Proceeding of the IASTED International Conference
Signal and Image Processing
October 27-31, 1998 Las Vegas, USA

A new Blind Signal Separation Algorithm based on Second Order Statistics

D.W.E. Schobben and *P.C.W. Sommen*

Electrical Engineering Department
Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven, the Netherlands

`tel. +31/402472366 fax. +31/402455674`
`D.W.E.Schobben@ele.tue.nl and P.C.W.Sommen@ele.tue.nl`
`Keywords : Blind Signal Separation`

Abstract

This paper addresses the problem of separating multiple speakers from mixtures of these that are obtained using multiple microphones in a room. A new adaptive blind signal separation algorithm is derived which is entirely based on Second Order Statistics, which is entitled 'CoBliSS'. The CoBliSS algorithm can run in offline or online (adaptive) mode. One of the advantages of the CoBliSS algorithm is that no assumptions are made about the probability density functions or other properties of the signals. Experiments with real recordings were carried out in a normal living room, which show that the algorithm has good performance. As opposed to most other algorithms, no parameters need to be tuned.

1 Introduction

Blind Signal Separation (BSS) deals with the problem of recovering independent signals using only observed mixtures of these. For acoustical applications, these observed mixtures are signals of multiple microphones. In this context a convolutive separation algorithm is used, i.e. the separation consists of employing Multi-Channel Finite Impulse Response (MC-FIR) filtering to these signals. Recently, several algorithms have been developed for convolutive separation, e.g. [1, 2, 3]. Some authors claim that Second Order Statistics (SOS) are insufficient to achieve BSS [4, 5]. Most of the algorithms therefore make use of Higher Order Statistics (HOS). Among others, mutual information and maximum likelihood approaches are followed. The HOS algorithms contain non-linear elements that can be tuned to the data in order to obtain a good performance. In this paper, a new BSS algorithm is presented, which is based only on SOS and does not require any parameters to be tuned.

The remainder of this paper is outlined as follows. In Section 3 the optimization criterion that will be used in the BSS algorithm is described. The optimization is done by minimizing the crosscorrelations among the outputs of the multi-channel separating filter. In order to achieve a computationally inexpensive algorithm with fast convergence, this criterion is transformed to the frequency domain in Section 4. The filter coefficients are calculated in the frequency domain such that the crosscorrelations become equal to zero. No restrictions are imposed however to ensure that the filter coefficients correspond to real filters of a given length in the time domain. This is discussed in Section 5 and a method is suggested to remedy this problem. After applying this method, the crosscorrelations are no longer zero. As there are two sets of constraints in two different domains an iterative method is proposed in which the weights are adjusted iteratively in alternately one and the other domain. The key issue to obtain a good performance in terms of separation and convergence is to find a suitable adaptation in the frequency domain which leaves the time domain constraint as much intact as possible. This Time-Frequency Domain Compliance is discussed in Section 6. In order to prevent the algorithm from whitening the signals, a normalization must be applied. This is discussed in Section 7. The previously mentioned building blocks together form the new Convolutive Blind Signal Separation algorithm; CoBliSS, which is summarized in Section 8. CoBliSS has been tested using people that are speaking recorded in a room. These experimental results are discussed in Section 9. The paper concludes with conclusions and future work.

2 Notation

Throughout, time and frequency signals will be denoted by lower case and upper case characters respectively. A character which denotes a vector will be underlined. Superscripts denote the vector or matrix dimensions, a matrix with one superscript is square. Also, A^*, A^T and A^{-1} denote complex conjugate, matrix transpose and matrix inverse respectively and $\jmath^2 = -1$. Element-wise multiplication is denoted by $\otimes$. The expectation operator will be denoted by $E\{.\}$. The $N \times N$ identity matrix and the $K \times L$ zero matrix will be denoted by $\mathbf{I}^N$ and $\mathbf{0}^{K,L}$ respectively. The $M \times M$ Fourier matrix $\mathcal{F}^M$ is defined as $(\mathcal{F}^M)_{kl} = e^{\frac{-2\jmath\pi kl}{M}}$ and diag$\{.\}$ converts the elements on the diagonal of a matrix to a vector.

