\mathbf{H}_N) = \text{null}_r(\bar{\mathbf{C}}\mathbf{H}_N) = \text{null}_r(\bar{\mathbf{C}}\mathbf{W})$  (the last equality comes from the fact that  $\mathbf{H}_N$  and  $\mathbf{W}$  span both the same (signal) subspace). According to the proof of Theorem 1,  $\dim(\text{null}_r(\bar{\mathbf{C}}\mathbf{W})) = 1$ .

Let  $\mathbf{b}_1, \dots, \mathbf{b}_p$  be a basis of  $\text{null}_r(\bar{\mathbf{C}})$  such that  $\mathbf{b}_1$  belongs to the signal subspace (i.e.,  $\text{range}(\mathbf{H}_N)$ ). Now, the solution of

<sup>2</sup> We assume here the use of a differential modulation to get rid of the phase indeterminacy inherent to the blind equalization problem.

(14) would be unique (up to a scalar constant) if

$$\text{range}(\mathbf{W}) \cap \text{range}\left(\begin{bmatrix} \mathbf{b}_1 & \cdots & \mathbf{b}_p \end{bmatrix}\right) = \text{range}(\mathbf{b}_1), \quad (29)$$

or equivalently

$$\text{range}(\mathbf{W}) \cap \text{range}\left(\begin{bmatrix} \mathbf{b}_2 & \cdots & \mathbf{b}_p \end{bmatrix}\right) = \{\mathbf{0}\}. \quad (30)$$

The above condition would be verified if the intersection of the subspace spanned by the projections of  $\mathbf{b}_2, \dots, \mathbf{b}_p$  onto the noise subspace and the subspace spanned by the  $L' - L$  noise vectors of  $\mathbf{W}$  introduced by the overestimation error is empty (except for the zero vector). As the latter are randomly introduced by the eigenvalue decomposition (EVD) of  $\mathbf{C}$  and since  $p + L' - L < n - d$ , then one can expect this subspace intersection to be empty almost surely.

Note also that, by using linear constraint, one obtains better robustness than with quadratic constraint. The reason is that the solution of (14) is, in general, a linear combination of the desired solution  $\mathbf{v}_0$  (that lives in the signal subspace) and noise subspace vectors (introduced by the channel order overestimation errors). However, it is observed that, for a finite sample size and for moderate and high SNRs the contribution of the desired solution  $\mathbf{v}_0$  in (14) is much higher than that of the noise subspace vectors. This is due to the fact that the low energy output of the noise subspace vectors comes from their orthogonality with the system matrix  $\mathbf{H}_N$  (this is a structural property, independent of the sample size), while the desired solution  $\mathbf{v}_0$  belongs to the kernel of  $\bar{\mathbf{C}}$  due to the decorrelation (whiteness) property of the input signal which is valid asymptotically for large sample size. Indeed, one can observe (see Figure 6) that when increasing  $K$  (the sample size), the robustness of the quadratically constrained equalizer improves significantly. Consequently, in the context of small or moderate sample sizes, solving (14) in the least squares sense under unit norm constraint leads to a solution that lives almost in the noise subspace (i.e., the part of  $\mathbf{v}_0$  in the final solution becomes very small). On the other hand, by solving (14) subject to linear constraints (24) and (25), one obtains a solution where the linear factor of  $\mathbf{v}_0$  is more significant (which is due to the fact that vector  $\mathbf{a}$  in (24) belongs to the range subspace of  $\mathbf{A}$ ).

This argument, eventhough not a rigorous proof of robustness, has been confirmed by our simulation results (see simulation example given below where one can see that the performance loss of the equalization due to the channel order overestimation error remains relatively limited).

#### 4. FAST ADAPTIVE IMPLEMENTATION

In tracking applications, we are interested in estimating the equalizer vector recursively with low computational complexity. We introduce here a fast adaptive implementation of the proposed blind MMSE equalization algorithms. The computational reduction is achieved by exploiting the idea of the projection approximation [19] and the shift-invariance property of the temporal data covariance matrices [20].

Matrix  $\mathbf{C}$  is replaced by its recursive estimate

$$\mathbf{C}(t) = \sum_{k=0}^t \beta^{t-k} \mathbf{x}_N(k) \mathbf{x}_N^H(k) = \beta \mathbf{C}(t-1) + \mathbf{x}_N(t) \mathbf{x}_N^H(t), \quad (31)$$

where  $0 < \beta < 1$  is a forgetting factor. The weight matrix  $\mathbf{W}$  corresponding to the  $d$  dominant eigenvectors of  $\mathbf{C}$  can be estimated using a fast subspace estimation and tracking algorithm. In this paper, we use the YAST algorithm (yet another subspace tracker) [21]. The choice of YAST algorithm is motivated by its remarkable tracking performance compared to other existing subspace tracking algorithms of similar computational complexity (PAST [19], OPAST [22], etc.). The YAST algorithm is summarized in Algorithm 4. Note that only  $O(nd)$  operations are required at each time instant (instead of  $O(n^3)$  for a full EVD). Vector  $\mathbf{x}'(t) = \mathbf{C}(t-1) \mathbf{x}_N(t)$  in Algorithm 4 can be computed in  $O(n)$  operations, by using the shift-invariance property of the correlation matrix, as seen in Appendix A.

Applying, to (12), the projection approximation

$$\mathbf{C}(t) \mathbf{W}(t) \approx \mathbf{C}(t) \mathbf{W}(t-1), \quad (32)$$

which is valid if matrix  $\mathbf{W}(t)$  is slowly varying with time [22], yields

$$\mathbf{T}(t) = \beta \mathbf{T}(t-1) + \mathbf{J}_{p,n-p,0}^T \mathbf{x}_N(t) \mathbf{y}^H(t), \quad (33)$$

where vector  $\mathbf{J}_{p,n-p,0}^T \mathbf{x}_N(t)$  is a subvector of  $\mathbf{x}_N(t)$  given by its last  $(n-p)$  elements and vector  $\mathbf{y}(t) = \mathbf{W}^H(t-1) \mathbf{x}_N(t)$  is computed by YAST (cf. Algorithm 4).

##### 4.1. The SIMO case

In this case, our objective is to estimate recursively the  $d$ -dimensional vector  $\tilde{\mathbf{v}}$  in (17) as the least eigenvector of matrix  $\mathbf{Q}$  or equivalently as the dominant eigenvector of its inverse.<sup>3</sup> Using (18), (33) can be replaced by the following recursion:

$$\mathbf{Q}(t) = \beta^2 \mathbf{Q}(t-1) - \mathbf{D}_Q(t) \mathbf{\Gamma}_Q^{-1}(t) \mathbf{D}_Q^H(t), \quad (34)$$

where  $\mathbf{D}_Q(t)$  is the  $d \times 2$  matrix

$$\mathbf{D}_Q(t) = \begin{bmatrix} \beta \mathbf{T}^H(t-1) \mathbf{J}_{p,n-p,0}^T \mathbf{x}_N(t) & \mathbf{y}(t) \end{bmatrix}, \quad (35)$$

and  $\mathbf{\Gamma}_Q(t)$  is the  $2 \times 2$  nonsingular matrix

$$\mathbf{\Gamma}_Q(t) = \begin{bmatrix} \|\mathbf{J}_{p,n-p,0}^T \mathbf{x}_N(t)\|^2 & -1 \\ -1 & 0 \end{bmatrix}. \quad (36)$$

Consider the  $d \times d$  Hermitian matrix  $\mathbf{F}(t) \stackrel{\text{def}}{=} \mathbf{Q}^{-1}(t)$ , using the matrix (Schur) inversion lemma [1], we obtain

$$\mathbf{F}(t) = \frac{1}{\beta^2} \mathbf{F}(t-1) + \mathbf{D}_F(t) \mathbf{\Gamma}_F(t) \mathbf{D}_F^H(t), \quad (37)$$

<sup>3</sup>  $\mathbf{Q}$  is a singular matrix when dealing with the exact statistics. However, when considering the sample averaged estimate of  $\mathbf{C}$ , due to the estimation errors and the projection approximation, the estimate of  $\mathbf{Q}$  is almost surely a nonsingular matrix.

$$\begin{aligned}
\mathbf{y}(t) &= \mathbf{W}^H(t-1)\mathbf{x}_N(t) \\
\mathbf{x}'(t) &= \mathbf{C}(t-1)\mathbf{x}_N(t) \\
\mathbf{y}'(t) &= \mathbf{W}^H(t-1)\mathbf{x}'(t) \\
\sigma(t) &= (\mathbf{x}_N^H(t)\mathbf{x}_N(t) - \mathbf{y}^H(t)\mathbf{y}(t))^{1/2} \\
\mathbf{h}(t) &= \mathbf{Z}(t-1)\mathbf{y}(t) \\
\gamma(t) &= (\beta + \mathbf{y}^H(t)\mathbf{h}(t))^{-1} \\
\tilde{\mathbf{Z}}(t) &= \frac{1}{\beta}(\mathbf{Z}(t-1) - \mathbf{h}(t)\gamma(t)\mathbf{h}^H(t)) \\
\alpha(t) &= \mathbf{x}_N^H(t)\mathbf{x}_N(t) \\
\mathbf{y}''(t) &= \beta\mathbf{y}'(t) + \mathbf{y}(t)\alpha(t) \\
c_{yy}(t) &= \beta\mathbf{x}_N^H(t)\mathbf{x}'(t) + \alpha^*(t)\alpha(t) \\
\mathbf{h}'(t) &= \tilde{\mathbf{Z}}(t-1)\mathbf{y}''(t) \\
\gamma'(t) &= (c_{yy}(t) - [\mathbf{y}''(t)]^H\mathbf{h}'(t))^{-1} \\
\mathbf{h}''(t) &= \mathbf{h}'(t) - \mathbf{y}(t) \\
\tilde{\mathbf{Z}}'(t) &= \tilde{\mathbf{Z}}(t) + \mathbf{h}''(t)\gamma'(t)[\mathbf{h}''(t)]^H \\
\mathbf{g}(t) &= \mathbf{h}''(t)\gamma'(t)\sigma^*(t) \\
\mathbf{y}'''(t) &= \sigma(t)\gamma'(t)\sigma^*(t) \\
\mathbf{Z}'(t) &= [\tilde{\mathbf{Z}}'(t), -\mathbf{g}(t); -\mathbf{g}^H(t), \mathbf{y}'''(t)] \\
(\phi(t), \lambda(t)) &= \text{eigs}(\mathbf{Z}'(t), 1) \\
\boldsymbol{\varphi}(t) &= \boldsymbol{\phi}_{(1;d)}(t) \\
z(t) &= \boldsymbol{\phi}_{(d+1)}(t) \\
\rho(t) &= |z(t)| \\
\theta(t) &= e^{j\arg(z(t))}, \text{ (arg stands for the phase argument)} \\
\mathbf{f}(t) &= \boldsymbol{\varphi}(t)\theta^*(t) \\
\mathbf{f}'(t) &= \mathbf{f}(t)(1 + \rho(t))^{-1} \\
\mathbf{y}''''(t) &= \mathbf{y}(t)\sigma^{-1}(t) - \mathbf{f}'(t) \\
\mathbf{e}(t) &= \mathbf{x}(t)\sigma^{-1}(t) - \mathbf{W}(t-1)\mathbf{y}''''(t) \\
\mathbf{W}(t) &= \mathbf{W}(t-1) - \mathbf{e}(t)\mathbf{f}^H(t) \\
\mathbf{g}'(t) &= \mathbf{g}(t) + \mathbf{f}'(t)(\mathbf{y}''(t) - \theta(t)\lambda(t)\theta^*(t)) \\
\mathbf{Z}(t) &= \tilde{\mathbf{Z}}'(t) + \mathbf{g}'(t)[\mathbf{f}'(t)]^H + \mathbf{f}'(t)\mathbf{g}'^H(t)
\end{aligned}$$

