A New Robust Adaptive Algorithm for Second Order Blind Source Separation

Zineb Bekhtaoui, Karim Abed-Meraim, Abdelkrim Meche, and Messaoud Thameri

Abstract—This paper deals with the adaptive blind source separation problem in presence of impulsive noise. New algorithms extending the well known SOBI method from batch to adaptive scheme are introduced. At first, the standard Gaussian noise case is considered, leading to our first algorithm referred to as Adaptive SOBI (A-SOBI). Later on, a robust version of A-SOBI, referred to as RA-SOBI is derived to handle the impulsive noise case. RA-SOBI relies on robust subspace tracking for the whitening stage together with robust correlation estimation for the separation stage. All proposed algorithms are of relatively low complexity and allow to achieve good separation quality as illustrated by our simulation results.

***Keywords*—Blind source separation, Adaptive algorithm, Impulsive noise, Robustness, Subspace tracking**

I. Introduction

Blind source separation (BSS) consists of the extraction of source signals from their observed mixtures without prior knowledge of the mixing matrix or its inputs. BSS is widely used in many signal processing applications, and a plethora of works have been devoted to develop solutions in different contexts and under different mixing models, e.g., [1–7]. In particular, second order statistics based methods are highly regarded due to their low computation load and efficiency to separate temporally coherent (colored) sources. These features make them suitable for adaptive scheme when dealing with streaming data. Several algorithms have been already dedicated to such an adaptive scheme including [4, 5, 8–13]. Most of existing algorithms consider noise as being Gaussian or negligible. However, in many applications the measurements are affected by impulsive noise or outliers, e.g., [21–23], in which context standard methods fail to achieve the BSS. To deal with impulsive noise, some authors have proposed robust batch BSS algorithms, e.g., [7, 14–18].

In this work, we propose to deal with both streaming data (i.e. adaptive scheme) and impulsive noise. Hence, a new approach that ensures both robustness and adaptivity is introduced based on second order decorrelation approach. At first, we consider only the streaming data case and introduce the adaptive Second Order Blind Identification (SOBI) algorithm [12] for the Gaussian noise case. In this algorithm, referred to as Adaptive SOBI (A-SOBI), the source separation is performed in two steps: whitening and joint diagonalization. The whitening is achieved using an adaptive Principal Components Analysis (PCA) algorithm followed by the joint diagonalization conducted on several non zero lag correlation matrices. Then, our Robust Adaptive SOBI (RA-SOBI) algorithm is derived via the use of robust PCA tracking algorithm [19] for the whitening step by minimizing a weighted least criterion. Moreover, to improve furthermore the robustness of the algorithm, we propose here to estimate the non zero lag correlation matrices, considered in the joint diagonalization step, via robust estimation techniques [24].

This paper is organized as follows: Objectives and problem formulation are stated in section II.. The A-SOBI algorithm is given in section III.. Section IV. introduces RA-SOBI algorithm, while section V. is dedicated to simulation results providing the evaluation of our algorithms effectiveness. Finally, our concluding remarks are given in section VI..

Notations: The conjugate transpose, the transpose, the conjugate, the inverse, and the trace operations are represented by $()^H$, $()^T$, $()^*$, $()^{-1}$, and $tr()$ respectively. a denotes a scalar, $\mathbf{a}$ denotes a vector, $\mathbf{A}$ denotes a matrix, $\|\cdot\|^2$ denotes the Euclidean norm, $\mathbf{I}$ represents the identity matrix, and $\mathbf{A}_{ij}$ represents the (i, j)-th element of $\mathbf{A}$.

II. Problem Statement

Consider a streaming data of multivariate $n \times 1$ dimensional vectors $\mathbf{x}(t)$ corresponding to the noisy mixtures of $p < n$ sources, i.e. $\mathbf{s}(t) = [s_1(t), \cdots, s_p(t)]^T$, according to

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t) \qquad (1)$$

where $\mathbf{A}$ is the unknown $n \times p$ mixing matrix assumed to be of full column rank and $\mathbf{n}(t)$ is the additive noise vector assumed to be of zero mean and spatially white with covariance $E(\mathbf{n}(t)\mathbf{n}^H(t)) = \sigma^2\mathbf{I}$. The zero mean source signals are assumed to be temporally coherent but mutually decorrelated, i.e. $E(s_i(t+\tau)s_j^*(t)) = \delta_{ij}\rho_i(\tau)$ where $\rho_i(\tau)$ represents the correlation function of the i-th source signal and δ_{ij} is the Kronecker index.

Our objective in this work is to exploit the statistical (mutual) decorrelation information of the sources to retrieve the latter

Manuscript received April 22, 2022; revised June 13, 2022.

Z. Bekhtaou and A. Meche are with Laboratoire Signaux et Images, University of Sciences and Technology of Oran Mohamed Boudiaf (USTOMB), B.P 1505, El M'Naouar - Bir el Djir- Oran, Algeria. (e-mails: zineb.bekhtaoui@univ-usto.dz, abdelkrim.meche@univ-usto.dz)

K. Abed-Meraim, IUF Member, is affiliated to PRISME Lab., University of Orleans, France. (e-mail: karim.abed-meraim@univ-orleans.fr)

M. Thameri is with Ecole Supérieure Ali Chabati, Réghaia, Alger, Algérie. (e-mail: m_thameri@hotmail.com)

Digital Object Identifier (DOI): 10.53907/enpesj.v2i1.86

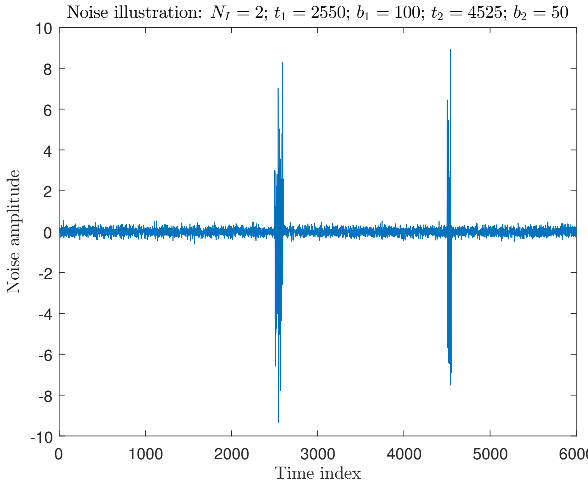

Fig. 1: Illustration of the noise $\mathbf{n}(t)$. The burst noise here appears in two occasions: at time 2500 and 4500 with duration 100 and 50, respectively.

