Underdetermined Blind Separation of Delayed Sound Sources in the Frequency Domain

Pau Bofill [1]

Dept. d'Arquitectura de Computadors, UPC
Campus Nord, Mòdul D6
Jordi Girona 1-3
08034 Barcelona, Spain
pau@ac.upc.es

Abstract

This paper focuses on the blind separation of n sources from m mixtures when the underlying system is *underdetermined* (i.e., when $m < n$), as applied to mixtures with only attenuations and delays. The separation is realized in the frequency domain where, at least for speech and music signals, the representation is *sparse*. The matrix of relative attenuations is infered by angular clustering of the *magnitude* of the input, yielding a coarse partition of the data into their nearest sources. For each partition, then, the differential delay is infered by scattering the *real* and *imaginary* components until the cluster reappears. Given the attenuation and delay matrices, the sources are infered at each frequency bin by assuming that *the magnitude* of the spectral coefficients is laplacian distributed, which leads to the minimization of the sum of magnitudes, subject to the mixing equations. The resulting problem is an instance of second-order cone programming.

The approach is illustrated here for $m = 2$ sensors and n up to six speech and music sources, using synthetic mixtures simulating real acoustic scenarios without reverberation.

Key words: Blind source separation, Underdetermined, Sparsity, Potential-function clustering, Cluster maximization, A posteriori likelihood, Second-order cone programming

[1] Research developped during the author's visting scholarship at CARTAH, University of Washington

Preprint submitted to Neurocomputing, special issue ICA and BSS, 2 March 2001

1 Problem Formulation

This work deals with the inference or separation of n sources using the signals from m sensors, assuming that only attenuation and delay are present in the mixture. The assumed mixture model is thus the following:

$$\begin{aligned} x_1(t) = &\ \beta_{11}s_1(t-\tau_{11}) + \ldots + \beta_{1n}s_n(t-\tau_{1n}) \\ \ldots& \\ x_m(t) = &\beta_{m1}s_1(t-\tau_{m1}) + \ldots + \beta_{mn}s_n(t-\tau_{m1}), \end{aligned} \tag{1}$$

where $t = 1, \ldots, P$ is the discrete time, $x_i(t)$ are the sensor signals, $s_j(t)$ are the sources, and β_{ij} and τ_{ij} are the attenuation and delay from source j to sensor i, respectively. The problem of *blind source separation* [4] consists of infering the sources from the sensor signals when sources, attenuation and delay are *unknown*. The scope of this work is the underdetermined case (i.e., $m < n$), and the algorithms and experiments described here are particularized for the case $m = 2$.

In order to tackle such a problem some assumption must be made. The usual approach is to assume the statistical independence of the source components, or *independent component analysis* [3,6]. In this work we assume further that the source components have a Laplacian pdf, which means that their representation is *sparse* [8] (only a few components are significantly different from zero).

Yet, for speech and music applications it has been shown experimentally [2,13] that signals in the time domain are *not* sparse, whereas their frequency domain representations are sparse. Thus, by taking Discrete Fourier Transform DFT of the above system with a time window of size K, we obtain

$$\begin{aligned} X_1^k = &\ \beta_{11}e^{-j2\pi k\tau_{11}/K}S_1^k + \ldots + \beta_{1n}e^{-j2\pi k\tau_{1n}/K}S_n^k \\ \ldots& \\ X_m^k = &\beta_{m1}e^{-j2\pi k\tau_{m1}/K}S_1^k + \ldots + \beta_{mn}e^{-j2\pi k\tau_{mn}/K}S_n^k, \end{aligned}$$

with $k = 0, \ldots, K/2$ the discrete frequency.

In matrix notation

$$\mathbf{X}^k = \mathbf{Z}^k\mathbf{S}^k, \tag{2}$$

where the data point $\mathbf{X}^k$, from the sensor signals, is an m-channel column vector, the source vector $\mathbf{S}^k$ has n unknown components, and the $m\times n$ mixing matrix $\mathbf{Z}^k$ is

$$\mathbf{Z}^k = [\beta_{ij} e^{-j2\pi k \tau_{ij}/K}] = \mathbf{B} \star e^{-j2\pi k \mathbf{T}/K}, \tag{3}$$

with $\mathbf{B} = [\beta_{ij}]$ the unknown attenuation matrix, $\mathbf{T} = [\tau_{ij}]$ the unknown delay matrix, and $\star$ denoting the component to component product. Notice that while $\mathbf{Z}^k$ takes $1 + K/2$ different values, they all depend on the same $\mathbf{B}$ and $\mathbf{T}$.