3 Optimization Criterion

The MC-FIR separation filter will be controlled by an algorithm that minimizes cross-correlations among the outputs of this filter. The notation is in accordance to Fig. 1 which depicts the mixing/unmixing system. The independent sources $s_1 \dots s_J$ are mixed by the mixing system H to obtain the sensor signals $x_1 \dots x_J$. Throughout, both the number of sources and the number of sensors are equal to J. Time indexes are not mentioned explicitly in all formulas. The

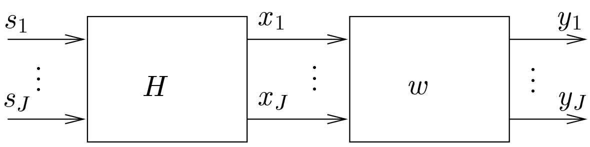

Figure 1: Cascaded mixing/unmixing system

transfer function from the l^{th} input to the m^{th} output of the separation filter is denoted as $\underline{w}_{ml}^N$. The m^{th} output of the separation filter y_m is calculated from the observations $\underline{x}_l^N$

$$y_m[n] = \sum_{l=1}^{J} (\underline{w}_{ml}^N)^T \underline{x}_l^N[n].$$

with J the number of microphones, N the filter length (all filters are of equal length for simplicity),

$$\underline{w}_{ml}^N = \begin{pmatrix} w_{ml}[N-1] \\ \vdots \\ w_{ml}[0] \end{pmatrix} \text{ and } \underline{x}_l^N[n] = \begin{pmatrix} x_l[n-N+1] \\ \vdots \\ x_l[n] \end{pmatrix}$$

The cross-correlation among the outputs can be written as [1]

$$\begin{aligned} r_{y_i y_j}[l] &= E\{y_i[n] y_j[n+l]\} \\ &= \sum_{a=1}^{J}\sum_{c=1}^{J}\sum_{b=0}^{N-1}\sum_{d=0}^{N-1} w_{ia}[b] w_{jc}[d] r_{x_a x_c}[l+b-d] \quad (1) \end{aligned}$$

with $r_{x_a x_c}[l] = E\{x_a[n] x_c[n+l]\}$. The advantage of this expression is that it can be used to optimize the filters w_{ia} using the cross-correlations of the observed data. These crosscorrelations do not do not depend on the separation filters so that they do not need to be estimated again every time the separation filter is updated.

Next, a cost function can be formed directly from (1) using for example the sum of squares of the cross-correlation coefficients. The straightforward minimization of such a cost function is not eligible however due to the large number of filter coefficients involved. A typical example is that two sources and two microphones are used. In that case 4 FIR filters need to be calculated, each with several hundreds to thousands coefficients. Furthermore, all these coefficients are dependent on each other which makes the problem even more difficult. Therefore, an approach is required that solves for filter coefficients subsets which are as independent of each other as possible. In order to achieve this, (1) is transformed to the frequency domain.

The cross-correlations are stacked in a vector for all lags that are considered $l = l_1 \dots l_2$

$$\underline{r}_{y_i y_j}^L = \sum_{a=1}^{J}\sum_{c=1}^{J} R_{ac}^{L,2N-1} A_{jc}^{2N-1,N} \underline{w}_{ia}^N \qquad (2)$$

with $L = l_2 - l_1 + 1$, $\underline{r}_{y_i y_j}^L = \left(r_{y_i y_j}[l_1] \dots r_{y_i y_j}[l_2]\right)^T$ and

$$A_{jc}^{2N-1,N} = \begin{pmatrix} w_{jc}[0] & 0 & \dots & 0 \\ \vdots & \ddots & & \vdots \\ w_{jc}[N-1] & & \ddots & 0 \\ 0 & \ddots & & w_{jc}[0] \\ \vdots & & \ddots & \vdots \\ 0 & \dots & 0 & w_{jc}[N-1] \end{pmatrix}$$

$$R_{ac}^{L,2N-1} = \begin{pmatrix} r_{x_a x_c}[l_1+N-1] & \dots & r_{x_a x_c}[l_1-N+1] \\ \vdots & \ddots & \vdots \\ r_{x_a x_c}[l_2+N-1] & \dots & r_{x_a x_c}[l_2-N+1] \end{pmatrix}$$