from the observed data $\mathbf{x}(t)$ using only its second order statistics. Hence, this blind source separation problem consists of finding a separation matrix $\mathbf{B}$ so that (2) represents an estimation of the source vectors $\hat{\mathbf{s}}(t)$ up to permutation and scaling factors[2].

$$\hat{\mathbf{s}}(t) = \mathbf{B}\mathbf{x}(t) \tag{2}$$

As mentioned earlier, we consider here an adaptive scheme, where the model in (1) can be non-stationary but 'slowly' varying, and address the case where observed data is corrupted by impulsive noise. After considering first the adaptive scheme in a Gaussian noise case, we extend our algorithm to the burst noise[3] context modeled by additional centered Gaussian white noise with larger amplitudes occurring in short periods of time. The burst noise $\mathbf{n}(t)$ can therefore be written as

$$\mathbf{n}(t) = \mathbf{n}_G(t) + \sum_{i=1}^{N_I} u\Big(\frac{t - t_i}{b_i}\Big)\mathbf{n}_I^i(t) \tag{3}$$

where $\mathbf{n}_G(t)$ and $\mathbf{n}_I^i(t)$ are white centered Gaussian noises of variances $\sigma_G^2 << \sigma_I^2$. $\mathbf{n}_I^i(t)$ is weighted by $u(\cdot)$ a rectangular function which is used to describe the short duration appearance of the burst noise as illustrated by Fig. 1. N_I refers to the number of impulsive events, t_i is the center of the i-th impulsive event, and b_i is its duration.

III. A-SOBI Algorithm: Gaussian Noise Case

A-SOBI algorithm consists of a separation approach by second order decorrelation, which proceeds in two steps: Whitening and diagonalization which are detailed next.

A. Whitening step

This step consists of projecting the observed vector $\mathbf{x}(t)$ onto the principal subspace spanned by the column vectors of the mixing matrix $\mathbf{A}$, with the purpose of transforming it into a unitary matrix. The used matrix in this transformation is called the whitening matrix $\mathbf{W}$.

[2] The latter are inherent indeterminacies of the BSS problem [3].
[3] This is the noise model used in our simulations, but other impulsive noise models can be considered as well [27].

In [3], it has been shown that the whitening matrix $\mathbf{W}$ can be computed from the eigen-decomposition of the covariance matrix of $\mathbf{x}(t)$, denoted $\mathbf{C}_x$, as follows:

$$\mathbf{C}_x = \mathbf{U}_s\mathbf{\Lambda}\mathbf{U}_s^H + \sigma^2\mathbf{I} \tag{4}$$
$$\mathbf{W} = \mathbf{\Lambda}^{-1/2}\mathbf{U}_s^H \tag{5}$$

where $\mathbf{U}_s$ and $\mathbf{\Lambda}$ are the matrices of the p principal eigenvectors and eigenvalues of the noise free covariance matrix. In an streaming data scheme, the exact eigenvectors and eigenvalues are replaced by their adaptive estimates using Givens-Orthogonal Projection Approximation Subspace Tracking (GOPAST) algorithm (see details in [12, 20]) according to:

$$\mathbf{W}(t) = \mathbf{\Lambda}(t)^{-1/2}\mathbf{U}_s^H(t) \tag{6}$$

The estimates of the principal components were obtained by using the linear cost GOPAST algorithm [20], i.e. its cost is $O(np)$ flops per iteration. This latter consists of estimating adaptively a basis $\mathbf{D}(t)$ of the principal subspace of the covariance matrix $\mathbf{C}_x$ with the GOPAST algorithm, starting with minimizing ($0 < \beta \leq 1$ being a chosen forgetting factor):

$$J\left(\mathbf{U}_s(t)\right) = \sum_{j=1}^{t} \beta^{t-j}\left\|\mathbf{x}(j) - \mathbf{D}(t)\mathbf{D}^H(t)\mathbf{x}(j)\right\|^2 \tag{7}$$

followed by an appropriate diagonalization using Givens rotations to get $\mathbf{U}_s(t)$ and $\mathbf{\Lambda}(t)$.

Note that since we are dealing with a slowly time varying system, β can be chosen to have a high value to insure faster convergence rate.

Remark: In GOPAST algorithm, $\mathbf{\Lambda}$ corresponds to the diagonal matrix of the principal eigenvalues of the covariance matrix $\mathbf{C}_x$. Implicitly, this means that the noise term (i.e. noise power σ^2) has been neglected in (4). In case, the latter cannot be considered as negligeable, one can estimate it as shown in [19] and replace in (6) $\mathbf{\Lambda}(t)$ by $\mathbf{\Lambda}(t) - \sigma^2(t)\mathbf{I}$.

B. Diagonalization step

After whitening, the mixing matrix is approximately reduced to a $p \times p$ unitary matrix denoted $\mathbf{U}^H(t)$ and hence the noiseless whitened signal can be written as $\tilde{\mathbf{x}}(t) = \mathbf{W}(t)\mathbf{x}(t) \approx \mathbf{U}^H(t)\mathbf{s}(t)$. To estimate the separation matrix $\mathbf{U}(t)$, A-SOBI uses a joint diagonalization of K correlation matrices corresponding to K chosen non-zero lags $\tau_1, \cdots, \tau_K$. The latter are adaptively estimated as:

$$\mathbf{R}_t(\tau_k) = \beta\mathbf{R}_{t-1}(\tau_k) + \hat{\mathbf{s}}(t)\hat{\mathbf{s}}^H(t - \tau_k) \tag{8}$$

Where $0 < \beta < 1$ is a forgetting factor and $\hat{\mathbf{s}}(t) = \mathbf{U}(t)\tilde{\mathbf{x}}(t)$. To achieve the joint diagonalization, it is first stated that the unitary separation matrix can be computed as a product of elementary Givens rotations:

$$\mathbf{U} = \prod_{l=1}^{p-1}\prod_{m=l+1}^{p} \mathbf{G}_{l,m}(\theta, \phi) \tag{9}$$

where $\mathbf{G}_{l,m}(\theta,\phi)$ is a $p \times p$ matrix equal to the identity except for its (l,l), (l,m), (m,l) and (m,m) entries. It is given by:

$$\mathbf{G}_{l,m}(\theta,\phi) = \begin{bmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & c & \cdots & -s^* & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & s & \cdots & c & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{bmatrix} \quad (10)$$

where $c = \cos(\theta)$ and $s = \sin(\theta)e^{-j\phi}$. Thanks to the mutual decorrelation of the source signals, the separation is achieved when the correlation matrices $\mathbf{R}(\tau_k)$ are diagonalized (see [3] for more details). To perform the latter diagonalization in an adaptive way, one multiplies at each time step $\mathbf{R}(\tau_k)$ at its left and right sides by the elementary Givens rotation $\mathbf{G}(t)$ according to:

$$\mathbf{R}'(\tau_k) = \mathbf{G}(t)\mathbf{R}(\tau_k)\mathbf{G}^H(t) \quad (11)$$

