Digital Signal Processing 31 (2014) 79–92

Contents lists available at ScienceDirect

Digital Signal Processing

www.elsevier.com/locate/dsp

Multi-band multi-centroid clustering based permutation alignment for frequency-domain blind speech separation

Lin Wang [a,b]

[a] Institute of Physics, University of Oldenburg, Germany
[b] School of Electronic and Information Engineering, Dalian University of Technology, China

ARTICLE INFO

Article history:
Available online 28 April 2014

Keywords:
Blind source separation
Convolutive mixture
Frequency domain
Permutation ambiguity

ABSTRACT

This paper investigates the permutation ambiguity problem in frequency-domain blind source separation and proposes a robust permutation alignment algorithm based on inter-frequency dependency, which is measured by the correlation coefficient between the time activity sequences of separated signals. To calculate a global reference for permutation alignment, a multi-band multi-centroid clustering algorithm is proposed where at first the permutation inside each subband is aligned with multi-centroid clustering and then the permutation among subbands is aligned sequentially. The multi-band scheme can reduce the dynamic range of the activity sequence and improve the efficiency of clustering, while the multi-centroid clustering scheme can improve the precision of the reference and reduce the risk of wrong permutation among subbands. The combination of two techniques enables to capture the variation of the time–frequency activity of a speech signal precisely, promising robust permutation alignment performance. Extensive experiments are carried out in different testing scenarios (up to reverberation time of 700 ms and 4×4 mixtures) to investigate the influence of two parameters, the number of subbands and the number of clustering-centroids, on the performance of the proposed algorithm. Comparison with existing permutation alignment algorithms proves that the proposed algorithm can improve the robustness in challenging scenarios and can reduce block permutation errors effectively.

1. Introduction

Blind source separation has attracted considerable attention in research communities in recent years. Its main objective is to separate multiple sources mixed through unknown channels using only the observations of their mixtures. It has found a lot of potential applications including noise robust speech recognition, crosstalk separation in telecommunications, biomedical signal analysis and so on. The term "blind" means that separation is done without using any information about the mixing channels and the sources. The simplest BSS model assumes the existence of independent signals and the observation of their mixtures being linear and instantaneous. This problem can be solved by independent components analysis (ICA) [1,2]. A more challenging case is when the sources are mixed through convolutive channels [3]. This situation is common for audio applications where signals are recorded in a reverberant environment. Long reverberation time and non-stationary mixing conditions make the estimation of the original source signals a challenging task [4,5]. In addition, the high computational complexity hampers the application of the algorithms in real-time devices.

Traditional approaches to solve the convolutive blind source separation problem can be classified into two categories: time-domain and frequency-domain. In time-domain BSS, the separation network is derived by optimizing a time-domain cost function [6–8]. These approaches may not be effective due to slow convergence and large computation load. In frequency-domain BSS, the observed time-domain signals are converted into the time–frequency domain by short-time Fourier transform (STFT), and then an instantaneous BSS algorithm is applied to each frequency bin, after which the separated signals of all frequency bins are combined and inverse-transformed back to the time domain [9–13]. Although satisfactory instantaneous separation may be achieved at all frequency bins, combining them to recover the original sources is a challenge because there are unknown permutations associated with individual frequency bins. This permutation ambiguity should be dealt with properly so that the separated frequency components from the same source are grouped together.

There are generally three strategies to tackle the permutation ambiguity problem:

E-mail addresses: lin.wang@uni-oldenburg.de, wanglin_2k@sina.com.

http://dx.doi.org/10.1016/j.dsp.2014.04.009
1051-2004/

- The first strategy is to exploit the continuity of the separation matrices across frequencies. In [7,14,15], separation filters of shorter length relative to the fast Fourier transform (FFT) block size are used so that the separation filters are smooth across frequencies and hence the permutation ambiguities are avoided. In [16], the magnitude continuity of the separation filters is used to align the permutation. In [17] a recursively regularized ICA (RR-ICA) algorithm is proposed which uses a predicted separation matrix from previous frequency bins as the initialization for the current bin. With this recursive initialization scheme, the phase continuity of the separation filters is maintained and the permutation ambiguities are minimized.
- The second strategy is to use the time structure of separated frequency bins, such as the inter-frequency dependency of the amplitude of separated signals [18,19], assuming high correlation between neighboring bins. Aligning consecutive bins using inter-frequency dependency may be precise but not robust, since a single wrong permutation leads to whole blocks of falsely permuted bins. This is referred to as misalignment spread. To solve this problem, clustering-based and region-wise permutation alignment schemes are proposed [19–23]. As a particular formulation of joint blind source separation [24,25], independent vector analysis (IVA) algorithms are proposed which directly incorporate the inter-frequency dependency measure into instantaneous ICA so that permutation ambiguities can be minimized by a joint optimization across all the frequency bins [26–30].
- The last strategy is to use position information of the sources such as direction of arrival (DOA) or time difference of arrival (TDOA) [31–34]. It is believed that contributions from the same source are likely to come from the same direction. By estimating the arriving delay of the sources or analyzing the directivity pattern formed by a separation matrix, source directions can be estimated and permutations aligned. The major drawback of this approach is that the assumption of sources originating from specific directions is only valid in low reverberation. Although robust in low reverberation, the performance of this strategy degrades significantly in highly reverberant environments. In addition, it fails to align the permutation when the two sources are closely spaced. In [35–38], source direction information and inter-frequency dependency of the separated signals are combined to get a precise and robust permutation result.

In this paper we aim at solving the permutation ambiguity problem based on inter-frequency dependency of separated signals, which can be measured by the correlation of the time activity sequences at individual frequency bins. The frequency-dependent time activity sequence, as can be calculated from the power ratio of the separated signals, has proven to show strong dependency between two frequencies if they come from the same source [19]. The key of permutation alignment is to find a frequency-independent global reference for each source respectively. By aligning to the reference the permutation can be corrected across the whole frequency band. Several ways have already been proposed to estimate such a reference:

- The simplest way is to use the time-activity sequence at an adjacent bin as a reference to align the permutation at the current bin [22]. Obviously, this bin-wise processing is sensitive to the permutation error at an individual bin, leading to misalignment spread.
- In [22], a region-growing algorithm is proposed, which divides the full frequency band into multiple regions based on the bin-wise permutation alignment result and then merges these regions together in a region-growing way, and shown to prevent misalignment spread effectively. The region-growing algorithm calculates the reference adaptively by doing permutation alignment and reference update simultaneously during its region-growing procedure. One disadvantage is that the permutation errors at individual bins may accumulate during the region-growing procedure, leading to possibly wrong updates of the reference.
- Clustering-based algorithms, which perform one-centroid [19] or multi-centroid clustering across the whole frequency band [20], are proposed to estimates centroid sequences for each source and uses them as a global reference for permutation alignment. In [21] this work is extended to an underdetermined separation problem. Compared with the region-growing algorithm, the clustering-based algorithm performs the two tasks of reference calculation and permutation alignment separately. It is less affected by the permutation alignment results at individual bins and thus tends to be more robust especially when a multi-centroid scheme is employed. However, although the time activity of a speech signal shows strong inter-frequency similarity, it still varies slowly with frequency. In some case, it is difficult to find a global reference that is consistent to all the bins throughout the frequency band. As a result, permutation errors still occur at some bins or even throughout a block of bins.

Given the deficiencies of the existing algorithms above, we propose a multi-band multi-centroid clustering algorithm to better estimate the reference. The proposed algorithm in essence is a combination of multi-band processing and multi-centroid clustering, i.e., the whole frequency band is split into multiple bands, and in each subband the permutation reference is estimated in a multi-centroid way. After the permutation correction inside each subband, the permutation between these subbands is aligned sequentially to recover the full frequency band. The multi-band scheme can reduce the dynamic range of the activity sequences and improve the efficiency of clustering; while the multi-centroid clustering scheme can improve the precision of the reference and reduce the risk of wrong permutation among subbands. The combination of two techniques can capture the variation of the time–frequency activity of a speech signal more precisely and hence promises better permutation alignment results. Based on the principle of the proposed algorithm, the number of subbands and the number of centroids play important roles on the permutation alignment performance. The impact of the two parameters will be investigated in this paper.

One issue which has not been studied well in previous papers is the robustness of a permutation alignment algorithm, i.e., it may perform well for one mixing scenario but fails in another case. For this reason, the performance of the permutation alignment algorithm is investigated extensively in various mixing scenarios with different reverberation time (up to 700 ms), number of sources (up to 4), and testing files. To make a comprehensive evaluation, four objective measures (the mean performance and robustness performance in terms of permutation error and signal-interference-ratio) are newly defined and computed with multiple testing files. Experimental results clearly show the superior performance of the proposed method over the single-band multi-centroid algorithm. Especially, it can improve the robustness and reduce block permutation errors effectively. Finally, comparison with other referenced permutation alignment algorithms in both simulated and real environments also proves the advantage of the proposed algorithm.

The rest of the paper is organized as follows. The principle of frequency-domain blind source separation is reviewed in Section 2. The proposed multi-band multi-centroid permutation alignment scheme is described in detail in Section 3. Experimental results are presented in Section 4. Computational cost analysis of the

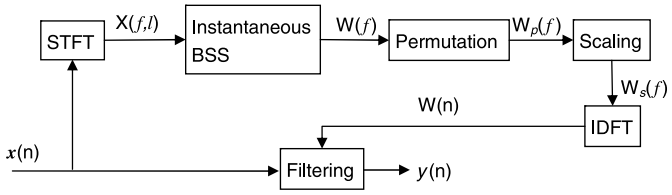


Fig. 1. Workflow of the frequency-domain blind source separation.

proposed algorithm is given in Section 5. Finally, Section 6 concludes the paper.

2. Frequency-domain blind source separation

Depending on the number of sources N and microphones M, the BSS problem can be classified into overdetermined ($N < M$), determined ($N = M$), and underdetermined ($N > M$) cases, each involving a different ICA procedure [3,21]. The paper focuses on the determined case, i.e., with equal number of sources and microphones.

Supposing N sources and N microphones in a real-world acoustic scenario, the source vector $\boldsymbol{s}(n) = [s_1(n), \cdots, s_N(n)]^{\mathrm{T}}$, and the observed vector $\boldsymbol{x}(n) = [x_1(n), \cdots, x_M(n)]^{\mathrm{T}}$. In reverberant scenarios, the acoustical mixing channels are modeled by finite impulse response (FIR) filters of length P, and the convolutive mixing process is formulated as

$$\boldsymbol{x}(n) = \boldsymbol{H}(n) * \boldsymbol{s}(n) = \sum_{p=0}^{P-1} \boldsymbol{H}(p)\boldsymbol{s}(n-p), \tag{1}$$

where $\boldsymbol{H}(n)$ is a sequence of $N \times N$ matrices containing the impulse responses of mixing channels, and n is time index. For separation, we use FIR demixing filters of length Q and obtain the estimated source signal vector $\boldsymbol{y}(n) = [y_1(n), \cdots, y_N(n)]^{\mathrm{T}}$ by

$$\boldsymbol{y}(n) = \boldsymbol{W}(n) * \boldsymbol{x}(n) = \sum_{q=0}^{Q-1} \boldsymbol{W}(q)\boldsymbol{x}(n-q), \tag{2}$$

where $\boldsymbol{W}(n)$ is a sequence of $N \times N$ matrices containing the demixing filters.

