BLIND SEPARATION OF MULTICHANNEL SPEECH MIXTURES USING PARAFAC ANALYSIS AND INTEGER LEAST SQUARES

By
Kleanthis Mokios

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
AT
TECHNICAL UNIVERSITY OF CRETE
CHANIA, GREECE
FEBRUARY 2006

Blind Separation of Multichannel Speech Mixtures Using PARAFAC Analysis and Integer Least Squares

Kleanthis N. Mokios, Nicholas D. Sidiropoulos, *Senior Member, IEEE,* and Alexandros Potamianos, *Member, IEEE*

Abstract

We propose two new low-complexity frequency-domain algorithms for blind speech separation (BSS) for unknown channel order. The proposed methods consist of two separate stages: in the first stage parallel factor analysis (PARAFAC) is employed in order to separate the speech signals, while in the second stage the task of matching the arbitrary permutations in the frequency domain is performed, via a novel integer-least-squares-based method. The proposed algorithms offer guaranteed convergence and good separation performance at considerably reduced complexity relative to previous methods, when applied to real-world speech data. Our approach reveals the much broader identifiability potential of joint-diagonalization-based BSS methods, which went unrecognized in the past. The new algorithms are compared with previous BSS methods.

Index Terms

Blind speech separation, multipath channel, nonstationary signals, parallel factor analysis, integer least squares.

I. Introduction

ONE of the key problems in speech processing for teleconferencing and mobile telephony applications is that of speaker separation from multichannel measurements. In a typical teleconferencing application, multiple microphones can be deployed in a room, each providing a different linear combination of the speaker signals. In a mobile telephony application, two microphones can be deployed, e.g., one on the headset, another on the dashboard or elsewhere in a car. Such multichannel measurements provide the opportunity for speaker separation or speaker-background noise separation, which can be crucial for intelligibility. The objective of the "blind" speech separation problem is to separate the multiple speaker signals, using only the measured microphone signals, i.e., without assuming knowledge of the mixing channels.

A significant attribute of speech signals that one can exploit in order to separate multichannel speech mixtures, is their inherent non-stationarity. This attribute, that apart from speech signals is encountered in most real world signals (e.g., biological signals), has led several researchers towards developing methods for tackling the BSS problem which rely on proper utilization of the signals' non-stationary nature. The majority of the proposed methods that exploit non-stationarity, consider the simple case of instantaneous linear mixtures [8], [11], [21]. In recent years, however, the focus of the blind source separation community has shifted towards the more general and realistic case of convolutive linear mixtures. Various approaches have been proposed, which can be divided into time-domain [12], [26], [17] and frequency-domain methods [30], [19], [18], [23], [24].

Time-domain BSS methods are applied directly to the convolutive mixture model. This approach may achieve good separation once the algorithm converges, but has the disadvantage of being computationally

The authors are with the Department of Electronic and Computer Engineering, Technical University of Crete, 73100 Chania, Greece (e-mails: kleanthis@telecom.tuc.gr, nikos@telecom.tuc.gr, potam@telecom.tuc.gr)

K. Mokios and A. Potamianos were partially supported by the IST-FP6 HIWIRE Project.

Earlier conference version of part of this work appears in Proc. ICASSP 2006, May 14-19, 2006, Toulouse, France.

expensive due to calculating many convolutions. In addition the mixing channel order need be known, which is rather impractical in real-world BSS problems.

On the other hand, frequency-domain BSS methods suggest moving to the frequency domain in order to decompose the time-domain convolutive BSS problem into multiple independent instantaneous BSS problems, one at each frequency bin. In general the frequency-domain algorithms have a simpler implementation as well as the mixing channel order need not be known, in contrast with their time domain counterparts. However, there is an inherent frequency-dependent permutation and scaling ambiguity problem in all frequency-domain BSS methods which does not exist in time-domain BSS methods.

In this paper we propose two new frequency domain BSS techniques that exploit non-stationarity of speech signals and deal with the more general case of convolutive linear mixtures. Both methods presented in this paper consist of two separate stages: the first stage performs the separation of the speech signals while the second one resolves the frequency-dependent permutation and scaling ambiguity, which is a key problem when the separation of the signals is conducted in the frequency-domain. As mentioned earlier, various frequency domain methods [30], [19], [18], [23], [24] have been proposed over the years. The methods in [30], [19], [18], however, employ a least-squares (LS) gradient descent procedure in order to separate signals while our techniques are based on the formulation of the BSS problem as a conjugate symmetric PARAFAC model that is fitted optimally, using a more efficient alternating LS algorithm that converges monotonically. The method presented in [24] is also based on an alternating LS algorithm in order to separate the speech signals, but doesn't establish the link to the conjugate-symmetric PARAFAC model. The conjugate-symmetric PARAFAC model exhibits strong identifiability properties that allows the separation of a much higher number of signals than what is suggested in any of the above methods. Regarding the frequency-dependent permutation ambiguity, e.g., in [19] the latter is dealt with by imposing a constraint on the length of the inverse-channel impulse response $\boldsymbol{W}(\tau)$, an approach that is loosely motivated. We propose an interpretable approach that follows naturally, by considering the physical properties of the system under study. Our approach is based on an integer-least-squares (ILS) formulation to adjust the arbitrary permutations, and its implementation assures that the occurrence of an error at any point of the permutation correction procedure, does not have a catastrophic impact on the quality of the final separated signals, unlike e.g. [30], [18], [23]. Finally, it should be stressed that our methods are characterized by low computational cost, unlike [19] or [24], which is a feature that is highly desired, especially in applications where real-time processing is essential. In [24] for example, the number of frequency bins used, for the method to attain good separation performance, is at least 4096 as apposed to 16 or 32 frequency bins in our methods. Thus, the alternating LS algorithm in [24] is iterated for at least 4064 times more, relative to our methods, which can be very costly computationally. To support our arguments we present numerical simulations using real speech data and compare the performance of our algorithm with the one proposed in [19][1].

Our contributions in this paper can be summarized as follows:

- Improved performance relative to the method in [19].
- Complexity reduced by 1-2 orders of magnitude relative to the method in [19], bringing execution times within range for on-line application, at least for teleconference, possibly also cellular telephony applications.
- Guaranteed convergence of overall algorithm.
- Interpretable criteria for resolving the permutation-scaling ambiguities, leading to well-known ILS problem, for which many good approximate solutions exist.
- Our approach reveals the much broader identifiability potential of joint-diagonalization-based BSS methods, which went unrecognized in the past. Prior references assumed $J \geq I$, where J is the number of microphones and I the number of speakers.
- Our permutation correction scheme prevents catastrophic errors that can happen as a result of a wrongly adjusted permutation.

[1]We downloaded the MATLAB implementation of the algorithm proposed in [19] from *http://newton.bme.columbia.edu/ lparra/publish/*.

- We used a much larger corpus, validated with real-world data in our simulations.

The paper is organized as follows: Section II states the BSS problem along with our assumptions and some preliminaries. Necessary background in PARAFAC theory is presented in section III. Section IV reviews the Trilinear Alternating Least Squares technique that is subsequently used to fit the PARAFAC model. The proposed method for resolving the permutation and scaling ambiguities is discussed in section V. Our algorithms are summarized in section VI. The results we obtained after application of our algorithms to real-world speech data are presented in section VII, whereas our conclusions and final remarks are presented in section VIII.

Some notation conventions that will be used in this paper follow.

$*$	Convolution operator.
$(.)^*$	Complex conjugate.
$E[.]$	Expectation operator.
$\lfloor x \rfloor$	Integer floor of x.
$\boldsymbol{a}_i$	ith column of $\boldsymbol{A}$.
$\boldsymbol{A}^T$	Transpose of $\boldsymbol{A}$.
$\boldsymbol{A}^H$	Conjugate transpose of $\boldsymbol{A}$.
$\boldsymbol{A}^\dagger$	Moore-Penrose pseudo-inverse of $\boldsymbol{A}$.
$\mathcal{D}_p\{\ \boldsymbol{A}\}$	Diagonal matrix constructed from the pth row of $\boldsymbol{A}$.
$\boldsymbol{A} \odot \boldsymbol{B}$	Khatri-Rao (column-wise Kronecker) product of $\boldsymbol{A}, \boldsymbol{B}$.
$diag\{\boldsymbol{x}\}$	Forms a diagonal matrix from vector $\boldsymbol{x}$.
$\|\|.\|\|_F$	Frobenius norm.
$\|\|.\|\|_2$	Euclidean vector norm.

II. Blind Speech Separation

A. Data Model and Problem Statement

Assume I mutually uncorrelated (not necessarily statistically independent) speaker signals $\boldsymbol{s}(t) = [s_1(t), \ldots, s_I(t)]^T$. These signals are convolved and mixed in a linear medium leading to J microphone signals $\boldsymbol{x}(t) = [x_1(t), \ldots, x_J(t)]^T$. We consider the following I-speaker J-microphone multi-input multi-output (MIMO) linear model for the received signal for the convolutive mixing problem

$$\boldsymbol{x}(t) = \boldsymbol{A} * \boldsymbol{s}(t) + \boldsymbol{n}(t) = \sum_{\tau=0}^{L} \boldsymbol{A}(\tau)\boldsymbol{s}(t-\tau) + \boldsymbol{n}(t),\ t = 1, \ldots, N \tag{1}$$

where $\boldsymbol{A}(\tau) = [\boldsymbol{\alpha}_1(\tau), \ldots, \boldsymbol{\alpha}_I(\tau)] \in \mathbb{R}^{J \times I}$ for $\tau = 0, \ldots, L$ is the mixing impulse response matrix, $\boldsymbol{\alpha}_i(\tau) = [\alpha_{1,i}(\tau), \ldots, \alpha_{J,i}(\tau)]^T \in \mathbb{R}^{J \times 1}$ is the spatial signature of the ith speaker for lag τ, $\boldsymbol{n}(t) = [n_1(t), \ldots, n_J(t)]^T$ is the additive noise vector, L is the maximum (unknown) channel length and N is the number of snapshots. The objective of the blind separation problem is to estimate the inverse-channel impulse response matrix $\boldsymbol{W}(\tau)$ from the observed signals $\boldsymbol{x}(t)$, since

$$\boldsymbol{s}(t) = \boldsymbol{W} * \boldsymbol{x}(t) \tag{2}$$

Following, e.g. [19], one approach towards solving the problem is to transform it into the frequency domain and to solve the joint separation problem across frequency bins.

Recall the well known property of the Discrete Fourier Transform (DFT) that allows us to express circular convolutions as products. In (1), however, we assumed a linear convolution. A linear convolution can be approximated by a circular convolution if the size T of the DFT's frame is much larger than the length of the convolution sum. In such a case we can write approximately[2]

$$\boldsymbol{x}(f,t) \approx \boldsymbol{A}(f)\boldsymbol{s}(f,t) + \boldsymbol{n}(f,t), for L \ll T \tag{3}$$

[2]For simplicity, we use the same symbol to denote time-domain and frequency-domain representation of a signal or a filter, depending on its argument.

where $\boldsymbol{x}(f,t) = \sum_{\tau=0}^{T-1} \boldsymbol{x}(t+\tau)e^{\frac{-i2\pi f\tau}{T}}$ is the DFT of the frame of size T starting at t, $[\boldsymbol{x}(t),\ldots,\boldsymbol{x}(t+T-1)]$. Accordingly $\boldsymbol{s}(f,t) = \sum_{\tau=0}^{T-1} \boldsymbol{s}(t+\tau)e^{\frac{-i2\pi f\tau}{T}}$ and $\boldsymbol{A}(f) = \sum_{\tau=0}^{T-1} \boldsymbol{A}(\tau)e^{\frac{-i2\pi f\tau}{T}} = \sum_{\tau=0}^{L} \boldsymbol{A}(\tau)e^{\frac{-i2\pi f\tau}{T}}$, because $\boldsymbol{A}(\tau) = \boldsymbol{0}^{J\times I}$ for $\tau > L$. The ith column of $\boldsymbol{A}(f)$ represents now the spatial signature of the ith speaker in the frequency domain, at frequency f.

