Single-Microphone Blind Audio Source Separation via Gaussian Short+Long Term AR Models

Antony Schutz, Dirk Slock

***Abstract*— Blind audio source separation (BASS) arises in a number of applications in speech and music processing such as speech enhancement, speaker diarization, automated music transcription etc. Generally, BASS methods consider multichannel signal capture. The single microphone case is the most difficult underdetermined case, but it often arises in practice. In the approach considered here, the main source identifiability comes from exploiting the presumed quasi-periodic nature of sources via long-term autoregressive (AR) modeling. Indeed, musical note signals are quasi-periodic and so is voiced speech, which constitutes the most energetic part of speech signals. We furthermore exploit (e.g. speaker or instrument related) prior information in the spectral envelope of the source signals via short-term AR modeling, to also help unravel spectral portions where source harmonics overlap, and to provide a continuous treatment when sources (e.g. speech) temporarily lose their periodic nature. The novel processing considered here uses windowed signal frames and alternates between frequency and time domain processing for optimized computational complexity and approximation error. We consider Variational Bayesian techniques for joint source extraction and estimation of their AR parameters, the simplified versions of which correspond to EM or SAGE algorithms.**

***Index Terms*— Variational Bayes, Expectation Maximization, Blind Source Separation, Speech Processing, Autoregressive process, Linear Prediction**

I. Introduction

The need for Blind Audio Source Separation (BASS) arises with various real-world signals, including speech enhancement, speaker diarization, automated music transcription etc.. Generally, BASS methods consider multichannel signal capture and has been dealt with extensively in the literature. In the over determined case of BSS the source separation can be performed satisfactorily, especially in clean environment, for example by using Independent Component Analysis (ICA) [1], [2] or Computational Auditory Scene Analysis (CASA) [3]. ICA assumes that there are at least as many observation mixtures as there are independent sources. For underdetermined BSS (UBSS), the problem is ill-defined and its solution requires some additional assumptions.

This paper is organized as follow. In section II we present the model of joint speech production. In section III we discuss the introduction of windows. Then, in sections IV, V and VI we explain the methodology and the algorithm for the source separation problem using a variational bayesian framework. In section VII we present some results and finally we conclude.

II. Signal Model

We consider the problem of estimating K Gaussian sources from a single mixture. We use the short+long term autoregressive (AR) voice production model [4]:

$$y_t = \sum_{k=1}^{K} x_{k,t} + v_t, \tag{1}$$

$$x_{k,t} = \sum_{n=1}^{p_k} a_{k,n} x_{k,t-n} + f_{k,t}, \quad f_{k,t} = b_k f_{k,t-\tau_k} + e_{k,t} \ .$$

Here, y_t is the measured mixture of signals, K is the number of sources x_k. v_t is an additive white Gaussian noise of variance σ_v^2 and is assumed to be uncorrelated with the sources. $e_{k,t}$ is the excitation signal of source k, also assumed to be Gaussian and white with variance σ_k^2. For each source x_k, τ_k is the period (its fractional part can be implemented by linear interpolation if the samplinf frequency is high enough), b_k its long-term prediction coefficient and the short-term prediction coefficientscoefficient, of order p_k, are $a_{k,n}$; f_k is the short-term prediction error. If we introduce the short-term and long-term prediction error transfer functions

$$A_k(z) = \sum_{n=0}^{p_k} a_{k,n} z^{-n}, \ B_k(z) = 1 - b_k z^{-\tau_k} \tag{2}$$

with $a_{k,0} = 1$, then we can rewrite the various signals as

$$f_{k,t} = A_k(q)\, x_{k,t}, \ \ e_{k,t} = B_k(q)\, f_{k,t} = B_k(q)\, A_k(q)\, x_{k,t}$$

where q^{-1} is the unit sample delay operator: $q^{-1} x_{k,t} = x_{k,t-1}$. We shall also need the signals

$$g_{k,t} = B_k(q)\, x_{k,t} \ , \ k = 1, \ldots, K \ . \tag{3}$$

In the approach considered here, identifiability comes essentially from exploiting the presumed quasi-periodic nature of sources via long-term AR modeling introduced above. Indeed, musical note signals are quasi-periodic and so is voiced speech, which constitutes the most energetic part of speech signals. We furthermore exploit (e.g. speaker or instrument related) prior information in the spectral envelope of the source signals via short-term AR modeling, to also help unravel spectral portions where source harmonics overlap, and to provide a continuous treatment when sources (e.g. speech) temporarily lose their periodic nature.

EURECOM's research is partially supported by its industrial members: Swisscom, Thales, SFR , Orange, STEricsson, Cisco, BMW Group , SAP, Monaco Telecom and Symantec. The research reported herein was also supported in part by the EU NoE K-Space and by the Institut Telecom ”Futur et Ruptures” project SELIA.

The authors are with the Mobile Communications Dept., EURECOM, 2229 Route des Crêtes, BP 193, 06904 Sophia Antipolis Cedex, France; `firstname.lastname@eurecom.fr`

III. Windowing for Frame-Based Processing

The signals considered are by nature non-stationary. If we can consider the parameters constant during a short time, we can process the signal in frames (time segments), over which the signal can be considered stationary, which corresponds to time-invariant filtering. Many of the signal processing operations (e.g. linear time-invariant filtering and filter computation) could be largely simplified by passing to the frequency domain. However, transforming a frame of signal to the frequency domain directly via the DFT (FFT) leads to approximations due to the periodic extension of the frame assumption inherent in the DFT.

