A METHOD OF REAL-WORLD BLIND SEPARATION IMPLEMENTED IN FREQUENCY DOMAIN

M. Kawamoto[1,2], *A. K. Barros*[2], *K. Matsuoka*[3], *and N. Ohnishi*[1,4]

1. Department of Electronic and Control Systems Engineering, Shimane University,
1060 Nishikawatsu, Matsue, Shimane 690-8504, Japan
kawa@ecs.shimane-u.ac.jp
2. Bio-mimetic Control Research Center, RIKEN,
3. Department of Control Engineering, Kyushu Institute of Technology (Japan),
4. Graduate School of Engineering, Nagoya University (Japan)

ABSTRACT

In this paper, we propose a method of real-world blind separation for non-stationary signals (e.g., speech signal, music). The proposed method is implemented in the time-frequency domain and achieves real-world blind separation by minimizing a cost function with the algorithm derived using the natural gradient under a given condition. We carried out computer simulations to demonstrate the validity of our proposed method. Moreover, we apply our method to an experiment which extracts two source signals from their mixtures observed in a normal room with computer and air conditioning noises. The results show that our method can achieve real-world blind separation.

1. INTRODUCTION

This paper deals with real-world blind separation which extracts original signals from their mixtures observed by the sensors in a real world. To solve the real-world blind separation, many methods have been proposed until now [e.g., [1]-[8]]. The methods have been implemented in time domain or frequency domain. In the time domain, the following equation has been used as a transfer function between the sources and sensors.

$$\boldsymbol{x}(t) = \overline{\boldsymbol{A}}(z)\boldsymbol{s}(t), \tag{1}$$

where $\boldsymbol{x}(t) = [x_1(t), ..., x_N(t)]^T$ and $\boldsymbol{s}(t) = [s_1(t), \ldots, s_N(t)]^T$. $\overline{\boldsymbol{A}}(z)$ is a matrix which has elements $\overline{a}_{ij}(z)$ $(i, j = 1, ..., N)$:

$$\overline{a}_{ij}(z) = \sum_{k=-\infty}^{\infty} a_{ij}(k) z^{-k} \quad (i, j = 1, ..., N), \tag{2}$$

where z^{-k} is a delay operator, i.e., $s_i(t)z^{-k} = s_i(t-k)$. Using a method which can separate $\boldsymbol{s}(t)$ from $\boldsymbol{x}(t)$, real-world blind separation has been solved [e.g., [2]].

On the other hand, in the frequency domain, discrete Fourier transform is applied to $\boldsymbol{x}(t)$:

$$\boldsymbol{x}(\omega, t) = \sum_{t'=t}^{t+M} \boldsymbol{x}(t') exp(-j\omega t'), \tag{3}$$

where $\omega = 2\pi k/M$, $k = 0, 1, ..., M-1$, and M denotes the window size of the discrete Fourier transform. Using eqn (3), eqn (1) can be written as:

$$\boldsymbol{x}(\omega, t) = \boldsymbol{A}(\omega)\boldsymbol{s}(\omega, t), \tag{4}$$

where $\boldsymbol{A}(\omega)$ can be approximated by applying the discrete Fourier transform to $\overline{\boldsymbol{A}}(z)$, if the window size M is sufficiently long compared to the impulse response of $\overline{\boldsymbol{A}}(z)$, and $\boldsymbol{s}(\omega, t)$ is the discrete Fourier transform of $\boldsymbol{s}(t)$. The separation is implemented at each frequency. Therefore, the treatment of the separation in the frequency domain is easier than that in the time domain. The method of separating $\boldsymbol{s}(\omega, t)$ from $\boldsymbol{x}(\omega, t)$ has been applied to real-world blind separation [e.g., [4]].

