# On the robustness of distributed EM based BSS in asynchronous distributed microphone array scenarios

*Yasufumi Uezu*[1], *Keisuke Kinoshita*[2], *Mehrez Souden*[2], *Tomohiro Nakatani*[2]

[1]Kyushu University, Fukuoka, Japan
[2]NTT Communication Science Laboratories, Kyoto, Japan
uezu@s.kyushu-u.ac.jp

## Abstract

Distributed microphone array (DMA) processing has recently started attracting a lot of attention as a promising alternative to conventional microphone arrays with co-located elements. To perform efficient blind source separation (BSS) in distributed manner, we have recently proposed a clustering based BSS approach that utilizes a distributed expectation-maximization algorithm. In this paper, we further investigate the effectiveness of this proposed method in typical DMA situations with drift error, i.e., the inevitable sampling frequency mismatch between different nodes/devices, which is a very important aspect that has yet to be examined. We compare our method experimentally with some conventional methods and their variants newly proposed here, and confirm that our method is robust against drift error and achieves stable performance in various typical DMA scenarios, while the performance of the conventional approaches changes greatly depending on the DMA setup and amount of drift error.

**Index Terms**: Distributed microphone array, blind source separation, drift error, synchronization.

## 1. Introduction

Single- and multi-channel speech enhancement technologies have been extensively investigated over the last decades to cope with the acoustic interference recorded along with the target speech signal, with the aim of improving both the quality of recorded speech signal and the performance of automatic speech recognition. This acoustic interference includes ambient noise and speech from non-target speakers. Single-channel speech enhancement algorithms [1–3], which work with only 1 microphone, tend to impose fewer hardware restrictions in general, and thus can be employed conveniently in wide range of application areas. However, since the single-channel approaches do not use spatial information, which is reported to be a reliable cue for distinguishing target speech and interference, their performance often degrades greatly in such realistic adverse environments as meeting scenarios with several competing speakers. In these situations, multi-channel speech enhancement algorithms [4–7] provide superior performance by taking advantage of such cues, although it leads to the fact that it requires a large carefully designed microphone array. This fact may prevent the multi-channel technologies from leaving the research laboratory and spread widely in our daily lives.

To this end, research on the distributed microphone array (DMA) has recently started attracting a lot of attention as a promising alternative to conventional microphone array processing [8–13]. DMA here represents an array composed of, for example, multiple independent collaboratively-working recording devices such as IC recorders or smart phones. In such cases, each device is referred to as a *node* within the DMA.

A DMA scenario inherently includes some challenging but interesting research topics. The first apparent topic is the research on collaborative processing between nodes [11–13]. In [11], on the assumption that each node deals with overdetermined problem, a distributed independent component analysis (ICA)-based algorithm was proposed to accomplish a global blind source separation (BSS) task by regularizing the local ICA process with Wiener-like weighting. In [13], by extending the clustering based BSS proposed in [7], we proposed a DMA-based BSS approach that does not require such a node-wise overdetermined assumption. [13] casts the BSS task as a clustering problem assuming sparseness of the speech signal [6], and performs clustering of observed features using the distributed expectation-maximization (EM) algorithm, within which we perform local clustering/separation at each node, fuse the hypotheses from all the nodes, and feed back the refined hypotheses to each node. Experimental results showed that the performance of this DMA-based method approaches that of global processing [7]. The second research topic for DMA is an investigation into the way of dealing with synchronization problems between nodes [9, 10]. There, synchronization problems can be further divided into two categories, namely, the problem of identifying and synchronizing the starting point of the recording between nodes, and the problem of dealing with the inevitable mismatch between the sampling frequencies of the nodes/recording devices, i.e., drift error. While the former synchronization problem has been partly solved in the literature on conventional microphone arrays, there are few studies [9, 10] focusing on the latter problem. Lienhart et al. [9] developed a system for synchronizing the signals by having the microphone devices send special synchronization signals over a dedicated link, and performing ICA. The method proposed in [10] blindly estimates the amount of drift error in a rather heuristic way by performing re-sampling and ICA-based separation repeatedly, and adjusting the sampling frequencies between nodes.

