Parameterized Source Separation For Delayed Spectroscopic Signals

Hassan Mortada, Vincent Mazet, Charles Soussen, Christophe Collet, Lionel Poisson

Parameterized Source Separation For Delayed Spectroscopic Signals

Hassan Mortada[1], Vincent Mazet[1], Charles Soussen[2], Christophe Collet[1] and Lionel Poisson[3]

[1] *ICube, Université de Strasbourg - CNRS, Illkirch, France*
[2] *Laboratoire des signaux et systèmes, CentraleSupélec–CNRS–Université Paris-Sud, Université Paris-Saclay, 91192 Gif-sur-Yvette, France*
[3] *Laboratoire Francis Perrin, CEA, CNRS, IRAMIS/Service des Photons Atomes et Molécules, 91191 Gif-sur-Yvette, France*

Abstract

This paper addresses delayed (also known as anechoic) source separation when the source shape can be modeled by parameterized waveforms. An Alternating Least Squares (ALS) scheme is proposed to estimate the source shape parameters in a first step and both the mixing coefficients and the delays in a second step. For the challenging delay parameter estimation step, we adopt a strategy inspired by greedy algorithms. For highly correlated sources, the separation becomes ambiguous, and a second algorithm is proposed: a regularization term is added to favor slow delay evolution within each source. Results on synthetic and real data demonstrate the effectiveness of both algorithms compared to state-of-the-art methods for highly correlated Gaussian waveforms.

Keywords: Source separation, anechoic unmixing, correlated sources, delays.

1. Introduction

We consider the detection and estimation of patterns with varying characteristics in a temporal sequence of signals. This problem arises when dealing with decomposition of spectra, *i.e.*, the estimation of emission lines (the so-called peaks) in a series of spectroscopic signals. The peak characteristics are assumed to evolve slowly through the sequence. In optical spectroscopy, acquisitions are obtained when varying a physical parameter. For instance, in time-resolved

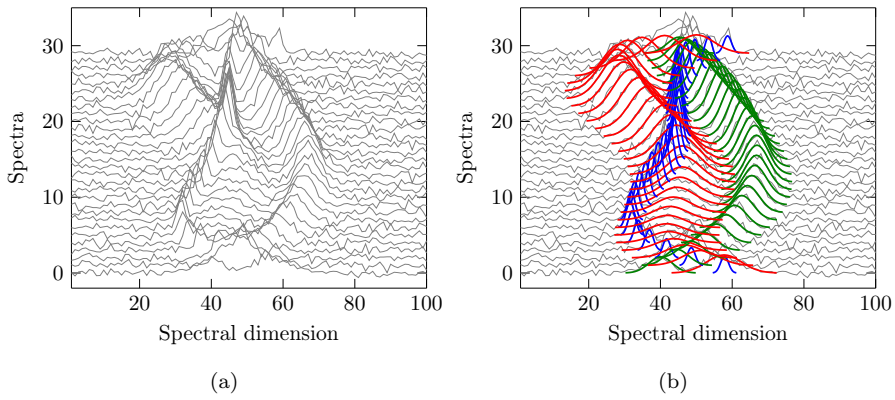


Figure 1: (a) Synthetic sequence of spectra, each containing 3 Gaussian waveforms (peaks). (b) The peaks and their labels. Each color represents the peaks sharing the same label.

photoelectron spectroscopy, several spectra are acquired at different times [37]. In astrophysics, the study of galaxy kinematics [11, 43] leads to the acquisition of multispectral images where each pixel identifies with a spectrum whose peaks undergo varying delays due to the redshift and internal gas motions (Doppler effect). In chemometrics as well [20], external variations such as temperature or viscosity induce a spectrum shift and variation.

In many applications, prior knowledge or experiments allow the practitioner to model the peaks by a parameterized waveform, *e.g.*, Gaussian [43, 24] or Lorentzian [14] functions. The first goal of this work is to estimate the amplitudes, delays and shape parameters of the peaks in each spectrum. The second goal is to match together the estimated peaks occurring in different spectra, that is to assign a distinct label to each peak arising in a given spectrum. We will assume that the shape parameters of the peaks having the same label are invariant from one spectrum to another. Fig. 1 shows an example of synthetic spectra and the expected output.

The problem was addressed in the Bayesian framework for a one-dimensional sequence of spectra [24], modeled as the sum of Gaussian waveforms with Markovian priors on their parameters to promote a slow evolution. The parameters

and labels were then estimated using a reversible jump Monte Carlo Markov chain algorithm. This method can deal with varying shape parameters of the peaks having the same label but suffers from high computation time.

Another statistical approach has been proposed in [44], which models each signal in the sequence as a shifted version of a periodic, unknown waveform. The shift and the amplitude of the function are also unknown. In contrast to our problem, the author considers a unique waveform in each signal and there is no assumption of slow evolution of the unknowns through the sequence.

The problem can also be seen as a sparse approximation problem using an overcomplete dictionary, constructed by sampling the peak parameters in the case of parameterized peaks [1, 3, 39, 45]. Processing multiple spectra can be addressed by estimating the peak parameters separately for each spectrum, then matching the peaks together. However, this approach does not take the slow evolution knowledge into account during peak estimation. Alternatively, joint sparse approximation use a common dictionary for all the signals. Most approaches assume that the data signals are simultaneously sparse (they share a common support), which is often referred to as the Multiple Measurement Vector case [41]. However, simultaneous sparsity is a very restrictive assumption which does not hold when the peaks have different delays in the spectra. Dynamic approaches like recursive sparse approximation [42] compute the sparse approximation of a spectrum from the knowledge of the sparse approximation of the previous spectra. Social sparsity was introduced in [18] by promoting a structure between the sparse representation of consecutive data signals with less restrictive support assumptions. However, these approaches require a precise definition of the neighborhood which is not always possible, and hence they are not well-suited to the problem of peak estimation and labeling. Sparse approximation methods have also been used for deep learning [33] in case of a convolutional dictionary. However, such approaches require a large dataset of already decomposed data which is not always available.

Finally, the problem can be seen as a delayed source separation problem [7] where peaks with the same label are associated with unknown sources, and

observed spectra are seen as mixtures. The reader is referred to [30] for a review of source mixing models, *e.g*, instantaneous, anechoic and echoic, and the related source separation approaches. Delayed source separation is a specific case of convolutive source separation [5, 17] where the filter kernels are Dirac functions. The main advantage of this approach is that estimation and labeling of peaks are done simultaneously, thus avoiding a two-step approach. Note that in optical spectroscopy, the number of sources is often smaller than the number of mixtures, yielding overdetermined source separation. Many contributions of the literature were addressed in audio signal processing where delayed source separation is also referred to as anechoic source separation. The sources are not parameterized and have to be estimated either in a blind or semi-blind framework. Since the problem is ill-posed, regularization is necessary, *e.g.*, by imposing non-negativity of the sources and mixing coefficients (see [28] in the case of instantaneous mixtures). Besides, strong assumptions such as source independence and decorrelation are often made. When dealing with delays, most approaches rely on delay linearization strategies such as applying Taylor expansion on the mixtures in the temporal domain [6, 16]. Another strategy is to analyze the mixtures in the frequency [26, 32] or time-frequency [29, 31] domains, so that delays become phases which allows one to use instantaneous source separation approaches coupled with phase estimation methods. However, the independence and non-correlation assumptions are often not valid in many real-world applications [8]. Efforts have been done though to relax these assumptions. In [34], the strong W-disjoint orthogonality assumption needed in the DUET algorithm [46] and the assumption that the sources must be orthogonal in the time-frequency domain are relaxed: each source must be dominant at least in one time-frequency window. However, this assumption becomes invalid when the evolution of a source delay is fast from one mixture to another or when two delayed source signals significantly overlap in some mixtures. In optical spectroscopy, some delayed source separation methods have been proposed. In [13, 15], a time warping strategy was used to cancel the effect of the delays on each mixture after finding the delays over a predefined discrete grid using an

exhaustive search strategy. The sources and amplitudes were then estimated in the linear least squares sense. However, the exhaustive search is not feasible when the delays take many values. In [27], the Non-negative Matrix Factorization (NMF) method was extended to consider delays: non-negative amplitudes and sources were found by using multiplicative updates, and delays were estimated with a gradient descent algorithm coupled with a maximizing cross-correlation procedure to reduce the effect of local minima. However, our experiments (see Section 5) showed that this method performs poorly for highly correlated sources. In addition, it is not straightforward to impose a slow evolution of the delays.

Hereafter, the terms mixture and source respectively represent an observed spectrum and a peak having the same label in the observed spectra. An original algorithm is proposed to address delayed source separation when the sources are defined as parameterized functions. The problem is stated as an optimization problem in Section 2 and a sparse-based Alternating Least Squares (ALS) strategy is proposed in Section 3. This method was sketched in [25] but in this paper we provide more detailed problem formulation and we analyze the separation limits of the method. The latter limits motivate us to develop the brand new method of Section 4: the slow delay evolution prior is considered to better discriminate spectrally overlapping and similar sources. Finally, results on synthetic and real data are presented in Section 5.

The notations are as follows. Bold and lowercase variables correspond to vectors. Bold and uppercase variables correspond to matrices. The i-th row and j-th column of a matrix $\boldsymbol{M}$ are respectively denoted as $\boldsymbol{M}_{i:}$ and $\boldsymbol{M}_{:j}$. The notation $\big(u\big)_+$ refers to $\big(u\big)_+ = \max(u, 0)$. The arrow notations $\searrow$ and $\nearrow$ respectively refer to the decrement and increment operators.

2. Parameterized Source Separation

Let us consider I mixtures $x_i(\lambda)$, each being the noisy sum of J parameterized sources $s(\lambda; w_j)$ such that:

$$x_i(\lambda) = \sum_{j=1}^{J} a_{ij} s(\lambda - c_{ij}; w_j) + n_i(\lambda) \quad i = 1, \ldots, I \tag{1}$$

where $a_{ij} \in \mathbb{R}_+$ is the amplitude of source j in mixture i (the application considered in this paper shows emission lines but not absorption lines), $c_{ij} \in \mathbb{R}$ is the delay of source j in mixture i, $w_j \in \mathbb{R}$ is the shape parameter of source j, and $n_i(\lambda)$ represents observation and modeling errors. Throughout the paper, the number of sources J is supposed to be known. Since the sources are parameterized, the delayed source separation problem comes to the estimation of the amplitudes a_{ij}, delays c_{ij} and shape parameters w_j. For the sake of clarity, we will consider that the shape parameter is scalar, but the extension to a multidimensional parameter is straightforward. Furthermore, the sources are supposed to be modeled with the same parameterized function, however, the proposed method works even when the functions are different. Of course, using the same parameterized function yields highly correlated sources; we will see that the proposed method remains accurate in this case.

It is known [29, 35] that even in the noiseless case, model (1) suffers from at least three indeterminacies, namely, scale, permutation and phase inderminacies, the latter being related to the fact that a source signal is known up to some arbitrary delay. The use of parameterized sources helps to overcome the scale and the phase ambiguities by imposing their energy to be equal to a specified value and to set their maximum amplitude for $\lambda = 0$. Besides, the permutation ambiguity cannot be alleviated but does not yield any aftereffect.

