EXTENSION OF SCORE FUNCTION DIFFERENCE FOR FREQUENCY DOMAIN BLIND SOURCE SEPARATION

Jani Even, Hiroshi Saruwatari, and Kiyohiro Shikano

Graduate School of Information Science Nara Institute of Science and Technology
8916-5 Takayama-cho, Ikoma-shi, Nara 630-0101, Japan
email: even@is.naist.jp

ABSTRACT

Recent theoretical developments in the field of blind source separation (BSS) introduced the score function difference (SFD) and showed its close relationship with the differential of the mutual information. These results are of great interest for developing BSS methods since the mutual information can be seen as the canonical cost function for BSS. However these results were only proposed in the real case. In this paper, we extend these results to the complex case and propose a frequency domain approach of BSS based on the complex equivalent of the SFD.

1. INTRODUCTION

The goal of BSS is to recover some signals, called sources, when only some mixtures of these sources are observed. This problem has received much attention for the last decades resulting in several algorithms and applications (see book [1]). Recently, theoretical developments [2, 3, 4, 5] introduced the score function difference (SFD) and showed its importance in BSS. In particular a theorem expresses the differential of the mutual information as a function of the SFD.

However, these results were only derived for real valued BSS preventing their use in fields like acoustic signal processing, where the frequency domain approach of BSS (FD-BSS), which implies complex valued BSS, is prevalent (see review paper [6]).

The main contribution of this paper is the extension of the theorem concerning the differential of the mutual information to the complex case. This leads to the definition of the complex equivalent of the SFD. Then we also present a FD-BSS algorithm based on this new result and illustrate its effectiveness by some numerical simulation using real measurements.

2. PRELIMINARIES

2.1 Blind source separation

Figure 1 represents the simplest BSS model. The multidimensional observed signal $V(t) = [v_1(t), \dots, v_n(t)]^T$ is assumed to be an instantaneous linear mixture of the unknown signal $S(t) = [s_1(t), \dots, s_n(t)]^T$, called source vector. The mixing process is represented by the relation

$$V(t) = AS(t)$$

where A is a $n \times n$ matrix.

The estimate $Y(t)$ of the source vector is obtained by applying a $n \times n$ matrix B to the observed signal (see Fig. 1)
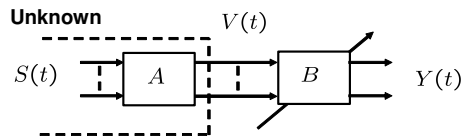
$$Y(t) = BV(t).$$

Figure 1: Blind source separation model.

The determination of the matrix B is based on the following theorem (see [7] and reference in [1]).

Theorem 1.
If the components of $S(t)$ are statistically independent and have non Gaussian distribution, then the components of $Y(t)$ are statistically independent if and only if B is such that

$$\begin{aligned} Y(t) &= BAS(t) \\ &= P\Lambda S(t) \end{aligned}$$

where P is a $n \times n$ permutation matrix and Λ is a diagonal $n \times n$ matrix. $\square$

Namely, assuming that the components of $S(t)$ are statistically independent and have non Gaussian distribution, it is possible to recover them up to scaling and permutation by finding B such that the components of $Y(t)$ are statistically independent. In the following section, we will present the framework for iterative blind separation algorithms based on Theorem 1.

2.2 Iterative algorithms

As a result of Theorem 1, a large class of the BSS methods are using a cost function measuring the statistical dependency among the components of $Y(t)$, say a function $J(Y(t))$ that is minimum when the components of $Y(t)$ are statistical independent. In fact this cost function is a function of the separating matrix B since $Y(t) = BV(t)$. Thus the separating matrix B can be estimated by an iterative algorithm minimizing the cost function.

Let B_k denotes the separation matrix at iteration k, the natural gradient descent update rule of B_k is

$$\begin{aligned} B_{k+1} &= B_k - \mu_k \frac{\partial}{\partial B} J(Y(t))|_k B_k^T B_k \\ &= \left(I - \mu_k \frac{\partial}{\partial B} J(Y(t))|_k B_k^T \right) B_k \end{aligned}$$

where μ_k is a positive adaptation step and $\frac{\partial}{\partial B} J(Y(t))|_k$ is the gradient of the cost function with respect to B at iteration

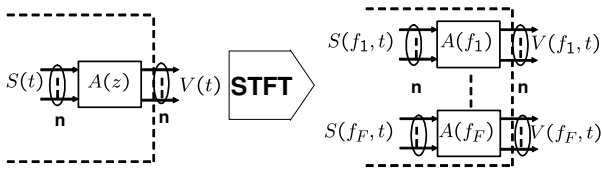Figure 2: Time domain convolutive mixture and equivalent frequency domain instantaneous mixtures.