A. Windowing Methodology

The introduction of a window allows to reduce the approximation error. Consider e.g. the stacked N samples in a frame of the prediction error signal (vectors and matrices are denoted by bold letters)

$$\mathbf{f}_k = \mathbf{T}_{A_k}\,\mathbf{x}_k \tag{4}$$

where $\mathbf{T}_{A_k}$ is the $N \times (N{+}p_k{-}1)$ banded Toeplitz matrix corresponding to the prediction error filter $A_k(q)$ (to ease the notation we shall suppress the time index of the frame). To transform a filtering matrix easily, it should be circulant, in which case the DFT diagonalizes the matrix. The direct approximation of a Toeplitz matrix by a circulant matrix is only acceptable when the matrix dimension is much larger than the filter length. To aid in the approximation, we shall introduce an analysis window $\mathbf{w} = [w_0\, w_1 \ldots w_{N-1}]^T$, with associated diagonal weighting matrix $\mathbf{W} = \text{diag}\,\{\mathbf{w}\}$. The windowed prediction error $\mathbf{Wf}_k$ requires $\mathbf{WT}_{A_k}$. Now, assume the window decays to zero at its edges and varies sufficiently slowly, then the following approximations become valid:

$$\mathbf{W}\,\mathbf{T}_{A_k} \approx \mathbf{W}\,\mathbf{A}_k \approx \mathbf{A}_k\,\mathbf{W} \tag{5}$$

where $\mathbf{A}_k$ is a $N \times N$ square circulant matrix, corresponding to circulant convolution with $A_k(q)$. We shall similarly introduce the circulant $\mathbf{B}_k$, though the approximations considered above will be rougher for the filter $B_k(q)$ (or $B_k^{-1}(q)$) since long-term prediction has larger delay spread than short-term prediction. Note that just like $A_k(q)B_k(q) = B_k(q)A_k(q)$, also $\mathbf{A}_k\mathbf{B}_k = \mathbf{B}_k\mathbf{A}_k$. Then we get the following signal relations

$$\begin{aligned}\mathbf{We}_k &= \mathbf{A}_k\mathbf{B}_k\,\mathbf{Wx}_k = \mathbf{A}_k\,\mathbf{Wg}_k = \mathbf{B}_k\,\mathbf{Wf}_k\\ \mathbf{Wg}_k &= \mathbf{B}_k\,\mathbf{Wx}_k\ ,\ \ \mathbf{Wf}_k = \mathbf{A}_k\,\mathbf{Wx}_k\ .\end{aligned} \tag{6}$$

Just like the original data signal y_k will be cut into a series of windowed frames, a processed signal (e.g. extracted source) will be reconstructed by superposing its reconstructed windowed frame segments. Since the window needs to decay towards its edges, consecutive frames need to overlap. Let M be the hop size (time jump) from one frame to the next, then a perfect reconstruction (PR) window w_t requires

$$\sum_{i=-\infty}^{\infty} w_{t-iM} \;=\; 1\ ,\ \ \forall n \tag{7}$$

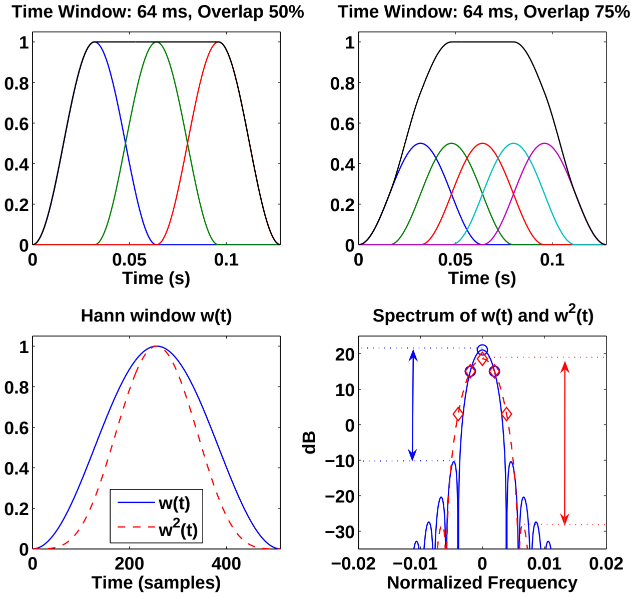

Fig. 1. Perfect reconstruction windowing.

see the top figures in Fig. 1 for the cases of relative overlap of $(N{-}M)/N = 50\%, 75\%$ (both the individual windows and their sum are shown for a finite set of windows). Note that one could consider extensions to non-PR windows, in which the superposition of windowed signal frames could be followed by a zero-forcing rescaling with $1/(\sum_{i=-\infty}^{\infty} w_{t-iM})$ or (multi-window) MMSE versions thereof. The PR window that will be used in the simulations in this paper is a Hann (or raised cosine) window [5]

$$w_t \;=\; \frac{1}{2}\left[1 - cos\left(2\pi\frac{t}{N}\right)\right]\ ,\quad t = 0, 1, \ldots, N-1\,. \tag{8}$$