The rotation indices l and m are selected to be the corresponding indices of the entries of largest amplitude according to:

$$(l,m) = \arg \max_{\{(i,j)|i<j\}} \sum_{k=1}^{K} |\mathbf{R}_{ij}(\tau_k)| \quad (12)$$

Remark: Another way to select rotation indices (l,m) at time t would be to visit periodically along the iterations all entries of the correlation matrices, according to:

$$(l,m) = \begin{cases} (l', m'+1) & \text{if } m < p \\ (l'+1, l'+2) & \text{if } m = p \text{ and } l' < p-1 \\ (1,2) & \text{if } m = p \text{ and } l' = p-1 \end{cases} \quad (13)$$

(l', m') being the chosen indices at time $t-1$. Also, instead of one single rotation per time instant, one can use two rotations with indices chosen according to the previous two methods in order to increase the algorithm's convergence rate, at the cost of increased computational complexity.

Finally, the rotation angles are obtained by minimizing the sum of off diagonal elements of the K considered correlation matrices:

$$(\theta,\phi) = \arg\min_{\theta,\phi} \sum_{a\neq b} \sum_{k=1}^{K} |\mathbf{R}'_{ab}(\tau_k)|^2 \quad (14)$$

This is proven to be equivalent to solving

$$(\theta,\phi) = \arg\max_{\theta,\phi} \mathbf{v}^H \mathbf{F}\mathbf{F}^H \mathbf{v} \quad (15)$$

where

$$\mathbf{v} = \begin{bmatrix} \cos(2\theta) \\ \sin(2\theta)\cos(\phi) \\ \sin(2\theta)\sin(\phi) \end{bmatrix}$$

and

$$\mathbf{F} = \begin{bmatrix} \mathbf{R}_{ll}(\tau_1) - \mathbf{R}_{mm}(\tau_1) & \cdots & \mathbf{R}_{ll}(\tau_K) - \mathbf{R}_{mm}(\tau_K) \\ 2\Re(\mathbf{R}_{lm}(\tau_1)) & \cdots & 2\Re(\mathbf{R}_{lm}(\tau_K)) \\ 2\Im(\mathbf{R}_{lm}(\tau_1)) & \cdots & 2\Im(\mathbf{R}_{lm}(\tau_K)) \end{bmatrix}$$

$\Re(\cdot)$ and $\Im(\cdot)$ denote the real and imaginary parts of their argument and $\mathbf{R}_{lm}(\tau_k)$ is the (l,m)-th entry of matrix $\mathbf{R}(\tau_k)$. An optimal solution is finally given by $\mathbf{v} = [v_1, v_2, v_3]^T = sign(u_1)\mathbf{u}$ (u_1 being the 1^{st} entry of $\mathbf{u}$) where $\mathbf{u}$ is the unit-norm eigenvector corresponding to the principal eigenvalue of matrix $\mathbf{F}\mathbf{F}^H$. The angle parameters are finally obtained as:

$$c = \sqrt{\frac{v_1+1}{2}} \quad \text{and} \quad s = \frac{v_2 - jv_3}{2c} \quad (16)$$

IV. RA-SOBI: Impulsive noise case

Here, we extend the previous algorithm and propose a robust version to deal with impulsive noise.

To do so, we modify A-SOBI algorithm in two spots:

- First, we replace the GOPAST algorithm with a more robust one to estimate the eigen components of the covariance matrix $\mathbf{C}_x(t)$.
- We apply the diagonalization step to robust estimates of the correlation matrices of the whitened signal $\mathbf{R}_t(\tau_k)$.

These two steps are detailed next.

A. Robust whitening

In order to estimate the principal components used in the whitening step, we propose here to use the recently robust algorithms proposed in [19]: Givens-Mahalanobis-Fast Approximated Power Iteration (GMFAPI) and its low-cost version Givens-Hard Thresholding-Fast Approximated Power Iteration (GHTFAPI). These algorithms consist of minimizing, under unitary constraint, the following weighted least squares criterion in order to estimate the principal subspace of the covariance matrix $\mathbf{C}_x$ (represented by its $n \times p$ orthonormal basis $\mathbf{D}(t)$):

$$J(\mathbf{D}(t)) = \sum_{j=1}^{t} \beta^{t-j}\omega(j) \left\| \mathbf{x}(j) - \mathbf{D}(t)\mathbf{D}^H(t)\mathbf{x}(j) \right\|^2 \quad (17)$$

where $\omega(j)$ is a soft weighting (resp. hard thresholding) factor considered by GMFAPI (resp. GHTFAPI). This, implicitly, consists of using a robust instantaneous estimate of the covariance matrix $\mathbf{C}_x(t)$ according to

$$\mathbf{C}_x(t) = \beta\mathbf{C}_x(t-1) + \omega(t)\mathbf{x}(t)\mathbf{x}^H(t). \quad (18)$$

One can solve this minimization problem using the power iteration method resumed by the data compression expressed in (19) and the orthonormalization (20)

$$\mathbf{C}_{xy}(t) = \mathbf{C}_x(t)\mathbf{D}(t-1) \quad (19)$$

$$\mathbf{D}(t)\mathbf{R}(t) = \mathbf{C}_{xy}(t) \quad (20)$$

A fast implementation is reached with the introduction of an intermediate $p \times p$ matrix denoted $\mathbf{Z}(t)$ representing an estimation of $\mathbf{C}_{yy}^{-1}(t)$ with $\mathbf{y}(t) = \mathbf{D}(t-1)^H\mathbf{x}(t)$. This implementation is summarized in table 1 One can refer to [19] for the detailed implementation.