In this paper, we carefully investigate the effect of drift error on the general clustering based BSS algorithm [7] with different features and the DMA-based BSS [13], which has not been undertaken in previous studies. Specifically, to obtain a new insight regarding the general relationship between clustering based BSS and drift errors, we examine these algorithms in some typical DMA scenarios using various feature vectors for clustering, some of which are first proposed/tested here. In subsequent sections, we first describe the data model, and the above methods. Then, in an experiment, we evaluate the effect of drift errors on their performance, and provide comprehensive discussions that consequently suggest that our DMA-based BSS algorithm [13] can perform very robustly in the presence of drift errors in a wide range of DMA scenarios.

## 2. data model

In this paper, we consider the case where $L > 1$ competing speech signals are recorded by a DMA containing $N$ nodes. In such an environment, at time frame $t$ and frequency $k = 1, ..., K$, where $K$ is the number of frequency bins, the observed signal at the $n$th node is expressed in the short time Fourier transform domain as

$$\begin{aligned} \mathbf{y}_n(k,t) &= \sum_{l=1}^{L} \mathbf{g}_{n,l}(k) S_l(k,t), \quad (1) \\ \mathbf{y}_n(k,t) &= [y_{n,1}(k,t), ..., y_{n,M_n}(k,t)]^T. \end{aligned}$$

Here $M_n$ indicates the number of microphones in the $n$th node, and total number of microphones in a DMA is $M$. $S_l(k,t)$ corresponds to the $l$th speech source signal, and $\mathbf{g}_{n,l}(k) = [G_{n,1,1}(k), ..., G_{n,M_n,l}(k)]^T$ contains the channel transfer functions between the $l$th source and the microphones of the $n$th node. Furthermore, we define the global observation vector as

$$\mathbf{y}(k,t) = \left[\mathbf{y}_1(k,t)^T, ..., \mathbf{y}_N(k,t)^T\right]^T. \quad (2)$$

Hereafter, we drop the frequency index $k$ in our notations because all our processing is performed frequency bin-wise.

## 3. Clustering-based BSS algorithms

In this section, we explain the clustering-based BSS algorithms [7, 13] and some of their potential variants, all of which we employ for performance analysis.

### 3.1. Global clustering-based algorithm (Global method)

The global clustering-based algorithm, hereafter referred to as "global method", performs the BSS task with the assumption that the algorithm has access to the global observation vector, $\mathbf{y}(t)$, or more specifically the features calculated based on $\mathbf{y}(t)$. One of the typical features used for this method is the normalized observation vector expressed as

$$\boldsymbol{\psi}(t) = \frac{\mathbf{y}(t)}{\|\mathbf{y}(t)\|}. \quad (3)$$

Some other possible variants of this feature are described in Section 3.3. The normalization operation for the recording vector maps the recording onto a complex unit hyper-sphere, and conceptually the feature in eq. (3) can be seen as some sort of an approximation of the set of transfer functions [7]. When modeling such a vector for each source, it is known that we can follow the line orientation idea in [14, 15] and employ the following mixture model,

$$p(\boldsymbol{\psi}(t)|\theta) = \sum_{l=1}^{L} \alpha_l p(\boldsymbol{\psi}(t)|H_l; \mathbf{a}_l, \sigma_l), \quad (4)$$

where

$$p(\boldsymbol{\psi}(t)|H_l) = \frac{1}{(\pi\sigma_l^2)^{M-1}} \exp\left(-\frac{\|\boldsymbol{\psi}(t) - \left[\mathbf{a}_l^H \boldsymbol{\psi}(t)\right] \mathbf{a}_l\|^2}{\sigma_l^2}\right), \quad (5)$$

where $\theta = \{\alpha_1 ... \alpha_L, \mathbf{a}_1 ... \mathbf{a}_L, \sigma_1 ... \sigma_L\}$, and $\mathbf{a}_l$ and $\sigma_l^2$, respectively, represent the centroid and spread of the $l$th cluster. The *a priori* probabilities of $H_l$, $\alpha_l = p(H_l)$ sum to 1. Note that, since speech is sparse [6], each of the modes of such distribution is concentrated around the normalized observation vector calculated based on one of the $L$ competing sources. Consequently, separating the speech sources amounts to determining a latent variable, $H$, that identifies at a given time-frequency slot the most likely speech source among $L$ sources. In other words, the clustering and separation of recorded sounds becomes possible by determining the $L$ posterior probabilities $p(H_l|\psi(t)), l = 1, ..., L.$, which indicate the dominance/activity of the $l$-th source at each time-frequency slot.