Solutions for the MC-FIR are guaranteed to be nonambiguous if $l_1 \leq -N+1$ and $l_2 \geq N-1$. In the sequel, $l_1 = -N+1$ and $l_2 = N-1$ so that (2) can be written as

$$\underline{r}_{y_i y_j}^L = \sum_{a=1}^{J}\sum_{c=1}^{J} (\mathbf{I}^L \mathbf{0}^{L-M,M}) R_{ac}^M \breve{A}_{jc}^M \underline{\dot{w}}_{ia}^M, \qquad (3)$$

with $M = L = 2N-1$, $\underline{\dot{w}}_{ia}^M = \begin{pmatrix} \underline{w}_{ia}^N \\ \underline{\mathbf{0}}^{M-N} \end{pmatrix}$ and $\breve{A}_{jc}^M$ is formed by extending $A_{jc}^{2N-1,N}$ on the right such that it becomes circulant

$$\breve{A}_{jc}^{M}=\begin{pmatrix} w_{jc}[0] & 0 & \dots & 0 & \ddots & \vdots \\ \vdots & \ddots & & \vdots & & w_{jc}[N-1] \\ w_{jc}[N-1] & \ddots & \ddots & 0 & & 0 \\ 0 & \ddots & \ddots & w_{jc}[0] & & \vdots \\ \vdots & & \ddots & \vdots & \ddots & 0 \\ 0 & \dots & 0 & w_{jc}[N-1] & \dots & w_{jc}[0] \end{pmatrix}$$

Next, the cross correlation matrix R_{ac}^{M} is approximated by its circulant variant $\breve{R}_{ac}^{M}=E\{\breve{\mathcal{X}}_a^{M}(\breve{\mathcal{X}}_c^{M})^T\}$, with $\breve{\mathcal{X}}_l^{M}$ the circulant data matrix

$$\breve{\mathcal{X}}_l^{M}[\eta B]=\begin{pmatrix} x_l[\eta B-M+1] & x_l[\eta B] & \dots & x_l[\eta B-M+2] \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & x_l[\eta B] \\ x_l[\eta B] & \dots & \dots & x_l[\eta B-M+1] \end{pmatrix}$$

4 Frequency Domain Approach

In this section the crosscorrelation expression in (3) is transformed to the frequency domain and a first step is made towards solving the BSS problem. Replacing R_{ac}^{M} by it's circulant approximation in (3) makes it possible to diagonalize the matrices in this equation using FFT's [6]. This is done by inserting the identity matrix $(\mathcal{F}^M)^{-1}\mathcal{F}^M$ in between all matrices which results in

$$\begin{aligned} \underline{r}_{y_iy_j}^{L} &= (\mathbf{I}^L\mathbf{0}^{L,L-M})\sum_{a=1}^{J}\sum_{c=1}^{J}(\mathcal{F}^M)^{-1} \\ &\cdot \left(\underline{\breve{R}}_{ac}^{M}\otimes\underline{\breve{W}}_{jc}^{M}\otimes(\underline{\breve{W}}_{ia}^{M})^{*}\otimes(\underline{V}^{M})^{N-1}\right) \quad (4) \end{aligned}$$

with

$$\begin{aligned} \underline{\breve{R}}_{ac}^{M} &= \operatorname{diag}\left\{\mathcal{F}^M\breve{R}_{ac}^{M}(\mathcal{F}^M)^{-1}\right\} \\ \underline{\breve{W}}_{jc}^{M} &= \mathcal{F}^M\begin{pmatrix}\mathbf{J}^N\underline{w}_{jc}^{N} \\ \mathbf{0}^{M-N}\end{pmatrix} \\ \underline{V}^{M} &= \begin{pmatrix}1 & e^{\jmath\frac{2\pi}{M}} & \dots & e^{\jmath\frac{2\pi(M-1)}{M}}\end{pmatrix}^T \end{aligned}$$

with $\mathbf{J}^N$ the $N\times N$ mirror matrix which has ones on its anti-diagonal and zeros elsewhere. Note that $\underline{V}^M$ and the complex conjugate in (4) compensate for the fact that $\underline{w}_{ia}^{N}$ is not flipped upside down in $\underline{\dot{w}}_{ia}^{M}$ as opposed to $\underline{w}_{jc}^{N}$.

