TIME-FREQUENCY DOMAIN BLIND SOURCE SEPARATION
–THE INDEPENDENCE PROBLEM AND PROPOSED SOLUTION

Xuebin Hu and Hidefumi Kobatake
Graduate School of Bio-Applications and Systems Engineering, Tokyo University of Agri. & Tech.
2-24-16 Naka-cho, Koganei-shi, Tokyo 184-8588, Japan

ABSTRACT

Time-frequency domain blind source separation (BSS) leads to a seldom mentioned but important problem that generally the independence assumption between source signals collapses in frequency domain due to inadequate samples. It consequently degrades the performance of all the ICA-based BSS methods. In this paper, we set up a criterion on the performance of separation at each frequency bin and propose a recursive algorithm to correct the bin separations which are thought improper. The bin mixtures are separated into the components of the sources as practical instead of the "independent" bins as achieved by the conventional ICA. The signal-to-noise ratio is greatly increased at certain bins, which results in a much better separation.

1. INTRODUCTION

Blind source separation has received extensive attention in signal and speech processing, machine intelligence, and neuroscience communities. The goal of BSS is to recover the unobserved source signals without any prior information given only the sensor observations that are unknown linear mixtures of the independent sources. A variety of successful ICA methods have been developed for this purpose [1-5].

Due to the multi-path effect and reverberation in real environment, computationally blind speech separation is often implemented in time-frequency domain. A number of approaches for the convoluted source separation have been reported. The formation of BSS could be summarized as follows. Source signals are assumed to be independent with each other, zero mean, and are denoted by a vector $\boldsymbol{s}(t) = (s_1(t), \cdots, s_N(t))^T$. When the signals are recorded in a real environment, the observations can be approximated with convolutive mixtures of source signals,

$$\boldsymbol{x}(t) = \boldsymbol{A} * \boldsymbol{s}(t) = \left(\sum_i a_{ik} * s_i(t) \right), \qquad (1)$$

where $\boldsymbol{A}$ is an unknown polynomial matrix, a_{ik} is the impulse response from source i to microphone k, and the symbol * refers to convolution. In frequency domain, the convolutive mixing problem is decomposed into multiple instantaneous mixing problems.

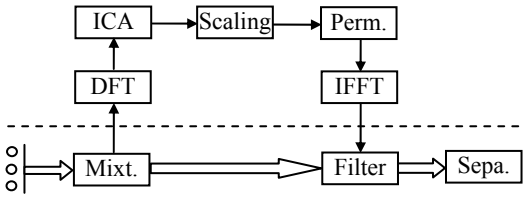

Fig 1. Flow chart of BSS system. Perm.: Permutation. Mixt.: Mixture. Sepa.:Separation. The process above the dot line is in frequency domain, and the process below is in time domain.

$$\boldsymbol{X}(f,t) = \boldsymbol{A}(f)\boldsymbol{S}(f,t). \qquad (2)$$

The instantaneous mixing problem then can be solved using any desired ICA method. With the derived unmixing filter $\boldsymbol{W}(f)$, we recover the source signals by

$$\hat{\boldsymbol{S}}(f,t) = \boldsymbol{PDW}(f)\boldsymbol{X}(f,t). \qquad (3)$$

where, $\boldsymbol{P}$ and $\boldsymbol{D}$ are the solution to the ambiguity of permutation and scaling. The bin unmixing filters are then transferred into time domain unmixing filter. Fig 1 shows the commonly adopted process flow of conventional time-frequency domain implementation.

2. THE PROBLEM

Due to the dynamic mixing process in real environment, BSS is normally implemented on a short time period of observations. Frequency domain implementation leads to much less samples than that in time domain. Speech signals normally are non-stationary. As a result, there often exists large estimation error in the second and higher order statistics. For example, the correlation function between the source signals can no longer be expected to be zeros. We say that the independence assumption collapses in frequency domain [6]. The frequency components of source signals, corresponding to the observations of the limited samples, are *correlated*.