In this paper, we deal with real-world blind separation implemented in the frequency (time-frequency) domain. Several methods have been proposed [e.g., [3][4][5]]. An attractive feature of our proposed method, which differs from the previously reported ones, is that non-minimum phase systems can be treated and the signals just before entering the sensors, that is, for example, $\overline{a}_{ii}(z)s_i(t)$ $(i = 1, ..., N)$, can be acquired. The second feature of our method is that it offers good advantage in some applications, for example, speech recognition.

To determine $\boldsymbol{x}(\omega, t)$, we use the following equation:

$$\boldsymbol{x}(\omega, t_w) = \sum_{t'=t_w}^{t_w+M} \boldsymbol{x}(t') exp(-j\omega t') h(t' - t_w), \tag{5}$$

where $h(t)$ is the Hamming window, and $t_w = 0$, ΔT, $2\Delta T$,... denotes the window position, and ΔT is the shifting interval of the moving window. This transformation is called the windowed-Fourier transform. Then, eqn (4) can be written as

$$\boldsymbol{x}(\omega, t_w) = \boldsymbol{A}(\omega)\boldsymbol{s}(\omega, t_w). \quad (6)$$

The problem is to separate sources $s_i(\omega, t_w)$ from their mixtures $\boldsymbol{x}(\omega, t_w)$ at each frequency. Our method overcomes this by modifying the parameters of a network such that a cost function takes the minimum (zero) at any time. The validity of the proposed method was confirmed by computer simulations and experiments, the results of which show that real-world blind separation can be achieved.

2. SEPARATION PROCESS

We assume that source signals are mutually statistically independent and non-stationary signals (e.g., speech signals, music). As for $\overline{\boldsymbol{A}}(z)$ in eqn (1), we assume that $\overline{\boldsymbol{A}}(z)$ does not have poles and zeros on the unit circle $|z| = 1$ and does not depend on the time index t.

2.1. Signal Separation Network

The problem is to extract $\boldsymbol{s}(\omega, t_w)$ from $\boldsymbol{x}(\omega, t_w)$. To this end, we use a network given by the following equation (see Fig. 1).

$$\boldsymbol{y}(\omega, t_w) = \boldsymbol{B}(\omega)^{-1}\boldsymbol{x}(\omega, t_w), \quad (7)$$

where $\boldsymbol{B}(\omega) = [b_{ij}(\omega)]$ is obtained by applying the windowed-Fourier transform to the following equation:

$$\overline{\boldsymbol{B}}(z) = \sum_{k=0}^{M} \boldsymbol{B}(k) z^{-k}. \quad (8)$$

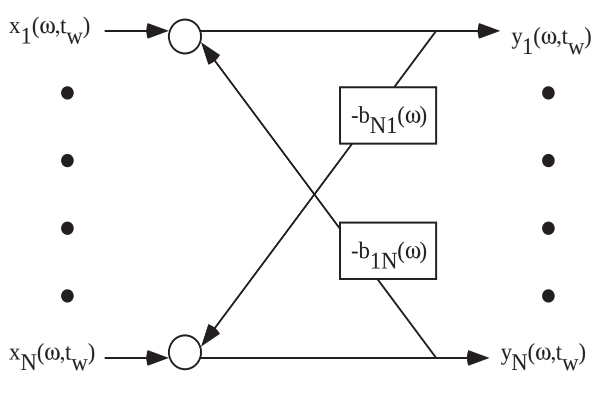

Figure 1: Signal Separation Network.

Note that the diagonal elements $b_{ii}(\omega)$ $(i = 1, ..., N)$ of matrix $\boldsymbol{B}(\omega)$ are fixed to 1. $\overline{\boldsymbol{B}}(z)$ does not depend on the time index t.