Before proceeding further, we list our main assumptions:

Assumption 2.1: The speaker signals $\boldsymbol{s}(t)$ are zero mean, second-order quasi-stationary signals; i.e. the variances of the signals are slowly varying with time such that over short time intervals they can be assumed approximately stationary.

Assumption 2.2: The number of speakers is known.

Assumption 2.3: The contribution of the noise term $\boldsymbol{n}(t)$ is negligible compared to the contribution of the speaker signals. In our context, this is an accurate approximation in many real-world cases. When noise is not negligible, its power can be estimated from silence periods and subtracted from correlation matrix estimates, which is the only place where noise comes into play in our algorithms.

B. Direct Channel Estimation versus Inverse Channel Estimation

•Direct channel estimation (DCE) approach: Equation (3) dictates that the BSS problem could have been solved, had we had in our disposal the frequency domain counterparts of the multipath channel matrices $\boldsymbol{A}(\tau)$, that is $\boldsymbol{A}(f)$, for all frequency bins. Provided that $rank[\boldsymbol{A}(f)] = I$, by taking the left Moore-Penrose pseudo-inverse of each frequency's corresponding matrix $\boldsymbol{A}(f)$, and applying the Inverse DFT to the collection of the acquired pseudo-inverses $\{\boldsymbol{A}^{\dagger}(f), f = 0,\ldots,T-1\}$, we could determine estimates of the inverse-channel matrices $\{\boldsymbol{W}(\tau), \tau = 0,\ldots,T-1\}$, and subsequently estimate the speaker signals via (2). The same could be achieved by computing $\hat{\boldsymbol{s}}(f,t) \approx \boldsymbol{A}^{\dagger}(f)\boldsymbol{x}(f,t)$ for all frequency bins, and applying Inverse DFT to $\{\hat{\boldsymbol{s}}(f,t), f = 0,\ldots,T-1\}$. When $rank[\boldsymbol{A}(f)] < I$ (in particular, when $J < I$) perfect separation is not possible; substantial reduction of crosstalk is still possible, however, using matched filtering to the columns of $\boldsymbol{A}(f)$, or more sophisticated array processing methods, like Capon beamforming. Therefore, the blind speech separation problem boils down to the problem of estimating the matrices $\{\boldsymbol{A}(f), f = 0,\ldots,T-1\}$.

Let us focus on a time interval over which the measured signals can be assumed stationary (see assumption 2.1). Furthermore we consider one specific frequency bin f, ignoring for the time being the rest $T-1$ frequencies. The autocorrelation matrix of the vector of microphone outputs at frequency f is then

$$\boldsymbol{R}_x(f,t) = E[\boldsymbol{x}(f,t)\boldsymbol{x}^H(f,t)] \approx \boldsymbol{A}(f)E[\boldsymbol{s}(f,t)\boldsymbol{s}^H(f,t)]\boldsymbol{A}^H(f) = \boldsymbol{A}(f)\boldsymbol{D}_s(f,t)\boldsymbol{A}^H(f) \tag{4}$$

Since we assume mutually uncorrelated speaker signals we postulate diagonal autocorrelation matrix $\boldsymbol{D}_s(f,t)$, while by assumption 2.3 the autocorrelation matrix of the noise vector $\boldsymbol{n}(f,t)$ has been neglected. As we will see in Section III, proper processing of the autocorrelation data (4) via PARAFAC analysis, for all frequency bins and all intervals over which the measured signals are assumed stationary, enables us to specify the matrices $\{\boldsymbol{A}(f), f = 0,\ldots,T-1\}$ up to an individual permutation and scaling of their columns (spatial signatures), that vary arbitrarily from one frequency bin to another.

•Inverse channel estimation (ICE) approach: Instead of first estimating and then inverting the forward-channel matrices $\{\boldsymbol{A}(f), f = 0,\ldots,T-1\}$, we can directly estimate the inverse-channel matrices in the frequency domain, $\{\boldsymbol{W}(f), f = 0,\ldots,T-1\}$, perform the Inverse DFT to obtain $\{\boldsymbol{W}(\tau), \tau = 0,\ldots,T-1\}$ and subsequently estimate the speaker signals via (2), or, again we can compute $\hat{\boldsymbol{s}}(f,t) \approx \boldsymbol{W}(f)\boldsymbol{x}(f,t)$ for all frequency bins, and then apply Inverse DFT to $\{\hat{\boldsymbol{s}}(f,t), f = 0,\ldots,T-1\}$.

Assume, for the moment, that the number of microphones equals the number of the speakers $(J = I)$ and $rank\{\boldsymbol{A}(f)\} = I$. In this case, the inverse of the autocorrelation matrix $\boldsymbol{R}_x(f,t)$ in (4) can be written as

$$\boldsymbol{R}_x^{-1}(f,t) = [\boldsymbol{A}(f)\boldsymbol{D}_s(f,t)\boldsymbol{A}^H(f)]^{-1} = [\boldsymbol{A}^{-1}(f)]^H\boldsymbol{D}_s^{-1}(f,t)\boldsymbol{A}^{-1}(f) = \boldsymbol{W}^H(f)\boldsymbol{D}_s^{-1}(f,t)\boldsymbol{W}(f) \tag{5}$$

where the last equality follows from the fact that the inverse-channel matrix $\boldsymbol{W}(f)$ equals the inverse of the forward-channel matrix $\boldsymbol{A}^{-1}(f)$, in this case. Employing parallel factor analysis of the "inverse" autocorrelation data (5), for all frequency bins and all intervals over which the speaker signals are assumed stationary, we may obtain the beamforming matrices $\{\boldsymbol{W}(f), f = 0, \ldots, T-1\}$, up to a frequency-dependent scaling and permutation ambiguity of their rows.

In the case of fewer speakers than microphones $(J > I)$, however, the inversion of the autocorrelation matrix $R_x(f,t)$ cannot be performed, and therefore PARAFAC analysis cannot be applied. In order to deal with this problem, we reduce our model for the received signal back to the $J = I$ case, by resorting to an appropriate dimension-reducing prefiltering of the data in (3). The following paragraphs describe how this is done.

Consider one specific frequency bin f, ignoring for the time being the rest $T-1$ frequencies. Let $N_T = \lfloor \frac{N}{T} \rfloor$ and $\boldsymbol{X}(f) = [\boldsymbol{x}(f,0), \boldsymbol{x}(f,T), \ldots, \boldsymbol{x}(f,T(N_T-1))]$, then from (3), the $J \times N_T$ matrix $\boldsymbol{X}(f)$ has the following factorization

$$\boldsymbol{X}(f) = \boldsymbol{A}(f)\boldsymbol{S}(f) \tag{6}$$

where $\boldsymbol{S}(f) = [\boldsymbol{s}(f,0), \boldsymbol{s}(f,T), \ldots, \boldsymbol{s}(f,T(N_T-1))] \in \mathbb{C}^{I \times N_T}$, and the noise term has been neglected in accordance with assumption 2.3. We choose $\boldsymbol{F}(f)$ to be the matrix whose columns are the I dominant eigenvectors which correspond to the eigenvalue decomposition (ED) of the autocorrelation matrix $\bar{\boldsymbol{R}}_x(f) = \frac{1}{N_T}\boldsymbol{X}(f)\boldsymbol{X}^H(f)$. It is known that $\boldsymbol{F}(f)$ is an asymptotically unbiased estimate of $\boldsymbol{U}_A(f)$ where $\boldsymbol{A}(f) = \boldsymbol{U}_A(f)\boldsymbol{\Sigma}_A(f)\boldsymbol{V}_A(f)$ is the economy-size singular value decomposition (SVD) of $\boldsymbol{A}(f)$.

Let us now consider a prefiltered version of the data in (6), using the matrix $\boldsymbol{F}^H(f)$ as the filtering matrix, i.e.,

$$\tilde{\boldsymbol{X}}(f) \triangleq \boldsymbol{F}^H(f)\boldsymbol{X}(f) = \tilde{\boldsymbol{A}}(f)\boldsymbol{S}(f) \tag{7}$$

with $\tilde{\boldsymbol{A}}(f) \triangleq \boldsymbol{F}^H(f)\boldsymbol{A}(f) \in \mathbb{C}^{I \times I}$. By properties of the SVD, $\tilde{\boldsymbol{A}}(f)$ is a nonsingular $I \times I$ matrix. Hence, the autocorrelation matrix $\tilde{\boldsymbol{R}}_x(f)$ that corresponds to the prefiltered data model in (7), can be inverted to yield the following factorization

$$\tilde{\boldsymbol{R}}_x^{-1}(f,t) = [\tilde{\boldsymbol{A}}(f)\boldsymbol{D}_s(f,t)\tilde{\boldsymbol{A}}^H(f)]^{-1} = [\tilde{\boldsymbol{A}}^{-1}(f)]^H\boldsymbol{D}_s^{-1}(f,t)\tilde{\boldsymbol{A}}^{-1}(f) = \tilde{\boldsymbol{W}}^H(f)\boldsymbol{D}_s^{-1}(f,t)\tilde{\boldsymbol{W}}(f) \tag{8}$$

with $\tilde{\boldsymbol{W}}(f) = \tilde{\boldsymbol{A}}^{-1}(f)$. Employing parallel factor analysis of the "inverse" autocorrelation data (8), for all frequency bins and all intervals over which the speaker signals are assumed stationary, we may obtain the matrices $\{\tilde{\boldsymbol{W}}(f), f = 0, \ldots, T-1\}$, up to a frequency-dependent scaling and permutation ambiguity of their rows. In this case, the beamforming matrices $\{\boldsymbol{W}(f), f = 0, \ldots, T-1\}$ can be written as

$$\boldsymbol{W}(f) = \tilde{\boldsymbol{W}}(f)\boldsymbol{F}^H(f) \tag{9}$$

Note that the ICE approach cannot be employed when we have fewer microphones than sensors $(J < I)$. In this case we have to resort to the DCE approach.

Of course, the beamforming matrices $\{\boldsymbol{W}(f), f = 0, \ldots, T-1\}$ in the ICE approach, just as the acquired mixing matrices $\{\boldsymbol{A}(f), f = 0, \ldots, T-1\}$ in the DCE approach, are specified up to a frequency-dependent individual permutation and scaling of their columns. This constitutes a serious problem since only consistent permutations and scaling across frequencies will correctly reconstruct the source signals. Hence a second major problem arises, that of resolving the frequency-dependent permutation and scaling issue. The following section reviews the basics of PARAFAC analysis, and shows how it is employed on the BSS problem, via our two approaches.

III. Parallel Factor Analysis

The present section is devoted to the presentation of the necessary background in PARAFAC theory, and also shows how blind speech separation comes under this category of problems. We only consider the model in (4), since the analysis for the model of (8) follows by substituting $\tilde{\boldsymbol{R}}_x^{-1}(f,t), \tilde{\boldsymbol{W}}^H(f)$ and $\boldsymbol{D}_s^{-1}(f,t)$ for $\boldsymbol{R}_x(f,t), \boldsymbol{A}(f)$ and $\boldsymbol{D}_s(f,t)$, respectively.

A. PARAFAC Model

Let us divide the whole data block of N snapshots into P sub-blocks, so that each sub-block contains $N_P = \lfloor \frac{N}{P} \rfloor$ snapshots. We assume that each sub-block corresponds to a time interval over which the speaker signals are stationary, in accordance with assumption 2.1. Under this framework, the measured snapshots within any pth sub-block correspond to the following autocorrelation matrix

$$\boldsymbol{R}_x(f,t_p) = \boldsymbol{A}(f)\boldsymbol{D}_s(f,t_p)\boldsymbol{A}^H(f), f = 0,\ldots,T-1 \tag{10}$$

where $\boldsymbol{D}_s(f,t_p)$ is the diagonal autocorrelation matrix of the speaker signals in the pth sub-block. Using all P sub-blocks, we will have P different autocorrelation matrices $\{\boldsymbol{R}_x(f,t_1),\ldots,\boldsymbol{R}_x(f,t_P)\}$ for each frequency. We observe that for each frequency, these matrices differ from each other only because the source signal autocorrelation matrices $\boldsymbol{D}_s(f,t_p)$ differ from one sub-block to another.

