IFAC Workshop on Adaptation and Learning in Control and
Signal Processing, and IFAC Workshop on Periodic Control
Systems, Yokohama, Japan, August 30 – September 1, 2004

EFFECTS OF NUMBER OF SENSORS IN A BSS CASCADING GROUP SEPARATION AND LINEARIZATION APPLIED TO NONLINEAR MIXTURE

Kenji Nakayama * **Akihiro Hirano** *
Takayuki Nishiwaki *

* *Dept. of Information and Systems Eng., Faculty of Eng., Kanazawa Univ.*
2-40-20, Kodatsuno, Kanazawa, 920-8667, JAPAN
e-mail:nakayama@t.kanazawa-u.ac.jp

Abstract: A blind source separation (BSS), cascading a separation block and a linearization block has been proposed for low-order nonlinear mixtures. In the separation block, the signal sources are separated into each group, including its high-order components. The high-order components are further suppressed through the linearization block. The number of the sensors is increased from that of the signal sources in order to cancel the noise components. In this paper, separation performance for the reduced number of the sensors is analyzed. Effects of the ratio of nonlinearity in the observed signals is also analyzed. Through theoretical analysis and computer simulations, it is cleared that the theoretical number of the sensors is the best, however, the reduced numbers can provide good separation performance when the ratio of the nonlinearity is small.

Keywords: Blind source separation, Nonlinear mixture, Linearization

1. INTRODUCTION

In practical applications of a blind signal source separation (BSS), processes of generating, mixing and sensing signals include nonlinearity, caused by loud speakers, microphones, amplifiers and so on. Statistical independency is not enough to separate the signal sources, some additional prior knowledge are required. Furthermore, since a unique solution is not guaranteed, some regularization techniques are required [6]. For the post-nonlinear (PNL) mixtures, a mirror structure BSS has been mainly used [7]. Nonlinear distortion is suppressed in the first stage assuming some prior conditions. Spline nonlinear functions or spline neural networks have been applied to the linearization process [3], [4]. Furthermore, a maximum likelihood estimator has been applied [5]. Also, neural networks have been applied [8].

Assuming nonlinearity is limited to low-order polynomial expressions, a BSS model cascading a separation block and a linearization block has been proposed [9],[10]. In the first block, signal groups, including the linear and high-order components, are separated. In the second block, the high-order components are suppressed. The number of the sensors is increased from that of the signal sources in order to cancel the noise components. The necessary number of the sensors can be theoretically determined.

In this paper, effects of the reduced number of the sensors on the separation performance is analyzed. Furthermore, in practical applications, nonlinearity is not so dominant, therefore, the separation performance is also analyzed based on the ratio of nonlinearity in the observed signals.

2. NOLINEAR MIXTURES

In this paper, the nonlinearity is expressed by polynomials. Thus, the observed signals include the high-order terms of the signal sources and the cross terms among the different signal sources. Letting s_i be the signal sources, and nonlinearity be a 2nd-order function, the observed signal $x_j(n)$ is expressed as

$$x_k = \sum_{i=1}^{n} a_{k,i} s_i + \sum_{i=1}^{n} \sum_{j=1}^{n} a_{k,ij} s_i s_j \quad (1)$$

Thus, x_k contains the original s_i, the high-order terms s_i^2, and the cross terms $s_i s_j, i \neq j$. Nonlinearity is not limited to post-nonlinearity, rather it can be included in the processes of generating, transmitting, and sensing signals. Equation (1) can express a general case. Order of nonlinearity is limited to 2nd or 3rd-order. However, in many practical applications, linear processing is a main part, and nonlinearity is parasitic phenomena, which can be approximated by low-order nonlinear functions.

If the signal sources are statistically independent, then $a_i s_i + b_i s_i^2$ and $a_j s_j + b_j s_j^2, i \neq j$ are also statistically independent, and can be separated by minimizing the mutual information [1]. The cross term $s_i s_j, i \neq j$ has some correlation with both s_i and s_j, then it can be suppressed through the above learning process.

3. SEPARATION BLOCK

3.1 Network Structure

A cascade form BSS is shown in Fig.1. The post-nonlinear (PNL) mixture model is used here [3], [4], [5]. However, this approach is not limited to the PNL mixtures, rather can cover a general case. First, the signal sources s_i are mixed through linear combination resulting in u_j. After that, they are transmitted through nonlinear functions F_k resulting in x_k.

