Underdetermined convolutive blind source separation via time-frequency masking

Reju, Vaninirappuputhenpurayil Gopalan; Koh, Soo Ngee; Soon, Ing Yann

2009

Reju, V. G., Koh, S. N., & Soon, I. Y. (2010). Underdetermined Convolutive Blind Source Separation via Time-Frequency Masking. IEEE Transactions on Audio, Speech, and Language Processing, 18(1), 101-116.

https://hdl.handle.net/10356/79797

https://doi.org/10.1109/TASL.2009.2024380

Underdetermined Convolutive Blind Source Separation via Time-Frequency Masking

V. G. Reju*, Soo Ngee Koh, *Senior Member, IEEE,* and Ing Yann Soon

***Abstract*— In this paper we consider the problem of separation of unknown number of sources from their underdetermined convolutive mixtures via time-frequency (TF) masking. We propose two algorithms, one for the estimation of the masks which are to be applied to the mixture in the TF domain for the separation of signals in the frequency domain, and the other for solving the permutation problem. The algorithm for mask estimation is based on the concept of angles in complex vector space. Unlike the previously reported methods, the algorithm does not require any estimation of the mixing matrix or the source positions for mask estimation. The algorithm clusters the mixture samples in the TF domain based on the Hermitian angle between the sample vector and a reference vector using the well known k-means or fuzzy c-means clustering algorithms. The membership functions so obtained from the clustering algorithms are directly used as the masks. The algorithm for solving the permutation problem clusters the estimated masks by using k-means clustering of small groups of nearby masks with overlap. The effectiveness of the algorithm in separating the sources, including collinear sources, from their underdetermined convolutive mixtures obtained in a real room environment, is demonstrated.**

***Index Terms*— Keywords: Blind source separation, Sparse component analysis, Time-Frequency Masking.**

I. Introduction

THE separation of signals from their mixtures without any information about the sources or the mixing process is called blind source separation (BSS). Independent component analysis is one of the techniques commonly used for the separation of sources from their mixtures. Many algorithms have been proposed for both instantaneous and convolutive BSS. In the case where the number of sources is less than or equal to the number of mixtures, methods based on independent component analysis (ICA) [1], [2], [3], [4] are the most popularly used. However, in practical situations the number of sources may be more than the number of mixtures and cases like this are called underdetermined BSS. When mixing is underdetermined, sparse component analysis (SCA) is shown to outperform ICA [5]. In SCA, sparsity of the signals is utilized to separate the signals from their mixtures. A signal is said to be sparse if the signal amplitude is zero during most of the time period. However, in practice natural signals such as speech are not very sparse in the time domain. In [6], Bofill, et al. show that signals like speech are more sparse in the frequency domain than in the time domain and hence if we transform the time domain signal into the frequency domain, the sparsity can be utilized to separate the signals from their mixtures. By utilizing sparsity in the TF domain, many algorithms have been proposed for blind source separation of underdetermined instantaneous mixtures. The problem of underdetermined convolutive blind source separation has also been addressed by many researchers [7], [8], [5], [9]. Convolutive mixing of the signals can be mathematically expressed as

$$x_p(n) = \sum_{q=1}^{Q} \sum_{l=0}^{L-1} h_{pq}\left(l\right) s_q\left(n-l\right) \tag{1}$$

where $p = 1, \cdots, P$, $q = 1, \cdots, Q$, P is the number of mixtures, Q is the number of sources, N is the total number of samples, L is the length of the mixing filters, $\mathbf{x} = [x_1, x_2, \cdots, x_P]^T$ are the P sensor outputs, T is the transpose operator, $x_p = [x_p(0), \cdots, x_p(N-1)]^T$ are the the mixture samples at the p^{th} sensor output, $\mathbf{s} = [s_1, s_2, \cdots, s_Q]^T$ are the sources, $s_q = [s_q(0), \cdots, s_q(N-1)]^T$ are the samples of the q^{th} source and $h_{pq}(l)$, $l = 0, \cdots, L-1$ is the impulse response from the q^{th} source position to the p^{th} sensor. Using the convolution-multiplication property, the mixing process can be expressed in the TF domain as

$$\mathbf{X}(k,t) = \boldsymbol{\mathcal{H}}\left(k\right) \mathbf{S}(k,t) = \sum_{q=1}^{Q} \mathbf{H}_q(k) S_q(k,t) \tag{2}$$

where $\mathbf{X}(k,t) = [X_1(k,t), \cdots, X_P(k,t)]^T$ is a column vector of the short time Fourier transform (STFT) [10] coefficients of the P mixed signals in the k^{th} frequency bin at time frame t, $\mathbf{S}(k,t) = [S_1(k,t), \cdots, S_Q(k,t)]^T$ is the column vector of the STFT coefficients of the Q source signals, $\mathbf{H}_q(k) = [H_{1q}(k), \cdots, H_{Pq}(k)]^T$ is the q^{th} column vector of the mixing matrix at the k^{th} frequency bin, i.e., $\boldsymbol{\mathcal{H}}\left(k\right) = [\mathbf{H}_1\left(k\right), \cdots, \mathbf{H}_Q\left(k\right)]$ is the mixing matrix at the k^{th} frequency bin and $H_{pq}(k)$ is the k^{th} DFT coefficient of the impulse response (or mixing filter) from the q^{th} source to the p^{th} sensor. Here, it is assumed that the impulse responses remain the same for all t. In this paper all the signals in the time domain are represented by small letters whereas signals in the frequency domain are represented by capital letters.

For underdetermined BSS of speech signals, the most widely used assumption is its disjoint orthogonality property in the TF domain [11]. Two speech signals s_1 and s_2 with supports Ω_1 and Ω_2 in the TF plane are said to be TF-disjoint if $\Omega_1 \cap \Omega_2 = \emptyset$. However, in practice the signals may not be perfectly disjoint. In [11] it is shown that for practical purposes an approximate disjoint orthogonality is sufficient for the separation of speech signals from their mixtures.

The authors are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore-639798. (E-mail: reju@ntu.edu.sg; esnkoh@ntu.edu.sg; eiysoon@ntu.edu.sg Phone: +65-96261479; +65-67905429; +65-67905638. Fax:+65-67927660)

The disjoint orthogonality property of the speech signals has been successfully utilized for the generation of binary masks which can be applied to the mixtures in the TF domain for the separation of the sources from their underdetermined convolutive mixtures [7], [8], [9], [5]. The techniques used for the estimation of masks in some of the recent papers are reviewed below.

The direction of arrival (DOA) information is utilized in [8] for the estimation of the binary masks. For the case of three sources and two mixtures demonstrated in [8], the DOA $\theta_{DOA}(k,t)$ is estimated at each time-frequency point. A histogram is then plotted using the DOAs, $\theta_{DOA}(k,t)$, $\forall t$, and the three peaks obtained from the histogram is taken as the DOAs of the three sources at that frequency. The q^{th} signal is then extracted using the binary mask

$$M_q(k,t) = \begin{cases} 1 & \theta_{DOA_q} - \Delta \leq \theta_{DOA}(k,t) \leq \theta_{DOA_q} + \Delta \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

i.e., $Y_q(k,t) = M_q(k,t)X_p(k,t)$, where $q = 1, 2, 3$; $p = 1$ or 2 and Δ is the extraction range parameter.

In [7], a two stage algorithm for the extraction of the dominant sources from their mixtures is proposed. The main assumption is that the total number of dominant sources is less than the number of microphones, but the number of dominant sources plus the interfering sources can be greater than the number of microphones. Thus, in the first stage, the frequency domain ICA algorithm is applied to the output of the microphones under the assumption that the number of independent components are equal to the number of microphones and in the second stage, time-frequency masking is used to improve the performance as the components separated by the ICA algorithm will contain some residuals caused by the interfering sources, when the total number of sources is more than the number of microphones. After solving the permutation problem and estimating the number of sources in the first stage, binary masks are obtained based on the angles between the mixture sample vectors $\mathbf{X}(k,t)$ and the Fourier transform of the estimated mixing filters $\hat{\boldsymbol{\mathcal{H}}}(k)$.

For the estimation of the binary masks, in [5], the impulse responses of the channels (i.e., the mixing filters) are estimated first. For the estimation of the mixing filters, it is assumed that the sources are sparse in the time domain so that the time interval during which only one of the sources is effectively present is estimated; then, for each estimated time interval the cross-correlation technique [12], [13] for the blind single input multiple output (SIMO) channel identification is applied. Since the single source intervals for the same source can exist at many different time slots, after estimation of the mixing filters, they are clustered into Q clusters using the k-means clustering algorithm. The centroids of the clusters are then taken as the estimated channel parameters. Under the assumption that the sources in their TF domain are disjoint, the spatial direction vectors, $\mathbf{v}(k,t) = \frac{\mathbf{X}(k,t)}{||\mathbf{X}(k,t)||}$, of the mixture at each point in the k^{th} frequency bin (after forcing the first entry of the spatial vector to real and positive) are clustered into Q clusters by minimizing the criterion

$$\mathbf{v}(k,t) \in \mathscr{C}_i \Leftrightarrow i = \arg\min_q \left\| \mathbf{v}(k,t) - \frac{\hat{\mathbf{H}}_q(k)e^{-j\angle H_{q1}(k)}}{\left\|\hat{\mathbf{H}}_q(k)\right\|} \right\| \quad (4)$$

where $\mathscr{C}_i$ is the i^{th} cluster and $\hat{\mathbf{H}}_q(k)$ is the Fourier transform of the q^{th} channel vector estimate. The samples in each cluster are then taken as the samples corresponding to one source.

The main shortcoming with the algorithm proposed in [8] is that it requires the DOA of the sources. The accurate estimation of DOA is very difficult in a reverberant environment and when the sources are very close or collinear with the microphone array. For the algorithms in both [7] and [5], the approximate mixing parameters are to be estimated first. In [7], this is done using the ICA algorithm and hence it cannot be used when the number of dominant (or the required) sources are more than the number of microphones. The channel estimation algorithm in [5] utilizes the assumption that the sources are sparse enough in the time domain for effective channel estimation.

In this paper, utilizing the concept of angles in complex vector space [14], we propose a simple algorithm for the design of the separation masks which are used to separate the sources from their underdetermined convolutive mixtures under the assumption that the sources are sufficiently disjoint in the TF domain. Same as in the TF masking approach, the proposed algorithm does not have the well known scaling problem. In addition to that, the algorithm does not require any geometrical information about the sources or microphones. Another advantage is that well known clustering algorithms can be directly used and the membership function obtained from the clustering algorithms can be used as the mask. Also, the additional computational complexity in estimating the masks due to the increase in the number of microphones is very low. In addition to the TF masking method for the separation of the signals, we also propose an algorithm to solve the well known permutation problem. The algorithm is based on k-means clustering, where the estimated masks will be clustered to solve the permutation problem. Since the already available masks are used to solve the permutation problem, instead of using magnitude envelopes or power ratios of the separated signals, some computation time can be saved. A similar approach for solving the permutation problem is previously reported in [15], see Section.II-D for a brief discussion on the difference between the proposed algorithm and that in [15]. Unlike the conventional DOA based algorithms [16], [17], [18], the proposed algorithms for solving the permutation problem does not require any geometrical information of the source positions and hence it can be used even when the sources are very close or collinear. We will also show that the proposed algorithm is suitable for separation of collinear sources in a real room environment.

This paper is organized as follows. In the next section the proposed algorithms for estimation of the masks and automatic detection of the number of sources, followed by the algorithm for solving the permutation problem, are described. The experimental results are given in Section III. Finally

Section IV concludes the paper.