Let us stack the P matrices $\{\boldsymbol{R}_x(f,t_p), p = 1,\ldots,P\}$ together to form a three-way array $\underline{\boldsymbol{R}}_x(f)$. The (j,l,p)th element of such an array can be written as

$$r_{j,l,p}(f) \triangleq [\underline{\boldsymbol{R}}_x(f)]_{j,l,p} = \sum_{i=1}^{I} \alpha_{j,i}(f) v_i(f,p) \alpha_{l,i}^*, f = 0,\ldots,T-1 \tag{11}$$

where $v_i(f,p) \triangleq [\boldsymbol{D}_s(f,t_p)]_{i,i}$ is the power-spectral-density of the ith source in the pth sub-block. Defining the matrices $\boldsymbol{P}(f) \in \mathbb{C}^{P\times I}, f = 0,\ldots,T-1$ as

$$\boldsymbol{P}(f) \triangleq \begin{bmatrix} v_1(f,1) & \ldots & v_I(f,1) \\ \vdots & \ddots & \vdots \\ v_1(f,P) & \ldots & v_I(f,P) \end{bmatrix} \tag{12}$$

we can write the following relationship between $\boldsymbol{D}_s(f,t_p)$ and $\boldsymbol{P}(f)$

$$\boldsymbol{D}_s(f,t_p) = \mathcal{D}_p\{\boldsymbol{P}(f)\} \tag{13}$$

for all $p = 1,\ldots,P$ and all $f = 1,\ldots,T$.

Equation (11) implies that $r_{j,l,p}(f)$ is a sum of rank-1triple products; this equation is known as (conjugate-symmetric) *parallel factor* (PARAFAC) analysis of $r_{j,l,p}$ [9], [13], [28], [29]. If I is sufficiently small (11) represents a low-rank decomposition of $\underline{\boldsymbol{R}}_x(f)$. Therefore, the problem of estimating the matrix $\boldsymbol{A}(f)$ for a specific frequency f can be reformulated as the problem of low-rank decomposition of the three-way autocorrelation array $\underline{\boldsymbol{R}}_x(f)$. By solving a similar problem separately for every frequency we obtain the entire collection of the frequency-domain mixing matrices $\{\boldsymbol{A}(f), f = 0,\ldots,T-1\}$, up to certain inherent indeterminacies, discussed next.

B. Identifiability

By identifiability we mean the uniqueness (up to inherently unresolvable source permutation and scale ambiguities) of all speaker spatial signatures at a given frequency, given the exact frequency-domain autocorrelation data at that frequency.

In this subsection, we briefly review identifiability properties of the PARAFAC-model based spatial signature estimation. Towards this end, we discuss conditions under which the trilinear decomposition of $\underline{\boldsymbol{R}}_x(f)$ is unique. Identifiability conditions on the number of sub-blocks and the number of microphones

are presented. In the following we consider the frequency index fixed, and omit it for convenience since the results presented do not depend on the frequency f.

We start with the definition of the Kruskal rank of a matrix [13].

Definition: The Kruskal rank (or k-rank) of a matrix $\boldsymbol{C}$ is $k_{\boldsymbol{C}}$ if and only if every $k_{\boldsymbol{C}}$ columns of $\boldsymbol{C}$ are linearly independent, and either $\boldsymbol{C}$ has $k_{\boldsymbol{C}}$ columns or $\boldsymbol{C}$ contains a set of $k_{\boldsymbol{C}}+1$ linearly dependent columns. Note that k-rank is always less than or equal to the conventional matrix rank. It can be easily checked that if $\boldsymbol{C}$ is full column rank, then it is also full k-rank.

Using (13), we can rewrite (10) as

$$\boldsymbol{R}_x(t_p) = \boldsymbol{A}\mathcal{D}_p\{\boldsymbol{P}\}\boldsymbol{A}^H, p = 1, \ldots, P \tag{14}$$

To establish identifiability, we have to obtain under which conditions matrices $\boldsymbol{P}$ and $\boldsymbol{A}$ are the unique (up to the scaling and permutation ambiguities) matrices that give rise to the data $\{\boldsymbol{R}_x(t_p), p = 1, \ldots, P\}$ of (14). In the context of our present application, which involves a conjugate-symmetric PARAFAC model the following theorem follows from [13] (also cf. [25]).

Theorem 1: Consider the set of matrices (14). For $I > 1$, if

$$k_{\boldsymbol{A}} + k_{\boldsymbol{P}} + k_{\boldsymbol{A}^*} = 2k_{\boldsymbol{A}} + k_{\boldsymbol{P}} \geq 2I + 2 \tag{15}$$

then $\boldsymbol{A}$ and $\boldsymbol{P}$ are unique up to inherently unresolvable permutation and scaling of columns, i.e. if there exists any other pair $\{\bar{\boldsymbol{A}}, \bar{\boldsymbol{P}}\}$ which satisfies (15), then this pair is related to the pair $\{\boldsymbol{A}, \boldsymbol{P}\}$ via

$$\bar{\boldsymbol{A}} = \boldsymbol{A}\boldsymbol{\Pi}\boldsymbol{\Delta}_1, \bar{\boldsymbol{P}} = \boldsymbol{P}\boldsymbol{\Pi}\boldsymbol{\Delta}_2 \tag{16}$$

where $\boldsymbol{\Pi}$ is a permutation matrix, and $\boldsymbol{\Delta}_1, \boldsymbol{\Delta}_2$ are diagonal scaling matrices satisfying

$$\boldsymbol{\Delta}_1\boldsymbol{\Delta}_1^*\boldsymbol{\Delta}_2 = \boldsymbol{I} \tag{17}$$

For $I = 1$, $\boldsymbol{A}$ and $\boldsymbol{P}$ are always unique, irrespective of (15).

It is worth noting that condition (15) is sufficient but not necessary for identifiability [2]. For $I > 1$ the condition $k_{\boldsymbol{P}} \geq 2$ becomes necessary [10]. In terms of the number of sub-blocks the latter condition requires that $P \geq 2$.

Alternative uniqueness conditions can be derived if random component matrices are considered, giving rise to the concept of almost sure-identifiability. The best known result on almost sure-identifiability conditions for the conjugate-symmetric PARAFAC model has been established in [31] and is presented in the next theorem

Theorem 2: Suppose that the elements of $\boldsymbol{A}$ and $\boldsymbol{P}$ are drawn from a jointly continuous distribution. If

$$\frac{I(I-1)}{2} \leq \frac{J(J-1)}{4}\left[\frac{J(J-1)}{2}+1\right] - \binom{J}{4} 1_{\{J\geq 4\}} \tag{18}$$

where $1_{\{J\geq 4\}} = \begin{cases} 0 & \text{if } J < 4 \\ 1 & \text{if } J \geq 4 \end{cases}$, then for $I \leq 30$, $\boldsymbol{A}$ and $\boldsymbol{P}$ are almost surely unique up to inherently unresolvable permutation and scaling of columns.

In our context J corresponds to the number of microphones we use while I corresponds to the number of speakers that can be separated. In Table I below, we give for values $J = 2, \ldots, 8$ the upper bound for I, according to the formula given in (18). Observe that 6 microphones it is possible to estimate the steering vectors of up to 15 speakers, whereas with 8 microphones it is possible to estimate the steering vectors of up to 26 speakers - more than 3 times the number of microphones, in theory.

Note that the formula in (18) provides theoretical bounds on the number of speaker signals that can be separated, relying on the link to the conjugate symmetric PARAFAC model established in the previous subsection. The ICE approach, however, requires the number of speakers to be at most equal to the number of microphones used. Thus, practically in the ICE approach, the maximum number of speakers that can be separated, cannot surpass the number of microphones used to achieve this separation, regardless of the bounds resulting from (18) and shown in Table I.

TABLE I

VALUES OF J AND UPPER BOUNDS FOR I, WHICH FOLLOW FROM (18)

J	2	3	4	5	6	7	8
Upper bound for I	2	4	6	10	15	20	26

IV. TRILINEAR ALTERNATING LEAST SQUARES ESTIMATOR

We will now review the Trilinear Alternating Least Squares (TALS) technique [10], [28], [29] for fitting the PARAFAC model of Section III. Again, we solely consider the PARAFAC model of the DCE approach (5), since by letting $\tilde{\boldsymbol{x}}(f,t), \tilde{\boldsymbol{R}}_x^{-1}(f,t), \tilde{\boldsymbol{W}}^H(f)$ and $\boldsymbol{D}_s^{-1}(f,t)$ stand in for $\boldsymbol{x}(f,t), \boldsymbol{R}_x(f,t), \boldsymbol{A}(f)$ and $\boldsymbol{D}_s(f,t)$ respectively, the following analysis is identical for the model of the ICE approach. TALS will be subsequently used to estimate the matrices $\boldsymbol{A}(f)$ and $\tilde{\boldsymbol{W}}^H(f)$ up to a frequency-dependent permutation and scaling ambiguity. Exactly how these ambiguities are resolved (to yield our final separation solution), will be discussed in section V.

A. Preliminaries

In practice, the exact autocorrelation matrices $\boldsymbol{R}_x(f,t_p)$ are unavailable but can be estimated from the array snapshots $\boldsymbol{x}(t), t = 1,\ldots,N$. If we define $K = \lfloor \frac{N_P}{T} \rfloor$, the sample autocorrelation matrix estimates are given by

$$\hat{\boldsymbol{R}}_x(f,t_p) = \frac{1}{K}\sum_{k=0}^{K-1} \boldsymbol{x}(f,t_p+kT)\boldsymbol{x}^H(f,t_p+kT), p = 1,\ldots,P \tag{19}$$

We can choose t_p such that we have non-overlapping times for $\hat{\boldsymbol{R}}_x(f,t_p)$, i.e. $t_p = pTK$, but overlapping averaging times can be chosen, as well. The matrices $\hat{\boldsymbol{R}}_x(f,t_p), p = 1,\ldots,P$ can be used to form a sample three-way autocorrelation array denoted as $\underline{\hat{\boldsymbol{R}}}_x(f)$.

In order to explain TALS, we will need to define "slices" of the three-way arrays $\underline{\boldsymbol{R}}_x(f)$ and $\underline{\hat{\boldsymbol{R}}}_x(f)$ along different dimensions [29]. Towards this end, let us define the "slice" matrices[3]

$$\boldsymbol{R}_\alpha^{(j)}(f) \triangleq [r_{j,:,:}(f)], \boldsymbol{R}_b^{(l)}(f) \triangleq [r_{:,l,:}(f)], \boldsymbol{R}_c^{(p)}(f) \triangleq [r_{:,:,p}(f)] \tag{20}$$

$$\hat{\boldsymbol{R}}_\alpha^{(j)}(f) \triangleq [\hat{r}_{j,:,:}(f)], \hat{\boldsymbol{R}}_b^{(l)}(f) \triangleq [\hat{r}_{:,l,:}(f)], \hat{\boldsymbol{R}}_c^{(p)}(f) \triangleq [\hat{r}_{:,:,p}(f)] \tag{21}$$

where $j,l = 1,\ldots,J$; $p = 1,\ldots,P$; $r_{j,l,p}(f) \triangleq [\underline{\boldsymbol{R}}_x(f)]_{j,l,p}$; and $\hat{r}_{j,l,p}(f) \triangleq [\underline{\hat{\boldsymbol{R}}}_x(f)]_{j,l,p}$.