3.2 Number of Sensors

In this model, in order to cancel the other components, the number of the sensors is increased from that of the signal sources. The observed signal x_k given by Eq.(1) includes $n(n-1)/2 + 2n$ terms. that is s_i and $s_i s_j$. In order to extract one group,

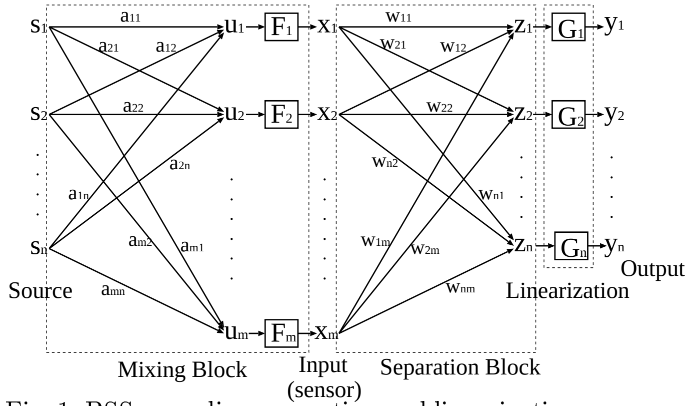

Fig. 1. BSS cascading separation and linearization blocks.

that is $a_{k,i}s_i + a_{k,ii}s_i^2$, the remaining part, which includes $n(n-1)/2 - 2n - 2$ terms, should be cancelled. Therfore, $n(n-1)/2 + 2n - 1$ equations are needed. This means $x_k, 1 \leq k \leq n(n-1)/2 + 2n - 1$ are required. Thus, the number of the sensors is $n(n-1)/2 + 2n - 1$.

One example is shown below. Two signal sources are received by four sensors.

$$x_1 = b_{11}s_1 + b_{12}s_2 + b_{13}s_1^2 + b_{14}s_1 s_2 + b_{15}s_2^2 \quad (2)$$
$$x_2 = b_{21}s_1 + b_{22}s_2 + b_{23}s_1^2 + b_{24}s_1 s_2 + b_{25}s_2^2 \quad (3)$$
$$x_3 = b_{31}s_1 + b_{32}s_2 + b_{33}s_1^2 + b_{34}s_1 s_2 + b_{35}s_2^2 \quad (4)$$
$$x_4 = b_{41}s_1 + b_{42}s_2 + b_{43}s_1^2 + b_{44}s_1 s_2 + b_{45}s_2^2 \quad (5)$$

The coefficient expression is different from Eq.(1) for simplisity. x_k can be treated as a constant. From these linear equations, one group, including s_i and s_i^2 components, is separated by cancelling the cross term $s_i s_j$ and the other components s_j and s_j^2. This can be done by minimizing the mutual information among the outputs z_i. The ideal results become

$$z_1 = c_{11}s_1 + c_{12}s_1^2 \quad (6)$$
$$z_2 = c_{21}s_2 + c_{22}s_2^2 \quad (7)$$

This process is equivalent to multiplying a vector $[x_1, x_2, x_3, x_4]^T$ by a 2×4 linear matrix $\boldsymbol{W} = \{w_{lk}\}$.

3.3 Learning Algorithm

In this block, the signal sources are separated based on their statistical independency. Therefore, the conventional learning algorithm, that is likelihood estimation minimizing the mutual information can be applied [1].

$$\boldsymbol{W}(n+1) = \boldsymbol{W}(n) + \eta[\boldsymbol{\Lambda}(t) - \varphi(\boldsymbol{z}(n))\boldsymbol{z}^T(n)]\boldsymbol{W}(n) \quad (8)$$

η is a learning rate, $\boldsymbol{\Lambda}(t)$ is a diagonal matrix, and $\varphi()$ is a nonlinear function [2].