II. PROPOSED METHOD

A. Basic idea

Let us first consider the case of instantaneous mixing. For instantaneous mixing, the impulse responses will be simple pulses of amplitude h_{pq}, where h_{pq} is the $(p,q)^{\mathrm{th}}$ element of the mixing matrix. If the impulse response is a simple pulse, the imaginary part of $H_{pq}(k)$ will be zero and the real part will be the same as h_{pq}, i.e., $I\{H_{pq}(k)\} = 0$ and $R\{H_{pq}(k)\} = h_{pq}, \forall k$. Hence $\mathbf{H}_q(k) = \mathbf{h}_q = [h_{1q}, \cdots, h_{Pq}]^T$, $\forall k$, where $\mathbf{h}_q$ is the q^{th} column of the mixing matrix in the time domain and $\mathbf{H}_q(k)$ is the q^{th} column of the mixing matrix in the frequency domain at the k^{th} frequency bin. For ease of explanation assume that $P = Q = 2$. Now consider a point (k_1, t_1) in the TF plane where only the components of source s_1 is present. Then from (2)

$$\mathbf{X}(k_1,t_1) = \mathbf{H}_1(k_1)S_1(k_1,t_1) \tag{5}$$

This can be written as:

$$R\{\mathbf{X}(k_1,t_1)\} + jI\{\mathbf{X}(k_1,t_1)\} = \mathbf{H}_1(k_1)\left(R\{S_1(k_1,t_1)\} + jI\{S_1(k_1,t_1)\}\right) \tag{6}$$

Since $R\{S_1(t_1,k_1)\}$ and $I\{S_1(t_1,k_1)\}$ are real, comparing real and imaginary parts of (6), it can be seen that the direction of the column vectors $R\{\mathbf{X}(k_1,t_1)\}$ and $I\{\mathbf{X}(k_1,t_1)\}$ are the same and it is also the same as that of $\mathbf{H}_1(k_1)$, which is the same as that of the first column vector of the mixing matrix $\mathbf{h}_1$. Similarly, at another instant (k_2,t_2), if only source s_2 is present, then

$$R\{\mathbf{X}(k_2,t_2)\} + jI\{\mathbf{X}(k_2,t_2)\} = \mathbf{H}_2(k_2)\left(R\{S_2(k_2,t_2)\} + jI\{S_2(k_2,t_2)\}\right) \tag{7}$$

Here the directions of both $R\{\mathbf{X}(k_2,t_2)\}$ and $I\{\mathbf{X}(k_2,t_2)\}$ are the same as that of $\mathbf{H}_2(k_2)$, which is the same as that of the second column vector of the mixing matrix $\mathbf{h}_2$. Hence if the sources are sparse in the TF domain, the scatter plot of both $R\{\mathbf{X}(k,t)\}$ and $I\{\mathbf{X}(k,t)\}$ will show a clear orientation towards the directions of the column vectors of the mixing matrix and once we know the directions, we can determine the mixing matrix and hence the sources can be estimated up to a scaling factor with permutation.

When the mixing is convolutive, the column vectors $\mathbf{H}_q(k)$ in (2) will be a complex column vector and multiplication of this complex vector by a complex scalar, $S_q(k,t)$, will change the complex-valued angle of the vectors. Hence the above approach, used for instantaneous mixing, cannot be directly applied for convolutive mixing. Now consider two complex vectors $\mathbf{u}_1$ and $\mathbf{u}_2$. The cosine of the complex-valued angle between $\mathbf{u}_1$ and $\mathbf{u}_2$ is defined as [14]

$$\cos(\theta_C) = \frac{\mathbf{u}_1^H \mathbf{u}_2}{||\mathbf{u}_1||\,||\mathbf{u}_2||} \tag{8}$$

where $||\mathbf{u}|| = \sqrt{\mathbf{u}^H\mathbf{u}}$ and H represents the complex conjugate transpose operation. $\cos(\theta_C)$ in (8) can be expressed as

$$\cos(\theta_C) = \rho e^{j\varphi} \tag{9}$$

where $\rho \le 1$ [14]

$$\rho = \cos(\theta_H) = |\cos(\theta_C)| \tag{10}$$

In addition, $0 \le \theta_H \le \pi/2$ and $-\pi \le \varphi \le \pi$ are called the Hermitian and pseudo angle respectively between the vectors $\mathbf{u}_1$ and $\mathbf{u}_2$ [14]. The Hermitian angle between the complex vectors $\mathbf{u}_1$ and $\mathbf{u}_2$ will remain the same even if we multiply the vectors by any complex scalars, whereas φ will change (see appendix for proof). This fact can be used for the design of masks for the BSS of underdetermined convolutive mixtures as follows. Since multiplication of a complex vector by a complex scalar is not affecting the Hermitian angle between the vector and another vector (reference vector), we take a P element vector **r**, with all the elements equal to $1+j1$ as the reference vector . The Hermitian angle between the reference vector **r** and $\mathbf{H}_q(k)$ will remain the same even if we multiply $\mathbf{H}_q(k)$ by any complex scalar $S_q(k,t)$. If the signals s_q, $q = 1, \cdots, Q$ are sparse in the TF domain, at any point in the TF plane only one of the source components will be present and the Hermitian angle between the reference vector and the mixture vectors $\mathbf{X}(k,t)$ at that point will be the same as that between $\mathbf{H}_q(k)$ corresponding to the source component $S_q(k,t)$ present at that point and the reference vector **r**. Hence the mixture samples in each frequency bin, k, will form Q clusters with a clear orientation with respect to the reference vector **r**; all the samples in one cluster will belong to the same source. It is not necessary to make all the elements of the reference vector equal to $1+j1$. In fact, any random vector can be taken. The only difference is that, for different reference vectors, the Hermitian angles between the reference vectors and $H_q(k)$, $q = 1, \cdots, Q$ will be different whereas those between the column vectors $H_q(k)$, $q = 1, \cdots, Q$ will remain the same, for a particular frequency bin. By finding the clusters, we are finding the samples which belong to the sources corresponding to those particular clusters. In the following section this idea is illustrated with two sources and two sensors, i.e., $P = Q = 2$.

Assume that at point (k_1,t_1) only the contribution of source s_1 is present, i.e., $S_1(k_1,t_1) \ne 0$ and $S_2(k_1,t_1) = 0$. Let the reference vector be $\mathbf{r} = [1+j1, 1+j1]^T$. At pont (k_1,t_1) the Hermitian angle $\Theta_H^{(k_1)}(t_1)$ between the reference vector **r** and the mixture vector $\mathbf{X}(k_1,t_1) = [X_1(k_1,t_1), X_2(k_1,t_1)]^T$ will be the same as that between **r** and $\mathbf{H}_1(k_1,t_1) = [H_{11}(k_1), H_{21}(k_1)]^T$. This angle, $\Theta_H^{(k_1)}(t_1)$, will be the same for all the points in the frequency bin k_1, where only the component of the source s_1 is present. Similarly at another point, (k_1,t_2), if $S_1(k_1,t_2) = 0$ and $S_2(k_1,t_2) \ne 0$, the Hermitian angle $\Theta_H^{(k_1)}(t_2)$ between **r** and $\mathbf{X}(k_1,t_2)$ will be the same as that between **r** and $\mathbf{H}_2(k_1) = [H_{12}(k_1), H_{22}(k_1)]^T$ and this will remain the same for all the points in the frequency bin k_1 where only the component of source s_2 is present. Hence if we calculate the Hermitian angle between **r** and $\mathbf{X}(k_1,t)$, $\forall t$, depending on presence or absence of the components of the sources, there will be a clear grouping of the mixture vectors according to the Hermitian angles between the reference vector and the mixture vectors. This is demonstrated in Fig.1(a) where the Hermitian angle between the reference vector **r** and $\mathbf{H}_1(k)$ is 14.96^o and that between **r** and $\mathbf{H}_2(k)$ is 29.40^o for $k = 54$. In practice the signals in the TF domain may not be fully sparse, i.e., there may be instants where both the components of sources s_1 and s_2 are present. However, as demonstrated in [11] for the case of instantaneous mixing, for speech signals, approximate sparsity or disjoint orthogonality is sufficient for the separation of sources from their mixtures via binary masking.

For a general case of P mixtures and Q sources, the Hermitian angle between the reference vector $\mathbf{r}$ having P elements (say each element is $1+j1$), and each of the mixture vectors in the k_1^{th} frequency bin, $\mathbf{X}(k_1, t)$, $\forall t$ is calculated, to obtain a vector of Hermitian angles, $\Theta_H^{(k_1)}$, where the value of $\Theta_H^{(k_1)}$ at t_1 is given by

$$\Theta_H^{(k_1)}(t_1) = \cos^{-1}\left(|\cos(\theta_C(k_1, t_1))|\right) \tag{11}$$

$$\cos(\theta_C(k_1, t_1)) = \frac{\mathbf{X}(k_1, t_1)^H \mathbf{r}}{||\mathbf{X}(k_1, t_1)||\,||\mathbf{r}||} \tag{12}$$

The Hermitian angle vector, $\Theta_H^{(k)}$, calculated for the frequency bin k is used for partitioning the mixture samples in the k^{th} frequency bin. The membership functions for the partitioning of the samples so obtained from the clustering algorithm are used as the mask, $M_q(k, t)$, $\forall t$, which will be multiplied by the mixture in the TF domain, $X_p(k, t)$, $\forall t$, to obtain the separated signal $Y_q(k, t)$, $\forall t$ in the TF domain, i.e.,

$$Y_q(k, t) = M_q(k, t) X_p(k, t), \forall t, q = 1, \cdots, Q \tag{13}$$

where $p \in \{1, \cdots, P\}$ is the index of the microphone output to which the mask is applied.

B. Clustering of mixture samples and mask estimation

The partitioning of the values of $\Theta_H^{(k)}$ and hence the corresponding mixture samples in the TF domain into different groups can be done using the well established data clustering algorithms [19], [20]. In this paper we examine the use of two well known clustering algorithms namely, k-means [20] and fuzzy c-means (FCM) [21] clustering algorithms for the partitioning of samples in $\Theta_H^{(k)}$. The k-means algorithm is a hard partitioning technique, which means that any sample in the data vector to be clustered will be fully assigned to any one of the clusters, i.e., the membership function will be binary (0 or 1). Hence if we use the membership function obtained from the k-means algorithm as the mask, it will be a binary mask. On the other hand the FCM algorithm is a soft partitioning technique and hence the mask generated by FCM will be a smooth one compared to that from the k-means algorithm. In the following section the clustering and the mask estimation procedure using the k-means and fuzzy c-means algorithms are explained in detail.

1) k-means clustering: If the samples in the TF domain is perfectly sparse, the vector of Hermitian angles $\Theta_H^{(k)}$ will contain only Q different values, each corresponding to a particular source and hence we can partition the samples perfectly without any ambiguity. However, in a real situation this may not be the case. Hence we have to use a clustering algorithm to partition the samples into different clusters. The Hermitian angles in degree, calculated for $k = 54$, $P = Q = 2$ is shown in Fig.1(a). From the Figure, it is clear that most of the values are either close to 14.96^o or to 29.40^o, which are the actual directions of the mixing vectors $\mathbf{H}_1(k)$ and $\mathbf{H}_2(k)$ respectively with respect to the reference vector $\mathbf{r}$. Using the k-means algorithm we can partition the samples in $\Theta_H^{(k)}$ into 2 clusters. Since the k-means algorithm is a hard partitioning technique, each sample will belong to either one of the clusters and the membership function obtained will be binary (0 or 1). The direction of the estimated mixing matrix is the centroid of the angles corresponding to that particular cluster. Since we are estimating the signals by masking, our main interest is on the estimated membership function, which will be used as the mask. The membership functions obtained from k-means clustering are purely binary. To make it smoother, the samples away from the mean direction or centroid by $\Delta\phi$ are given the membership value $\cos(\Delta\phi)$. The membership function so obtained are used as the mask, as shown in Fig.1(b), which is multiplied with the mixture samples obtained from one of the microphone outputs in the TF plane. Fig.1(c) is the magnitude envelope of the DFT coefficient of the clean signals picked up by the microphone on which the mask is applied. Fig.1(d) is the magnitude envelope of the estimated signals obtained by applying the mask on the mixture samples in the TF domain.