To determine the posterior probabilities $p(H_l|\psi(t))$, we employ the EM algorithm that estimates posterior probabilities as a byproduct of the estimation of the parameter set $\theta$. The EM algorithm repeats the following E-step and M-step until convergence. In the E-step, the posterior probability is calculated via Bayes' rule as

$$p(H_l|\boldsymbol{\psi}(t); \theta') = \frac{\alpha_l' p(\boldsymbol{\psi}(t)|H_l; \mathbf{a}_l', \sigma_l')}{\sum_{l=1}^{L} \alpha_l' p(\boldsymbol{\psi}(t)|H_l; \mathbf{a}_l', \sigma_l')}, l = 1, ..., L, \quad (6)$$

Hereafter, the superscript " ' " is used to express the prior estimate of the corresponding parameter. In the M-step, the set of parameters is updated by maximizing the following auxiliary function.

$$\begin{aligned} Q_c(\theta, \theta') &= E\{\log(p(\boldsymbol{\psi}(t), H; \theta))|\boldsymbol{\psi}(t); \theta'\} \quad (7) \\ &= \sum_{t=1}^{T} \sum_{l=1}^{L} p(H_l|\boldsymbol{\psi}(t); \theta') \log(\alpha_l p(\boldsymbol{\psi}(t)|H_l; \mathbf{a}_l', \sigma_l')) \end{aligned}$$

where $\theta'$ is the prior estimate of $\theta$, and $T$ is the number of samples. In the end, the update of centroid $\mathbf{a}_l$ is the eigenvector corresponding to the maximum eigenvalue of:

$$\mathbf{R}_l = \sum_{t=1}^{T} p(H_l|\boldsymbol{\psi}(t); \theta') \boldsymbol{\psi}(t) \boldsymbol{\psi}^H(t), \quad (8)$$

while $\sigma_l^2$ and $\alpha_l$ are obtained as

$$\sigma_l^2 = \frac{\sum_{t=1}^{T} p(H_l|\boldsymbol{\psi}(t); \theta') \|\boldsymbol{\psi}(t) - (\mathbf{a}_l^H \boldsymbol{\psi}(t)) \mathbf{a}_l\|^2}{(M-1) \sum_{t=1}^{T} p(H_l|\boldsymbol{\psi}(t); \theta')}, \quad (9)$$

$$\alpha_l = \frac{1}{T} \sum_{t=1}^{T} p(H_l|\boldsymbol{\psi}(t); \theta'). \quad (10)$$

### 3.2. Distributed fusion-based algorithm

The distributed-fusion-based algorithm, hereafter referred to as "fusion method", performs the BSS task with access only to features calculated based on the local observation vector, $\mathbf{y}_n(t)$, and supplemental compressed information from the other nodes [13]. A typical feature used for this method is:

$$\boldsymbol{\psi}_n(t) = \frac{\mathbf{y}_n(t)}{\|\mathbf{y}_n(t)\|}. \quad (11)$$

Since we now have access only to the node specific observation, we are not able to estimate the posterior probabilities $p(H_l|\psi(t))$ in the same manner as with the global method. To this end, we define node-specific posterior probability $p(H_l|\boldsymbol{\psi}_n(t); \tilde{\theta})$, and introduce the following independence assumption between nodes to bridge between $\psi_n(t)$ and $\psi(t)$.

$$p(\boldsymbol{\psi}(t)|H_l; \theta) = \prod_{n=1}^{N} p(\boldsymbol{\psi}_n(t)|H_l; \tilde{\theta}). \quad (12)$$

This assumption indicates that each node observes independent location feature vectors for a given source.