For evaluation of the correlation between $\boldsymbol{s}_1(f,t), \cdots, \boldsymbol{s}_K(f,t)$, we define the following matrix:

$$\boldsymbol{V}(f) = diag\left(\left\langle \boldsymbol{S}(f,t)\boldsymbol{S}^H(f,t) \right\rangle \right) - \left\langle \boldsymbol{S}(f,t)\boldsymbol{S}^H(f,t) \right\rangle, \qquad (4)$$

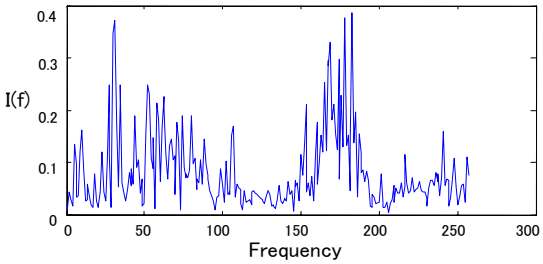

Fig 2. Correlation distribution against frequency

where $\langle\cdot\rangle$ denotes the expectation operator, and $\langle \boldsymbol{S}(f,t)\boldsymbol{S}^{\mathrm{H}}(f,t)\rangle$ is the normalized covariance matrix. The correlation between the frequency components of source signals, denoted as $I(f)$, is quantified by the Frobenius norm of $\boldsymbol{V}(f)$. It is further normalized and defined as,

$$I(f)=\|\boldsymbol{V}(f)\|=\sqrt{\sum_{l\neq k}\left|\boldsymbol{V}_{lk}(f)\right|^2\Big/(N^2-N)} \qquad (5)$$

The higher the value $I(f)$ is, the lower the independence will be. Fig. 2 shows the correlation distribution against frequency between two speech samples, "Good morning" and "Konbanwa". The signals are about one second in length, and the sampling rate is 16kHz. The DFT length is 512 samples (32msec) and the window shift is 5 samples. Hamming window is used. It is noted that the correlation may vary in some degree when using different frame length and window shift. But it will not cause significant change to it.

Regardless of the ICA method, the off-diagonal elements of the covariance matrix of the separated signals are to be minimized as practical, ideally minimized to zeros. In other words, ICA will make $I(f)$ close to or equal to zero. The existing correlation between the sources as shown in Fig.2 apparently shows that ICA will not work perfectly in such case. It degrades the performance of ICA, sometimes makes the separation completely failed. Furthermore, it makes the solution to the ambiguity of scaling and permutation more difficult.

3. DETECTION OF IMPROPER SEPARATION

Fig.3 shows an example of improper separation. Y1 and Y2 are the separated results achieved by the time delayed decorrelation method (TDD) [4], and S1, S2 are the components of original sources. The waveforms are the norms of the complex-valued signals respectively. Although Y1 and Y2 are uncorrelated with each other, comparing with S1 and S2, it is obvious that Y1 and Y2 are quite similar at certain segment pairs, for example at [1900, 2200] and [2400, 3000].

The criterion is set up on the above observation. We divide the separated signals $\hat{\boldsymbol{S}}(f,t)$ into a number of continuous segments $\hat{\boldsymbol{S}}(f,l)$, and use the averaged