ALGORITHM 4: YAST algorithm.

where  $\mathbf{D}_F(t)$  is the  $d \times 2$  matrix

$$\mathbf{D}_F(t) = \frac{1}{\beta^2}\mathbf{F}(t-1)\mathbf{D}_Q(t), \quad (38)$$

and  $\Gamma_F(t)$  is the  $2 \times 2$  matrix

$$\Gamma_F(t) = (\Gamma_Q(t) - \mathbf{D}_F^H(t)\mathbf{D}_Q(t))^{-1}. \quad (39)$$

The extraction of the dominant eigenvector of  $\mathbf{F}(t)$  is obtained by power iteration as

$$\tilde{\mathbf{v}}(t) = \frac{\mathbf{F}(t)\tilde{\mathbf{v}}(t-1)}{\|\mathbf{F}(t)\tilde{\mathbf{v}}(t-1)\|}. \quad (40)$$

The complete pseudocode for the SIMO adaptive blind MMSE equalization algorithm is given in Algorithm 5. Note that the whole processing requires only  $O(nd)$  flops per iteration.

Update  $\mathbf{W}(t)$  and  $\mathbf{y}(t)$  using YAST (cf. Algorithm 4)

$$\begin{aligned}
\bar{\mathbf{x}}(t) &= \mathbf{x}_N(t)_{(p+1:n)} \\
\Gamma_Q(t) &= \begin{bmatrix} \|\bar{\mathbf{x}}(t)\|^2 & -1 \\ -1 & 0 \end{bmatrix} \\
\mathbf{D}_Q(t) &= [\beta\mathbf{T}^H(t-1)\bar{\mathbf{x}}(t) \quad \mathbf{y}(t)] \\
\mathbf{D}_F(t) &= \frac{1}{\beta^2}\mathbf{F}(t-1)\mathbf{D}_Q(t) \\
\Gamma_F(t) &= (\Gamma_Q(t) - \mathbf{D}_F^H(t)\mathbf{D}_Q(t))^{-1} \\
\mathbf{F}(t) &= \frac{1}{\beta^2}\mathbf{F}(t-1) + \mathbf{D}_F(t)\Gamma_F(t)\mathbf{D}_F^H(t) \\
\tilde{\mathbf{v}}(t) &= \frac{\mathbf{F}(t)\tilde{\mathbf{v}}(t-1)}{\|\mathbf{F}(t)\tilde{\mathbf{v}}(t-1)\|} \\
\mathbf{v}(t) &= \mathbf{W}(t)\tilde{\mathbf{v}}(t) \\
\mathbf{T}(t) &= \beta\mathbf{T}(t-1) + \bar{\mathbf{x}}(t)\mathbf{y}^H(t)
\end{aligned}$$

ALGORITHM 5: SIMO adaptive blind equalization algorithm.

## 4.2. The MIMO case

Here, we introduce a fast adaptive version of the MIMO blind MMSE equalization algorithm given in Algorithm 2. First note that, due to the projection approximation and the finite sample size effect, matrix  $\mathbf{A}$  is almost surely full column rank and hence

$$\mathbf{A}^\# = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H. \quad (41)$$

Therefore vector  $\bar{\mathbf{v}}$  in (24) can be expressed as

$$\bar{\mathbf{v}}(t) = [\bar{\mathbf{v}}_1^T(t) \quad \bar{\mathbf{v}}_2^T(t) \quad \cdots \quad \bar{\mathbf{v}}_q^T(t)]^T, \quad (42)$$

where vectors  $\bar{\mathbf{v}}_k(t)$ , for  $k = 1, \dots, q$ , are given by

$$\begin{aligned}
\bar{\mathbf{v}}_k(t) &= -\mathbf{F}_k(t)\mathbf{f}_k(t), \\
\mathbf{F}_k(t) &= (\mathbf{J}_k^T\mathbf{Q}(t)\mathbf{J}_k)^{-1}, \\
\mathbf{f}_k(t) &= \mathbf{J}_k^T\mathbf{Q}(t)\mathbf{J}_{k-1,1,d-k}.
\end{aligned} \quad (43)$$

Using (34) and the matrix (Schur) inversion lemma [1], matrix  $\mathbf{F}_k(t)$  can be updated by the following recursion:

$$\begin{aligned}
\mathbf{F}_k(t) &= \frac{1}{\beta^2}\mathbf{F}_k(t-1) + \mathbf{D}_{F_k}(t)\Gamma_{F_k}(t)\mathbf{D}_{F_k}^H(t), \\
\mathbf{D}_{F_k}(t) &= \frac{1}{\beta^2}\mathbf{F}_k(t-1)\mathbf{J}_k^T\mathbf{D}_Q(t), \\
\Gamma_{F_k}(t) &= (\Gamma_Q(t) - \mathbf{D}_{F_k}^H(t)\mathbf{J}_k^T\mathbf{D}_Q(t))^{-1},
\end{aligned} \quad (44)$$

where matrices  $\mathbf{D}_Q(t)$  and  $\Gamma_Q(t)$  are given by (35) and (36).

Algorithm 6 summarizes the fast adaptive version of the MIMO blind MMSE equalization algorithm. Note that the whole processing requires only  $O(qnd)$  flops per iteration.

## 4.3. Two-step procedure

Let  $\mathbf{W} \in \mathbb{C}^{n \times d}$  be an orthonormal basis of the signal subspace. Since  $\mathbf{G}_T$  belongs to the signal subspace, one can write

Update  $\mathbf{W}(t)$  and  $\mathbf{y}(t)$  using YAST (cf. Algorithm 4)

$$\bar{\mathbf{x}}(t) = \mathbf{x}_N(t)_{(p+1:n)}$$

$$\mathbf{\Gamma}_Q(t) = \begin{bmatrix} \|\bar{\mathbf{x}}(t)\|^2 & -1 \\ -1 & 0 \end{bmatrix}$$

$$\mathbf{D}_Q(t) = \begin{bmatrix} \beta \mathbf{T}^H(t-1) \bar{\mathbf{x}}(t) & \mathbf{y}(t) \end{bmatrix}$$

$$\mathbf{Q}(t) = \beta^2 \mathbf{Q}(t-1) - \mathbf{D}_Q(t) \mathbf{\Gamma}_Q^{-1}(t) \mathbf{D}_Q^H(t)$$

For  $k = 1, \dots, q$ :

$$\mathbf{f}_k(t) = \mathbf{Q}(t)_{(k+1:d,k)}$$

$$\mathbf{D}_{F_k}(t) = \frac{1}{\beta^2} \mathbf{F}_k(t-1) \mathbf{D}_Q(t)_{(k+1:d,:)}$$

$$\mathbf{\Gamma}_{F_k}(t) = (\mathbf{\Gamma}_Q(t) - \mathbf{D}_{F_k}^H(t) \mathbf{D}_Q(t)_{(k+1:d,:)} )^{-1}$$

$$\mathbf{F}_k(t) = \frac{1}{\beta^2} \mathbf{F}_k(t-1) + \mathbf{D}_{F_k}(t) \mathbf{\Gamma}_{F_k}(t) \mathbf{D}_{F_k}^H(t)$$

$$\bar{\mathbf{V}}_k(t) = -\mathbf{F}_k(t) \mathbf{f}_k(t)$$

end

$$\bar{\mathbf{V}}(t) = [\bar{\mathbf{V}}_1^T(t) \quad \bar{\mathbf{V}}_2^T(t) \quad \dots \quad \bar{\mathbf{V}}_q^T(t)]^T$$

$$\tilde{\mathbf{V}}(t) = \text{vec}^{-1}(\mathbf{J} \bar{\mathbf{V}}(t)) + \mathbf{J}_{0,q,d-q}$$

$$\mathbf{V}(t) = \mathbf{W}(t) \tilde{\mathbf{V}}(t)$$

$$\mathbf{T}(t) = \beta \mathbf{T}(t-1) + \bar{\mathbf{x}}(t) \mathbf{y}^H(t)$$

ALGORITHM 6: MIMO adaptive blind MMSE equalization algorithm.

(see [23])

$$\mathbf{V}_\tau = \mathbf{W}(\mathbf{W}^H \mathbf{C} \mathbf{W})^{-1} \mathbf{W}^H \mathbf{G}_\tau. \quad (45)$$

This expression of  $\mathbf{V}_\tau$  is used for the fast adaptive implementation of the two-step algorithm since  $\mathbf{Z} = (\mathbf{W}^H \mathbf{C} \mathbf{W})^{-1}$  is already computed by the YAST. The recursive expression of vector  $\mathbf{G}_\tau$  is given by

$$\mathbf{G}_\tau(t) = \beta \mathbf{G}_\tau(t-1) + \mathbf{x}_N(t) \hat{\mathbf{s}}^H(t-\tau), \quad (46)$$

where  $\hat{\mathbf{s}}(t)$  is an estimate of  $\mathbf{s}(t)$  given by applying a BSS to  $\tilde{\mathbf{s}}(t)$  in (28). In our simulation, we used the A-CMS algorithm in [18]. Thus, (45) can be replaced by the following recursion:

$$\begin{aligned} \mathbf{V}_\tau(t) &= \beta \mathbf{V}_\tau(t-1) + \mathbf{z}(t) \hat{\mathbf{s}}^H(t-\tau), \\ \mathbf{z}(t) &= \mathbf{W}(t) \mathbf{Z}(t) \mathbf{W}^H(t) \mathbf{x}_N(t). \end{aligned} \quad (47)$$

Note that, by choosing a nonzero equalizer delay  $\tau$ , we improve the equalization performance as shown below. The adaptive two-step blind MMSE equalization algorithm is summarized in Algorithms 5, 6, and 7. The overall computational cost of this algorithm is  $(q+8)nd + O(qn + qd^2)$  flops per iteration.