The delays are supposed to be discretized over a grid with step Δ, thus $c_{ij} = \ell_{ij}\Delta$ with $\ell_{ij} \in \mathbb{N}$. Note that Δ may be lower than the sampling step of the mixture signals, which is set to 1 without loss of generality. In addition, the sources are supposed to be normalized. As said before, normalization allows us to overcome the scale indeterminacy. We denote by

$\boldsymbol{x}_i = \begin{bmatrix} x_i(1) & x_i(2) & \dots & x_i(N) \end{bmatrix}^T$ the vector gathering the samples of the i-th mixture. Similarly, the samples of each delayed source are gathered in a vector $\boldsymbol{s}[\ell_{ij}; w_j]$ defined as the unit-norm vector satisfying:

$$\boldsymbol{s}[\ell_{ij}; w_j] \propto \begin{bmatrix} s(1 - \ell_{ij}\Delta; w_j) & s(2 - \ell_{ij}\Delta; w_j) & \dots & s(N - \ell_{ij}\Delta; w_j) \end{bmatrix}^T, \quad (2)$$

where $\propto$ refers to proportionality. Equation (1) now reads:

$$\boldsymbol{x}_i = \sum_{j=1}^{J} a_{ij} \boldsymbol{s}[\ell_{ij}; w_j] + \boldsymbol{n}_i, \quad i = 1, \dots, I, \quad (3)$$

where $\boldsymbol{n}_i$ refers to the noise in mixture i. Supposing the noise to be white and Gaussian, the maximum likelihood estimator is obtained by minimizing the criterion:

$$E(\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{w}) = \sum_{i=1}^{I} \varepsilon(\boldsymbol{A}_{i:}, \boldsymbol{L}_{i:}, \boldsymbol{w}), \quad (4)$$

where $\varepsilon(\boldsymbol{A}_{i:}, \boldsymbol{L}_{i:}, \boldsymbol{w})$ is the quadratic error related to mixture i:

$$\varepsilon(\boldsymbol{A}_{i:}, \boldsymbol{L}_{i:}, \boldsymbol{w}) = \left\| \boldsymbol{x}_i - \sum_{j=1}^{J} a_{ij} \boldsymbol{s}[\ell_{ij}; w_j] \right\|_2^2, \quad (5)$$

and $\boldsymbol{A} \in \mathbb{R}_+^{I \times J}$, $\boldsymbol{L} \in \mathbb{N}^{I \times J}$, $\boldsymbol{w} \in \mathbb{R}_+^J$ respectively gather the amplitudes a_{ij}, delays ℓ_{ij} and shape parameters w_j for mixtures i and sources j:

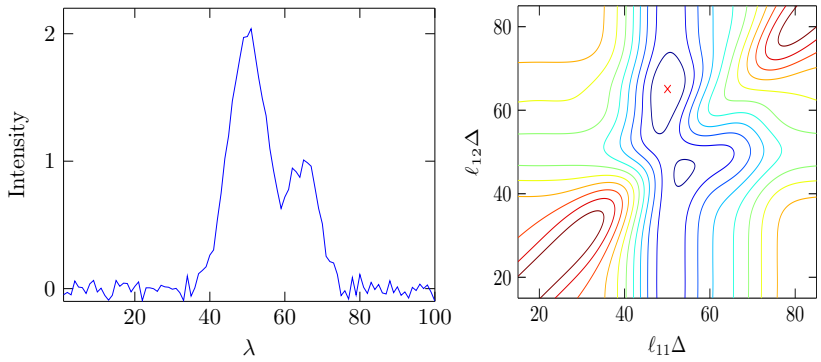
$$\boldsymbol{A} = \begin{bmatrix} a_{11} & \dots & a_{1J} \\ \vdots & \ddots & \vdots \\ a_{I1} & \dots & a_{IJ} \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_{1:} \\ \vdots \\ \boldsymbol{A}_{I:} \end{bmatrix} \quad (6)$$

$$\boldsymbol{L} = \begin{bmatrix} \ell_{11} & \dots & \ell_{1J} \\ \vdots & \ddots & \vdots \\ \ell_{I1} & \dots & \ell_{IJ} \end{bmatrix} = \begin{bmatrix} \boldsymbol{L}_{1:} \\ \vdots \\ \boldsymbol{L}_{I:} \end{bmatrix} \quad (7)$$

$$\boldsymbol{w} = \begin{bmatrix} w_1 & \dots & w_J \end{bmatrix}^T. \quad (8)$$

Therefore, the source separation problem is formulated as the following constrained minimization problem:

$$\min_{\boldsymbol{A} \geq \boldsymbol{0}, \boldsymbol{L}, \boldsymbol{w}} E(\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{w}). \quad (9)$$



(a) Simulated mixture (b) $\boldsymbol{L}_{1:} \to \varepsilon(\boldsymbol{A}^*_{1:}, \boldsymbol{L}_{1:}, \boldsymbol{w}^*)$

Figure 2: (a) A noisy mixture of two Gaussian sources ($I = 1$, $J = 2$) with $a^*_{11} = 6.0, a^*_{12} = 2.7$, $\ell^*_{11}\Delta = 50$, $\ell^*_{12}\Delta = 65$, $w^*_1 = 5$, $w^*_2 = 4$, $N = 100$. (b) The criterion $\boldsymbol{L}_{1:} \to \varepsilon(\boldsymbol{A}^*_{1:}, \boldsymbol{L}_{1:}, \boldsymbol{w}^*)$ admits local minimizers and flat surfaces. The red $\times$ indicates the global minimizer.

3. Sparse-based Alternating Least Squares

The optimization problem (9) is challenging because of the non-convexity of the criterion E, induced by the nonlinearity of model (3) with respect to $\boldsymbol{L}$ and $\boldsymbol{w}$. As an example, Fig. 2 displays the variations of criterion $E(\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{w})$ with respect to $\boldsymbol{L}$ in the case of $I = 1$ mixture and $J = 2$ sources: one can see that it admits multiple local minimizers as well as flat regions, making its optimization difficult even for this simple example.

ALS is an iterative descent strategy in which criterion E is minimized with respect to a block of variables while fixing the others, and *vice versa*. The algorithm stops if the criterion decrease at one iteration becomes lower than a tolerance ρ. ALS is not guaranteed to converge towards the global minimizer of (9) since it is a block minimization of a non-convex criterion. However, this method is often used in delayed source separation, where *e.g.*, the sources, delays and amplitudes are alternately estimated [15, 26, 29]. However, the latter methods suffer from several limitations, as stated in section 1, making them unsuitable for the considered problem.

The proposed ALS scheme is given in Algorithm 1. The criterion is alternately

Algorithm 1: ALS scheme for $\min_{\boldsymbol{A},\boldsymbol{L},\boldsymbol{w}} E(\boldsymbol{A},\boldsymbol{L},\boldsymbol{w})$.

Initialization: $\widehat{\boldsymbol{A}} = \widehat{\boldsymbol{L}} = \boldsymbol{0}_{I\times J}$, $\widehat{\boldsymbol{w}} \sim \mathcal{U}_{[w_{\min},w_{\max}]^J}$

do

  $(\boldsymbol{A}^0,\boldsymbol{L}^0,\boldsymbol{w}^0) \leftarrow (\widehat{\boldsymbol{A}},\widehat{\boldsymbol{L}},\widehat{\boldsymbol{w}})$

  for $i = 1 \nearrow I$ **do**

    $(\widehat{\boldsymbol{A}}_{i:},\widehat{\boldsymbol{L}}_{i:}) \leftarrow \underset{\boldsymbol{A}_{i:},\boldsymbol{L}_{i:}}{\operatorname{argmin}}\, \varepsilon(\boldsymbol{A}_{i:},\boldsymbol{L}_{i:},\widehat{\boldsymbol{w}})$ (see Algorithm 2)

  end

  $\widehat{\boldsymbol{w}} \leftarrow \underset{\boldsymbol{w}}{\operatorname{argmin}}\, E\big(\widehat{\boldsymbol{A}},\widehat{\boldsymbol{L}},\boldsymbol{w}\big)$

while $\frac{E(\boldsymbol{A}^0,\boldsymbol{L}^0,\boldsymbol{w}^0)-E(\widehat{\boldsymbol{A}},\widehat{\boldsymbol{L}},\widehat{\boldsymbol{w}})}{E(\boldsymbol{A}^0,\boldsymbol{L}^0,\boldsymbol{w}^0)} \geq \rho$

optimized with respect to the source shape parameters $\boldsymbol{w}$ on the one hand and the delays $\boldsymbol{L}$ and amplitudes $\boldsymbol{A}$ on the other hand. The shape estimation subproblem is a continuous non-linear least-squares problem:

$$\widehat{\boldsymbol{w}} \leftarrow \underset{\boldsymbol{w}}{\operatorname{argmin}}\, E(\boldsymbol{A},\boldsymbol{L},\boldsymbol{w}), \tag{10}$$

and is solved using the Levenberg-Marquardt algorithm [21]. In the first iteration, the shape parameters are initialized from the uniform distribution $\mathcal{U}_{[w_{\min},w_{\max}]^J}$ where $w_{\min}$ and $w_{\max}$ are the extreme values defined by the user. In the following iterations, $\boldsymbol{w}$ is initialized with the estimates obtained at the previous iteration. The amplitude and delay estimation is the main challenge:

$$(\widehat{\boldsymbol{A}},\widehat{\boldsymbol{L}}) \leftarrow \underset{\boldsymbol{A}\geq\boldsymbol{0},\boldsymbol{L}}{\operatorname{argmin}}\, E(\boldsymbol{A},\boldsymbol{L},\boldsymbol{w}). \tag{11}$$

It is detailed hereafter.

3.1. Amplitude and Delay Estimation

It follows from (4) that problem (11) is separable to I independent subproblems:

$$\min_{\boldsymbol{A}_{i:}\geq\boldsymbol{0},\boldsymbol{L}_{i:}} \varepsilon(\boldsymbol{A}_{i:},\boldsymbol{L}_{i:},\boldsymbol{w}) \quad \forall i. \tag{12}$$

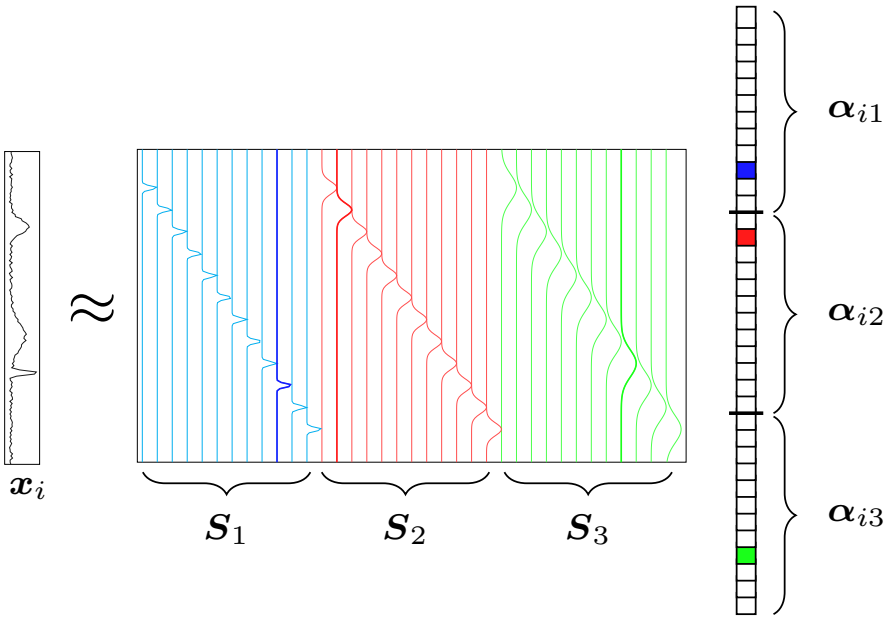


Figure 3: Sparse representation model of a mixture $\boldsymbol{x}_i$ with $J = 3$ sources. Each block $\boldsymbol{S}_j$ of the dictionary gathers the delayed versions of $\boldsymbol{s}[\ell_{ij}; w_j]$. The sparse representation is structured so that each block vector $\boldsymbol{\alpha}_{ij}$ is 1-sparse.