Although the demixing network $\boldsymbol{W}(n)$ can be estimated directly in the time domain, the task of estimating many parameters simultaneously has to face the challenge of slow convergence and high computational demand. A more popular way is to do it in the frequency domain. The workflow of the frequency-domain BSS is shown in Fig. 1.

By using a blockwise Q-point short-time Fourier transform (STFT), the time-domain convolution regarding the mixing process can be converted into frequency-domain multiplications and correspondingly the convolutive BSS problem is converted into multiple instantaneous BSS problems at individual frequency bins. This is expressed as

$$\boldsymbol{x}(f,l) = \boldsymbol{H}(f)\boldsymbol{s}(f,l), \tag{3}$$

where l is a decimated version of the time index n, f is the frequency index, $\boldsymbol{H}(f)$ is the Fourier transform of $\boldsymbol{H}(n)$, and $\boldsymbol{x}(f,l)$ and $\boldsymbol{s}(f,l)$ are the STFTs of $\boldsymbol{x}(n)$ and $\boldsymbol{s}(n)$, respectively.

The instantaneous BSS problem at each frequency bin can be solved relative easily by applying a complex-valued ICA to the time series $\boldsymbol{x}(f,l)$. The ICA algorithms for instantaneous BSS have been studied for many years and are considered to be quite mature [1, 39–41]. For instance, the demixing matrix can be estimated iteratively by using the well-known Infomax algorithm [1], i.e.,

$$\begin{cases} \boldsymbol{y}(f,l) = \boldsymbol{W}(f)\boldsymbol{x}(f,l) \\ \boldsymbol{W}(f) = \boldsymbol{W}(f) + \eta\big(\boldsymbol{I} - \mathrm{E}\big[\Phi\big(\boldsymbol{y}(f,l)\big)\boldsymbol{y}^{\mathrm{H}}(f,l)\big]\big)\boldsymbol{W}(f), \end{cases} \tag{4}$$

where $\boldsymbol{I}$ is an identity matrix, $\Phi(\cdot)$ is a nonlinear function, and $\mathrm{E}[\cdot]$ is the expectation operator.

With the estimated demixing matrix $\boldsymbol{W}(f)$, the original source signals can be recovered up to scaling and permutation ambiguities:

$$\boldsymbol{y}(f,l) = \boldsymbol{W}(f)\boldsymbol{x}(f,l) = \Lambda(f)\boldsymbol{D}(f)\boldsymbol{s}(f,l), \tag{5}$$

where $\boldsymbol{D}(f)$ is a permutation matrix and $\Lambda(f)$ a scaling matrix at frequency f. It is necessary to correct the scaling and permutation ambiguities before transforming the signals back to the time domain:

- The permutation at each bin should be aligned so that the separated components originating from the same source are grouped together. Permutation alignment is a challenging problem and will be addressed in Section 3.
- The scaling ambiguity can be resolved by using the Minimal Distortion Principle [42], i.e.,

$$\boldsymbol{W}_s(f) = \mathrm{diag}\big(\boldsymbol{W}_p^{-1}(f)\big) \cdot \boldsymbol{W}_p(f), \tag{6}$$

  where $\boldsymbol{W}_p(f)$ is $\boldsymbol{W}(f)$ after permutation correction, $(\cdot)^{-1}$ denotes inversion of a square matrix or pseudo inversion of a rectangular matrix, and $\mathrm{diag}(\cdot)$ retains only the main diagonal components of a matrix.

Finally, the demixing network $\boldsymbol{W}(n)$ is obtained by inverse Fourier transforming $\boldsymbol{W}_s(f)$, and the estimated source $\boldsymbol{y}(n)$ is obtained by filtering $\boldsymbol{x}(n)$ through $\boldsymbol{W}(n)$.

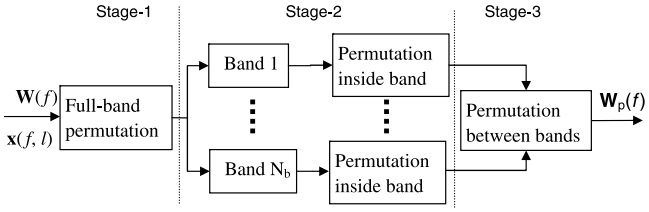
3. Multi-band multi-centroid permutation alignment

In this section we aim to solve the permutation ambiguity problem based on the inter-frequency dependency of separated signals. A key issue of permutation alignment is to find a frequency-independent global reference for each source, by aligning to which the permutation can be corrected across the whole frequency band. Although some clustering based algorithms have been proposed to calculate to calculate the global reference, these algorithms typically operate in a full-band style and may suffer block permutation errors due to the fact that the inter-frequency dependency of a speech signal becomes weak among far-apart frequency bins [19–21]. To solve this problem, a multi-band multi-centroid clustering based permutation alignment method is proposed, which divides the whole frequency band into multiple bands and estimates the reference in each subband with a multi-centroid clustering algorithm. In this way, the variation of the inter-frequency dependency of a speech signal can be captured precisely, producing satisfactory permutation alignment results.

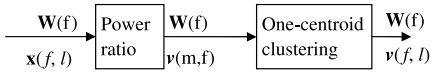
The remaining part of the section is organized as below. After defining the measure for inter-frequency dependency of speech sources, a multi-band multi-centroid clustering based permutation alignment algorithm is proposed in this section, followed by a discussion regarding the relationship between some existing permutation alignment algorithms and the proposed one. At last, local permutation alignment based post-processing method is presented.

3.1. Inter-frequency dependency

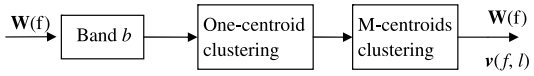
The inter-frequency dependency of speech sources can be used for permutation alignment based on the fact that neighboring frequency bins tend to show high dependency if they come from the same speech source. The inter-frequency dependency is usually measured by the correlation coefficient between some kind of time sequences of the separated signals. Although a commonly



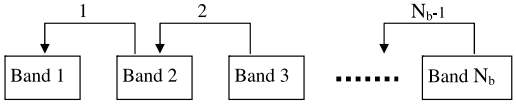
(a) Three-stage processing of the multi-band multi-centroid permutation alignment algorithm



(b) Stage-1: full-band permutation alignment with one centroid clustering



(c) Stage-2: permutation alignment with M-centroid clustering inside one subband



(d) Stage-3: permutation alignment between subands

Fig. 2. Block diagrams of the proposed multi-band multi-centroid permutation alignment method.

used time sequence is the envelopes of separated signals, recently it has been reported that the time activity sequence, which is calculated based on the power ratios of separated signals, can better explore the dependency among frequency bins [35]. Thus, this time activity sequence is employed in the proposed method for calculating the inter-frequency dependency.

In order to obtain the time activity sequence, the power ratio of separated signals is defined first [19]. Given the demixing matrix $\boldsymbol{W}(f)$ at the frequency f, the acoustical mixing matrix can be estimated as $\boldsymbol{A}(f) = \boldsymbol{W}^{-1}(f) = [\boldsymbol{a}_1(f), \cdots, \boldsymbol{a}_N(f)]$, where the $N \times 1$ vector $\boldsymbol{a}_i(f)$ describes the path from the i-th source to the N microphones. The power ratio is defined at each time–frequency bin for the i-th separated source $y_i(f, l)$ as

$$v_i^f(l) = \frac{\|\boldsymbol{a}_i(f) y_i(f, l)\|^2}{\sum_{j=1}^{N} \|\boldsymbol{a}_j(f) y_j(f, l)\|^2}, \tag{7}$$

where $\|\cdot\|^2$ denotes the norm-2 operation. The numerator of (7) represents the power of the i-th separated signal at all N microphones while the denominator represents the power of all separated signals. This definition measures the dominance of the i-th separated signal in the observed mixtures at the time–frequency bin (f, l): being in the range $[0, 1]$, (7) is close to 1 when the i-th separated signal is dominant, and close to 0 when others are dominant. The time sequence of the power ratio in some sense reflects the time activity of the signal and thus is called the "time activity sequence".

The correlation coefficient between two time activity sequences $\boldsymbol{v}_i^{f_1}$ and $\boldsymbol{v}_j^{f_2}$, the i-th separated signal at frequency f_1 and the j-th separated signal at frequency f_2, is defined as

$$\rho\left(\boldsymbol{v}_i^{f_1}, \boldsymbol{v}_j^{f_2}\right) = \frac{r_{ij}(f_1, f_2) - \mu_i(f_1)\mu_j(f_2)}{\sigma_i(f_1)\sigma_j(f_2)}, \tag{8}$$

where $r_{ij}(f_1, f_2) = E\{\boldsymbol{v}_i^{f_1} \boldsymbol{v}_j^{f_2}\}$, $\mu_i(f) = E\{\boldsymbol{v}_i^{f}\}$, $\sigma_i(f) = \sqrt{E\{(\boldsymbol{v}_i^{f})^2\} - \mu_i^2(f)}$ are, respectively, the correlation, mean, and standard deviation, and $E\{\cdot\}$ denotes expectation regarding the time l. In general, (8) tends to be high if the separated channels i and j originate from the same source and low if they represent different sources. This property can be exploited for aligning the permutation among frequency bins.

3.2. Multi-band multi-centroid clustering

Given the measure of inter-frequency dependency, the following task is to find a frequency-independent global reference for each source. A multi-band multi-centroid clustering based permutation alignment algorithm is proposed, which can be divided into three stages as shown in Fig. 2a. The details of the three stages are shown in Fig. 2b to Fig. 2d, respectively. The first stage provides an initialization for the second stage by aligning the permutation across the whole frequency band with a one-centroid clustering method. In the second stage, the full frequency band is divided into multiple subbands, and the permutation inside each subband is aligned independently with a multi-centroid clustering method. Finally, in the third stage, the permutation among these subbands is aligned sequentially. In the following, the details of these stages will be described.

Stage-1: one-centroid clustering

In this stage, each source is assumed to have only one centroid time-activity sequence. This centroid is estimated by maximizing the correlation coefficient between the time-activity sequence of a source and its centroid. The cost function (originally defined in [19]) is given as

$$J\left(\{\boldsymbol{c}_k\}, \{\Pi_f\}\right) = \sum_{f \in F} \sum_{k=1}^{N} \rho\left(\boldsymbol{v}_i^f, \boldsymbol{c}_k\right)\big|_{i=\Pi_f(k)}, \tag{9}$$

where $\boldsymbol{c}_k$ is the centroid sequence of the k-th source, Π_f is the permutation at the f-th frequency bin, and the set F denotes the frequency bins under consideration (in Stage-1 it consists of all the bins in the full frequency band). The maximization of the cost function can be achieved via expectation maximization (EM) iteration procedure:

(1) Given the current permutation Π_f at all frequency bins, the centroid $\boldsymbol{c}_k$ can be calculated for the k-th source by using

$$\boldsymbol{c}_k = \frac{1}{N_F}\sum_{f\in F}\boldsymbol{v}_i^f\big|_{i=\Pi_f(k)}, \quad k=1,\cdots,N \tag{10}$$

where N_F is the number of bins in the set F.