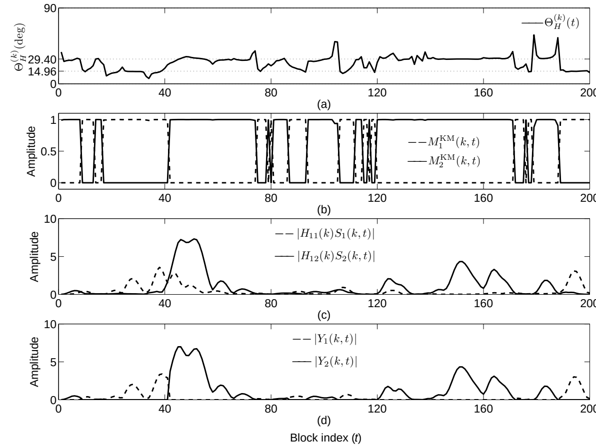


Fig. 1. Masks generated by k-means clustering algorithm for frequency bin $k = 54$.

It is a well known fact that the starting centroid of the k-means clustering algorithm will have an impact on the final centroid of the clusters [22]. Hence in our algorithm, we initialized the k-means algorithm with the result obtained from the histogram method on $\Theta_H^{(k)}$, i.e., the k-means algorithm is initialized with the bin centers of the highest Q bins in the histogram. We start with $max(10, Q)$ bins and if any one of the highest Q bins are empty (this happens when the angle between the column vectors $\mathbf{H}_q(k)$, $q = 1, \cdots, Q$ are very small), the number of bins are doubled to reduce the bin width and the histogram estimation is repeated. This process is repeated until none of the Q bins is empty.

2) Fuzzy c-means clustering: The k-means algorithm described in Section II-B.1 is a hard partitioning method, and as a result of which the estimated signal will contain abrupt

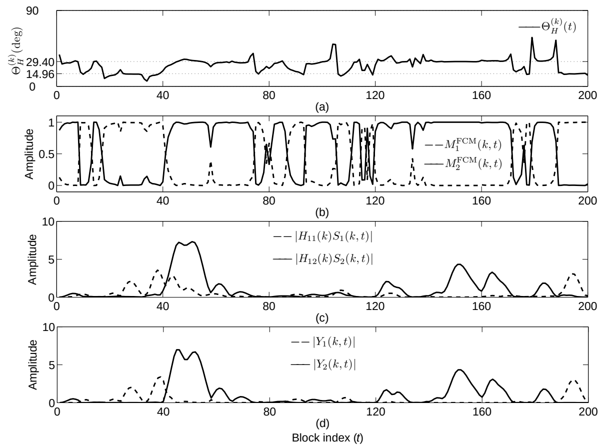


Fig. 2. Masks generated by FCM clustering algorithm for frequency bin $k = 54$.

changes in their amplitude as shown in Fig.1(d). These abrupt changes in the amplitude will introduce artifacts in the reconstructed signals in the time domain. To avoid this problem we examine the use of the FCM clustering algorithm, which partitions the samples into clusters with membership values which are inversely related to the distance of $\Theta_H^{(k)}(t)$ to the centroids of the clusters. For example, if a sample is equidistant from the estimated centroids of the clusters, the k-means clustering algorithm will assign that sample to one of the clusters, with membership value equal to 1 with respect to the cluster into which the sample is assigned and zero for the other clusters, i.e., the membership function will be binary. In the case of the FCM algorithm, for the same condition, the sample will be assigned to all the clusters with equal membership values of $1/Q$, where Q is the number of clusters. The FCM algorithm when applied to the same frequency bin as that used in Section II-B.1 is shown is Fig.2. From the Figure it can be seen that the mask, which is the same as the membership function obtained from the FCM algorithm, is smooth and hence the magnitude envelope of the DFT coefficients of the estimated signals are also smooth. Consequently, it will reduce the artifacts in the reconstructed speech signals in the time domain. However, as shown in Section III-A, the reduction in artifacts is at the cost of reduction is signal to interference ratio (SIR).

C. Automatic detection of the number of sources

In the previous section, we assumed that the total number of sources is known in advance. However, in a practical situation this may not be the case. Hence we need to estimate the number of sources present in the mixture before clustering $\Theta_H^{(k)}$ for the mask estimation, i.e, we need to estimate the number of clusters in $\Theta_H^{(k)}$. Many algorithms are available in the literature for the estimation of the number of clusters [23], [24], [25], [26]. One commonly used technique is the cluster validation technique. In this technique, we must have some knowledge about the possible maximum number of clusters. Then the data is clustered for different number of clusters, c, $c = 2, \cdots, c_{\max}$, where $c_{\max}$ is the possible maximum number of clusters. The clusters so obtained for different values of c are validated using the cluster validation technique [23], [24], [25] and the number of clusters in the best clustering is taken as the actual number of clusters. In this paper we use a recently reported cluster validation technique [24] for the estimation of the number of clusters. Since our data are one dimensional, the validation index proposed in [24] for multidimensional data can be simplified as

$$\text{Validation index} \quad V(U, \Psi, c) = Scat(c) + \frac{Sep(c)}{Sep(c_{\max})} \tag{14}$$

where the different column vectors of $U \in \mathbb{R}^{\mathcal{T}\times c}$ contains the membership values of the data to different clusters, $\Psi = [\psi_1, \cdots, \psi_c]^T$, ψ_i is the centroid of the i^{th} cluster, c is the total number of clusters, $\mathcal{T}$ is the total number of samples in $\Theta_H^{(k)}$. Here $Scat(c)$ represents the compactness of the obtained cluster when the number of clusters is c

$$Scat(c) = \frac{\frac{1}{c}\sum_{i=1}^{c} \sigma_{\psi_i}}{\sigma_{\Theta_H^{(k)}}} \tag{15}$$

$$\sigma_{\Theta_H^{(k)}} = \frac{1}{\mathcal{T}} \sum_{t=1}^{\mathcal{T}} \left(\Theta_H^{(k)}(t) - \bar{\Theta}_H^{(k)}\right)^2 \tag{16}$$

$$\sigma_{\psi_i} = \frac{1}{\mathcal{T}} \sum_{t=1}^{\mathcal{T}} u_{ti} \left(\Theta_H^{(k)}(t) - \psi_i\right)^2 \tag{17}$$

$$\bar{\Theta}_H^{(k)} = \frac{1}{\mathcal{T}} \sum_{t=1}^{\mathcal{T}} \Theta_H^{(k)}(t) \tag{18}$$

The range of $Scat(c)$ is between 0 and 1. For compact clustering $Scat(c)$ will be smaller. The term $Sep(c)$ represents the separation between the clusters, which is given by

$$Sep(c) = \frac{d_{\max}^2}{d_{\min}^2} \sum_{i=1}^{c} \left(\sum_{j=1}^{c} (\psi_i - \psi_j)^2\right)^{-1} \tag{19}$$

$$d_{\min} = \min_{i \neq j} |\psi_i - \psi_j| \tag{20}$$

and

$$d_{\max} = \max_{i \neq j} |\psi_i - \psi_j| \tag{21}$$

The value of $Sep(c)$ will be smaller when the cluster centers are well distributed and larger for irregular cluster centers. Hence the best clustering is the one which minimizes $V(U, \Psi, c)$.

The source contribution from different sources will be different in each frequency bins and in some bins the contribution from some of the sources may be very weak. Hence the number of clusters (or sources) estimated from a single frequency bin will not be reliable. To make the estimation more robust, the cluster validation technique is applied to many frequency bins and the number which is most frequently detected over these frequency bins is taken as the actual number, Q, of sources present.

D. Permutation problem

The main weaknesses with frequency domain blind source separation are the scaling and the permutation problems. Since we are applying the masks directly to the mixture in the TF domain without any other stage in front of it, the well known scaling problem is avoided. In general, this is true for all TF making approaches. We therefore only need to solve the permutation problem. In the literature many algorithms have been reported for solving the permutation problems [27], [16], [17], [18], [28], [29], [30], [31]. The DOA based algorithms [16], [17], [18], [28] are not effective in highly reverberant environments or when the sources are collinear or very close to one another [30]. In [27] it is shown that for speech signals, the magnitude envelopes of the adjacent frequency bin in the TF domain are highly correlated and this property can be used to solve the permutation problem. Later in [29] it is shown that the correlation between the power ratios are more suitable than that between the magnitude envelopes. This fact is further verified in Fig.3, where in Fig.3(a), the correlation matrix whose entries are the correlations between the bin wise magnitude envelopes of the STFT coefficients of the two clean signals $\hat{s}_1$ and $\hat{s}_2$ picked up by the microphones are shown. In the Figure, the magnitudes of the entries in the correlation matrix are shown by gray levels. The above correlation matrix $\mathbf{C}^{\text{mag}}_{\hat{S}_1\hat{S}_2} \in \mathbb{R}^{2K' \times 2K'}$ is calculated as:

$$\mathbf{C}^{\text{mag}}_{\hat{S}_1\hat{S}_2} = \begin{bmatrix} \mathbf{R}_{\tilde{S}_1\tilde{S}_1} & \mathbf{R}_{\tilde{S}_1\tilde{S}_2} \\ \mathbf{R}_{\tilde{S}_2\tilde{S}_1} & \mathbf{R}_{\tilde{S}_2\tilde{S}_2} \end{bmatrix} \tag{22}$$

where $\mathbf{R}_{\tilde{S}_i\tilde{S}_j} \in \mathbb{R}^{K' \times K'}$, $i,j \in \{1,2\}$ is the correlation matrix whose $(m,n)^{\text{th}}$ element, $\left(\mathbf{R}_{\tilde{S}_i\tilde{S}_j}\right)_{mn}$, is the Pearson correlation coefficient between m^{th} and n^{th} rows of $\tilde{S}_i \in \mathbb{R}^{K' \times \mathcal{T}}$ and $\tilde{S}_j \in \mathbb{R}^{K' \times \mathcal{T}}$ respectively, $K' = \frac{K}{2} + 1$ if DFT length K is even; otherwise $K' = \frac{K+1}{2}$ and $\mathcal{T}$ is the total number of samples in each frequency bin. Because of the conjugate smmetry property of the DFT coefficients, only the first K' bins are taken. The $(k,t)^{\text{th}}$ element of $\tilde{S}_q$, $q \in \{1,2\}$, is given by

$$\tilde{S}_q(k,t) = \left|\hat{S}_q(k,t)\right| \tag{23}$$

Here, $\hat{S}_q(k,t)$ are the STFT coefficients of $\hat{s}_q = h_{pq} * s_q$, which is the clean signal picked up by the p^{th} microphone to which the mask is applied.