Figure 3: FD-BSS model at frequency bin f.

k. Consequently, iterative algorithms are defined by the cost function used and the gradient of the cost function has to be estimated at each iteration. Several cost functions were proposed (see reference in [1]), but most of them are approximations of the mutual information of $Y(t)$ or equivalent to it (defined in next section).

2.3 Differential of mutual information: Real case

This section presents a recent result concerning the differential of the mutual information [2, 3, 5]. The motivation is the ability to derive the gradient of the mutual information from its differential in a simple way. Moreover this approach is less sensitive to estimation bias present in most of the methods based on cost functions that approximate the mutual information.

The mutual information of the real random vector $X = [x_1, x_2, \ldots, x_n]^T$ is defined by

$$\mathrm{I}(X) = \int P(X) \log \frac{P(X)}{\prod_{i=1}^{n} P(x_i)} dX.$$

It is a non negative quantity that equals zero if and only if the random variables x_i are statistically independent.

Theorem 2 (Theorem 2 in [3])
The differential of the mutual information with respect to a small variation of $X \rightarrow X + \Delta_X$ is

$$I(X + \Delta_X) - I(X) = \mathscr{E} \left\{ \Delta_X^T \beta(X) \right\} + o\Delta_X \qquad (1)$$

where $\beta(X)$ is the score function difference (SFD). □

The SFD is a $n \times 1$ vector which is the difference of the marginal score function and the joint score function $\beta(X) = \phi(X) - \Phi(X)$. The ith entry of the marginal score function is

$$\{\phi(X)\}_i = -\frac{\partial}{\partial x_i} \log P_{x_i}(x_i)$$

and the ith entry of the joint score function is

$$\{\Phi(X)\}_i = -\frac{\partial}{\partial x_i} \log P_X(X).$$

2.4 Frequency domain blind signal separation

In a reverberant environment, the propagation of n sounds from their locations of emission until a n element microphone array is modeled by a convolutive mixture of size $n \times n$ (left side of Fig. 2). The frequency domain model of the mixture is obtained by applying a short time Fourier transform
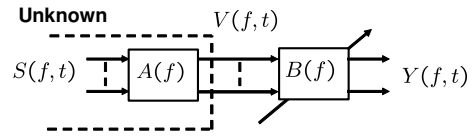
(STFT) using a F points analysis frame to the observed signals. In the remainder of the paper t denotes the frame index. In the frequency domain, the $n \times n$ convolutive mixture is transformed in F instantaneous mixtures of size $n \times n$ (right side of Fig. 2). At the fth frequency bin, the observed signal $V(f,t)$ is

$$V(f,t) = A(f)S(f,t)$$

where the $n \times n$ complex valued matrix $A(f)$ represents the instantaneous mixture and $S(f,t)$ is the emitted signal components at the fth frequency bin, (see Fig. 3).

In FD-BSS, the component of the emitted signals in each frequency bin, $S(f,t)$, are estimated by applying a $n \times n$ complex valued matrix $B(f)$ to the observed signals (see 1)

$$Y(f,t) = B(f)V(f,t).$$

Theorem 1 in Sect.2.1 is also valid for complex mixtures. If the components of $S(f,t)$ are statistically independent, the component of $Y(f,t)$ are statistically independent if and only if $B(f)$ is such that

$$Y(f,t) \quad = \quad P(f)\Lambda(f)S(f,t)$$

where $P(f)$ is a $n \times n$ permutation matrix and $\Lambda(f)$ is a diagonal $n \times n$ complex valued matrix.
Namely, it is possible to recover the components of $S(f,t)$ up to scaling and permutation indeterminacy by finding the matrix $B(f)$ that gives statistically independent estimates.