Substituting eqn (6) into eqn (7), we have

$$\boldsymbol{y}(\omega, t_w) = \boldsymbol{B}(\omega)^{-1}\boldsymbol{A}(\omega)\boldsymbol{s}(\omega, t_w) = \boldsymbol{C}(\omega)\boldsymbol{s}(\omega, t_w), \quad (9)$$

where $\boldsymbol{C}(\omega) \equiv \boldsymbol{B}(\omega)^{-1}\boldsymbol{A}(\omega)$. For the separated signals, in the blind separation problem, there is an ambiguity of permutation and amplitude. Therefore, if we can find $\boldsymbol{B}_0(\omega)$ such that $\boldsymbol{C}(\omega)$ is equal to $\boldsymbol{D}(\omega)\boldsymbol{P}$, the separation can be achieved. $\boldsymbol{P}$ is a permutation matrix which represents the ambiguity of permutation. $\boldsymbol{D}(\omega)$ is a diagonal matrix which represents the ambiguity of amplitude. In our method, however, since the diagonal elements $b_{ii}(\omega)$ $(i = 1, ..., N)$ of matrix $\boldsymbol{B}(\omega)$ are fixed to 1, we can determine $\boldsymbol{D}(\omega)$, that is,

$$\boldsymbol{D}(\omega) = diag[\boldsymbol{A}(\omega)\boldsymbol{P}^T], \quad (10)$$

where $diag[\boldsymbol{X}]$ denotes a diagonal matrix which has the diagonal elements of matrix $\boldsymbol{X}$. Therefore, the desired solution $(\boldsymbol{B}_0(\omega))$ is $\boldsymbol{A}(\omega)\boldsymbol{P}^T(diag[\boldsymbol{A}(\omega)\boldsymbol{P}^T])^{-1}$. Then, the separated signal $\boldsymbol{y}(\omega, t_w)$ becomes

$$\boldsymbol{y}(\omega, t_w) = diag[\boldsymbol{A}(\omega)\boldsymbol{P}^T]\boldsymbol{P}\boldsymbol{s}(\omega, t_w). \quad (11)$$

Here, let us consider the separated signals. The signals just before entering the microphones can be defined as

$$\overline{a}_{1p_1}(z)s_{p_1}(t), \overline{a}_{2p_2}(z)s_{p_2}(t), ..., \overline{a}_{Np_N}(z)s_{p_N}(t), \quad (12)$$

where $\{p_1, ..., p_N\}$ is an arbitrary permutation of $\{1, ..., N\}$. When the signals of eqn (11) are applied to the inverse Fourier transform, the signals are as follows.

$$\boldsymbol{y}(t) = diag[\overline{\boldsymbol{A}}(z)\boldsymbol{P}^T]\boldsymbol{P}\boldsymbol{s}(t). \quad (13)$$

Therefore, one can see that if the matrix $\boldsymbol{B}_0(\omega)$ can be determined, the signals of eqn (12) can be acquired.

Our aim now is to develop a method which can acquire the matrix $\boldsymbol{B}_0(\omega)$.

2.2. Separation Method

In order to find the matrix $\boldsymbol{B}_0(\omega)$, we use the following function:

$$Q(t_w, \boldsymbol{B}(\omega)) = \frac{1}{2}\Big\{(\sum_{i=1}^{N} logE[y_i(\omega, t_w)y_i(\omega, t_w)^*]) - logdetE[\boldsymbol{y}(\omega, t_w)\boldsymbol{y}(\omega, t_w)^H]\Big\}, \quad (14)$$

where $y_i(\omega, t_w)^*$ is the conjugate value of $y_i(\omega, t_w)$, and the superscript H denotes the Hermitian operator, and det$\boldsymbol{X}$ denotes the determinant of the matrix $\boldsymbol{X}$. The same function as that in eqn (14) has been used by the authors in the case of the time domain [2]. We applied the function used in the time domain to the frequency domain. Wu, et al. [4] have used the same cost function as in eqn (14). They used the summation of the cost function with respect to time t_w. However, we advocate that real-world blind separation can be achieved by using only the function in eqn (14).