Algorithm 1 (table 1): Subspace basis $\mathbf{D}(t)$ implementation

Initialization

$\mathbf{D}(0) = \begin{bmatrix} \mathbf{I}_p \\ 0_{(n-p)\times p} \end{bmatrix}$; $\mathbf{Z}(0) = \mathbf{I}_p$

for each time step do :

input vector $\mathbf{x}(t)$

$\mathbf{y}(t) = \mathbf{D}^H(t-1)\mathbf{x}(t)$

$\mathbf{h}(t) = \mathbf{Z}(t-1)\mathbf{y}(t)$

$\epsilon^2(t) = ||\mathbf{x}(t)||^2 - ||\mathbf{y}(t)||^2$

weight computation: see table 2

$\mathbf{g}(t) = \frac{\mathbf{h}(t)\omega(t)}{\beta + \mathbf{y}^H(t)\mathbf{h}(t)\omega(t)}$

$\tau(t) = \frac{\epsilon^2(t)}{1+\epsilon^2(t)||\mathbf{g}(t)||^2 + \sqrt{1+\epsilon^2(t)||\mathbf{g}(t)||^2}}$

$\eta(t) = 1 - \tau(t)||\mathbf{g}(t)||^2$

$\mathbf{y}'(t) = \eta(t)\mathbf{y}(t) + \tau(t)\mathbf{g}(t)$

$\mathbf{h}'(t) = \mathbf{Z}^H(t-1)\mathbf{y}'(t)$

$\boldsymbol{\epsilon}(t) = \frac{\tau(t)}{\eta(t)}(\mathbf{Z}(t-1)\mathbf{g}(t) - (\mathbf{h}'(t)^H\mathbf{g}(t))\mathbf{g}(t))$

$\mathbf{Z}(t) = \frac{1}{\beta}(\mathbf{Z}(t-1) - \mathbf{g}(t)\mathbf{h}'(t)^H + \mathbf{e}(t)\mathbf{g}^H(t))$

$\mathbf{e}'(t) = \eta(t)\mathbf{x}(t) - \mathbf{D}(t-1)\mathbf{y}'(t)$

$\mathbf{D}(t) = \mathbf{D}(t-1) + \mathbf{e}'(t)\mathbf{g}^H(t)$

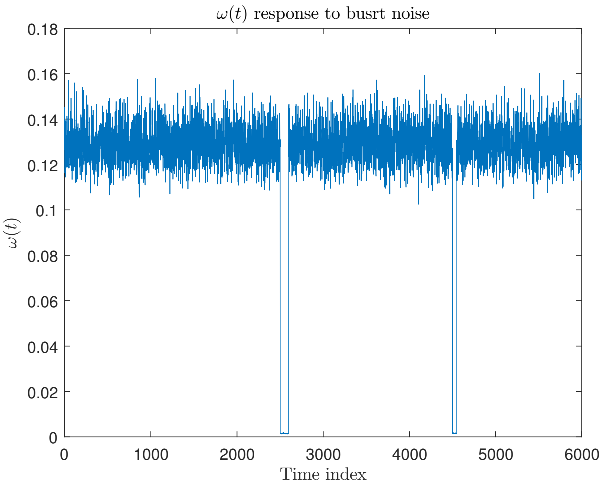

Fig. 2: Illustration of the weight criterion $\omega(t)$ when the burst noise appears in two occasions: at time 2500 and 4500 with duration 100 and 50, respectively.

Now, this criterion aims to mitigate the corrupted observations impact thanks to the weighting factor. Therefore, $\omega(t)$ should take small values when impulsive noises occur (see illustration in Fig. 2 for the weight factor of GMFAPI). To achieve that, GMFAPI and GHTFAPI algorithms use the following approaches:

1— GMFAPI: This algorithm uses the inverse of the Mahalanobis distance [25] to be the weight

$$\omega(t) = \frac{1}{d_M^2\left(\mathbf{x}(t), \mathbf{C}_x^{-1}(t-1)\right)} = \frac{1}{\mathbf{x}^H(t)\mathbf{C}_x^{-1}(t-1)\mathbf{x}(t)} \tag{21}$$

The following fast implementation of the latter was proposed in [19]

$$\mathbf{x}(t)^H\mathbf{C}_x^{-1}(t-1)\mathbf{x}(t) = \mathbf{y}(t)^H\mathbf{h}(t) + \frac{||\mathbf{x}(t)||^2 - ||\mathbf{y}(t)||^2}{\sigma_n^2(t-1)} \tag{22}$$

Where $\mathbf{h}(t) = \mathbf{Z}(t-1)\mathbf{y}(t)$. Now, the noise power $\sigma_n^2(t)$ can be estimated as

$$\sigma_n^2(t) = \frac{tr\big(\mathbf{C}_x(t)\big) - tr\big(\mathbf{C}_{yy}(t)\big)}{n-p} \tag{23}$$

Where $tr(\cdot)$ refers to the matrix trace operator. $T_x(t) = tr\big(\mathbf{C}_x(t)\big)$ and $T_y(t) = tr\big(\mathbf{C}_{yy}(t)\big)$ can be effectively calculated as

$$T_x(t) = \beta T_x(t-1) + \omega(t)||\mathbf{x}(t)||^2$$
$$T_y(t) = \beta T_y(t-1) + \omega(t)||\mathbf{y}(t)||^2$$

hence, the weight factor is obtained by $\omega(t) = 1/\delta'(t)$ where

$$\delta'(t) = \mathbf{y}(t)^H\mathbf{h}(t) + \frac{(||\mathbf{x}(t)||^2 - ||\mathbf{y}(t)||^2)(n-p)}{T_x(t-1) - T_y(t-1)} \tag{24}$$

2— GHTFAPI: To reduce further the computational cost of $\omega(t)$, a hard thresholding method was used in GHTFAPI. Indeed, considering the fact that $\epsilon^2(t) = ||\mathbf{x}(t)||^2 - ||\mathbf{y}(t)||^2$ represents approximately the instantaneous noise power, GHTFAPI uses it to determine whether to include the corresponding observation or not, according to

$$\begin{cases} \omega(t) = 0 & \text{if } \epsilon^2(t) > threshold(t) \\ \omega(t) = 1 & \text{if } \epsilon^2(t) < threshold(t) \end{cases} \tag{25}$$

where the threshold is determined using the well known Inter Quartile Range (IQR) method [26] applied on the last L samples. Hence, we define:

$$\tilde{\epsilon}(t) = \{\epsilon^2(t-L+1)\cdots\epsilon^2(t)\}$$
$$IQR(t) = Q3\big(\tilde{\epsilon}(t)\big) - Q1\big(\tilde{\epsilon}(t)\big)$$

$$threshold(t) = Q3\big(\tilde{\epsilon}(t)\big) + 1.5IQR(t) \tag{26}$$

with $Q1(\cdot)$ and $Q3(\cdot)$ representing respectively the lower and the upper quartiles.

One can refer to table 2 for an overview of the two methods regarding the computation of the weighting factor.