We propose to minimize each $\varepsilon(\boldsymbol{A}_{i:}, \boldsymbol{L}_{i:}, \boldsymbol{w})$ using a sparse approximation algorithm aiming to sparsely represent mixture i in an overcomplete dictionary. This choice is justified because spectroscopic signals contain very few peaks.

3.1.1. Dictionary Formulation

The dictionary is a block matrix $[\boldsymbol{S}_1, \ldots, \boldsymbol{S}_J]$, with $\boldsymbol{S}_j \in \mathbb{R}^{N \times M}$ the block gathering M delayed versions of source j (see Fig. 3):

$$\boldsymbol{S}_j = \Big[\boldsymbol{s}[0; w_j] \quad \boldsymbol{s}[1; w_j] \quad \ldots \quad \boldsymbol{s}[M-1; w_j]\Big]\,. \tag{13}$$

Each mixture $\boldsymbol{x}_i$ is approximated as:

$$\boldsymbol{x}_i \approx \sum_{j=1}^{J} \boldsymbol{S}_j \boldsymbol{\alpha}_{ij} \tag{14}$$

where $\boldsymbol{\alpha}_{ij} = [0, \ldots, \ a_{ij}, \ \ldots, 0]^T \in \mathbb{R}_+^M$ is a 1-sparse vector, so that each source appears at most once in each mixture (see Fig. 3). The value and index of the non-zero element in $\boldsymbol{\alpha}_{ij}$ respectively indicate the amplitude a_{ij} and delay ℓ_{ij}.

Thus, the optimization problem (12) can be rewritten as:

$$\min_{\forall j, \boldsymbol{\alpha}_{ij} \geq \mathbf{0}} \left\| \boldsymbol{x}_i - \sum_{j=1}^{J} \boldsymbol{S}_j \boldsymbol{\alpha}_{ij} \right\|_2^2 \text{ s.t. } \forall\ j, \quad \|\boldsymbol{\alpha}_{ij}\|_0 \leq 1, \tag{15}$$

where the ℓ_0 "norm" $\|\cdot\|_0$ counts the number of non-zero coefficients in a vector.

3.1.2. NN-OMP-like Implementation for Delayed Source Separation

Greedy algorithms are effective and efficient when the sparsity level J is small and known [40]. In addition, their structure is simple and can be easily adapted to the recovery of structured sparse representations. The non-negative orthogonal matching pursuit (NN-OMP) algorithm [4] is an iterative algorithm composed of three steps:

1. the so-called forward selection step consists in choosing the column of the dictionary that is the most positively correlated with the residual;
2. the amplitudes corresponding to the chosen columns are updated by solving a non-negative least–squares estimation problem [19];
3. the residual is updated by removing the contributions of the chosen columns.

NN-OMP considers the constraint in (15) to be $\sum_{j=1}^{J} \|\boldsymbol{\alpha}_{ij}\|_0 \leq J$. However, this constraint does not enforce the sources to appear at most once in each mixture. Therefore, the proposed implementation (Algorithm 2) consists in forcing the sparse vector to be structured in blocks, each $\boldsymbol{\alpha}_{ij}$ being 1-sparse:

1. the dictionary column that is the most positively correlated with the residual is selected, yielding the corresponding source $\widehat{j}$ and its delay $\widehat{\ell}_{i\widehat{j}}$ (lines 2–6). The selected source $\widehat{j}$ is then added to the list $\mathcal{J}$ of selected sources (line 7) so that the dictionary columns embedded in the blocks $\boldsymbol{S}_j$ for sources $j \in \mathcal{J}$ will not be tested in the next iterations;
2. the amplitudes of the selected sources gathered in $\mathcal{J}$ are estimated by solving the non-negative linear least–squares problem:

$$\widehat{\boldsymbol{A}}_{i:} \leftarrow \underset{\boldsymbol{A}_{i:}}{\operatorname{argmin}}\, \varepsilon(\boldsymbol{A}_{i:}, \widehat{\boldsymbol{L}}_{i:}, \widehat{\boldsymbol{w}}) \text{ s.t. } \begin{cases} \boldsymbol{A}_{i\mathcal{J}} \geq \mathbf{0} \\ \boldsymbol{A}_{i\bar{\mathcal{J}}} = \mathbf{0}, \end{cases} \tag{16}$$

   where $\bar{\mathcal{J}}$ denotes the complementary subset of $\mathcal{J}$ (line 8);

Algorithm 2: Implementation of $(\widehat{\boldsymbol{A}}_{i:}, \widehat{\boldsymbol{L}}_{i:}) \leftarrow \underset{\boldsymbol{A}_{i:}, \boldsymbol{L}_{i:}}{\operatorname{argmin}}\, \varepsilon(\boldsymbol{A}_{i:}, \boldsymbol{L}_{i:}, \widehat{\boldsymbol{w}})$

Initialization: $\widehat{\boldsymbol{A}}_{i:} = \widehat{\boldsymbol{L}}_{i:} = \boldsymbol{0}_{1\times J},\ \mathcal{J} = \emptyset,\ \boldsymbol{r}_i = \boldsymbol{x}_i$

1 **for** $k = 1 \nearrow J$ **do**

2 **for** $j \in \{1, \dots, J\} \backslash \mathcal{J}$ **do**

3 $\tilde{\ell}_{ij} \leftarrow \underset{\ell}{\operatorname{argmax}}\ \left(\boldsymbol{r}_i^T \boldsymbol{s}[\ell; \widehat{w}_j]\right)_+$

4 **end**

5 $\widehat{j} \leftarrow \underset{j \notin \mathcal{J}}{\operatorname{argmax}}\ \left(\boldsymbol{r}_i^T \boldsymbol{s}[\tilde{\ell}_{ij}; \widehat{w}_j]\right)_+$

6 $\widehat{\ell}_{i\widehat{j}} \leftarrow \tilde{\ell}_{i\widehat{j}}$

7 $\mathcal{J} \leftarrow \mathcal{J} \cup \{\ \widehat{j}\ \}$

8 Update amplitudes $\widehat{\boldsymbol{A}}_{i\mathcal{J}}$ according to (16)

9 $\boldsymbol{r}_i \leftarrow \boldsymbol{x}_i - \sum_{j \in \mathcal{J}} \widehat{a}_{ij} \boldsymbol{s}[\widehat{\ell}_{ij}; \widehat{w}_j]$

10 **end**

11 **if** $\varepsilon(\boldsymbol{A}^0_{i:}, \boldsymbol{L}^0_{i:}, \widehat{\boldsymbol{w}}) < \varepsilon(\widehat{\boldsymbol{A}}_{i:}, \widehat{\boldsymbol{L}}_{i:}, \widehat{\boldsymbol{w}})$ **then** $(\widehat{\boldsymbol{A}}_{i:}, \widehat{\boldsymbol{L}}_{i:}) \leftarrow (\boldsymbol{A}^0_{i:}, \boldsymbol{L}^0_{i:})$ **end**

3. lastly, the residual vector $\boldsymbol{r}_i$ is updated (line 9).

The main difference between NN-OMP and our implementation lies in the first step. Also, line 11 ensures a decrease of the criterion by invalidating the estimates if they produce a criterion value greater than the value obtained at the previous iteration of Algorithm 1. Furthermore, Algorithm 2 offers the possibility to obtain a variable number of sources per mixture: an additional stopping criterion may be added such that the loop breaks if the residual norm $\|\boldsymbol{r}_i\|_2$ becomes lower than a threshold, *e.g.*, related to the noise variance.

Finally, we also investigated recent methods to refine delay estimation, which is constrained to be on a grid. Indeed, the discretization induced by sampling the sources in the dictionary yields a bias in the estimation, as already been discussed in [38]. The easiest way to reduce this bias is to decrease the value of Δ, but this results in a bigger dictionary, and in turn, in an increase of the computational burden. So, we tested an interpolation extension proposed in [9, 10]. It turned

out that this extension is more time-consuming than a simple decrease in the sampling step Δ. Therefore, we rather choose to work with a fine grid.

3.2. Shape Discriminating Limit

Algorithm 1 is able to assign the estimated peaks to the right source because the sources can be discriminated by their shape parameter (see Fig. 3). We are interested in finding the resolution limit, that is the least difference between the shape parameters of two sources beyond which the sources can be discriminated. For this purpose, we consider $I = 40$ mixtures each with $N = 200$ samples, and $J = 2$ Gaussians sources of widths w_1 and w_2 and with constant delays through the mixtures. We gradually vary the ratio w_2/w_1 from 0.5 to 1.5. For each ratio, we measure the switch percentage defined as the percentage of wrongly assigned peaks over the total number of peaks (a peak is wrongly assigned if it belongs to source 1 while it is assigned to source 2 and *vice versa*). The experiment is repeated for three SNRs (Fig. 4). The SNR is defined as ten times the log-ratio of the mean energy of the noiseless mixtures with the noise variance σ_n^2:

$$\mathrm{SNR} = 10 \log_{10} \left[\frac{1}{\sigma_n^2 \cdot I \cdot N} \sum_{i=1}^{I} \left\| \sum_{j=1}^{J} a_{ij}^* \boldsymbol{s}[\ell_{ij}^*; w_j^*] \right\|_2^2 \right], \tag{17}$$

where the * symbol represents the value of the ground-truth parameters. The results show that for high SNR, the proposed method can separate the sources whatever the ratio w_2/w_1 (except of course when both parameters are equal). The ability to separate the sources decreases with the SNR. This motivates us to introduce a regularization term to overcome this limitation.