Since the S_j are complex, a previous approach has been to assume that both the real part $Re(S_j)$ and the imaginary part $Im(S_j)$ are sparse [14,2,13]. Instead, the approach here will be to assume that the *magnitude* $Mag(S_j)$ of the source components is sparse. Several approaches exist in the literature for the blind decomposition of delayed *and* convoluted sources for the determined case (see for instance [11,12,**?**,9], among others).

2 Infering Attenuations, Delays and Sources

2.1 A Clustering Procedure for the Inference of the Attenuation Matrix

When the distribution is laplacian, there is a low probability that the energy of a particular source is large. Therefore, when a data point is large the most likely decomposition is to assume that it belongs to a single source, because the probability of two sources being large simultaneously is much lower. Thus, the reason why sparse representations lend themselves to good separability is because most of the significant data belong to a single source. In other words, sparse representations are separable because sources are to some extend *disjoint*.

Instantaneous mixture of real values

Let's first consider an instantaneous mixture (attenuation only)

$$\mathbf{X}^k = \mathbf{B}\mathbf{S}^k,$$

and let's assume for the time being that $\mathbf{X}^k$ and $\mathbf{S}^k$ are real. Expanding along the columns of $\mathbf{B}$ we have

$$\mathbf{X}^k = \mathbf{B}_1 S_1^k + \ldots + \mathbf{B}_n S_n^k. \tag{4}$$

Since both $\mathbf{B}_j$ and S_j^k are unknown, it is impossible to recover the scale (we could reciprocally rescale both and still get the same product). In other words, we can only know the *relative* attenuation from one source to the different sensors. For convenience and without loss of generality, then, we will assume hereafter that the $\mathbf{B}_j$ are of unit length.

Let's now assume that for data point $\mathbf{X}^k$ only a particular source j is significanly different from zero. In that case from (4) we obtain

$$\mathbf{X}^k = \mathbf{B}_j S_j^k + \varepsilon, \tag{5}$$

where ε is the perturbation due to the remaining sources. That is, $\mathbf{X}^k$ is nearly proportional to $\mathbf{B}_j$. In $\Re^m$, in mixture space, $\mathbf{X}^k$ lies in the direction of $\mathbf{B}_j$ and has length S_j^k.

Extending to all k, although most of the data will be close to the origin, those of significant length will tend to *cluster* along the directions of the different $\mathbf{B}_j$'s. Thus, we call the $\mathbf{B}_j$ the *director vectors* of the sources. Finding the prominent orientation of those clusters ammounts to infering the attenuation matrix $\mathbf{B}$.

In this work, we use the potential-function procedure developped in [2], designed for $m = 2$ (two sensors), but general purpose clustering algorithms have also been proved usefull for $m \geq 2$ [13].

In $\Re^2$ directions can be specified by a single angle θ, thus let l^k and θ^k be the angle and length of $\mathbf{X}^k$, respectively. Let α be the relative angle between an arbitrary direction and θ^k. We choose a *basis function* ϕ around $\mathbf{X}^k$ as a triangular function of the local angle α,

$$\begin{aligned} \phi(\alpha) &= 1 - \frac{\alpha}{\pi/4}, \quad \text{for} \quad |\alpha| < \pi/4, \\ &= 0 \quad \text{elsewhere,} \end{aligned}$$

and we define a global *potential function* Φ over the absolute angle θ as

$$\Phi(\theta) = \sum_k l^k \phi(\lambda(\theta - \theta^k)), \tag{6}$$

with λ a parameter to adjust the resolution, and l^k a weight to put more emphasis on the more reliable data (for equal perturbations, angular errors will be smaller for data points that are farther from the origin). In practice, $Phi(\theta)$ is computed over a sampling grid with G points.