At this stage, we have a robust estimate of a basis of the principal subspace as well as an estimate of $\mathbf{Z}(t) \approx \mathbf{C}_{yy}^{-1}(t)$. Thus, we can perform an adaptive diagonalization on $\mathbf{Z}(t)$ using elementary Givens rotations to obtain the principal components $\mathbf{U}_s$ from the principal subspace basis and the principal eigenvalues Λ from $\mathbf{Z}(t)$. Indeed, we have $\mathbf{U}_s = \mathbf{DQ}$ where $\mathbf{Q}$ is a unitary matrix that can be expressed by $\mathbf{Q} = \prod \tilde{\mathbf{G}}_{\tilde{l},\tilde{m}}(\tilde{\theta},\tilde{\phi})$ and $\tilde{\mathbf{G}}_{\tilde{l},\tilde{m}}(\tilde{\theta},\tilde{\phi})$ is a Givens rotation as expressed in (10).

Now, to compute the latter, let us address the fact that the exact matrix $\mathbf{Z}'$ is diagonal. Hence, we define the Givens parameters so that the off diagonal elements of $\mathbf{Z}' = \mathbf{GZG}^H$ are minimized. That is to say:

$$(\tilde{\theta},\tilde{\phi}) = \arg\min_{\tilde{\theta},\tilde{\phi}} \sum_{a\neq b} |Z'_{ab}|^2 = \arg\max_{\tilde{\theta},\tilde{\phi}} |\tilde{\mathbf{v}}^T\tilde{\mathbf{f}}(t)|^2 \tag{27}$$

with

$$\tilde{\mathbf{v}} = \begin{bmatrix} \cos(2\tilde{\theta}) \\ \sin(2\tilde{\theta})\cos(\tilde{\phi}) \\ \sin(2\tilde{\theta})\sin(\tilde{\phi}) \end{bmatrix} \text{ and } \tilde{\mathbf{f}}(t) = \begin{bmatrix} Z_{ll}(t) - Z_{mm}(t) \\ 2\Re\left(Z_{lm}(t)\right) \\ 2\Im\left(Z_{lm}(t)\right) \end{bmatrix}$$

Algorithm 2 (table 2): weighting factor computation

1: *for each time step do :*
Mahalanobis distance:
2: $\delta'(t) = \mathbf{y}(t)^H\mathbf{h}(t) + \frac{(||\mathbf{x}(t)||^2 - ||\mathbf{y}(t)||^2)(n-p)}{T_x(t-1) - T_y(t-1)}$
3: $\omega(t) = 1/\delta'(t)$
4: $T_x(t) = \beta T_x(t-1) + \omega(t)||\mathbf{x}(t)||^2$
5: $T_y(t) = \beta T_y(t-1) + \omega(t)||\mathbf{y}(t)||^2$
Hard thresholding:
6: $\tilde{\epsilon}(t) = [\epsilon^2(t-L+1)\cdots\epsilon^2(t)]$
7: $IQR(t) = Q3(\tilde{\epsilon}(t)) - Q1(\tilde{\epsilon}(t))$
8: $threshold(t) = Q3(\tilde{\epsilon}(t)) + 1.5IQR(t)$
9: **if** $\epsilon^2(t) < threshold(t)$ **then**
10: $\omega(t) = 1$
11: **else**
12: $\omega(t) = 0$
13: **end if**

Algorithm 3 (table 3): Whitening

1: **Initialization**
2: $\mathbf{D}(t)$ and $\mathbf{Z}(t)$ from Table 1
3: $(\tilde{l}, \tilde{m}) = (1, 2)$

Givens rotations:
1^{st} rotation:
4: Select rotation indices as in (28)
5: $\tilde{\mathbf{f}}(t) = \left[Z_{\tilde{l}\tilde{l}}(t) - Z_{\tilde{m}\tilde{m}}(t); 2\Re(Z_{\tilde{l}\tilde{m}}(t)); 2\Im(Z_{\tilde{l}\tilde{m}}(t))\right]^T$
6: $\tilde{\mathbf{v}} = sign(\tilde{f}_1(t))\tilde{\mathbf{f}}(t)/||\tilde{f}(t)||$
7: $c = \sqrt{\frac{\tilde{v}_1+1}{2}}$ and $s = \frac{\tilde{v}_2 - \tilde{v}_3}{2c}$
8: Determine $\tilde{\mathbf{G}}$ as in equation (10)
9: $\mathbf{Z}(t) = \tilde{\mathbf{G}}\mathbf{Z}(t)\tilde{\mathbf{G}}^H$
10: $\mathbf{D}(t) = \mathbf{D}(t)\tilde{\mathbf{G}}^H$
2^{nd} rotation
11: Select rotation indices as in (13)
12: Apply the same steps in lines: (5-10)
13: $\mathbf{W}(t) = \mathbf{Z}(t)^{1/2}\mathbf{U}_s^H(t)$

Algorithm 4 (table 4): Joint diagonalization

1: **Initialization**
2: $\mathbf{W}(t)$ from Table 3 ; $\mathbf{U}(0) = \mathbf{I}_p$

for each time step do :
for $k = 1, \cdots, K$
3: $\hat{\mathbf{s}}(t-\tau_k) = \mathbf{U}(t)\mathbf{W}(t)\mathbf{x}(t-\tau_k); k = 1, \cdots, K$
4: $\mathbf{R}_t(\tau_k) = \beta\mathbf{R}_{t-1}(\tau_k) + \sqrt{\omega(t)\omega(t-\tau_k)}\hat{\mathbf{s}}(t)\hat{\mathbf{s}}^H(t-\tau_k)$
end for
5: determine (l, m) as in (12) or (13)
6: $\mathbf{F} = \begin{bmatrix} \mathbf{R}_{ll}(\tau_1) - \mathbf{R}_{mm}(\tau_1) & \cdots & \mathbf{R}_{ll}(\tau_K) - \mathbf{R}_{mm}(\tau_K) \\ 2\Re(\mathbf{R}_{lm}(\tau_1)) & \cdots & 2\Re(\mathbf{R}_{lm}(\tau_K)) \\ 2\Im(\mathbf{R}_{lm}(\tau_1)) & \cdots & 2\Im(\mathbf{R}_{lm}(\tau_K)) \end{bmatrix}$
7: $\mathbf{u} = eigs(\mathbf{F}\mathbf{F}^H, 1)$
8: $\mathbf{v} = sign(u_1)\mathbf{u}$
9: $c = \sqrt{\frac{\tilde{v}_1+1}{2}}$ and $s = \frac{\tilde{v}_2 - \tilde{v}_3}{2c}$
10: define $\mathbf{G}(t)$ as in (10)
for $k = 1, \cdots, K$
11: $\mathbf{R}(\tau_k) = \mathbf{G}(t)\mathbf{R}(\tau_k)\mathbf{G}^H(t)$
end for
12: $\mathbf{U}(t) = \mathbf{U}(t)\mathbf{G}^H(t)$

An optimal solution to that is

$$\tilde{\mathbf{v}} = [\tilde{v}_1, \tilde{v}_2, \tilde{v}_3]^T = sign(\tilde{f}_1(t))\tilde{\mathbf{f}}(t)/||\tilde{\mathbf{f}}(t)||$$

where $sign(\tilde{f}_1(t))$ refers to the sign of the first entry of $\tilde{\mathbf{f}}(t)$. The parameters c and s are defined as in (16) with replacing $\mathbf{v}$ by $\tilde{\mathbf{v}}$.