4. Slow Delay Evolution Enforcement

4.1. Regularized Criterion

This section aims at promoting slow evolution for each source delay $\boldsymbol{L}$ through the mixtures for two reasons. First, the slow evolution is, in practice, the consequence of a short acquisition time between measurements (such as in photoelectron spectroscopy [12]) or neighboring sensors (such as in galaxy

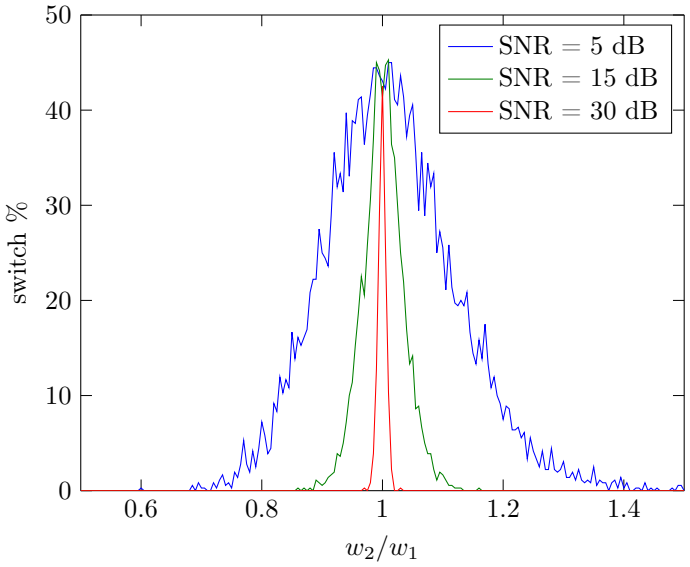


Figure 4: Switch percentage with respect to the ratio of Gaussian widths, for three SNRs.

kinematics [11] or audio recorded mixtures [6]). Second, it would help to discriminate highly correlated sources. To this end, a regularization term $R(\boldsymbol{L})$ is added to the data-fit term $E(\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{w})$ defined in (4):

$$F(\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{w}) = E(\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{w}) + \tau R(\boldsymbol{L}) \tag{18}$$

where τ is the regularization parameter to be set by the user. For the sake of clarity, we assume in the sequel that i is a one-dimensional index. The regularization term $R(\boldsymbol{L})$ measures the sum of squared differences between consecutive delays:

$$R(\boldsymbol{L}) = \sum_{i=2}^{I} \sum_{j=1}^{J} \left(\ell_{ij} - \ell_{(i-1)j}\right)^2. \tag{19}$$

Criterion F is optimized using Algorithm 3 which is based on an ALS scheme. Since $R(\boldsymbol{L})$ does not depend on $\boldsymbol{w}$, the shape estimation (line 4) is identical to (10). Thus, it is computed using the Levenberg-Marquardt algorithm. On the contrary, the amplitude and delay estimation step (line 3) differs from (11)-(12) because F does not read as a separable sum with respect to $\boldsymbol{L}_{i:}$. Indeed, the

Algorithm 3: ALS scheme for $\min_{\boldsymbol{A},\boldsymbol{L},\boldsymbol{w}} F(\boldsymbol{A},\boldsymbol{L},\boldsymbol{w})$.

Initialization: $\widehat{\boldsymbol{A}} = \widehat{\boldsymbol{L}} = \boldsymbol{0}_{I\times J}$, $\widehat{\boldsymbol{w}} \sim \mathcal{U}_{[w_{\min},w_{\max}]^J}$

do

  $(\boldsymbol{A}^0,\boldsymbol{L}^0,\boldsymbol{w}^0) \leftarrow (\widehat{\boldsymbol{A}},\widehat{\boldsymbol{L}},\widehat{\boldsymbol{w}})$

  $(\widehat{\boldsymbol{A}},\widehat{\boldsymbol{L}}) \leftarrow \operatorname{argmin}_{\boldsymbol{A},\boldsymbol{L}} F(\boldsymbol{A},\boldsymbol{L},\widehat{\boldsymbol{w}})$ (see Algorithm 4)

  $\widehat{\boldsymbol{w}} \leftarrow \operatorname{argmin}_{\boldsymbol{w}} F\big(\widehat{\boldsymbol{A}},\widehat{\boldsymbol{L}},\boldsymbol{w}\big)$

while $\frac{F(\boldsymbol{A}^0,\boldsymbol{L}^0,\boldsymbol{w}^0)-F(\widehat{\boldsymbol{A}},\widehat{\boldsymbol{L}},\widehat{\boldsymbol{w}})}{F(\boldsymbol{A}^0,\boldsymbol{L}^0,\boldsymbol{w}^0)} \geq \rho$

terms $(\ell_{ij}-\ell_{(i-1)j})^2$ appearing in the regularization term $R(\boldsymbol{L})$ not only depend on ℓ_{ij} but also on $\ell_{(i-1)j}$. Let us now detail the update rules for $\boldsymbol{A}$ and $\boldsymbol{L}$.

Algorithm 4: Implementation of $(\widehat{\boldsymbol{A}},\widehat{\boldsymbol{L}}) \leftarrow \operatorname{argmin}_{\boldsymbol{A},\boldsymbol{L}} F(\boldsymbol{A},\boldsymbol{L},\widehat{\boldsymbol{w}})$

Initialization: $\widehat{\boldsymbol{A}} = \widehat{\boldsymbol{L}} = \boldsymbol{L}^{\text{temp}} = \boldsymbol{0}_{I\times J}$, $\mathcal{J} = \emptyset$, $\boldsymbol{r}_i = \boldsymbol{x}_i\ \forall i$

for $k = 1 \nearrow J$ **do**

  for $j \in \{1,\dots,J\}\backslash\mathcal{J}$ **do**

    Compute $\widetilde{\boldsymbol{L}}_{:j}$ defined in (22) using Algorithm 5

  end

  $\widehat{j} \leftarrow \operatorname{argmax}_{j\notin\mathcal{J}} \sum_{i=1}^{I} \big(\boldsymbol{r}_i^T \boldsymbol{s}[\widetilde{\ell}_{ij};\widehat{w}_j]\big)_+^2 - \tau \sum_{i=2}^{I} \Big(\widetilde{\ell}_{ij} - \widetilde{\ell}_{(i-1)j}\Big)^2$

  $\mathcal{J} \leftarrow \mathcal{J} \cup \{\,\widehat{j}\,\}$

  $\boldsymbol{L}^{\text{temp}}_{:\widehat{j}} \leftarrow \widetilde{\boldsymbol{L}}_{:\widehat{j}}$

  $\boldsymbol{A}^{\text{temp}} \leftarrow \operatorname{argmin}_{\boldsymbol{A}} F(\boldsymbol{A},\boldsymbol{L}^{\text{temp}},\widehat{\boldsymbol{w}})$ s.t. $\{\boldsymbol{A}_{:\mathcal{J}} \geq \boldsymbol{0}, \boldsymbol{A}_{:\overline{\mathcal{J}}} = \boldsymbol{0}\}$

  if $F(\boldsymbol{A}^{\text{temp}},\boldsymbol{L}^{\text{temp}},\widehat{\boldsymbol{w}}) > F(\widehat{\boldsymbol{A}},\widehat{\boldsymbol{L}},\widehat{\boldsymbol{w}})$ **then** Break **end**

  $(\widehat{\boldsymbol{A}},\widehat{\boldsymbol{L}}) \leftarrow (\boldsymbol{A}^{\text{temp}},\boldsymbol{L}^{\text{temp}})$

  for $i = 1 \nearrow I$ **do** $\boldsymbol{r}_i \leftarrow \boldsymbol{x}_i - \sum_{j\in\mathcal{J}} \widehat{a}_{ij}\boldsymbol{s}[\widehat{\ell}_{ij};\widehat{w}_j]$ **end**

end

if $F(\boldsymbol{A}^0,\boldsymbol{L}^0,\widehat{\boldsymbol{w}}) < F(\widehat{\boldsymbol{A}},\widehat{\boldsymbol{L}},\widehat{\boldsymbol{w}})$ **then** $(\widehat{\boldsymbol{A}},\widehat{\boldsymbol{L}}) \leftarrow (\boldsymbol{A}^0,\boldsymbol{L}^0)$ **end**

4.2. Amplitude and Slow Delay Estimation

Algorithm 4 is proposed to optimize $F(\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{w})$ with respect to $\boldsymbol{A}$ and $\boldsymbol{L}$. It is a greedy algorithm in the spirit of Algorithm 2, and, in this way, reproduces the three steps of the NN-OMP framework already stated in Section 3.1.2. However, Algorithm 4 takes all mixtures as inputs while Algorithm 2 considers a single mixture. The estimation of the source delays in line 3 of Algorithm 4 is deferred to Algorithm 5.

At each iteration, the source $\widehat{j}$ inducing the largest decrease of criterion F is selected (lines 2–5) and added to the list of selected sources $\mathcal{J}$ (line 6). The corresponding delays $\boldsymbol{L}_{:\widehat{j}}^{\text{temp}}$ are computed. Then, the amplitudes of the selected sources in $\mathcal{J}$ in all the mixtures are estimated using a non-negative least squares solver (line 8). Finally, the residual vectors of all the mixtures are updated (line 11). Note that adding a new source results in a decrease of the data-fit term $E(\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{w})$ but makes the regularization term $R(\boldsymbol{L})$ increasing. In other words, the criterion $F(\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{w}) = E(\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{w}) + \tau R(\boldsymbol{L})$ can either increase or decrease. Therefore, the condition in line 9 is set to break the loop in case of an increase of F, so that Algorithm 4 is indeed a descent algorithm. Similarly, the condition in line 13 ensures a decrease of the criterion by invalidating the estimates if they produce a criterion value that is larger than the value obtained at the previous iteration of Algorithm 3.

4.3. Delay Estimation with an ICM-like Algorithm

Let us now specify the rule for selecting the source $\widehat{j}$ among all candidate sources $j \notin \mathcal{J}$. Since $\widehat{j}$ is defined as the source yielding the largest decrease of criterion F, lines 2–4 in Algorithm 4 aim at estimating, for each source j that has not already been selected, the value of the corresponding delays $\boldsymbol{L}_{:j}$ (denoted by $\widetilde{\boldsymbol{L}}_{:j}$) as well as the value of F obtained while considering the set of sources $\mathcal{J} \cup \{j\}$. To do so, one needs to consider the minimization of $F(\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{w})$ with respect to $\boldsymbol{A}_{:j}$ and $\boldsymbol{L}_{:j}$, while fixing the values of $\boldsymbol{L}_{:j'}$ and $\boldsymbol{A}_{:j'}$ for $j' \in \mathcal{J}$:

$$\widetilde{\boldsymbol{L}}_{:j} \leftarrow \underset{\boldsymbol{L}_{:j}}{\operatorname{argmin}} \min_{\boldsymbol{A}_{:j} \geq \boldsymbol{0}} \sum_{i=1}^{I} \left\| \boldsymbol{r}_i - a_{ij} \boldsymbol{s}[\ell_{ij}; \widehat{w}_j] \right\|_2^2 + \tau \sum_{i=2}^{I} (\ell_{ij} - \ell_{(i-1)j})^2. \tag{20}$$

The minimization of (20) with respect to $\boldsymbol{A}_{:j}$ while fixing $\boldsymbol{L}_{:j}$ has a closed form solution:

$$\forall i, \quad a_{ij} = \left(\boldsymbol{r}_i^T \boldsymbol{s}[\ell_{ij}; \widehat{w}_j]\right)_+. \tag{21}$$

Plugging back (21) into (20), (20) simplifies to:

$$\widetilde{\boldsymbol{L}}_{:j} \leftarrow \underset{\boldsymbol{L}_{:j}}{\operatorname{argmax}} \sum_{i=1}^{I} \left(\boldsymbol{r}_i^T \boldsymbol{s}[\ell_{ij}; \widehat{w}_j]\right)_+^2 - \tau \sum_{i=2}^{I} (\ell_{ij} - \ell_{(i-1)j})^2. \tag{22}$$

which is the cost function appearing at line 5 in Algorithm 4.

Because (22) is a combinatorial problem, we resort to the Iterated Conditional Modes (ICM) algorithm [2] which is a popular coordinate-wise optimization method in image processing. While it converges to a local optimizer, it generally gives good results. The proposed implementation to solve (22) is given in Algorithm 5. Specifically, at each iteration (lines 9–15), all mixtures $i \in \{1, \ldots, I\}$ are swept using the following way: a starting mixture i_0 is randomly chosen (line 10), then the delays in mixtures $i_0 + 1$ to I are sequentially estimated, and the same procedure is used for mixtures $i_0 - 1$ to 1. Whenever a mixture $i \in \{1, \ldots, I\}$ is visited, the related delay $\widetilde{\ell}_{ij}$ is estimated (in lines 12–14) by maximizing the criterion in (22) with respect to ℓ_{ij} whilst fixing the other delays $\widetilde{\ell}_{i'j}$ with $i' \neq i$. The convergence is reached when the relative distance between the estimates of two consecutive iterates is smaller than a tolerance threshold ξ (line 16).