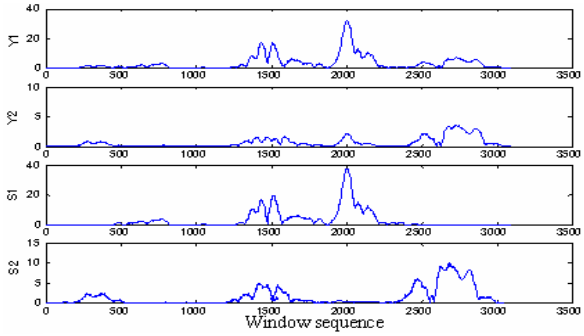

Fig 3. An example of improper separation

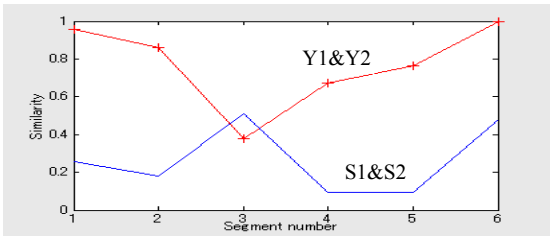

Fig 4. Comparison of segmental similarities between improperly separated signals and original signals.

segmental similarity (ASS) as the criterion to make the decision. The ASS is defined as,

$$ASS_f=\frac{1}{L}\sum_{l=1}^{L}\frac{\left\|\hat{\boldsymbol{S}}_1(f,l)\cdot\hat{\boldsymbol{S}}_2(f,l)^{\mathrm{H}}\right\|}{\left\|\hat{\boldsymbol{S}}_1(f,l)\right\|\left\|\hat{\boldsymbol{S}}_2(f,l)\right\|}. \qquad (6)$$

$$\hat{\boldsymbol{S}}(f,l)=\hat{\boldsymbol{S}}(f,(l-1)\times m+1:l\times m), \qquad (7)$$

where, l is the segment sequence number and m is the segmental length.

Figure 4 shows the segmental similarities between Y1, Y2 and S1, S2, respectively. They are divided into six segment pairs with m equals to 500, respectively. We can see that the average of the segmental similarities of improper separation (Y1, Y2) is higher than that of complete separation (S1, S2). In other words, those with higher averaged segmental similarity tend to be improper separation.

4. CORRECTION

4.1. Method

The ideal correction method is to make the separation work on the mixtures of which the source signals become independent. However, it is impractical because the correlation comes from the whole signal components. Then we turn to assume that among the whole signal component, a few of the segments play a critical role in the formation of the correlation, or in other words, a few of the segments have critical effect on the improper separation. We try to take away these segments and make

the separation work on the remaining mixtures of which the corresponding source signals become less correlated. Then we can use the derived unmixing filter to separate the original mixtures and expect a better separation.

We use two trial-and-error schemes to find these segments. One is called scheme 1 (exhaustive search) in which all combinations of segments are searched to detect segments leading to improper separation. However, the number of combinations of segments becomes huge. Computationally, it is too heavy to search every combination. In consideration that the segment with higher variance has heavier weight in the learning of the unmixing filter, the scheme 2 selects the segments with high variance as candidates to be removed. Of course the segments with high variance actually are not necessarily the reason for the improper separation and also convey important cues in the separation. To avoid worse separation, we use ASS to distinguish the result.

4.2. Algorithm

Because there isn't an exact correspondence between ASS and the performance of separation, we set up two thresholds in the algorithm. Threshold 1 is set up for distinguishing the bin separation, whether the correction is needed or not. Threshold 2, which is lower than threshold 1, is set up for making the decision whether to accept the new separation and stop the loop or not.

The process flow of the scheme 2 within each frequency bin is depicted in Fig 5. First if ASS is greater than threshold 1, the separation is considered to be improper and needed to be corrected. We remove the segments with high variance in sequence until ASS is below threshold 2. N is the designated maximal iteration times. That means, among the whole L segments, totally there might be N segments to be removed. N should at least be smaller than L/2 because too short samples cannot ensure good derivation of the unmixing matrix. When the maximal iteration times are reached and threshold 2 still cannot be satisfied, among the iterations, the separation that outputs the minimum ASS is selected as a final result.

5. SIMULATION TESTS

In the simulation tests, the proposed method is used combined with the time delayed decorrelation method (TDD), Infomax and Jade, respectively [1, 4-5].