## 5. PERFORMANCE ANALYSIS

As mentioned above, the extraction of the equalizer matrix needs some blind source separation algorithms to solve the indeterminacy problem which is inherent to second-order

Estimate  $\hat{\mathbf{s}}(t)$ , using  $\mathbf{V}(t)$  given by Algorithm 5 or Algorithm 6 followed by BSS (e.g., A-CMS in [18]).

$$\begin{aligned} \mathbf{z}(t) &= \mathbf{W}(t) \mathbf{Z}(t) \mathbf{W}^H(t) \mathbf{x}_N(t) \\ \mathbf{V}_\tau(t) &= \beta \mathbf{V}_\tau(t-1) + \mathbf{z}(t) \hat{\mathbf{s}}^H(t-\tau) \end{aligned}$$

ALGORITHM 7: Adaptive two-step equalization procedure.

MIMO blind identification methods. Thus, the performance of our MIMO equalization algorithms depends, in part, on the choice of the blind source separation algorithm which leads to a very cumbersome asymptotic convergence analysis. For simplicity, we study the asymptotic expression of the estimated zero-delay blind equalization MSE in the SIMO case only, where, the equalizer vector is given up to an unknown nonzero scalar constant. To evaluate the performance of our algorithm, this constant is estimated according to

$$r = \arg \min_{\alpha} \|\mathbf{v}_0 - \alpha \mathbf{v}\|^2 = \frac{\mathbf{v}^H \mathbf{v}_0}{\|\mathbf{v}\|^2}, \quad (48)$$

where  $\mathbf{v}_0$  represents the exact value of the zero-delay MMSE equalizer and  $\mathbf{v}$  the blind MMSE equalizer presented previously.

### 5.1. Asymptotic performance loss

Theoretically, the optimal MSE is given by

$$\text{MSE}_{\text{opt}} = E(|s(t) - \mathbf{v}_0^H \mathbf{x}_N(t)|^2) = 1 - \mathbf{g}_0^H \mathbf{C}^{-1} \mathbf{g}_0, \quad (49)$$

where vector  $\mathbf{g}_0$  is given by (6) (for  $q = 1, \tau = 0$ ). Let  $\widehat{\text{MSE}}_{\text{opt}}$  denotes the MSE reached by  $\hat{\mathbf{v}}_0$  the estimate of  $\mathbf{v}_0$ :

$$\widehat{\text{MSE}}_{\text{opt}} \stackrel{\text{def}}{=} E(|s(t) - \hat{\mathbf{v}}_0^H \mathbf{x}_N(t)|^2). \quad (50)$$

In terms of MSE, the blind estimation leads to a performance loss equal to

$$\widehat{\text{MSE}}_{\text{opt}} - \text{MSE}_{\text{opt}} = \text{trace}(\mathbf{C}(\hat{\mathbf{v}}_0 - \mathbf{v}_0)(\hat{\mathbf{v}}_0 - \mathbf{v}_0)^H). \quad (51)$$

Asymptotically (i.e., for large sample sizes  $K$ ), this performance loss is given by

$$\varepsilon \stackrel{\text{def}}{=} \lim_{K \rightarrow +\infty} KE(\widehat{\text{MSE}}_{\text{opt}} - \text{MSE}_{\text{opt}}) = \text{trace}(\mathbf{C} \boldsymbol{\Sigma}_{\mathbf{v}}), \quad (52)$$

where  $\boldsymbol{\Sigma}_{\mathbf{v}}$  is the asymptotic covariance matrix of vector  $\hat{\mathbf{v}}_0$ . As  $\hat{\mathbf{v}}_0$  is a "function" of the sample covariance matrix of the observed signal  $\mathbf{x}_N(t)$ , denoted here by  $\hat{\mathbf{C}}$  and given, from  $K$ -sample observation, by

$$\hat{\mathbf{C}} = \frac{1}{K} \sum_{t=0}^{K-1} \mathbf{x}_N(t) \mathbf{x}_N^H(t), \quad (53)$$

it is clear that  $\boldsymbol{\Sigma}_{\mathbf{v}}$  depends on the asymptotic covariance matrix of  $\hat{\mathbf{C}}$ . The following lemma gives the explicit expression of the asymptotic covariance matrix of the random vector  $\hat{\mathbf{C}} = \text{vec}(\hat{\mathbf{C}})$ .



**Lemma 1.** Let  $\mathbf{C}_\tau$  be the  $\tau$ -lag covariance matrix of the signal  $\mathbf{x}_N(t)$  defined by

$$\mathbf{C}_\tau \stackrel{\text{def}}{=} E(\mathbf{x}_N(t + \tau)\mathbf{x}_N^H(t)) \quad (54)$$

and let  $\text{cum}(x_1, x_2, \dots, x_k)$  be the  $k$ th-order cumulant of the random variables  $(x_1, x_2, \dots, x_k)$ .

Under the above data assumptions, the sequence of estimates  $\hat{\mathbf{c}} = \text{vec}(\hat{\mathbf{C}})$  is asymptotically normal with mean  $\mathbf{c} = \text{vec}(\mathbf{C})$  and covariance  $\Sigma_{\mathbf{c}}$ . That is,

$$\sqrt{K}(\hat{\mathbf{c}} - \mathbf{c}) \xrightarrow{\mathcal{L}} \mathcal{N}(\mathbf{0}, \Sigma_{\mathbf{c}}). \quad (55)$$

The covariance  $\Sigma_{\mathbf{c}}$  is given by

$$\Sigma_{\mathbf{c}} = \kappa \tilde{\mathbf{c}} \tilde{\mathbf{c}}^H + \sum_{\tau=-(m-1)}^{m-1} \mathbf{C}_\tau^T \otimes \mathbf{C}_\tau^H, \quad (56)$$

$$\tilde{\mathbf{c}} = \text{vec}(\mathbf{C} - \sigma_b^2 \mathbf{I}_n),$$

$$\kappa = \text{cum}(s(t), s^*(t), s(t), s^*(t)),$$

where  $\kappa$  is the kurtosis of the input signal  $s(t)$ .

*Proof.* see Appendix B.  $\square$

Now, to establish the asymptotic normality of vector estimate  $\hat{\mathbf{v}}_0$ , we use the so-called “continuity theorem,” which states that an asymptotically normal statistic transmits its asymptotic normality to any parameter vector estimated from it, as long as the mapping linking the statistic to the parameter vector is sufficiently regular in a neighborhood of the true (asymptotic) value of the statistic. More specifically, we have the following theorem [24].

**Theorem 2.** Let  $\boldsymbol{\theta}_K$  be an asymptotically normal sequence of random vectors, with asymptotic mean  $\boldsymbol{\theta}$  and asymptotic covariance  $\Sigma_{\boldsymbol{\theta}}$ . Let  $\boldsymbol{\omega} = [\omega_1 \ \dots \ \omega_{n_\omega}]^T$  be a real-valued vector function defined on a neighborhood of  $\boldsymbol{\theta}$  such that each component function  $\omega_k$  has nonzero differential at point  $\boldsymbol{\theta}$ , that is,  $\mathbf{D}\omega_k(\boldsymbol{\theta}) \neq \mathbf{0}$ ,  $k = 1, \dots, n_\omega$ . Then,  $\boldsymbol{\omega}(\boldsymbol{\theta}_K)$  is an asymptotically normal sequence of  $n_\omega$ -dimensional random vectors with mean  $\boldsymbol{\omega}(\boldsymbol{\theta})$  and covariance  $\Sigma = [\Sigma_{i,j}]_{1 \leq i,j \leq n_\omega}$  given by

$$\Sigma_{i,j} = \mathbf{D}\omega_i^T(\boldsymbol{\theta}) \Sigma_{\boldsymbol{\theta}} \mathbf{D}\omega_j(\boldsymbol{\theta}). \quad (57)$$

Applying the previous theorem to the estimate of  $\mathbf{v}_0$  leads to the following theorem.

**Theorem 3.** Under the above data assumptions and in the SIMO case ( $q = 1$ ), the random vector  $\hat{\mathbf{v}}_0$  is asymptotically Gaussian distributed with mean  $\mathbf{v}_0$  and covariance  $\Sigma_{\mathbf{v}}$ , that is,

$$\sqrt{K}(\hat{\mathbf{v}}_0 - \mathbf{v}_0) \xrightarrow{\mathcal{L}} \mathcal{N}(\mathbf{0}, \Sigma_{\mathbf{v}}). \quad (58)$$

The expression of  $\Sigma_{\mathbf{v}}$  is given by

$$\Sigma_{\mathbf{v}} = \mathbf{M} \Sigma_{\mathbf{c}} \mathbf{M}^H, \quad (59)$$

where  $\Sigma_{\mathbf{c}}$  is the asymptotic covariance matrix of the sample estimate of vector  $\mathbf{c} = \text{vec}(\mathbf{C})$  given in Lemma 1 and matrix  $\mathbf{M}$  is given by

$$\mathbf{M} = r \left( \mathbf{I}_n - \frac{\mathbf{v}\mathbf{v}^H}{\|\mathbf{v}\|^2} \right) [(\tilde{\mathbf{v}}^T \otimes \mathbf{I}_n) \boldsymbol{\Gamma} - \mathbf{W} \mathbf{M}_2 \mathbf{M}_1],$$

$$\boldsymbol{\Gamma} = \begin{bmatrix} \mathbf{W}^T(:, 1) \otimes (\lambda_1 \mathbf{I}_n - \mathbf{C})^\# \\ \vdots \\ \mathbf{W}^T(:, d) \otimes (\lambda_d \mathbf{I}_n - \mathbf{C})^\# \end{bmatrix},$$

$$\mathbf{M}_1 = \left[ (\mathbf{C} \mathbf{J}_{p,n-p,0} \mathbf{T})^T \otimes \mathbf{I}_d \right] \mathbf{U}_{n,d} \boldsymbol{\Gamma}^* \mathbf{U}_{n,n} + \left[ \mathbf{I}_d \otimes (\mathbf{T}^H \mathbf{J}_{p,n-p,0}^T \mathbf{C}) \right] \boldsymbol{\Gamma} + (\mathbf{J}_{p,n-p,0} \mathbf{T})^T \otimes \mathbf{W}^H + \mathbf{W}^T \otimes (\mathbf{T}^H \mathbf{J}_{p,n-p,0}^T),$$

$$\mathbf{M}_2 = \tilde{\mathbf{v}}^T \otimes \mathbf{Q}',$$

$$\mathbf{U}_{\alpha,\beta} = \sum_{i=1}^{\alpha} \sum_{j=1}^{\beta} \left( \mathbf{e}_i^\alpha [\mathbf{e}_j^\beta]^T \right) \otimes \left( \mathbf{e}_j^\beta [\mathbf{e}_i^\alpha]^T \right),$$

$$\mathbf{Q}' = \begin{cases} \mathbf{Q}^\#, & \text{in the quadratic constraint case} \\ \mathbf{J}_1 (\mathbf{J}_1^T \mathbf{Q} \mathbf{J}_1)^{-1} \mathbf{J}_1^T, & \text{in the linear constraint case,} \end{cases} \quad (60)$$

where  $\mathbf{U}_{\alpha,\beta}$  is a permutation matrix,  $\mathbf{e}_k^l$  denotes the  $k$ th column vector of matrix  $\mathbf{I}_l$  and  $\lambda_1 > \lambda_2 \geq \dots \geq \lambda_d$  are the  $d$  principal eigenvalues of  $\mathbf{C}$  associated to the eigenvectors  $\mathbf{W}(:, 1), \dots, \mathbf{W}(:, d)$ , respectively.