For the sake of convenience, let us introduce $\boldsymbol{B}(f) \triangleq \boldsymbol{A}^H(f)$. Define the matrices

$$\boldsymbol{R}_\alpha(f) \triangleq \begin{bmatrix} \boldsymbol{R}_\alpha^{(1)}(f) \\ \boldsymbol{R}_\alpha^{(2)}(f) \\ \vdots \\ \boldsymbol{R}_\alpha^{(J)}(f) \end{bmatrix} = [\boldsymbol{P}(f) \odot \boldsymbol{A}(f)]\boldsymbol{B}(f) \tag{22}$$

$$\boldsymbol{R}_b(f) \triangleq \begin{bmatrix} \boldsymbol{R}_b^{(1)}(f) \\ \boldsymbol{R}_b^{(2)}(f) \\ \vdots \\ \boldsymbol{R}_b^{(J)}(f) \end{bmatrix} = [\boldsymbol{B}^T(f) \odot \boldsymbol{P}(f)]\boldsymbol{A}^T(f) \tag{23}$$

[3]We use MATLAB notation.

$$\boldsymbol{R}_c(f) \triangleq \begin{bmatrix} \boldsymbol{R}_c^{(1)}(f) \\ \boldsymbol{R}_c^{(2)}(f) \\ \vdots \\ \boldsymbol{R}_c^{(P)}(f) \end{bmatrix} = [\boldsymbol{A}(f) \odot \boldsymbol{B}^T(f)]\boldsymbol{P}^T(f) \tag{24}$$

and similarly for the updates of their respective sample estimates $\hat{\boldsymbol{R}}_\alpha(f), \hat{\boldsymbol{R}}_b(f), \hat{\boldsymbol{R}}_c(f)$.

Note that for the sake of algorithm simplicity, we will not exploit the fact that our specific PARAFAC model is symmetric. That is, the algorithm that follows treats $\boldsymbol{A}(f)$ and $\boldsymbol{B}(f)$ as independent variables; symmetry will only be exploited in the calculation of the final estimate of $\boldsymbol{A}(f)$.

B. Algorithm

The basic idea behind the TALS procedure for PARAFAC fitting is to update each time a subset of parameters using LS regression while keeping the previously obtained estimates of the rest of parameters fixed. This alternating projections-type procedure is iterated for all subsets of parameters until convergence is achieved [9], [3], [28], [29].

In application to our problem, the PARAFAC TALS procedure can be formulated as follows

- **Step 1:** Initialize $\boldsymbol{P}(f)$ and $\boldsymbol{A}(f)$.
- **Step 2:** Find the estimate $\boldsymbol{B}(f)$ by solving the following LS problem

  $$\hat{\boldsymbol{B}}(f) = \arg\min_{\boldsymbol{B}(f)} \|\hat{\boldsymbol{R}}_\alpha(f) - [\boldsymbol{P}(f) \odot \boldsymbol{A}(f)]\boldsymbol{B}(f)\|_F^2 \tag{25}$$

  whose solution is given by

  $$\hat{\boldsymbol{B}}(f) = [\boldsymbol{P}(f) \odot \boldsymbol{A}(f)]^\dagger \hat{\boldsymbol{R}}_\alpha(f) \tag{26}$$

  Set $\boldsymbol{B}(f) = \hat{\boldsymbol{B}}(f)$.
- **Step 3:** Find the estimate of $\boldsymbol{A}(f)$ by solving the following LS problem

  $$\hat{\boldsymbol{A}}(f) = \arg\min_{\boldsymbol{A}(f)} \|\hat{\boldsymbol{R}}_b(f) - [\boldsymbol{B}^T(f) \odot \boldsymbol{P}(f)]\boldsymbol{A}^T(f)\|_F^2 \tag{27}$$

  whose solution is given by

  $$\hat{\boldsymbol{A}}(f) = \hat{\boldsymbol{R}}_b^T(f)\Big[[\boldsymbol{B}^T(f) \odot \boldsymbol{P}(f)]^\dagger\Big]^T \tag{28}$$

  Set $\boldsymbol{A}(f) = \hat{\boldsymbol{A}}(f)$.
- **Step 4:** Find the estimate of $\boldsymbol{P}(f)$ by solving the following LS problem

  $$\hat{\boldsymbol{P}}(f) = \arg\min_{\boldsymbol{P}(f)} \|\hat{\boldsymbol{R}}_c(f) - [\boldsymbol{A}(f) \odot \boldsymbol{B}^T(f)]\boldsymbol{P}^T(f)\|_F^2 \tag{29}$$

  whose solution is given by

  $$\hat{\boldsymbol{P}}(f) = \hat{\boldsymbol{R}}_c^T(f)\Big[[\boldsymbol{A}(f) \odot \boldsymbol{B}^T(f)]^\dagger\Big]^T \tag{30}$$

  Set $\boldsymbol{P}(f) = \hat{\boldsymbol{P}}(f)$.
- **Step 5:** Repeat steps 2-4 until convergence is achieved and then compute the final estimate of $\boldsymbol{A}(f)$ as $\hat{\boldsymbol{A}}(f) = [\boldsymbol{A}(f) + \boldsymbol{B}^H(f)]/2$.

The TALS algorithm is repeated for every frequency. The complexity of the TALS algorithm is $\mathcal{O}(I^3 + J^2IP)$ ($\mathcal{O}(I^3 + I^3P)$ for the model of the ICE approach) per iteration. It is worth noting that when I is small relative to J and P, only a few iterations of the algorithm are usually required to achieve convergence [28]. The TALS algorithm can be initialized randomly, or via algebraic eigenvalue-based solutions, similar to ESPRIT, in certain cases; see [3].

V. Resolving the scaling and permutation ambiguities

As already mentioned, in order to achieve adequate separation performance, the resolution of the frequency-dependent permutation and scaling ambiguity problems must be addressed. In this section, we suggest a novel method for solving the permutation and scaling problems that arise in the blind speech separation context. In the first two subsections, our method of dealing with the frequency-dependent permutation ambiguities is presented, while the third subsection describes how the scaling problem is resolved.

A column-permuted and scaled version of the true mixing matrix $\boldsymbol{A}(f)$, can be written as

$$\boldsymbol{A}_{s,p}(f) = \boldsymbol{A}_s(f)\boldsymbol{\Pi}(f) = \boldsymbol{A}(f)\boldsymbol{D}(f)\boldsymbol{\Pi}(f) \tag{31}$$

where $\boldsymbol{\Pi}(f)$ is a permutation matrix and $\boldsymbol{D}(f)$ is a diagonal scaling matrix. This is the output matrix of the TALS algorithm at frequency f, when following the DCE approach.

Similarly, a column-permuted and scaled version of the matrix $\boldsymbol{U}(f) = \boldsymbol{W}^H(f)$, where $\boldsymbol{W}(f)$ is the true beamforming matrix, can be written as

$$\boldsymbol{U}_{s,p}(f) = \boldsymbol{U}_s(f)\boldsymbol{\Pi}(f) = \boldsymbol{U}(f)\boldsymbol{D}(f)\boldsymbol{\Pi}(f) \tag{32}$$

This is the output matrix of the TALS algorithm at frequency f, when following the ICE approach.

In order to separate the individual speech signals, it is imperative to reduce this frequency-dependent permutation and scaling to a single, frequency-independent permutation and scaling. How this is done, is discussed next.

A. Resolving the frequency-dependent permutation problem

First, let us consider the data model of the first approach. We resolve the permutation problem by exploiting two inherent attributes of the system under study. Proper interpretation and utilization of each of these attributes allows us to transform the frequency dependent permutation problem into an optimization problem with each attribute providing us with a different optimization criterion. We then formulate a joint optimization criterion that incorporates these two constituent criteria.

•Criterion 1: Based on the underlying physical model of the electro-acoustic $J\times I$ (microphones$\times$speakers) system, the elements of the mixing matrices $\{\boldsymbol{A}(f), f = 0,\ldots,T-1\}$ may be decomposed into products of the form

$$\alpha_{j,i}(f) = G_{j,i}(f)H_j(f)e^{-\frac{2\pi\sqrt{-1}f}{T}\tau_{j,i}} \tag{33}$$

where

- $G_{j,i}(f)$ represents the frequency response of the acoustic channel between the jth sensor and the ith source.
- $H_j(f)$ represents the jth microphone's frequency response.
- $\tau_{j,i}$ represents the delay, due to the time needed for the sound to cover the distance between the ith speaker and the jth microphone.

For this subsection only (derivation of criterion 1), we make the following assumptions:

Assumption 5.1: The speech signals travel in an almost non-reverberant (unechoic) environment. This means that, aside from a real multiplicative factor that is associated with the power loss over distance of the signals, the frequency responses of the acoustic channels are almost identical. Hence, $G_{j,i}(f) \approx n_{j,i}G_(f)$, where $n_{j,i}$ is the aforementioned multiplicative factor, that depends solely on the distance between the ith speaker and the jth microphone.

Assumption 5.2: The microphones' frequency responses are almost flat over the frequency band of interest (i.e., 20Hz-3.4KHz approximately, which is the frequency band of human speech). Hence, $H_j(f) \approx H_j$ for all $j = 1,\ldots,J$, with $H_j \in \mathbb{C}$.

Under the assumptions made, (33) can be rewritten as

$$\alpha_{j,i}(f) \approx n_{j,i}G(f)H_j e^{-\frac{2\pi\sqrt{-1}f}{T}\tau_{j,i}} \tag{34}$$

Now, normalize the spatial signatures of the output matrices (31) with respect to an arbitrary reference microphone (e.g., the first microphone for $j = 1$). Consider the vector whose elements are the magnitudes of the respective elements of the ith normalized spatial signature. Using (34), the latter can be written as

$$\boldsymbol{\alpha}_i^{(n)}(f) \approx \begin{bmatrix} 1 \\ \frac{n_{2,i}|H_2|}{n_{1,i}|H_1|} \\ \vdots \\ \frac{n_{J,i}|H_J|}{n_{1,i}|H_1|} \end{bmatrix} \tag{35}$$

for all $i = 1, \ldots, I$.

Hence, in non-reverberant (or, mildly reverberant) environments, where the recordings have been conducted using microphones with (almost) flat frequency responses, the magnitude of the elements of the normalized spatial signature, is approximately independent of frequency, as is readily seen from (35). Thus, the collection of vectors $\{\boldsymbol{\alpha}_i^{(n)}(f), f = 0, \ldots, T-1, i = 1, \ldots, I\}$, can be divided into I separate clusters, each of the latter being associated with a different speaker. Each cluster consists of T resemblant (in the Euclidean distance sense) vectors $\{\boldsymbol{\alpha}_i^{(n)}(f), f = 0, \ldots, T-1\}$, i.e., the vectors (35) which correspond to the same specific speaker (fixed i).

We apply a vector quantization (VQ) clustering procedure [7], over all the available vectors $\{\boldsymbol{\alpha}_i^{(n)}(f), f = 0, \ldots, T-1, i = 1, \ldots, I\}$, in order to determine the I center vectors (centroids) $\{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_I\}$, with $\boldsymbol{c}_i \in \mathbb{R}^J$, of the I clusters. With the I centroids at our disposal, the frequency-dependent permutation problem boils down to the following Integer Least Squares (ILS) minimization problem

$$\min_{\{s_f\}_{f=0}^{T-1} \in \{1,\ldots,I!\}^T} \sum_{f=0}^{T-1} \|\boldsymbol{C} - \boldsymbol{A}_n(f)\boldsymbol{\Pi}_1 1_{\{s_f=1\}} - \ldots - \boldsymbol{A}_n(f)\boldsymbol{\Pi}_{I!} 1_{\{s_f=I!\}}\|_F^2 \tag{36}$$

where $\boldsymbol{C} \in \mathbb{R}^{J\times I}$ is the matrix whose columns are the centroids $\{\boldsymbol{c}_i, i = 1, \ldots, I\}$, $\boldsymbol{A}_n(f) \in \mathbb{R}^{J\times I}$ is the matrix whose columns are the arbitrarily permuted[4] vectors $\{\boldsymbol{\alpha}_i^{(n)}(f), i = 1, \ldots, I\}$, $\boldsymbol{\Pi}_l \in \mathbb{R}^{I\times I}$ for $l = 1, \ldots, I!$ is one of the total number of[5] $I!$ permutation matrices of dimensionality $I \times I$, $1_{\{x=\alpha\}} = \begin{cases} 0 & \text{if } x \neq \alpha \\ 1 & \text{if } x = \alpha \end{cases}$, and s_f takes on the integer values $\{1, \ldots, I!\}$.