We showed that real-world blind separation in the time domain could be achieved by minimizing the same function as in eqn (14) [2]. We apply the same method for the frequency domain. Namely, $\boldsymbol{B}_0(\omega)$ is found by minimizing the function $Q(t_w, \boldsymbol{B}(\omega))$. With respect to the minimization method, we use the natural gradient under the constraint where the diagonal elements $b_{ii}(\omega)$ of $\boldsymbol{B}(\omega)$ are not modified, that is, $b_{ii}(\omega)$ $(i = 1, ..., N)$ are fixed to 1. The following equation is the minimization algorithm of $\boldsymbol{B}(\omega)$ derived from the natural gradient under that constraint.

$$\begin{aligned}\Delta \boldsymbol{B}(\omega) &= \alpha\Big[\frac{\partial Q(t_w, \boldsymbol{V}(\omega))}{\partial \boldsymbol{V}(\omega)}\boldsymbol{V}(\omega)^H\boldsymbol{B}(\omega) \\ &- \Big\{diag\Big[\frac{\partial Q(t_w, \boldsymbol{V}(\omega))}{\partial \boldsymbol{V}(\omega)}\boldsymbol{V}(\omega)^H\boldsymbol{B}(\omega)\Big]\Big\} \\ &\times (diag[\boldsymbol{B}(\omega)^H\boldsymbol{B}(\omega)])^{-1}\boldsymbol{B}(\omega)^H\boldsymbol{B}(\omega)\Big] \quad (15)\end{aligned}$$

where α is a small positive constant, and $\boldsymbol{V}(\omega)$ denotes $\boldsymbol{B}(\omega)^{-1}$. Eqn (15) is used to modify only the non-diagonal elements of $\boldsymbol{B}(\omega)$. In order to derive eqn (15), we used the following relation.

$$\frac{\partial Q(t_w, \boldsymbol{B}(\omega))}{\partial \boldsymbol{B}(\omega)} = -\frac{\partial Q(t_w, \boldsymbol{V}(\omega))}{\partial \boldsymbol{V}(\omega)}\boldsymbol{V}(\omega)^H\boldsymbol{V}(\omega)^H \quad (16)$$

We can prove that the relation in eqn (16) is satisfied.

Calculating the right-hand side of eqn (15), we obtain

$$\begin{aligned}\Delta \boldsymbol{B}(\omega) &= \alpha\Big[\{(diag[E[\boldsymbol{y}(\omega, t_w)\boldsymbol{y}(\omega, t_w)^H]])^{-1} \\ &\times E[\boldsymbol{y}(\omega, t_w)\boldsymbol{y}(\omega, t_w)^H] - \boldsymbol{I}\}\boldsymbol{B}(\omega) \\ &- diag[\{(diag[E[\boldsymbol{y}(\omega, t_w)\boldsymbol{y}(\omega, t_w)^H]])^{-1} \\ &\times E[\boldsymbol{y}(\omega, t_w)\boldsymbol{y}(\omega, t_w)^H] - \boldsymbol{I}\}\boldsymbol{B}(\omega)] \\ &\times (diag[\boldsymbol{B}(\omega)^H\boldsymbol{B}(\omega)])^{-1}\boldsymbol{B}(\omega)^H\boldsymbol{B}(\omega)\Big]. \quad (17)\end{aligned}$$

Eqn (17) is the algorithm proposed by us to update $\boldsymbol{B}(\omega)$ $(\omega = 2\pi k/M,\ k = 0, 1, ...M-1)$.

3. EXPERIMENTAL RESULTS

We carried out some computer simulations and experiments to confirm the validity of our method. This section describes one simulation result and two experimental results.