5.1. Simulation 1: Effect of the method

First, the performance of the recursive method was evaluated in the simplest condition. The same signals as

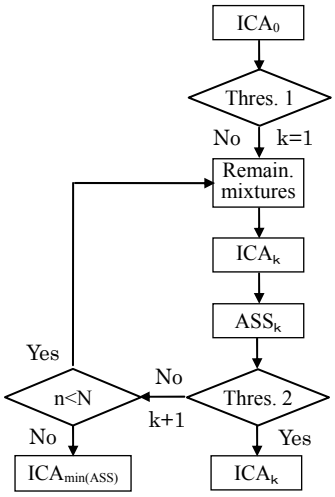

Fig 5. Process flow of the correction algorithm (scheme 2).

in section 2 were used, where the mixing matrix was [1,0.5; 0.7,1]. The mixtures were divided into frames of 32 msec length each and the window shift was 5 samples. The length of FFT is 512, and Hamming window was used. Threshold 1 and 2 were set to 0.55 and 0.4 respectively. Experimentally, segment number L equals to 15-20 is adequate to provide a considerable improvement. Here L and m were set to 15 and 200, respectively. The maximal iteration times N was 7. We compared our results with those of the conventional methods.

Figure 6 shows a significantly improved example achieved at one frequency bin. The 1st and 4th waveforms are the norm of the separation results $\hat{\boldsymbol{S}}(f,t)$ achieved by TDD. Comparing with the 3rd and 6th waveforms of the original sources, the 1st and 4th waveforms are far away from the 3rd and 6th waveforms respectively. The separation is improperly implemented. The 2nd and 5th waveforms are those of the result of the recursive method. They are very close to the original sources and obviously much better than the 1st and 4th waveforms. Apparently the result achieved by the recursive method, better both in amplitude and waveform, will make it easy to do a proper selection on the ambiguity of permutation.

Figure 7 shows the comparison of SNR and ASS between TDD and the recursive method. Considerable improvement in SNR has been achieved. At the designated threshold 1, about 40 percent of the bins were considered to be improperly separated and were corrected by the method. Accompanying with the improvement of SNR, ASS decreased about 0.17 in average. Although SNR became a little worse at a few bins, an overall improvement is achieved. Fig 8 shows the averaged results of 24 trails on different pairs of speeches using TDD, Infomax and Jade, respectively.

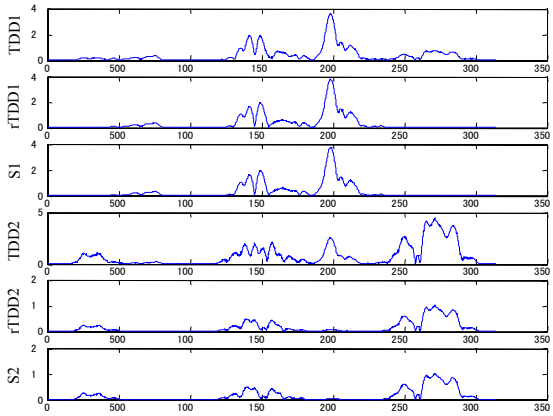

Fig 6. An example of significant improvement

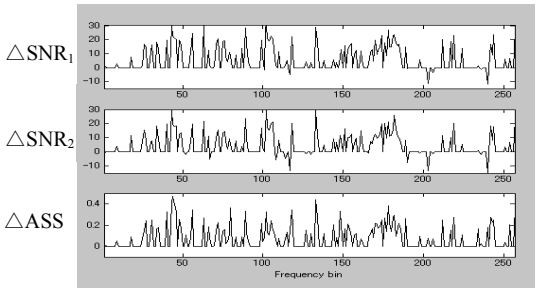

Fig 7. The upper two figures show the improvement of SNR at source 1 and 2, respectively. $\triangle$SNR=SNR_{rTDD}-SNR_{TDD}. The lower figure depicts the difference of ASS before and after the recursive method. $\triangle$ASS=ASS_{TDD}-ASS_{rTDD}.