(2) The permutation Π_f at each frequency bin is recalculated so that the correlation coefficient between each time activity sequence and the corresponding centroid is maximized:

$$\Pi_f \leftarrow \arg\max_{\Pi}\sum_{k=1}^{N}\rho\big(\boldsymbol{v}_i^f,\boldsymbol{c}_k\big)\big|_{i=\Pi_f(k)}, \quad \forall f\in F. \tag{11}$$

The two operations (10) and (11) are iterated until convergence, when the correlation coefficient in (11) does not change any more. Generally, less than 15 iterations are required to reach convergence.

Stage-2: multi-centroid clustering

In this stage, the whole frequency band is equally divided into N_b subbands, and each subband is processed independently. Inside each subband, the permutation alignment result from Stage-1 is further improved with a cascaded system consisting of one-centroid clustering and multi-centroid clustering.

The one-centroid clustering employs the same procedure as in Stage-1. The only difference is that the number of frequency bins considered is confined to a subband.

In multi-centroid clustering, each source is assumed to have N_c candidated centroids instead of one. In this way, the variation of the time activity across frequencies can be better captured. The set of N_c centroid set per source is estimated by maximizing the correlation coefficient between the time-activity sequence of a source and its centroid set. The cost function was originally defined in [20] and here we rewrite it for better understanding as

$$J\big(\{\mathcal{C}_k\},\{\Pi_f\}\big)=\sum_{f\in F_b}\sum_{k=1}^{N}\bar{\rho}\big(v_i^f,\mathcal{C}_k\big)\big|_{i=\Pi_f(k)}, \tag{12}$$

where the set F_b consists of the frequency bins in the b-th subband, $\mathcal{C}_k=\{\boldsymbol{c}_{k,1},\cdots,\boldsymbol{c}_{k,N_c}\}$ denotes the N_c-centroid set for k-th source and $\bar{\rho}$ denotes the correlation coefficient between a time activity sequence and a centroid set. The definition of $\bar{\rho}$ is expressed

$$\bar{\rho}(\boldsymbol{v},\mathcal{C}_k)=\max_m\big\{\rho(\boldsymbol{v},\boldsymbol{c}_{k,m})\big\}. \tag{13}$$

The cost function in (12) is maximized with expectation maximization (EM) iteration, distinguishing from the optimization procedure in Stage-1:

(1) For each source, the N_c-centroid set are calculated by applying a K-means clustering algorithm to its time activity sequences throughout the subband.

$$\mathcal{C}_k = k\,\mathrm{mean}\big(\boldsymbol{v}_i^f\big|_{i=\Pi_f(k),f\in F_b}\big), \quad k=1,\cdots,N \tag{14}$$

The K-means algorithm, which typically operates in an iterative way, is a popular clustering method which aims to partition observations into specified clusters in which each observation belongs to the cluster with the nearest mean [43].

(2) The permutation Π_f is recalculated so that the correlation coefficient between each time activity sequence and the corresponding centroid set is maximized:

$$\Pi_f \leftarrow \arg\max_{\Pi}\sum_{k=1}^{N}\bar{\rho}\big(\boldsymbol{v}_i^f,\mathcal{C}_k\big)\big|_{i=\Pi_f(k)}. \tag{15}$$

The two operations (14) and (15) are iterated until convergence, when the correlation coefficient in (15) does not change any more. With a good initialization from the one-centroid clustering, it generally requires less than 5 iterations for the multi-centroid clustering to reach the convergence.

Stage-3: permutation alignment among subbands

After permutation correction inside each subband, these subbands are further aligned sequentially from low to high frequency to form a full band (i.e., by firstly aligning the band 2 to the band 1, then aligning the band 3 to the band 2, and so on) as shown in Fig. 2(d). For two neighboring subbands B1 and B2, the permutation alignment procedure is as follows:

(1) Calculate the centroids $\boldsymbol{c}_k^{B1}$ and $\boldsymbol{c}_k^{B2}$ for the two subbands with

$$\boldsymbol{c}_k = \frac{1}{N_{F_b}}\sum_{f\in F_b}\boldsymbol{v}_i^f(n)\big|_{i=\Pi_f(k)}, \quad k=1,\cdots,N. \tag{16}$$

(2) Assuming the permutation of the subband B1 is already known as Π_{B1}, the permutation Π_{B2} which aligns the subband B2 to B1 is determined by maximizing the correlation coefficient between two centroids $\boldsymbol{c}_k^{B1}$ and $\boldsymbol{c}_k^{B2}$:

$$\Pi_{B2} \leftarrow \arg\max_{\Pi}\sum_{j=1}^{N}\rho\big(\boldsymbol{c}_k^{B1},\boldsymbol{c}_{k'}^{B2}\big)\big|_{k=\Pi_{B1}(j),k'=\Pi_{B2}(j)}. \tag{17}$$

For N_b subbands, the processing above is carried out N_b-1 times in total. Since each subband consists of a number of correctly aligned frequency bins, the region-based permutation alignment is quite robust.

In the proposed algorithm, the one-centroid clustering estimates one centroid of the time-activity sequences for each source, which may not be precise enough as a reference for permutation alignment; thus, the subsequent M-centroid clustering further splits the one centroid into multiple ones, which are used together as a reference for permutation alignment. On the one hand, the one-centroid clustering provides the M-centroid clustering initialization with coarsely aligned frequency bins. On the other hand, the M-centroid clustering improves the results from one-centroid clustering.

3.3. Remarks on the proposed algorithm

In fact, the clustering-based framework has been exploited for the permutation ambiguity problem in [19–21]. Compared with the proposed one, these algorithms do clustering in the whole frequency band and can be seen as a special case of the proposed algorithm. For instance, the algorithm proposed in [19] is equivalent to the proposed algorithm with $N_b=1$ and $N_c=1$; while the algorithm in [20] is equivalent to the proposed algorithm with $N_b=1$ and $N_c=2$. Since the time activity of a speech signal may vary with frequency, it is difficult to find a global reference that is consistent to all the bins throughout the whole frequency band. In some cases, the full-band algorithm may suffer from blocks of permutation errors.

The multi-band processing can tackle this problem efficiently. After band division, there are fewer bins available in each processing band, and the dynamic range of the time activity sequences becomes smaller accordingly, making it easier to find a reference that is consistent to all bins. Obviously, the multi-band processing is imposed to the risk of block permutation error between two subbands. However, with a multi-centroid clustering algorithm, the variation of the time activity sequence inside each subband can be better captured, promising better permutation alignment inside the subband. Consequently, the risk of wrong permutation

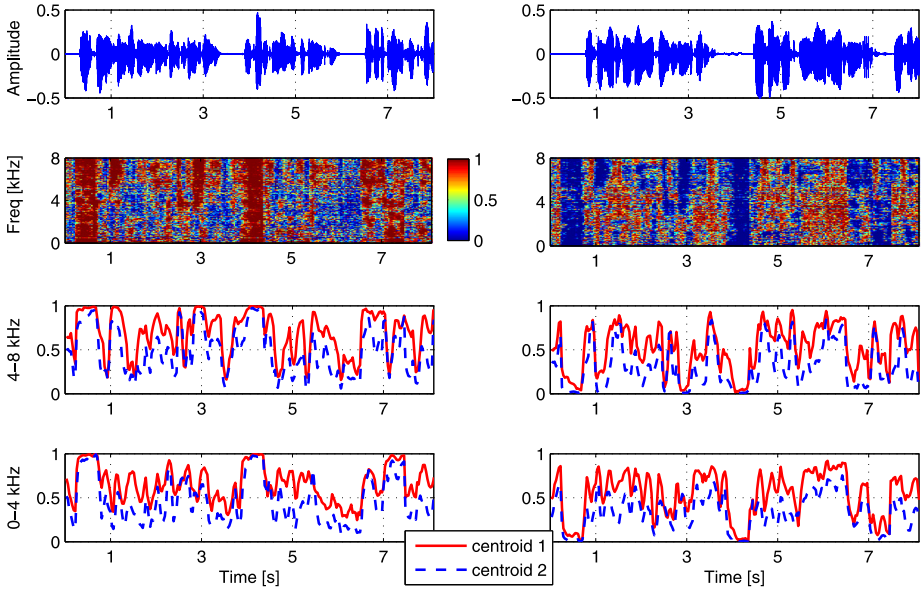


Fig. 3. Centroids estimated for a 2×2 mixture by the proposed multiple-band multiple-centroid algorithm with $N_b = 2$, $N_c = 2$.

between two subbands will be minimized once the bins inside them are already permutation corrected. Therefore, both multi-band and multi-centroid processing is important for permutation alignment. The multi-band processing reduces the dynamic range of the activity sequences and improves the efficiency of the clustering algorithm. The multi-centroid clustering further improves the precision of the reference and reduces the risk of wrong permutation among subbands. The two strategies work together, ensuring better performance.

For better understanding, a simple example is shown in Fig. 3 which depicts the multi-centroid clustering results for a 2×2 mixture by the proposed algorithm (2 bands, 2 centroids). The first row depicts the time-domain waveforms of the two source signals; the second row depicts the activity of the two sources in the time–frequency domain; the third row depicts the two centroids estimated for each source in the high frequency band (4–8 kHz) while the fourth depicts the centroids in the low frequency band (0–4 kHz). The left column represents the source S1 while the right column S2. It can be clearly seen in the first row of Fig. 3 that the time activity of the speech signal in the high frequency band is different from the one in the low frequency band. Furthermore, even in each subband, the time activity pattern may still vary slightly with frequency. With a multi-band multi-centroid strategy, the obtained reference (centroids) can better capture these differences, leading to better permutation alignment results.

3.4. Postprocessing with local permutation alignment

The 3-stage processing can be referred to as a global permutation alignment since the permutation is corrected by aligning the frequency bins to a global reference. In [19,20,35], it is additionally mentioned that a local permutation alignment can be employed to further enhance the permutation alignment performance. The local permutation alignment is performed at each frequency bin f so that the correlation coefficient between the current bin and a set of selected bins is maximized. This is expressed as [20]

$$\Pi_f \leftarrow \arg\max_{\Pi} \sum_{g \in \mathcal{G}(f)} \sum_{k=1}^{N} \rho\left(v_i^f, v_{i'}^g\right)\big|_{i=\Pi_f(k),\, i'=\Pi_g(k)}, \tag{18}$$

where the set $\mathcal{G}(f) = \{f - 3\Delta f : f + 3\Delta f,\ \ f/2 - \Delta f : f/2 + \Delta f,\ \ 2f - \Delta f : 2f + \Delta f\}$ contains adjacent frequencies and harmonic frequencies, and $\Delta f = f_s/Q$ with Q being the FFT size.