The initialization is performed in the same way (lines 1–7), the major difference being that, at the first iteration, some delays are not yet estimated and, in consequence, cannot be considered in (22). So, we simply discard the terms with unknown values from the equation. Note also that the first mixture to be considered is the one maximizing the data-fit term (line 1).

4.4. Remarks

If no regularization is considered (*i.e.*, $\tau = 0$), then $F(\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{w}) = E(\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{w})$. We then recommend to use Algorithm 1 rather than Algorithm 3 since the former exploits the separability of the criterion. Conversely, when τ tends to infinity, the

Algorithm 5: Implementation of

$$\widetilde{\boldsymbol{L}}_{:j} \leftarrow \underset{\boldsymbol{L}_{:j}}{\operatorname{argmax}} \sum_{i=1}^{I} \left(\boldsymbol{r}_i^T \boldsymbol{s}[\ell_{ij}; \widehat{w}_j]\right)_+^2 - \tau \sum_{i=2}^{I} (\ell_{ij} - \ell_{(i-1)j})^2$$

Initialization: $\widetilde{\boldsymbol{L}}_{:j} = \boldsymbol{0}_{I\times 1}$

$(i_0, \widetilde{\ell}_{i_0 j}) \leftarrow \underset{i,\ell}{\operatorname{argmax}} \left(\boldsymbol{r}_i^T \boldsymbol{s}[\ell; \widehat{w}_j]\right)_+^2$

for $i = (i_0 + 1) \nearrow I$ **do**

  $\widetilde{\ell}_{ij} \leftarrow \underset{\ell}{\operatorname{argmax}} \left(\boldsymbol{r}_i^T \boldsymbol{s}[\ell; \widehat{w}_j]\right)_+^2 - \tau(\ell - \widetilde{\ell}_{(i-1)j})^2$

end

for $i = (i_0 - 1) \searrow 1$ **do**

  $\widetilde{\ell}_{ij} \leftarrow \underset{\ell}{\operatorname{argmax}} \left(\boldsymbol{r}_i^T \boldsymbol{s}[\ell; \widehat{w}_j]\right)_+^2 - \tau(\ell - \widetilde{\ell}_{(i+1)j})^2$

end

do

  $\widetilde{\boldsymbol{L}}_{:j}^{\text{old}} \leftarrow \widetilde{\boldsymbol{L}}_{:j}$

  $i_0 \sim \mathcal{U}_{\{1,\ldots,I\}}$

  for $i = i_0 \ \nearrow \ I$ and $i = (i_0 - 1) \ \searrow \ 1$ **do**

    if $i = 1$ **then** $\widetilde{\ell}_{ij} \leftarrow \underset{\ell}{\operatorname{argmax}} \left(\boldsymbol{r}_i^T \boldsymbol{s}[\ell; \widehat{w}_j]\right)_+^2 - \tau(\ell - \widetilde{\ell}_{(i+1)j})^2$ **end**

    if $i = I$ **then** $\widetilde{\ell}_{ij} \leftarrow \underset{\ell}{\operatorname{argmax}} \left(\boldsymbol{r}_i^T \boldsymbol{s}[\ell; \widehat{w}_j]\right)_+^2 - \tau(\ell - \widetilde{\ell}_{(i-1)j})^2$ **end**

    if $1 < i < I$ **then** $\widetilde{\ell}_{ij} \leftarrow$
    $\underset{\ell}{\operatorname{argmax}} \left(\boldsymbol{r}_i^T \boldsymbol{s}[\ell; \widehat{w}_j]\right)_+^2 - \tau(\ell - \widetilde{\ell}_{(i-1)j})^2 - \tau(\ell - \widetilde{\ell}_{(i+1)j})^2$ **end**

  end

while $\|\widetilde{\boldsymbol{L}}_{:j} - \widetilde{\boldsymbol{L}}_{:j}^{\text{old}}\|_2^2 / \|\widetilde{\boldsymbol{L}}_{:j}^{\text{old}}\|_2^2 < \xi$

delays related to each source are necessarily constant: $\ell_{ij} = \ell_{i'j}$, $\forall i, i'$. Therefore, the sparse vectors $\boldsymbol{\alpha}_{ij}$ $\forall i$ share a common support (see Fig. 5(c)). The estimation of $\boldsymbol{L}_{:j}$ in (22) becomes:

$$\widetilde{\boldsymbol{L}}_{:j} = [\widetilde{\ell}_j, \ldots, \widetilde{\ell}_j] \quad \text{with} \quad \widetilde{\ell}_j \leftarrow \underset{\ell}{\operatorname{argmax}} \sum_{i=1}^{I} \left(\boldsymbol{r}_i^T \boldsymbol{s}[\ell; \widehat{w}_j]\right)_+^2, \tag{23}$$

and can be obtained using an exhaustive search by testing the M possibilities for ℓ. This is very similar to the S-OMP algorithm [41] (which is a greedy algorithm

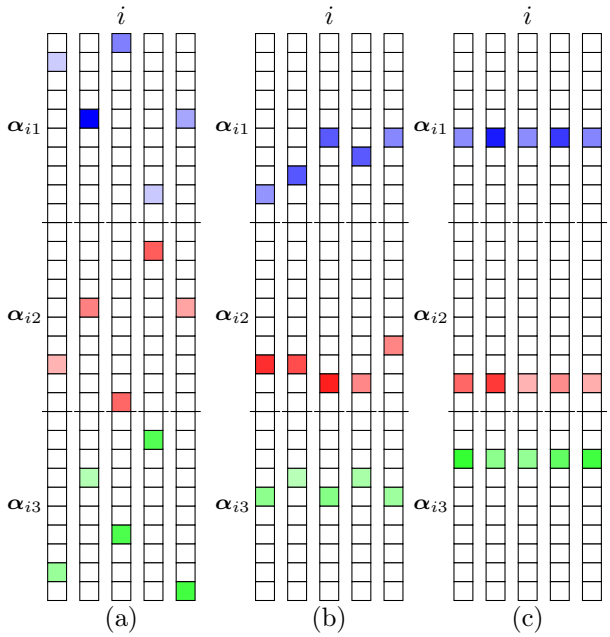


Figure 5: Examples of sparse model for $J = 3$ sources and $I = 6$ mixtures where the white and the colored squares respectively represent the zero and non-zero amplitudes. Each column corresponds to a mixture i, and is divided into J blocks representing the sources (each source is indicated with a unique color). (a) Case of no delay regularization. (b) Case of a slow-moving regularization. (c) Case of a very strong regularization, resulting in a simultaneous sparse approximation with constant support for all mixtures.

for sparse recovery of vectors having a common support) with the difference that the vectors $\boldsymbol{\alpha}_{ij}$ are 1-sparse.

5. Numerical Results

5.1. Comparison on Synthetic Mixtures

Algorithms 1 and 3 are compared with state-of-the-art methods that are able to separate highly correlated sources.

5.1.1. Evaluation of Non-Parameterized Methods

First of all, we evaluate two competing methods dealing with delayed source separation [27] and convolutive source separation [5]. The latter estimates the

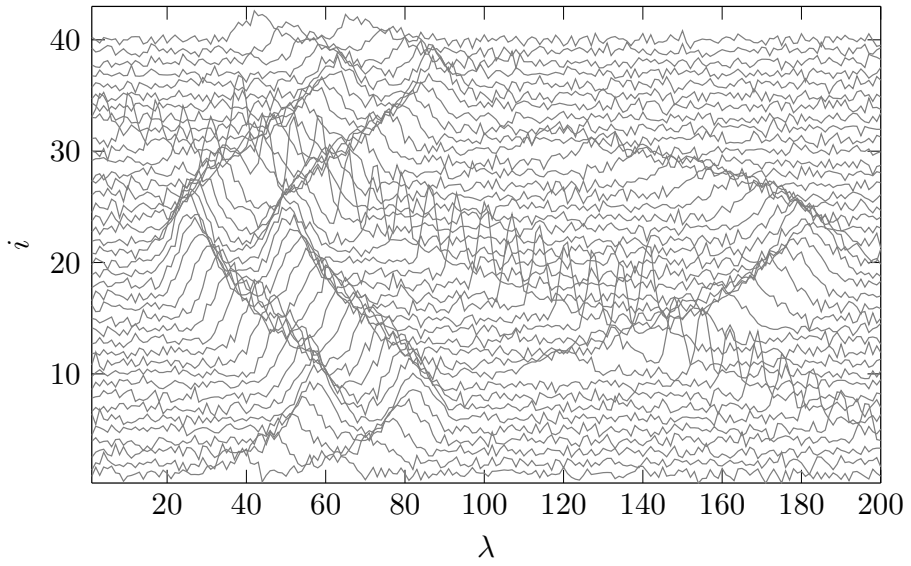
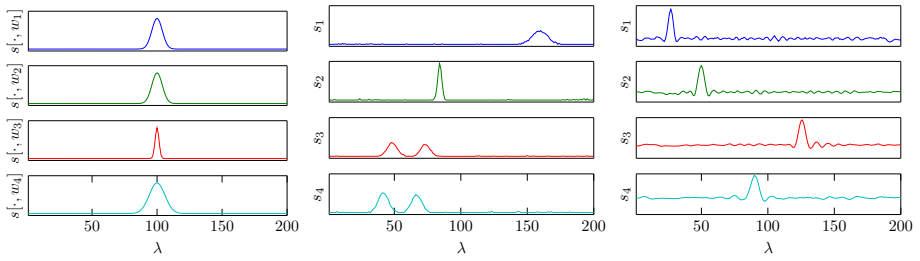


Figure 6: Synthetic data used for comparison: 40 mixtures with 4 sources and SNR = 15 dB.

impulse responses connecting each source to each mixture, whose location and intensity of the maximum values yield the delays and amplitudes.

The methods are evaluated on simulated data with $I = 40$ mixtures of $N = 200$ samples, $J = 4$ sources and SNR = 15 dB. The sources are Gaussian, *i.e.*, $s(\lambda; w_j) = \exp(-\lambda^2/2w_j^2)$. The amplitudes and delays of each Gaussian waveform are continuously generated from polynomials of degree 2, 3 or 4. Besides, the shape parameters, *i.e.*, the standard deviations of the Gaussian sources are set to $\boldsymbol{w}^* = \begin{bmatrix} 4 & 4 & 1.5 & 6 \end{bmatrix}^T$. The data are displayed in Fig. 6 and show noticeable behavior: two sources (around $\lambda = 50$) are very close and highly correlated (Gaussians with same width, equal to 4); there are several overlaps of the sources; and the number of sources per mixture is not constant.

The estimated sources with the non-parameterized methods [5, 27] are shown in Fig. 7, and the mixture reconstruction and estimated amplitudes and delays are displayed in Fig. 8. The mixture reconstruction obtained with the method of [27] is half satisfactory. On the one hand, the sources $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ are Gaussian-shaped and their amplitudes and delays roughly correspond to the ground truth. On the



(a) Ground truth sources (centered in 100) (b) Mörup *et. al* [27] (c) Castella *et. al* [5]

Figure 7: Ground truth and estimated sources with the methods of [27] and [5].

other hand, the sources $\boldsymbol{s}_3$ and $\boldsymbol{s}_4$ are bimodal (so they do not match with the ground truth) and actually fit the two ground-truth sources with similar shapes around $\lambda = 50$. Regarding the method of [5], the estimated sources present a dominant peak with small oscillations, so they are not close to the ground truth. Besides, the mixture reconstruction is very noisy and the parameter estimates do not completely match with the ground-truth. In conclusion, this example shows that the methods of [5, 27] are not able to deal with highly correlated sources, partly because they do not consider the highly informative knowledge of parametric sources.