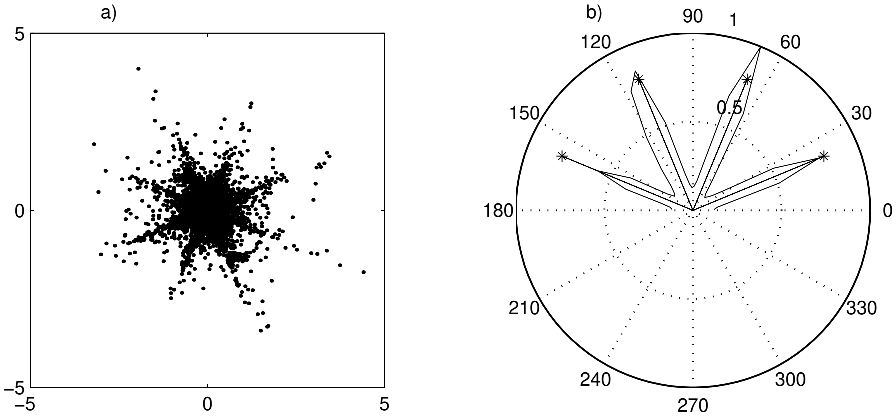

Fig. 1. (a) Two-sensor scatter plot of the instantaneous mixture of six flute notes, toghether with (b) the corresponding plot of the potential function $\Phi(\theta)$, using the real and imaginary parts of the DFT.

When the clustering is good, the maxima θ_j^* of the potential function define the desired directions, and by taking $\hat{\beta}_{1j} = cos(\theta_j^*)$ and $\hat{\beta}_{2j} = sin(\theta_j^*)$ (i.e., unit length) we infer the estimated vector $\hat{\mathbf{B}}_j$ of *relative* attenuations. Extending to all j, the estimated attenuation matrix is $\hat{\mathbf{B}}$.

Figure 1 shows a scatter plot (the plot of X_2^k vs X_1^k) of the real and imaginary coefficients of the mixtures of six flute notes, toghether with a plot of the corresponding potential function, with all point projected onto the upper semiplane. Stars show the original director vectors, and radial axes show the result of the algorithm. Notice that the estimates are exact.

The above procedure is roughly equivalent to finding the maxima of a weighted histogram, filtered with a triangular window.

Complex mixture of complex values

For the complex-valued case our approach is to take magnitudes on both sides of equation (4)

$$Mag(\mathbf{X}^k) = Mag(\mathbf{Z}_1 S_1^k + \ldots + \mathbf{Z}_n S_n^k).$$

which, assuming again that only one source is significant, leads to

$$Mag(\mathbf{X}^k) = \mathbf{B}_j Mag(S_j^k) + \varepsilon,$$

where ε includes now all errors incurred in the process. This equation is an instance of equation (5), and we use the same procedure ($Mag(X_2^k)$ vs $Mag(X_1^k)$) in order to estimate $\hat{\mathbf{B}}$.

Now, all data points whose direction is close to director vector $\hat{\mathbf{B}}_j$ can be assumed to "belong mostly to" source j, and we have a *partition* of the input data in terms of their nearest source.

2.2 Inference of the Delay Matrix by Maximizing Clustering over Phase

Following the discussion above, let's try applying the same strategy to the real (and imaginary) parts[2],

$$Re(\mathbf{X}^k) = Re(\mathbf{Z}_1 S_1^k + \ldots + \mathbf{Z}_n S_n^k).$$

Assuming a single source

$$Re(\mathbf{X}^k) = Re(\mathbf{Z}_j S_j^k) + \varepsilon, \tag{7}$$

and developping for the real part, we obtain

$$Re(\mathbf{X}^k) = Re(\mathbf{Z}_j) Re(S_j^k) - Im(\mathbf{Z}_j^k) Im(S_j^k) + \varepsilon.$$

That is, in the general case, $Re(\mathbf{X})$ is *not* proportional to either the real or imaginary parts of S_j. What is worse, different data belonging to the same source will align in different directions. The scatter plot of the real parts, then, will *not* show the same clusters as before.

Let's now explore the role of delays on the mixture model. Taking equation (2) and (3) and expanding again by columns, we have

$$\mathbf{X}^k = \mathbf{B}_1 \star e^{-j2\pi k \mathbf{T}_1/K} S_1^k + \ldots + \mathbf{B}_n \star e^{-j2\pi k \mathbf{T}_n/K} S_n^k.$$

Vector $\mathbf{B}_j$ has already been fixed to unit length, thus arbitrarily setting the magnitude of S_j^k, but since both $\mathbf{T}_j$ and S_j^k are unknown, the phase is undetermined as well, up to an additive delay. In other words, we can only estimate the *differential* delay from one source to the different sensors. For convenience and without loss of generality, we will assume hereafter that the average value of the components of $\mathbf{T}_j$ is zero and, in doing so, we are also assuming that the average delay from the source to the sensors is implicit in the phase of S_j.