Signal separation is achieved when all crosscorrelations among the outputs equal zero, i.e. $\forall i\neq j$: $\underline{r}_{y_iy_j}^{L}=\underline{\mathbf{0}}^{L}$. Using (4), a sufficient but not necessary criterion to achieve uncorrelated outputs is therefore $\forall i\neq j$

$$\sum_{a=1}^{J}\sum_{c=1}^{J}\underline{\breve{R}}_{ac}^{M}\otimes\underline{\breve{W}}_{jc}^{M}\otimes(\underline{\breve{W}}_{ia}^{M})^{*}=\underline{\mathbf{0}}^{M}. \quad (5)$$

This expression has the advantage that the frequency domain filter coefficients are no longer related by the window $(\mathbf{I}^L\mathbf{0}^{L,L-M})$; the expression is reduced to a set of scalar equations. Next an approach is followed where all the scalar equations are solved individually. The p^{th} elements of $\underline{\breve{W}}_{ij}^{M}$ and $\underline{\breve{R}}_{ij}^{M}$ are put in a matrix $\forall i,j$

$$\begin{aligned} \breve{W}_p^J &= \begin{pmatrix}(\underline{\breve{W}}_{11}^{M})_p & \dots & (\underline{\breve{W}}_{1J}^{M})_p \\ \vdots & \ddots & \vdots \\ (\underline{\breve{W}}_{J1}^{M})_p & \dots & (\underline{\breve{W}}_{JJ}^{M})_p\end{pmatrix} \\ \breve{R}_p^J &= \begin{pmatrix}(\underline{\breve{R}}_{11}^{M})_p & \dots & (\underline{\breve{R}}_{1J}^{M})_p \\ \vdots & \ddots & \vdots \\ (\underline{\breve{R}}_{J1}^{M})_p & \dots & (\underline{\breve{R}}_{JJ}^{M})_p\end{pmatrix} \end{aligned}$$

In practical situations $\breve{R}_p^J$ is full rank, so that (5) can be rewritten $\forall p$ as

$$\begin{aligned} (\breve{W}_p^J)^*\breve{R}_p^J(\breve{W}_p^J)^T &= \Lambda_p^J \\ \Leftrightarrow (\breve{W}_p^J)^T(\Lambda_p^J)^{-1}(\breve{W}_p^J)^* &= (\breve{R}_p^J)^{-1} \quad (6) \end{aligned}$$

with Λ_p^J a diagonal matrix. The off-diagonal zeros elements are due to (5) and the diagonal elements determine the autocorrelations of the outputs of the BSS in frequency bin p. Since Λ_p^J is real by definition and its inverse is also diagonal, it can be absorbed by the weight matrices. For this reason, Λ_p^J is set equal to the identity matrix from now on. The impact of this will be discussed in detail in Section 7. The $\breve{R}_p^J$ is symmetrical by definition as the circulant crosscorrelation matrices $\underline{\breve{R}}_{ac}$ are symmetrical, i.e. $\underline{\breve{R}}_{ac}=\underline{\breve{R}}_{ca}^{*}$. The inverse of this symmetric matrix $\breve{R}_p^J$ is also symmetric, so that the right hand side of (6) can be decomposed in several ways (e.g. matrix square root). In general the matrix decomposition should be different $\forall p$ in order to obtain the right solution. In practice these decompositions are unknown so that initially all $\breve{R}_p^J$ are decomposed in the same manner.

5 Convolution Constraint

Now that (6) is solved independently $\forall p$, the following no longer holds

$$(\mathcal{F}^M)^{-1}\underline{\breve{W}}_{jc}^{M}=\begin{pmatrix}\underline{w}_{jc}^{N} \\ \underline{\mathbf{0}}^{M-N}\end{pmatrix}. \quad (7)$$

In other words, the frequency domain filters no longer correspond to real time domain filters of length N. The fact that the time domain filters must be real is not a problem, because the frequency domain cross correlation vectors have the same symmetric properties as the frequency domain filters. The fact that the time domain filters must be of length N is achieved by doing $\forall j, c$

$$\underline{\breve{W}}_{jc}^{M} := \mathcal{F}^{M} \begin{pmatrix} \mathbf{I}^{N} & \mathbf{0}^{N,M-N} \\ \mathbf{0}^{M-N,N} & \mathbf{0}^{M-N} \end{pmatrix} (\mathcal{F}^{M})^{-1} \underline{\breve{W}}_{jc}^{M} \tag{8}$$

This means that the filter coefficients that should be zero are set to zero in the time domain. Clearly this destroys the solution of (6) so that there is a need for compliance between the frequency domain solution and the convolution constraint.