The local permutation alignment scheme can be used as a post-processing step of any inter-frequency dependency based permutation alignment algorithm. By fine tuning the permutation at each frequency bin, the permutation alignment performance can be improved in a certain degree especially when most frequency bins are already correctly aligned. However, employing local information only, the local permutation alignment cannot correct permutation errors effectively when a block of misalignment occurs. This statement will be proved in the experiment part. In the remaining part of the paper, this local postprocessing will be excluded from the permutation alignment algorithm unless particularly mentioned.

4. Experiment results and analysis

The proposed algorithm is an empirical method which mainly relies on the observation that the time activity of a speech signal varies with frequency and hence tries to capture this variation with a multi-band multi-centroid processing. Two parameters, the number of subbands (N_b) and the number of candidated centroids for each source (N_c), play a crucial role on the permutation alignment performance of the proposed algorithm. It may be difficult to predict the influence of the parameters on the performance of the algorithm analytically. For this reason, extensive experiments are carried out in different testing scenarios to examine how the performance varies with respect to the two parameters.

Another important issue is the robustness of a permutation alignment algorithm, i.e., it may perform well for one mixing scenario or testing file but fail in another case. Thus, the performance of the permutation alignment algorithm should be investigated with different mixing scenarios and testing files. Experiments in both simulated and real environments are conducted. The organization of the experiments is as below. After introduction of the simulation environment and the objective measures, four experiments are carried out. The first experiment investigates how the two parameters, the number of subbands N_b and the number of centroids N_c, affect the performance of the proposed algorithm. The second experiment examines the influence of the postprocessing with local permutation alignment. The third experiment compares the proposed algorithm with some existing permutation alignment algorithms with simulated data. Finally, the last experiment evaluates the performance of the considered algorithms with real recorded data.

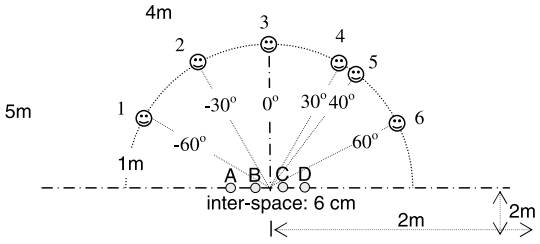


Fig. 4. Simulated room environment.

4.1. Simulation environment

In the simulated acoustical environment, several microphones and loudspeakers are placed inside a room of size 5 m × 4 m × 3 m as shown in Fig. 4. The height of all loudspeakers and microphones is 1.5 m. The room impulse response is simulated by using the image method with the reverberation time T_{60} controlled by varying the absorption coefficients [44]. For 2×2 mixtures, the microphones (B, C) and loudspeakers (3, 6) are used; for 3×3 mixtures, the microphones (A, B, C) and loudspeakers (1, 2, 3) are used; for 4×4 mixtures, the microphones (A, B, C, D) and loudspeakers (1, 2, 3, 5) are used. The testing speech signals are composed of 5 male and 5 female speakers, with each one being 10 seconds long. All possible combinations of the 10 speakers are tested, i.e., we have 45, 120 and 210 testing files for the 2×2, 3×3, and 4×4 mixtures, respectively. The sampling rate is 16 kHz. Three reverberation times are used in the experiment: 200 ms, 400 ms and 700 ms, with the simulated impulse responses for the latter two shown in Fig. 5.

The implementation detail of the blind source separation algorithm is as below. The Tukey window is used in STFT, with a shift size of 1/4 window length. The window length varies depending on the reverberation time: it being 4096 for $T_{60} = 200$ and 400 ms while being 6144 for $T_{60} = 700$ ms, under a sampling rate of 16 kHz. The instantaneous BSS is implemented by means of the Scaled Infomax [45], with an iteration number of 120. The scaling ambiguity is solved by using the Minimum Distortion Principle (6). A smoothing method proposed in [46] is applied in order to reduce spikes due to the circularity effect of the FFT.

4.2. Objective measures

With multiple testing files, it is possible to examine the mean performance and the robustness performance of the permutation alignment algorithm. The mean performance can be measured by averaging the multiple results. The robustness performance can be measured by counting the outlier results only. Both performances are evaluated from two aspects: permutation alignment error and signal-to-interference ratio (SIR).

To calculate the permutation alignment error, the correct permutation should be known, which can be calculated from the mixing and demixing network. Given the mixing matrix $\boldsymbol{H}(f)$ and the demixing matrix $\boldsymbol{W}(f)$ at each frequency bin, we consider a combined response $\boldsymbol{G}(f) = \boldsymbol{W}(f)\boldsymbol{H}(f)$. The correct permutation for the i-th channel (the i-th source) corresponds to the maximal value in the i-th row of $\boldsymbol{G}(f)$:

$$\text{perm}_i = \arg\max_j \left|\boldsymbol{G}_{ij}(f)\right|. \tag{19}$$

In cases that the mixing matrix is unknown, the correct permutation can also be estimated using the individual contributions from the sources to the microphones in a similar way.

Using the correct permutation as a reference, the accuracy of the permutation alignment can be easily calculated. For one separation task with N sources, the permutation error is defined as

$$E = \frac{1}{N}\sum_{i=1}^{N} e_i, \tag{20}$$

where e_i is the number of bins with erroneous permutation at the i-th source. For multiple testing files, the mean permutation error is defined as

$$\boxed{\text{E}_{\text{mean}} = \frac{1}{K}\sum_{k=1}^{K} E_k \times 100\%}, \tag{21}$$

where K is the total number of testing files and E_k is the permutation error for the k-th testing file. We assume that a block of permutation errors occur if the error E_k of a testing file is larger than a threshold E_{Th}, and this test file is flagged as an outlier (we use $E_{Th} = 20\%$ in this paper). The number of files with outlier results among all the testing files is used to measure the robustness against block errors. This robustness measure is defined as

$$\boxed{\text{N}_{\text{outlier}} = \frac{\text{num(outlier)}}{K} \times 100\%}, \tag{22}$$

where K is the number of testing files and num(outlier) is the number of outlier files with $E_k > E_{Th}$. A small E_{mean} as well as a small $\text{N}_{\text{outlier}}$ indicate good permutation alignment performance.

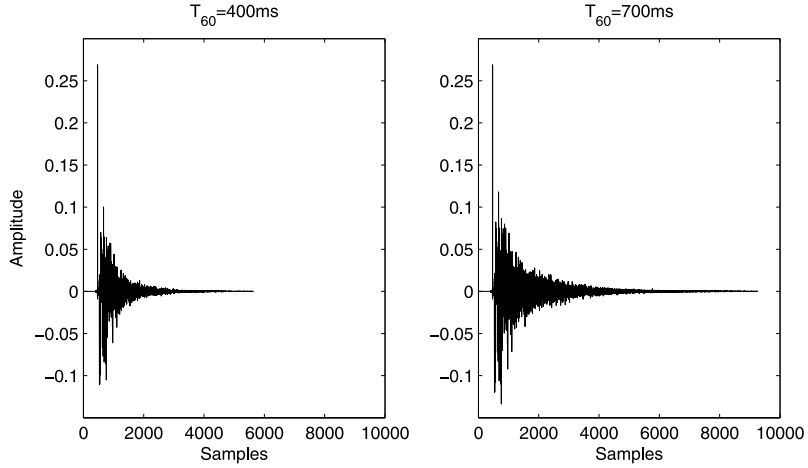


Fig. 5. Simulated room impulse responses with reverberation time of 400 ms and 700 ms, sampling rate 16 kHz.

Signal-to-interference ratio (SIR) is an objective measure to evaluate the global separation performance of a BSS algorithm. The input and output SIRs at the i-th channel are defined as

$$\mathrm{SIR}_{\mathrm{in}}^{i} = 10\log_{10}\frac{\sum_n |\sum_l h_{ii}(l)s_i(n-l)|^2}{\sum_{j\neq i}\sum_n |\sum_l h_{ij}(l)s_j(n-l)|^2}, \tag{23}$$

$$\mathrm{SIR}_{\mathrm{out}}^{i} = 10\log_{10}\frac{\sum_n |\sum_l g_{iq(i)}(l)s_{q(i)}(n-l)|^2}{\sum_{j\neq q(i)}\sum_n |\sum_l g_{ij}(l)s_j(n-l)|^2}, \tag{24}$$

where n is time index, $q(i)$ is the index of the output channel in which the i-th source appears, $h_{ij}(n)$ is an element of $\boldsymbol{H}(n)$, and $g_{ij}(n)$ is an element of the overall impulse response matrix $\boldsymbol{G}(n) = \boldsymbol{W}(n) * \boldsymbol{H}(n)$. For one separation task with N sources, an averaged SIR is calculated as

$$\mathrm{SIR}_{\mathrm{in}} = \frac{1}{N}\sum_{i=1}^{N}\mathrm{SIR}_{\mathrm{in}}^{i}, \tag{25}$$

$$\mathrm{SIR}_{\mathrm{out}} = \frac{1}{N}\sum_{i=1}^{N}\mathrm{SIR}_{\mathrm{out}}^{i}. \tag{26}$$

In our experiment, the input SIR does not vary greatly for different reverberation time, valuing around -5 dB for 4×4 mixtures, -3 dB for 3×3 mixtures, and 0 dB for 2×2 mixtures. Hence, we only compare the output SIR. For multiple testing files, the mean performance in terms of SIR is defined as

$$\boxed{\mathrm{SIR}_{\mathrm{mean}} = \frac{1}{K}\sum_{k=1}^{K}\mathrm{SIR}_{\mathrm{out}}(k)}, \tag{27}$$

where $\mathrm{SIR}_{\mathrm{out}}(k)$ is the output SIR of the k-th testing file. The robustness of the BSS algorithm can be evaluated by the averaged SIR of a set of K_{worst} testing files with worst SIR performance (we use $K_{\mathrm{worst}} = 10$ in this paper). This measure indicates the performance of the BSS algorithm in worst scenarios and is defined as

$$\boxed{\mathrm{SIR}_{\mathrm{robust}} = \frac{1}{K_{\mathrm{worst}}}\sum_{k\in K_{\mathrm{worst}}}\mathrm{SIR}_{\mathrm{out}}(k)}. \tag{28}$$

A large $\mathrm{SIR}_{\mathrm{mean}}$ as well as a large $\mathrm{SIR}_{\mathrm{robust}}$ indicate good permutation alignment performance.

We want to point out that SIR as a widely-used BSS performance measure does not evaluate the permutation alignment performance accurately since it is more sensitive to the separation result for low frequencies at which a speech signal has more energy than at high frequencies. That is why we need both measures in terms of permutation alignment error and SIR.

In short summary, four objective measures ($\mathrm{E}_{\mathrm{mean}}$, $\mathrm{N}_{\mathrm{outlier}}$, $\mathrm{SIR}_{\mathrm{mean}}$, and $\mathrm{SIR}_{\mathrm{robust}}$) are used in the experiment, aiming at a comprehensive evaluation of the permutation alignment performance. We want to mention that, apart from objective measures, subjective measures such as mean opinion score (MOS) and revised MOS (R-MOS) [47] can also be employed for the performance evaluation. However, the subjective evaluation work will be left for future research when incorporating BSS into a practical speech enhancement system.