Based on this assumption, we can derive the following *distributed* EM algorithm to estimate the global posterior

probability $p(H_l|\boldsymbol{\psi}(t);\theta)$ and node-specific parameters, $\tilde{\theta} = [\alpha_1, \ldots \alpha_L, \mathbf{a}_{1,1}, \ldots, \mathbf{a}_{N,L}, \sigma_{1,1}, \ldots, \sigma_{N,L}]$. This EM algorithm consists of two E steps and an M step. Firstly, the initial E-step estimates node-specific posterior probabilities $p(H_l|\boldsymbol{\psi}_n(t);\tilde{\theta})$ at each node, similarly to eq. (6), but by using a node-specific centroid $\mathbf{a}_{n,l}$ and variance $\sigma_{n,l}$. Then, the second E-step fuses the posterior probabilities of all the nodes and calculates the global posterior probability $p(H_l|\boldsymbol{\psi}(t);\tilde{\theta})$, which is assumed to be common to all the nodes, as

$$p(H_l|\boldsymbol{\psi}(t);\tilde{\theta}') = \xi_l(\boldsymbol{\psi}(t);\tilde{\theta}') \cdot \chi(\boldsymbol{\psi}(t);\tilde{\theta}') \tag{13}$$

where $\chi(\boldsymbol{\psi}(t);\tilde{\theta}')$ is a normalization term that is independent of $l$,

$$\xi_l(\boldsymbol{\psi}(t);\tilde{\theta}') = {\alpha_l'}^{1-N} \prod_{n=1}^{N} p(H_l|\boldsymbol{\psi}_n(t);\tilde{\theta}'). \tag{14}$$

Then, M-step estimates the node-specific centroid $\mathbf{a}_{n,l}$ and variance $\sigma_{n,l}$ at each node by using the global posterior probability $p(H_l|\boldsymbol{\psi}(t);\tilde{\theta})$ such that it maximizes the following auxiliary function,

$$Q_d(\tilde{\theta},\tilde{\theta}') = Q_{d1}(\tilde{\theta},\tilde{\theta}') + Q_{d2}(\tilde{\theta},\tilde{\theta}') \tag{15}$$

where $Q_{d1}$ and $Q_{d2}$ are given by

$$Q_{d1}(\tilde{\theta},\tilde{\theta}') = \sum_{t=1}^{T}\sum_{l=1}^{L} p(H_l|\boldsymbol{\psi}(t);\tilde{\theta}') \log(\alpha_l) \tag{16}$$

$$Q_{d2}(\tilde{\theta},\tilde{\theta}') = \sum_{n=1}^{N}\sum_{l=1}^{L}\sum_{t=1}^{T} p(H_l|\boldsymbol{\psi}(t);\tilde{\theta}) \log(p(\boldsymbol{\psi}_n(t)|H_l;\mathbf{a}_{n,l},\sigma_{n,l})) \tag{17}$$

This procedure basically boils down to the calculation of eq. (8) using local recordings only at each node and eq. (10) [13].

It is very important to note that, as we can see from eq. (13), the information exchanged between nodes, which may have different sampling frequencies due to drift error, consists simply of posterior probabilities indicating source activities. Since the source activities are very similar at neighboring frequencies, and may not differ significantly due to the small difference in sampling frequency, we can expect the overall process to be less sensitive to drift error.

### 3.3. Other variants of the clustering feature

The normalized observation vector introduced in eq. (3) is reported to be effective in BSS tasks with ordinary co-located microphone arrays with no drift error [7]. Since this feature can be viewed conceptually as an approximation of a set of transfer functions in the *complex* spectrum domain, it allows us to utilize both the phase and amplitude differences to distinguish each source signal. However, if the observed signal is captured by the DMA with some amount of drift error, the phase difference information may turn out to be unreliable, since such information is obtained through the exact calculation of the time difference between microphones. On the other hand, the amplitude difference may not be very distorted as long as the time discrepancy between microphones does not exceed the analysis frame length. To carry out a detailed analysis of the global method against drift error and demonstrate the advantage of the fusion method, we propose the following features for the global method, namely, the phase-only feature $\boldsymbol{\psi}^{\text{Ph.}}(t)$ and the amplitude-only feature $\boldsymbol{\psi}^{\text{Amp.}}(t)$.