The correlations between the bin wise power ratios of the STFT coefficients of the signals are shown in Fig.3(b). The correlation matrix is defined as:

$$\mathbf{C}^{P^{\text{ratio}}}_{\hat{S}_1\hat{S}_2} = \begin{bmatrix} \mathbf{R}_{P^{\text{ratio}}_{\hat{S}_1}P^{\text{ratio}}_{\hat{S}_1}} & \mathbf{R}_{P^{\text{ratio}}_{\hat{S}_1}P^{\text{ratio}}_{\hat{S}_2}} \\ \mathbf{R}_{P^{\text{ratio}}_{\hat{S}_2}P^{\text{ratio}}_{\hat{S}_1}} & \mathbf{R}_{P^{\text{ratio}}_{\hat{S}_2}P^{\text{ratio}}_{\hat{S}_2}} \end{bmatrix} \tag{24}$$

where $P^{\text{ratio}}_{\hat{S}_q}(k,t) = \frac{\left|\left|\hat{S}_q(k,t)\right|\right|^2}{\left|\left|\hat{S}_1(k,t)\right|\right|^2 + \left|\left|\hat{S}_2(k,t)\right|\right|^2}$, $q = 1,2$, $k = 1,\cdots,K'$, $\forall t$ and the correlation matrix $\mathbf{R}_{P^{\text{ratio}}_{\hat{S}_i}P^{\text{ratio}}_{\hat{S}_j}} \in \mathbb{R}^{K' \times K'}$, $i,j \in \{1,2\}$ is defined in a similar way as that in (22). (The size of all the correlation matrices shown in Fig.3 are the same as that of $\mathbf{C}^{\text{mag}}_{\hat{S}_1\hat{S}_2}$). Comparing Fig.3(a) and (b), it can be seen that the correlation between the power ratios is the better choice than that between the magnitude envelops for solving the permutation problem. The reason for the improvement in performance are [29] as follows: 1) The values of power ratios are clearly bounded between 0 and 1. 2) Because of the sparseness of the signals, most of the time, the power ratios will be closer to either 0 or 1. 3) The power ratios of different sources are exclusive to each other, i.e., for a two source case, if $P^{\text{ratio}}_{\hat{S}_1}(k,t)$ is close to 1 then $P^{\text{ratio}}_{\hat{S}_2}(k,t)$ will be close to 0. This shows that the binary mask or the membership functions obtained from the clustering algorithms in Section II-B are the ideal candidates to replace the power ratios in solving the permutation problem as their values are also close to either 1 or 0. This approach has another advantage that we need not calculate the power ratios; instead, the already available masks/membership functions can be used which will save some computation time. The correlations calculated between the power ratios of the STFT coefficients in each frequency bin of the separated signals, $\mathbf{C}^{P^{\text{ratio}}}_{Y_1Y_2}$, and that between the masks, $\mathbf{C}_{M_1M_2}$, are shown respectively in Fig.3(c) and (d). (In cases where it is necessary to specify the algorithm used to estimate the masks, the name of the clustering algorithm will be added as superscript to $\mathbf{C}_{M_1M_2}$ and M_q. For example the correlation matrix and the masks estimated by the k-means algorithm will be represented as $\mathbf{C}^{\text{KM}}_{M_1M_2}$ and M^{KM}_q respectively whereas those by the FCM algorithm will be represented as $\mathbf{C}^{\text{FCM}}_{M_1M_2}$ and M^{FCM}_q respectively). The correlation matrix $\mathbf{C}^{P^{\text{ratio}}}_{Y_1Y_2}$ is defined similarly to $\mathbf{C}^{P^{\text{ratio}}}_{\hat{S}_1\hat{S}_2}$, except that $\hat{S}_1$ and $\hat{S}_2$ are replaced by Y_1 and Y_2 respectively. The correlation matrix $\mathbf{C}_{M_1M_2}$ is calculated as

$$\mathbf{C}_{M_1M_2} = \begin{bmatrix} \mathbf{R}_{M_1M_1} & \mathbf{R}_{M_1M_2} \\ \mathbf{R}_{M_2M_1} & \mathbf{R}_{M_2M_2} \end{bmatrix} \tag{25}$$

where $M_1 \in \mathbb{R}^{K' \times \mathcal{T}}$ and $M_2 \in \mathbb{R}^{K' \times \mathcal{T}}$ are the arrays of first K' masks corresponding to the first and second sources respectively. The correlation matrix $\mathbf{R}_{M_iM_j}$, $i,j \in \{1,2\}$ is defined in a similar way as that in (22). For both Fig.3(c) and (d) the permutation problem is solved based on the correlation between the bin-wise power ratios of the separated signals and that of the clean signals picked up by the microphone on which the masks are applied. From the figures it is clear that both the methods will give almost the same performance. A quantitative comparison is given in Section III-A.

The main disadvantage of the correlation based method in solving the permutation problem is that, as the permutation in one frequency bin is solved based on the permutation of the previous frequency bins, failure in one frequency bin will lead to a complete misalignment beyond that frequency bin. Many algorithms have been proposed to circumvent this problem [32], [31], [30]. Sawada et al. [32] combined the DOA and correlation based approaches to improve the robustness of the algorithm. However, the algorithm cannot be used when the sources are collinear [30]. The partial separation method [31], [30] improved the robustness of the correlation method by incorporating a time domain stage in front of the frequency domain stage. To reduce the computational cost, the time domain stage is normally implemented using

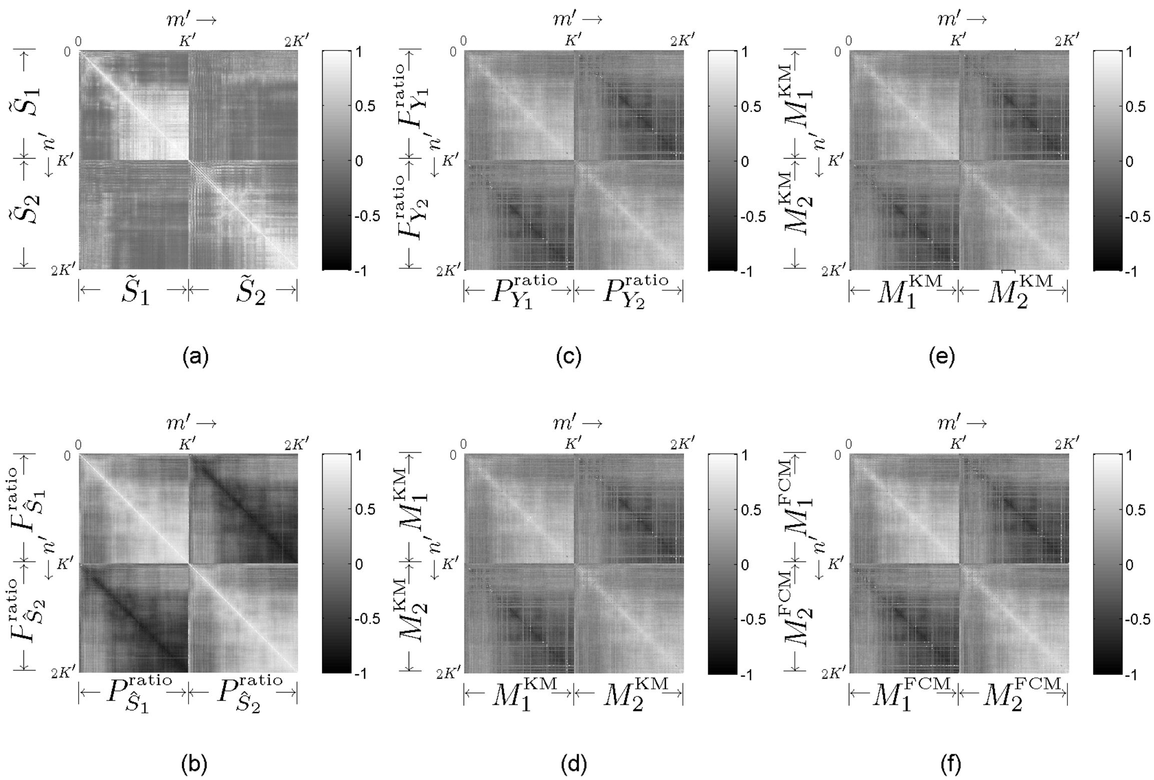


Fig. 3. Correlation matrices (a) $\left(\mathbf{C}^{\mathrm{mag}}_{\tilde{S}_1\tilde{S}_2}\right)_{m'n'}$, correlation between the bin-wise magnitude envelopes of the clean signals picked up by the microphones (b) $\left(\mathbf{C}^{P^{\mathrm{ratio}}}_{\hat{S}_1\hat{S}_2}\right)_{m'n'}$, correlation between the bin-wise power ratios of the clean signals picked up by the microphones (c) $\left(\mathbf{C}^{P^{\mathrm{ratio}}}_{Y_1Y_2}\right)_{m'n'}$, Correlation between the bin-wise power ratios of the separated signals (d) $\left(\mathbf{C}^{\mathrm{KM}}_{M_1M_2}\right)_{m'n'}$, correlation between the masks estimated using k-means clustering algorithm; in both (c) and (d) the permutation problem is solved based on the correlation between the bin-wise power ratios of the separated signals and that of the clean signals picked up by the microphone on which masks are applied (e) $\left(\mathbf{C}^{\mathrm{KM}}_{M_1M_2}\right)_{m'n'}$, correlation between the masks estimated using k-means clustering (f) $\left(\mathbf{C}^{\mathrm{FCM}}_{M_1M_2}\right)_{m'n'}$, correlation between the masks estimated using fuzzy c-means clustering; in both (e) and (f) the permutation problem is solved by the proposed algorithm based on k-means clustering.

Freq. bin	No. of masks assigned by k-means algorithm to different clusters $\mathscr{C}_q$, $q = 1, 2, \cdots, 6$					
	$\mathscr{C}_1$	$\mathscr{C}_2$	$\mathscr{C}_3$	$\mathscr{C}_4$	$\mathscr{C}_5$	$\mathscr{C}_6$
k	1	2	1	1	1	0
$k+1$	1	1	0	1	3	0
$k+2$	1	0	0	1	3	1
$k+3$	0	1	1	1	2	1
$k+4$	1	1	1	1	1	1
$k+5$	0	4	1	0	0	1
$k+6$	0	2	2	0	1	1
$k+7$	1	1	1	1	1	1
$k+8$	1	0	1	1	2	1
$k+9$	1	1	1	1	1	1
$k+10$	1	1	2	1	0	1
$k+11$	1	1	1	1	1	1
$k+12$	3	1	1	0	0	1
$k+13$	1	2	1	1	1	0
$k+14$	1	1	1	1	1	1
$k+15$	1	1	1	1	1	1

TABLE I

ILLUSTRATION OF MASK ASSIGNMENT TO DIFFERENT CLUSTERS

computationally efficient algorithms [33] with a small number of unmixing filter taps so as to obtain the partially separated signals. The partially separated signal is then input to the frequency domain stage where it is fully separated. Then the permutation problem in each frequency bin is solved based on the bin wise correlation between the magnitude envelops of the DFT coefficients of the fully separated and partially separated signals. Though we can use the partial separation method with an additional time domain stage in front of the masking stage, the separation of the signals using a time domain ICA algorithm will be very poor when the mixtures are underdetermined and hence this approach could not be used. In this paper we propose an algorithm based on k-means clustering to solve the permutation problem, where the masks are clustered into Q clusters, $\mathscr{C}_q$, $q = 1, \cdots, Q$, in such a way that the sum of the distances $D_q, q = 1, \cdots, Q$, is minimum. D_q is the total distance between the masks within the q^{th} cluster to its cluster centroid, i.e.,

$$\text{minimize} \quad \mathcal{D} = \sum_{q=1}^{Q} \sum_{\substack{M_i^{(k)} \in \mathscr{C}_q \\ i=1,\cdots,Q \\ k=k_{\mathrm{st}},\cdots,k_{\mathrm{end}}}} \left(1 - r_{M_i^{(k)} C_q}\right) \qquad (26)$$

where $M_i^{(k)}$ is the i^{th} mask in the k^{th} frequency bin, C_q is the centroid of the q^{th} cluster $\mathscr{C}_q$, $r_{_{M_i^{(k)}C_q}}$ is the Pearson correlation between $M_i^{(k)}$ and the cluster centroid C_q, k_{st} and k_{end} are the indices of the starting and ending frequency bins of the group of adjacent frequency bins used for clustering, i.e., the total number of frequency bins used is $k_{\text{end}} - k_{\text{st}} + 1$. Here we use $1 - r_{_{M_i^{(k)}C_q}}$ as the distance measure so that masks which are highly correlated (smaller distance) will form one cluster. Since there are Q sources, we form Q clusters using the k-means algorithm. In an ideal case, each cluster must contain one and only one mask from each frequency bin after clustering. But in practice this may not be the case, especially when the number of sources are large. Under such situations, we need to identify the bins in each cluster where the permutation could not be solved perfectly. This can be done as follows:

After clustering, if any of the clusters is missing the mask for any frequency bin or containing more than one mask for the same frequency bin, it is assumed that the k-means clustering algorithm fails to solve the permutation problem in that particular frequency bin for those clusters. A typical example for the case of six sources (hence six clusters) is shown in Table I, where the masks from 16 adjacent bins are clustered. The number of masks assigned by the clustering algorithm to different clusters are shown in the table. In the table, entries other than ‘1’ indicates that the algorithm fails to solve the permutation problem for that cluster at that particular frequency bin. For example at the k^{th} frequency bin, the algorithm fails in clusters $\mathscr{C}_2$ and $\mathscr{C}_6$. For frequency bins where the k-means clustering algorithm fails to solve the permutation problem, the correlation between the cluster centroids of the failed clusters and the masks in those clusters are used to solve the permutation problem. This is done by reassigning the masks in the failed clusters in such a way that the sum of the correlations between the centroids of the clusters and the masks is maximum, i.e., the permutation matrix $\mathbf{\Pi}_k$ for the k^{th} frequency bin among the failed clusters is calculated as

$$\mathbf{\Pi}_k = \arg\max_{\mathbf{\Pi}} \sum_{i}^{F} \sum_{j}^{F} (\mathbf{\Pi} \bullet \mathbf{R_{CM}})_{ij} \tag{27}$$

where $\bullet$ represents element wise multiplication between the matrices, F is the number of failed clusters, $\mathbf{\Pi}$ is the permutation matrix with one and only one element, which is 1, in any row or column, $\mathbf{R_{CM}} \in \mathbb{R}^{F\times F}$ is the correlation matrix, $(\mathbf{R_{CM}})_{ij}$ is the Pearson correlation between the i^{th} and j^{th} rows of $\mathbf{C}$ and $\mathbf{M}$ respectively, $\mathbf{C} = [\cdots, C_q^T, \cdots]^T$, $C_q \in \mathbb{R}^{1\times\mathcal{T}}$ is the centroid of the q^{th} cluster, $q \in \{\text{indices of failed clusters}\}$, $\mathbf{M} = [\cdots, M_q^T, \cdots]^T$, $M_q \in \mathbb{R}^{1\times\mathcal{T}}$ are the masks in the failed clusters at the k^{th} frequency bin. Then the matrix of *permutation solved masks* at frequency bin k will be $\mathbf{\Pi}_k\mathbf{M}$.

For example, for the $(k+1)^{\text{th}}$ frequency bin in Table I three masks are assigned to cluster $\mathscr{C}_5$ whereas none is assigned to clusters $\mathscr{C}_3$ and $\mathscr{C}_6$. Hence for the $(k+1)^{\text{th}}$ frequency bin, the permutation problem is to be solved among the clusters $\mathscr{C}_3$, $\mathscr{C}_5$ and $\mathscr{C}_6$ by calculating the correlations between the centroids of the clusters, $\mathbf{C} = [C_3^T, C_5^T, C_6^T]^T$, and the masks assigned to $\mathscr{C}_5$. The masks assigned to clusters $\mathscr{C}_1$, $\mathscr{C}_2$ and $\mathscr{C}_4$ are not altered.

For speech signals in the TF domain, when the frequency bins are far apart, the correlation between them will decrease [29]. To overcome this problem, instead of taking all the masks to form the clusters, we take only a few adjacent frequency bins at a time with overlap (for example 16 bins with 75% overlap in our experiments) and cluster them using the k-means clustering algorithm as explained previously. For k-means algorithm, it is a well known fact that the initialization vector, used as the initial centroids, has an impact on the final clusters obtained [20], [22]. In our case, we use the centroids of the current clusters as the starting centroids (initializing vector) for the next group of masks for clustering. For the starting group of masks (i.e., for bins $k = 1$ to 16 in our experiments) the centroids of the clusters obtained by applying the k-means algorithm on the masks in the frequency range of 500Hz to 1000Hz are used as the initialization vectors. The advantages of taking small groups of adjacent masks, overlapping and initializing with the centroids of the previous clusters are: 1) The correlations between the masks corresponding to the same source will be high if the masks belong to the nearby frequency bins and hence there will be a clear separation between the clusters. 2) The centroids of the current clusters will be close to those of the clusters formed by the next group of masks, if both groups are overlapped. This will decrease the convergence time of the k-means clustering algorithm. 3) When initialized with the centroids of the previous group, because of the overlap, the starting centroids will be close to the actual centroids. Hence the permutation of the present group will be the same as that of the previous group of masks.

A similar approach for solving permutation problem using the masks is reported in [15]. Like our method, [15] also uses the correlation between the masks as the distance measure. The main difference between our method and that in [15] is that, in our method, the well known k-means algorithm is used for clustering the masks. There are many improved versions for the basic k-means clustering algorithm (see [20] and the references therein) and any of these algorithms can be used. Moreover, we used small groups of adjacent frequency bins with overlap and each group is initialized with the cluster centroids of the previous group. As explained above and shown in [22], this kind of initialization will increase the convergence speed and significantly reduce the computation time. However, there may be some frequency bins where the k-means algorithm fails to fully solve the permutation problem. The permutation of these bins could be solved by maximizing the sum of the correlations between the centroids of the failed clusters and the masks in those clusters, using (27).

E. Construction of the output signals

Using the separated signals Y_q obtained by applying the masks to one of the microphone outputs in the TF domain, i.e., $Y_q(k,t) = M_q(k,t)X_p(k,t)$, $q = 1, \cdots, Q$, $p \in \{1, \cdots, P\}$, the separated signals in the time domain is constructed by taking inverse STFT followed by the overlap add method

[10]. The masks can be applied to any one of the microphone outputs. However, the performance will be slightly affected by the microphone position, please read Section.III-D for more explanation.

III. Experimental results

For performance evaluation of the proposed algorithm, both real room and simulated impulse responses are used. In Section III-A the impulse response of a real furnished room is used whereas for the remaining experiments, to have a fine control on the position of the microphones and sources as well as on the acoustic environment, simulated impulse responses are used [34]. In all the experiments in this paper, average performances of 50 combinations of speech utterances, selected randomly from 16 speech utterances are used. For the same number of sources, in all the experiments, the combination of speech utterances used are the same. For experiments in Sections III-D and III-E, the wall reflections up to 29^{th} order is taken and humidity, temperature, absorption of sound due to air, etc., are considered while calculating the impulse responses. The reverberation time, TR_{60}, of the simulated room is 115ms.

During the separation process, the signals may be distorted especially when the sources are overlapped in their TF domain. Hence it is necessary to measure the distortion and the artifacts introduced by the algorithm to assess the quality of separation. The quality of separation of the algorithm are measured using the method proposed in [35], [36], where the separated (estimated) signals are first decomposed into three components as

$$y_q = y_{q_{\mathrm{target}}} + e_{q_{\mathrm{interf}}} + e_{q_{\mathrm{artif}}} \tag{28}$$

where $y_{q_{\mathrm{target}}}$ is the target source with allowed deformation such as filtering or gain, $e_{q_{\mathrm{interf}}}$ accounts for the interference due to unwanted sources and $e_{q_{\mathrm{artif}}}$ corresponds to the artifacts introduced by the separation algorithm. Then the source to distortion ratio (SDR), source to interference ratio (SIR) and source to artifacts ratio in dB are calculated as

$$\mathrm{SDR} = 10\log_{10}\frac{\left|\left|y_{q_{\mathrm{target}}}\right|\right|^2}{\left|\left|e_{q_{\mathrm{interf}}} + e_{q_{\mathrm{artif}}}\right|\right|^2} \tag{29}$$

$$\mathrm{SIR} = 10\log_{10}\frac{\left|\left|y_{q_{\mathrm{target}}}\right|\right|^2}{\left|\left|e_{q_{\mathrm{interf}}}\right|\right|^2} \tag{30}$$

$$\mathrm{SAR} = 10\log_{10}\frac{\left|\left|y_{q_{\mathrm{target}}+e_{q_{\mathrm{interf}}}}\right|\right|^2}{\left|\left|e_{q_{\mathrm{artif}}}\right|\right|^2} \tag{31}$$

In the proposed algorithm, since we are applying the mask to one of the microphone outputs in the TF domain, the target signal is taken as the signal picked up by the microphone to which the mask is applied. Here the target source is $y_{q_{\mathrm{target}}} = h_{pq} * s_q$ where h_{pq} is the impulse response from q^{th} source to p^{th} microphone, if the mask is applied to the p^{th} microphone output. The other experimental conditions are: length of speech utterances are 5 seconds, speech sampling frequency is 16 kHz, DFT frame size K=2048 and the window function used is Hanning window.

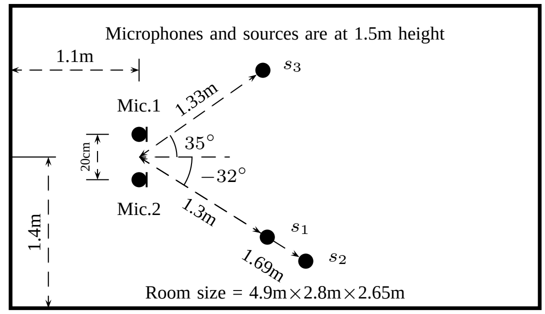


Fig. 4. The source-microphone configuration for the measurement of real room impulse response

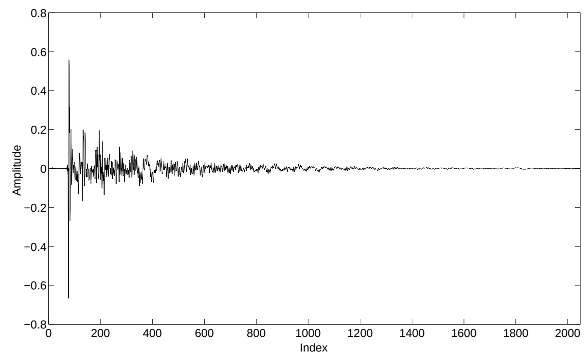


Fig. 5. Measured real room impulse response from source s_3 to first microphone.

A. Experiments using real room impulse responses

In this experiment we use the impulse responses measured in a real furnished room. The reverberation time of the room (TR_{60}) is 187 ms and the impulse response is measured with the help of an acoustic impulse response measuring software 'Sample Champion' [37]. The microphone and loud speaker transfer function are neglected in the measurements. The position of the microphones and sources are shown in Fig.4. One of the impulse responses (from source s_3 to the first microphone) is shown in Fig.5. The sources s_1 and s_2 are collinear. The separation of the sources when they are collinear is a challenging task using independent component analysis. For example, using the computationally efficient implementation [33] of the time domain convolutive BSS algorithm proposed in [38], [3], with an unmixing filter length of 512, the SIR obtained is 10.9dB for noncollinear sources (s_1 and s_3) and only 3.8dB for collinear sources (s_1 and s_2). Here we have taken the unmixing filter length to be equal to 512 because, as discussed in [30], if the filter length is longer, the interdependency of the unmixing filter coefficients will cause the convergence to be poor. On the other hand, an unmixing filter with a shorter filter length will not be able to achieve any significant unmixing effect.