4.3. Performance versus parameters

In this simulated experiment, the relationship between the permutation performance and the two parameters are examined in testing scenarios with different mixing conditions (3×3 and 4×4) and reverberation times (400 ms and 700 ms). N_b is chosen from the set $\{1, 2, 4, 8, 16\}$, and N_c is chosen from the set $\{1, 2, 4, 8, 16\}$. All 25 combinations of the two parameters are tested.

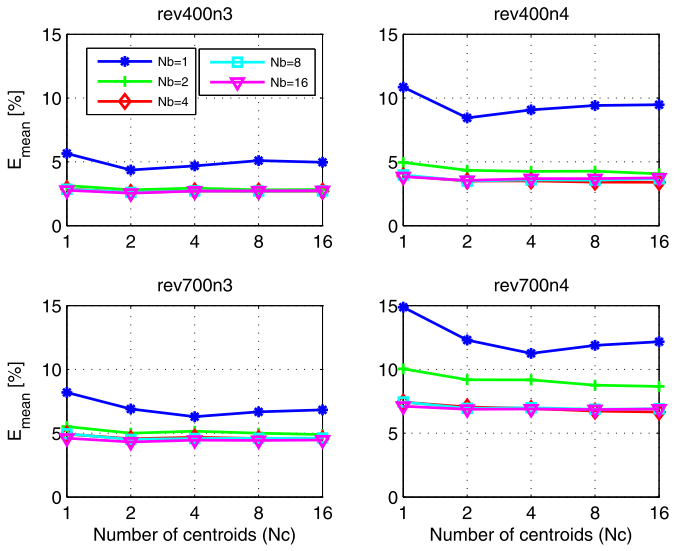


Fig. 6. $\mathrm{E}_{\mathrm{mean}}$ (mean permutation error) versus the number of subbands (N_b) and the number of centroids (N_c) in testing scenarios with different reverberation (400 ms, 700 ms) and sources (3×3, 4×4).

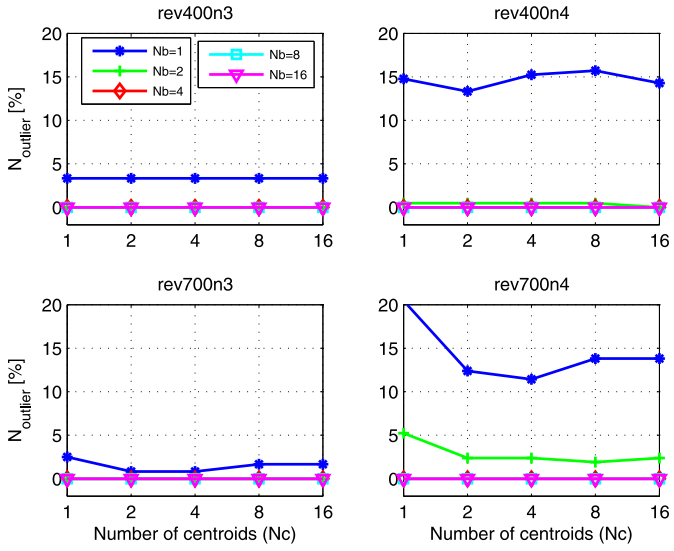


Fig. 7. $\mathrm{N}_{\mathrm{outlier}}$ (number of outlier files among all the testing files) versus the number of subbands (N_b) and the number of centroids (N_c) in testing scenarios with different reverberation (400 ms, 700 ms) and sources (3×3, 4×4).

The permutation errors in terms of $\mathrm{E}_{\mathrm{mean}}$ and $\mathrm{N}_{\mathrm{outlier}}$ obtained by the proposed algorithm are plotted in Figs. 6 and 7, respectively. In Fig. 6, each panel plots the $\mathrm{E}_{\mathrm{mean}}$ results for one testing scenario, with the horizontal axis representing N_c, the vertical axis representing $\mathrm{E}_{\mathrm{mean}}$, and each curve in the figure depicting the results for a fixed value of N_b. The following trends can be observed from the four panels in Fig. 6:

(1) $\mathrm{E}_{\mathrm{mean}}$ increases with the complexity of the testing scenario.
(2) $\mathrm{E}_{\mathrm{mean}}$ decreases when N_b is increased from 1 to 4, and only varies slightly afterwards.
(3) The influence of N_c on $\mathrm{E}_{\mathrm{mean}}$ is smaller compared to the influence of N_b.
(4) For a fixed N_b, $\mathrm{E}_{\mathrm{mean}}$ decreases when N_c grows from 1 to 2, and only varies slightly afterwards.

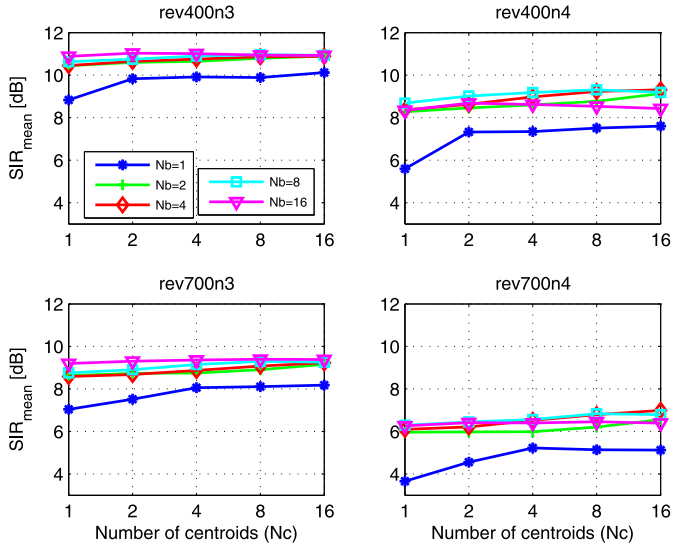


Fig. 8. $\mathrm{SIR}_{\mathrm{mean}}$ (mean SIR performance) versus the number of subbands (N_b) and the number of centroids (N_c) in testing scenarios with different reverberation (400 ms, 700 ms) and sources (3×3, 4×4).

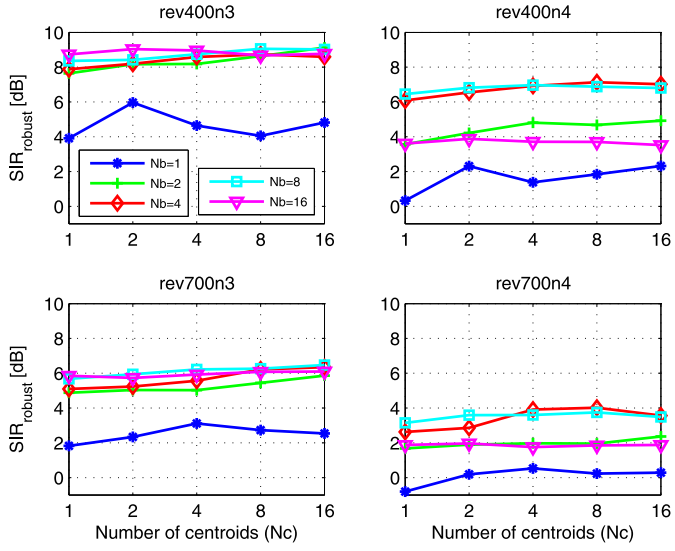


Fig. 9. $\mathrm{SIR}_{\mathrm{robust}}$ (worst SIR performance) versus the number of subbands (N_b) and the number of centroids (N_c) in testing scenarios with different reverberation (400 ms, 700 ms) and sources (3×3, 4×4).

For the robustness performance given in Fig. 7, the variation of $\mathrm{N}_{\mathrm{outlier}}$ (the number of outliers) with respect to N_b and N_c shows similar trends as $\mathrm{E}_{\mathrm{mean}}$ in Fig. 6. Especially for 4×4 mixtures, the proposed algorithm shows a large number of outliers at $N_b = 1$ while almost no outliers at $N_b \geq 4$. This demonstrates that the multiple-band processing can reduce the block permutation errors effectively.

The global separation performance in terms of $\mathrm{SIR}_{\mathrm{mean}}$ and $\mathrm{SIR}_{\mathrm{robust}}$ obtained by the proposed algorithm are given in Figs. 8 and 9, respectively. For the mean performance shown in Fig. 8, the following trends can be observed:

(1) $\mathrm{SIR}_{\mathrm{mean}}$ decreases with increasing complexity of the testing scenarios.
(2) $\mathrm{SIR}_{\mathrm{mean}}$ increases when N_b is increased from 1 to 2, and only varies slowly afterwards, with the highest SIR observed at $N_b = 8$ in almost all the testing scenarios.
(3) For a fixed N_b value, $\mathrm{SIR}_{\mathrm{mean}}$ increases when N_c is increased from 1 to 4, and only varies slowly afterwards.
(4) For almost all the testing scenarios, the highest SIR can be observed at around $N_b = 8$, $N_c = 8$. The best performance obtained by the proposed algorithm is close the benchmark for the first three scenarios (rev400n3, rev400n4 and rev700n3), and slightly worse for the most challenging scenario (rev700n4).

For the robustness performance shown in Fig. 9, the variation of $\mathrm{SIR}_{\mathrm{robust}}$ (SIR in worst scenarios) with respect to N_b and N_c shows similar trends as $\mathrm{SIR}_{\mathrm{mean}}$ in Fig. 8. It should be reminded that a smaller $\mathrm{SIR}_{\mathrm{robust}}$ indicates better robustness performance. Additionally, the following trends can be observed.

(1) The variation trend of $\mathrm{SIR}_{\mathrm{robust}}$ with respect to N_b becomes more evident than the variation trend of $\mathrm{SIR}_{\mathrm{mean}}$ in Fig. 8.
(2) In complex scenarios (rev400n4 and rev700n4), $\mathrm{SIR}_{\mathrm{robust}}$ is smaller than the one at $N_b = 4$ or 8. This phenomenon demonstrates that the proposed algorithm becomes less robust at $N_b = 16$, although a higher $\mathrm{SIR}_{\mathrm{mean}}$ is observed in Fig. 8.
(3) For a fixed N_b value, the variation of $\mathrm{SIR}_{\mathrm{robust}}$ with N_c is not evident. However, increased robustness can still be observed for N_c rising from 1 to 4.
(4) In almost all testing scenarios, the highest $\mathrm{SIR}_{\mathrm{robust}}$ can be observed at $N_b = 8$, $N_c = 8$.

Finally, with promising and consistent results observed for all the testing scenarios, two temporary conclusions can be drawn. First, the advantage of using multiple bands and multiple centroids over using single band or single centroid can be clearly observed. Especially, the multiple-band processing can increase the robustness of the algorithm significantly. Second, the proposed algorithm with the parameters N_b chosen from $\{4, 8\}$ and N_c chosen from $\{4, 8\}$ works well for all the testing cases.