In FD-BSS, after performing blind separation in each of the frequency bins, it is mandatory to resolve the permutation indeterminacy before transforming the separated signals back to the time domain otherwise the time signals will not be separated. For this reason FD-BSS architectures includes a permutation resolution method that forces all $P(f)$ to be equal. Several algorithms were proposed to estimate $B(f)$ in each bins and resolve the permutation indeterminacy (see review paper for references [6]).

3. MAIN RESULTS

3.1 Mutual information differential: Complex case

In this section, we will extend the result of Theorem 2 to the complex case and define the complex score function difference (CSFD).

A complex random variable z is defined as a random variable $z = x + \mathbf{i} y$ where x and y are real random variables. The probability density function $P_z(z)$ of z is the joint probability density function $P_z(z) = P_{xy}(x,y)$. The mutual information of the complex random vector $Z = [z_1, \ldots, z_n]$ with $z_i = x_i + \mathbf{i} y_i$ is noted

$$\mathrm{I}_c(Z) \quad = \quad \int P_Z(Z) \log \frac{P_Z(Z)}{\prod_{i=1}^{n} P_{z_i}(z_i)} dZ. \qquad (2)$$

Theorem 3

The differential of the complex mutual information relatively to a small variation of $Z \to Z + \Delta_Z$ is

$$I_c(Z + \Delta_Z) - I_c(Z) = \mathscr{E}\left\{Re\left\{(\Delta_Z)^H \beta_c(Z)\right\}\right\} + o\{\Delta_Z\} \quad (3)$$

where $\beta_c(Z)$ is the complex score function difference (CSFD). $\square$

The CSFD is the difference between the complex marginal score function and the complex joint score function $\beta_c(Z) = \phi(Z) - \Phi(Z)$. The complex marginal score function is a $n \times 1$ vector with ith entry

$$\{\phi(Z)\}_i = -\left(\frac{\partial}{\partial x_i}\log P_{z_i}(x_i, y_i) + \mathbf{i}\frac{\partial}{\partial y_i}\log P_{z_i}(x_i, y_i)\right). \quad (4)$$

The complex joint score function's ith entry is

$$\{\Phi(Z)\}_i = -\left(\frac{\partial}{\partial x_i}\log P_Z(Z) + \mathbf{i}\frac{\partial}{\partial y_i}\log P_Z(Z)\right) \quad (5)$$

where $P_Z(Z) = P_{x_1, y_1, \dots, x_n, y_n}(x_1, y_1, \dots, x_n, y_n)$.

Proof of Theorem 3

First we express the mutual information using the joint probability density functions of the real and imaginary parts of the complex random variables. Introducing these joint probability density functions in (2) we get

$$
\begin{aligned}
I_c(Z) &= \int P_{x_1, y_1, \dots, x_n, y_n}(x_1, y_1, \dots, x_n, y_n) \\
&\times \log \frac{P_{x_1, y_1, \dots, x_n, y_n}(x_1, y_1, \dots, x_n, y_n)}{\prod_{i=1}^n P_{x_i, y_i}(x_i, y_i)} dx_1 \dots dy_n.
\end{aligned}
$$

By multiplying both the numerator and the denominator of the logarithm's argument with $\prod_{i=1}^n P_{x_i}(x_i)P_{y_i}(y_i)$ we obtain

$$
\begin{aligned}
I_c(Z) &= \int P_{x_1, y_1, \dots, x_n, y_n}(x_1, y_1, \dots, x_n, y_n) \\
&\times \log \frac{P_{x_1, y_1, \dots, x_n, y_n}(x_1, y_1, \dots, x_n, y_n)}{\prod_{i=1}^n P_{x_i}(x_i)P_{y_i}(y_i)} dx_1 \dots dy_n \\
&- \int P_{x_1, y_1, \dots, x_n, y_n}(x_1, y_1, \dots, x_n, y_n) \\
&\times \log \frac{\prod_{i=1}^n P_{x_i, y_i}(x_i, y_i)}{\prod_{i=1}^n P_{x_i}(x_i)P_{y_i}(y_i)} dx_1 \dots dy_n
\end{aligned}
$$