Particularizing for $m = 2$ (two channels) the above assumption implies that $\tau_{1j} = -\tau_{2j} = \delta_j/2$, with δ_j the differential delay. Let's now suppose that $\delta_j = 0$

[2] Overall this section the discussion refers to real parts, but in practice we use both real and imaginary coefficients.

(the sensor channels are in phase). Now $\mathbf{T}_j = [0]$ and $e^{-j2\pi k \mathbf{T}_j/K} = 1$, and from equation (7) we have

$$Re(\mathbf{X}^k) = \mathbf{B}_j Re(S_j^k) + \varepsilon.$$

All data points belonging to source j will follow the same equation, yielding the corresponding cluster, with the same director vector $\mathbf{B}_j$ as before.

When $\delta_j \neq 0$ and $\mathbf{T}_j = [\delta_j/2 \quad -\delta_j/2]^t]^t$, we apply a phase correction to the channels in order to induce an external differential delay ρ and $\mathbf{R} = [\rho/2 \quad -\rho/2]^t]^t$, yielding

$$\mathbf{X}^k \star e^{-j2\pi k \mathbf{R}/K} = \mathbf{B}_j \star e^{-j2\pi k(\mathbf{T}_j + \mathbf{R})/K} + \varepsilon.$$

When $\rho = -\delta$ the right hand side phases cancel and we are in the same situation as above. The cluster for source j has been recovered and $-\rho$ is our estimate of the delay.

Based on the above, our approach consists of taking the partition of the data "belonging mostly to" source j (obtained as a result of the previous section), and iteratively rectifying the delay until the cluster reappears. At each iteration, we evaluate $\Phi(\theta_j^*)$ (evaluating the potential function in the direction of source j). The value ρ^* that maximizes $\Phi(\theta)$ is then the estimated delay, and we set $\hat{\mathbf{T}}_j = -\mathbf{R}^*$. Repeating the procedure for all sources $\hat{\mathbf{T}}$ is obtained, and toghether with with $\hat{\mathbf{B}}$ from the previous section, the estimate $\hat{\mathbf{Z}}^k = \hat{\mathbf{B}} \star e^{-j2\pi k \hat{\mathbf{T}}/K}$ of the mixing matrix is completed.

Figure 2 shows a plot of $\Phi(\theta_j^*)$ vs ρ for a violin and a singer (the VoiceOnViolin data set described in section 3). The estimates are again exact.

The described procedure is extensible to $m > 2$ by estimating the differential delays between every pair of channels.

2.3 Infering the Source Components by Second-order Cone Programming

For $n = m$, after finding $\mathbf{Z}^k$ the problem has been solved, assuming that $\mathbf{Z}^k$ is invertible, of course. Yet, for $n > m$ the system is still underdetermined, and we need some additional criteria.

In this work we use a maximum a posteriori approach by assuming (a) the statistical independence of the components of the sources, (b) uniformly distributed phases, (c) a lapacian distribution of the magnitudes with (d) equal