*Proof.* see Appendix C.  $\square$

## 5.2. Validation of the asymptotic covariance expressions and performance evaluation

In this section, we assess the performance of the blind equalization algorithm by Monte-Carlo experiments. We consider a SIMO channel ( $q = 1$ ,  $p = 3$ , and  $L = 4$ ), chosen randomly using Rayleigh distribution for each tap. The input signal is an iid QAM4 sequence. The width of the temporal window is  $N = 6$ . The theoretical expressions are compared with empirical estimates, obtained by Monte-Carlo simulations (100 independent Monte-Carlo simulations are performed in each experiment). The performance criterion used here is the relative mean square error (RMSE), defined as the sample average, over the Monte-Carlo simulations, of the total estimation of MSE loss, that is,  $\widehat{\text{MSE}}_{\text{opt}} - \text{MSE}_{\text{opt}}$ . This quantity is compared with its exact asymptotic expression divided by the sample size  $K$ ,  $\varepsilon_K = (1/K)\varepsilon = (1/K)\text{trace}(\mathbf{C}\Sigma_{\mathbf{v}})$ . The signal-to-noise ratio (SNR) is defined (in dB) by  $\text{SNR} = -20 \log(\sigma_b)$ .

Figure 1(a) compares, in the quadratic constraint case, the empirical RMSE (solid line) with the theoretical one  $\varepsilon_K$  (dashed line) as a function of the sample size  $K$ . The SNR is set to 15 dB. It is seen that the theoretical expression of

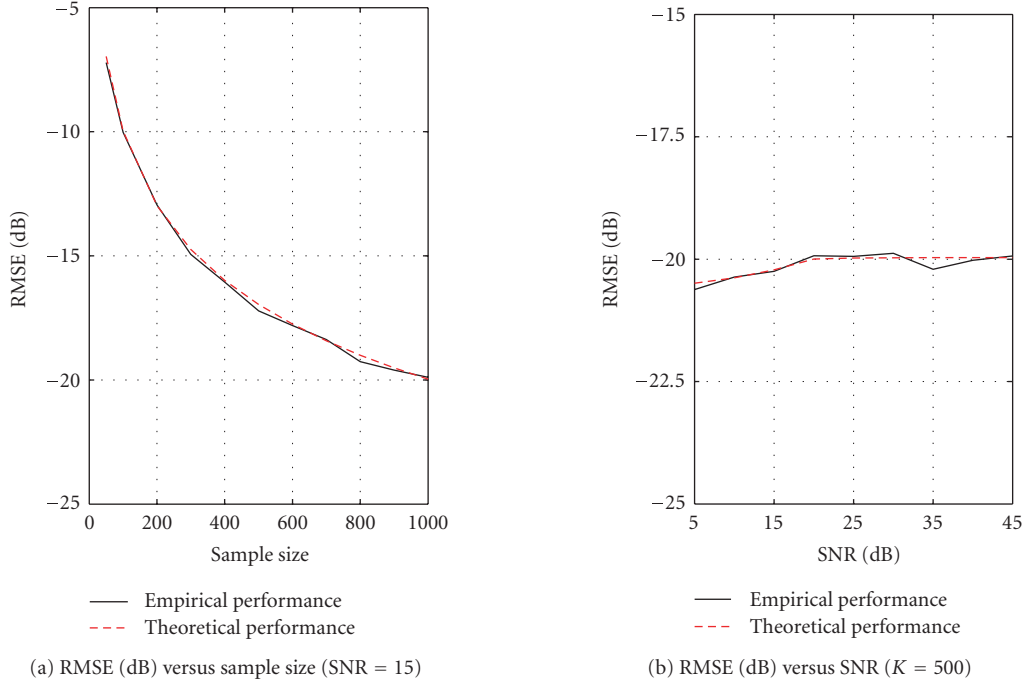


FIGURE 1: Asymptotic loss of performance: quadratic constraint.

the RMSE is valid from snapshot length as short as 50 samples, this means that the asymptotic conditions are reached for short sample size. In Figure 1(b) the empirical (solid line) and the theoretical (dashed line) RMSEs are plotted against the SNR. The sample size is set to  $K = 500$  samples. This figure demonstrates that there is a close agreement between theoretical and experimental values. Similar results are obtained when the linear constraint is used.

## 6. SIMULATION RESULTS AND DISCUSSION

We provide in this section some simulation examples to illustrate the performance of the proposed blind equalizer. Our tests are based on SIMO and MIMO channels. The channel coefficients are chosen randomly at each run according to a complex Gaussian distribution. The input signals are iid QAM4 sequences. As a performance measure, we estimate the average MSE given by

$$\text{MSE} = \frac{1}{q} E(\|\mathbf{s}(t - \tau) - \hat{\mathbf{V}}_{\tau}^H \mathbf{x}_N(t)\|^2), \quad (61)$$

over 100 Monte-Carlo runs. The MSE is compared to the optimal MSE given by

$$\text{MSE}_{\text{opt}} = \frac{1}{q} \text{trace}(\mathbf{I}_q - \mathbf{G}_{\tau}^H \mathbf{C}^{-1} \mathbf{G}_{\tau}). \quad (62)$$

### 6.1. Performance evaluation

In this experiment, we investigate the performance of our algorithm. In Figure 2(a) (SIMO case with quadratic con-

straint) and Figure 2(b) (MIMO case) we plot the MSE (in dB) against SNR (in dB) for  $K = 500$ . One can observe the performance loss of the zero-delay MMSE filter compared to the optimal one, due (as shown above) to the blind estimation procedure. Also, it illustrates the effectiveness of the two-step approach, which allows us to compensate for the performance loss and to choose a nonzero equalization delay, that improves the overall performance.

Figure 3(a) (SIMO case with quadratic constraint) and Figure 3(b) (MIMO case) represent the convergence rate of the adaptive algorithm with  $\text{SNR} = 15$  dB. Given the low computational cost of the algorithm, a relatively fast convergence rate is observed. Figure 4 compares, in fast time varying channel case, the tracking performance of the adaptive algorithm using respectively, YAST and OPASt as a subspace trackers. The channel variation model is the one given in [25] and the SNR is set to 15 dB. As we can observe, the adaptive equalization algorithm using YAST succeeds to track the channel variation, while it fails when using OPASt. Figure 5 compares the performance of our zero-delay MMSE equalizer with those given by the algorithms in [10, 11], respectively. The plot represents the estimated signal MSE versus the SNR for  $K = 500$ . As we can observe, our method outperforms the methods in [10, 11] for low SNRs.

### 6.2. Robustness to channel order overestimation errors

This experiment is dedicated to the study of the robustness against channel order overestimation errors. Figure 6(a) (resp., Figure 6(b)) represents the MSE versus the overestimated channel order for  $\text{SNR} = 15$  and  $K = 500$  (resp.,

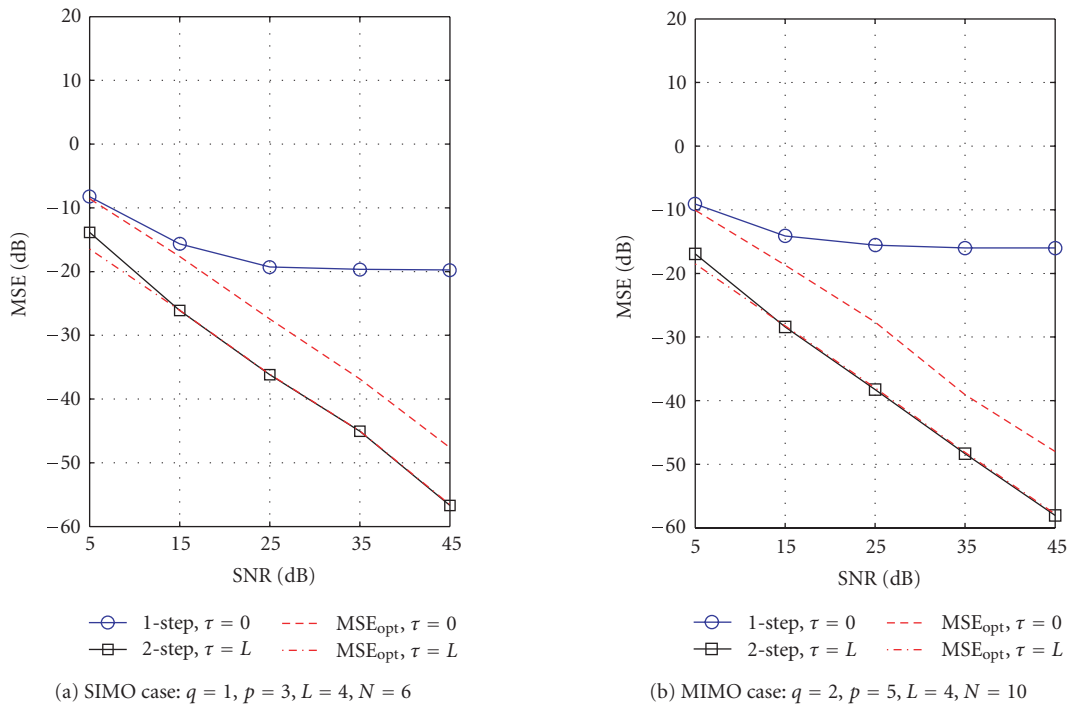


FIGURE 2: Performance of the equalizer.