B. Detection of the number of sources

For the detection of the number of sources present in the mixture, the cluster validation technique explained in Section.II-C is applied to $\Theta_H^{(k)}$ for the three different cases

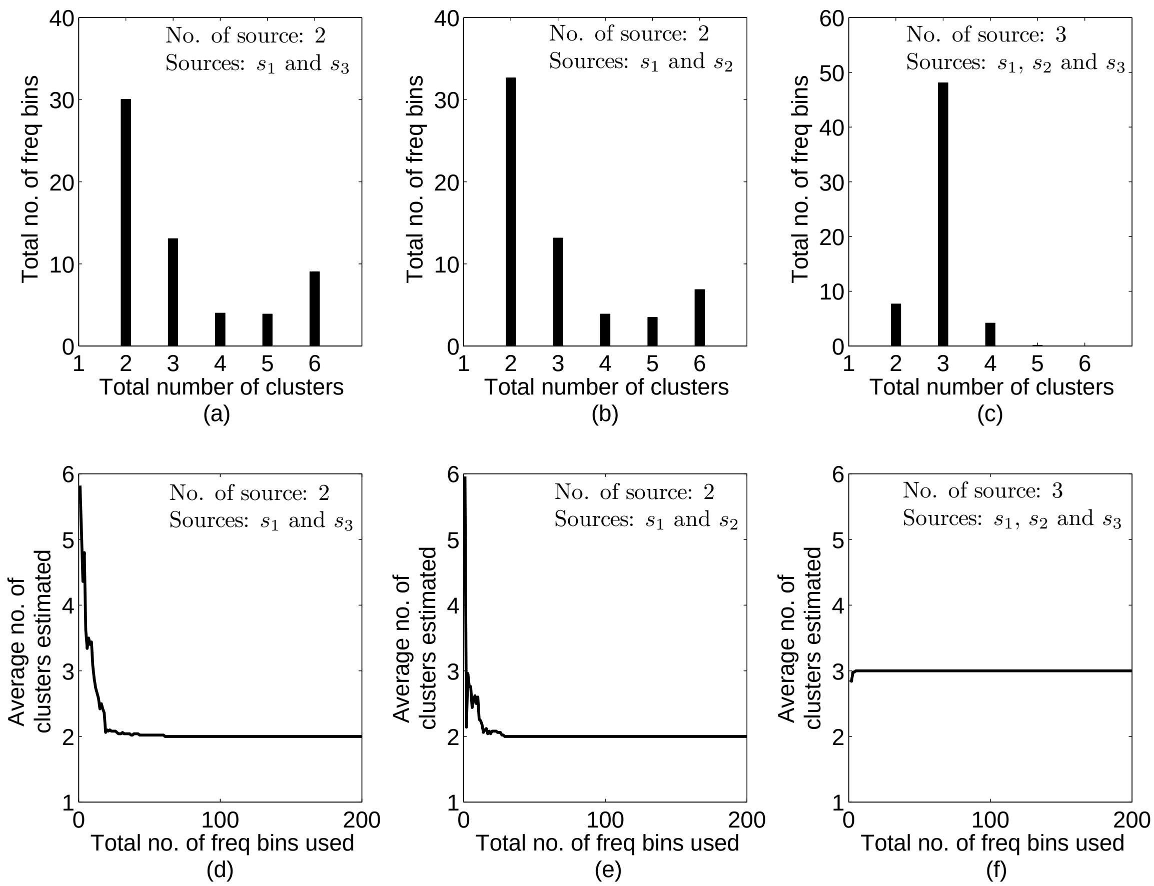


Fig. 6. (a), (b) and (c) Mean histogram of the 'estimated number of clusters (or sources)' for the first 60 frequency bins. (d), (e) and (f) Total number of frequency bins used versus 'estimated number of clusters (or sources) '; the estimation result will be more reliable with higher number of frequency bins used. In the figures, at some points, the 'number of clusters estimated' are not integers because it is the mean performance of 50 sets of speech utterances. All the source positions are with reference to Fig.4.

shown in Fig.4. The first case involves non-collinear sources (s_1 and s_3), the second case collinear sources (s_1 and s_2) and finally the third case all the three sources (s_1, s_2 and s_3). The mean performance obtained for 50 combinations of speech utterances are shown in Fig.6. Fig.6(a), (b) and (c) show the mean histogram of the estimated number of clusters (or sources) over the first 60 frequency bins for three cases of s_1 and s_3, s_1 and s_2 and s_1, s_2 and s_3 respectively. From the Figure it can be seen that the algorithm successfully estimated the number of sources in all the three cases. Fig.6(d), (e) and (f) show the total number of frequency bins used versus the estimated number of sources. The Figures clearly show that it is not necessary to apply the cluster validation technique to all the frequency bins, instead a fraction of the total frequency bins is sufficient for the successful estimation of the number of sources. Since the Hermitian angle calculated at any instant depends on the relative amplitude of the source, the variations in the calculated Hermitian angles will be high during the period where the unvoiced parts of the sources overlap. For example in Figs.1 and 2, during the period $t = 80$ to 120 the magnitude envelop amplitudes of the sources are small and the variation in Hermitial angles are high. In contrast, during the periods where the magnitude envelop amplitudes are high, the variations in Hermitian angles are low. Considering this fact in our experiments, $\Theta_H^{(k)}(t)$ at any point where $\|\mathbf{X}(k,t)\| < 0.1\frac{1}{\mathcal{T}}\sum_{t=1}^{\mathcal{T}}\|\mathbf{X}(k,t)\|$ are removed from $\Theta_H^{(k)}$ before clustering them for the estimation of the number of sources. This will reduce not only the estimation error but also the computation time. It may be noted that the samples with smaller amplitudes are removed only for the estimation of the number of clusters. For mask estimation all the samples are used.

C. Separation performance

The separation performance obtained using the proposed algorithm for the three cases namely collinear, non-collinear and underdetermined with collinear sources are shown in Table II. In the Table, the performances of the algorithm when k-means and fuzzy c-means clustering are used for the design of masks are shown for the cases where the permutation problem is solved by 1) comparing the correlation between power ratios of the separated signals with that of the clean signals picked up by the microphones, and 2) using the proposed k-means clustering approach. Here the correlation between the power

Active sources	Performance measure	Input (dB)	Permutation solved using clean signals				Permutation solved by k-means clustering			
			k-means		FCM		k-means		FCM	
			Output (dB)	Improvement (dB)	Output (dB)	Improvement (dB)	Output (dB)	Improvement (dB)	Output (dB)	Improvement (dB)
s_1 and s_3 (Non-collinear)	SDR	-0.2	6.1	6.4	6.5	6.8	6.5	6.8	6.8	**7.1**
	SIR	0.0	18.2	18.2	16.8	16.8	18.9	**18.9**	17.3	17.3
	SAR	16.1	6.6	-9.5	7.2	-8.9	6.9	-9.1	7.4	**-8.6**
s_1 and s_2 (Collinear)	SDR	-0.3	4.7	5.0	5.1	5.3	5.4	5.7	5.7	**5.9**
	SIR	-0.0	15.6	15.6	14.5	14.5	16.9	**16.9**	15.6	15.6
	SAR	16.2	5.4	-10.8	5.9	-10.3	6.0	-10.2	6.4	**-9.7**
s_1, s_2 and s_3 (Underdetermined with collinear)	SDR	-3.4	1.8	5.2	2.0	**5.4**	0.5	3.9	1.0	4.4
	SIR	-3.2	11.9	**15.1**	10.4	13.6	10.0	13.2	9.1	12.3
	SAR	16.0	2.6	-13.4	3.2	**-12.8**	1.7	-14.3	2.5	-13.5

TABLE II

PERFORMANCE COMPARISON OF THE PROPOSED ALGORITHM USING k-MEANS AND FCM CLUSTERING.

Mask estimation method	No. of sources (Each of 5 sec length)	Time to solve the permutation problem alone (using the proposed algorithm based on K-means clustering) (seconds)	Total time to separate the sources from their mixtures. (seconds)
k-means	2	2.3306	5.7911
	3	3.2150	9.6001
FCM	2	2.2950	5.0497
	3	3.3325	10.4966

TABLE III

ALGORITHM EXECUTION TIME

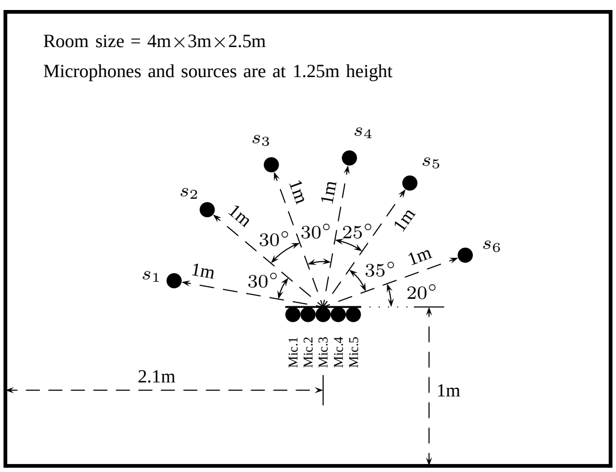
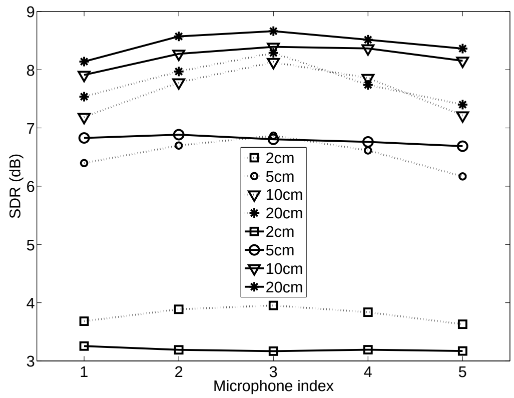


Fig. 7. The source-microphone configuration for the simulated room impulse response

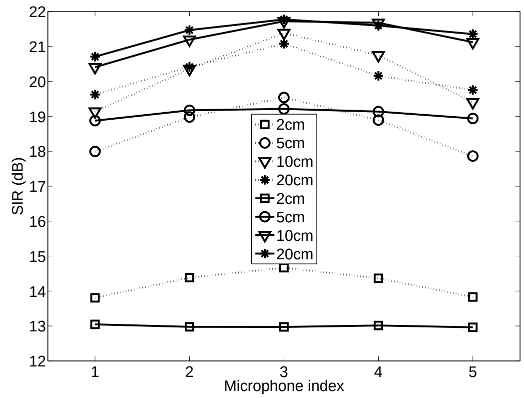
ratios of the clean signals and the separated signals for solving the permutation problem is used as the bench mark to evaluate the proposed k-means clustering algorithm for solving the permutation problem because it is very robust, independent of the quality of separation in each bin and in the ideal case where the separation is perfect, the permutation can be solved perfectly. The permutation matrix estimation procedure can be mathematically expressed as follows:

$$\mathbf{\Pi}_k = \arg\max_{\mathbf{\Pi}} \sum_i^Q \sum_j^Q \left(\mathbf{\Pi} \bullet \mathbf{R}_{\mathbf{P}_{\mathbf{Y}}^{\text{ratio}} \mathbf{P}_{\hat{\mathbf{S}}}^{\text{ratio}}} \right)_{ij} \tag{32}$$