4. LINEARIZATION BLOCK

4.1 Linearization Based on Solving Equations

At the outputs of the separation block, it is assumed that the signal sources are completely separated as shown in Eqs.(6) and (7). Since z_1 and z_2 include only s_1 and s_2, respectively, they can be linearized through the following nonlinear functions.

$$y_1 = G_1(z_1) = \frac{-c_{11} \pm \sqrt{c_{11}^2 + 4c_{12}z_1}}{2c_{12}} \quad (9)$$

$$y_2 = G_2(z_2) = \frac{-c_{21} \pm \sqrt{c_{21}^2 + 4c_{22}z_2}}{2c_{22}} \quad (10)$$

Finaly, the separated and linearized signal sources are obtained.

$$y_1 = d_1 s_1 \quad (11)$$

$$y_2 = d_2 s_2 \quad (12)$$

4.2 Learning Algorithm

Transformations in the linearization block are given by Eqs.(9) and (10). However, in real applications, the coefficients c_{ij} are not known. So, they should be adjusted through an iterative method. Equations (9) and (10) can be expressed by using two parameters as follows:

$$y_i(n) = -\frac{\alpha_i}{2} \pm \sqrt{\frac{\alpha_i^2}{4} + \frac{z_i(n)}{\beta_i}} \quad (13)$$

$$\alpha_i = \frac{c_{i1}}{b_{i2}}, \qquad \beta_i = \frac{1}{c_{i2}} \quad (14)$$

α_i and β_i are adjusted through an iterative method.

In this paper, 2nd-order nonlinearity is assumed. Thus, after the linear source separation, the outputs include 1st-order and 2nd-order terms of the signal sources. Furthermore, if we take speech and music signals into account, their average is almost zero. Therefore, the output average can be used as a cost function.

$$E_i(n) = \frac{1}{M}\sum_{l=0}^{M-1} y_i(n-l) \quad (15)$$

The gradient descent algorithm is used for adjusting the parameters.

$$\alpha_i(n) = \alpha_i(n-1) - \eta\frac{\partial E_i(n)}{\partial \alpha_i(n)} \quad (16)$$

$$\beta_i(n) = \beta_i(n-1) - \eta\frac{\partial E_i(n)}{\partial \beta_i(n)} \quad (17)$$

$$\begin{aligned}\frac{\partial E_i(n)}{\partial \alpha_i(n)} &= \frac{1}{M}\sum_{l=0}^{M-1}\frac{\partial y_i(n-l)}{\partial \alpha_i(n)} \\ &= \frac{1}{M}\sum_{l=0}^{M-1}(-\frac{1}{2} \pm \frac{\alpha_i(n)}{4}(\frac{\alpha_i^2(n)}{4} \\ &\quad + \frac{1}{\beta_i(n)}z(n-l))^{-\frac{1}{2}}) \quad (18)\end{aligned}$$

$$\begin{aligned}\frac{\partial E_i(n)}{\partial \beta_i(n)} &= \frac{1}{M}\sum_{i=0}^{M-1}\frac{\partial y_i(n-l)}{\partial \beta_i(n)} \\ &= \frac{1}{M}\sum_{l=0}^{M-1}(\mp\frac{z(n-l)}{2\beta_i^2}(\frac{\alpha_i(n)^2}{4} \\ &\quad + \frac{1}{\beta_i(n)}z(n-l))^{-\frac{1}{2}}) \quad (19)\end{aligned}$$

5. REDUCTION IN NUMBER OF SENSORS

As described in Sec.3.2, the number of the sensors is $n(n-1)/2 + 2n - 1$, where n is the number of the signal sources. When many signal sources exist, many sensors are requred. In practical applications, reduction in the number of the sensors is important to make a system compact. For this reason, effects of reduction in the number of the sensors is analyzed. This effect is also dependent on the ratio of the nonlinearity in the observed signals.

6. SIMULATIONS AND DISCUSSIONS

6.1 Simulation Conditions

Three signal sources and 4∼8 sensors are used. The signal sources are male and female speech signals. The mixing matrix is

$$A = \begin{bmatrix} 1 & 0.5 & -0.8 \\ 0.3 & -0.7 & 0.2 \\ -0.2 & 1 & 0.1 \\ 0.9 & 0.2 & -1 \\ -0.4 & -0.6 & 0.5 \\ 1 & 0.2 & 0.4 \\ 0.3 & -0.1 & 1 \\ -0.5 & -0.3 & -0.3 \end{bmatrix}$$

The learning rates are $\eta = 0.001$ and 0.5 in the separation block and the linearization block, respectively. The nonlinear functions in the mixing block are

$$F_1(u_1) = u_1 - 0.8u_1^2$$
$$F_2(u_2) = u_2 + 0.7u_2^2$$
$$F_3(u_3) = u_3 - 0.5u_3^2$$
$$F_4(u_4) = u_4 + 0.6u_4^2$$
$$F_5(u_5) = u_5 - 0.8u_5^2$$

$$F_6(u_6) = u_6 - 0.7u_6^2$$
$$F_7(u_7) = u_7 + 0.9u_7^2$$
$$F_8(u_8) = u_8 - 0.6u_8^2$$