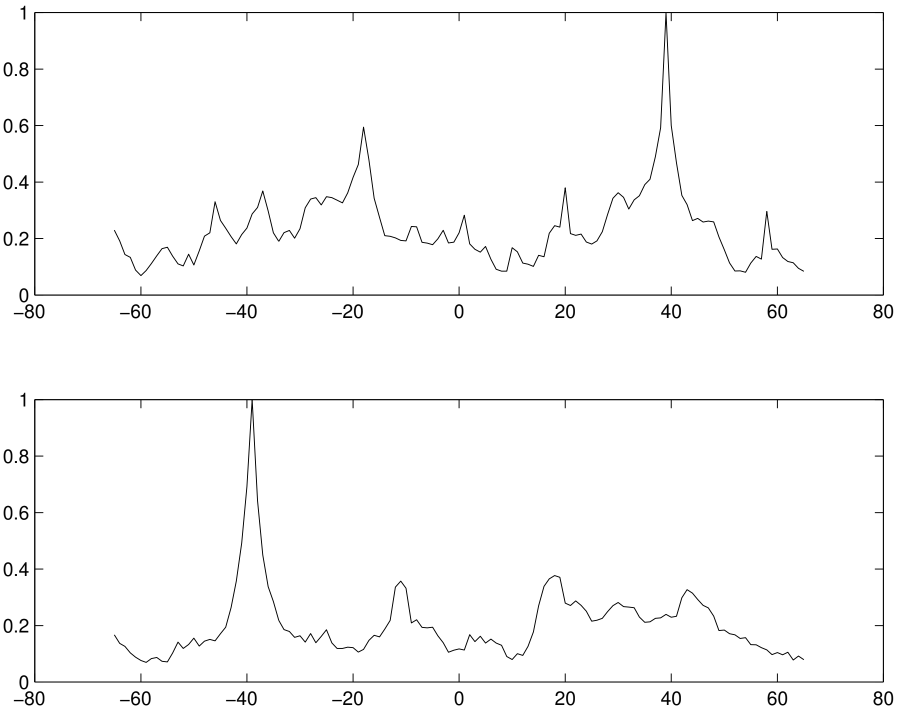

Fig. 2. A plot of $\Phi(\theta_j^*)$ vs ρ for two signals mixed with $\delta = -40, 40$, respectively. The one above corresponds to a violin and the one below to a singer, both playing the same tune in unison.

variances. The last two assumptions are modeled by

$$P(S_j^k) \propto e^{-\lambda Mag(S_j^k)}.$$

For a particular k and when the mixing matrix is known, we want to maximize the posterior probability of the sources given the data, that is

$$max_{\mathbf{S}} P(\mathbf{S}^k|\mathbf{X}^k) \propto max_{\mathbf{S}} P(\mathbf{X}^k|\mathbf{S}^k)P(\mathbf{S}^k).$$

Since there is no noise in the model, $\mathbf{X}^k = \mathbf{Z}^k\mathbf{S}^k$, and $P(\mathbf{X}^k|\mathbf{S}^k) = 1$. Therefore, under the assumptions above,

$$max_{\mathbf{S}} P(\mathbf{S}^k|\mathbf{X}^k) \propto \Pi_j e^{-\lambda Mag(\mathbf{S}_j^k)}$$

Taking logarithms, the above is equivalent to

$$\begin{aligned} &min_{\mathbf{S}} \sum_j Mag(S_j^k) \\ &\text{subject to} \quad \mathbf{X}^k = \mathbf{Z}^k\mathbf{S}^k. \end{aligned} \tag{8}$$

Notice that, even though the objective function depends only on the magnitudes, the phase information is preserved by the constraints.

This problem is a particular instance of second-order cone programming [7] and our experiments have been performed using the software package in [10]. The number m of sensors in this case is unrestricted.

After this stage, the resulting estimated values $\hat{\mathbf{S}}^k$ where transformed back to the time domain with the inverse DFT to obtain the vector of estimated signals $\hat{\mathbf{s}}(t)$.

3 Experiments

The experiments were conducted on the following 4 sets of signals:

VoiceOnViolin A set of 2 signals of 6.8 s of duration, with a female voice and a violin singing and playing the same tune (unison). The signals were recorded with a low quality microphone on a home computer and sampled at 22,050 Hz and 8 bits.

FourVoices A set of 4 signals of 2.9 s of duration with speech from four different people, recorded on the same computer with the same parameters.

FiveSongs A set of 5 signals with 5 s long excert of comercial CDs (two classical and three pop/folk music), downsampled to 11,025 Hz.

SixFluteMelodies A set of 6 signals of 5.7 s of duration with different flute melodies, recorded with a high-quality registration at 44,100 Hz/16 bits, and down-sampled to 22,050 Hz [1].

3.1 Synthetic Mixtures in a Simulated Acoustic Scenario

The experiments reported in this work were performed on synthetic mixtures simulating a real scenario without noise, echo or reverberation. That is, only attenuation and delay were considered, but the selected values were chosen to be spatially consistent.