3.1. Example 1 (Computer simulation)

In this case, the source signals were three; two male voices, and music (the sound of a drum). The channel matrix $\overline{\boldsymbol{A}}(z)$ was given as

$$\overline{\boldsymbol{A}}(z) = \begin{bmatrix} z^{-1} + 0.4z^{-2} & 0.4 + 0.4z^{-1} & 0.25z^{-2} \\ 0.4 + 0.3z^{-1} & 1 + 0.4z^{-1} + 0.2z^{-2} & 0.4 + 0.4z^{-1} \\ 0.25z^{-3} & 0.5z^{-2} & z^{-1} + 0.5z^{-2} \end{bmatrix}$$

The poles of $\overline{\boldsymbol{A}}(z)^{-1}$ are 5.19, -0.67, $-0.32 - 0.52I$, $-0.32 + 0.52I$, $0.055 - 0.24I$, $0.055 + 0.24I$. Therefore, $\overline{\boldsymbol{A}}(z)^{-1}$ has one pole outside the unit circle $|z| = 1$, from which we can deduce that $\overline{\boldsymbol{A}}(z)^{-1}$ is a non-minimum phase system. The value for parameter M in eqn (8) was set at 31. The parameter for the learning algorithm was chosen as $\alpha = 0.001$ (see eqn (17)). The initial values of $b_{ij}(\omega)$ $(i, j = 1, 2, 3; i \neq j; \omega = 2\pi k/M, k = 0, 1, ..., M-1)$ were set to 0. The estimation of $E[\boldsymbol{y}\ (\omega, t_w)\boldsymbol{y}(\omega, t_w)^H]$ was replaced by its instantaneous value, that is, $\boldsymbol{y}(\omega, t_w)\boldsymbol{y}(\omega, t_w)^H$. To estimate $diag[E[\boldsymbol{y}(\omega, t_w)\boldsymbol{y}(\omega, t_w)^H]]$, we use the following moving average:

$$\phi_i(\omega, t) = \beta\phi_i(\omega, t-1) + (1-\beta)y_i(\omega, t)y_i(\omega, t)^* \quad (0 < \beta < 1). \quad (18)$$

The initial values of $\phi_i(\omega, t)$ $(i = 1, 2, 3; \omega = 2\pi k/M, k = 0, 1, ..., M-1)$ and β were set to 1 and 0.9, respectively.

The separation is implemented by using the following process.

- Calculation of $\boldsymbol{x}(t)$ using eqn (1)
- Calculation of $\boldsymbol{x}(\omega, t_w)$ using eqn (5)
- Calculation of $\boldsymbol{y}(\omega, t_w)$ using eqn (7)
- Modification of $\boldsymbol{B}(\omega)$ using eqn (17) and (18)
- Repeat first four steps until learning convergence.

To obtain the separated signals, we must calculate $\overline{\boldsymbol{B}}(z)^{-1}$. We obtain $\overline{\boldsymbol{B}}(z)^{-1}$ by calculating the inverse Fourier transform of $\boldsymbol{B}(\omega)^{-1}$ $(\omega = 2\pi k/M, k = 0, 1, ..., M-1)$. Since $\overline{\boldsymbol{A}}(z)^{-1}$ is a non-minimum phase system, we need to rotate the leading weights of the filters of $\overline{\boldsymbol{B}}(z)^{-1}$ to the middle of the filters. We implemented this calculation using MATLAB (command name, Shift()). Figure 2 shows the elements $\overline{c}_{ij}(z)$