B. Frequency Domain Window Design

When applying the $N \times N$ DFT matrix $\mathbf{F}$ to the windowed signals in (6), we get

$$\mathbf{FWe}_k = (\mathbf{FA}_k\mathbf{F}^{-1})(\mathbf{FB}_k\mathbf{F}^{-1})\,(\mathbf{FWF}^{-1})\,\mathbf{Fx}_k$$

where we get diagonal frequency domain filtering matrices $\breve{\mathbf{A}}_k = \mathbf{FA}_k\mathbf{F}^{-1}$ etc. The main non-diagonal matrix will be the covariance matrix of $\mathbf{FWe}_k$, which is proportional to $\breve{\mathbf{W}}_2 = \mathbf{FW}^2\mathbf{F}^{-1}$ ($\mathbf{e}_k$ being white). For the case of the Hann window, both the window and a zoom on the main antidiagonal of the circulant $\breve{\mathbf{W}}_2$ appear in the bottom half of Fig. 1. The time domain window design criteria of decaying edges and smooth behavior translate in the frequency domain to decaying spectral smear and high sidelobe attenuation. Indeed, in order to keep a low computational complexity approach, the window spectrum will be approximated by only its main lobe. This leads to an approximation error that derives from the sidelobe attenuation level. The resulting processing will no longer involve pure diagonal matrices, but banded matrices. As the FFT points in the bottom right figure indicate, for the case of a Hann window, $\breve{\mathbf{W}}_2$ can be approximated by a symmetric banded circulant matrix with 5 diagonals, with (elementwise) approximation error attenuated by at least 30dB.

IV. Variational Bayesian Techniques

A recent tutorial on Variational Bayesian (VB) estimation techniques can be found in [6], see also [7]. It provides an approximate technique to determine the posterior probability density function (pdf) of the quantities to be estimated. Let θ denote the vector of all quantities to be estimated, including parameters and possibly signals (e.g. the "hidden variables" in EM terminology) and Y denotes the measurements. In many problems, the joint posterior pdf $p(\theta|Y)$ can be complicated to determine. Consider now a partition of θ into K subgroups of quantities that will get estimated per subgroup $\theta = \{\theta_k,\ k = 1,\ldots,K\}$. The idea of VB is to approximate $p(\theta|Y)$ by a product form $q(\theta|Y) = \prod_{k=1}^{K} q(\theta_k|Y)$ where the $q(\theta_k|Y)$ in general will differ from the true marginal pdfs $p(\theta_k|Y)$. The $q(\theta_k|Y)$ are determined by minimizing the Kullback-Leibler distance between $\prod_{k=1}^{K} q(\theta_k|Y)$ and $p(\theta|Y)$. This leads to the following implicit relations

$$\ln q(\theta_k|Y) = \mathrm{E}_{q(\theta_{\bar{\mathbf{k}}}|Y)} \ln p(Y,\theta_k,\theta_{\bar{\mathbf{k}}})\ , k = 1,\ldots,K \tag{9}$$

where $\theta_{\bar{\mathbf{k}}}$ is θ minus θ_k, hence $\theta = \{\theta_k, \theta_{\bar{\mathbf{k}}}\}$. In practice, (9) needs to be solved iteratively by consecutively sweeping through $k = 1,\ldots,K$, at all times using for $q(\theta_{\bar{\mathbf{k}}}|Y)$ the latest version available. This iterative process can be shown to converge monotonically. Typically, when $p(Y|\theta)$ and the prior $p(\theta)$ are exponential pdfs (typically Gaussian), then one can see from (9) that $q(\theta_k|Y)$ will also be an exponential pdf. Note that Variational Bayesian techniques can also be applied in the presence of deterministic unknowns θ_D. There are two ways to think about deterministic unknowns:

(i) as truly deterministic, with prior $p(\theta_D) = \delta(\theta_D - \theta_D^o)$ where θ_D^o is the unknown true value of θ_D; in other words, $\theta_D \sim \mathcal{N}(\theta_D^o, R_{\theta_D})$ where $R_{\theta_D} = 0\, I$.

(ii) as random with no prior information, hence $\theta_D \sim \mathcal{N}(\theta_D^o, R_{\theta_D})$ where $R_{\theta_D} = \infty\, I$.

In case (i), VB becomes EM [6], in which case during the iterations the deterministic parameters are simply substituted by their current estimate.
Case (ii) is closer to the VB spirit. If $\theta = \{\theta_D, \theta_S\}$ where θ_S are the stochastic parameters, then it suffices to replace $p(Y,\theta)$ in (9) by $p(Y,\theta_S|\theta_D) = p(Y|\theta)\, p(\theta_S)$. In this case also for the deterministic parameters not only their current estimates (posterior means) are accounted for but also their estimation error.

To summarize, EM is a special case of VB, with 2 subsets of parameters (stochastic and deterministic). Note that in the VB context the difference between EM and SAGE algorithm is the splitting of the subsets.

V. Variational Bayesian BSS

The overall set of parameters contains the following subsets (source, short term and long term parameters):

$$\theta = [\theta_1^T \cdots \theta_k^T\ \lambda_v]^T \tag{10}$$

$$\theta_k = [\ \mathbf{a}_k\ \varphi_{\mathbf{k}}\ \mathbf{x}_{\mathbf{k}}\ ]^{\mathbf{T}} \tag{11}$$

$$\mathbf{a}_k = [a_{k,1} \cdots a_{k,p_k}]^T \tag{12}$$

$$\varphi_{\mathbf{k}} = [\ b_k\ \tau_k\ \lambda_k\ ]^T \tag{13}$$

where $\lambda_k = 1/\sigma_k^2$ and $\lambda_v = 1/\sigma_v^2$ are the inverse variances or precisions. The prior probability distributions for the various parameters are chosen as follows. Let ψ be any of the groups $\{\mathbf{x}_k,\ k = 1,\ldots,K\}$, $\mathbf{a}_k$, $\varphi_k \setminus \lambda_k$. Then for any such subset of parameters ψ and for the λ_k, λ_v, the priors are chosen as

$$p(\psi) = \mathcal{N}(m_\psi, C_\psi) \tag{14}$$

$$p(\lambda_v) = \mathit{Exponential}(m_{\lambda_v}) \tag{15}$$

$$p(\lambda_k) = \mathit{Exponential}(m_{\lambda_k})\ . \tag{16}$$

With this choice of prior distributions, the posterior distributions obtained by VB will be of the same nature (Gaussian or Exponential). However, in this paper we shall consider a further simplification.

