Blind Audio Source Separation: State-of-Art

Abouzid Houda
RSAID Laboratory, University Abdelmalek Esaadi
Tetouan, Morocco

Chakkor Otman
RSAID Laboratory, University Abdelmalek Esaadi
Tetouan, Morocco

ABSTRACT
The word is surrounded by sounds what makes it difficult when it becomes impossible to obtain a desired speech because of the noisy environment. Thus, digital signal processing is a discipline that interest to extract useful information on physical phenomena from measures generally disturbed. Its most well know problem is the blind sources separation which is a specific method that in which several signals have been mixed and the purpose is to recover the original component signals from the mixed signals without any knowledge about the sources.

This work, provides some of many existing algorithms solving the problem of blind source separation the most known in literature and at the end of this article there are some examples applied to real-world audio separation tasks using Matlab.

General Terms
Algorithms, formulations and blind signal separation.

Keywords
Blind source separation (BSS); convolutive mixture; instant linear mixture; independent component analysis; principle component analysis.

1. INTRODUCTION
Source separation is a very active research topic because of its theoretical interest and its many practical applications. Its principle is to retrieve signals when no prior information of the sources or channel mixing system is available, so then the task is called Blind Source Separation (BSS).

Source separation occurs in various applications such as locating, tracking radar targets and sonar, noise removal (cocktail party, deleting the vuvuzela, karaoke), detection and separation in multiple access communication systems, separating instruments in polyphonic music, remixing, transformations, etc [1]. The solutions proposed for this BSS differ according to the nature of the sources, the type of mixture (instantaneous or convolutive) and the case of mixture (over-determined, determined or undetermined). This problem has been intensively studied in the literature and many effective solutions have already been proposed.

2. PROBLEM REFORMULATION
Let’s start by illustrating the problem of source separation by the following simple example:

Robert is an old men he gives an order to Romeo while being listening to the radio. The voice of Robert arrives at the robot mixed with the signal from the radio and that can be recorded using the microphones or Romeo (acoustic sensor). Let’s look at the figure (1); the two signals are the two sources that sounds are mixed before being received by each sensor.

The subject interesting here is the blind audio source separation; therefore, the goal is to restore the signals transmitted using only the received signals. To do so, researchers in this field are essentially based on the assumption of the mutual independence of source signals, ie that the probability density of the source signals is equal to the product of their marginal probabilities densities.

In this case the problem is also known by the name of the Independent Component Analysis ICA) [2].

3. TYPE OF MIXTURES
3.1 Instant Linear Mixture
In 1985, Hérault, Jutten and Ans published the first source separation algorithm. The algorithms of Hérault, al. and Cardoso and many others use the higher order of statistics (HOS) for the constraint of the independence of sources [3].

We consider m as unknown channels of signal with 1 ≤ i ≤ m and n Mixture of sensor signals 1 ≤ j ≤ n.

The following figure illustrates this model of mixture:

\[
x(k) = \sum_{i=1}^{n} a_{i} s_{i}(k) + v_{j}(k)
\]

(1.1)

Where \( a_{i} \) the coefficients of the linear time-invariant mixing system are represented by the \((n \times m)\) matrix \( A \) and \( v_{j}(k) \) is additive noise at the jth sensor.

Figure 1: Example of mixture (ref: https://tel.archives-ouvertes.fr/pastel-00758370/document p: 20)

Figure 2: Example of BSS for instant mixture
The goal of BSS for instantaneous mixtures is to adjust the coefficients of an m × n separation or demixing matrix B such that:

\[ y_i(k) = \sum_{j=1}^n b_j(k)x_j(k) \]  

(1.2)

Where the coefficients \((k)\) represent the impulse response of the multichannel separation system filter [4].

a) The algorithm of JADE of J.-F. Cardoso

The JADE method collects second and fourth order statistics from segments of sound signals to form a set of matrices for joint orthogonal diagonalization, which provide an estimation of de-mixing matrix and independent sources. This algorithm is based on the joint diagonalization of a set of matrices. JADE algorithm starts with a bleaching step where the whitened observations are diagonal spectral matrix. This step is based on the use of the two order statistics. It makes independent observations in that order. Then a second step is called rotation will be to make independent observations to order four by a criterion based on cumulants of order four. The strongest point of JADE for applications of ICA is that it works without the using of parameter tuning. The weakest point of the current implementation is that the number of sources (but not of sensors) is limited in practice (by the available memory) to something like 40 or 50 depending on your computer [5, 6].