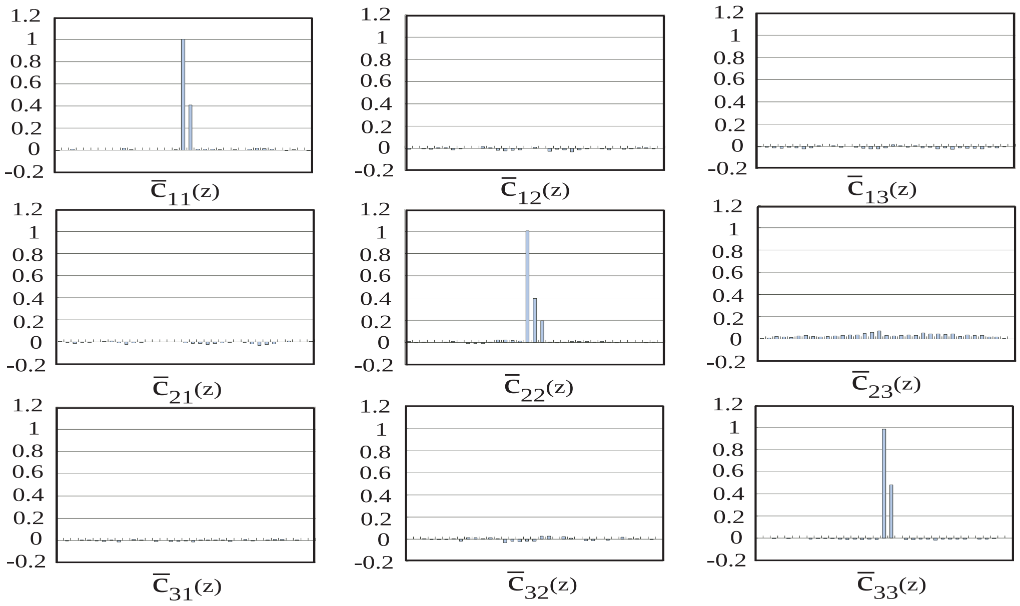

Figure 2: The elements $\overline{c}_{ij}(z)$ of the matrix $\overline{C}(z)$

$(i,j = 1,2,3)$ of the matrix $\overline{\boldsymbol{C}}(z) = \overline{\boldsymbol{B}}_0(z)^{-1}\ \overline{\boldsymbol{A}}(z)$, where $\overline{\boldsymbol{B}}_0(z)^{-1}$ is the inverse Fourier transform of $\boldsymbol{B}_0(\omega)^{-1}$ $(\omega = 2\pi k/M, k = 0,1,...,M-1)$. The matrix $\overline{\boldsymbol{C}}(z)$ is a transfer function between the sources and output signals. The elements $\overline{c}_{ij}(z)$ are denoted as

$$\overline{c}_{ij}(z) = \sum_{k=0}^{34} c_{ij}(k) z^{-k} \qquad (i,j = 1,2,3).$$

In Fig. 2, the horizontal axis is the number k $(k = 0,...,34)$. It can be seen that the non-diagonal elements are nearly equal to zero and the diagonal ones, $\overline{c}_{11}(z)$, $\overline{c}_{22}(z)$, and $\overline{c}_{33}(z)$ are equal to $\overline{a}_{11}(z)$, $\overline{a}_{22}(z)$, and $\overline{a}_{33}(z)$, respectively. From this result, one can see that our method can be used successfully to achieve the separation and to acquire the signals just before entering the microphones.