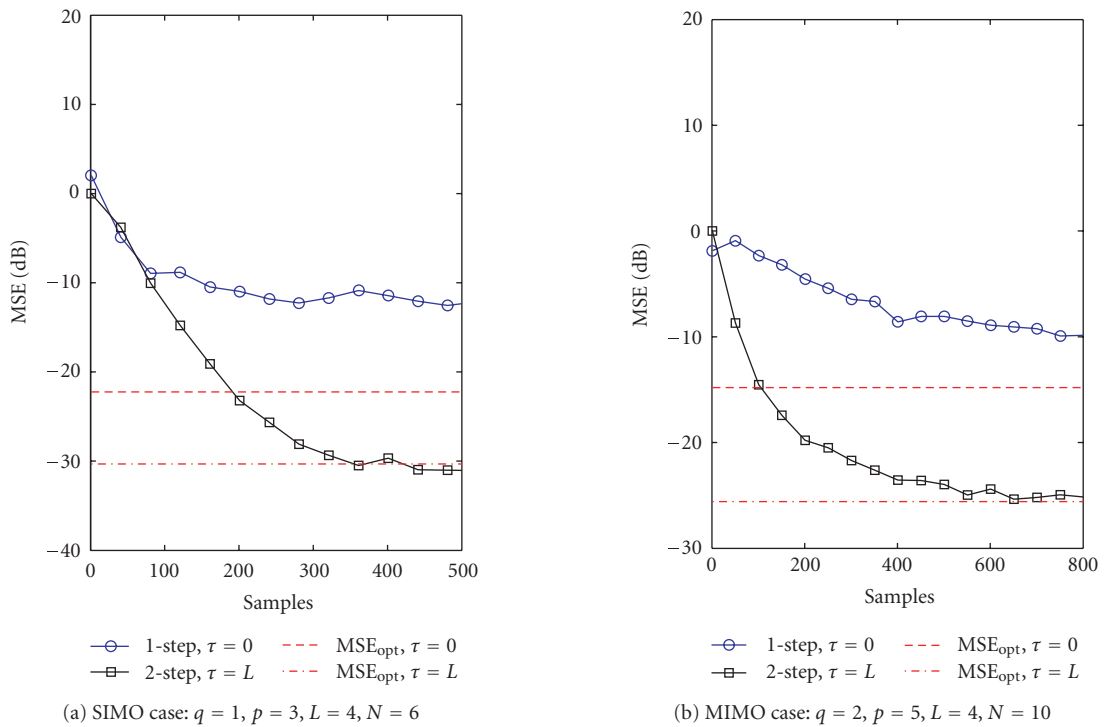


FIGURE 3: Convergence of the adaptive equalizer.

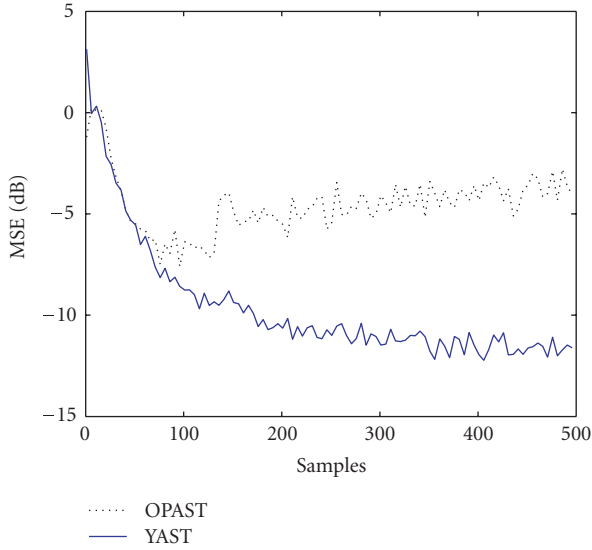


FIGURE 4: Convergence of the adaptive equalization algorithm in the time varying channel case.

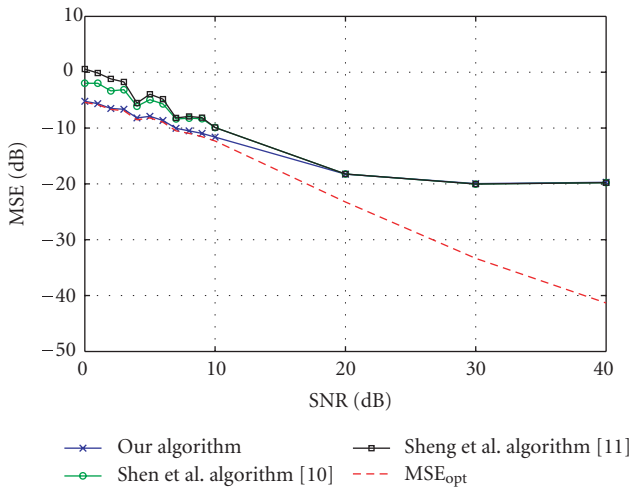


FIGURE 5: Performance comparison of batch-type SIMO equalizers ( $q = 1, p = 3, L = 4, N = 6$ ).

$K = 1000$ ). The plot compares, in the SIMO case, the MSE obtained by our algorithm using linear constraint (l.c.) and quadratic constraint (q.c.), respectively, to that obtained by algorithm in [10] (identical results are obtained with algorithm in [11]). Clearly, the use of linear constraint improves significantly the robustness against channel order overestimation errors of the blind MMSE filter. Note that, as explained in Section 3.5, improved results are obtained with the proposed algorithm using quadratic constraint, when the sample size increases. This is observed by comparing the

results of the quadratic constraint method of Figure 6(b) with those of Figure 6(a).

### 6.3. Robustness against small values of $\mathbf{H}(0)$

In general, the main weakness of a zero-delay equalizer is its sensitivity to small values of the first channel coefficient  $\mathbf{H}(0)$ . In Figure 7, we illustrate the robustness of the proposed algorithm, when  $\mathbf{H}(0)$  takes small value. More precisely, we plot the MSE versus the variance of  $\mathbf{H}(0)$ :  $\sigma_{\mathbf{H}(0)}^2 \stackrel{\text{def}}{=} E(\|\mathbf{H}(0)\|^2)$ , for  $q = 1, p = 3, K = 500$ , and SNR = 15 in Figure 7(a) (resp., SNR = 30 in Figure 7(b)). It is clear that for low and moderate SNRs a minimum variance of  $\mathbf{H}(0)$  is needed (in the plot  $\sigma_{\mathbf{H}(0)}^2 \geq 0.2$  is required) for the algorithm to provide satisfactory results. However, this threshold value can be quite small for high SNR as shown by Figure 7(b).

### 6.4. Influence of the number of sensors

Figure 8 represents the evolution of the MSE versus the number of sensors for  $q = 1, K = 500$ , and SNR = 5 in Figure 8(a) (resp., SNR = 15 in Figure 8(b)). One can observe that for low SNR, the algorithm requires a minimum degree of freedom in terms of number of sensors (typically  $p - q$  should be larger than 2 or 3), while at moderate and large SNRs,  $p$  can be as small as  $q + 1$ . Eventhough not included here, due to space limitation, similar results have been observed in the MIMO case.

### 6.5. Discussion

These results highlight one of the main advantages of our method which is the improved robustness against channel order overestimation errors. Also, even when the channel order is known, the proposed algorithm outperforms the algorithms in [10, 11] for low SNR. Another strong advantages of the proposed algorithm is its low computational cost and higher convergence rate (in its adaptive version) compared to those in [10–12]. However, the methods in [10–12] have the advantages of allowing direct estimation (in one step) of the nonzero-delay equalizer which is important in certain limit cases, where the zero-delay equalizer fails to provide satisfactory performance (see Figure 7).

## 7. CONCLUSION

In this contribution, we have presented an original method for blind equalization of multichannel FIR filters. Batch and fast adaptive implementation algorithms are developed. A two-step version using the a priori knowledge of the source signal finite alphabet has been proposed in order to control the equalization delay and improve the estimation performance. An asymptotic performance analysis of the proposed algorithm has been carried out in the single input case (SIMO case). Robustness against channel

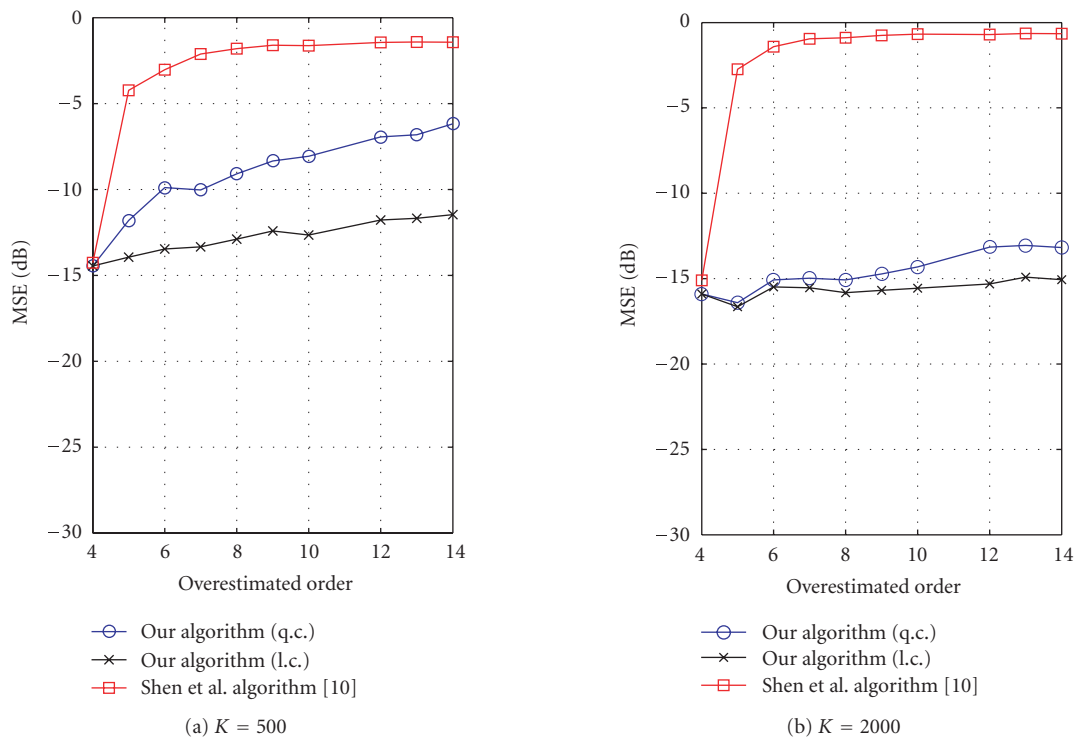


FIGURE 6: Robustness comparison (against channel order overestimation errors). The exact order is  $L = 4$ .