6.2 SNR Evaluation

6.2.1. Separation Block Assuming s_i is dominant in z_j, and letting σ_{si}^2 and σ_{ni}^2 be the power of s_i in z_j, and the power of the remaining terms, respectively. SNR is defined by

$$SNR_1 = 10\log_{10}\frac{\sum_{i=1}^{n}\sigma_{si}^2}{\sum_{i=1}^{n}\sigma_{ni}^2} \quad (20)$$

Based on SNR_1 defined by the above equation, the separation block learning is not complete, because the high-order components s_i^2 cannot be suppressed.

6.2.2. Linearization Block The SNR evaluation is the same as that defined by Eq.(20). However, the same formula cannot be used, because s_i is not directly appeared. The s_i component and the other components are discriminated as follows:

$z_i(n)$ is linearized through

$$y_i(n) = -\frac{\alpha_i}{2} + \sqrt{\frac{\alpha_i^2}{4} + \frac{z_i(n)}{\beta_i}} \quad (21)$$

Let

$$\sqrt{\frac{\alpha_i^2}{4} + \frac{z_i(n)}{\beta_i}} = \sqrt{a_i s_i^2(n) + b_i(n)s_i(n) + c_i(n)} \quad (22)$$

Furthermore,

$$\sqrt{a_i s_i^2(n) + b_i s_i(n) + c_i(n)} = d_i s_i(n) + e_i(n) \quad (23)$$

$$a_i s_i^2(n) + b_i s_i(n) + c_i(n)$$
$$= d_i^2 s_i^2(n) + 2d_i s_i(n)e_i(n) + e_i(n)^2 \quad (24)$$

Comparing the coefficients, the following relations are obtained.

$$d_i^2 = a_i \quad (25)$$
$$2d_i e_i(n) = b_i(n) \quad (26)$$
$$e_i^2(n) = c_i(n) \quad (27)$$

a_i and $c_i(n)$ are calculated using α_i, β_i and $z_i(n)$ at each iteraton. SNR_2 is calculated by

$$SNR_2 = 10\log\frac{p(n)}{q(n)} \quad (28)$$

$$p(n) = \frac{1}{M}\sum_{i=0}^{M-1}(y_i(n) + \frac{\alpha_i}{2} - e_i(n))^2 \quad (29)$$

$$q(n) = \frac{1}{M}\sum_{i=0}^{M-1}(-\frac{\alpha_i}{2} + e_i(n))^2 \quad (30)$$

6.3 Simulation Results

6.3.1. SNR Improvement SNR_1 for x_i and z_j and SNR_2 for y_j, which are calculate by fixing the separation block and the linearization block after the learning converge, are shown in Fig.2. The vertical axis indicates SNR_1 or SNR_2 in dB, the horizontal axis is the sample number of the speech signals. From these results, this BSS method can improve $SNR_{1,2}$ by approximately 15 dB.

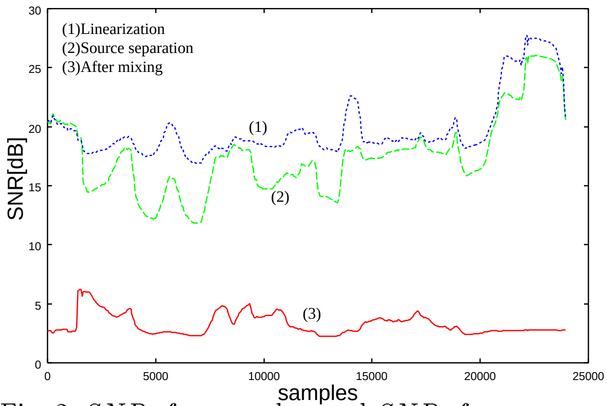

Fig. 2. SNR_1 for x_i and z_j and SNR_2 for y_j.

6.3.2. Relations between SNR and Number of Sensors The number of the sensors are changed from 4 to 8. The simulation results are shown in Fig.3. The case using 8 sensors is the best. However, another case using 7 sensors is close to the best. The cases using 4 and 5 sensors are not good.