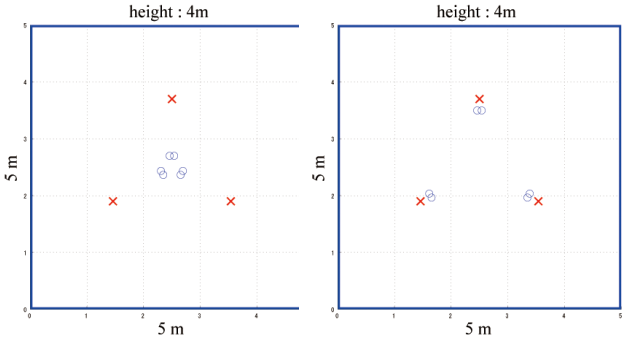
$$\boldsymbol{\psi}^{\text{Ph.}}(t) = \frac{1}{\sqrt{M}}\left[\exp\left(j\angle\frac{y_{1,1}(t)}{\|\mathbf{y}(t)\|}\right), ..., \exp\left(j\angle\frac{y_{N,M_N}(t)}{\|\mathbf{y}(t)\|}\right)\right]^T$$

$$\boldsymbol{\psi}^{\text{Amp.}}(t) = \left[\left|\frac{y_{1,1}(t)}{\|\mathbf{y}(t)\|}\right|, \ldots, \left|\frac{y_{N,M_N}(t)}{\|\mathbf{y}(t)\|}\right|\right]^T$$

Since these features can be viewed as degenerated versions of $\boldsymbol{\psi}(t)$, it can be still modeled with the probability density function in eq. (4).

Table 1: The list of the compared methods

| | Clustering algorithm | Feature for clustering |
|---|---|---|
| 1 | Global method | Amp. & ph. : $\boldsymbol{\psi}(t)$ |
| 2 | Global method | Amp. only : $\boldsymbol{\psi}^{\text{Amp}}(t)$ |
| 3 | Global method | Ph. only : $\boldsymbol{\psi}^{\text{Ph}}(t)$ |
| 4 | Fusion method | Amp. & ph. : $\boldsymbol{\psi}_n(t)$ |




(a) Co-located nodes (b) Distributed nodes

Figure 1: Simulated DMA environments

## 4. Experiments

In this section, we evaluate the performance of the aforementioned clustering based BSS algorithms in DMA scenarios with different drift errors with a view to gaining new insights into their behavior.

### 4.1. Experimental conditions

#### *4.1.1. Methods for comparison*

In Table 1, we summarize the compared methods. The table shows the clustering algorithm and clustering feature combination for each method. In addition to the four summarized methods, we also include local BSS using $\boldsymbol{\psi}_n(t)$ with the oracle selection of the best results among all nodes. This 5th process is hereafter referred to as the "best local" method.

#### *4.1.2. Simulated acoustic environments*

To perform thorough evaluations of the aforementioned methods in a DMA scenario, we simulated, using the image method [16], the two different anechoic DMA environments depicted in Fig. 1, which are referred to as co-located and distributed scenarios, respectively. In each environment, we have three speakers (shown by "×" in the figure), and three nodes, each of which is equipped with two microphone elements (shown by "○"). For each microphone setup condition, we artificially introduced drift errors for the 2nd and the 3rd nodes, which varied from 0 (i.e., no drift error) to 32 sample/sec. More specifically, for each condition in Fig. 1, we prepared the following six DMA environments with different drift errors: A. 0-0-0 (no drift errors for all three nodes), B. 0-1-2 (no errors for the 1st node, +1 sample/sec drift error for the 2nd node compared with the 1st node, and +2 sample/sec drift error for the 3rd node compared with the 1st node), C. 0-2-4, D. 0-4-8, E. 0-8-16, F. 0-16-32. Note that the basic sampling frequency common to all the nodes was assumed to be 16 kHz. The amount of drift error introduced here was determined according to our preliminary measurements made using several commercially available recording devices. For example, we observed a drift error of about 29 sample/sec between two recording devices produced