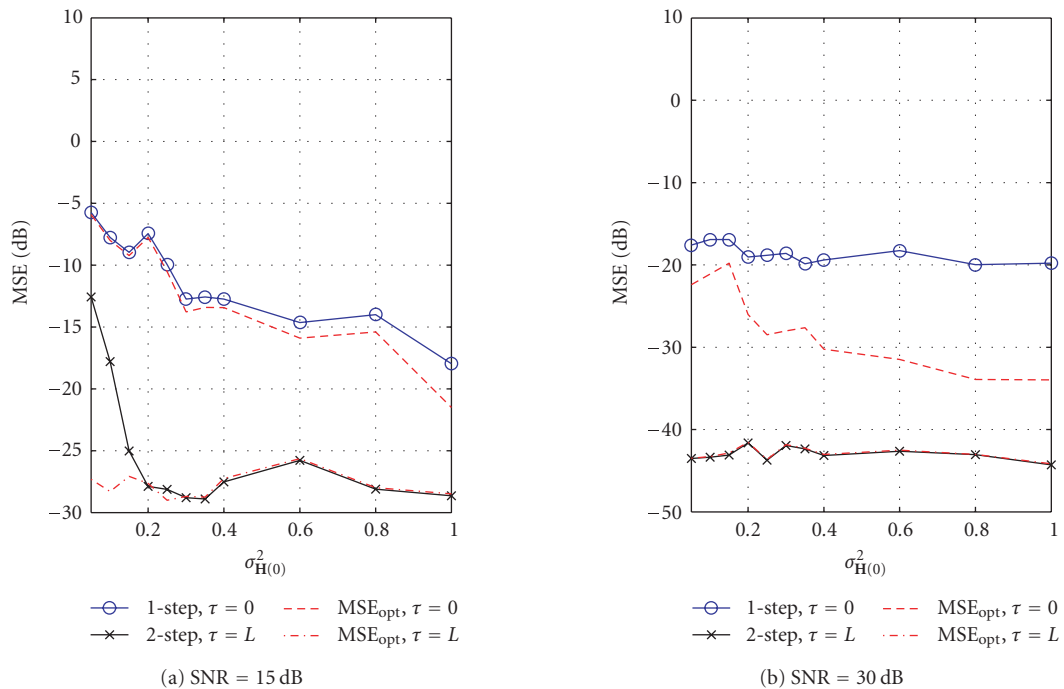


FIGURE 7: Robustness against small values of  $\mathbf{H}(0)$ .

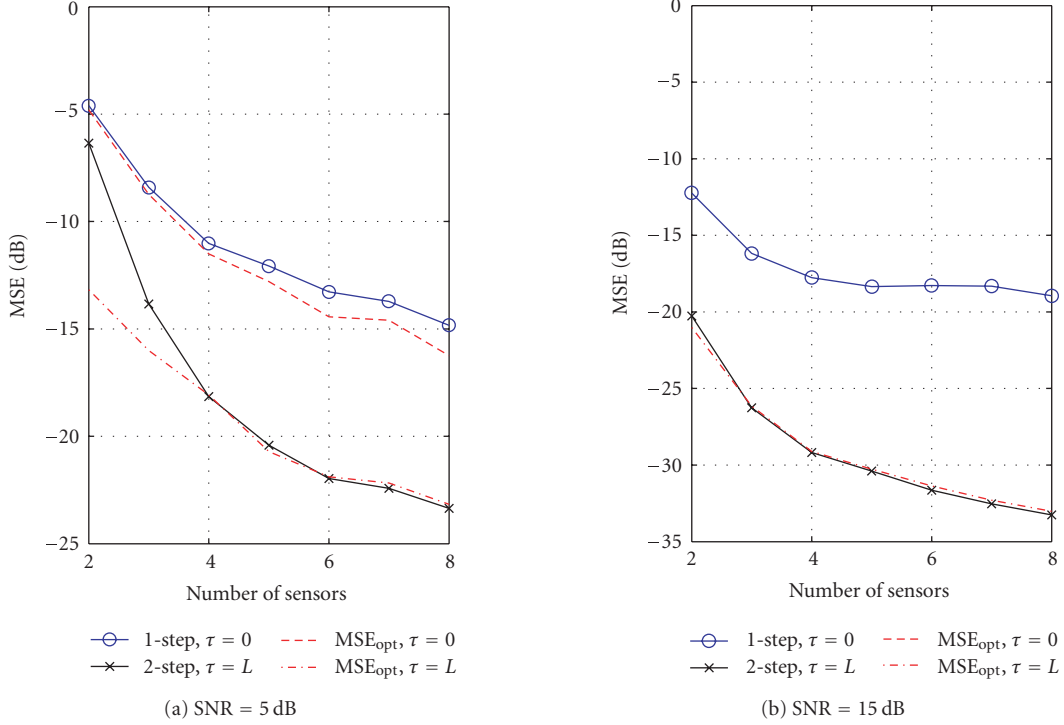


FIGURE 8: Mean square error versus the number of sensors.

order overestimation errors and performance of the proposed equalization method are studied.

## APPENDICES

### A. $O(n)$ COMPUTATION OF $\mathbf{x}'(t) = \mathbf{C}(t-1)\mathbf{x}_N(t)$

A technique to reduce the computation of the vector  $\mathbf{x}'(t) = \mathbf{C}(t-1)\mathbf{x}_N(t)$  from  $O(n^2)$  to  $O(n)$  operations, is presented herein. This technique was proposed in [20] for time series data, and here we generalize it for multivariate data.

We begin by defining the  $(n+p)$ -dimensional vector

$$\underline{\mathbf{g}}(t) \stackrel{\text{def}}{=} \underline{\mathbf{C}}(t-1)\mathbf{x}_{N+1}(t), \quad (\text{A.1})$$

where  $\underline{\mathbf{C}}(t)$  is the extended covariance matrix given by

$$\underline{\mathbf{C}}(t) = \sum_{k=1}^t \beta^{t-k} \mathbf{x}_{N+1}(k)\mathbf{x}_{N+1}^H(k). \quad (\text{A.2})$$

Taking into account the fact that

$$\mathbf{x}_{N+1}(t) = [\mathbf{x}^T(t) \quad \mathbf{x}_N^T(t-1)]^T = [\mathbf{x}_N^T(t) \quad \mathbf{x}^T(t-N)]^T, \quad (\text{A.3})$$

one can write

$$\underline{\mathbf{C}}(t) = \begin{bmatrix} \mathbf{C}^1(t) & \mathbf{C}^2(t) \\ [\mathbf{C}^2(t)]^H & \mathbf{C}(t-1) \end{bmatrix} = \begin{bmatrix} \mathbf{C}(t) & \mathbf{C}^3(t) \\ [\mathbf{C}^3(t)]^H & \mathbf{C}^1(t-N) \end{bmatrix}, \quad (\text{A.4})$$

where

$$\mathbf{C}^1(t) = \sum_{k=1}^t \beta^{t-k} \mathbf{x}(k)\mathbf{x}^H(k),$$

$$\mathbf{C}^2(t) = \sum_{k=1}^t \beta^{t-k} \mathbf{x}(k)\mathbf{x}_N^H(k-1), \quad (\text{A.5})$$

$$\mathbf{C}^3(t) = \sum_{k=1}^t \beta^{t-k} \mathbf{x}_N(k)\mathbf{x}^H(k-N).$$

Using (A.3) and (A.4), we have

$$\underline{\mathbf{g}}(t) = \begin{bmatrix} \mathbf{C}^1(t-1)\mathbf{x}(t) + \mathbf{C}^2(t-1)\mathbf{x}_N(t-1) \\ [\mathbf{C}^2(t-1)]^H\mathbf{x}(t) + \mathbf{x}'(t-1) \end{bmatrix} \quad (\text{A.6})$$

$$= \begin{bmatrix} \mathbf{x}'(t) + \mathbf{C}^3(t-1)\mathbf{x}(t-N) \\ [\mathbf{C}^3(t-1)]^H\mathbf{x}_N(t) + \mathbf{C}^1(t-N-1)\mathbf{x}(t-N) \end{bmatrix}. \quad (\text{A.7})$$

Equation (A.6) is used to compute  $\underline{\mathbf{g}}(t)$  and, from (A.7),  $\mathbf{x}'(t)$  is updated as follows:

$$\mathbf{x}'(t) = \underline{\mathbf{g}}(t)_{(1:n)} - \mathbf{C}^3(t-1)\mathbf{x}(t-N). \quad (\text{A.8})$$

The only other quantities that need updating are the matrices

$$\begin{aligned}
\mathbf{g}(t)_{(1:p)} &= \mathbf{C}^1(t-1)\mathbf{x}(t) + \mathbf{C}^2(t-1)\mathbf{x}_N(t-1) \\
\mathbf{g}(t)_{(p+1:n+p)} &= [\mathbf{C}^2(t-1)]^H \mathbf{x}(t) + \mathbf{x}'(t-1) \\
\mathbf{x}'(t) &= \mathbf{g}(t)_{(1:n)} - \mathbf{C}^3(t)\mathbf{x}(t-N) \\
\mathbf{C}^1(t) &= \beta \mathbf{C}^1(t-1) + \mathbf{x}(t)\mathbf{x}^H(t) \\
\mathbf{C}^2(t) &= \beta \mathbf{C}^2(t-1) + \mathbf{x}(t)\mathbf{x}_N^H(t-1) \\
\mathbf{C}^3(t) &= \beta \mathbf{C}^3(t-1) + \mathbf{x}_N(t)\mathbf{x}^H(t-N)
\end{aligned}$$

ALGORITHM 8: Algorithm for updating  $\mathbf{x}'(t) = \mathbf{C}(t-1)\mathbf{x}_N(t)$  in  $O(n)$  operations.

in (A.4), which can be efficiently computed as

$$\begin{aligned}
\mathbf{C}^1(t) &= \beta \mathbf{C}^1(t-1) + \mathbf{x}(t)\mathbf{x}^H(t), \\
\mathbf{C}^2(t) &= \beta \mathbf{C}^2(t-1) + \mathbf{x}(t)\mathbf{x}_N^H(t-1), \\
\mathbf{C}^3(t) &= \beta \mathbf{C}^3(t-1) + \mathbf{x}_N(t)\mathbf{x}^H(t-N).
\end{aligned} \tag{A.9}$$

The algorithm listing is found in Algorithm 8.