In an idealized model, let's assume that the sources are punctual and that the pressure at the wave front decreases as $1/r$, with r the distance from the source (the assumption is plausible for magnetic microphones far enough from the sources). Let's set the microphones at coordinates $(d/2, 0)$ and $(-d/2, 0)$, with d the distance between the microphones. Then, given the coordinates (x, y) of source j, let R and L be the distance from the source to the right and left microphones, respectively. The relative attenuation at the sensors is then defined as $\alpha = R/L$ and, imposing unit lenght, we built $\mathbf{B}_j$ as

$$\begin{aligned}\beta_{1j} &= \sqrt{\frac{1}{1+\alpha^2}} \\ \beta_{2j} &= \alpha B_{1j}.\end{aligned}$$

Similarly, the time delay from source j to the microphones is proportional to the distance and, since we are interested in differential delays, we set $\delta = (L-R)f_s/v$ in samples, with f_s the sampling rate and v the speed of sound. Then, imposing average zero, we take

$$\begin{aligned}\tau_{1j} &= \delta/2 \\ \tau_{2j} &= -\delta/2.\end{aligned}$$

Repeating the procedure for all sources in the scenario, we build the simulated mixing matrices $\mathbf{B}$ and $\mathbf{T}$.

Under this model, given d, the overall relationship between the position (x, y) and the pair (α, δ) is biunivocal, except for a singularity in the central axis, where $\alpha = 1$ and $\delta = 0$. Thus, determining α and δ for a source can be used not only for its separation, but also for its spatial localization.

For the four sets of signals described before, a plausible arrangement of the sources was designed, as depicted in figure 3, where circles represent the microphones and stars represent the sources. In the background, solid lines are curves of constant relative attenuation, and dotted lines are curves of constant differential delay.

3.2 Simulations and Results

The four sets of sources where mixed with the corresponding matrices of the previous section using the mixing model of equation (1), to obtain four sets of two-channel sensor signals. These were the input to the simulation experiments. Results are summarized in table 1, and described in the following.

Estimating the attenuation matrix

The attenuation matrix was estimated using the procedure of section 2.1. The potential function was computed with a grid with $G = 90$ points from 0 (left channel equals 0) to 90 degrees (right channel equals 0), yielding an angular resolution of 1 degree. To reduce computational load, only the 1000 largest (most significant) data points were used. Parameter λ in equation (6) was

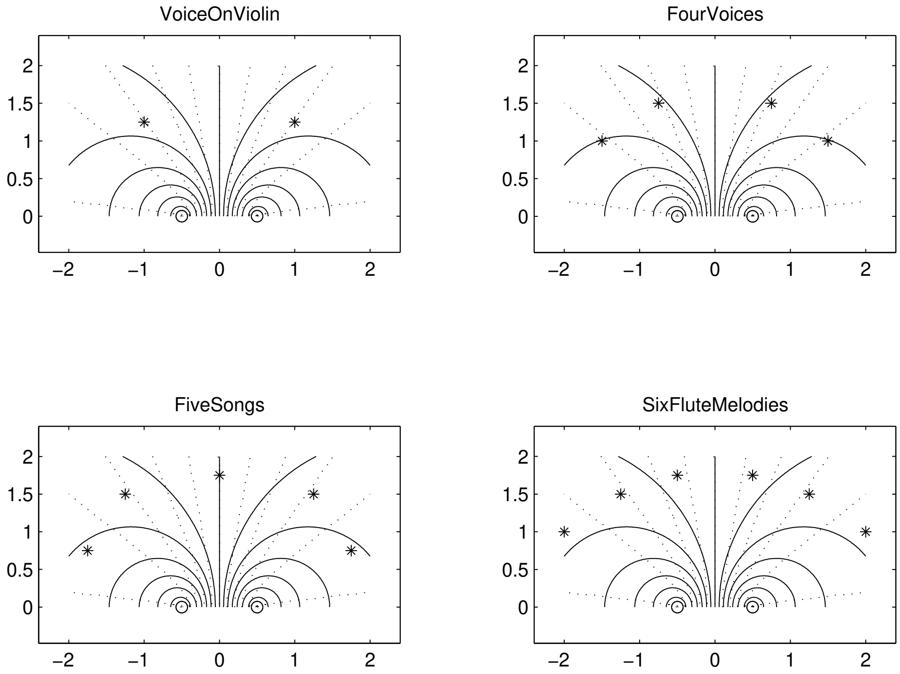

Fig. 3. Spatial arrangement of the sources for each of the four sets of signals, over curves of constant relative attenuation (solid) and constant differential delay (dotted).

adjusted by automatically repeating the procedure until the desired number of sources were obtained. The actual values used in the following experiments were λ=8, 22.6, 19.6 and 75.4, for the four data sets, respectively.