6 Time-Frequency Compliance

The two sets of equations (6) and (7) do not have a joint solution in closed form. Therefore an iterative approach is followed. The weight matrices are initialized once so that $\breve{W}_p^T \breve{W}_p^* = \breve{R}_p^{-1}$. Then two steps must be performed which destroy these equations;

- the filters are constraint in the time domain according to (8).
- the crosscorrelation matrices $\breve{R}_p$ are updated

The key issue is to find a way to adapt the weight matrices slightly so that (6) holds again. This weight adaptation and (8) can be performed iteratively until convergence is achieved. This corresponds to finding the individual decompositions of $\breve{R}_p^J$ as discussed in the previous section. Two methods are derived for the weight update; one is exact and one is an approximated version that exhibits a low computational complexity.

6.1 Exact Weight Update

In the following derivation, all matrices are of size $J \times J$ and the corresponding superscripts will be omitted. After constraining the filters according to (7) the weight matrix product becomes $\breve{W}_p^T \breve{W}_p^* = B_p$ with $B_p \neq \breve{R}_p^{-1}$. Also, the crosscorrelation matrices are updated $\breve{R}_p \Rightarrow \breve{R}_p'$. The goal is therefore to find a matrix C_p, such that $\breve{W}_p'^T \breve{W}_p'^* = \breve{R}_p'^{-1}$ with $\breve{W}_p' = \breve{W}_p C_p$. The matrix C_p must be near to the matrix identity when B_p is near to $\breve{R}_p'^{-1}$. In this way, the previous solution is preserved as much as possible and therefore fast convergence is ensured. Using the decomposed matrices

$$\begin{aligned} D_p = \mathrm{sqrtm}(B_p)^* &\quad\Leftrightarrow\quad D_p^T D_p^* = B_p \\ D_p' = \mathrm{sqrtm}(R_p'^{-1})^* &\quad\Leftrightarrow\quad D_p'^T D_p'^* = \breve{R}_p'^{-1} \end{aligned} \tag{9}$$

with sqrtm(.) the matrix square root, i.e. $A = \mathrm{sqrtm}(B) \Leftrightarrow A^H A = B$, such that $A^H = A$, with B a complex symmetric matrix, i.e. $B^H = B$. The transform matrix C_p can be found from

$$\begin{aligned} B_p &= D_p^T (D_p'^T)^{-1} \breve{R}_p'^{-1} (D_p'^*)^{-1} D_p^* \\ \Leftrightarrow \breve{W}_p^T \breve{W}_p^* &= D_p^T (D_p'^T)^{-1} \breve{W}_p'^T \breve{W}_p'^* (D_p'^*)^{-1} D_p^* \\ \Leftrightarrow \quad \breve{W}_p &= \breve{W}_p' (D_p')^{-1} D_p \\ \Leftrightarrow \quad \breve{W}_p' &= \breve{W}_p (D_p)^{-1} D_p'. \end{aligned} \tag{10}$$

So, $C_p = (\mathrm{sqrtm}(B_p)^*)^{-1} \mathrm{sqrtm}(\breve{R}_p'^{-1})^*$. In an offline implementation of the algorithm the crosscorrelations would first be estimated and $\mathrm{sqrtm}(\breve{R}_p^{-1})^*$ would only have to be calculated once. In an online implementation however, the crosscorrelations estimates change in time and require that the C_p is recomputed every update. The matrix square root involved is computationally demanding when there are many signals to separate (large J). In the next subsection a method is derived with a low computational complexity that is suitable for online implementation.