Expanding the second integral gives

$$I_c(Z) = I(x_1, y_1, \dots, x_n, y_n) - \sum_{i=1}^n I(x_i, y_i). \quad (6)$$

The variation $Z \to Z + \Delta_Z$ can be broken down into

$$
\begin{aligned}
x_i &\to x_i + \Delta_{x_i} \\
y_i &\to y_i + \Delta_{y_i}.
\end{aligned}
$$

Using (6) we rewrite the left hand side of (3)

$$
I_c\left(\begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix} + \begin{bmatrix} \Delta_{z_1} \\ \vdots \\ \Delta_{z_n} \end{bmatrix}\right) - I_c\left(\begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix}\right) =
$$

$$
I\left(\begin{bmatrix} x_1 \\ \vdots \\ x_n \\ y_1 \\ \vdots \\ y_n \end{bmatrix} + \begin{bmatrix} \Delta_{x_1} \\ \vdots \\ \Delta_{x_n} \\ \Delta_{y_1} \\ \vdots \\ \Delta_{y_n} \end{bmatrix}\right) - I\left(\begin{bmatrix} x_1 \\ \vdots \\ x_n \\ y_1 \\ \vdots \\ y_n \end{bmatrix}\right)
$$

$$
- \sum_{i=1}^n \left\{ I\left(\begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} \Delta_{x_i} \\ \Delta_{y_i} \end{bmatrix}\right) - I\left(\begin{bmatrix} x_i \\ y_i \end{bmatrix}\right) \right\}
$$

Applying Theorem 2 to obtain the differential of the mutual information for the real case yields

$$I_c(Z + \Delta_Z) - I_c(Z) =$$

$$
\mathscr{E}\left\{ \begin{bmatrix} \Delta_{x_1} \\ \vdots \\ \Delta_{x_n} \\ \Delta_{y_1} \\ \vdots \\ \Delta_{y_n} \end{bmatrix}^T \beta(x_1, \dots, x_n, y_1, \dots, y_n) \right\} \quad (7)
$$

$$
- \sum_{i=1}^n \left\{ \mathscr{E}\left\{ \begin{bmatrix} \Delta_{x_i} \\ \Delta_{y_i} \end{bmatrix}^T \beta(x_i, y_i) \right\} \right\} + o\{\Delta_Z\}.
$$

Using

$$
\sum_{i=1}^n \left\{ \mathscr{E}\left\{ \begin{bmatrix} \Delta_{x_i} \\ \Delta_{y_i} \end{bmatrix}^T \beta(x_i, y_i) \right\} \right\} =
$$

$$
\mathscr{E}\left\{ \begin{bmatrix} \Delta_{x_1} \\ \Delta_{y_1} \\ \vdots \\ \Delta_{x_n} \\ \Delta_{y_n} \end{bmatrix}^T \begin{bmatrix} \beta(x_1, y_1) \\ \vdots \\ \beta(x_n, y_n) \end{bmatrix} \right\}
$$

and re-ordering the terms in the first part of the right hand side of (7) we obtain

$$I_c(Z + \Delta_Z) - I_c(Z) =$$

$$
\mathscr{E}\left\{ \begin{bmatrix} \Delta_{x_1} \\ \Delta_{y_1} \\ \vdots \\ \Delta_{x_n} \\ \Delta_{y_n} \end{bmatrix}^T \left(\beta(x_1, y_1, \dots, x_n, y_n) - \begin{bmatrix} \beta(x_1, y_1) \\ \vdots \\ \beta(x_n, y_n) \end{bmatrix} \right) \right\}
$$

$$+ o\{\Delta_Z\}. \quad (8)$$

With the definition of the SFD we have

$$\beta(x_1,y_1,\ldots,x_n,y_n) - \begin{bmatrix} \beta(x_1,y_1) \\ \vdots \\ \beta(x_n,y_n) \end{bmatrix} =$$

$$\begin{bmatrix} -\frac{\partial}{\partial x_1}\log P_{x_1,y_1} + \frac{\partial}{\partial x_1}\log P_{x_1,y_1,\ldots,x_n,y_n} \\ -\frac{\partial}{\partial y_1}\log P_{x_1,y_1} + \frac{\partial}{\partial y_1}\log P_{x_1,y_1,\ldots,x_n,y_n} \\ \vdots \\ -\frac{\partial}{\partial x_n}\log P_{x_n,y_n} + \frac{\partial}{\partial x_n}\log P_{x_1,y_1,\ldots,x_n,y_n} \\ -\frac{\partial}{\partial y_n}\log P_{x_n,y_n} + \frac{\partial}{\partial y_n}\log P_{x_1,y_1,\ldots,x_n,y_n} \end{bmatrix} = \begin{bmatrix} g_{x_1} \\ g_{y_1} \\ \vdots \\ g_{x_n} \\ g_{y_n} \end{bmatrix}.$$