For each source, then, the error in the estimation was measured as the angular distance between the actual and the estimated director vectors. The first column of table 1 shows, for each data set, the average of the absolute values of the errors (in degrees), followed by the worst case. In all cases but one (one of the sources in the FourVoices data set), the errors were smaller than 1 degree, the resolution of the grid.

Estimating the delay matrix

The delay matrix was estimated using the procedure of section 2.2. In order to measure the contribution of this stage, without accumulated error, $\hat{\mathbf{T}}$ was first estimated by using the real values of $\mathbf{B}$, rather than the results of the previous experiments. The range of differential delays was set to explore all possible spatial orientations of the sources, from one side of the right microphone to the other side of the left microphone. Translated into samples, and depending on the sampling rate, this led to the following values: 65, 65, 32 and 65, with a resolution of one sample. Delays, being synthetic, had the same resolution and thus the correct answers should be exact.

The second column of table 1 shows the average absolute errors, in samples, followed by the worst case. As shown, all the estimates were correct.

Repeating the experiment with the estimated attenuations $\hat{\mathbf{B}}$, only one source, in the SixFluteMelodies case, was incorrectly estimated. Considering that only 1000 data points were used, the performance was quite good. In fact, in additional experiments, the error could be corrected by adding more data points.

Recovering the Sources with Known Matrices

The sources were recovered using the procedure of section 2.3. As before, we first evaluated the results using the real attenuation and delay matrices. For a particular source j, the error measure that we used is the S/N ratio defined as

$$S/N = 10 \log \frac{||\hat{S}_j(t) - S_j(t)||^2}{||S_j(t)||^2},$$

with $|| \cdot ||$ the sum of squares over discrete time t. Column (IV) of table 1 shows the average S/N ratios, followed by the worst case. Notice how in the VoiceOnViolin case, being determined ($n = m = 2$), the performance was much better, limited only by the inaccuracy in the analysis and resynthesis procedures. The quality of the separation is remarcable for a unison mix. For the rest, notice that the worst case is allways above 10dB, which is the standard threshold of inteligibility.

Repeating the experiments with $\hat{\mathbf{B}}$ and $\hat{\mathbf{T}}$, results are shown in the next column. The loss in the VoiceOnViolin case is due to the inaccuracy in the estimation of $\mathbf{B}$, but the impact of this inaccuracy is much less important in the remaining cases, where the main difficulty lies in the recovery of the sources. Notice how the wrong estimate in delay for the above mentioned flute totally disrupted its reconstruction (-9.3dB).

For the sake of comparison, and with the exception of the VoiceOnViolin case (not available), the last column of the table shows the results obtained in [2] under a much easier mixing setting (a real mixing matrix with attenuations only and maximum separation between the director vectors), using the procedure of section 2.1 and assuming laplacianity of the real and imaginary parts (linear programming). Notice that results are very close, suggesting that the major limitation is the spectral overlap between the sources.

Apart from the missing flute, listening to the recovered signals one can clearly identify the corresponding source in the foreground, although the remaining voices are still noticeble in the background (with a remarcable time distor-

	I		II		III	
VoiceOnViolin	0.09	0.09	0	0	0	0
FourVoices	0.59	1.59	0	0	0	0
FiveSongs	0.33	0.54	0	0	0	0
SixFluteMelodies	0.41	0.47	0	0	3.5	21
	IV		**V**		**VI**	
VoiceOnViolin	69.6	69.6	56.3	56.2		
FourVoices	17.1	11.7	17.3	11.9	18.4	15.7
FiveSongs	13.6	10.4	13.5	10.2	15.3	15.0
SixFluteMelodies	19.8	10.8	15.6	-9.3	20.7	14.2

Table 1
Results of the separation experiments (average followed by worst case values). (I) Angular error in $\hat{\mathbf{B}}$ (in degrees). (II) Delay error in $\hat{\mathbf{T}}$ using the original $\mathbf{B}$ (in samples). (III) The same as (II) using $\hat{\mathbf{B}}$. (IV) S/N ratio for the separation using the original $\mathbf{B}$ and $\mathbf{T}$ (in dB). (V) The same as (IV) using $\hat{\mathbf{B}}$ and $\hat{\mathbf{T}}$. (VI) S/N ratio for the instantaneous mixtures in [2].

tion due to their different delays). The wave files with sources, mixtures and recovered signals are available on-line in [2].