5.1.2. Comparison with a Parameterized Method

We now compare the non-regularized and regularized methods (which respectively refer to Algorithms 1 and 3) with a Bayesian method [24] that explicitely exploits the knowledge of parametrized source shapes for decomposing a sequence of spectra. The methods are tested with the data displayed in Fig. 6. The three methods model the sources by Gaussians so the shape parameter is the Gaussian width. This is a positive scalar, therefore a trust-region reflective algorithm is used instead of the Levenberg-Marquardt algorithm because it is able to consider the positivity constraint. The stopping constant of the proposed methods is $\rho = 10^{-4}$; the delay sampling step is set to $\Delta = 0.2$; the regularization parameter is empirically tuned to $\tau = 1.5 \cdot 10^{-2}$. The ICM stopping constant is set to

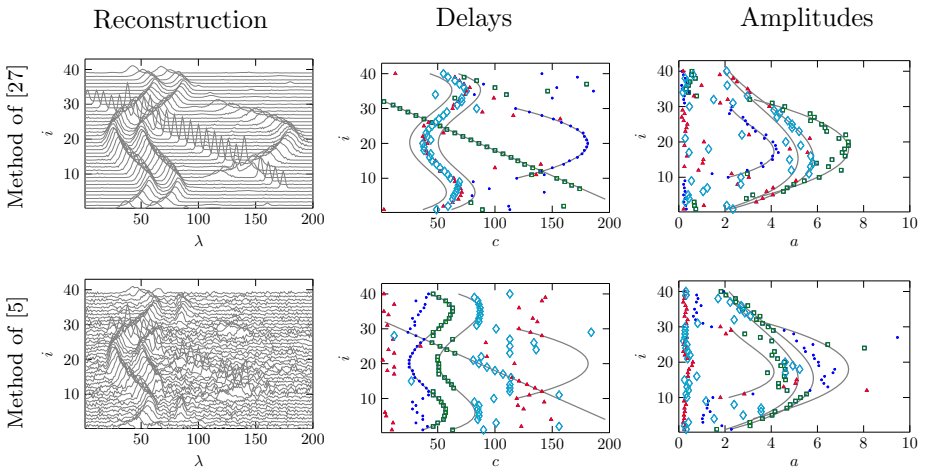


Figure 8: Results on the synthetic data of Fig. 6. The columns respectively the reconstructed mixtures, the estimated delays and the estimated amplitudes. Each source is represented with unique color and marker. The ground truth delays and amplitudes are plotted in gray lines.

$\xi = 10^{-4}$.

The results are displayed in Fig. 9. The reconstruction of the mixtures is equally good for the proposed methods and the method of [24]. This is confirmed by Tab. 1 which compares the methods in terms of *(i)* computation time; *(ii)* Mean Squared Error (MSE) defined as $\sum_i^I \|\boldsymbol{r}_i\|_2^2/(N \cdot I)$; *(iii)* amplitude, delay and shape error, defined as:

$$\frac{\|\boldsymbol{C}^* - \widehat{\boldsymbol{C}}\|_F^2}{\|\boldsymbol{C}^*\|_F^2}, \quad \frac{\|\boldsymbol{A}^* - \widehat{\boldsymbol{A}}\|_F^2}{\|\boldsymbol{A}^*\|_F^2} \quad \text{and} \quad \frac{\|\boldsymbol{w}^* - \widehat{\boldsymbol{w}}\|_2^2}{\|\boldsymbol{w}^*\|_2^2} \tag{24}$$

where $\|\cdot\|_F$ denotes the Frobenius norm and $\boldsymbol{C} = \Delta\boldsymbol{L}$ is the delay matrix for the proposed methods.

The estimated shape parameters are very close to the ground-truth, respectively $\widehat{\boldsymbol{w}} = \begin{bmatrix} 4.0 & 3.9 & 1.4 & 5.6 \end{bmatrix}^T$ and $\widehat{\boldsymbol{w}} = \begin{bmatrix} 3.9 & 4.0 & 1.4 & 5.9 \end{bmatrix}^T$ for the non-regularized and regularized methods. With the method of [24], the shape parameters can vary within a source, and a source does not necessarily appear in all the mixtures. But the variations remain small for the considered data so that we can consider the means of the estimated shape parameters: $\widehat{\boldsymbol{w}} = \begin{bmatrix} 3.9 & 3.9 & 1.5 & 6.0 \end{bmatrix}^T$ which is also close to the ground-truth.

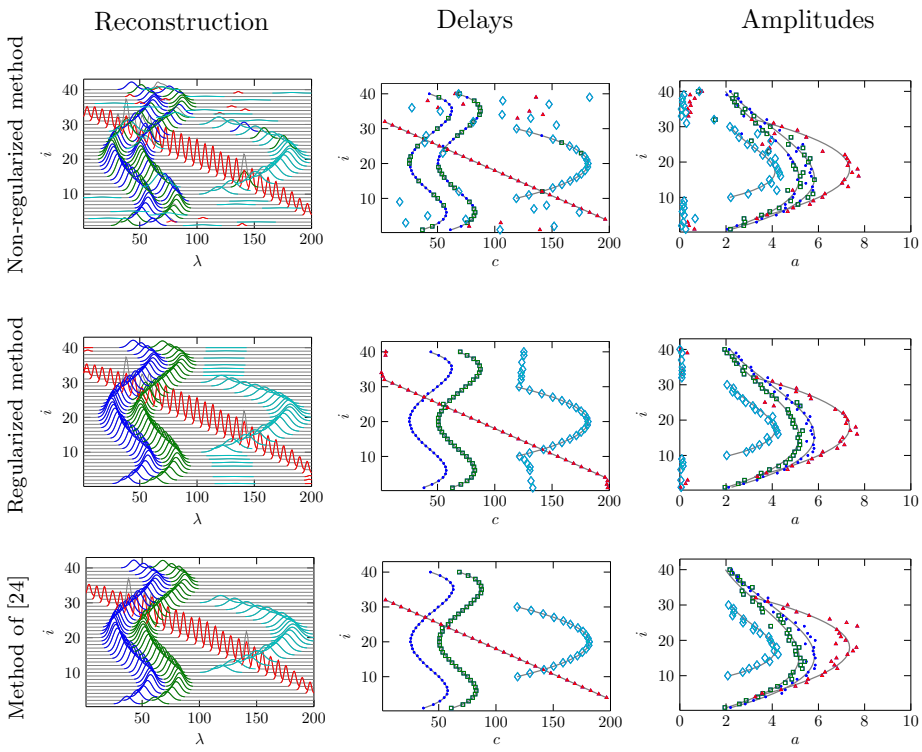


Figure 9: Results on the synthetic data of Fig. 6. The first column displays the reconstructed mixtures as well as the estimated sources. The second and third columns display the estimated delays and amplitudes. Each source is represented with unique color and marker. The ground truth delays and amplitudes are plotted in gray lines.

Considering the non-regularized method, the delay and amplitude estimates are not satisfactory despite the good reconstruction and source estimation. This can be explained by the fact that when some sources have the same shape parameter, the identification becomes ambiguous (see Section 3.2). On the contrary, the regularized method yields improved estimates as well as the method of [24]. These results are numerically validated in Tab. 1.

5.1.3. Conclusion

These simulations show that a parametrized method is of interest when dealing with highly correlated sources. The proposed regularization ensures to

	non-regularized	regularized	[24]
Time (s)	1.4	**0.5**	44.2
MSE	0.11	0.13	**0.09**
Shape error	$4.9 \cdot 10^{-3}$	$\mathbf{7.0 \cdot 10^{-4}}$	$1.4 \cdot 10^{-3}$
Delay error	$8.4 \cdot 10^{-2}$	$3.5 \cdot 10^{-5}$	$\mathbf{1.2 \cdot 10^{-5}}$
Amplitude error	$1.7 \cdot 10^{-2}$	$6.1 \cdot 10^{-3}$	$\mathbf{2.5 \cdot 10^{-3}}$

Table 1: Numerical performance of the compared methods on the synthetic mixtures plotted in Figure 6. The delay and amplitude error are defined in (24).

separate correctly the sources with very similar shape parameters. The proposed methods give accurate results within a very low computation time. On the contrary, the method of [24], while very effective, is 15 to 30 times slower than the proposed methods. Note however, that the latter model is more versatile since the shape parameter are allowed to vary within a source and additional priors on the shapes and amplitudes are considered.

5.2. Influence of the SNR

We now investigate the influence of the SNR on the performance of the proposed methods. A first example of synthetic mixtures with $I = 15$ and $J = 2$ is displayed in Fig. 10 to show that the proposed methods are robust to the SNR: the conclusion of section 5.1 remains, that is, both proposed methods yield good reconstruction and source quality whatever the SNR. Also, Fig. 10 shows the benefit of using a regularization on the delays. Recall that the permutation indetermination, which can be clearly noticed in this example, has no influence on the parameter estimation.

Furthermore, we perform statistical simulations to compare both proposed methods with the method of [24]. The methods of [5] and [27] are not tested since they do not yield accurate outputs. The simulations are set for SNR values varying between 0 and 30 dB. For each SNR, 100 datasets with Gaussian sources are generated, each with $I = 30$ mixtures, $J = 3$ sources and $N = 200$ samples. The shape parameters are chosen randomly between 0.5 and 5; the delays and