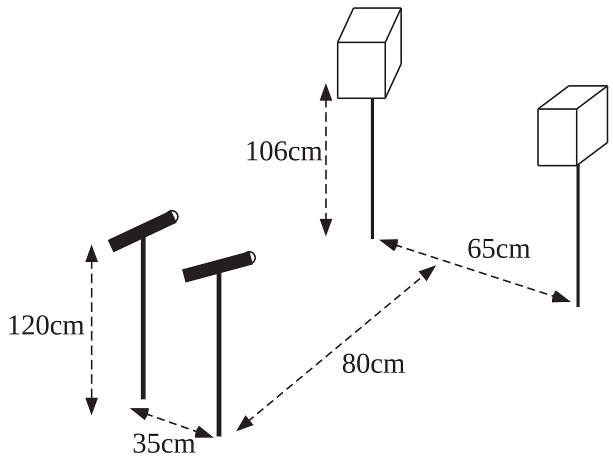

Figure 3: Configuration of two speakers and two microphones

3.2. Example 2 (Experimental result I)

In example 2, 3, we deal with an experiment which extracts two source signals from their mixtures observed in a normal room with air conditioning and computer noises. The source signals $s_1(t)$ and $s_2(t)$ were male voices. And they were input at the same time to two speaker devices. The observed signals $x_1(t)$ and $x_2(t)$ were detected by two directional microphones (Sony ECM-670). Two speakers and two microphones were placed like Figure 3. Parameter M of eqn (8) was set to 511. The parameters of the learning algorithm were chosen as $\alpha = 0.01$ (see eqn (17)) and $\beta = 0.9$ (see eqn (18)). The initial values of $b_{ij}(\omega)$ ($\omega = 0, \frac{1}{M}2\pi,..., \frac{M-1}{M}2\pi$; $i,j = 1,2; i \neq j$) and $\phi(\omega,k)$ were set to 0 and 1, respectively.

Fig. 4 shows the plots of $s_i(t)$, $x_i(t)$, and $y_i(t)$ $(i = 1,2)$. It can be seen that the output signals $y_1(t)$ and $y_2(t)$ are close to the original speech signals $s_1(t)$ and $s_2(t)$, respectively. Therefore, one can see that our method could separate the source signals from their mixtures observed in a normal room.

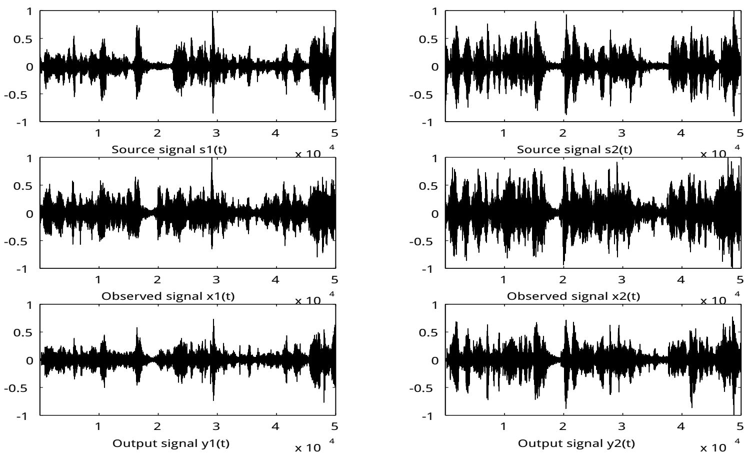

Figure 4: The plots of $s_i(t)$, $x_i(t)$, $y_i(t)$ $(i = 1, 2)$

3.3. Example 3 (Experimental result II)

In this example, source signals $s_1(t)$ and $s_2(t)$ were music and a male voice, respectively. The configuration of two speakers and two microphones is the same as the case of example 2. We used the same parameters (M, α, β) and the same initial values of $b_{12}(\omega)$, $b_{21}(\omega)$, and $\phi(\omega, k)$ as in example 2.

Fig. 5 shows the plots of $s_i(t)$, $x_i(t)$, and $y_i(t)$ $(i = 1, 2)$. It can be seen that the output signals $y_1(t)$ and $y_2(t)$ are close to the original signals $s_1(t)$ and $s_2(t)$, respectively.

4. CONCLUSIONS

We have presented a new method of real-world blind separation which has been derived from the natural gradient method under a constraint. An attractive feature of our proposed method is that non-minimum phase systems can be treated and the signals just before entering the sensors can be acquired.

We have shown the result of a computer simulation. In the example, we have dealt with the problem which separates three source signals from three observed signals, and we have demonstrated that our method makes it possible to acquire the signals just before entering the microphones. Moreover, we have conducted some experiments on separating two source signals from their mixtures observed in a normal room, and they have shown that the proposed method can be used successfully to achieve real-world blind separation.

When real-world blind separation is implemented in the time-frequency domain, one must solve the problem of ambiguity of permutation $\boldsymbol{P}$. We have not obtained the solution to this problem. In our computer simulation, however, our algorithm was not influenced by the ambiguity of permutation $\boldsymbol{P}$. In a future work, we will investigate the problem of ambiguity of permutation $\boldsymbol{P}$.