b) The algorithm AMUSE of L. Tong and al

That algorithm belongs to the group of second-order-statistics spatio-temporal decorrelation (SOS-STD) BSS algorithms. It can be considered as two consecutive PCAs: First, PCA is applied to input data; secondly, PCA (SVD) is applied to the time-delayed covariance matrix of the output of previous step. The AMUSE algorithm is applied on the matrix \( R_x \) following the steps below:

The method

1) Calculate the covariance matrix \( R_x = X.X' \).
2) Find the singular values \( \Phi \) of \( R_x \) to determine the three matrices: \( U \) as a unit matrix, \( V \) is the diagonal matrix for transformation and \( \Phi \) is the required matrix: \( R_x = V.\Phi.U \).
3) Eliminate the Gaussian noise variance as the mean of the singular matrix with the equation \( \sigma^2 = mean(\Phi^{-1.2}.V.X) \) : \( X'_s = X - \sigma^2 \).
4) Determine the covariance of \( X'_s \) and decompose it to find its eigenvalues and the corresponding eigenvectors \( \Lambda \).
5) Obtain the transformation matrix \( H = \Lambda^{1/2}.\Phi^{-1/2}.V \).
6) Finally, estimate the independent component as \( Y(n) = H.X \).

AMUSE is the faster algorithm that its processing speed is mainly defined by the PCA processing within it and it is very easy to use because no parameters are required [7]. The structure of the algorithm is illustrated in figure 3a below:

![Figure 3a: AMUSE algorithm](http://cate.fiu.edu/sites/default/files/Publications/Venda_JCNP_2012_0.pdf)

c) The algorithm SOBI of A. Belouchrani

In the same way (colored sources of different spectra), based on second-order statistics, Belouchrani developed a new method of source separation instantly mixed and noisy, he uses the correlation property of the source signals which is performed by the matrix intercovariance several delays. SOBI algorithm utilizes the joint diagonalization procedure unlike AMUSE and FOBI which use EVD / SVD techniques. It operates on time delayed covariance matrices. SOBI utilizes the pre-whitening transformation similar to that described in case of AMUSE. This is followed by estimation of set of covariance matrices for a preselected set of time lags \( \{P_1, P_2, ..., P_L\} \) mathematically, this means that for all time lags, the source correlation matrices are diagonal:

\[ R_x(\tau) = E[x(t)x(t+\tau)^T] = AR_x(\tau)A' \]  

(1.3)

Where represents the correlation matrix of the source signals. Considering that this equation holds for all values of \( \tau \), the mixing matrix \( A \) is the one that jointly diagonalizes all the correlation matrices [8].

d) The algorithm of D.-T. Pham and J.-F Cardoso

In this work the authors developed a simple and efficient algorithm for separating non-stationary source signals with variable amplitude; for example for the separation of speech signals. The algorithm optimizes mathematically, "on-line" or "off-line", two cost functions using an approach based on maximum likelihood (vraisamblance) and as we know, this method, based on an observed sample, calculates the best values of a parameter of a probability.

3.2 Convolutive mixture

The first known work for the convolutive mixing systems was introduced in the early 90s and it is still be considered present in the concerns of many researchers. Generally, all methods of blind sources separation for the context convolutional fall into three main classes:

- Temporal approaches.
- Frequency approaches.
• Time-frequency approaches.

The methods in context convolutional like instant context exploit at least one of the following assumptions: independence, non-stationary, non Gaussianity and temporal consistency (i.e. non-white sources) sources.

**Definition 2:** the convolutive mixture is due to time delays resulting from sound propagation over space and the multipath generated by reflections of sound of different objects, particularly in rooms and other enclosed settings.