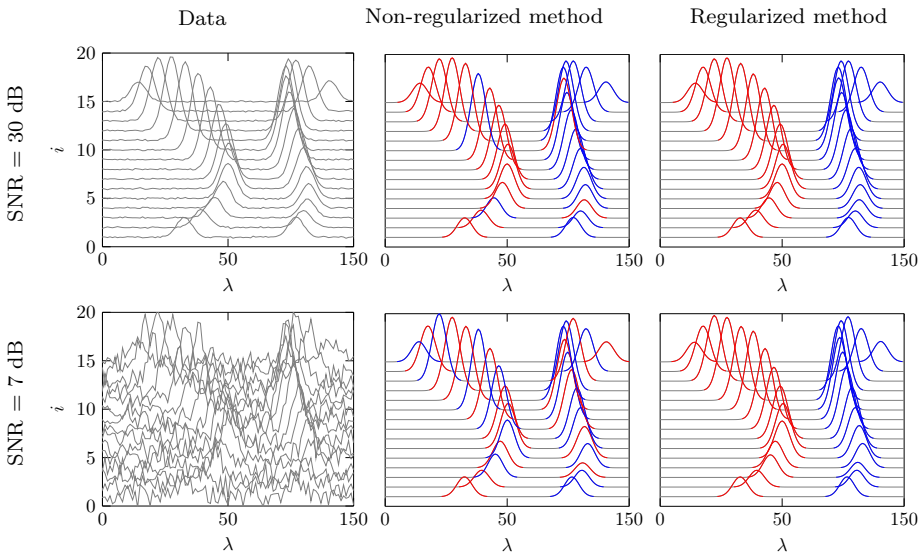
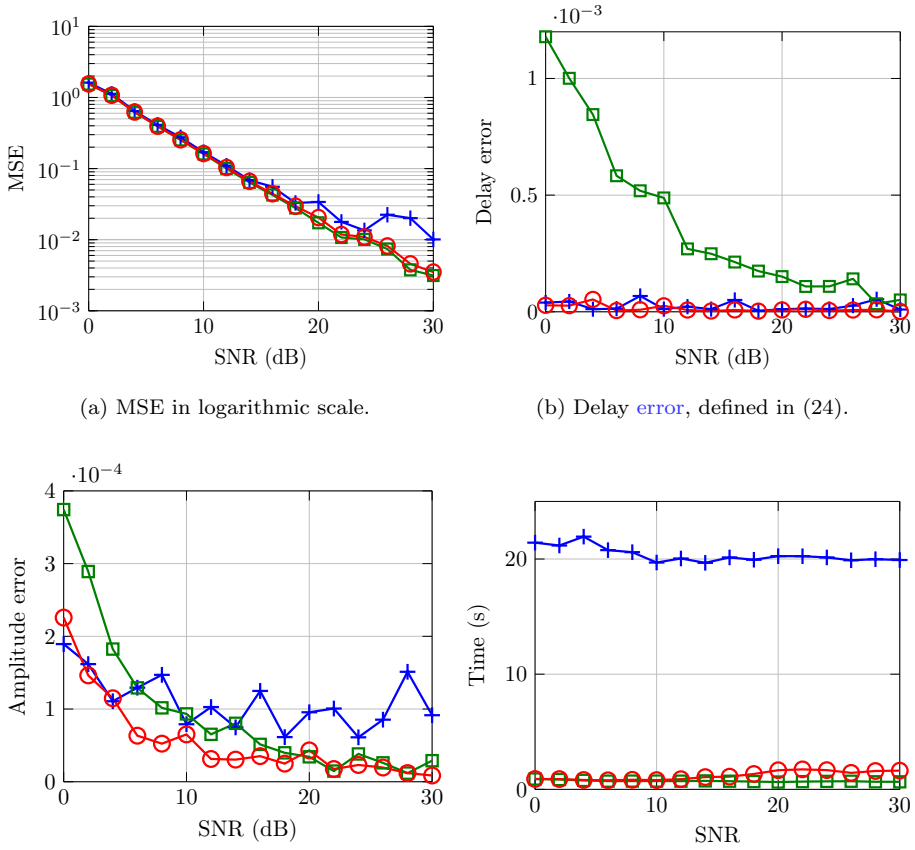


Figure 10: Synthetic data with 2 sources at two different SNR, and the related reconstruction and estimated sources with both proposed methods.

amplitudes are generated by polynomial functions with random coefficients and degrees varying between 2 and 4; the delay sampling step is equal to $\Delta = 0.2$; the stopping constant is set to $\rho = 10^{-4}$ for both proposed methods; the ICM stopping constant is set to $\xi = 10^{-4}$ and the regularization parameter is set to $\tau = 4 \cdot 10^{-3}$. The results are shown in Fig. 11 where, for each SNR, the average of the results obtained for the 100 generated datasets is plotted.

As expected, the MSE decreases as the SNR increases for the three methods (Fig. 11a), and it should be noticed that the variations are very similar. The delay and amplitude errors of the non-regularized method improve when the SNR increases but it is shown that the non-regularized method has lower performance than its competitors. On the contrary, the regularized method gives improved estimation of the amplitudes and delays. In addition, introducing a regularization on the delays yields estimates that are robust to the noise variation; this is not the case for the amplitudes for which no assumption of slow evolution is considered. The method of [24] behaves similar to the regularized method in terms of delay error, whilst its amplitude error is generally worse than the proposed methods



(a) MSE in logarithmic scale.

(b) Delay error, defined in (24).

(c) Amplitude error, defined in (24).

(d) Time (s).

Figure 11: Comparison of the non-regularized (green □) and the regularized (red ○) proposed methods with the method of [24] (blue +) with respect to the SNR.

(except for low SNR). Again, it is shown that the method of [24] is much slower than the proposed methods (approximately 20 times slower). Therefore, the regularized method is competitive with the method in [24] with much lower computation time.

5.3. Results on Real Photoelectron Data

Time-resolved photoelectron spectroscopy [37] is an experimental tool which allows to study the energy relaxation occurring after absorption of a photon by

an isolated molecule, atom or a blend of both. The energy relaxation is probed by ionization of the excited system with another delayed photon, thereby ejecting a so-called photoelectron. The distribution of the photoelectrons according to their energy is measured at different times to get a temporal sequence of photoelectron spectra and is given by the estimated source characteristics (amplitudes, delays, shapes). The delays $\boldsymbol{C}$ indicate the energy from which photoemission is occurring and the area under each source, which depends on both the amplitudes $\boldsymbol{A}$ and shapes $\boldsymbol{w}$, corresponds to the relative number of emitted photoelectrons at the corresponding energy. The experiment presented in this paper studies the relaxation of an atom of barium [23, 24].

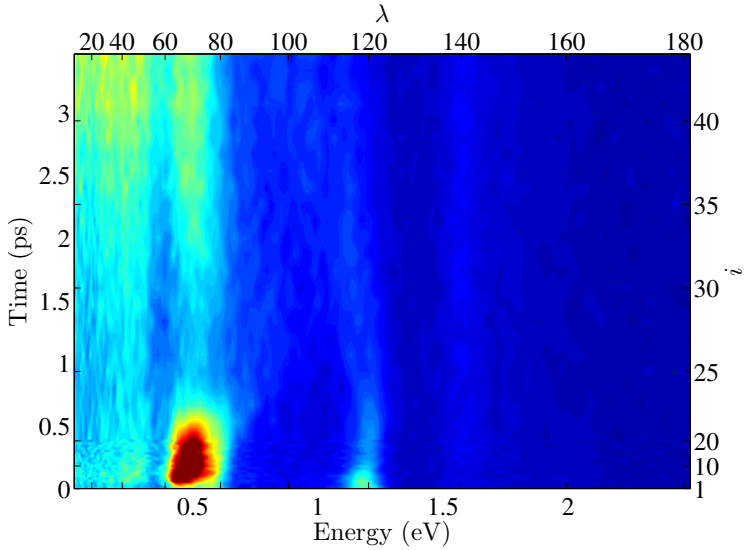
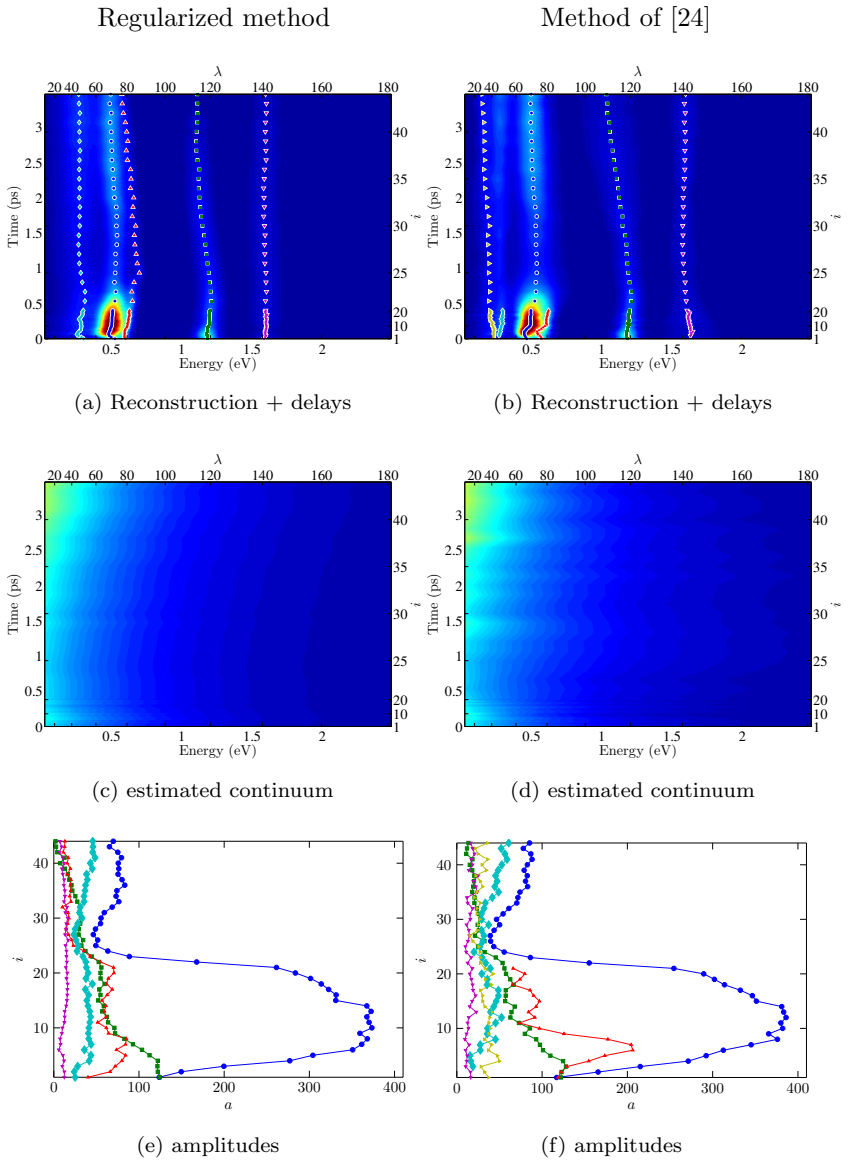


Figure 12: Photoelectron spectra ($I = 44$ mixtures).

The regularized method was applied to the sequence of photoelectron spectra shown in Fig. 12: the goal is to determine how the energy, intensity, and width of the peaks (*i.e.*, delays, amplitudes and shapes of the sources) evolve through the sequence, indicating the temporal changes undergone by the studied system. The sequence gathers $I = 44$ spectra (covering a duration of 3.47 ps), each of $N = 181$ samples (from 0.02 eV to 2.52 eV). The sources are modeled by Gaussian



(a) Reconstruction + delays

(b) Reconstruction + delays

(c) estimated continuum

(d) estimated continuum

(e) amplitudes

(f) amplitudes

Figure 13: The reconstruction and the estimated continuum, delays and amplitudes using the regularized method (first column) and the method of [24] (second column). The estimated shape parameters of regularized method are $\widehat{\boldsymbol{w}} = [0.057, 0.049, 0.025, 0.09, 0.06]^T$ and for the method of [24] after averaging the estimates for each source are $\widehat{\boldsymbol{w}} = [0.066, 0.064, 0.039, 0.037, 0.071, 0.05]^T$.

functions, which is the usual model for the peaks in photoelectron spectroscopy[1]. The grid sampling step is set to $\Delta = 5 \cdot 10^{-4}$ and the regularization term is set to $\tau = 2 \cdot 10^{-5}$. The stopping constant ρ and the ICM stopping constant are respectively set to $\rho = 10^{-4}$ and $\xi = 10^{-4}$. Besides, the photoelectron spectra include a background that must be estimated and removed from the spectra. The background in each mixture i is modeled as an exponential of the form $\alpha_i \exp(-\lambda/\beta)$. An additional step is added to the ALS scheme in Algorithm 3 to estimate the new unknowns (the I weights α_i and a single β value) by using the Levenberg-Marquardt algorithm.