The solution $\{s_1, \ldots, s_T\}$ to the minimization problem (36) provides us with all the information we need for dealing with the frequency-dependent permutation problem. Suppose that $s_f = l$ with $l \in \{1, \ldots, I!\}$. Then, by permuting the columns of $A_{s,p}(f)$ in accordance with the permutation matrix $\boldsymbol{\Pi}_l$, the permutation ambiguity at frequency f is lifted. Thus, based on the values $\{s_f, f = 1, \ldots, T\}$ that criterion (36) provides, and performing the appropriate permutations across all frequency bins, the frequency-dependent permutation problem can be adequately resolved.

•Criterion 2: For adjacent frequency bins, $f_1, f_2 = f_1+1$, $\boldsymbol{\alpha}_i(f_1) \approx \boldsymbol{\alpha}_i(f_2)$ and consequently $\boldsymbol{\alpha}_i^{(n)}(f_1) \approx \boldsymbol{\alpha}_i^{(n)}(f_2)$. This is valid for dense FFT grids irrespective of whether or not assumptions 5.1 and 5.2 hold. Based on this fact, we formulate our second ILS minimization criterion

$$\min_{\{s_f\}_{f=0}^{T-1} \in \{1,\ldots,I!\}^T} \sum_{f=0}^{T-1} \|\boldsymbol{A}_n(f)\boldsymbol{\Pi}_1 1_{\{s_f=1\}} + \ldots + \boldsymbol{A}_n(f)\boldsymbol{\Pi}_{I!} 1_{\{s_f=I!\}} - \boldsymbol{A}_n(f-1)\boldsymbol{\Pi}_1 1_{\{s_{f-1}=1\}} - \ldots$$
$$- \boldsymbol{A}_n(f-1)\boldsymbol{\Pi}_{I!} 1_{\{s_{f-1}=I!\}}\|_F^2 \tag{37}$$

where $\boldsymbol{A}_n(-1) = \boldsymbol{0}^{J\times I}$, by convention. Criterion (37) should also provide adequate resolution of the permutation problem, by reordering the columns of the matrices $\boldsymbol{A}_{s,p}(f)$ of (31), as outlined in the previous paragraph for our first criterion.

[4]in accordance with the arbitrary permutations of the columns of the output matrices $\{\boldsymbol{A}_{s,p}(f), f = 0, \ldots, T-1\}$

[5]$I!$ equals the total number of possible permutations of the columns or the rows of an $I \times I$ matrix or equivalently the total number of permutation matrices of dimensionality $I \times I$

The criteria of (36) and (37) can be combined into one overall ILS minimization criterion

$$\min_{\{s_f\}_{f=0}^{T-1}\in\{1,\ldots,I!\}^T}\sum_{f=0}^{T-1}\{\|\boldsymbol{C}-\boldsymbol{A}_n(f)\boldsymbol{\Pi}_1 1_{\{s_f=1\}}-\ldots-\boldsymbol{A}_n(f)\boldsymbol{\Pi}_{I!}1_{\{s_f=I!\}}\|_F^2+\lambda\|\boldsymbol{A}_n(f)\boldsymbol{\Pi}_1 1_{\{s_f=1\}}+\ldots$$
$$+\boldsymbol{A}_n(f)\boldsymbol{\Pi}_{I!}1_{\{s_f=I!\}}-\boldsymbol{A}_n(f-1)\boldsymbol{\Pi}_1 1_{\{s_{f-1}=1\}}-\ldots-\boldsymbol{A}_n(f-1)\boldsymbol{\Pi}_{I!}1_{\{s_{f-1}=I!\}}\|_F^2 \quad (38)$$

where λ is a weighting factor. When $\lambda\rightarrow 0$ (practically, for $\lambda << 1$), the combined criterion (38) coincides with criterion 1, while for $\lambda\rightarrow\infty$ (practically, for $\lambda >> 1$), (38) coincides with criterion 2.

Note that the ILS minimization criterion (38), fixes the permutation ambiguity globally across all frequency bins. This is an important advantage versus a sequential approach to the permutation problem, where starting from the first frequency bin f_1, we adjust the permutation of each bin relative to its previous bin. This is because the sequential approach, although simpler, has a major drawback as explained next. Consider the situation where an error is made in estimating the correct permutation matrix for frequency bin f_k. Using the sequential approach, it is quite possible that many of the estimated permutation matrices for the frequency bins after f_k will be wrong. In the worst case scenario we will have half of the frequency bins with one permutation and the other half with a different permutation, which will result in very poor or no separation. Such a catastrophic situation is prevented by the global ILS minimization criterion (38).

The criterion of (38) can be properly modified in order to resolve the frequency-dependent permutation problem that arises when the second approach is followed. In this case we have

$$\min_{\{s_f\}_{f=0}^{T-1}\in\{1,\ldots,I!\}^T}\sum_{f=0}^{T-1}\{\|\boldsymbol{C}-\boldsymbol{U}_n(f)\boldsymbol{\Pi}_1 1_{\{s_f=1\}}-\ldots-\boldsymbol{U}_n(f)\boldsymbol{\Pi}_{I!}1_{\{s_f=I!\}}\|_F^2+\lambda\|\boldsymbol{U}_n(f)\boldsymbol{\Pi}_1 1_{\{s_f=1\}}+\ldots$$
$$+\boldsymbol{U}_n(f)\boldsymbol{\Pi}_{I!}1_{\{s_f=I!\}}-\boldsymbol{U}_n(f-1)\boldsymbol{\Pi}_1 1_{\{s_{f-1}=1\}}-\ldots-\boldsymbol{U}_n(f-1)\boldsymbol{\Pi}_{I!}1_{\{s_{f-1}=I!\}}\|_F^2 \quad (39)$$

where $\boldsymbol{U}_n(-1)=\boldsymbol{0}^{J\times I}$, by convention. In (39), $\boldsymbol{U}_n(f)=[\boldsymbol{u}_1^{(n)}(f),\ldots,\boldsymbol{u}_I^{(n)}(f)]\in\mathbb{R}^{J\times I}$ with $\boldsymbol{u}_i^{(n)}(f)$ representing the vector whose elements are the magnitudes of the respective elements of the (normalized with respect to an arbitrary microphone) ith column of the corresponding frequency's output matrix (32). Again, $\boldsymbol{C}\in\mathbb{R}^{J\times I}$ is the matrix whose columns are the centroids $\{\boldsymbol{c}_i, i=1,\ldots,I\}$, obtained after application of a clustering procedure via VQ over all the available vectors $\{\boldsymbol{u}_i^{(n)}(f), f=0,\ldots,T-1, i=1,\ldots,I\}$.

The ILS minimization criteria of (38), (39) proved to be satisfactory for resolving the permutation problem of the real-world BSS problems we examine in section VI. The following subsection considers the 2-sources case and discusses the selection of the algorithm we used for the implementation of the criterion (38). The pertinent analysis for the criterion (39) is identical, thus is omitted for brevity.

B. Special case: $J\times 2$ mixture

We will henceforth focus on the 2-sources (i.e., $(J,I)=(J,2)$) case, but all our results readily extend to the case of general (J,I). By letting $s_f\in\{-1,1\}$, $\boldsymbol{\Pi}_{-1}=\boldsymbol{I}=\begin{bmatrix}1&0\\0&1\end{bmatrix}$ and $\boldsymbol{\Pi}_1=\boldsymbol{\Pi}=\begin{bmatrix}0&1\\1&0\end{bmatrix}$, the optimization criterion of (38) simplifies to

$$\min_{\{s_f\}_{f=1}^{T}\in\{\pm1\}^T}\sum_{f=1}^{T}\Big\{\Big\|\boldsymbol{C}-\frac{1-s_f}{2}\boldsymbol{A}_n(f)-\frac{1+s_f}{2}\boldsymbol{A}_n(f)\boldsymbol{\Pi}\Big\|_F^2+$$
$$\lambda\Big\|\frac{1-s_f}{2}\boldsymbol{A}_n(f)+\frac{1+s_f}{2}\boldsymbol{A}_n(f)\boldsymbol{\Pi}-\frac{1-s_{f-1}}{2}\boldsymbol{A}_n(f-1)-\frac{1+s_{f-1}}{2}\boldsymbol{A}_n(f-1)\boldsymbol{\Pi}\Big\|_F^2\Big\} \quad (40)$$

It is shown in Appendix A that (40) can be put in the following more familiar form:

$$\min_{\boldsymbol{s}\in\{\pm1\}^T}\|\boldsymbol{x}-\boldsymbol{B}\boldsymbol{s}\|_2^2 \quad (41)$$

where $\boldsymbol{s} = [s_1, s_2, \ldots, s_T]^T$.

The ILS optimization problem of (41) is known to be NP-hard. Over the years, several methods have been developed to obtain high-quality solutions at manageable complexity cost. The trade-off between performance and complexity makes the selection of the most suitable algorithm an application-dependent issue.

Sphere Decoding (SD) [32] along with numerous variants of it [1], [5], [33], [34], and Semi-Definite Relaxation (SDR) [16], are methods that can provide near-optimal performance at often acceptable complexity. Unfortunately, they are too complex when the dimensionality of $\boldsymbol{s}$ exceeds 100 (e.g. 128 in our context). Block Minimum Mean Squared Error - Decision Feedback Equalization (BMMSE-DFE) [6] and Probabilistic Data Association (PDA) [15], [22] are two simple methods that sometimes perform well, but they need variance estimates for "signal" and "noise" at their input, which are not obvious in the context of our application. A very suboptimal solution of the ILS optimization problem (41) is to use the sign of the unconstrained LS solution, known as Quantized Zero-Forcing (QZF) in the digital communications community, but performance leaves much to be desired. The best performance-complexity trade-off in our context was exhibited by the Successive Interference Cancelation - Iterative Least Squares (SIC-ILS) method [14]. In our specific application, SIC-ILS performs equally well as the near optimal Sphere Decoding method for low dimensionalities (e.g., 32 or 64). We therefore used SIC-ILS throughout.

C. Resolving the frequency-dependent scaling problem

After the resolution of the frequency-dependent permutation ambiguity, as described in the previous subsections, the only remaining problem is the frequency-dependent scaling ambiguity: i) of columns of the estimated mixing matrices $\{\boldsymbol{A}(f), f = 0, \ldots, T-1\}$ for the DCE approach, ii) of rows of the estimated unmixing matrices $\{\boldsymbol{W}(f), f = 0, \ldots, T-1\}$ for the ICE approach. First, let us consider the resolution of the scaling problem for the DCE approach, that is based upon the following simple fact whose proof can be found in Appendix B.

Fact 1: Consider a full rank matrix $\boldsymbol{A} \in \mathbb{C}^{J \times I}$, with $J \geq I$. Let $\boldsymbol{A}_n \in \mathbb{C}^{J \times I}$ denote the matrix whose ith column $\boldsymbol{a}_i^{(n)}$ is the ith column $\boldsymbol{a}_i$ of $\boldsymbol{A}$, divided by the element $a_{j_i,i}$ of $\boldsymbol{a}_i$, i.e., $\boldsymbol{a}_i^{(n)} = \frac{1}{a_{j_i,i}}\boldsymbol{a}_i$, with $i = 1, \ldots, I, j_i \in \{1, \ldots, J\}$. Then $\boldsymbol{A}_n^{\dagger}\boldsymbol{A} = diag\{[a_{j_1,1}, \ldots, a_{j_I,I}]^T\}$.