As a result, the m sources are mixed by a time-dispersive multichannel system as:

$$x_i(k) = \sum_{l=-\infty}^{\infty} a_{il} x_l(k-l) + V_i(k)$$  \hspace{1cm} (1.4)$$

Where \((x_i(k)), 1 \leq j \leq n\) are the n sensor signals and \(\{a_{il}\}\) are the coefficients of the discrete-time linear time-invariant mixing system and the matrix A is of dimension \((n \times m)\).

e) **FPICA algorithm**

Suppose that we have a prewhitened random vector \(x\), which is a collection of linear mixtures of independent sources.

- To estimate one independent component:
  1. Choose a random initial vector \(w(0)\) of norm 1.
  2. Let \(k = 1\).
  3. \(w(k) = E\{x(w(k-1)x)^2\} - w(k-1)\), and then divide \(w(k)\) by its room.
  4. If \(w(k)w(k-1)\) is very different from 1, let \(k = k + 1\) and go back to step 2 and output the vector \(w(k)\).

f) **SIMO-ICA algorithm of T. Takatani and al.**

The authors propose in this work a new algorithm for the separation of acoustic signals in a system with one input and multiple outputs. The input is a single source and the outputs are its transmitted signals observed at multiple microphones. The algorithm is composed of \((L-1)\) principal component analysis and fidelity controller separation system. The experiment of this algorithm shows a good quality of the separate audio signals (see the figure 4). The separated signals of the first ICA \((1, \cdots, L-1)\) are defined by:

$$y_{IICA}(t) = \left[\begin{array}{c} y_{1IICA}(t) \\ \vdots \\ y_{L-1IICA}(t) \end{array}\right] = \sum_{n=0}^{D-1} w_{iIICA}(n)x(t-n)$$  \hspace{1cm} (1.5)$$

Each element of the separation matrix is represented as an FIR filter and \(w(n) = \left[\begin{array}{c} w_{iIICA}(n) \end{array}\right]\) is the separation filter matrix, \(w(\cdot)\) is the z-transform of \(w(n)\) and D is the filter length of \(w(n)\) [10].

g) **MSICA algorithm of T. Nishikawa and al.**

In this algorithm two ICA (INDEPENDENT COMPONENT ANALYSIS) approaches, time and frequency, are combined to build a more strong separation procedure, using only forth order statistics of the data. Its speed of convergence and its robustness to the spatial correlation (unknown to the receiver) of Gaussian noise makes it one of the most efficient algorithms at present.


 Figure 4: Input and output relations in SIMO-ICA: ref [10]

h) **The algorithm of J. Anemüller and B. Kollmeier**

Anemüller and al. have studied the separation of convolutive mixtures of speech signals. The algorithm that they proposed for this is derived from the principle of decorrelation by amplitude modulation. This algorithm collects information necessary for the separation of multiple frequencies simultaneously. From the analysis of a speech spectrogram, the authors conclude that the production of the human voice naturally follows the amplitude change based on the time variable of the spectrogram in different frequencies. That is why they call this property “amplitude modulation correlation” (Amcor) [12].


 Figure 5: MSICA algorithm using FDIICA and TDICA: ref [11]
covariance matrix of input data and the goal is to find \( w_k \) with \( \tilde{x} = w_k x \).

1. PCA can be accomplished by EVD or SVD methods:

   PCA with **EVD**: to obtain \( w_k \), we use EDV method to decompose the input correlation matrix \( R = E \{ xx^T \} \) which is symmetrical and find the eigenvectors and eigenvalues:

   \[
   R = E D E^T
   \]

   Where \( E \) is an eigenvector matrix and \( D = \text{diag}(d_1, d_2, ..., d_n) \) diagonal matrix of eigenvalues.

   \[
   E D E^T = ED^{1/2} D^{1/2} E^T = E \{ xx^T \}
   \]

   \[
   = E \{ w_h^T \tilde{x} \tilde{x}^T w_h^{-1} \} = w_h^T E \{ \tilde{x} \tilde{x}^T \} w_h^{-1}
   \]

   \[
   = w_h = \sqrt{k} SU^T
   \]

   From (1.6) we obtain this:

   \[
   w_h = (ED^{1/2})^{-1} D^{1/2} E^{-1} = D^{-1/2} E^T
   \]

   Where \( D^{1/2} = \text{diag}(d_1^{1/2}, d_2^{1/2}, ..., d_n^{1/2}) \) [13].