4.4. Performance with postprocessing

We investigate the performance of the proposed algorithm with and without local permutation alignment postprocessing in a testing scenario with 4×4 mixtures and the reverberation time of 400 ms. For simplicity, we only check the performance of proposed algorithm for N_b chosen from $\{1, 8\}$ and N_c chosen from $\{1, 2, 4, 8\}$. The results are shown in Fig. 10 with each panel representing one of the four objective measures: $\mathrm{E}_{\mathrm{mean}}$, $\mathrm{E}_{\mathrm{robust}}$, $\mathrm{SIR}_{\mathrm{mean}}$, and $\mathrm{SIR}_{\mathrm{robust}}$. The following phenomena can be observed from Fig. 10:

(1) The local permutation alignment can only reduce the permutation error limitedly. Especially, the performance in terms of $\mathrm{N}_{\mathrm{outlier}}$ only changes slightly. This indicates that the local processing cannot correct the permutation error effectively when a block of misalignment occurs.
(2) Although with limited improvement in reducing permutation errors, the local permutation alignment can improve the SIR performance evidently. The reason for the improvement of SIR is that the SIR measure is more indicative to the separation result for low frequencies at which a speech signal has more energy than at high frequencies. In low frequency, even fine tuning at several bins can improve the SIR value remarkably. In high frequency, even a block of permutation errors does not change the SIR value greatly. However, the perceptual quality of the separated speech is severely degraded with clearly audible distortion.

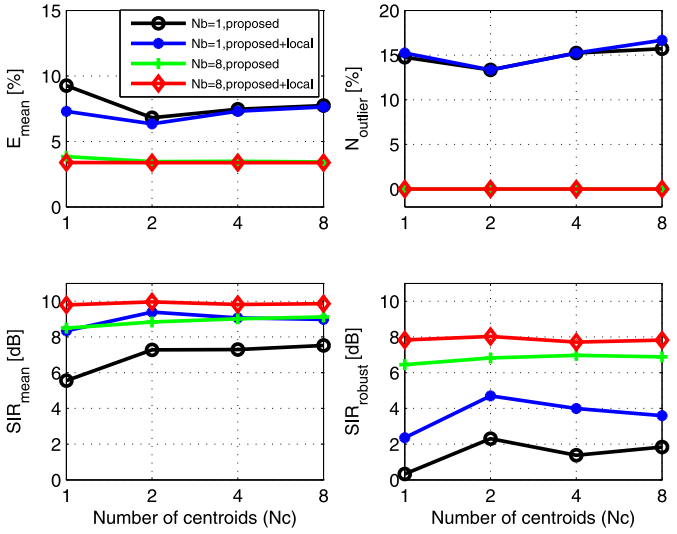


Fig. 10. The performance of the proposed algorithm with and without local permutation alignment postprocessing, for 4×4 mixture and reverberation time 400 ms.

Finally, we reach a temporary conclusion that the local permutation alignment can improve the SIR performance evidently but cannot handle the situation with blocks of permutation errors.

4.5. Performance comparison with other algorithms

In this experiment, the performance of the following algorithms is compared in the simulated environments as shown in Fig. 4. The same testing files in Section 4.3 are used, i.e., we have 45, 120 and 210 testing files for 2×2, 3×3, and 4×4 mixtures, respectively.

(1) The proposed multi-band multi-centroid algorithm (Proposed) with the parameters $N_b = 8$, $N_c = 8$.
(2) The originally proposed one-centroid clustering algorithm (Centroid-1), which is equivalent to the proposed algorithm with the parameters $N_b = 1$, $N_c = 1$ [19].
(3) The originally proposed multi-centroid clustering algorithm (Centroid-M), which is equivalent to the proposed algorithm with the parameters $N_b = 1$, $N_c = 2$ [20].
(4) The region-growing algorithm (RG), which aligns the permutation in a region-growing way based on inter-frequency dependency [22].
(5) The improved Murata algorithm (Murata), which also aligns the permutation in a region-wise way based on inter-frequency dependency [23].
(6) The improved chain-like overlapped independent vector analysis algorithm (IVA), which aims to solve the permutation ambiguity problem by conducting optimization jointly for all frequencies [28]. In the algorithm, the overlapped chain is chosen to be 1/4 size of all frequency bins and shifted by half of the chain size.
(7) The benchmark algorithm (Benchmark), which assumes perfect permutation alignment.

For the six algorithms (Proposed, RG, Centroid-1, Centroid-M, Murata, Benchmark), which do instantaneous separation and permutation alignment separately, the same instantaneous BSS algorithm, the Scaled Infomax algorithm [45], is used so that the permutation alignment performance of these algorithms can be fairly compared. For the IVA algorithm, which does instantaneous separation and permutation alignment jointly, it is difficult to incorporate a new instantaneous separation algorithm into their own processing. Therefore, the original implementation of the algorithm is used. It should be noted that the instantaneous separation performance of the Scaled Infomax algorithm is generally better than the one used in IVA as it is performed at each frequency bins independently and hence can converge to the optimal result more easily.

The separation results in terms of E_{mean}, $N_{outlier}$, SIR_{mean} and SIR_{robust} obtained by the considered algorithms are shown in the four panels of Fig. 11 respectively. In each panel, the horizontal axis represents different testing scenarios while the vertical axis represents the objective measure. As can be seen from Fig. 11 the performance of all the algorithms decreases when the complexity of the testing scenarios is increased.

The five algorithms (Proposed, RG, Centroid-1, Centroid-M, Murata), which do instantaneous separation and permutation alignment separately, obviously outperform the IVA algorithm, which does the two tasks jointly. Although IVA performs well for simple scenarios, its instantaneous separation is coupled with the permutation alignment procedure and tends to converge to a local optimum when the number of sources or the reverberation time increases, leading to block permutation errors. For example, in the simulated experiment it is a very challenging task for IVA to find the optimal solution with so many frequency bins (4096 or more) considered simultaneously. Consequently, as shown in Fig. 11, it sufferers from severe block permutation errors and shows significantly decreased SIR performance in challenging scenarios.

The five algorithms (Proposed, RG, Centroid-1, Centroid-M, Murata) perform similarly in simple scenarios (2×2). In more complicated scenarios (3×3 and 4×4), the difference of these algorithms becomes evident, ranking from good to poor as Proposed $>$ RG $>$ Centroid-M $>$ Centroid-1 $>$ Murata. As can be seen from the "E_{mean}" and "SIR_{mean}" panels, the proposed algorithm outperforms all the other algorithms with its E_{mean} and SIR_{mean} curves closest to the benchmark. Consistent results can also be observed for the robustness performance in terms of $N_{outlier}$ and SIR_{robust}. As can be seen in the "$N_{outlier}$" panel, the proposed algorithm can reduce block permutation errors (measured by the number of outlier files from all the testing files) significantly. In the "SIR_{robust}" panel for the performance in worst scenarios, the proposed algorithm performs much better than all the other algorithms.

As discussed in Section 3, the proposed multi-band multi-centroid algorithm can better capture the variation of the time–frequency activity of the speech signals and thus can improve the mean performance and the robustness significantly with respect to the Centroid-1 and Centroid-M algorithms. It is noticed that the RG algorithm also performs quit robust in all testing scenarios, although worse than the proposed algorithm. The RG and the proposed algorithm both do permutation alignment by aligning the local frequency bins to a global reference, but estimate this reference in different ways. The RG algorithm does reference update and permutation alignment simultaneously during its region-growing procedure. Although updating the reference in a region-wise way is quite robust to permutation errors at individual bins, these errors will accumulate during the region-growing procedure, leading to possible erroneous update of the reference. In contrast, the proposed algorithm estimates the reference in a clustering-based way first and then aligns the frequency bins to the reference. By doing reference estimation and permutation alignment separately, the clustering-based way is less affected by the permutation errors at individual bins. Especially, multiple-band multiple-centroid strategy improves the accuracy of the reference significantly, leading to better performance in challenging scenarios.

Finally, a temporary conclusion can be drawn from this experiment: the proposed algorithm outperforms other algorithms for all testing scenarios. Especially, it can improve the robustness performance in challenging scenarios and reduce block permutation errors efficiently.

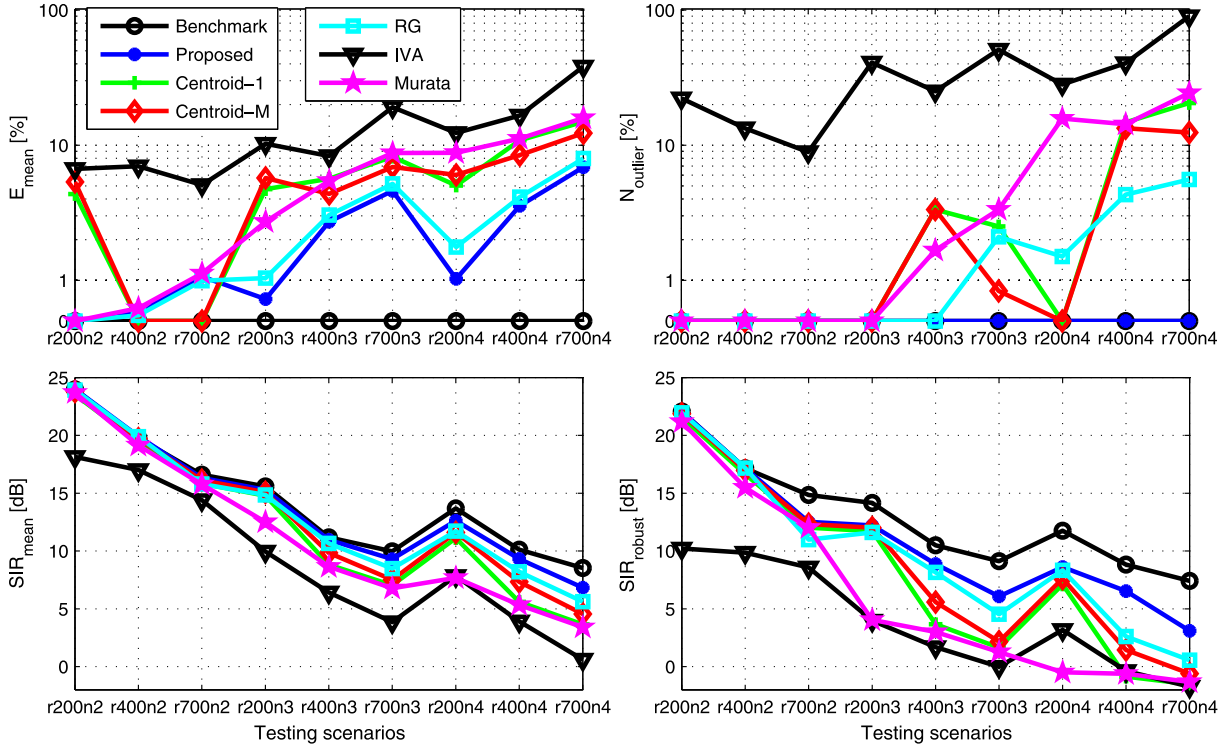


Fig. 11. Comparison of the permutation alignment algorithms in various testing scenarios.