Let

$$\boldsymbol{A}_s(f) = [d_1(f)\boldsymbol{\alpha}_1(f), \ldots, d_I(f)\boldsymbol{\alpha}_I(f)] = \boldsymbol{A}(f) \begin{bmatrix} d_1(f) & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & d_I(f) \end{bmatrix} = \boldsymbol{A}(f)\boldsymbol{D}(f) \tag{42}$$

where $d_i(f) \in \mathbb{C}$ for all $i = 1, \ldots, I$, represent a column scaled version of the true mixing matrix $\boldsymbol{A}(f)$ at frequency f. Divide the columns of matrix $\boldsymbol{A}_s(f)$ by their respective elements $\{d_1(f)\alpha_{j_1,1}(f), \ldots, d_I(f)$ $\alpha_{j_I,I}(f)\}$, with $j_i \in \{1, \ldots, J\}$, to yield matrix $\boldsymbol{A}_{s,n}(f)$

$$\boldsymbol{A}_{s,n}(f) = \boldsymbol{A}_s(f) \begin{bmatrix} \frac{1}{d_1(f)\alpha_{j_1,1}(f)} & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & \frac{1}{d_I(f)\alpha_{j_I,I}(f)} \end{bmatrix} = \boldsymbol{A}(f)\boldsymbol{D}(f)\boldsymbol{D}^{-1}(f) \begin{bmatrix} \frac{1}{\alpha_{j_1,1}(f)} & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & \frac{1}{\alpha_{j_I,I}(f)} \end{bmatrix} =$$

$$\boldsymbol{A}(f) \begin{bmatrix} \frac{1}{\alpha_{j_1,1}(f)} & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & \frac{1}{\alpha_{j_I,I}(f)} \end{bmatrix} = \boldsymbol{A}_n(f) \tag{43}$$

where $\boldsymbol{A}_n(f)$ denotes the matrix whose ith column $\boldsymbol{a}_i^{(n)}(f)$ is the ith column $\boldsymbol{a}_i(f)$ of the true mixing matrix $\boldsymbol{A}(f)$, divided by element $a_{j_i,i}(f)$ of $\boldsymbol{a}_i(f)$.

Let us multiply (3) (we neglect the noise term) from the left with the pseudoinverse $\boldsymbol{A}_{s,n}^{\dagger}(f) = \boldsymbol{A}_{n}^{\dagger}(f)$. Invoking Fact 1, we have

$$\boldsymbol{A}_{s,n}^{\dagger}(f)\boldsymbol{x}(f,t) \approx \boldsymbol{A}_{n}^{\dagger}(f)\boldsymbol{A}(f)\boldsymbol{s}(f,t) = diag\{[\alpha_{j_1,1}(f),\ldots,\alpha_{j_I,I}(f)]^T\}\boldsymbol{s}(f,t) = [\alpha_{j_1,1}(f)s_1(f,t),\ldots,\alpha_{j_I,I}(f)s_I(f,t)]^T \quad (44)$$

Regarding the ICE approach, it is shown in Appendix C that by performing the appropriate manipulations on the row scaled version $\boldsymbol{W}_s(f)$ of the true unmixing matrix $\boldsymbol{W}(f)$, we can construct matrix $\boldsymbol{W}_{s,n}(f)$ given in (53). Multiplying (3) from the left with $\boldsymbol{W}_{s,n}(f)$ yields

$$\boldsymbol{W}_{s,n}(f)\boldsymbol{x}(f,t) \approx [\alpha_{j_1,1}(f)s_1(f,t),\ldots,\alpha_{j_I,I}(f)s_I(f,t)]^T \quad (45)$$

We see from (44) and (45) that each speaker signal $s_i(f,t)$ is multiplied by the element $a_{j_i,i}(f)$, for all $i = 1,\ldots,I$. But $a_{j_i,i}(f)$ is just the total frequency response[6] that corresponds to the link between the ith speaker and the j_ith microphone, at frequency f, which is almost constant across all frequencies of interest (see assumption 5.2). Thus, the frequency-dependent scaling of the individual speaker signals that would result if we had used the pseudoinverses of the matrices $\{\boldsymbol{A}_s(f), f = 0,\ldots,T-1\}$ in (44) or the matrices $\{\boldsymbol{W}_s(f), f = 0,\ldots,T-1\}$ in (45), now is reduced to a single, frequency-independent scaling, i.e., the frequency-dependent scaling ambiguity problem is resolved. Intuitively, equations (44) and (45) imply that after separation we essentially listen to the ith speaker's speech signal the way it is captured by the j_ith microphone, as if there were no other interfering speakers.

VI. Algorithms

In this section, we summarize our two BSS algorithms, namely DCE-PARAFAC-ILS and ICE-PARAFAC-ILS for $J \geq I$. The two algorithms are formulated as follows

- **Step 1**
  DCE-PARAFAC-ILS: Compute the autocorrelation data (4) for P time intervals over which the measured signals are assumed stationary, using the sample average (19) on the frequency-domain data (3).
  ICE-PARAFAC-ILS: If $J = I$ compute the "inverse" autocorrelation data (5) for P time intervals over which the measured signals are assumed stationary, using the sample average (19) on the frequency-domain data (3). If $J > I$ perform a dimension reducing prefiltering of the data in (3), as outlined in section II. Subsequently, compute the "inverse" autocorrelation data (8) for P time intervals over which the measured signals are assumed stationary, using the sample average (19) on the prefiltered frequency-domain data.
- **Step 2**
  DCE-PARAFAC-ILS (*ICE-PARAFAC-ILS*): Apply the TALS algorithm to the autocorrelation data calculated in Step 1, iteratively for all frequency bins $f = 0,\ldots,T-1$, to obtain the matrices $\{\boldsymbol{A}_{s,p}(f), f = 0,\ldots,T-1\}$ ($\{\boldsymbol{W}_{s,p}(f), f = 0,\ldots,T-1\}$) i.e., the estimates of the true mixing (unmixing) matrices up to a frequency-dependent permutation and scaling ambiguity of their columns (rows).
- **Step 3**
  DCE-PARAFAC-ILS (*ICE-PARAFAC-ILS*): Apply a VQ clustering procedure over all the available vectors $\{\boldsymbol{a}_i^{(n)}(f), f = 0,\ldots,T-1, i = 1,\ldots,I\}$ $\left(\{\boldsymbol{u}_i^{(n)}(f), f = 0,\ldots,T-1, i = 1,\ldots,I\}\right)$ as desctribed in subsection A of section V, to obtain the associated with the I speakers centroids $\{\boldsymbol{c}_1,\ldots,\boldsymbol{c}_I\}$. Form matrix $\boldsymbol{C}$ whose columns are the centroids $\{\boldsymbol{c}_1,\ldots,\boldsymbol{c}_I\}$.
- **Step 4**

[6]which equals the triple product of the frequency response of the acoustic channel between ith speaker-j_ith microphone, the frequency response of the j_ith microphone and an exponential term that is dependent on the delay, as verified from (33).

DCE-PARAFAC-ILS (*ICE-PARAFAC-ILS*): Apply the SIC-ILS algorithm in order to solve the ILS optimization problem in (38) (in (39)), and fix the arbitrary permutations as described in subsection A of section V, thus determining the matrices $\{\boldsymbol{A}_s(f), f = 0, \ldots, T-1\}$ $\big(\{\boldsymbol{W}_s(f), f = 0, \ldots, T-1\}\big)$, i.e., the estimates of the true mixing (unmixing) matrices up to a frequency-dependent scaling ambiguity of their columns (rows).

- **Step 5**
  DCE-PARAFAC-ILS (*ICE-PARAFAC-ILS*): Compute the matrices $\{\boldsymbol{A}_{s,n}(f), f = 0, \ldots, T-1\}\big(\{\boldsymbol{W}_{s,n}(f), f = 0, \ldots, T-1\}\big)$, as outlined in subsection C of section V.
- **Step 6**
  DCE-PARAFAC-ILS (*ICE-PARAFAC-ILS*): Apply the Inverse DFT to the collection of matrices $\{\boldsymbol{A}^{\dagger}_{s,n}(f), f = 0, \ldots, T-1\}$ $\big(\{\boldsymbol{W}_{s,n}, f = 0, \ldots, T-1\}\big)$ to yield the estimate of the inverse-channel impulse response matrix $\{\hat{\boldsymbol{W}}(\tau), \tau = 0, \ldots, T-1\}$.
- **Step 7**
  Estimate the speaker signals via $\hat{\boldsymbol{s}}(t) = \hat{\boldsymbol{W}} * \boldsymbol{x}(t)$

VII. Experimental Results

In the present section, we report the results obtained after application of our two algorithms to real-world data of the PEACH multi-microphone database [20].

A. Experimental setup and performance measurement considerations

The mixed speech signals of the PEACH database were recorded using $I = 2$ loudspeakers and $J = 4$ omni-directional microphones. Analytically, the speech mixtures' acquisition procedure was conducted as follows: First, three utterances were acquired with a close-talk microphone for each one of eight different speakers. The speakers were separated into two groups of four speakers each, with speakers of the same group pronouncing the same three sentences every time. After the completion of the close-talk signals' acquisition procedure, the speakers were divided into four pairs, with each pair consisting of speakers that belong to different groups. The close-talk signals of each pair were reproduced together in a quiet room, by two loudspeakers in front of a microphone array consisting of four microphones, for four different geometric configurations. Specifically, utterance i of one speaker was played back simultaneously with utterance i of the pair's other speaker by the two loudspeakers, for $i = 1, 2, 3$. Thus, each pair of speakers provided three different speech mixtures for each geometric configuration. The measurements can be separated into four sessions of twelve experiments each, giving a total of 48 experiments, and with each session corresponding to a different geometric configuration. As shown in Fig. 1, the loudspeakers and the microphones were uniformly-spaced on two parallel lines at distance d_1. The distance between microphones was d_2, which was also the distance between the loudspeakers. For session I: $d_1 = 150cm, d_2 = 10cm$, for session II: $d_1 = 150cm, d_2 = 30cm$, for session III: $d_1 = 100cm, d_2 = 10cm$ and for session IV: $d_1 = 100cm, d_2 = 30cm$.

In order to quantify the separation performance, we measured the average input Signal-to-Interference Ratio (SIR) across microphones and the SIR for each of the unmixed signals that are the output of the BSS algorithms. SIR is measured following the method proposed in [24]: The close-talk signals of each speaker were played back through the corresponding loudspeaker one at a time, i.e. at each time only one source (loudspeaker) was active. Let $P^x_{j,i} = \sum_{t=0}^{T-1} x_j^2(t)$, where T is the number of samples, represent the power of the recorded signal at the jth microphone when only source i is active and all the other sources are inactive. The signal-to-interference ratio in dB of the recorded signal $x_j(t)$ at the jth microphone can be estimated using

$$SIR_x(j,i) = 10 \log \frac{\max_i P^x_{j,i}}{\sum_{i=1}^{I} P^x_{j,i} - \max_i P^x_{j,i}} \tag{46}$$

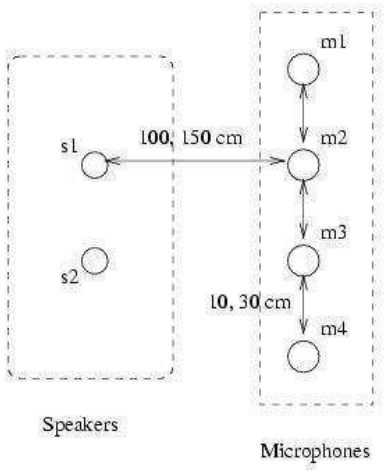

Fig. 1. Geometry of the measurement setup

for $j = 1, \ldots, J$ and $i = 1, \ldots, I$. In our case the number of microphones $J = 4$ and the number of sources $I = 2$. The formula (46) can also be used to measure $SIR_y(j, i)$, that is the SIR of the jth output of the separating algorithm, simply by substituting $P^x_{j,i}$ with $P^y_{j,i}$, the latter representing the power of the signal at the jth output of the algorithm when only source i is active, for $j, i = 1, \ldots, I$. For our experiments $I = 2$. Finally, the improvement on the SIR for the ith source is defined as $SIR^+(i) = \max_j SIR_y(j, i) - \underline{SIR}_x(i)$, where $\underline{SIR}_x(i)$ is the average input SIR across microphones for the ith source, i.e. $\underline{SIR}_x(i) = \frac{\sum_{j=1}^{J} SIR_x(j,i)}{J}$.