Then again, to determine the indices $(\tilde{l}, \tilde{m})$, we select the largest off diagonal element of $\mathbf{Z}(t)$:

$$(\tilde{l}, \tilde{m}) = \arg \max_{\{(a,b)|a<b\}} |Z_{ab}(t)| \quad (28)$$

Finally, for a better convergence speed, another rotation per iteration is performed using automatic sweeping as in (13).

The whitening can be performed as in table 3

B. Robust estimation of correlation matrices

To further enhance the robustness od th proposed algorithm, we propose to estimate the correlation matrices via robust estimation techniques.

Hence, since we consider several correlation matrices with different time lags for the diagonalization step, it is important to take into account whether the observations at these time lags are corrupted or valid. Thus, we propose here to weight the estimates of the correlations with a combination of $\omega(t)$ and $\omega(t-\tau_k)$ to ensure the mitigation of erroneous data.

Hence, the correlation matrices are estimated as

$$\mathbf{R}_t(\tau_k) = \beta\mathbf{R}_{t-1}(\tau_k) + \sqrt{\omega(t)\omega(t-\tau_k)}\hat{\mathbf{s}}(t)\hat{\mathbf{s}}^H(t-\tau_k) \quad (29)$$

Therefore, RA-SOBI performs the diagonalization step exactly as A-SOBI by replacing the correlation matrices estimates in (8) by their robust ones given in (29) and its summary is given in table 4

V. Simulation Results

In order to investigate our algorithms performance, we simulate the streaming data vectors $\mathbf{x}(t)$ of dimension $n = 8$ during $N = 6000$ time steps. Those observations are generated using $p = 3$ source signals corresponding to filtered complex circular white Gaussian processes by three AR filters of order 1 with respective coefficients $a_1 = 0.95\exp(j0.5)$, $a_2 = 0.75\exp(j0.7)$ and $a_3 = 0.55\exp(j0.3)$.

These signals are then mixed and corrupted with additive white centered Gaussian noise $\mathbf{n}_G(t)$ imposing an SNR of $5dB$.

We run $M = 100$ Monte Carlo simulations for all scenarios and we evaluate the algorithms performance using the mean rejection level defined in [3] as:

$$I(t) = \frac{1}{M}\sum_{m=1}^{M}\left(\sum_{i=1}^{p}\frac{\sum_{j\neq i}|\mathbf{L}_{ij}(t)|^2}{|\mathbf{L}_{ii}(t)|^2}\right) \quad (30)$$

Where $\mathbf{L}(t) = \mathbf{B}(t)\mathbf{A}(t)$ is close to a diagonal matrix (after removing the permutation indeterminacy).

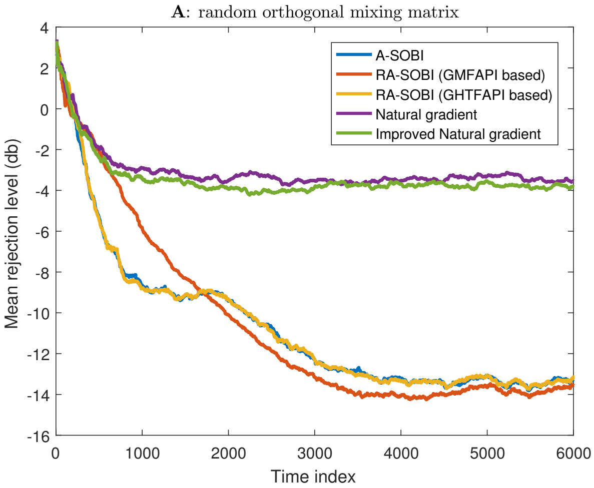

Fig. 3: Algorithms performance in case of random mixing matrix and Gaussian noise only environment

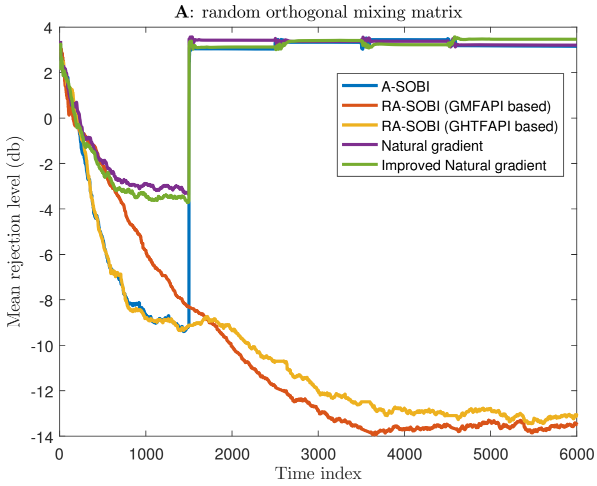

Fig. 4: Algorithms performance in case of random mixing matrix

Note that for all algorithms the forgetting factor is set to be $\beta = 0.999$ and the number of considered correlation matrices is $K = 10$. Also, the number of samples used in the threshold determination for the GHTFAPI algorithm is $L = 1000$.

To make sure our algorithm performs well in an adaptive manner, we first run the simulation without adding the burst noise. From Fig. 3, we can clearly notice that our proposed algorithm RA-SOBI in its two versions as well as the A-SOBI outperform the other state of the art algorithms when dealing with Gaussian noise environment.

In addition, to evaluate the robustness of the algorithm, we simulate a burst noise as defined in (3) that occurs during four periods of time, namely: $P_1 = [1500, 1550]$, $P_3 = [2500, 2600]$, $P_3 = [3500, 3600]$ and $P_4 = [4500, 4600]$ causing the SNR to drop to $-40dB$.