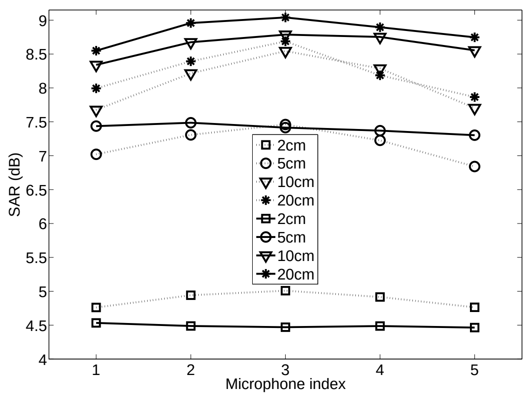
where $\mathbf{R}_{\mathbf{P}_{\mathbf{Y}}^{\text{ratio}} \mathbf{P}_{\hat{\mathbf{S}}}^{\text{ratio}}}$ is the correlation matrix, $\left(\mathbf{R}_{\mathbf{P}_{\mathbf{Y}}^{\text{ratio}} \mathbf{P}_{\hat{\mathbf{S}}}^{\text{ratio}}}\right)_{ij}$ is the Pearson correlation between i^{th} and j^{th} rows of $\mathbf{P}_{\mathbf{Y}}^{\text{ratio}}$ and $\mathbf{P}_{\hat{\mathbf{S}}}^{\text{ratio}}$ respectively, $\mathbf{P}_{\mathbf{Y}}^{\text{ratio}}$ is the matrix of power ratios of the separated signals in the k^{th} frequency bin whose t^{th} column is given by $\mathbf{P}_{\mathbf{Y}}^{\text{ratio}}(t) = \left[\frac{\|Y_1(k,t)\|^2}{\sum_{q=1}^{Q} \|Y_q(k,t)\|^2}, \cdots, \frac{\|Y_Q(k,t)\|^2}{\sum_{q=1}^{Q} \|Y_q(k,t)\|^2} \right]^T$. Similarly, $\mathbf{P}_{\hat{\mathbf{S}}}^{\text{ratio}}$ is the matrix of power ratios of the signal picked up by the p^{th} microphone at the k^{th} frequency bin whose column vectors are given by $\mathbf{P}_{\hat{\mathbf{S}}}^{\text{ratio}}(t) = \left[\frac{\|H_{p1}(k)S_1(k,t)\|^2}{\sum_{q=1}^{Q} \|H_{pq}S_q(k,t)\|^2}, \cdots, \frac{\|H_{pQ}(k)S_Q(k,t)\|^2}{\sum_{q=1}^{Q} \|H_{pq}(k)S_q(k,t)\|^2} \right]^T$, where $p \in \{1, \cdots, P\}$ is the index of the microphone to which the mask is applied. From Table II, it can be seen that the SIR improvement is higher when k-means clustering is used compared to FCM clustering. However, the improvement in artifacts and distortion are higher when the FCM clustering algorithm is used. It can also be seen from the Table that the proposed method based on k-means clustering for solving the permutation problem is as good as solving the permutation problem by comparing the separated signals with the clean signals. Table II show that the k-means clustering for solving



(a)



(b)



(c)

Fig. 8. SDR/SIR/SAR versus index of the microphone output on which mask is applied, for different microphone spacings. Dotted lines are for the cases where the permutation problem is solved by finding the correlation between the bin-wise power ratios of the separated signals and that of clean signals picked up by the microphones. Solid lines are for the cases where the permutation problem is solved by the proposed method based on the k-means clustering algorithm. The mean input SDR, SIR and SAR are -0.09dB, 0dB and 20.82dB respectively.

the permutation problem out-perform the correlation method using the clean signals in the experiments using two sources. The reason for this is can be explained as follows. In practice, the sources are not perfectly disjoint in their TF domain. Hence the separated signals will have some distortion when we use binary masking method. Due to this, the correlation between the separated signals or the corresponding masks in the adjacent bins will be higher than that between the separated signals and the clean signals. When the number of sources increases the distortion on the separated signals will be more because of the increased spectra overlap. If the distortions on the separated signals are too high, the robustness of the k-means clustering algorithm will decrease and the correlation method using the clean signals will out-perform the k-means clustering method.

The time taken to execute the proposed algorithm when coded in Matlab and run in a PC with Intel Core 2 Duo 2.66 GHz CPU, 2 GB of RAM is shown in Table III. Note that the k-means algorithm for the mask estimation is initialized with the result obtained from the histogram method on $\Theta_H^{(k)}$, whereas the FCM algorithm was initialized with randomly selected samples from $\Theta_H^{(k)}$.

D. Microphone spacing and selection of microphone output to apply mask.

The estimated mask can be applied to the mixture in the TF domain obtained from one of the microphone outputs. In this experiment we examine the output of the microphone on which the mask is to be applied to obtain the best performance. It is logical to apply the masks to the output of the center microphone which is proven experimentally and shown in Figs.8 to be the best choice.

In our experiments, the simulated impulse responses obtained for the source microphone configuration shown in Fig.7 is used. Out of the total six sources, only two sources are active at any time and hence we have a total of $\frac{6!}{2!(6-2)!} = 15$ combinations of source positions. For each combination of source positions the experiment is repeated for 50 sets of utterances. The performances shown in Figs.8 are the mean performances of these 750 experiments. To study the effect of microphone spacing, these 750 experiments are repeated for different microphone spacing. For this purpose microphone arrays consisting of five microphones with different spacings (2cm, 5cm, 10cm and 20cm) are used. For all the microphone spacings the center of the array is kept at the same point. The

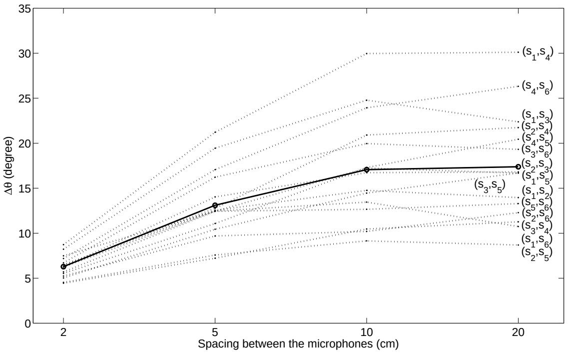


Fig. 9. Variation in angle between the column vectors $\mathbf{H}_q(k)$, $q = 1, 2$ versus microphone spacing. Dotted lines show the angles for different source combinations, as marked in the figure, and solid line shows the mean angle.

experimental results show that the performance improves as the spacing between the microphones increases, and after a certain distance this improvement begins to drop. The reason for the variation in performance because of the variation in spacing between the microphones can be explained as follows.

When the microphones are very close the difference between the impulse responses of any one source and the microphones is small. For example, the impulse response between source s_1 and microphone Mic.1 will be almost the same as that between s_1 and microphone Mic.2 when both microphones are very close to one another. Hence in the frequency domain, the column vectors $\mathbf{H}_q(k)$, $q = 1, \cdots, Q$ will be very close to one another and as a result the angles between them will be small. When the angles between the mixing vectors are very small, partitioning of the samples will be difficult and the separation performance will also be poor. On the other hand, if we go on increasing the spacing, as the maximum Hermitian angle between the column vectors $\mathbf{H}_q(k)$, $q = 1, \cdots, Q$ is $\pi/2$, after a certain distance there will not be any increase in performance. This fact is illustrated in Fig.9 where the average angle between the column vectors $\mathbf{H}_q(k)$, $q = 1, \cdots, Q$, over the first 100 bins as a function of spacing between the microphones are shown.

It may be noted that, in Fig.8, for 2 cm microphone spacing the performance is lower when the proposed k-means clustering algorithm is used for solving the permutation problem than when the correlation between the power ratios of the separated and clean signals are used. This is because when the spacing is small, the clustering of $\Theta_H^{(k)}$ will be difficult which will lead to error in mask estimation. For the proposed algorithm for solving the permutation problem, the robustness of the cluster formation depends on the quality of the estimated masks. If the mask quality is poor, the permutation problem will not be solved perfectly which will result in poor separation in the time domain. When we use the correlation between the clean signals and separated signals for solving the permutation problem, the robustness will be very high and the decrease in performance will be mainly due to the imperfect separation in each frequency bin, and that due to the error in solving the permutation problem will be minimum.

E. Effect on the number of microphones

Generally in BSS, the larger the number of microphones, the better the performance. This observation also holds in our case. The SDR, SIR and SAR improvements for different combinations of number of sources and microphones are shown in Fig.10 where the masks are generated using k-means clustering. The source microphone positions are the same as that in Fig.7. The spacings between the microphones are fixed at 10cm for all the experiments. For the case of odd number of microphones, the masks are applied to the output of the centre microphone. When the numbers of microphones are 2 and 4, masks are applied to the first and the second microphone outputs respectively. As explained in Section III-D, for two sources, because of the 15 combinations of source position, 750 simulations were done. Similarly, 1000, 750, 300 and 50 simulations were done for 3, 4, 5 and 6 sources respectively and the mean performances so obtained are shown in Fig.10. From Figs.8 and 10, it can be seen that the binary masking method for the separation of the sources from their mixtures will introduce artifacts due to nonlinear distortions. This cannot be avoided and it will increase as the overlapping of the sources increases. To mitigate this problem, some post processing techniques have to be used [39].

IV. Conclusion

In this paper, an algorithm for separation of an unknown number of sources from their underdetermined convolutive mixtures via TF masking and a method for solving the permutation problem by clustering the masks using k-means clustering is proposed. The algorithm uses the membership functions from the clustering algorithm as the masks. The separation performance of the algorithm is evaluated for the two popular clustering algorithms, namely k-means and fuzzy c-means. The crisp nature of the membership functions generated by the k-means algorithm resulted in more artifacts in the separated signals compared to those by fuzzy c-means algorithm, which is a soft partitioning technique. For the automatic detection of the number of sources, the optimum number of clusters formed by the Hermitian angles in different frequency bins are estimated and the number that estimated most frequently is taken as the number of sources present in the mixture. In this paper, the cluster validation technique is used for the estimation of the number of cluster; however, other techniques can also be used. In TF masking methods for BSS, in general, the scaling problem does not exist and this is true for the proposed algorithm also. However, the well-known permutation problem still exists but could be solved by clustering. The validity of the proposed algorithms are demonstrated for both real room and simulated speech mixtures.

For the experiments in this paper, the signals used were not sparse in the time domain. Furthermore, in the frequency domain, the overlappings are high for larger numbers of sources. In a practical situation, for example, conversation in a meeting room, the signals will be sparse even in the time domain. Considering this fact and the speed of the algorithm (the algorithm will be much faster than that shown in Table III,

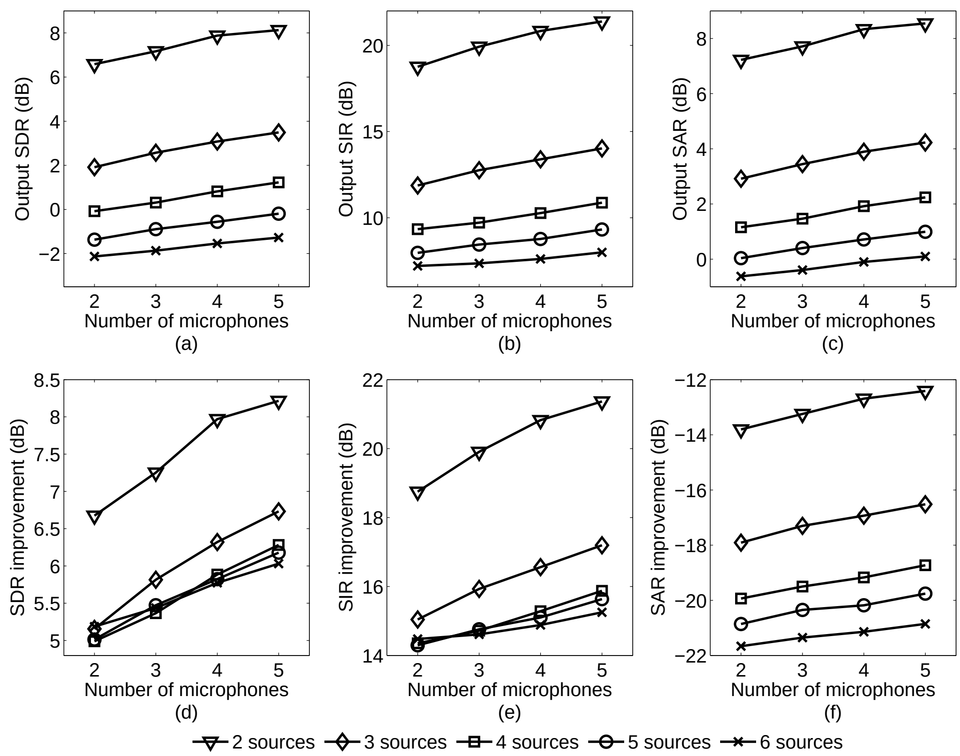


Fig. 10. Performance versus number of microphones. (a) output SDR (b) output SIR (c) output SAR (d) SDR improvement (i.e., output SDR - input SDR) (e) SIR improvement (f) SAR improvement.

if we implement it in other languages such as C or C++), the proposed algorithm is suitable for real world applications.