The experiments have been conducted using speech data of approximately equal duration in time (about 19 seconds), sampled at 16*KHz*. Throughout all BSS experiments using our proposed algorithms, the number of time segments the speech mixtures were divided into, was set to $P = 10$. Due to the approximately equal length of all speech signals (about 300,000 samples), the size of each segment was about 30,000 data samples long, for all experiments. For the solution of the permutation problem, the two ILS minimization criteria in (38) and (39) were weighted equally, i.e., $\lambda = 1$.

B. Discussion on the results

Figures 2 and 3 show how the performance of the proposed algorithms changes for various numbers of frequency bins used to estimate the autocorrelation matrices, for all geometric configurations. The plots of each figure were obtained by averaging the improvements on the SIR over all twelve experiments and speakers of the corresponding session, at each point. It should be mentioned that depending on the experiment the SIR improvement reached up to more than 12 dBs in certain cases, but we chose to report the aggregate behavior of the proposed algorithms instead of focusing on specific experiments [7]. Note that on the average, the performance seems to decrease as the size of the Fast Fourier Transform (FFT) grid gets bigger; this is the reason we don't report the performance for more than 256 frequency bins. In most cases, the highest separation performance for both algorithms, is achieved for small FFT grid sizes of 16 and 32 bins. A second remark is that the separation performance is a function of the geometry of the measurement setup, since the performance improves in the following cases: i) when the distance between microphones increases, ii) when the distance between loudspeakers and microphones decreases. The rationale for

[7]A subset of the recorded speech mixtures along with the corresponding separated signals, can be found in the following website: *http://www.speech.tuc.gr/res/research_bss.html.*

TABLE II

COMPARISON TABLE FOR DCE-PARAFAC-ILS, ICE-PARAFAC-ILS, AND PARRA'S METHOD: AVERAGE SIR IMPROVEMENT IN DB FOR EACH EXPERIMENTAL SESSION, ALONG WITH THE AVERAGE EXECUTION TIME FOR EACH OF THE ABOVE METHODS IN SECONDS. THE NUMBER OF TIME SEGMENTS $P = 10$ WHILE THE NUMBER OF FREQUENCY BINS $T = 32$ FOR DCE-, ICE-PARAFAC-ILS AND $T = 4096$ FOR PARRA'S METHOD.

Session	*DCE-PARAFAC-ILS SIR impr. mean*	*ICE-PARAFAC-ILS SIR impr. mean*	*Parra's Method SIR impr. mean*
I: 150cm-10cm	2.9	3.2	3.5
II: 150cm-30cm	4.2	5.0	3.1
III: 100cm-10cm	5.6	5.2	6.8
IV: 100cm-30cm	6.3	5.3	1.0
Overall Average	4.750	4.675	3.600
Execution Time	4.5	6.4	351.3

this, partially lies in the method that resolves the frequency-dependent permutation problem; the method virtually exploits the diversification in the distances between the speakers-microphones positions in each geometric configuration, via the first criterion of (38) (see Assumption 5.1). In addition, PARAFAC is sensitive to diversity. In the limit as $d_1 \to \infty$ or as $d_2 \to 0$ there is no diversity, which results in poor or no estimation of the matrices $\{\boldsymbol{A}(f), f = 0, \ldots, T-1\}$ and $\{\tilde{\boldsymbol{W}}(f), f = 0, \ldots, T-1\}$ (up to a frequency-dependent permutation and scaling ambiguity) in the PARAFAC step of DCE-PARAFAC-ILS and ICE-PARAFAC-ILS, respectively. Figure 4 depicts the overall behavior of the proposed separating algorithms in terms of SIR improvement averaged over all four geometric configurations versus the number of frequency bins used. Finally, bearing in mind that the total running time of BSS algorithms is a crucial parameter (especially in the case where the algorithms are used in real-time applications), we report in figure 5 the average time needed for each of the proposed algorithms, in order to perform the source signals' separation for several FFT grid sizes. The times reported correspond to the MATLAB implementations of the algorithms, which were tested on a Pentium M-based personal computer running at 2 GHz with 1.5 MB of RAM. This figure demonstrates convincingly the strength of the proposed algorithms: High separation performance is achieved at a low computational cost, using 16 or 32 FFT grid points.

To support our arguments, the performance of our methods is compared with the method presented in [19], henceforth referred to as "Parra's method", for the corpus of the PEACH database. Table II reports the results for DCE-PARAFAC-ILS, ICE-PARAFAC-ILS and Parra's method for each experimental session along with the average execution time for each method, in order to perform the separation of the speaker signals. The parameters for "Parra's method" were chosen so that the method exhibits the highest possible performance. Specifically, the number of frequency bins $T = 4096$ while the length of the inverse channel impulse response was set to $Q = 512$, as suggested in [19]. For DCE-, ICE-PARAFAC-ILS $T = 32$. The number of time segments the speech signals were divided into, was set to $P = 10$, for all three methods. We see from Table II, that our methods outperform the method proposed in [19] in sessions II and IV, while Parra's method exhibits slightly better separation performance in the first and third experimental session. However, the overall separation performance in Table II, which can be thought of as a index of a method's effectiveness regardless of the geometric configuration, favors our proposed algorithms over Parra's method by more than 1 *dB*. In addition, the robustness of the new methods in changes of the experimental setup's geometry, seems to be higher when compared to Parra's method: The separation improvement of the PARAFAC-ILS methods varies from 2.9 to 6.3 *dB's* while Parra's method varies from 1 to 6.8 *dB's*. Finally, our algorithms are faster than Parra's by two orders of magnitude on average.

VIII. CONCLUSIONS

We have presented two new approaches for the blind speech separation problem. The new approaches employ PARAFAC to separate signals in the frequency domain and a context-specific ILS method to

resolve the frequency-dependent permutation ambiguity, which is a key problem in this setting. Our approaches offer guaranteed convergence, unlike [19], and better separation performance in most cases, measured both quantitatively and in subjective tests, at considerably reduced complexity. The link to the conjugate symmetric PARAFAC model established here, also shows that a much higher number of signals can be separated than what was suggested in [19]. In the future, we will investigate the separation performance of our algorithm as a function of the source-sensor geometry and the permutation minimization criterion weight λ.

IX. Acknowledgment

The authors would like to thank Mr. Christian Zieger and Mr. Luca Cristoforetti at ITC-IRST for collecting the corpus of the PEACH multi-microphone database. Thanks are also due to Prof. Vassilis Digalakis for useful suggestions.

Appendix A

It suffices to bring (40) in the form

$$\min_{\boldsymbol{s}\in\{\pm 1\}^T}\{\|\boldsymbol{x}_1-\boldsymbol{B}_1\boldsymbol{s}\|_2^2+\lambda\|\boldsymbol{x}_2-\boldsymbol{B}_2\boldsymbol{s}\|_2^2\} \tag{47}$$

since $\|\boldsymbol{x}_1-\boldsymbol{B}_1\boldsymbol{s}\|_2^2+\lambda\|\boldsymbol{x}_2-\boldsymbol{B}_2\boldsymbol{s}\|_2^2=\|\boldsymbol{x}-\boldsymbol{B}\boldsymbol{s}\|_2^2$ for $\boldsymbol{x}=\begin{bmatrix}\boldsymbol{x}_1\\ \sqrt{\lambda}\boldsymbol{x}_2\end{bmatrix}$ and $\boldsymbol{B}=\begin{bmatrix}\boldsymbol{B}_1\\ \sqrt{\lambda}\boldsymbol{B}_2\end{bmatrix}$.

Considering the first term of (40) we have

$$\begin{aligned}\sum_{f=0}^{T-1}\Big\|\boldsymbol{C}-\frac{1-s_f}{2}\boldsymbol{A}_n(f)-\frac{1+s_f}{2}\boldsymbol{A}_n(f)\boldsymbol{\Pi}\Big\|_F^2=\\ \sum_{f=0}^{T-1}\Big\|\boldsymbol{C}-\frac{\boldsymbol{A}_n(f)}{2}-\frac{\boldsymbol{A}_n(f)\boldsymbol{\Pi}}{2}+\Big(\frac{\boldsymbol{A}_n(f)}{2}-\frac{\boldsymbol{A}_n(f)\boldsymbol{\Pi}}{2}\Big)s_f\Big\|_F^2=\\ \sum_{f=0}^{T-1}\Big\|\boldsymbol{c}_1-\frac{\boldsymbol{a}_1^{(n)}(f)}{2}-\frac{\boldsymbol{a}_2^{(n)}(f)}{2}+\Big(\frac{\boldsymbol{a}_1^{(n)}(f)}{2}-\frac{\boldsymbol{a}_2^{(n)}(f)}{2}\Big)s_f\Big\|_2^2+\\ \sum_{f=0}^{T-1}\Big\|\boldsymbol{c}_2-\frac{\boldsymbol{a}_2^{(n)}(f)}{2}-\frac{\boldsymbol{a}_1^{(n)}(f)}{2}+\Big(\frac{\boldsymbol{a}_2^{(n)}(f)}{2}-\frac{\boldsymbol{a}_1^{(n)}(f)}{2}\Big)s_f\Big\|_2^2=\|\boldsymbol{x}_1-\boldsymbol{B}_1\boldsymbol{s}\|_2^2\end{aligned} \tag{48}$$

where $\boldsymbol{x}_1=\begin{bmatrix}\boldsymbol{c}_1-\frac{\boldsymbol{a}_1^{(n)}(1)}{2}-\frac{\boldsymbol{a}_2^{(n)}(1)}{2}\\ \vdots\\ \boldsymbol{c}_1-\frac{\boldsymbol{a}_1^{(n)}(T)}{2}-\frac{\boldsymbol{a}_2^{(n)}(T)}{2}\\ \boldsymbol{c}_2-\frac{\boldsymbol{a}_1^{(n)}(1)}{2}-\frac{\boldsymbol{a}_2^{(n)}(1)}{2}\\ \vdots\\ \boldsymbol{c}_2-\frac{\boldsymbol{a}_1^{(n)}(T)}{2}-\frac{\boldsymbol{a}_2^{(n)}(T)}{2}\end{bmatrix}$, $\boldsymbol{B}_1=\begin{bmatrix}\frac{\boldsymbol{a}_1^{(n)}(1)}{2}-\frac{\boldsymbol{a}_2^{(n)}(1)}{2} & \dots & \boldsymbol{0}\\ \vdots & \ddots & \vdots\\ \boldsymbol{0} & \dots & \frac{\boldsymbol{a}_1^{(n)}(T)}{2}-\frac{\boldsymbol{a}_2^{(n)}(T)}{2}\\ \frac{\boldsymbol{a}_2^{(n)}(1)}{2}-\frac{\boldsymbol{a}_1^{(n)}(1)}{2} & \dots & \boldsymbol{0}\\ \vdots & \ddots & \vdots\\ \boldsymbol{0} & \dots & \frac{\boldsymbol{a}_2^{(n)}(T)}{2}-\frac{\boldsymbol{a}_1^{(n)}(T)}{2}\end{bmatrix}$ and $\boldsymbol{a}_i^{(n)}(f)$ denotes the ith column of $\boldsymbol{A}_n(f)$.