Then we rewrite the term inside the expectation of (8) as

$$\begin{bmatrix} \Delta x_1 \\ \vdots \\ \Delta x_n \end{bmatrix}^T \begin{bmatrix} g_{x_1} \\ \vdots \\ g_{x_n} \end{bmatrix} + \begin{bmatrix} \Delta y_1 \\ \vdots \\ \Delta y_n \end{bmatrix}^T \begin{bmatrix} g_{y_1} \\ \vdots \\ g_{y_n} \end{bmatrix}$$

$$= \begin{bmatrix} \Delta x_1 \\ \vdots \\ \Delta x_n \end{bmatrix}^T \begin{bmatrix} g_{x_1} \\ \vdots \\ g_{x_n} \end{bmatrix} - \left(\mathbf{i} \begin{bmatrix} \Delta y_1 \\ \vdots \\ \Delta y_n \end{bmatrix} \right)^T \left(\mathbf{i} \begin{bmatrix} g_{y_1} \\ \vdots \\ g_{y_n} \end{bmatrix} \right)$$

$$= \mathrm{Re}\left\{ \left(\begin{bmatrix} \Delta x_1 \\ \vdots \\ \Delta x_n \end{bmatrix} + \mathbf{i} \begin{bmatrix} \Delta y_1 \\ \vdots \\ \Delta y_n \end{bmatrix} \right)^H \left(\begin{bmatrix} g_{x_1} \\ \vdots \\ g_{x_n} \end{bmatrix} + \mathbf{i} \begin{bmatrix} g_{y_1} \\ \vdots \\ g_{y_n} \end{bmatrix} \right) \right\}.$$

Grouping the $g(x_i)$ and $g(y_j)$ according to (4) and (5) we have

$$\begin{bmatrix} g(x_1) + \mathbf{i}\,g(y_1) \\ \vdots \\ g(x_n) + \mathbf{i}\,g(y_n) \end{bmatrix} = \beta_c(Z).$$

Finally combining the two previous equations and (8) we obtain

$$I_c(Z + \Delta_Z) - I_c(Z) = \mathscr{E}\left\{ \mathrm{Re}\left\{ (\Delta_Z)^H \beta_c(Z) \right\} \right\} + o\{\Delta_Z\}.$$

3.2 Blind signal separation algorithm

In [2, 3, 5], the authors use Theorem 2 to derive a BSS method based on the SFD. It is possible to obtain the equivalent method for complex mixtures using Theorem 3 to compute the gradient of the mutual information with respect to the separating matrix B.

Considering the estimate

$$Y(t) = BX(t)$$

where all terms are complex valued, for a small variation $B \rightarrow B + \Delta_B$, the output variation is $Y \rightarrow Y + \Delta_B X$ (we drop index t in the following). With Theorem 3 we get

$$I_c(Y + \Delta_B X) - I_c(Y) = \mathscr{E}\left\{ \mathrm{Re}\left\{ (\Delta_B X)^H \beta_c(Y) \right\} \right\} + o\{\Delta_B X\}.$$

Using this differential and defining the $(ij)^{th}$ entry of B as

$$b_{ij} = b_{ijR} + \mathbf{i}\,b_{ijI},$$

we obtain the derivative of the mutual information with respect to the real and imaginary parts

$$\frac{\partial I_c(Y)}{\partial b_{ijR}} = \mathscr{E}\left\{ \mathrm{Re}\left\{ x_j^\star \beta_c(Y)^{(i)} \right\} \right\},$$

$$\frac{\partial I_c(Y)}{\partial b_{ijI}} = \mathscr{E}\left\{ \mathrm{Re}\left\{ -\mathbf{i} x_j^\star \beta_c(Y)^{(i)} \right\} \right\}$$

$$= \mathscr{E}\left\{ \mathrm{Im}\left\{ x_j^\star \beta_c(Y)^{(i)} \right\} \right\}$$

where $\star$ denotes the complex conjugate operator. Consequently the complex derivative is

$$\frac{\partial I_c(Y)}{\partial b_{ijR}} + \mathbf{i}\frac{\partial I_c(Y)}{\partial b_{ijI}} = \mathscr{E}\left\{ x_j^\star \beta_c(Y)^{(i)} \right\}.$$