VI. Algorithm

We shall simplify the VB approach by splitting the overall parameters θ into two groups: the sources $\{\mathbf{x}_k,\ k = 1,\ldots,K\}$ on the one hand, and all AR and noise parameters on the other hand. Whereas the first group shall be treated as random, the second group shall be treated as deterministic (negligible variability, delta function posterior distribution). The resulting iterative algorithm leads to an EM-style algorithm consisting of two steps, the estimation of the sources (E-Step) and the parameters (M-Step). First an estimate of the sources $\mathbf{x}_k$ is obtained from the noisy observations, $\mathbf{y}$, with a fixed interval Wiener filter (instead of a Kalman filter as in [8]). Second, the noises variance, the short and long-term AR parameters are estimated based on the estimated source correlations.

A. Estimating the Sources

We shall estimate the sources $\mathbf{x}_k$ jointly, hence consider $\underline{\mathbf{x}} = [\mathbf{x}_1^T \cdots \mathbf{x}_k^T]^T$. Iterative estimation of the separate sources will only lead to a polynomial expansion style iterative solving of the Wiener estimation equation for $\underline{\mathbf{x}}$. This would slow down convergence, but also reduce computational complexity. It would only potentially improve performance if some non-Gaussianity is introduced and exploited.

Consider now also the following notation:
$\underline{\mathbf{W}} = \oplus_{k=1}^{K} \mathbf{W} = I_K \otimes \mathbf{W}, \underline{\mathbf{I}} = [I_N \ldots I_N] = \mathbf{1}_k^T \otimes I_N$,
$\underline{\mathbf{A}} = \oplus_{k=1}^{K} \mathbf{A}_k =$ blockdiag$\{\mathbf{A}_1,\ldots,\mathbf{A}_K\}$,
$\underline{\mathbf{B}} = \oplus_{k=1}^{K} \mathbf{B}_k =$ blockdiag$\{\mathbf{B}_1,\ldots,\mathbf{B}_K\}$,
$\underline{\Lambda} = \oplus_{k=1}^{K} \lambda_k I_N = \Lambda \otimes I_N,\ \Lambda = \mathit{diag}\{\lambda_1,\ldots,\lambda_k\}$,
$\underline{\mathbf{x}}' = \underline{\mathbf{W}}\, \underline{\mathbf{x}},\ \underline{\Lambda}' = \underline{\mathbf{W}}^{-1} \underline{\Lambda \mathbf{W}}^{-1} = \Lambda \otimes \mathbf{W}^{-2}$ and
$\underline{\mathbf{e}} = [\mathbf{e}_1^T \cdots \mathbf{e}_K^T]^T$.
With this notation, the signal model can be written as

$$\mathbf{W}\mathbf{y} = \underline{\mathbf{I}}\ \underline{\mathbf{x}}' + \mathbf{W}\ \mathbf{v},\ \ \underline{\mathbf{A}}\ \underline{\mathbf{B}}\ \underline{\mathbf{x}}' = \underline{\mathbf{W}}\ \underline{\mathbf{e}}. \tag{17}$$

with circulant $\mathbf{A}_k$, $\mathbf{B}_k$. We get the Gaussian

$$\begin{array}{l}
-2\ln\ p(\mathbf{y},\underline{\mathbf{x}}|\theta\setminus\underline{\mathbf{x}}) = \\
\lambda_v(\mathbf{W}\mathbf{y}-\underline{\mathbf{I}}\ \underline{\mathbf{x}}')^T\mathbf{W}^{-2}(\mathbf{W}\mathbf{y}-\underline{\mathbf{I}}\ \underline{\mathbf{x}}') + \underline{\mathbf{x}}'^T\left[\underline{\mathbf{B}}^T\underline{\mathbf{A}}^T\underline{\Lambda}'\underline{\mathbf{A}}\,\underline{\mathbf{B}}\right]\underline{\mathbf{x}}' \\
= c + (\underline{\mathbf{x}}' - m_{\underline{\mathbf{x}}'})^T C^{-1}_{\underline{\mathbf{x}}'\underline{\mathbf{x}}'}(\underline{\mathbf{x}}' - m_{\underline{\mathbf{x}}'})\ .
\end{array} \tag{18}$$

Averaging this over the parameters $\theta\setminus\underline{\mathbf{x}}$ now simply means evaluating at the latest estimates of these parameters, since they are considered deterministic in the simplification. We get from (18), after introducing the auxiliary quantities

$$\begin{array}{rcl}
\mathbf{C} & = & \underline{\mathbf{B}}^T\underline{\mathbf{A}}^T\underline{\Lambda}'\underline{\mathbf{A}}\,\underline{\mathbf{B}} \\
 & = & \text{blockdiag}\{\lambda_k\mathbf{B}_k^T\mathbf{A}_k^T\mathbf{W}^{-2}\mathbf{A}_k\mathbf{B}_k\}_{k=1}^K \\
\mathbf{D} & = & \frac{1}{\lambda_v}\mathbf{W}^2 + \underline{\mathbf{I}}\mathbf{C}^{-1}\underline{\mathbf{I}}^T \\
 & = & \frac{1}{\lambda_v}\mathbf{W}^2 + \sum_k \frac{1}{\lambda_k}\mathbf{B}_k^{-1}\mathbf{A}_k^{-1}\mathbf{W}^2\mathbf{A}_k^{-T}\mathbf{B}_k^{-T}
\end{array} \tag{19}$$