6.2 Fast Approximated Weight Update

A fast weight update is proposed in this subsection which does not use the matrix square root. Advantage is taken of the fact the the crosscorrelation matrices change only slowly in time.

As in the previous subsection, $\breve{W}_p^T \breve{W}_p^* \neq \breve{R}_p^{-1}$ after the time domain constraint is applied (8). Also the crosscorrelation matrices are updated so that $\breve{R}_p$ changes to $\breve{R}_p'$. Now ϵ_p must be found such that

$$\breve{W}_p'^T \breve{W}_p'^* = \breve{R}_p'^{-1} \quad \text{with} \quad \breve{W}_p' = (\mathbf{I} + \epsilon_p) \breve{W}_p \tag{11}$$

So, new weight matrices must be derived from the previous ones so that their product becomes equal to the inverse of the updated crosscorrelation matrices. Denote $\Delta \breve{R}_p = \breve{R}_p' - \breve{R}_p$ so that

$$\begin{aligned} \breve{W}_p^H (\mathbf{I} + \epsilon_p)^H (\mathbf{I} + \epsilon_p) \breve{W}_p &= (\breve{R}_p + \Delta \breve{R}_p)^{-1} \\ \breve{W}_p^H \breve{W}_p + \breve{W}_p^H (\epsilon_p^H + \epsilon_p) \breve{W}_p &\approx \breve{R}_p^{-1} - \breve{R}_p^{-1} \Delta \breve{R}_p \breve{R}_p^{-1} \\ \Leftrightarrow \breve{W}_p^H (\epsilon_p^H + \epsilon_p) \breve{W}_p &\approx -\breve{R}_p^{-1} \Delta \breve{R}_p \breve{R}_p^{-1} \\ \Leftrightarrow \epsilon_p^H + \epsilon_p &\approx -(\breve{W}_p^{-1})^H \breve{W}_p^H \breve{W}_p \Delta \breve{R}_p \breve{W}_p^H \breve{W}_p \breve{W}_p^{-1} \end{aligned} \tag{12}$$

The approximation corresponds to neglecting higher order terms of $\breve{R}_p^{-1} \Delta \breve{R}_p$ in the series expansion of $(\mathbf{I} + \breve{R}_p^{-1} \Delta \breve{R}_p)^{-1}$. Now, ϵ_p must be chosen in accordance with (12) and such that the changes to $\breve{W}_p$ are small so that fast convergence is ensured. Both the left and right hand side of (12) are symmetric by definition. It follows from the triangle inequality that the the ϵ_p with the smallest l_2 norm satisfying 12 is

$$\epsilon_p = \epsilon_p^H = -\tfrac{1}{2} \breve{W}_p \Delta \breve{R}_p \breve{W}_p^H$$

In accordance with (11) the weight update becomes

$$\begin{aligned}\breve{W}'_p &= (\mathbf{I} - \tfrac{1}{2}\breve{W}_p \Delta \breve{R}_p \breve{W}_p^H)\breve{W}_p \\ &= \breve{W}_p(\mathbf{I} - \tfrac{1}{2}\Delta \breve{R}_p \breve{W}_p^H \breve{W}_p) \qquad (13)\end{aligned}$$

7 Normalization

In Section 4 the constraint matrices Λ_p^J are set equal to the matrix identity. The impact of this is discussed in this section. The elements on the diagonal of Λ_p^J prescribe the power of the outputs of the separation filter at the corresponding frequency. First, the effect of choosing the constraint matrices equal to the matrix identity is discussed for sources that have equal energy distributions as a function of frequency. Typically for real world signals like speech the energy decays significantly for higher frequencies. When the BSS algorithm is forced to yield outputs with equal energy for all frequencies this will result in energy boosting for frequencies where the signals are weak. Also, energy will be lowered for frequencies where the signals are strong. For speech for example, this leads to unwanted signal equalization where the low frequencies are suppressed and high frequencies are boosted resulting in artificial sounding recovered speech. This problem cannot be solved directly as the ideal Λ_p^J depend on the unknown original sources. Therefore the following approach is followed. First, the $\breve{W}_p^J$ are calculated from (6) with $\Lambda_p^J = \mathbf{I}^J$. Then the weight matrices are normalized using

$$\breve{W}_p := \frac{\breve{W}_p}{||\breve{W}_p||} \qquad (14)$$

A norm that can be used and gives a good performance is the l_2 norm. The idea behind this is that all filter coefficients are of the same order of magnitude after this normalization is applied. Ideally all-pass filters are produced that leave the timbre of the signals unaffected. Another issue is that the powers of the source signals do not evolve similarly as a function of frequency. In that case unwanted equalization still occurs despite the scalar normalization. In that case a more sophisticated procedure could be followed where the Λ_p^J are estimated from the separated signals. This approach is not considered here in detail.