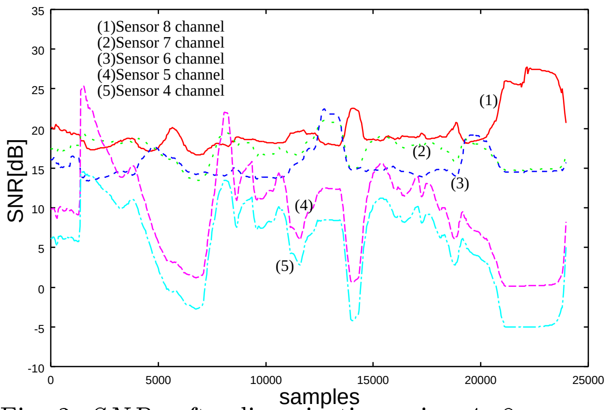

Fig. 3. SNR_2 after linearization using 4∼8 sensors.

In the next simulation, the ratio of the nolinearity is reduced to 50 percents of the previous example. The simulation results are shown in Fig.4. In this figure, difference among SNR obtained by using 6∼8 sensors becomes smaller than the previous. From these results, in practical applications, where the ratio of the nonlinearity is small, the reduced number of the sensors can provide good separation performance.

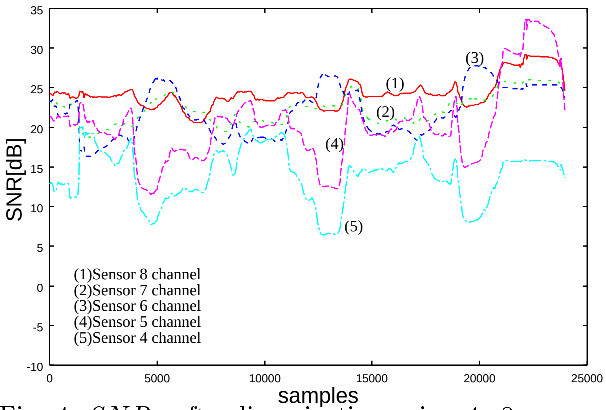

Fig. 4. SNR_2 after linearization using 4∼8 sensors. Ratio of nonlinearity is reduced.

7. CONCLUSIONS

In this paper, the BSS method cascading the separation block and the linearizatiion block in this order has been analyzed. SNR is improved by 15 dB after the linearization compared with the observed signals. Furthermore, the number of the sensors can be reduced for the small ratio of the nonlinearity in the observed signals, while maintaining good separation performance.

REFERENCES

[1] S.Amari, T.Chen and A.Cichocki, "Stability analysis of learning algorithms for blind source separation", Neural Networks, vol.10, no.8, pp.1345-1351, 1997.

[2] K.Nakayama, A.Hirano and T.Sakai, "An adaptive nonlinear function controlled by estimated output PDF for blind source separation", Proc. ICA'2003, Nara, pp.427-432, April 2003.

[3] M.Solazzi, F.Piazza and A.Uncini, "Nonlinear blind source separation by spline neural networks", IEEE Proc. ICASSP'2001, Salt Lake City, MULT-P3.4, May 2001.

[4] F.Milani, M.Solazzi and A.Uncini, "Blind source separation of convolutive nonlinear mixtures by flexible spline nonlinear functions", IEEE Proc. ICASSP'2002, Orlando, pp.1641-1644, May 2002.

[5] A.Koutras,"Blind separation of non-linear convoluved speech mixtures", IEEE Proc. ICASSP'2002, Orlando, pp.913-916, May 2002.

[6] C.Jutten and J.Karhunen, "Advance in nonlinear blind source separation", ICA03, Nara, pp.245-256, April 2003.

[7] A.Ziehe, M.Kawanabe, S.Harmeling and K.R.Muller, "Blind separation of post-nonlinear mixtures using Gaussianizing transformations and temporal decorrelation", Proc. ICA'03, Nara, pp.269-274, April 2003.

[8] R.M.Clemente, S.H.Mellado, J.I.Acha, F.Rojas and C.G.Puntone., "MLP-based source separation for MLP-like nonlinear mixtures", Proc. ICA'03, Nara, pp.155-160, April 2003.

[9] K.Nakayama, A.Hirano and T.Nishiwaki", A cascade form blind source separation connecting source separation and linearization for nonlinear mixtures", IEEE&INNS Proc. IJCNN'03, Portland, Oregon, July 2003.

[10] T.Nishiwaki, K.Nakayama and A.Hirano, "A blind source separation cascading separation and linearization for low-order nonlinear mixtures", IEEE Proc. ICASSP2004, Montreal, May 2004, to be presented.