Acknowledgments

This work was supported by a Grant-in Aid for the Scientific Research by the Ministry of Education, Science and Culture of Japan, No. 11750406.

5. REFERENCES

[1] A. Cichoki, S. Amari, and J. Cao, "Blind separation of delayed and convolutive signals with self-adaptive learning rate," *Proc. Of 1996 International Symposium on Nonlinear Theory and its Applications*, Kouchi, pp. 229-232, 1996.

[2] M. Kawamoto, A. K. Barros, A. Mansour, K. Matsuoka, and N. Ohnishi, "Real world blind separation of convolved non-stationary signals," *Proc. Of First International Workshop on Independent Component Analysis and Signal Separation*, France, pp. 347-352, January, 1999.

[3] S. Ikeda and N. Murata, "A method of ICA in time-frequency domain," *Proc. Of First International Workshop on Independent Component Analysis and Signal Separation*, France, pp. 365-370, January, 1999.

[4] H. C. Wu and J. C. Principe, "Simultaneous diagonalization in the frequency domain (SDIF) for source separation," *Proc. Of First International Workshop on Independent Component Analysis*

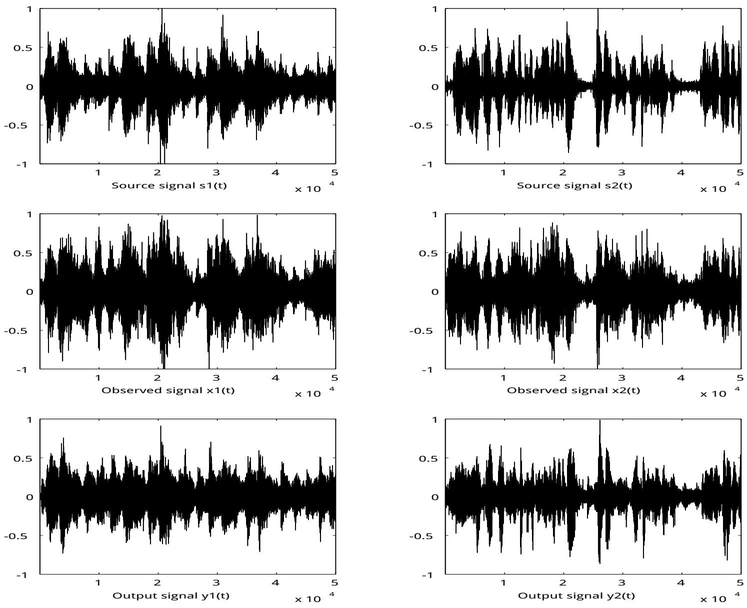

Figure 5: The plots of $s_i(t)$, $x_i(t)$, $y_i(t)$ $(i = 1, 2)$

and Signal Separation, France, pp. 245-250, January 1999.

[5] C. Mejuto and J. C. Principe, "A second-order method for blind source separation of convolutive mixtures," *Proc. Of First International Workshop on Independent Component Analysis and Signal Separation*, France, pp. 395-400, January 1999.

[6] J. Xavier, V. Barroso, and J.M.F. Moura, "Closed Form Blind Identification of MIMO Channels," *Proc. ICASSP98*, Vol. 6, pp. 3165-3168, Seattle WA, 1998.

[7] K.L. Diamantaras and A. Petropulu, "Blind Two-Input-Two-Output FIR Channel Estimation and Source Separation," *Proc. IEEE Information Theory Workshop on Detection, Estimation, Classification and Imaging (DECI-99)*, Santa Fe, NM (USA), February 24-26, 1999.

[8] L. Parra, C. Spense, and B. de Vries, "Convolutive blind source separation based on multiple decorrelation," *Proc. 8th IEEE Workshop Neural Networks for Signal Processing*, Cambridge, UK, Sept. 1998.