8 A New Adaptive Algorithm

In this section the CoBliSS algorithm is presented which consists of the building blocks discussed earlier in this paper. The adaptive procedure consists of the following steps;

1. Transform blocks of input data to the frequency domain $\forall a$:
   $$\underline{X}_a^M = \mathcal{F}^M \begin{pmatrix} x_a[nB - M + 1] \\ \vdots \\ x_a[nB] \end{pmatrix}$$
   The blocks are of length M and are overlapping; only B new samples are used per block.

2. Update crosscorrelation estimates efficiently in the frequency domain $\forall a, c$:
   $\underline{\breve{R}}_{ac}^M := \alpha \underline{\breve{R}}_{ac}^M + (1 - \alpha)\left((\underline{X}_a^M)^* \otimes \underline{X}_c^M\right)$
   The forgetting factor α may vary from 0 to 1 depending on the application. Usually α is chosen near to 1, e.g. $\alpha = 0.99$.

3. When the crosscorrelation matrices are updated several times the weights are initialized by decomposing (6) using the matrix square root
   $\forall p : \breve{W}_p^J = \mathrm{sqrtm}((\breve{R}_p^J)^{-1})^*$
   Note that $(\breve{R}_p^J)_{a,c} = (\underline{\breve{R}}_{ac}^M)_p$.

4. The weights are changed such that (6) holds again (all matrices are of size $J \times J$)
   $\forall p : \breve{W}_p := \breve{W}_p C_p$ with
   $C_p = (\mathrm{sqrtm}(\breve{W}_p^T \breve{W}_p^*)^*)^{-1} \mathrm{sqrtm}(\breve{R}_p^{-1})^*$
   Note: this step can omitted at initialization.

5. The weight matrices are normalized using
   $\breve{W}_p := \frac{\breve{W}_p}{||\breve{W}_p||}$

6. The weights are constraint according to (7) $\forall p$:
   $\underline{\breve{W}}_{jc}^M := \mathcal{F}^M \begin{pmatrix} \mathbf{I}^N & \mathbf{0}^{N,M-N} \\ \mathbf{0}^{M-N,N} & \mathbf{0}^{M-N} \end{pmatrix} (\mathcal{F}^M)^{-1} \underline{\breve{W}}_{jc}^M$
   Note that $(\breve{W}_p^J)_{ac} = (\underline{\breve{W}}_{ac}^M)_p$.

7. The filtering is performed efficiently in the frequency domain using the overlap-save method [7] to obtain the separated outputs
   $\underline{y}_j^B = (\mathbf{0}^{B,M-B} \mathbf{I}^B)(\mathcal{F}^M)^{-1} \sum_{a=1}^J (\underline{X}_a^M \otimes \underline{\breve{W}}_{ja}^M)$

8. All steps are repeated iteratively except for the initialization in item 3.

Note that the filtering and the update can be calculated independently. Reducing the update rate lowers the computational complexity at the cost of a slower convergence.

9 Experiments

Experiments were done with audio recorded in a real acoustical environment. The room which is used for the recordings was 3.4 x 3.8 x 5.2 m (height x width x depth) and is depicted in Figure 2. Two persons read 4 sentences aloud. Also, far end speech was introduced by playing the French news over a small loudspeaker. The resulting sound was recorded by two microphones which were spaced 58 cm apart. The recordings are 16 bit, 24kHz. The separation filters are of length 512 and are controlled by the CoBliSS algorithm. The far end speech was used as a third input for the BSS. For the sake of computational complexity, only one update per every 2560 samples is done. In this experiment,