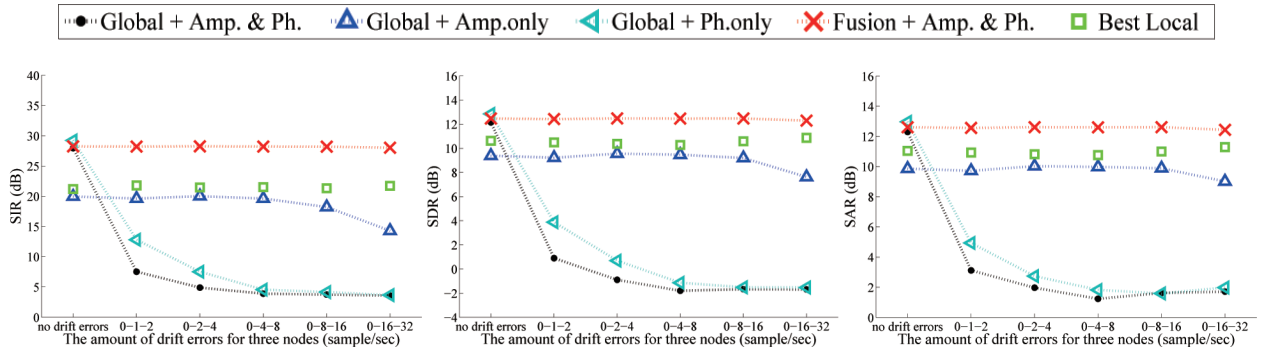



Figure 2: The performances of all the investigated methods in terms of SIR (left), SDR (middle) and SAR (right) in co-located nodes scenario. Input SIR and SDR are -3.1 dB.

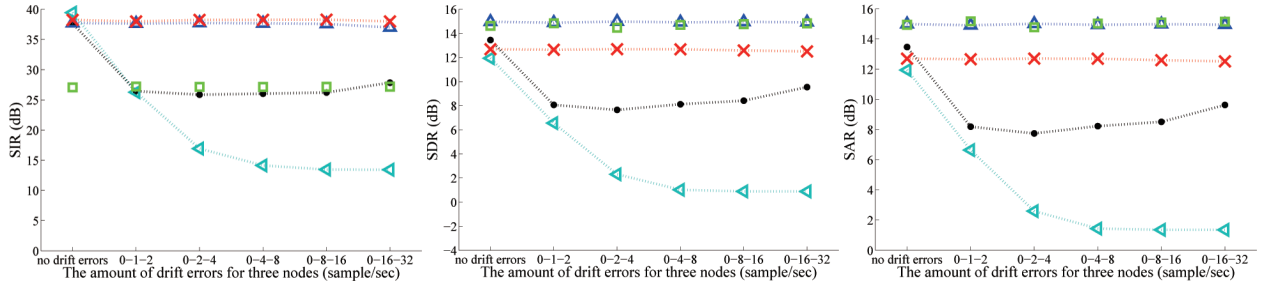



Figure 3: The performances of all the investigated methods in terms of SIR (left), SDR (middle) and SAR (right) in distributed nodes scenario. Input SIR and SDR are -5.9 dB.

by different companies, while in one case it was less than 1 sample/sec for two recording devices of the same model produced by the same company.

### 4.2. Results and discussion

The performance of all the investigated methods was evaluated using the method proposed in [17] where the BSS metrics are the output signal-to-interference ratio (SIR), the signal-to-artifacts ratio (SAR), and the signal-to-distortion ratio (SDR). For the evaluation, we used 20 random combinations of speech utterances of different speakers taken from the TIMIT database [18], and obtained the results by averaging over all combinations.

Figure 2 shows the performance of the investigated methods in co-located DMA scenarios, where the distance between a speaker and the nearest node was 1 m. First, it is clearly observed that the methods that employ phase information, namely "Global + amp. & phase" and "Global + phase", rapidly degrade in terms of performance as the amount of drift error increases, while the other methods are able to maintain their performance robustly. Among these robust methods, the fusion method achieved the best score under almost all conditions. We believe that the superiority of the fusion method over "Global + amp." and "Best local" can be explained by the amount of information they can use for clustering. While the fusion method uses amplitude and phase information within a local node and source activity information (i.e., amplitude information) for inter-node interaction, "Global + amp." and "Best local" respectively lack the intra- and inter-node phase information and the inter-node amplitude and phase information.

Figure 3 shows the performance of the investigated methods in distributed DMA scenarios, where the distance between a speaker and a corresponding node is only 0.2 m. As in Fig. 2, we can observe the same general tendency, namely that the method employing phase information seems to be very sensitive to drift error, while the other methods work very robustly. In terms of SIR, the fusion method and "Global + amp." achieve the best performance, while in the other metrics, the performance of the fusion method is slightly poorer than that of the "Global + amp." and "Best local" methods. This is simply due to the fact that the distance between the speaker and the nearest node is very small in this case. On average, the results suggest that the fusion method works well under various DMA conditions, and is robust against the drift error that is inevitable in many DMA scenarios.