we get

$$\begin{array}{rcl}
C_{\underline{\mathbf{x}}'} & = & (\lambda_v\underline{\mathbf{I}}^T\mathbf{W}^{-2}\underline{\mathbf{I}} + \mathbf{C})^{-1} \\
 & = & \mathbf{C}^{-1} - \mathbf{C}^{-1}\underline{\mathbf{I}}^T\mathbf{D}^{-1}\underline{\mathbf{I}}\mathbf{C}^{-1} \\
m_{\underline{\mathbf{x}}'} & = & \mathbf{C}^{-1}\underline{\mathbf{I}}^T\mathbf{D}^{-1}\mathbf{W}\mathbf{y}\,.
\end{array} \tag{20}$$

To implement this in the frequency domain, consider the diagonal $\breve{\mathbf{A}}_k = \mathbf{F}\mathbf{A}_k\mathbf{F}^{-1}$ etc. The only non-diagonal matrix is $\breve{\mathbf{W}}_2 = \mathbf{F}\mathbf{W}^2\mathbf{F}^{-1}$ which, due to careful window design, can be approximated by a banded matrix as discussed earlier. As a result, $\breve{\mathbf{C}}^{-1}$ and $\breve{\mathbf{D}}$ are equally banded matrices Now consider the LDU factorization

$$\begin{array}{l}
\mathbf{F}\mathbf{D}\mathbf{F}^{-1} = \mathbf{F}\left[\dfrac{1}{\lambda_v}\mathbf{W}^2 + \displaystyle\sum_k \dfrac{1}{\lambda_k}\mathbf{B}_k^{-1}\mathbf{A}_k^{-1}\mathbf{W}^2\mathbf{A}_k^{-T}\mathbf{B}_k^{-T}\right]\mathbf{F}^{-1} \\
= \dfrac{1}{\lambda_v}\breve{\mathbf{W}}_2 + \displaystyle\sum_k \dfrac{1}{\lambda_k}\breve{\mathbf{B}}_k^{-1}\breve{\mathbf{A}}_k^{-1}\breve{\mathbf{W}}_2\breve{\mathbf{A}}_k^{-H}\breve{\mathbf{B}}_k^{-H}\ =\ \mathbf{L}\mathbf{D}\mathbf{L}^H
\end{array} \tag{21}$$

where the unit diagonal lower triangular $\mathbf{L}$ is banded. The steps for computing $m_{\underline{\mathbf{x}}'}$ in the frequency domain are now:

- $\breve{\mathbf{y}}\ =\ \mathbf{F}\ \mathbf{W}\ \mathbf{y}$
- solve $\breve{\mathbf{u}}$ from $\mathbf{L}\breve{\mathbf{u}} = \breve{\mathbf{y}}$ by backsubstitution
- solve $\breve{\mathbf{z}}$ from $\mathbf{L}^H\breve{\mathbf{z}} = \mathbf{D}^{-1}\breve{\mathbf{u}}$ by backsubstitution
- $m_{\mathbf{x}'_k}\ =\ \frac{1}{\lambda_k}\ \mathbf{F}^{-1}\ \breve{\mathbf{B}}_k^{-1}\breve{\mathbf{A}}_k^{-1}\breve{\mathbf{W}}_2\breve{\mathbf{A}}_k^{-H}\breve{\mathbf{B}}_k^{-H}\ \breve{\mathbf{z}}$, each time multiplying a vector with a matrix and ending with IDFT and scaling.

In practice all operations with the Discrete Fourier Transform (DFT) matrix **F** are done by using the Fast Fourier Transform algorithm (FFT). As $\breve{\mathbf{B}}_k = \text{diag}\{\breve{\mathbf{b}}_k\}$ and $\breve{\mathbf{A}}_k = \text{diag}\{\breve{\mathbf{a}}_k\}$, we can write

$$\breve{\mathbf{B}}_k^{-1}\breve{\mathbf{A}}_k^{-1}\breve{\mathbf{W}}_2\breve{\mathbf{A}}_k^{-H}\breve{\mathbf{B}}_k^{-H}\ =\ \frac{1}{\breve{\mathbf{a}}_k}\frac{1}{\breve{\mathbf{a}}_k^H}\odot\frac{1}{\breve{\mathbf{b}}_k}\frac{1}{\breve{\mathbf{b}}_k^H}\odot\breve{\mathbf{W}}_2\ . \tag{22}$$

B. Updating the AR parameters

Given the Gaussian posterior of the sources $\underline{\mathbf{x}}$, the estimation of the AR parameters of the different sources is in principle coupled ($C_{\underline{\mathbf{x}}'}$ is not block diagonal) but we shall decouple their estimation. The estimation of short-term and long-term AR parameters for a given source is coupled also. Many updating schedules are possible, e.g. iterating between short-term and long-term parameters before returning to the updating of the source statistics. We get for source k

$$\begin{array}{l}
-2\,\mathrm{E}_{q(\mathbf{x}'_k)}\ln\ p(\mathbf{x}_k|\mathbf{a}_k,\varphi_k) \\
= c - N\ln\lambda_k + \lambda_k\,\mathrm{E}_{q(\mathbf{x}'_k)}\mathbf{x}'^T_k\mathbf{B}_k^T\mathbf{A}_k^T\mathbf{W}^{-2}\mathbf{A}_k\,\mathbf{B}_k\mathbf{x}'_k \\
= c - N\ln\lambda_k + \lambda_k\,\mathrm{tr}\{\mathbf{W}^{-2}\mathbf{A}_k\,\mathbf{B}_k\mathbf{R}_k\mathbf{B}_k^T\mathbf{A}_k^T\}
\end{array} \tag{23}$$