Rewriting the previous result in matrix form we have

$$\frac{\partial I_c(Y)}{\partial B} = \mathscr{E}\left\{ \beta_c(Y) X^H \right\}.$$

Using this expression, we can define an iterative blind separation algorithm based on the natural gradient descent as in Sect.2.2. Let B_k denotes the separation matrix at iteration k and define $Y_k(t) = B_k X(t)$ then the update rule is

$$B_{k+1} = \left(I - \mu_k\,\mathscr{E}\left\{ \beta_c(Y_k) X^H \right\} B_k^H \right) B_k$$
$$= \left(I - \mu_k\,\mathscr{E}\left\{ \beta_c(Y_k) Y_k^H \right\} \right) B_k \quad (9)$$

where μ_k is a positive adaptation step. This is the complex version of the algorithm presented for the real case in [3, 5] (Note that the proposed complex gradient $\mathscr{E}\left\{ \beta_c(Y_k) X^H \right\}$ coincides with the expression in the real case $\mathscr{E}\left\{ \beta(Y_k) X^T \right\}$).

3.3 Estimation of complex score function difference

The proposed algorithm requires the estimation of the CSFD. Since the frequency domain signal are obtained by using a short time Fourier transform, we can assume that all the signals are circular, i.e. the joint density of their modulus and phase is separable [8].

For $n = 2$, if $y_1(f,t)$ and $y_2(f,t)$ are circular and independent then (the frame index t and frequency index f are dropped)

$$\beta_c(y_1, y_2) = \begin{bmatrix} \frac{\partial}{\partial|y_1|}\log P(|y_2|/|y_1|)\frac{y_1}{|y_1|} \\ \frac{\partial}{\partial|y_2|}\log P(|y_1|/|y_2|)\frac{y_2}{|y_2|} \end{bmatrix} \quad (10)$$

$$= \beta(|y_1|, |y_2|) \circ \begin{bmatrix} \frac{y_1}{|y_1|} \\ \frac{y_2}{|y_2|} \end{bmatrix}$$

where $\circ$ denotes the element wise product.

In practice, to estimate the CSFD, we obtain the conditional distributions from a kernel base estimate of the joint distribution $P(|y_1|, |y_2|)$. Then we use finite difference to obtain the derivatives of the conditional probabilities that appear in the log derivate in (10). This is a crude estimate of the CSFD as the SFD estimate used in [3].

4. SIMULATIONS

To illustrate the effectiveness of the proposed method, we compare its performance with a BSS method based on adaptive score function for a speech/speech separation task. A
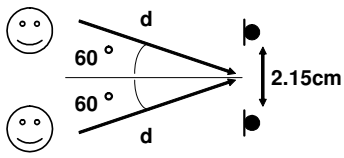
Figure 4: Experimental setting.

Figure 5: Comparison of noise reduction rate (proposed SFD, reference Score) for different distances.

two microphone array was used to record the impulse responses from different locations in a room with a 200 ms reverberation time. These impulse responses were used to generate mixtures of speech signals. We considered three mixtures of two speakers. For all mixtures the speakers are at 60 degrees on the right and the left of the microphone array but the distances d are set to 1, 1.5 and 2 meters (see Fig. 4). For each position five different sets of signals are used. Each set is composed of Japanese sentences read by one female and one male speaker (from the JNAS database [9]). The average length is 10s. The sampling frequency is 16 kHz and the short time Fourier transform is performed with 512 points hanning windows having 50% overlap.