## B. PROOF OF LEMMA 1

Matrix  $\Sigma_c$  is defined by

$$\Sigma_c = [\Sigma_{c,k,l}]_{1 \leq k,l \leq n^2} \stackrel{\text{def}}{=} \lim_{K \rightarrow +\infty} KE((\hat{c} - \mathbf{c})(\hat{c} - \mathbf{c})^H), \tag{B.1}$$

it follows that

$$\begin{aligned}
\Sigma_{c,k,l} &= \lim_{K \rightarrow +\infty} KE((\hat{c}_k - c_k)(\hat{c}_l - c_l)^*) \\
&= \lim_{K \rightarrow +\infty} KE((\hat{C}_{a,b} - C_{a,b})(\hat{C}_{c,d} - C_{c,d})^*),
\end{aligned} \tag{B.2}$$

where  $c_i$  (resp.,  $C_{\alpha,\beta}$ ) and  $\hat{c}_i$  (resp.,  $\hat{C}_{\alpha,\beta}$ ) denote the  $i$ th (resp., the  $(\alpha, \beta)$ th) entry of  $\mathbf{c}$  (resp.,  $\mathbf{C}$ ) and  $\hat{\mathbf{c}}$  (resp.,  $\hat{\mathbf{C}}$ ), respectively, which are given by

$$\begin{aligned}
c_i &= C_{\alpha,\beta} = E(x_{N,\alpha}(t)x_{N,\beta}^*(t)), \\
\hat{c}_i &= \hat{C}_{\alpha,\beta} = \frac{1}{K} \sum_{t=0}^{K-1} x_{N,\alpha}(t)x_{N,\beta}^*(t), \\
\alpha &= \alpha' + n\delta(\alpha'), \quad \beta = \beta' + 1 - \delta(\alpha'), \quad 1 \leq \alpha, \beta \leq n,
\end{aligned} \tag{B.3}$$

where  $x_{N,i}(t)$  is the  $i$ th entry of vector  $\mathbf{x}_N(t)$ ,  $\alpha'$  and  $\beta'$  denote, respectively, the rest and the quotient of the Euclidian division of  $i$  by  $n$ , and  $\delta$  is the Kronecker symbol.  $(a, b)$  and  $(c, d)$  are obtained in a similar way for  $k$  and  $l$ , respectively.

Then, after some calculation (see [15] for more details) and using the relationship between cumulants and moments, we obtain the following expression of  $\Sigma_{c,k,l}$ :

$$\Sigma_{c,k,l} = \kappa_{k,l} + \sum_{\tau \in \mathbb{Z}} C_{\tau,a,c} C_{-\tau,d,b}, \tag{B.4}$$

where

$$\kappa_{k,l} \stackrel{\text{def}}{=} \sum_{\tau \in \mathbb{Z}} \text{cum}(x_{N,a}(\tau), x_{N,b}^*(\tau), x_{N,d}(0), x_{N,c}^*(0)), \tag{B.5}$$

taking into account the particular structure of the data model (2) and applying some standard properties of cumulants, the fourth-order cumulant in (B.5) is then expressed as

$$\begin{aligned}
&\text{cum}(x_{N,a}(\tau), x_{N,b}^*(\tau), x_{N,d}(0), x_{N,c}^*(0)) \\
&= \kappa \sum_{i \in \mathbb{Z}} \mathbf{H}_N(a, i + \tau) \mathbf{H}_N^*(b, i + \tau) \mathbf{H}_N(d, i) \mathbf{H}_N^*(c, i),
\end{aligned} \tag{B.6}$$

where  $\kappa \stackrel{\text{def}}{=} \text{cum}(s(t), s^*(t), s(t), s^*(t))$  is the kurtosis of the input signal  $s(t)$ , and  $\mathbf{H}_N(i, j)$  is the  $(i, j)$ th entry of  $\mathbf{H}_N$ . Plugging this expression into (B.5) yields

$$\begin{aligned}
\kappa_{k,l} &= \kappa \sum_j \mathbf{H}_N(a, j) \mathbf{H}_N^*(b, j) \sum_i \mathbf{H}_N(d, i) \mathbf{H}_N^*(c, i) \\
&= \kappa (C_{a,b} - \sigma_b^2 \delta(a-b)) (C_{c,d} - \sigma_c^2 \delta(c-d))^*.
\end{aligned} \tag{B.7}$$

Finally, it is easy to verify from (B.4) and (B.7) that  $\Sigma_c$  is expressed by (56).

## C. PROOF OF THEOREM 3

Before proceeding, we first need to recall some basic properties of column vectorizing operator and, matrices and vectors differentiation (see [16] for more details). If  $\mathbf{A}$ ,  $\mathbf{B}$ , and  $\mathbf{C}$  are given matrices, then  $\text{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A}) \text{vec}(\mathbf{B})$  and  $\delta \text{vec}(\mathbf{A}) = \text{vec}(\delta \mathbf{A})$  where  $\delta$  denotes the differentiation operator. If  $\mathbf{A}$  is an  $\alpha \times \beta$  matrix, then  $\text{vec}(\mathbf{A}^T) = \mathbf{U}_{\alpha,\beta} \text{vec}(\mathbf{A})$  where  $\mathbf{U}_{\alpha,\beta}$  is the permutation matrix defined in Theorem 3. Let  $\lambda$  be an eigenvalue of an  $\alpha \times \alpha$  matrix  $\mathbf{A}$ ,  $\mathbf{w}$  the eigenvector corresponding to  $\lambda$ , the differential  $\delta \mathbf{w}$  of  $\mathbf{w}$  is given by  $\delta \mathbf{w} = (\lambda \mathbf{I}_\alpha - \mathbf{A})^\# \delta \mathbf{A} \mathbf{w} = [\mathbf{w}^T \otimes (\lambda \mathbf{I}_\alpha - \mathbf{A})^\#] \delta \text{vec}(\mathbf{A})$ . If  $\mathbf{A}$  is invertible, then  $\delta \mathbf{A}^{-1} = -\mathbf{A}^{-1} \delta \mathbf{A} \mathbf{A}^{-1}$ .

Let  $\hat{\mathbf{v}}$  be an estimate of the blind MMSE equalizer vector given from  $K$ -sample observations, then  $\hat{\mathbf{v}}_0$  is given according to  $\hat{\mathbf{v}}_0 = \hat{r} \hat{\mathbf{v}}$ , where  $\hat{r} = \hat{\mathbf{v}}^H \mathbf{v}_0 / \|\hat{\mathbf{v}}\|^2$ . Replacing  $\hat{\mathbf{v}}_0$ ,  $\hat{\mathbf{v}}$ , and  $\hat{r}$  by  $\mathbf{v}_0 + \delta \mathbf{v}_0$ ,  $\mathbf{v} + \delta \mathbf{v}$ , and  $r + \delta r$ , respectively, we obtain

$$\delta \mathbf{v}_0 = r \left( \mathbf{I}_n - \frac{\mathbf{v} \mathbf{v}^H}{\|\mathbf{v}\|^2} \right) \delta \mathbf{v}. \tag{C.1}$$

As  $\mathbf{v} = \mathbf{W} \tilde{\mathbf{v}}$ , it follows that

$$\delta \mathbf{v} = (\tilde{\mathbf{v}}^T \otimes \mathbf{I}_n) \delta \text{vec}(\mathbf{W}) + \mathbf{W} \delta \tilde{\mathbf{v}}. \tag{C.2}$$

*Quadratic constraint case*

In this case,  $\tilde{\mathbf{v}}$  is the least eigenvector (which correspond to zero-eigenvalue) of matrix  $\mathbf{Q}$  given by (12) and (18). The differentiation of  $\tilde{\mathbf{v}}$  gives

$$\delta \tilde{\mathbf{v}} = -(\tilde{\mathbf{v}}^T \otimes \mathbf{Q}^\#) \delta \text{vec}(\mathbf{Q}) = -\mathbf{M}_2 \delta \text{vec}(\mathbf{Q}). \tag{C.3}$$

From (12) and (18), matrix  $\mathbf{Q}$  is written as

$$\mathbf{Q} = \mathbf{W}^H \mathbf{C} \mathbf{J}_{p,n-p,0} \mathbf{J}_{p,n-p,0}^T \mathbf{C} \mathbf{W}. \quad (\text{C.4})$$

The differentiation of  $\mathbf{Q}$  gives

$$\begin{aligned} \delta \text{vec}(\mathbf{Q}) &= [(\mathbf{C} \mathbf{J}_{p,n-p,0} \mathbf{T})^T \otimes \mathbf{I}_d] \delta \text{vec}(\mathbf{W}^H) \\ &+ [\mathbf{I}_d \otimes (\mathbf{T}^H \mathbf{J}_{p,n-p,0}^T \mathbf{C})] \delta \text{vec}(\mathbf{W}) \\ &+ [(\mathbf{J}_{p,n-p,0} \mathbf{T})^T \otimes \mathbf{W}^H + \mathbf{W}^T \otimes (\mathbf{T}^H \mathbf{J}_{p,n-p,0}^T)] \delta \mathbf{c}. \end{aligned} \quad (\text{C.5})$$

As the columns of  $\mathbf{W}$  correspond to the  $d$  dominant eigenvectors of  $\mathbf{C}$ , thus  $\delta \mathbf{W} = [\delta \mathbf{W}(:,1) \cdots \delta \mathbf{W}(:,d)]$ , where  $\delta \mathbf{W}(:,i) = (\mathbf{W}^T(:,i) \otimes (\lambda \mathbf{I}_n - \mathbf{C})^\#) \delta \mathbf{c}$ ,  $i = 1, \dots, d$ . This implies

$$\delta \text{vec}(\mathbf{W}) = \mathbf{\Gamma} \delta \mathbf{c}, \quad (\text{C.6})$$

where  $\mathbf{\Gamma}$  is defined in Theorem 3. It follows that  $\delta \text{vec}(\mathbf{W}^H) = \mathbf{U}_{n,d} \delta \text{vec}(\mathbf{W}^*) = \mathbf{U}_{n,d} \mathbf{\Gamma}^* \delta \text{vec}(\mathbf{C}^*)$ , as  $\mathbf{C}^* = \mathbf{C}^T$ , we obtain

$$\delta \text{vec}(\mathbf{W}^H) = \mathbf{U}_{n,d} \mathbf{\Gamma}^* \mathbf{U}_{n,n} \delta \mathbf{c}. \quad (\text{C.7})$$