where

$$\mathbf{R}_k =\ \mathrm{E}_{q(\mathbf{x}'_k)}\mathbf{x}'_k\mathbf{x}'^T_k = m_{\mathbf{x}'_k}m^T_{\mathbf{x}'_k} + C_{\mathbf{x}'_k} \tag{24}$$

which are obtained from (20). After optimizing the $\mathbf{a}_k$, b_k, τ_k , one can find by minimizing (23)

$$\lambda_k = N/\ \mathrm{tr}\{\mathbf{W}^{-2}\mathbf{A}_k\,\mathbf{B}_k\mathbf{R}_k\mathbf{B}_k^T\mathbf{A}_k^T\} \tag{25}$$

while the $\mathbf{a}_k$ are obtained by minimizing

$$\mathrm{tr}\{\mathbf{W}^{-2}\mathbf{A}_k\,(\mathbf{B}_k\mathbf{R}_k\mathbf{B}_k^T)\mathbf{A}_k^T\} \tag{26}$$

for fixed b_k, τ_k, and the b_k, τ_k themselves are obtained by minimizing

$$\mathrm{tr}\{\mathbf{W}^{-2}\mathbf{B}_k\,(\mathbf{A}_k\mathbf{R}_k\mathbf{A}_k^T)\mathbf{B}_k^T\} \tag{27}$$

for fixed $\mathbf{a}_k$ (in a full VB approach, this quadratic form, e.g. in $\mathbf{a}_k$, would have to be identified with the exponent of a Gaussian pdf in order to find both mean and covariance). At this point, one might remark that the limited degree of nonstationarity of the signals leads to slow variation of the source statistics in time. Hence, in all this the instantaneous $\mathbf{R}_k$ for a given frame may be advantageously replaced by an exponentially weighted average of the $\mathbf{R}_k$ of the past frames. The time constant of the exponential weighting factor may be adjusted according to the degree of nonstationarity, which may be inferred by focusing mainly on the time variation of the long-term AR model parameters.

Alternatively, at this point one may consider pushing back the window into the source statistics by considering the cost function

$$\mathrm{tr}\{\mathbf{A}_k\,\mathbf{B}_k(\mathbf{W}^{-1}\mathbf{R}_k\mathbf{W}^{-1})\mathbf{B}_k^T\mathbf{A}_k^T\} \tag{28}$$

where for the mean, the unwindowed $\mathbf{W}^{-1}m_{\mathbf{x}'_k}$ may be advantageously replaced by the reconstructed multiframe source signal, still with the resulting sample correlations exponentially weighted into the past. For the source estimation error covariance part $\mathbf{W}^{-1}C_{\mathbf{x}'_k}\mathbf{W}^{-1}$, if not ignored completely, may be approximated by the expression without window:

$$\begin{array}{l}
C_{\mathbf{x}_k} = C_k^{-1} - C_k^{-1}(\frac{1}{\lambda_v}I_N + \sum_{i=1}^K C_i^{-1})^{-1}C_k^{-1} \\
\text{with } C_k = \lambda_k\mathbf{B}_k^T\mathbf{A}_k^T\mathbf{A}_k\mathbf{B}_k
\end{array} \tag{29}$$

in which all matrices are circulant, hence the computation is straightforward in the frequency domain. The values for the AR parameters to be used in the computation of $C_{\mathbf{x}_k}$ are those that were used for the computation of $m_{\mathbf{x}'_k}$.

Finally, the estimation of the overall noise variance can be obtained similarly as

$$\frac{1}{\lambda_v} = \frac{1}{N}||\mathbf{y} - \mathbf{W}^{-1}\underline{\mathbf{I}}m_{\underline{\mathbf{x}}'}||^2 + \frac{1}{N}\,\mathrm{tr}\{\mathbf{W}^{-2}\underline{\mathbf{I}}C_{\underline{\mathbf{x}}'}\underline{\mathbf{I}}^T\} \tag{30}$$

where the same multi-frame averaging and approximations are applicable.