Now, we investigate three scenarios with different mixing matrices:

- First, we consider a general case with $\mathbf{A}$ being an $(n \times p)$ random matrix.
- Then, we investigate a scenario where the mixing matrix $\mathbf{A}$ is structured as:

$$\mathbf{A} = [\mathbf{a}(\omega_1); \mathbf{a}(\omega_2); \cdots ; \mathbf{a}(\omega_p)] \tag{31}$$

  where:

  $\mathbf{a}(\omega_k) = [1; e^{j\omega_k}, \cdots, e^{j\omega_k(n-1)}]^T$, with $\omega_k = \pi \sin(\theta_k)$.

  θ_k is a direction of arrival chosen in our second experiment as $\theta_1 = 10°$, $\theta_2 = 30°$ and $\theta_3 = 50°$.

- Finally, we study the case where the system is slowly time varying. For that, we use the latter structure of $\mathbf{A}$ while linearly varying the directions of arrival such that it begins with $\theta_1(0) = 20°$, $\theta_2(0) = 10°$, $\theta_3(0) = -10°$ and it ends at $\theta_1(N-1) = 30°$, $\theta_2(N-1) = 0°$, $\theta_3(N-1) = -10°$ (i.e. θ_3 is kept time invariant).

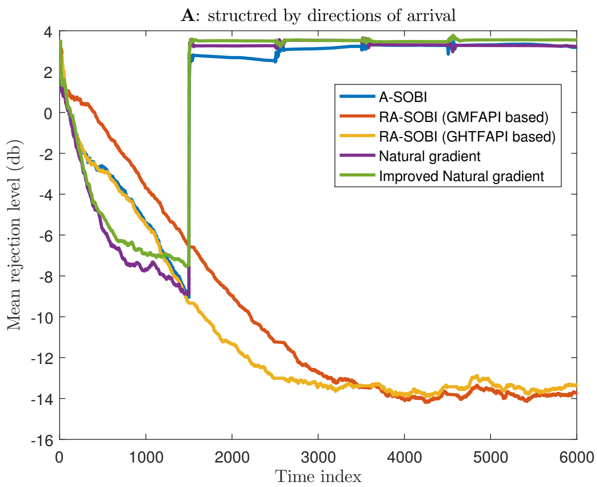

Fig. 5: Algorithms performance in case of directions of arrival dependent mixing matrix

As we can see from Fig. 4, Fig. 5, and 6, it is clear that our robust algorithm RA-SOBI (with its two versions) maintains a good source separation throughout the entire testing period, while the non-robust algorithms (A-SOBI, natural gradient and the improved natural gradient [12]) collapse at the occurrence of the first noise impulse.

VI. Conclusion

In this paper, we introduced robust adaptive algorithms for blind source separation based on second order decorrelation. The latter was achieved thanks to robust fast whitening, as well as Givens rotations based joint diagonalization performed on robust estimates of correlation matrices. Our algorithms are shown, via simulation experiments, to be effective in an impulsive noise environment, while having low computational complexity of order $O(np + pK)$ flops per iteration.

References

[1] S. haykin, "Unsupervised Adaptive Filtering: Blind source separation," John Wiley & Sons, Inc., 2000.

[2] P. Comon and C. Jutten, "Handbook of Blind Source Separa-

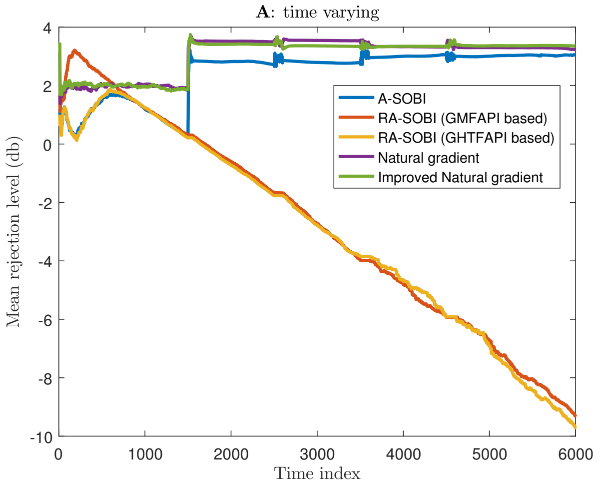

Fig. 6: Algorithms performance in case of time varying mixing matrix

tion: Independent component analysis and applications," 1st ed., Academic press, 2010.

[3] A. Belouchrani, K. Abed-Meraim, J.F. Cardoso and E. Moulines, "A blind source separation technique using second-order statistics," *IEEE Trans. Signal process.*, vol. 42, no. 2, pp. 434-444, 1997. DOI: 10.1109/78.554307

[4] J-F. Cardoso and B.H. Laheld, "Equivariant adaptive source separation," *IEEE Trans. Signal process.*, vol. 44, no. 12, pp. 3017-3030, 1996. DOI: 10.1109/78.553476

[5] L. Jiawen and L. Congxin, "A two-step adaptive blind source separation for machine sound," in *6th World Congress on Intelligent Control and Automation*, Dlaian, China, pp. 5424-5427, 2006. DOI: 10.1109/WCICA.2006.1714108

[6] C. Mühlmann (Ph.D. Dissertation), "Advances in blind source separation for spatial data," Dept, Math. Stat., Univ. Jyväskylä, Wien, Austria, 2021.

[7] Y. Pan, M. Matilainen, S. Taskinen and K. Nordhausen, "A review of second-order blind identification methods," *Wiley interdisciplinary reviews: computational statistics*, e1550, 2021. DOI: 10.1002/wics.1550

[8] A. Ziehe, P. Laskov, G. Nolte and K-R. Mãžller, "A fast algorithm for joint diagonalization with non-orthogonal transformations and its application to blind source separation," *Journal of Machine Learning Research*, vol. 5, pp. 777-800, Jul. 2004.

[9] Y. Xiang, K. Abed-Meraim and Y. Hua, "Adaptive blind source separation by second order statistics and natural gradient," in *ICASSP*, Phoenix, AZ, USA, pp. 2917-2920, 1999. DOI: 10.1109/ICASSP.1999.761373

[10] X-L. Zhu and X-D. Zhang, "Adaptive RLS algorithm for blind source separation using a natural gradient," *IEEE Signal Processing Letters*, vol. 9, no. 12, pp. 432-435, 2002. DOI: 10.1109/LSP.2002.806047

[11] X. Xie, Q. Shi and R. Wu, "A new variable step-size equivariant adaptive source separation algorithm," in *Asia-Pacific Conference on Communications*, pp. 479–482, 2007. DOI: 10.1109/APCC.2007.4433478

[12] M. Thameri, K. Abed-Meraim and A. Belouchrani, "New algorithms for adaptive BSS," in *11th ISSPA*, pp. 590–594, 2012. DOI: 10.1109/ISSPA.2012.6310620

[13] H.A. Al-Barhan, S.M. Elyass, T.R. Saeed, G.M. Hatem and H.T. Ziboon, "Blind Source Separation Based on random mixing coefficient," in *3rd IICETA*, pp. 164-168, 2020. DOI: 10.1109/IICETA50496.2020.9318954

[14] A. Belouchrani and A. Cichocki, "Robust whitening procedure in blind source separation context," *Electronics letters*, vol. 36, no. 24, pp. 2050-2051, 2000.