   1. PCA with **SVD**: singular value decomposition is another way more robust and simple to decompose the observation \( X = AZ \) into \( X = USV^T \) where \( S \) is a diagonal matrix with \( S = \text{diag}(\sigma_1, \sigma_2, ..., \sigma_n) \) is a singular value. Then the whitened data can be seen as:

   \[
   R_{xx} = E \{ xx^T \} = \frac{1}{k} X X^T = \frac{1}{k} U S V^T (U S V^T)^T = U S V^T
   \]

   Where \( UU^T = I \) and \( VV^T = I \).

   From \( x^T = w_k^{-1} \) and \( X = USV^T \) we obtain:

   \[
   \tilde{x} = \sqrt{kV^T} \text{ and } w_k = \sqrt{kSU^T} \] [14]

5. TEST AND MEASUREMENT

This section presents some simulation results on MATLAB. After loading a benchmark and selecting the data file desired for example “Sergio.mat”. This benchmark contains seven sources (some of them are asymmetrically distributed). This benchmark was proposed by Sergio Cruces.

It is a particular case (three signals and three sensors) with a sampling frequency of 500 Hz. The matrix of the mixture is the orthogonal matrix as seen in the following figure:

\[
H = \begin{pmatrix}
-0.468883 & 0.759592 & -0.450742 \\
-0.822756 & -0.561236 & 0.089928 \\
-0.321281 & 0.328685 & 0.888113
\end{pmatrix}
\]

With \( \det(H) = 1 \) and \( \text{cond}(H) = 1 \).

A Gaussian noise of 20dB was added to the signals and with no filtering at the beginning.

Test 1: starting by testing the **AMUSE** algorithm with its defaults settings. The results obtained are as follows:

![Figure 6: The three signals selected](image1)

![Figure 7: Mixture of sources](image2)

![Figure 8: Estimated sources](image3)

A simple observation shows that \( y_1 \) is almost similar to \( s_1 \), \( y_2 \) has the same appearance as \( s_2 \) but there are some differences in magnitude and also \( y_3 \) is similar to \( s_1 \).

In general, and from the figures its shows that there is a permutation in the sources, which means that the estimated sources do not keep the same order as the original sources, also there is a difference in the scale assumed by the difference of magnitude of each estimated signal.

![Figure 9: The estimated sources after “lowpass filtering (butterworth)”](image4)
Test 2: SOBI algorithm
Again with the same previous data and their parameters without the step of preprocessing. The algorithm is executed. The result is obtained as shown in the following figure:

Figure 10: The sources estimated by SOBI

Here, $y_1$ and $y_2$ are almost similar to their original sources like those shown in AMUSE algorithm but for the signal $y_3$, it is upturned from its original order.

Test 3: FPICA algorithm

It can be noted from the Figure 12 that a good results of estimated sources have been obtained and almost they are perfectly seem like their original signals.

Test 4: FCICA (Fast Constrained ICA) algorithm

Obviously, the signals are badly built; the purple line shows the shame of the original sources as they have to be instead of the blue line.

The table below shows the performance of some algorithms have been seen before:

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Performance of semi/orthogonality</th>
<th>Performance Index of separability</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMUSE</td>
<td>0.1404316741</td>
<td>0.0732550592</td>
</tr>
<tr>
<td>SOBI</td>
<td>0.1396732656</td>
<td>0.0727586825</td>
</tr>
<tr>
<td>FPICA</td>
<td>0.0383082425</td>
<td>0.0188579223</td>
</tr>
<tr>
<td>FCICA</td>
<td>0.8054618179</td>
<td>0.2663797978</td>
</tr>
</tbody>
</table>

6. CONCLUSION

Basically, the literature is very rich in many algorithms of blind source separation of instant mixtures where the dependency criteria are so different, and separation results are very good. However, these algorithms do not have the ability to separate the signals which are in the majority of the actual physical situations, which correspond to the convolutive case and where the mixture is under-dermined.

The next step will be to develop a new algorithm which separate audio signals from its mixture and its application will be for the humanoid robot in reverberate environment.

7. REFERENCES