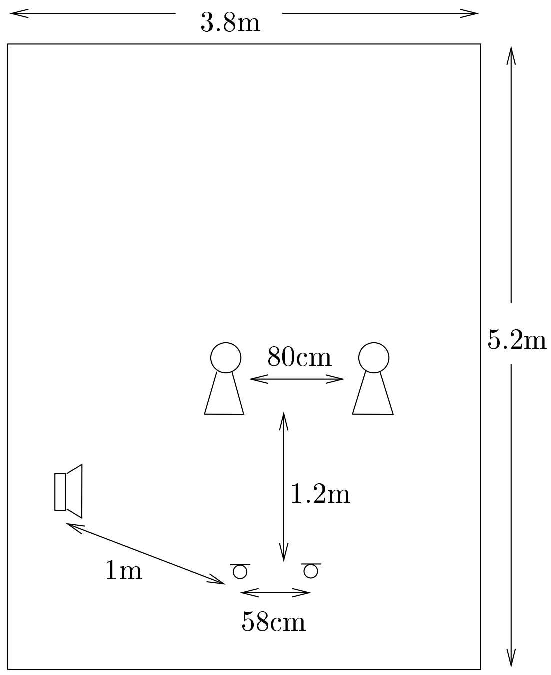

Figure 2: Recording setup

the algorithm converges to a good solution within 0.25 second. An additional advantage of this approach is that is gives good "echo cancellation" and it is not hampered by double talk. Clearly, CoBliSS can be extended using the knowledge that the far end speech readily is a source signal [8, 9]. The experimental results can be played from

`http://www.esp.ele.tue.nl/~daniels/`

Also music signals have been separated successfully using CoBliSS. The algorithm can process more than two microphone signals at a modest increase in computational power. In addition, it facilitates for the integration of acoustical echo cancellers which now can operate in double talk situations without complications [8, 9]. This makes it suitable for applications like teleconferencing, hands free telephony, etc.

10 Conclusions & Future Work

A new blind signal separation algorithm is presented which is based on second order statistics. Experiments show that the algorithm has a good performance when applied to real world audio problems involving speech and music. Future work includes incorporating prior knowledge about the room acoustics and the sources into the algorithm. Also, acoustical echo cancelling will be included explicitly in CoBliSS.

Acknowledgements

The authors would like to thank Jean-François Cardoso, Yves Grenier and Randal Douc for their valuable suggestions.

References

[1] D.C.B. Chan. *Blind Signal Separation*. Cambridge: Thesis University of Cambridge, 1997. Ph. D. Thesis.

[2] R.H. Lambert. *Multichannel Blind Deconvolution: FIR Matrix Algebra and Separation of Multipath Mixtures*. University of Southern California, May 1996. Ph.D. Thesis.

[3] T-W. Lee, A.J. Bell, and R. Lambert. Blind separation of delayed and convolved sources. *Advances in Neural Inf. Proc. Systems 9*, pages 758–764, 1997.

[4] A.J. Bell and T.J. Sejnowski. An information maximisation approach to blind separation and blind deconvolution. *Neural Computation 7*, pages 1129–1159, 1995. MIT Press, Cambridge MA.

[5] D.W.E. Schobben and P.C.W. Sommen. Second order statistics, a sufficient criterion in blind signal separation? In *Proc. Int. Workshop Independent Component Analysis and Blind Signal Separation*, Jan. 1999. Subm. for publ.

[6] P.J. Davis. *Circulant Matrices*. New York (USA): Wiley, 1979.

[7] D.W.E. Schobben, G.P.M. Egelmeers, and P.C.W. Sommen. Efficient realization of the block frequency domain adaptive filter. In *ICASSP 97 : Int. Conf. on Acoustics, Speech and Signal Proc.*, pages 2257–2260, Apr. 1997.

[8] D.W.E. Schobben and P.C.W. Sommen. Transparent communication. In *Proceedings IEEE Benelux Signal Processing Chapter Symposium*, pages 171–174, Mar. 1998.

[9] D.W.E. Schobben and P.C.W. Sommen. A new algorithm for joint blind signal separation and acoustic echo cancelling. In *ICASSP 99 : Int. Conf. on Acoustics, Speech and Signal Proc.*, March 1999. Subm. for publ.