[15] F.J. Theis, N.S. Müller, C. Plant and C. Böhm, "Robust second-order source separation identifies experimental responses in biomedical imaging," in *International conference on latent variable analysis and signal separation*, pp. 466-473, 2010. DOI: 10.1007/978-3-642-15995-4_58

[16] J. Miettinen, M. Matilainen, K. Nordhausen, and S. Taskinen, "Extracting conditionally heteroskedastic components using independent component analysis," *Journal of time series analysis*, vol. 41, no. 2, pp. 293-311, 2020. DOI: 10.1111/jtsa.12505

[17] P. Ilmonen, K. Nordhausen, H. Oja and F. Theis, " An affine equivariant robust second-order BSS method," in *International Conference on Latent Variable Analysis and Signal Separation*, pp. 328-335, 2015. DOI: 10.1007/978-3-319-22482-4_38

[18] N. Lietzén, K. Nordhausen and P. Ilmonen, "Complex valued robust multidimensional SOBI," in *International Conference on Latent Variable Analysis and Signal Separation*, pp. 131-140, 2017. DOI: 10.1007/978-3-319-53547-0_13

[19] Z. Bekhtaoui, K. Abed-Meraim and A. Meche, "Robust adaptive algorithms for fast principal component analysis," *Digital signal processing*, vol. 127, p. 103561, 2022. DOI: 10.1016/j.dsp.2022.103561

[20] M. Thameri, K. Abed-Meraim and A. Belouchrani, "Low complexity adaptive algorithms for Principal and Minor Component Analysis," *Digital Signal Processing*, vol. 23, no. 1, pp. 19-29, 2013. DOI: 10.1016/j.dsp.2012.09.007

[21] L. Clavier, G.W. Peters, F. Septier & I. Nevat, "Impulsive noise modeling and robust receiver design," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, pp. 1-30, 2021. DOI: 10.1186/s13638-020-01868-1

[22] A. Awad, "Denoising images corrupted with impulse, Gaussian, or a mixture of impulse and Gaussian noise," *Engineering Science and Technology, an International Journal*, vol. 22, no.3, pp. 746-753, Jun 2019. DOI: 10.1016/j.jestch.2019.01.012

[23] H. Gao, M. Chen, Y. Du and A. Jakobsson, "Monostatic MIMO radar direction finding in impulse noise," *Digital Signal Processing*, Vol. 117, p. 103198, Oct 2021. DOI: 10.1016/j.dsp.2021.103198

[24] R. Wilcox, "Introduction to Robust Estimation and Hypothesis Testing," 5th Ed. Elsevier, Sep 2021.

[25] P.C. Mahalanobis (1936), "On the generalised distance in statistics," *Proceedings of the National Institute of Sciences of India*, vol. 2, no. 1, pp. 49–55, Retrieved 2016.

[26] P.D. Domanski, "Study on statistical outlier detection and labelling," *International Journal of Automation and Computing*, vol. 17, no. 6, pp. 788-811, 2020. DOI: 10.1007/s11633-020-1243-2

[27] H. Shongwe, A.J. Han Vinck, and H.C. Ferreira, "A study on impulsive noise and its models," *J. of South African Institute of Electrical Engineers*, vol. 106, no. 3, pp. 119-131, Sep 2015. DOI: 10.23919/SAIEE.2015.8531938

Zineb Bekhtaoui was born in 1993. She received her State Engineering degree from Institut National des Télécommunications et des Techniques de l'Information et de la Communication (INTTIC), Oran, Algeria, in 2016, and the M.Sc. degree from INTTIC in the same year. She is currently a Ph.D. student at Université des Sciences et de la Technologie d'Oran, Mohamed

Boudiaf (USTOMB) affiliated to Signals and Images Laboratory (Laboratoire Signaux et Systèmes).

Karim Abed-Meraim was born in 1967. He received the State Engineering Degree from the École Polytechnique, Palaiseau, France, in 1990, the State Engineering Degree from the École Nationale Supérieure des Télécommunications (ENST), Paris, France, in 1992, the M.Sc. degree from Paris XI University, Orsay, France, in 1992, and the Ph.D. degree in the field of signal processing and communications from ENST, in 1995. From 1995 to 1998, he was a Research Staff with the Electrical Engineering Department, The University of Melbourne, where he worked on several research project related to "Blind System Identification for Wireless Communications", "Blind Source Separation," and "Array Processing for Communications." From 1998 to 2012, he has been an Assistant Professor, then an Associate Professor with the Signal and Image Processing Department, Télécom ParisTech. In September 2012, he joined the University of Orléans, France (PRISME Laboratory), as a Full Professor. Since October 2021, he is a member of the Institut Universitaire de France. His research interests include signal processing for communications, adaptive filtering and tracking, array processing, and statistical performance analysis. He is the author of about 500 scientific publications, including book chapters, international journal and conference papers, and patents. Dr. Abed-Meraim is currently a member of the IEEE SAM-TC and a Senior Area Editor of the IEEE Transactions on Signal Processing. Professor Karim Abed-Meraim is an IEEE Fellow.

Abdelkrim Meche was born in 1975. He received his B.S., M.S. and Ph.D. in 1997, 2004 and 2013, respectively, from the University of Sciences and Technology, Mohamed Boudiaf, (USTO, MB), Oran, Algeria. He is currently an associate professor at the Department of Automation, and also a member at the Signal and Image Laboratory at the Electrical Engineering Faculty of the same university. His research interests are in the fields of Statistical Signal Processing, Array Processing, Adaptive and Kalman Filtering, Image/Video Processing and Target Tracking.

Messaoud Thameri was born in 1985 (M'Sila, Algeria). He received the State Engineering Degree from Ecole Polytechnique Algiers, Algiers, Algeria, in 2009, and the Ph.D. degree in signal and images processing from Telecom ParisTech, Paris, France, in 2013. Currently, he is an assistant professor at Ecole Supérieure Ali Chabati, Algiers. His research interests include array and radar signal processing, adaptive filtering and subspace tracking, parameter estimation, and asymptotic performance analysis.