4.6. Experiment with real recordings

In this part we evaluate the performance of the permutation alignment algorithms (Proposed, Centroid-1, Centroid-M, RG, Murata, IVA) with two sets of real recorded data. For reference, the results obtained with extra local permutation alignment are also calculated. As a benchmark, the correct permutation can be calculated by using the individual contributions from the sources to the microphones.

The first data set was downloaded from the internet.[1] It consists of 7 seconds long speech recordings sampled at 8 kHz with individual contributions from the sources to the microphones. With the reverberation time around 130 ms, the separation task of this dataset is relatively easy. The separation results are calculated for the 2×2, 3×3 and 4×4 mixing scenarios, respectively. Since only one testing file for each mixing scenario is available, the objective measure of $\mathrm{SIR}_{\mathrm{out}}$ (cf. Eq. (26)) is used. For the proposed algorithm, we choose $N_b = 4$ and $N_c = 8$ (for the sampling rate of 8 kHz). The STFT frame size is 2048. The separation results in terms of $\mathrm{SIR}_{\mathrm{out}}$ are given in Table 1. The performance of the proposed algorithm is the closest to the benchmark. Furthermore, the postprocessing with local permutation alignment can improve the SIR performance slightly.

The second data set was recorded in an office environment with a reverberation time of around 450 ms. The separation task with this dataset is relatively challenging. The geometrical configuration of the microphones and loudspeakers is similar to the one shown in Fig. 4. The main difference is that all the loudspeakers are placed 2 m away from the center of the microphone array. The same 10 speech signals used in the simulated experiment are played through each loudspeaker and recorded by all the microphones. In this experiment, we evaluate the performance of the considered algorithms with a 4×4 mixture, where all the 210 combinations of the 10 speech files are tested. For the proposed algorithm, we choose $N_b = 4$ and $N_c = 8$ (for the sampling rate of 16 kHz). The STFT frame size is 4096. The separation results in terms of $\mathrm{SIR}_{\mathrm{mean}}$ and $\mathrm{SIR}_{\mathrm{robust}}$ are given in Table 2. The performance of the proposed algorithm is the closest to the benchmark. Slight improvement by postprocessing can also be observed.[2]

The results in Table 1 are obtained in a relatively simple testing scenario and with just one testing file. Even in such a simple scenario, the proposed algorithm can still achieve slightly better performance than other algorithms. The results in Table 2 are obtained in a more challenging scenario with multiple testing files

Table 1
Comparison of the permutation alignment algorithms in terms of $\mathrm{SIR}_{\mathrm{out}}$ [dB] for the first real recorded dataset (reverberation time 130 ms).

Algorithms	2×2	3×3	4×4
Benchmark	18.80	12.87	9.81
Proposed (+local)	17.10 (18.95)	12.74 (12.74)	9.16 (9.53)
Centroid-1	13.25	11.17	2.30
Centroid-M	14.49	12.51	4.13
RG	16.95	11.30	6.91
Murata	11.75	11.21	4.40
IVA	12.46	10.98	2.99

Table 2
Comparison of the permutation alignment algorithms in terms of $\mathrm{SIR}_{\mathrm{mean}}$ and $\mathrm{SIR}_{\mathrm{robust}}$ for the second real recorded dataset (4×4, reverberation time 450 ms).

Algorithms	$\mathrm{SIR}_{\mathrm{mean}}$ [dB]	$\mathrm{SIR}_{\mathrm{robust}}$ [dB]	$\mathrm{E}_{\mathrm{mean}}$ [%]	$\mathrm{N}_{\mathrm{outlier}}$ [%]
Benchmark	8.08	7.56	0	0
Proposed (+local)	7.44 (7.87)	5.02 (5.24)	5.6 (5.8)	0 (0)
Centroid-1	4.71	0.90	12.9	14.7
Centroid-M	5.91	2.35	11.4	15.2
RG	5.71	1.51	8.4	3.8
Murata	4.72	1.22	13.2	16.7
IVA	1.56	0.06	36.0	87.6

[1] <http://www.kecl.ntt.co.jp/icl/signal/sawada/demo/bss2to4/index.html>.

[2] Matlab codes and audio demos are available at <https://sites.google.com/site/linwangsig/bss_mbmc>.

and thus are more convincing. Although the proposed algorithm exhibits just a bit higher SIR_{mean} than some existing algorithms such as RG and Centroid-M, it shows much higher SIR_{robust}. This indicates that the proposed performs more robustly than others. In terms of permutation errors, the proposed algorithm shows smallest E_{mean} among all the algorithms. In addition, as indicated by the $N_{outlier}$ value, the proposed algorithm shows no block permutation errors while other algorithms suffers from block errors more or less. The IVA shows severe block permutation errors, which lead to poor SIR performance.

Finally, we can conclude that the experimental results in real environments are consistent to those obtained in simulated environments: the proposed algorithm outperforms other algorithms for all testing cases.

5. Computational complexity analysis

In this section, we analyze the influence of the two parameters, N_b and N_c, on the computational complexity of the proposed BSS algorithm. For convenience, only multiplication operations are considered. Suppose there are N sources and N microphones, the length of the input signals is T, and the STFT frame length is $Q = 2L$ with a window shift $= L/2$. After STFT, the number of data points available for each frequency bin is approximately $B = T/\text{shift} = 2T/L$. From Figs. 1 and 2, the computation of the frequency-domain BSS is mainly composed of two parts: (1) instantaneous BSS (ICA); (2) permutation alignment. The computation cost of the first part is generally a fixed value. The computation of the second part (the proposed permutation alignment algorithm) is mainly composed of three stages:

$$C_{perm} = C_{s1} + C_{s2} + C_{s3}, \tag{29}$$

with

$$\begin{cases} C_{s1} = C_{pwr} + C_{1c\_1}, \\ C_{s2} = N_b(C_{1c\_2} + C_{mc}). \end{cases} \tag{30}$$

The computation of the first stage C_{s1} is composed of power ratio computation C_{pwr} and one centroid clustering C_{1c_1}; the computation of the second stage C_{s2} is composed of N_b groups of one centroid clustering C_{1c_2} and multi-centroid clustering C_{mc}; the computation of the third stage C_{s3} is composed of $N_b - 1$ alignment processing, which can be neglected when compared with C_{s1} and C_{s2}.

Suppose we have N_F frequency bins in each subband with $L = N_b \cdot N_F$, it follows that

$$\begin{cases} C_{pwr} = L\left(N^2 B + NB\right) \\ C_{1c\_1} = I_{1c\_1} \cdot L \cdot P_N^N \cdot C_\rho \\ C_{1c\_2} = I_{1c\_2} \cdot N_F \cdot P_N^N \cdot C_\rho \\ C_{mc} = I_{mc} \cdot \left(N \cdot C_{km} + N_F \cdot P_N^N \cdot N_c \cdot C_\rho\right), \end{cases} \tag{31}$$

where P_N^N is the permutation number of N. I_{1c_1}, I_{1c_2} are the iteration numbers of the one-centroid clustering in the two stages, respectively; I_{mc} is the iteration number of the multi-centroid clustering; $C_\rho = 3B$ is the computation cost of calculating the correlation coefficient between two time activity sequences by (8); C_{km}, the computation cost of K-means clustering with N_c centroids, is proportional to the data size and can be approximately represented as $C_{km} = I_{km} \cdot \xi \cdot B \cdot N_F \cdot N_c$, with I_{km} being the iteration number inside the K-means algorithm and ξ being a constant.

Finally, combining (29)–(31), we have

$$\begin{aligned} C_{perm} = {} & C_{pwr} + (I_{1c\_1} + I_{1c\_2}) \cdot L \cdot P_N^N \cdot C_\rho \\ & + I_{mc} \cdot L \cdot N_c \cdot \left(P_N^N \cdot C_\rho + \xi I_{km}\right). \end{aligned} \tag{32}$$

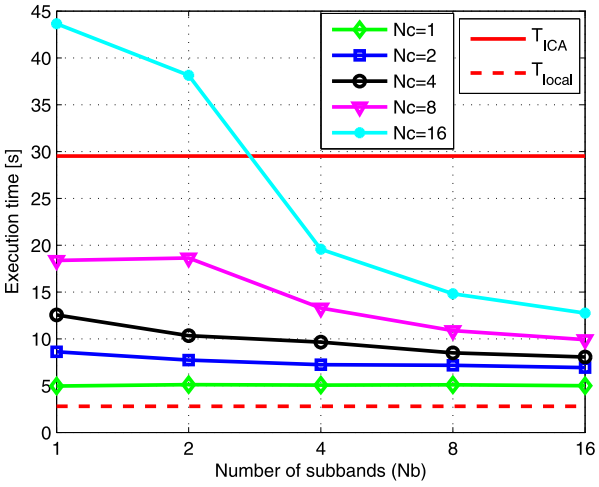


Fig. 12. Execution time of the permutation alignment algorithm versus the number of subbands (N_b) and the number of centroids (N_c) (the time for the instantaneous ICA and postprocessing with local permutation alignment are also given); Data: 4×4 mixtures of 10 s length under reverberation 400 ms and 16 kHz sampling rate.

It can be seen from (32) that the theoretically computational complexity of the multi-band multi-centroid algorithm is linearly proportional to N_c but independent of N_b, if all the values of the iteration numbers I_{1c_1}, I_{1c_2}, I_{mc} and I_{km} are fixed. However, in practice, the required iteration number for reaching convergence relates to the size of data involved, e.g., the number of frequency bins in each processing band. This implicates that the computational complexity of the permutation algorithm may decrease when N_b is increased.

Fig. 12 gives the execution time of the permutation alignment algorithm with different parameters when separating a 4×4 mixture of 10 seconds long generated under reverberation time of 400 ms and sampling rate of 16 kHz. Since ICA and permutation alignment are two essential parts of a frequency-domain blind source separation procedure, the computation time of ICA as well as the postprocessing with local permutation alignment, which are both independent of the two parameters, are also shown to give the readers a global impression of the computational complexity of the blind source separation algorithm. The program was coded in Matlab and run on Intel 64 Q8400 @ 2.66 GHz. It needs to be pointed out that the execution time of an algorithm depends on a lot of factors such as computational complexity, program structure, hardware pipeline, and thus may vary significantly for different implementations. In Fig. 12 the vertical axis denotes the execution time; the horizontal axis denotes N_b; each curve plots the execution time for a fixed value of N_c. It can be seen from Fig. 12 that for a fixed N_b the execution time increases with N_c. For a fixed small N_c value $(1, 2)$ the execution time keeps almost constant with respect to N_b. In addition, for a fixed large N_c value $(4, 8, 16)$, the execution time drops significantly with increasing N_b. This phenomenon indicates that the number of iterations (e.g., I_{1c_2}, I_{mc} and I_{km}) required for reaching convergence tends to decrease evidently with rising N_b, especially in case of large N_c values. This is an extra benefit when using a large value of N_b.