Plugging (C.6) and (C.7) in (C.5) yields

$$\delta \text{vec}(\mathbf{Q}) = \mathbf{M}_1 \delta \mathbf{c}, \quad (\text{C.8})$$

where  $\mathbf{M}_1$  is given in Theorem 3. Finally, from (C.1), (C.2), (C.3), (C.6), and (C.8), we obtain

$$\delta \mathbf{v}_0 = \mathbf{M} \delta \mathbf{c}. \quad (\text{C.9})$$

#### Linear constraint case

In this case, we use the expression of  $\tilde{\mathbf{v}}$  given by (25)

$$\tilde{\mathbf{v}} = \mathbf{J}_1 \bar{\mathbf{v}} + \mathbf{J}_{0,1,d-1}. \quad (\text{C.10})$$

From (23), (24), and (41), vector  $\bar{\mathbf{v}}$  is expressed as

$$\bar{\mathbf{v}} = -(\mathbf{J}_1^T \mathbf{Q} \mathbf{J}_1)^{-1} \mathbf{J}_1^T \mathbf{Q} \mathbf{J}_{0,1,d-1}. \quad (\text{C.11})$$

Differentiating  $\tilde{\mathbf{v}}$  yields

$$\begin{aligned} \delta \tilde{\mathbf{v}} &= \mathbf{J}_1 \delta \bar{\mathbf{v}} = \mathbf{J}_1 (\mathbf{J}_1^T \mathbf{Q} \mathbf{J}_1)^{-1} \mathbf{J}_1^T \delta \mathbf{Q} \mathbf{J}_1 (\mathbf{J}_1^T \mathbf{Q} \mathbf{J}_1)^{-1} \mathbf{J}_1^T \mathbf{Q} \mathbf{J}_{0,1,d-1} \\ &- \mathbf{J}_1 (\mathbf{J}_1^T \mathbf{Q} \mathbf{J}_1)^{-1} \mathbf{J}_1^T \delta \mathbf{Q} \mathbf{J}_{0,1,d-1} \\ &= -(\tilde{\mathbf{v}}^T \otimes \mathbf{Q}') \delta \text{vec}(\mathbf{Q}) = -\mathbf{M}_2 \mathbf{M}_1 \delta \mathbf{c}, \end{aligned} \quad (\text{C.12})$$

where  $\mathbf{Q}'$  is given as in Theorem 3. From (C.1), (C.2), (C.6), and (C.12), we obtain finally

$$\delta \mathbf{v}_0 = \mathbf{M} \delta \mathbf{c}. \quad (\text{C.13})$$

Using (C.9) (resp., (C.13)) in the quadratic constraint case (resp., in the linear constraint case) and Theorem 2 result leads to the expression of  $\Sigma_v$  given in Theorem 3.

#### ACKNOWLEDGMENT

Part of this work has been published in conferences [26, 27].

#### REFERENCES

- [1] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, Englewood Cliffs, NJ, USA, 3rd edition, 1996.
- [2] Y. Sato, "A method of self-recovering equalization for multi-level amplitude-modulation," *IEEE Transactions on Communications*, vol. 23, no. 6, pp. 679–682, 1975.
- [3] D. N. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Transactions on Communications*, vol. 28, no. 11, pp. 1867–1875, 1980.
- [4] L. Tong, G. Xu, and T. Kailath, "A new approach to blind identification and equalization of multipaths channels," in *Proceedings of 25th Asilomar Conference on Circuits, Systems and Computers*, pp. 856–860, Pacific Grove, Calif, USA, November 1991.
- [5] K. Abed-Meraim, W. Qiu, and Y. Hua, "Blind system identification," *Proceedings of the IEEE*, vol. 85, no. 8, pp. 1310–1322, 1997.
- [6] E. Moulines, P. Duhamel, J.-F. Cardoso, and S. Mayrargue, "Subspace methods for the blind identification of multichannel FIR filters," *IEEE Transactions on Signal Processing*, vol. 43, no. 2, pp. 516–525, 1995.
- [7] A. P. Liavas, P. A. Regalia, and J.-P. Delmas, "Blind channel approximation: effective channel order determination," *IEEE Transactions on Signal Processing*, vol. 47, no. 12, pp. 3336–3344, 1999.
- [8] W. H. Gerstacker and D. P. Taylor, "Blind channel order estimation based on second-order statistics," *IEEE Signal Processing Letters*, vol. 10, no. 2, pp. 39–42, 2003.
- [9] J. Xavier and V. Barroso, "A channel order independent method for blind equalization of MIMO systems," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '99)*, vol. 5, pp. 2897–2900, Phoenix, Ariz, USA, March 1999.
- [10] J. Shen and Z. Ding, "Direct blind MMSE channel equalization based on second-order statistics," *IEEE Transactions on Signal Processing*, vol. 48, no. 4, pp. 1015–1022, 2000.
- [11] M. Sheng and H. Fan, "Blind MMSE equalization: a new direct method," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '00)*, vol. 5, pp. 2457–2460, Istanbul, Turkey, June 2000.
- [12] X. Li and H. Fan, "Direct estimation of blind zero-forcing equalizers based on second-order statistics," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2211–2218, 2000.
- [13] H. Gazzah, P. A. Regalia, J.-P. Delmas, and K. Abed-Meraim, "A blind multichannel identification algorithm robust to order overestimation," *IEEE Transactions on Signal Processing*, vol. 50, no. 6, pp. 1449–1458, 2002.
- [14] F. D. Neeser and J. L. Massey, "Proper complex random processes with applications to information theory," *IEEE Transactions on Information Theory*, vol. 39, no. 4, pp. 1293–1303, 1993.



- [15] K. Abed-Meraim, P. Loubaton, and E. Moulines, "A subspace algorithm for certain blind identification problems," *IEEE Transactions on Information Theory*, vol. 43, no. 2, pp. 499–511, 1997.
- [16] J. W. Brewer, "Kronecker products and matrix calculus in system theory," *IEEE Transactions on Circuits and Systems*, vol. 25, no. 9, pp. 772–781, 1978.
- [17] A.-J. van der Veen and A. Paulraj, "An analytical constant modulus algorithm," *IEEE Transactions on Signal Processing*, vol. 44, no. 5, pp. 1136–1155, 1996.
- [18] A. Belouchrani and K. Abed-Meraim, "Constant modulus blind source separation technique: a new approach," in *Proceedings of the International Symposium on Signal Processing and Its Applications (ISSPA '96)*, vol. 1, pp. 232–235, Gold Coast, Australia, August 1996.
- [19] B. Yang, "Projection approximation subspace tracking," *IEEE Transactions on Signal Processing*, vol. 43, no. 1, pp. 95–107, 1995.
- [20] C. E. Davila, "Efficient, high performance, subspace tracking for time-domain data," *IEEE Transactions on Signal Processing*, vol. 48, no. 12, pp. 3307–3315, 2000.
- [21] R. Badeau, B. David, and G. Richard, "Yet another subspace tracker," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '05)*, vol. 4, pp. 329–332, Philadelphia, Pa, USA, March 2005.
- [22] K. Abed-Meraim, A. Chkeif, and Y. Hua, "Fast orthogonal PAST algorithm," *IEEE Signal Processing Letters*, vol. 7, no. 3, pp. 60–62, 2000.
- [23] A. Chkeif, K. Abed-Meraim, G. Kawas-Kaleh, and Y. Hua, "Spatio-temporal blind adaptive multiuser detection," *IEEE Transactions on Communications*, vol. 48, no. 5, pp. 729–732, 2000.
- [24] J.-F. Cardoso and E. Moulines, "Asymptotic performance analysis of direction-finding algorithms based on fourth-order cumulants," *IEEE Transactions on Signal Processing*, vol. 43, no. 1, pp. 214–224, 1995.
- [25] M. K. Tsatsanis and G. B. Giannakis, "Modelling and equalization of rapidly fading channels," *International Journal of Adaptive Control and Signal Processing*, vol. 10, no. 2-3, pp. 159–176, 1996.
- [26] I. Kacha, K. Abed-Meraim, and A. Belouchrani, "A fast adaptive blind equalization algorithm robust to channel order overestimation errors," in *Proceedings of the 3rd IEEE Sensor Array and Multichannel Signal Processing Workshop*, pp. 148–152, Barcelona, Spain, July 2004.
- [27] I. Kacha, K. Abed-Meraim, and A. Belouchrani, "A new blind adaptive MMSE equalizer for MIMO systems," in *Proceedings of the 16th Annual IEEE International Symposium on Personal Indoor and Mobile Radio Communications*, Berlin, Germany, September 2005.

**Ibrahim Kacha** received the State Engineering and M.S. degrees both in electrical engineering from the École Nationale Polytechnique (ENP), Algiers, Algeria, in 1990 and 1993, respectively. He was a Lecturer at the Department of Electrical Engineering of ENP from 1993 to 2005. He is currently a Ph.D. student in the area of signal processing at the Department of Signal and Image Processing, École Nationale Supérieure des Télécommunications (ENST), Paris, France. His research interests



are statistical signal processing and blind system identification and equalization for digital communications.

**Karim Abed-Meraim** was born in 1967. He received the State Engineering degree from the École Polytechnique, Paris, France, in 1990, as well as from École Nationale Supérieure des Télécommunications (ENST), Paris, France, in 1992, the M.S. degree from Paris XI University, Orsay, France, in 1992, and the Ph.D. degree from École Nationale Supérieure des Télécommunications (ENST), Paris, France, in 1995 (in the field of signal processing and communications). From 1995 to 1998, he was a Research Staff Member at the Electrical Engineering Department of the University of Melbourne where he worked on several research projects related to blind system identification for wireless communications, blind source separation, and array processing for communications, respectively. He is currently an Associate Professor (since 1998) at the Signal and Image Processing Department of ENST. His research interests are in signal processing for communications and include system identification, multiuser detection, space-time coding, adaptive filtering and tracking, array processing, and performance analysis. He is an IEEE Senior Member and a past Associate Editor for the IEEE Transactions on Signal Processing.



**Adel Belouchrani** received the State Engineering degree in 1991 from École Nationale Polytechnique (ENP), Algiers, Algeria, the M.S. degree in signal processing from the Institut National Polytechnique de Grenoble (INPG), France, in 1992, and the Ph.D. degree in signal and image processing from Télécom Paris (ENST), France, in 1995. He was a Visiting Scholar at the Electrical Engineering and Computer Sciences Department, University of California, Berkeley, from 1995 to 1996. He was with the Department of Electrical and Computer Engineering, Villanova University, Villanova, PA, as a Research Associate from 1996 to 1997. He is currently and since 1998 with the Electrical Engineering Department of ENP as a Full Professor. His research interests are in statistical signal processing and (blind) array signal processing with applications in biomedical and communications, time-frequency analysis, time-frequency array signal processing, and wireless and spread spectrum communications.