The results obtained with the regularized method and the method of [24] are presented in Fig. 13: both approaches appear roughly similar concerning delay and shape error. However, differences can be observed: *(i)* the estimated continuum provided is much smoother for the regularized method, which was a source of noise in the method [24]; *(ii)* the regularized method rather improves the shape of the main band by addition of nearby new sources than focusing on the low intensity energy ones as seen for method [24]. Such behavior looks reasonable according to the shape of the decay simulated [22]. *(iii)* The method of [24] is able to deal with a varying number of peaks through the mixtures, whereas in the proposed method the J sources are present in all the mixtures. Nevertheless, small amplitudes can be cancelled by applying a threshold. *(iv)* Last but not least, the proposed method is about 560 times faster (7.5 seconds for the regularized method versus 70 minutes for the method of [24]). In conclusion, the proposed method appears to be both effective and efficient.

Software

The Matlab codes associated with this article will be freely available in case of publication.

[1] Note that choosing a model not "consistent" with the shape of real peaks may yield an increase of the number of estimated sources.

Acknowledgment

This work was supported by the DSIM project (grant ANR-14-CE27-0005).

6. Conclusion

This paper studies the delayed source separation when source signals are supposed to be parameterized, which is a valid assumption in many spectroscopy applications. An ALS scheme is proposed to alternatively estimate the shape parameters and the amplitudes and delays. For the challenging delay parameter estimation step, we propose a greedy strategy using parametric dictionaries whereas the sparse representation is structured to respect the mixing model. We present two algorithms that follow the same ALS scheme: in the first, only the data-fit criterion is considered whilst in the second an additional regularization term is added to promote slow delay evolution within each source. The proposed methods outperform the state-of-the-art delayed source separation methods when sources are highly correlated. Furthermore, it is as effective as the best competitors with much lower computation time. Results on real data confirm the effectiveness and the efficiency of the proposed methods. Future works will be dedicated to studying the benefit of a warm-start initialization of $\boldsymbol{A}$ and $\boldsymbol{L}$ in Algorithms 2 and 4, so as to benefit from the estimated parameters of previous ALS iterations. Although this idea seems natural, the update of delay parameters would then require to design discrete search algorithms having a more complex structure than greedy search algorithms. Another perspective of this work would be to estimate the number of sources: strategies such as Akaike information criterion or Bayesian information criterion [36] could be a first step to explore. Finally, a third perspective would be to deal with sources whose shape parameters evolve through the mixtures.

References

[1] Austin, C. D., Moses, R. L., Ash, J. N., Ertin, E., 2010. On the relation between sparse reconstruction and parameter estimation with model order selection. IEEE J. Sel. Top. Signal Process. 4 (3), 560–570.

[2] Besag, J., 1986. On the statistical analysis of dirty pictures. J. R. Statist. Soc. B 48 (3), 259–302.

[3] Bourguignon, S., Mary, D., Slezak, É., 2011. Restoration of astrophysical spectra with sparsity constraints: Models and algorithms. IEEE J. Sel. Top. Signal Process. 5 (5), 1002–1013.

[4] Bruckstein, A. M., Elad, M., Zibulevsky, M., 2008. On the uniqueness of nonnegative sparse solutions to underdetermined systems of equations. IEEE Trans. Inf. Theory 54 (11), 4813–4820.

[5] Castella, M., Rhioui, S., Moreau, E., Pesquet, J., 2007. Quadratic higher order criteria for iterative blind separation of a MIMO convolutive mixture of sources. IEEE Trans. Signal Process. 55 (1), 218–232.

[6] Chabriel, G., Barrère, J., 2006. An instantaneous formulation of mixtures for blind separation of propagating waves. IEEE Trans. Signal Process. 54 (1), 49–58.

[7] Comon, P., Jutten, C., 2010. Handbook of Blind Source Separation: Independent Component Analysis and Applications. Academic Press.

[8] Duan, J., 2010. Restoration and separation of piecewise polynomial signals. Application to atomic force microscopy. Phd thesis, Université Henri Poincaré, Nancy, France.

[9] Ekanadham, C., Tranchina, D., Simoncelli, E. P., 2011. Recovery of sparse translation-invariant signals with continuous basis pursuit. IEEE Trans. Signal Process. 59 (10), 4735–4744.

[10] Fyhn, K., Jensenand, S. H., Duarte, M. F., 2015. Compressive parameter estimation for sparse translation-invariant signals using polar interpolation. IEEE Trans. Signal Process. 63 (4), 870–881.

[11] Gilmore, G., Wyse, R. F., Kuijken, K., 1989. Kinematics, chemistry, and structure of the galaxy. Annu. Rev. Astron. Astrophys 27 (1), 555–627.

[12] Gloaguen, E., Mestdagh, J.-M., Poisson, L., Lepetit, F., Visticot, J.-P., Soep, B., Coroiu, M., Eppink, A., Parker, D. H., 2005. Experimental evidence for ultrafast electronic relaxation in molecules, mediated by diffuse states. J. Am. Chem. Soc. 127 (47), 16529–16534.

[13] Harshman, R. A., Hong, S., Lundy, M. E., 2003. Shifted factor analysis – Part I: Models and properties. J. Chemometrics 17 (7), 363–378.

[14] Hollas, J. M., 2004. Modern spectroscopy. John Wiley & Sons.

[15] Hong, S., Harshman, R. A., 2003. Shifted factor analysis – Part II: Algorithms. J. Chemometrics 17 (7), 379–388.

[16] Jiang, N., Farina, D., 2011. Covariance and time-scale methods for blind separation of delayed sources. IEEE Trans. Biomed. Eng. 58 (3), 550–556.

[17] Kawamoto, M., Inouye, Y., 2003. Blind deconvolution of MIMO-FIR systems with colored inputs using second-order statistics. IEICE Trans. Fundamentals 86 (3), 597–604.

[18] Kowalski, M., Siedenburg, K., Dörfler, M., 2013. Social sparsity! Neighborhood systems enrich structured shrinkage operators. IEEE Trans. Signal Process. 61 (10), 2498–2511.

[19] Lawson, C. L., Hanson, R. J., 1995. Solving least squares problems, Society for Industrial and Applied Mathematics Edition. Vol. 15. SIAM.

[20] Malinowski, E. R., 2002. Factor analysis in chemistry. Wiley.

[21] Marquardt, D. W., 1963. An algorithm for least-squares estimation of nonlinear parameters. SIAM J. Appl. Mathematics 11 (2), 431–441.

[22] Masson, A., Heitz, M., Mestdagh, J., Gaveau, M., Poisson, L., Spiegelman, F., 2014. Coupled electronic and structural relaxation pathways in the postexcitation dynamics of Rydberg states of BaAr N Clusters. Phys. Rev. Lett. 113 (12), 123005.

[23] Masson, A., Poisson, L., Gaveau, M., Soep, B., Mestdagh, J., Mazet, V., Spiegelman, F., 2010. Dynamics of highly excited barium atoms deposited on large argon clusters. I. General trends. J. Chem. Phys 133 (5), 054307.

[24] Mazet, V., Faisan, S., Awali, S., Gaveau, M., Poisson, L., 2015. Unsupervised joint decomposition of a spectroscopic signal sequence. Signal Process. 109, 193–205.

[25] Mortada, H., Mazet, V., Soussen, C., Collet, C., 2017. Separation of delayed parameterized sources. In: Proc. Eur. Sig. Proc. Conf. pp. 1080–1084.

[26] Mørup, M., Madsen, K. H., Hansen, L. K., 2007. Shifted independent component analysis. In: Proc. ICA. pp. 89–96.

[27] Mørup, M., Madsen, K. H., Hansen, L. K., 2007. Shifted non-negative matrix factorization. In: Proc. MLSP. pp. 139–144.

[28] Moussaoui, S., Brie, D., Mohammad-Djafari, A., Carteret, C., 2006. Separation of non-negative mixture of non-negative sources using a Bayesian approach and MCMC sampling. IEEE Trans. Signal Process. 54 (11), 4133–4145.

[29] Nion, D., Vandewoestyne, B., Vanaverbeke, S., Van D. A., K., De Gersem, H., De Lathauwer, L., 2010. A time-frequency technique for blind separation and localization of pure delayed sources. In: Proc. LVA/ICA. pp. 546–554.

[30] O'Grady, P. D., Pearlmutter, B. A., Rickard, S. T., 2005. Survey of sparse and non-sparse methods in source separation. Int. J. Imag. Syst. Tech. 15 (1), 18–33.

[31] Omlor, L., Giese, M. A., 2011. Anechoic blind source separation using Wigner marginals. J. Mach. Learn. Res. 12, 1111–1148.

[32] Ouedraogo, W. S. B., Nicolas, B., Oudompheng, B., Mars, J. I., Jutten, C., 2014. A frequency method for blind separation of an anechoic mixture. In: Proc. Eur. Sig. Proc. Conf. pp. 521–525.

[33] Papyan, V., Romano, Y., Sulam, J., Elad, M., 2018. Theoretical foundations of deep learning via sparse representations: A multilayer sparse model and its connection to convolutional neural networks. IEEE Signal Process. Mag. 35 (4), 72–89.

[34] Puigt, M., Deville, Y., 2005. Time–frequency ratio-based blind separation methods for attenuated and time-delayed sources. Mech. Syst. Signal. Process. 19 (6), 1348–1379.

[35] Rivet, B., Girin, L., Jutten, C., 2005. Solving the indeterminations of blind source separation of convolutive speech mixtures. In: Proc. IEEE ICASSP. pp. 533–536.

[36] Stoica, P., Selen, Y., 2004. Model-order selection: a review of information criterion rules. IEEE Signal Process. Mag. 21 (4), 36–47.

[37] Stolow, A., Bragg, A. E., Neumark, D. M., 2004. Femtosecond time-resolved photoelectron spectroscopy. Chem. Rev 104 (4), 1719–1758.

[38] Trigano, T., Sepulcre, Y., Ritov, Y., 2017. Sparse reconstruction algorithm for nonhomogeneous counting rate estimation. IEEE Trans. Signal Process. 65 (2), 372–385.

[39] Trigano, T., Shevtsov, I., Luengo, D., 2017. Cosa: An accelerated ISTA algorithm for dictionaries based on translated waveforms. Signal Process. 139, 131–135.

[40] Tropp, J. A., 2004. Greed is good: Algorithmic results for sparse approximation. IEEE Trans. Inf. Theory 50 (10), 2231–2242.

[41] Tropp, J. A., Gilbert, A. C., Strauss, M. J., 2006. Algorithms for simultaneous sparse approximation. part I: Greedy pursuit. Signal Process. 86 (3), 572–588.

[42] Vaswani, N., Zhan, J., 2016. Recursive recovery of sparse signal sequences from compressive measurements: A review. IEEE Trans. Signal Process. 64 (13), 3523–3549.

[43] Villeneuve, E., Carfantan, H., 2014. Nonlinear deconvolution of hyperspectral data with MCMC for studying the kinematics of galaxies. IEEE Trans. Image Process. 23 (10), 4322–4335.

[44] Vimond, M., 2010. Efficient estimation for a subclass of shape invariant models. Ann. Stat. 38 (3), 1885–1912.

[45] Yaghoobi, M., Daudet, L., Davies, M. E., 2009. Parametric dictionary design for sparse coding. IEEE Trans. Signal Process. 57 (12), 4800–4810.

[46] Yilmaz, Ö., Rickard, S., 2004. Blind separation of speech mixtures via time-frequency masking. IEEE Trans. Signal Process. 52 (7), 1830–1847.