5.2. Simulation 2: Convolutive mixtures

We tested the method on convolutive mixtures. We trialed on 24 pairs of speech signals from ASJ Continuous Speech Database. They were mixed using the filters in eq. (8).

$$\begin{aligned} A_{11} &= 0.9 + 0.5z^{-1} + 0.3z^{-2} \\ A_{12} &= -0.7z^{-5} - 0.3z^{-6} - 0.2z^{-7} \\ A_{21} &= 0.5z^{-5} + 0.3z^{-6} + 0.2z^{-7} \\ A_{22} &= 0.8 - 0.1z^{-1} \end{aligned} \quad (8)$$

Same parameters were used as in 5.1 with the exception that: window shift was 20, L and N equaled to 20 and 8. Two and half seconds length of the mixed speech signals were used to learn the unmixing filter. The averaged noise reduction ratios (NRR) are shown in Fig 9. The recursive method gave about 3.4, 3.2 and 3.5 dB higher NRR in average than those of the conventional TDD, Jade and Infomax, respectively.

A comparison test between the scheme of removing high variance segments and exhaustive search was done. Table 1 shows the comparison result. The scheme of removing high variance segments gives a little worse SNR than the exhaustion one but costs much less computational loads.

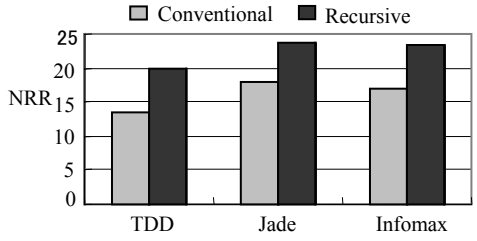

Fig 8. Separation result of simulation 1.

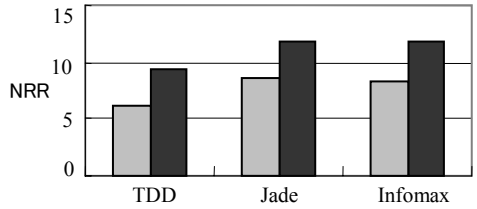

Fig 9. Separation result of simulation 2.

Table 1. Comparison between scheme 1 (exhaustive search) and scheme 2 (removing high variance segments).

	Normal	Scheme 1	Scheme 2
Maximum iteration times	1	$>10^5$	8
Averaged SNR	11.7 dB	16.5 dB	15.3 dB

6. CONCLUSION

This paper addressed the correlation problem existing in the time-frequency domain blind source separation. A criterion was set up for distinguishing the improper separation. And we proposed a recursive method to correct the one that were thought improper. Simulation tests proved its effectiveness.

REFERENCE

[1] A. Bell, and T. Sejnowski, "An information maximization approach to blind separation and blind deconvolution," *Neural Computation*, 7: 1129-1159, 1995.

[2] A. Hyvarinen and E. Oja, "A fast fixed-point algorithm for independent component analysis," *Neural Computation*, 9:1483-1492, 1997.

[3] S. Amari, A. Cichocki, and H. H. Yang, "A new learning algorithm for blind signal separation," In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, "*Advances in Neural Information Processing Systems*, vol. 8, pp. 757-763, MIT press, 1996.

[4] A. Belouchrani, K. Abed-Meraim, J. -F. Cardoso, and E. Moulines, "A blind source separation technique using second-order statistics," *IEEE Trans. Signal Processing*, vol. 45(2), pp. 434-443, February 1997.

[5] J.-F. Cardoso and A. Souloumiac, "Blind beamforming for non-Gaussian signals", *IEE Proceedings F,* vol. 140, No. 6, pp. 362-370, 1993.

[6] T. Nishikawa, S. Araki and S. Makino, "Optimization on the number of subbands in blind source separation with subband ICA," Proceedings of Acoustical Society Japan, Mar. 2001 (In Japanese).