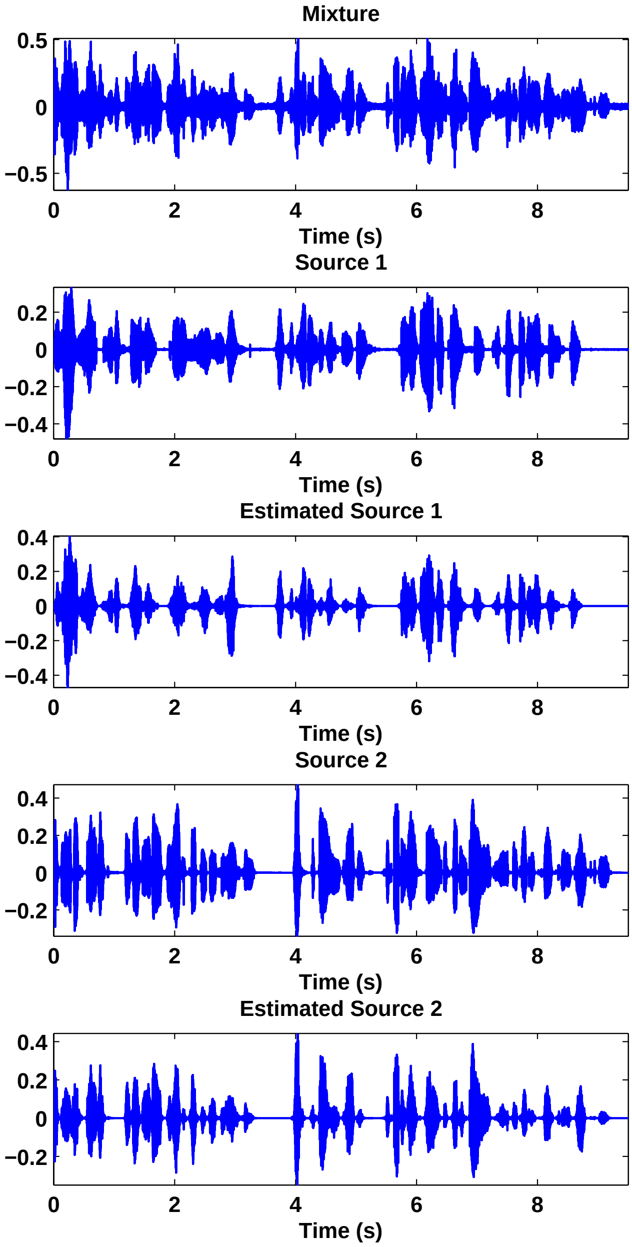

Fig. 2. Waveform of the mixture, sources and estimated sources.

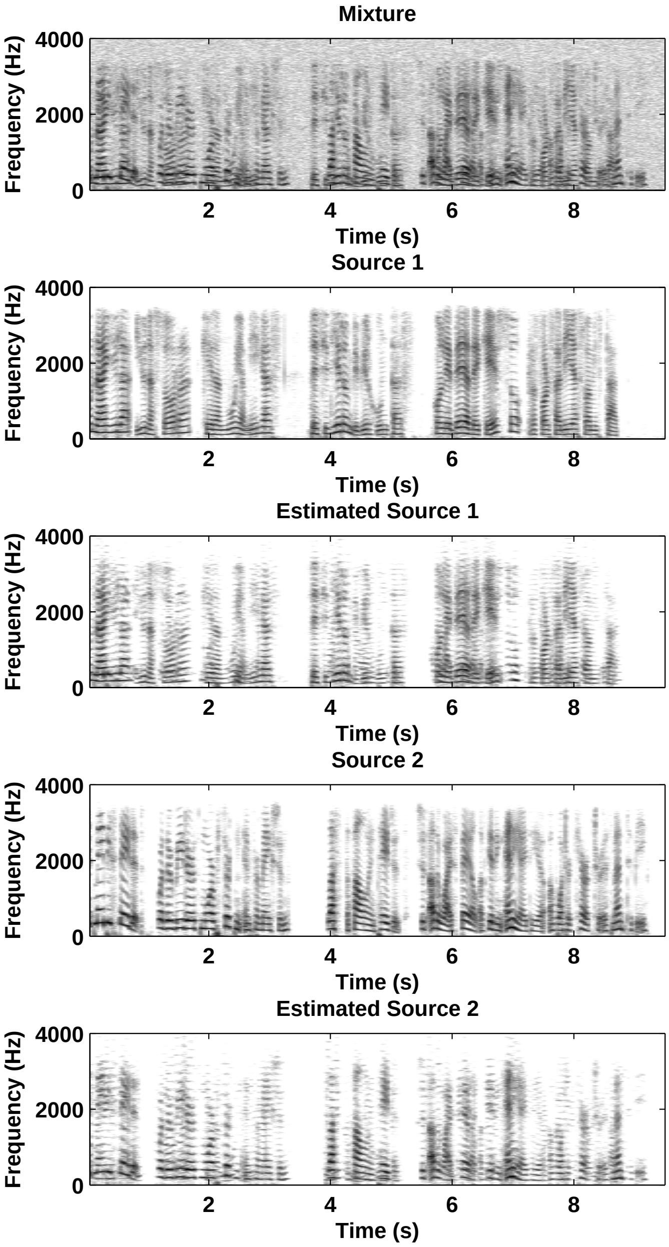

Fig. 3. Spectrogram of the mixture, sources and estimated sources.

C. *Initalization and Tracking*

When moving from one frame to the next, the AR and noise parameters from the previous frame can be used as initialization for the current frame. For the cold start, or when a new source appears or reappears after a silence, the algorithm needs initialization, mainly for the long-term AR parameters. For this any multipitch estimation algorithm can be used.

VII. Results

For testing the algorithm we have worked with real speech data (the sources), the mixing and noise adding are done artificially. The sources consist of two speech recordings of $10\ s$, a male and a female English speaker. The analysis window length is $64\ ms$ with an overlap of 50%. The (cold) initialization of the parameters $\mathbf{a}_k$, b_k and λ_k is done on the original sources (yielding the ”correct” values). The algorithm is stopped when the variation between two consecutive iterations is lower than 10^{-3} or when the algorithm reaches 20 iterations. Fig 2 and 3 show the results of the decomposition for the separation, with an SNR of $20dB$. The estimated parameters are close to the correct ones, measured on the original sources with a chosen order for the short term coefficients. Note that the signals contains silence segments, where we cannot build on the estimated parameters of the previous frame, we need to re-initialize them.

Fig 4 shows a zoom on a frame. One can see the automatic appearance of a windowed version of the extracted sources. In this particular frame the two pitch periods are very different, which faciliates the separation.