4 Conclusions

In this paper we have presented a procedure for the blind separation of delayed sources, using the frequential representation of the signals as obtained from the DFT. The procedure included three distinct stages: (a) clustering of the magnitudes to estimate the attenuation matrix, (b) clustering maximization of the real and imaginary parts to estimate the delay matrix, and (c) a maximum a posteriori approach to source decomposition which, under the assumption of laplacian magnitudes, lead to a second-order cone programming problem.

Except for one signal, the experiments performed on simulated acoustic scenarios showed a very good perfomance of the three stages, even when compared with the separation of the same sources after instantaneous mixtures. As a by-product of the synthesis of the scenarios a relationship was derived between the attenuation and delay of a source and its spatial localization.

For the determined case, the proposed procedures for the estimate of the mixing matrices work perfectly, and for the determined case the separability of the sources seems to be physically limited by the spectral overlap of the sources. The assumption of equal variances of the magnitude coefficients is probably not accurate since, at least for speech and music signals, most of the energy concentrates in the lower side of the spectrum.

Possible directions for future work include the analysis and exploitation of the input distribution, the search for highly sparse (and hopefully disjoint) representations, and the extension of the delay estimation procedure to $m > 2$. Finally, the spatial localization of the sources could be used as a filter to discard unconsistent results.

Acknowledgements

The author would like to thank the usefull discussions and comments from Paul Tseng (UW), Enric Monte (UPC) and Juan Pampín (UW).

References

[1] Antas L., all flute examples were performed by Linda Antas, University of Washington.

[2] Bofill P. and Zibulevsky M., "Underdetermined Blind Source Separation using Sparse Representations", submitted to *Signal Processing*, 2000. http://www.ac.upc.es/homes/pau/

[3] Hyvärinen A., "Survey on Independent Component Analysis", in *Neural Computing Surveys*, No 2, pp 94-128, 1999. http://www.cis.hut.fi/projects/ica/

[4] Jutten C. and Herault J., "Blind Separation of Sources, an Adaptive Algorithm Based on Neuromimetic Architecture", in *Signal Processing*, Vol 24, No 1, pp 1-10, 1991.

[5] Lee T.W., Ziehe A., Orglmeister R. and Sejnowski T., "Combining Time-Delayed Decorrelation and ICA: Towards Solving the Cocktail Party Effect", in *Proc. IEEE Int. Conf. on ASSP*, Vol 2, pp 1249-1252, 1998.

[6] Lee T.W., *Independent Component Analysis: Theory and Applications*, Kluwer Academic Publishers, 1998.

[7] Lobo M.S., Vandenberghe L., Boyd S. and Lebret H., "Applications of Second-order Cone Programming", in *Linear Algebra and its Applications*, 284, pp 193-228, 1998.

[8] Olshausen B.A. and Field D.J., "Sparse coding with an overcomplete basis set: A strategy employed by V1?", in *Vision Research*, 37: 3311-3325, 1997.

[9] Parra L. and Spence C., “Blind Source Separation based on Multiple Decorrelations”, in *IEEE Trans on Speech and Audio Processing*, pp 320-327, May 2000.

[10] Sturm J.F., “Using SeDuMi 1.0x, a Matlab Toolbox for Optimization over Symmetric Cones”, 1999, http://www.unimaas.nl/ sturm

[11] Torkkola K., “Blind Separation of Convolved Sources based on Information Maximization”, in *IEEE Workshop on NNSP'96*, Kyoto, Japan, Sept 1996.

[12] Zhang L. and Cichocki A., “Blind Deconvolution/Equalization Using State-Space Models”, in *Proc. of the IEEE Workshop on NNSP'98*, pp 123-131, 1998.

[13] Zibulevsky M., Pearlmutter B.A., Bofill P. and Kisiliev P., “Blind Source Separation by Sparse Decomposition in a Signal Dictionary”, in *Independent Component Analysis: Principles and Practice*, Roberts S.J and Everson R.M., Eds. Cambridge University Press, 2000. In press.

[14] Zibulevsky M. and Pearlmutter B.A., “Blind Source Separation by Sparse Decomposition”, TR No. CS99-1, University of New Mexico, Albuquerque, July 1999. http://www.cs.unm.edu/˜bap/papers/sparse-ica-99a.ps.gz