The reference BSS method is a complex version of IN-FOMAX [10] with nonlinearities that are adaptively estimated from the data and converges to the marginal score functions of the source $\phi(S)$ [11]. The estimation of the marginal score function also assume circularity [8]. With the same notation as in (9) the update rule is

$$B_{k+1} = \left(I - \mu_k \left(I - \mathscr{E}\left\{\phi(Y_k)Y_k^H\right\}\right)\right)B_k.$$

For both approaches the permutation indeterminacy is resolved with the method based on direction of arrival proposed in [12].

The quality of the separation is measured in term of noise reduction rate (NRR in dB): NRR 1 = SNR of speech 1 after processing (in the estimate) - SNR of speech 1 before processing (in the observation). A positive NRR means that the estimate quality is improved.

In all the experiments, the proposed method (SFD) achieved an average NRR comparable (slightly better) to the reference method (Score) (see Fig. 5, the plotted NRR is the average of NRR 1 and NRR 2 for all set of signals).

However, in the current form, the proposed method has a higher computation cost because of the joint density estimate used to estimate the SFD (mean computation time is 12 time longer). The proposed algorithm is a simple illustration of the complex valued SFD as the algorithm in [3] is for the real case. To obtain an efficient FDBSS algorithm based on CSFD, the techniques used in [13, 5] should be extended to the complex case.

5. CONCLUSIONS

In this paper, we extended a theorem giving the differential of the mutual information to the complex case by defining the CSFD. We also presented a simple FDBSS algorithm based on CSFD. This paves the way to the use of SFD based BSS method in the frequency domain. This is of particular interest for acoustical signal processing applications where the frequency domain approach of BSS is widespread.
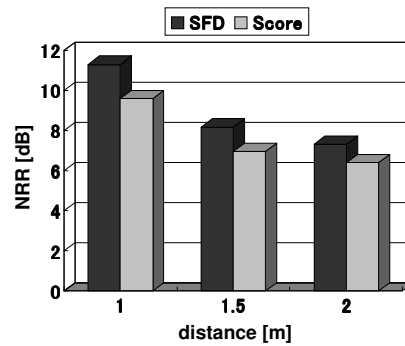
REFERENCES

[1] A. Hyvarinen et al., *Independent Component Analysis*. John Wiley & Sons, 2001.

[2] D.-T. Pham, "Mutual information approach to blind separation of stationary sources," *IEEE Trans. on Information Theory*, vol. 7, no. 48, pp. 1935–1946, 2002.

[3] M. Babaie-Zadeh et al., "Differential of the mutual information," *IEEE Signal Processing Letters*, vol. 11, no. 1, pp. 48–51, 2004.

[4] B. Bahmani, M. Babaie-Zadeh, and C. Jutten, "A new method for estimating score function difference (sfd) and its application to blind source separation," *in proc. of EUSIPCO 2005*, 2005.

[5] S. Samadi et al., "Quasi-optimal easi algorithm based on the score function difference (sfd)," *Neurocomputing (Elsevier)*, vol. 69, pp. 1415–1424, 2006.

[6] M. Pedersen et al., "A survey of convolutive blind source separation methods," *Springer Handbook on Speech Communication*, 2007.

[7] P. Comon, "Independent component analysis, a new concept ?" *Signal Processing*, vol. 36, pp. 287–314, 1994.

[8] H. Sawada et al., "Polar coordinate based nonlinear function for frequency-domain blind source separation," *IEICE Trans. Fundamentals*, vol. E86-A, no. 3, pp. 590–596, 2003.

[9] K. Ito et al., "Jnas: Japanese speech corpus for large vocabulary continuous speech recognition research," *The Journal of Acoustical Society of Japan*, vol. 20, pp. 196–206, 1999.

[10] A. J. Bell and T. J. Sejnowski, "An information maximimization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

[11] N. Vlassis and Y. Motomura, "Efficient source adaptivity in independent component analysis." *IEEE Trans. Neural Networks*, vol. 12, no. 3, pp. 559–566, 2001.

[12] H. Sawada et al., "A robust and precise method for solving the permutation problem of frequency-domain blind source separation," *IEEE Trans. Speech and Audio Proc.*, vol. 12, pp. 530–538, 2004.

[13] D.-T. Pham, "Fast algorithm for estimating mutual information, entropies and score functions," *Proceeding of ICA 2003 Conference Nara, Japan*, pp. 17–22, 2003.