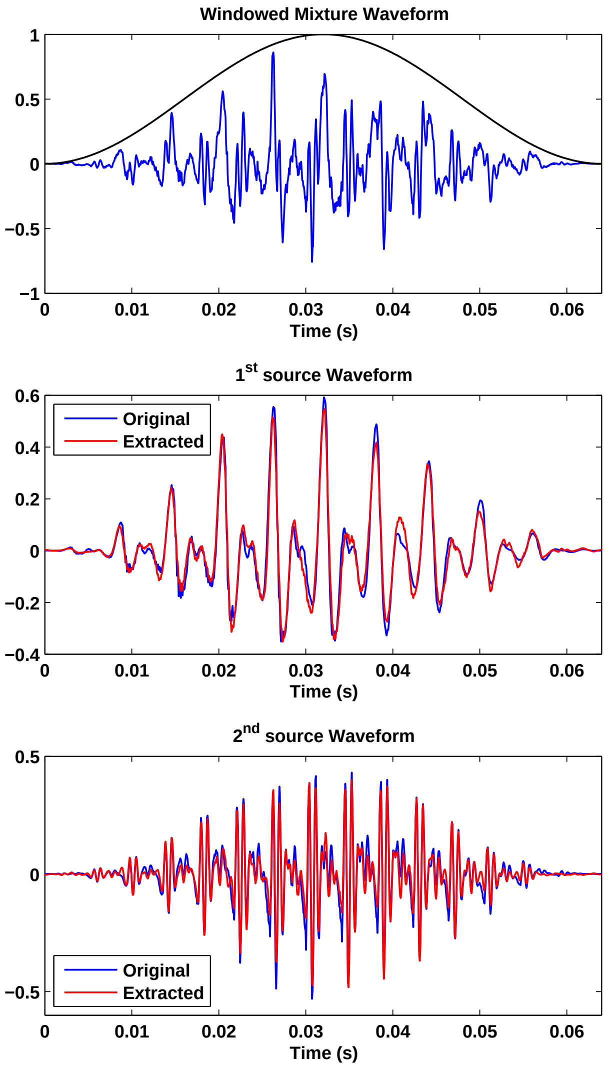

Fig. 4. Zoom on a frame

A second simulation consists in estimating the (output) Signal to Noise Ratio (SNR). We apply the algorithm on a mixture of two sources (two segments of length $64\ ms$), and we vary the (input) SNR. The (output) noise is determined by subtracting the estimated sources from the noisy mixture (and contains at least the input noise). The output SNR is defined by $SNR_{est} = 10\ log_{10}\left(\frac{\sum_t \|\sum_k \hat{\mathbf{x}}_k(t)\|^2}{\sum_t \|y(t)-\sum_k \hat{\mathbf{x}}_k(t)\|^2}\right)$. If the algorithm works well, the output SNR increases up to the input SNR. This is observed in the results obtained as shown in Fig 5.

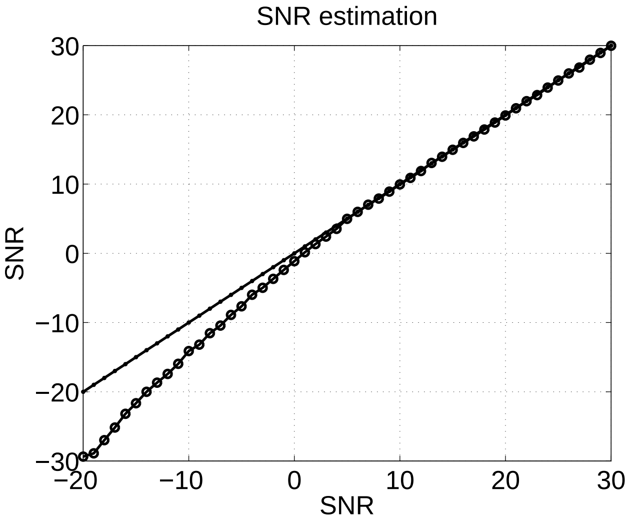

Fig. 5. SNR estimation for a mixture of 2 sources

VIII. Conclusions

In this paper we have proposed a VB-EM type algorithm for blind source separation. The long term correlation allows identifiability of the sources which, in the case of unvoiced speech segments, is maintained by the short term AR model. We have in particular introduced a more rigorous use of frequency domain processing via the introduction of carefully designed windows. The results for BASS are encouraging. Further extensions could include the determination of the number of sources. Also, multipitch estimation is required at initialization and at any reappearance of non-stationary sources.

References

[1] A.Hyvarinen, "Survey on independent component analysis," *Neural Computing Surveys, Vol. 2, pp. 94-128*, 1999.

[2] M.A. Casey, "Separation of mixed audio sources by independent subspaces analysis," *int. Computer Music Conference, Berlin, August*, 2000.

[3] D.F Rosenthal, H.G Okuno, "Computational auditory scene analysis," *LEA Publishers, Mahwah NJ*, 1998.

[4] Wai. C. Chu, *Speech coding algorithms-foundation and evolution of standardized coders*, John Wiley and Sons, NewYork, 2003.

[5] A.V Oppenheim and R.W Schafer, "Discrete-time signal processing," pp., 447–448, 1989.

[6] D.G. Tzikas and A.C. Likas and N.P. Galatsanos, "The Variational Approximation for Bayesian Inference, Life After the EM Algorithm," *IEEE Signal Processing Magazine*, Nov. 2008.

[7] M. Beal, "Variational algorithms for approximate bayesian inference," *Ph.D. Thesis, Gatsby Computational Neuroscience Unit, Univ. College London*, 2003.

[8] C.Couvreur and Y.Bresler, "Decomposition of a mixture of gaussian ar processes," *Proc. IEEE Int. Conf. Acous. Speech Signal Process*, 1995.