## 5. Conclusion

In this paper, we investigates the behavior of some clustering based BSS algorithms in DMA scenarios with drift errors. The investigated methods included our proposed distributed-fusion method, which employs a distributed EM algorithm, and conventional centralized BSS, which uses phase and/or amplitude features for clustering. We found experimentally that the proposed distributed-fusion method works efficiently in many DMA environments, and is robust against drift errors. This robustness against drift errors can be explained by the fact that our proposed method utilizes all the information available (i.e., amplitude and phase information) only within a local node, and uses, for the inter-node interaction, only source activity information, which usually remains unchanged in the presence of drift errors. We also observed that the conventional centralized clustering algorithm with an amplitude feature works in a DMA scenario where the distance between the speaker and the nearest node is very small, and it is robust against drift errors.

## 6. References

[1] S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Speech Audio Process.*,

vol. 27(2), pp. 113–120, 1979.

[2] R. Martin, "Noise power spectral density estimation based on optimal smoothing and minimum statistics," in *Proc. Interspeech*, 2005, pp. 3145–3148.

[3] T. Kristjansson, J. Hershey, P. Olsen, and R. Gopinath, "Super-human multi-talker speech recognition: The ibm 2006 speech separation challenge system," in *Proc. Int'l Conf. Speech and Language Process. (ICSLP)*, 2006, pp. 97–100.

[4] J. L. Flanagan, "Computer-steered microphone arrays for sound transduction in large rooms," *J. Acoust. Soc. Am.*, vol. 78(11), pp. 1508–1518, 1985.

[5] T.-W. Lee, M. S. Lewicki, M. Girolami, and T. J. Sejnowski, "Blind source separation of more sources than mixtures using overcomplete representations," *IEEE Signal Processing Letters*, vol. 6, no. 4, pp. 87–90, 1999.

[6] O. Yilmaz and S. Rickard, "Blind separation of speech mixture via time-frequency masking," *IEEE Trans. Signal Processing*, vol. 52, no. 7, pp. 1830–1847, 2004.

[7] H. Sawada, S. Araki, and S. Makino, "Underdetermined convolutive blind source separation via frequency bin-wise clustering and permutation alignment," *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 19, pp. 516–527, March 2011.

[8] A. Bertrand, "Applications and trends in wireless acoustic sensor networks: a signal processing perspective," in *Proc. IEEE Symposium on Communications and Vehicular Technology (SCVT)*, 2011, pp. 1–6.

[9] S. W. R. Lienhart, I. Kozintsev and M. Yeung, "On the importance of exact synchronization for distributed audio signal processing," in *Proc. IEEE Int'l Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2003, pp. 14–17.

[10] Z. Liu, "Sound source separation with distributed microphone arrays in the presence of clock synchronization errors," in *Proc. Int'l Workshop on Acoust. Echo and Noise Control (IWAENC).*, 2003, pp. 14–17.

[11] F. Nesta and M. Omologo, "Cooperative wiener-ica for source localization and separation by distributed microphone arrays," pp. 181–184, 2010.

[12] T. B. S. Doclo, M. Moonen and J. Wouters, "Reduced bandwidth and distributed mwf-based noise reduction algorithms for binaural hearing aids," *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 17, pp. 38–51, 2009.

[13] M. Souden, K. Kinoshita, M. Delcroix, and T. Nakatani, "Distributed microphone array processing for speech source separation with classifiar fusion," in *Proc. of IEEE Int'l Workshop on Machine Learning for Signal Processing*, September 2012.

[14] P. D. O'Grady and B. A. Pearlmutter, "Soft-lost: Em on a mixture of oriented lines," in *Proc. of Int'l. Conf. on Independent Component Analysis (LNCS 3195)*, 2004, pp. 430–436.

[15] ——, "The lost algorithm: Finding lines and separating speech mixtures," *EURASIP J. Adv. Signal Process.*, 2008, article ID 784296, 17 pages.

[16] J. B. Allen and D. A. Berkeley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Am.*, vol. 65(4), pp. 943–950, 1979.

[17] R. G. E. Vincent and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 14, pp. 1462–1469, 2006.

[18] G. R. D. W.M. Fisher and K. M. Goudie-Marshall, "The darpa speech recognition research database: specifications and status," in *Proc. DARPA Workshop on Speech Recognition*, 1986, pp. 93–99.