APPENDIX

If we multiply $\mathbf{u}_1$ and $\mathbf{u}_2$ by the complex scalars a and b respectively, then (8) will become

$$\begin{aligned}\cos(\theta_C) &= \frac{(a\mathbf{u}_1)^H(b\mathbf{u}_2)}{\sqrt{(a\mathbf{u}_1)^H(a\mathbf{u}_1)}\sqrt{(b\mathbf{u}_2)^H(b\mathbf{u}_2)}} \\ &= \frac{\sum_i a^* u_{i1}^* b u_{i2}}{\sqrt{\sum_i a^* u_{i1}^* a u_{i1}}\sqrt{\sum_i b^* u_{i2}^* b u_{i2}}}\end{aligned} \tag{33}$$

where u_{iq} is the i^{th} element of the column vector $\mathbf{u}_q$ and * represents the complex conjugate operation. Let

$$a = Ae^{j\theta_A} \tag{34}$$

$$b = Be^{j\theta_B} \tag{35}$$

$$u_{i1} = U_{i1}e^{j\phi_i} \tag{36}$$

$$u_{i2} = U_{i2}e^{j\psi_i} \tag{37}$$

then $\cos(\theta_C)$ will be as shown in (39) and

$$\begin{aligned}\cos(\theta_H) &= |\cos(\theta_C)| \\ &= \frac{\left|e^{j(\theta_B-\theta_A)}\sum_i U_{i1}U_{i2}e^{j(\psi_i-\phi_i)}\right|}{\sqrt{\sum_i U_{i1}^2}\sqrt{\sum_i U_{i2}^2}} \\ &= \frac{\left|e^{j(\theta_B-\theta_A)}\right|\left|\sum_i U_{i1}U_{i2}e^{j(\psi_i-\phi_i)}\right|}{\sqrt{\sum_i U_{i1}^2}\sqrt{\sum_i U_{i2}^2}} \\ &= \frac{\left|\sum_i U_{i1}U_{i2}e^{j(\psi_i-\phi_i)}\right|}{\sqrt{\sum_i U_{i1}^2}\sqrt{\sum_i U_{i2}^2}}\end{aligned} \tag{38}$$

which is independent of a and b.

$$\begin{aligned}\cos(\theta_C) &= \frac{\sum_i Ae^{-j\theta_A}U_{i1}e^{-j\phi_i}Be^{j\theta_B}U_{i2}e^{j\psi_i}}{\sqrt{\sum_i Ae^{-j\theta_A}U_{i1}e^{-j\phi_i}Ae^{j\theta_A}U_{i1}e^{j\phi_i}}\sqrt{\sum_i Be^{-j\theta_B}U_{i2}e^{-j\psi_i}Be^{j\theta_B}U_{i2}e^{j\psi_i}}} \\ &= \frac{ABe^{j(\theta_B-\theta_A)}\sum_i U_{i1}U_{i2}e^{j(\psi_i-\phi_i)}}{A\sqrt{\sum_i U_{i1}^2}B\sqrt{\sum_i U_{i2}^2}} \\ &= \frac{e^{j(\theta_B-\theta_A)}\sum_i U_{i1}U_{i2}e^{j(\psi_i-\phi_i)}}{\sqrt{\sum_i U_{i1}^2}\sqrt{\sum_i U_{i2}^2}}\end{aligned} \tag{39}$$

References

[1] A. Hyvrinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. John Wiley & Sons Ltd, New York, 2001.

[2] A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing*. John Wiley & Sons Ltd, New York, 2002.

[3] H.Buchner, R. Aichner, and W. Kellermann, "A generalization of blind source separation algorithms for convolutive mixtures based on second-order statistics," *IEEE Transactions on Speech and Audio Processing*, vol. 13, pp. 120–134, Jan. 2005.

[4] S. C. Douglas, M. Gupta, H. Sawada, and S. Makino, "Spatio-Temporal FastICA algorithms for the blind separation of convolutive mixtures," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, pp. 1511–1520, July 2007.

[5] A. Aissa-El-Bey, K. Abed-Meraim, and Y. Grenier, "Blind separation of underdetermined convolutive mixtures using their time-frequency representation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, pp. 1540–1550, July 2007.

[6] P. Bofill and M. Zibulevsky, "Underdetermined blind source separation using sparse representation," *Signal Processing*, vol. 81, pp. 2353–2362, Nov. 2001.

[7] H. Sawada, S. Araki, R. Mukai, and S. Makino, "Blind extraction of dominant target sources using ICA and Time-Frequency masking," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, pp. 2165–2173, Nov. 2006.

[8] S. Araki, S. Makino, A. Blin, R. Mukai, and H. Sawada, "Underdetermined blind separation for speech in real environments with sparseness and ICA," in *Proceedings of the ICASSP*, pp. iii–881–884, May 2004.

[9] S. Araki, H. Sawada, R. Mukai, and S. Makino, "Underdetermined blind sparse source separation for arbitrarily arranged multiple sensors," *Signal Processing*, vol. 87, pp. 1833–1847, Aug. 2007.

[10] A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete–Time Signal Processing*. Prentice Hall, 2003.

[11] O. Yilmaz and S. Rickard, "Blind separation of speech mixtures via time–frequency masking," *IEEE Transactions on Signal Processing*, vol. 52, pp. 1830–1847, July 2004.

[12] G. Xu, H. Liu, L. Tong, and T. Kailath, "A least-squares approach to blind channel identification," *IEEE Transactions on Signal Processing*, vol. 43, pp. 2982–2993, Dec. 1995.

[13] A. Aissa-El-Bey, M. Grebici, K. Abed-Meraim, and A. Belouchrani, "Blind system identification using cross-relation methods: Further results and developments," in *Proceedings of the Int. Symp. Signal Process. Applicat.*, pp. 649–652, July 2003.

[14] K. Scharnhorst, "Angles in complex vector spaces," *Acta Applicandae Mathematicae*, vol. 69, pp. 95–103, Nov. 2001.

[15] H. Sawada, S. Araki, and S. Makino, "A two–stage frequency–domain blind source separation method for underdetermined convolutive mixtures," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 139–142, Oct. 2007.

[16] S. Kurita, H. Saruwatari, S. Kajita, K. Takeda, and F. Itakura, "Evaluation of blind signal separation method using directivity pattern under reverberant conditions," in *Proceedings of the ICASSP*, pp. 3140–3143, 2000.

[17] H. Saruwatari, S. Kurita, K. Takeda, F. Itakura, T. Nishikawa, and K. Shikano, "Blind source separation combining independent component analysis and beamforming," *EURASIP Journal on Applied Signal Processing*, vol. 2003, no. 11, pp. 1135–1146, 2003.

[18] M. Ikram and D. Morgan, "A beamforming approach to permutation alignment for multichannel frequency domain blind speech separation," in *Proceedings of the ICASSP*, pp. 881–884, 2002.

[19] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Computing Surveys*, vol. 31, pp. 264–323, Sept. 1999.

[20] R. Xu and Donald II Wunsch, "Survey of clustering algorithms," *IEEE Transactions on Neural Networks*, vol. 16, pp. 645–678, May 2005.

[21] J. C. Bezdek, *Pattern Recognition With Fuzzy Objective Function Algorithms*. New York: Plenum Press, 1981.

[22] D. Arthur and S. Vassilvitskii, "k–means++: The advantages of careful seeding," in *Proceedings of the eighteenth annual ACM–SIAM symposium on Discrete algorithms*, pp. 1027–1035, 2007.

[23] Y. Zhang, W. Wang, X. Zhang, and Y. Li, "A cluster validity index for fuzzy clustering," *Information Sciences*, vol. 178, pp. 1205–1218, Feb. 2008.

[24] H. Sun, S. Wang, and Q. Jiang, "FCM–based model selection algorithms for determining the number of clusters," *Pattern Recognition*, vol. 37, pp. 2027–2037, Oct. 2004.

[25] M. K. Pakhira, S. Bandyopadhyay, and U. Maulik, "Validity index for crisp and fuzzy clusters," *Pattern Recognition*, vol. 37, pp. 487–501, Mar. 2004.

[26] P. Guo, C. L. P. Chen, and M. R. Lyu, "Cluster number selection for a small set of samples using the bayesian Ying-Yang model," *IEEE Transactions on neural networks*, vol. 13, pp. 757–763, May 2002.

[27] N. Murata, S. Ikeda, and A. Ziehe, "An approach to blind source separation based on temporal structure of speech signals," *Neurocomputing*, vol. 41, pp. 1–24, Oct. 2001.

[28] M. Ikram and D. Morgan, "Permutation inconsistency in blind speech separation: Investigation and solutions," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 13, pp. 1–13, Jan. 2005.

[29] H. Sawada, S. Araki, and S. Makino, "Measuring dependence of binwise separated signals for permutation alignment in frequency-domain BSS," in *IEEE Int. Symp. on Circuits and Systems*, pp. 3247–3250, May 2007.

[30] V. G. Reju, S. N. Koh, and I. Y. Soon, "Partial separation method for solving permutation problem in frequency domain blind source separation of speech signals," *Neurocomputing*, vol. 71, pp. 2098–2112, June 2008.

[31] V. G. Reju, S. N. Koh, and I. Y. Soon, "A robust correlation method for solving permutation problem in frequency domain blind source separation of speech signals," in *Proceedings of the APCCAS*, pp. 1893–1896, 2006.

[32] H. Sawada, R. Mukai, S. Araki, and S. Makino, "A robust and precise method for solving the permutation problem of frequency domain blind source separation," *IEEE Transactions on Speech and Audio Processing*, vol. 12, pp. 530–538, Sept. 2004.

[33] R. Aichner, H. Buchner, and W. Kellermann, "Real-time convolutive blind source separation based on a broadband approach," in *Fifth International Symposium on Independent Component Analysis and Blind Signal Separation*, pp. 840–847, 2004.

[34] http://bass-db.gforge.inria.fr/BASS-dB/?show=browse&id=filters.

[35] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, pp. 1462–1469, July 2006.

[36] C. Fevotte, R. Gribonval, and E. Vincent, "BSS EVAL toolbox user guide, IRISA technical report 1706," tech. rep., Rennes, France, Apr. 2005. http://www.irisa.fr/metiss/bss eval/.

[37] http://www.purebits.com/.

[38] H. Buchner, R. Aichner, and W. Kellermann, "A generalization of a class of blind source separation algorithms for convolutive mixtures," in *Proceedings of the Int. Symp. on Independent Component Analysis and Blind Signal Separation*, pp. 945–950, 2003.

[39] J. Rosca, T. Gerkmann, and D. C. Balcan, "Statistical inference of missing speech data in the ICA domain," in *Proceedings of the ICASSP*, vol. 5, pp. 617–620, May 2006.