6. Conclusions

Studying frequency-domain convolutive blind source separation, this paper proposes an improved permutation alignment algorithm based on inter-frequency dependency of separated signals. The main contribution of the paper is summarized as below:

(1) A multi-band multi-centroid clustering algorithm is proposed for the permutation alignment problem. It is shown that both multi-band and multi-centroid processing is important for permutation alignment. The proposed algorithm evidently outperforms the full-band clustering algorithms. Especially, it can improve the robustness in challenging scenarios and can reduce block permutation errors effectively.
(2) Extensive experiments (with different reverberation time, number of sources, and testing files) are carried out to investigate the influence of two parameters, the number of subbands and the number of clustering-centroids, on the performance of the proposed algorithm. A robust combination of the two parameters can be easily found from the experiment results.
(3) Computational analysis demonstrates that the multi-band processing can reduce the computational cost compared to a full-band algorithm.
(4) The proposed algorithm also outperforms other existing algorithms in both terms of mean performance and robustness for all testing scenarios.

Finally, with nearly perfect permutation results in most testing scenarios, the proposed algorithm is promising for practical applications.

The current version of the proposed blind source separation algorithm only considers batch processing, which however has to bear deficiencies such as high computational cost and long algorithm delay. Extending the proposed algorithm to an online implementation and making it more suitable for real-time processing will be our future work.

Acknowledgments

The work was supported by the Postdoctoral Researcher Fellowship by the Alexander von Humboldt Foundation in Germany. The authors would like to thank Dr. Wonil Chang and Dr. ChoongHwan Choi from KAIST for fruitful discussions and sharing the IVA codes.

References

[1] A. Hyvarien, J. Karhunen, E. Oja, Independent Component Analysis, John Wiley & Sons, New York, 2001.
[2] J. Cardoso, Blind signal separation: statistical principles, Proc. IEEE 86 (10) (1998) 2009–2025.
[3] M.S. Pedersen, J. Larsen, U. Kjems, L.C. Parra, A survey of convolutive blind source separation methods, in: Springer Handbook on Speech Processing and Speech Communication, Springer, 2007, pp. 1–34.
[4] L. Wang, H. Ding, F. Yin, Combining superdirective beamforming and frequency-domain blind source separation for highly reverberant signals, EURASIP J. Audio Speech Music Process. (2010), article ID 797962, pp. 1–13.
[5] L. Wang, H. Ding, F. Yin, Target speech extraction in cocktail party by combining beamforming and blind source separation, Acoust. Aust. 39 (2) (2011) 64–68.
[6] S.C. Douglas, X. Sun, Convolutive blind separation of speech mixtures using the natural gradient, Speech Commun. 39 (1) (2003) 65–78.
[7] R. Aichner, H. Buchner, F. Yan, W. Kellermann, A real-time blind source separation scheme and its application to reverberant and noisy acoustic environments, Signal Process. 86 (6) (2006) 1260–1277.
[8] S.C. Douglas, M. Gupta, H. Sawada, S. Makino, Spatio-temporal FastICA algorithms for the blind separation of convolutive mixtures, IEEE Trans. Audio Speech Lang. Process. 15 (5) (2007) 1511–1520.
[9] P. Smaragdis, Blind separation of convolved mixtures in the frequency domain, Neurocomputing 22 (1) (1998) 21–34.
[10] H. Sawada, S. Araki, S. Makino, Frequency-domain blind source separation, in: Blind Speech Separation, Springer, 2007, pp. 47–78.
[11] S. Araki, R. Mukai, S. Makino, T. Nishikawa, H. Saruwatari, The fundamental limitation of frequency domain blind source separation for convolutive mixtures of speech, IEEE Trans. Audio Speech Lang. Process. 22 (2) (2003) 109–116.
[12] R. Prasad, H. Saruwatari, I. Shikano, Enhancement of speech signals separated from their convolutive mixture by FDICA algorithm, Digit. Signal Process. 19 (1) (2009) 127–133.
[13] R. Mazur, A. Mertins, An approach for solving the permutation problem of convolutive blind source separation based on statistical signal models, IEEE Trans. Audio Speech Lang. Process. 17 (1) (2009) 117–126.
[14] L. Parra, C. Spence, Convolutive blind separation of non-stationary sources, IEEE Trans. Audio Speech Lang. Process. 8 (3) (2000) 320–327.
[15] T. Mei, A. Mertins, F. Yin, J. Xi, J.F. Chicharo, Blind source separation for convolutive mixtures based on the joint diagonalization of power spectral density matrices, Signal Process. 88 (8) (2008) 1990–2007.
[16] C. Serviere, D.T. Pham, Permutation correction in the frequency domain in blind separation of speech mixtures, EURASIP J. Appl. Signal Process. 2006 (1) (2006) 177–193.
[17] F. Nesta, P. Svaizer, M. Omologo, Convolutive BSS of short mixtures by ICA recursively regularized across frequencies, IEEE Trans. Audio Speech Lang. Process. 19 (3) (2011) 624–639.
[18] N. Murata, S. Ikeda, A. Ziehe, An approach to blind source separation based on temporal structure of speech signals, Neurocomputing 41 (1–4) (2001) 1–24.
[19] H. Sawada, S. Araki, S. Makino, Measuring dependence of bin-wise separated signals for permutation alignment in frequency-domain BSS, in: IEEE International Symposium on Circuits and Systems, 2007, pp. 3247–3250.
[20] H. Sawada, S. Araki, S. Makino, MLSP 2007 data analysis competition: frequency-domain blind source separation for convolutive mixtures of speech/audio signals, in: IEEE Workshop on Machine Learning for Signal Processing, 2007, pp. 45–50.
[21] H. Sawada, S. Araki, S. Makino, Underdetermined convolutive blind source separation via frequency bin-wise clustering and permutation alignment, IEEE Trans. Audio Speech Lang. Process. 19 (3) (2011) 516–527.
[22] L. Wang, H. Ding, F. Yin, A region-growing permutation alignment approach in frequency-domain blind source separation of speech mixtures, IEEE Trans. Audio Speech Lang. Process. 19 (3) (2011) 549–557.
[23] L. Wang, H. Ding, F. Yin, An improved method for permutation correction in convolutive blind source separation, Arch. Acoust. 35 (4) (2010) 493–504.
[24] X. Li, T. Adali, M. Anderson, Joint blind source separation by generalized joint diagonalization of cumulant matrices, Signal Process. 91 (10) (2011) 2314–2322.
[25] M. Anderson, T. Adali, X. Li, Joint blind source separation with multivariate Gaussian model: algorithms and performance analysis, IEEE Trans. Signal Process. 60 (4) (2012) 1672–1683.
[26] T. Kim, H.T. Attias, S.Y. Lee, T.W. Lee, Blind source separation exploiting higher-order frequency dependencies, IEEE Trans. Audio Speech Lang. Process. 15 (1) (2007) 70–79.
[27] I. Lee, T. Kim, T.W. Lee, Fast fixed-point independent vector analysis algorithms for convolutive blind source separation, Signal Process. 87 (8) (2007) 1859–1871.
[28] I. Lee, G. Jang, T.W. Lee, Independent vector analysis using densities represented by chain-like overlapped cliques in graphical models for separation of convolutedly mixed signals, Electron. Lett. 45 (13) (2009) 710–711.
[29] C. Choi, W. Chang, S.Y. Lee, Blind source separation of speech and music signals using harmonic frequency dependent independent vector analysis, Electron. Lett. 48 (2) (2012) 124–125.
[30] N. Ono, Stable and fast update rules for independent vector analysis based on auxiliary function technique, in: IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2011, pp. 189–192.
[31] H. Sawada, S. Araki, R. Mukai, S. Makino, Grouping separated frequency components with estimating propagation model parameters in frequency-domain blind source separation, IEEE Trans. Audio Speech Lang. Process. 15 (5) (2007) 1592–1604.
[32] H. Saruwatari, T. Kawamura, T. Nishikawa, A. Lee, K. Shikano, Blind source separation based on a fast-convergence algorithm combining ICA and beamforming, IEEE Trans. Audio Speech Lang. Process. 14 (2) (2006) 666–678.
[33] F. Nesta, P. Svaizer, M. Omologo, Multiple TDOA estimation by using a state coherence transform for solving the permutation problem in frequency-domain BSS, in: IEEE Workshop on Machine Learning for Signal Processing, 2008, pp. 43–48.
[34] M.Z. Ikram, D.R. Morgan, Permutation inconsistency in blind speech separation: investigation and solutions, IEEE Trans. Audio Speech Lang. Process. 13 (1) (2005) 1–13.
[35] H. Sawada, R. Mukai, S. Araki, S. Makino, A robust and precise method for solving the permutation problem of frequency-domain blind source separation, IEEE Trans. Audio Speech Lang. Process. 12 (5) (2004) 530–538.
[36] F. Nesta, T.S. Wada, B. Juang, Coherent spectral estimation for a robust solution of the permutation problem, in: IEEE Workshop on Application of Signal Processing to Audio and Acoustics, 2009, pp. 1–4.
[37] Q. Liu, W. Wang, P. Jackson, Use of bimodal coherence to resolve the permutation problem in convolutive BSS, Signal Process. 92 (8) (2012) 1916–1927.
[38] Y. Liang, S. Naqvi, J. Chambers, Overcoming block permutation problem in frequency domain blind source separation when using AuxIVA algorithm, Electron. Lett. 48 (8) (2012) 460–461.
[39] E. Bingham, A. Hyvarien, A fast fixed-point algorithm for independent component analysis of complex valued signals, Int. J. Neural Syst. 10 (1) (2000) 1–8.
[40] A.J. Bell, T.J. Sejonwski, An information maximization approach to blind separation and blind deconvolution, Neural Comput. 7 (6) (1995) 1129–1159.
[41] S. Amari, A. Cichocki, H.H. Yang, A new learning algorithm for blind signal separation, in: Advances in Neural Information Processing Systems, 1996, pp. 757–763.

[42] K. Matsuoka, S. Nakashima, Minimal distortion principle for blind source separation, in: International Workshop on Independent Component Analysis and Blind Signal Separation, 2001, pp. 722–727.
[43] C.M. Bishop, Pattern Recognition and Machine Learning, Springer-Verlag, New York, 2006.
[44] J.B. Allen, D.A. Berkley, Image method for efficiently simulating small room acoustics, J. Acoust. Soc. Am. 65 (1979) 943–950.
[45] S.C. Douglas, M. Gupta, Scaled natural gradient algorithms for instantaneous and convolutive blind source separation, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2007, pp. 637–640.
[46] H. Sawada, R. Mukai, S. Kethulle, S. Araki, S. Makino, Spectral smoothing for frequency-domain blind source separation, in: International Workshop on Acoustic Echo and Noise Control, 2003, pp. 311–314.
[47] M. Viswanathan, M. Viswanathan, Measuring speech quality for text-to-speech systems: development and assessment of a modified mean opinion score (MOS) scale, Comput. Speech Lang. 19 (1) (2005) 55–83.

Lin Wang was born in Anhui, China, in 1981. He received the B.S. degree in electronic engineering from Tianjin University, China, in 2003, and the Ph.D. degree in signal processing from Dalian University of Technology, China, in 2010. Since October 2011, he has been an Alexander von Humboldt Postdoctoral Researcher in University of Oldenburg, Germany. His research interests include video and audio compression, blind source separation, and 3D audio processing.