Similarly, the second term of (40) can be written as

$$\sum_{f=1}^{T}\Big\|\frac{1-s_f}{2}\boldsymbol{A}_n(f)+\frac{1+s_f}{2}\boldsymbol{A}_n(f)\boldsymbol{\Pi}-\frac{1-s_{f-1}}{2}\boldsymbol{A}_n(f-1)-\frac{1+s_{f-1}}{2}\boldsymbol{A}_n(f-1)\boldsymbol{\Pi}\Big\|_F^2=\|\boldsymbol{x}_2-\boldsymbol{B}_2\boldsymbol{s}\|_2^2 \tag{49}$$

with

$$\boldsymbol{x}_2 = \begin{bmatrix} \frac{\boldsymbol{a}_1^{(n)}(1)}{2} + \frac{\boldsymbol{a}_2^{(n)}(1)}{2} \\ \frac{\boldsymbol{a}_1^{(n)}(2)}{2} + \frac{\boldsymbol{a}_2^{(n)}(2)}{2} - \frac{\boldsymbol{a}_1^{(1)}(1)}{2} - \frac{\boldsymbol{a}_2^{(n)}(1)}{2} \\ \vdots \\ \frac{\boldsymbol{a}_1^{(n)}(T)}{2} + \frac{\boldsymbol{a}_2^{(n)}(T)}{2} - \frac{\boldsymbol{a}_1^{(n)}(T-1)}{2} - \frac{\boldsymbol{a}_2^{(n)}(T-1)}{2} \\ \frac{\boldsymbol{a}_1^{(n)}(1)}{2} + \frac{\boldsymbol{a}_2^{(n)}(1)}{2} \\ \frac{\boldsymbol{a}_1^{(n)}(2)}{2} + \frac{\boldsymbol{a}_2^{(n)}(2)}{2} - \frac{\boldsymbol{a}_1^{(1)}(1)}{2} - \frac{\boldsymbol{a}_2^{(n)}(1)}{2} \\ \vdots \\ \frac{\boldsymbol{a}_1^{(n)}(T)}{2} + \frac{\boldsymbol{a}_2^{(n)}(T)}{2} - \frac{\boldsymbol{a}_1^{(n)}(T-1)}{2} - \frac{\boldsymbol{a}_2^{(n)}(T-1)}{2} \end{bmatrix} \text{ and}$$

$$\boldsymbol{B}_2 = \begin{bmatrix} \frac{\boldsymbol{a}_2^{(n)}(1)}{2} - \frac{\boldsymbol{a}_1^{(n)}(1)}{2} & \mathbf{0} & \dots & \mathbf{0} \\ \frac{\boldsymbol{a}_1^{(n)}(1)}{2} - \frac{\boldsymbol{a}_2^{(n)}(1)}{2} & \frac{\boldsymbol{a}_2^{(n)}(2)}{2} - \frac{\boldsymbol{a}_1^{(n)}(2)}{2} & \dots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \dots & \frac{\boldsymbol{a}_1^{(n)}(T-1)}{2} - \frac{\boldsymbol{a}_2^{(n)}(T-1)}{2} & \frac{\boldsymbol{a}_2^{(n)}(T)}{2} - \frac{\boldsymbol{a}_1^{(n)}(T)}{2} \\ \frac{\boldsymbol{a}_1^{(n)}(1)}{2} - \frac{\boldsymbol{a}_2^{(n)}(1)}{2} & \mathbf{0} & \dots & \mathbf{0} \\ \frac{\boldsymbol{a}_2^{(n)}(1)}{2} - \frac{\boldsymbol{a}_1^{(n)}(1)}{2} & \frac{\boldsymbol{a}_1^{(n)}(2)}{2} - \frac{\boldsymbol{a}_2^{(n)}(2)}{2} & \dots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \dots & \frac{\boldsymbol{a}_2^{(n)}(T-1)}{2} - \frac{\boldsymbol{a}_1^{(n)}(T-1)}{2} & \frac{\boldsymbol{a}_1^{(n)}(T)}{2} - \frac{\boldsymbol{a}_2^{(n)}(T)}{2} \end{bmatrix}$$

By setting $\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_1 \\ \sqrt{\lambda}\boldsymbol{x}_2 \end{bmatrix}$ and $\boldsymbol{B} = \begin{bmatrix} \boldsymbol{B}_1 \\ \sqrt{\lambda}\boldsymbol{B}_2 \end{bmatrix}$, as already mentioned, the optimization problem of (47) is reformulated as in (41).

APPENDIX B

Proof of Fact 1: Matrix $\boldsymbol{A}_n$ can be written as

$$\boldsymbol{A}_n = \boldsymbol{A}\boldsymbol{V} \tag{50}$$

where the diagonal matrix $\boldsymbol{V} = \begin{bmatrix} \frac{1}{\alpha_{j_1,1}} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \frac{1}{\alpha_{j_I,I}} \end{bmatrix}$. Hence, the left Moore-Penrose pseudoinverse of $\boldsymbol{A}_n$ is written

$$\boldsymbol{A}_n^{\dagger} = (\boldsymbol{A}_n^H \boldsymbol{A}_n)^{-1}\boldsymbol{A}_n^H = \left((\boldsymbol{A}\boldsymbol{V})^H \boldsymbol{A}\boldsymbol{V}\right)^{-1}(\boldsymbol{A}\boldsymbol{V})^H = (\boldsymbol{V}^H\boldsymbol{A}^H\boldsymbol{A}\boldsymbol{V})^{-1}\boldsymbol{V}^H\boldsymbol{A}^H = \boldsymbol{V}^{-1}(\boldsymbol{A}^H\boldsymbol{A})^{-1}\boldsymbol{V}^{-H}\boldsymbol{V}^H\boldsymbol{A}^H = \boldsymbol{V}^{-1}(\boldsymbol{A}^H\boldsymbol{A})^{-1}\boldsymbol{A}^H \tag{51}$$

and

$$\boldsymbol{A}_n^{\dagger}\boldsymbol{A} = \boldsymbol{V}^{-1}(\boldsymbol{A}^H\boldsymbol{A})^{-1}\boldsymbol{A}^H\boldsymbol{A} = \boldsymbol{V}^{-1} = \begin{bmatrix} \alpha_{j_1,1} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \alpha_{j_I,I} \end{bmatrix} = diag\{[\alpha_{j_1,1}, \dots, \alpha_{j_I,I}]^T\} \tag{52}$$

This completes the proof.

APPENDIX C

Consider the resolution of the scaling problem for the ICE approach. In this case let

$$\boldsymbol{W}_s(f) = \begin{bmatrix} d_1(f) & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & d_I(f) \end{bmatrix} \boldsymbol{A}^{\dagger}(f) = \boldsymbol{D}(f)\boldsymbol{A}^{\dagger}(f) \tag{53}$$

where $d_i(f) \in \mathbb{C}$ for all $i = 1, \dots, I$, represent a row scaled version of the true unmixing matrix $\boldsymbol{W}(f) = \boldsymbol{A}^{\dagger}(f)$ at frequency f. The right Moore-Penrose pseudoinverse of $\boldsymbol{W}_s(f)$ can be written as

$$\boldsymbol{W}_s^{\dagger}(f) = \big(\boldsymbol{D}(f)\boldsymbol{A}^{\dagger}(f)\big)^{\dagger} = \boldsymbol{A}(f)\boldsymbol{D}^{-1}(f) \tag{54}$$

with entries $\boldsymbol{W}_s^{\dagger}(f)_{(j,i)} = \frac{\alpha_{j,i}(f)}{d_i(f)}$, where $\boldsymbol{X}_{(j,i)}$ denotes the (j,i)th entry of $\boldsymbol{X}$.

With $\boldsymbol{W}_s^{\dagger}(f)$ in our disposal, and using (53), (54) we can form the following matrix

$$\boldsymbol{W}_{s,n}(f) = \begin{bmatrix} \frac{\alpha_{j_1,1}(f)}{d_1(f)} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \frac{\alpha_{j_I,I}(f)}{d_I(f)} \end{bmatrix} \boldsymbol{W}_s(f) = \begin{bmatrix} \alpha_{j_1,1}(f) & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \alpha_{j_I,I}(f) \end{bmatrix} \boldsymbol{D}^{-1}(f)\boldsymbol{D}(f)\boldsymbol{A}^{\dagger}(f) = \begin{bmatrix} \alpha_{j_1,1}(f) & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \alpha_{j_I,I}(f) \end{bmatrix} \boldsymbol{A}^{\dagger}(f) \tag{55}$$

Now, by multiplying (3) (we neglect the noise term) from the left with the matrix $\boldsymbol{W}_{s,n}(f)$ given in (55) we have

$$\begin{aligned} \boldsymbol{W}_{s,n}(f)\boldsymbol{x}(f,t) \approx diag\{[\alpha_{j_1,1}(f), \dots, \alpha_{j_I,I}(f)]^T\}\boldsymbol{A}^{\dagger}(f)\boldsymbol{A}(f)\boldsymbol{s}(f,t) = \\ diag\{[\alpha_{j_1,1}(f), \dots, \alpha_{j_I,I}(f)]^T\}\boldsymbol{s}(f,t) = [\alpha_{j_1,1}(f)s_1(f,t), \dots, \alpha_{j_I,I}(f)s_I(f,t)]^T \end{aligned} \tag{56}$$

Again, we have reached to the result of (44).

REFERENCES

[1] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest Point search in lattices," *IEEE Transactions on Information Theory*, vol. 48, no. 8, pp. 2201-2214, Aug. 2002.

[2] J.M.F. ten Berge and N.D. Sidiropoulos, "On uniqueness in CANDECOMP/PARAFAC," *Psychometrica*, vol. 67, no. 3, Sept. 2002

[3] R. Bro, N.D. Sidiropoulos, and G.B. Giannakis, "A fast least squares algorithm for separating trilinear mixtures," in *Proc. Int. Workshop Independent Component Analysis and Blind Signal Separation*, Aussois, France, Jan. 1999.

[4] J.D. Carroll and J, Chang, "Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition," *Psychometrica*, vol. 35, no. 3, pp. 283-319, 1970.

[5] A. Chan and I. Lee, "A new reduced-complexity sphere decoder for multiple antenna systems," in *Proc. ICC*, vol. 1, New York, pp. 460-464, Apr. 28-May 2 2002.

[6] A. Duel-Hallen, "A family of multiuser decision-feedback detectors for asynchronous code-division multiple-access channels," *IEEE Trans. on Communications*, vol. 43, issue 234, pp. 421-434, Feb. 1995.

[7] A. Gersho and R.M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, 1992.

[8] S. Van Gerven and D. Van Compernolle, "Signal separation by symmetric adaptive decorrelation: Stability, convergence, and uniqueness," *IEEE Transactions on Signal Processing*, vol. 43, pp. 1602-1612, July 1995.

[9] R.A. Harshman, "Foundation of the PARAFAC procedure: Model and conditions for an 'explanatory' multi-mode factor analysis," *UCLA Working Papers in Phonetics*, vol. 16, pp. 1-84, Dec. 1970.

[10] R.A. Harshman, "Determination and proof of minimum uniqueness conditions for PARAFAC1," *UCLA Working Papers in Phonetics*, vol. 22, pp. 111-117, 1972.

[11] S. Ikeda and N. Murata, "A method of ICA in time-ferquency domain," in *Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, Oct. 1997.

[12] M. Kawamoto, K. Matsuoka, and N. Ohnishi, "A method of blind separation for convolved nonstationary signals," *Neurocomputing*, vol. 22, pp. 157-171, 1998.

[13] J.B. Kruskal, "Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics," *Linear Algebra Applications*, vol. 18, pp. 95-138, 1977.

[14] T. Li and N.D. Sidiropoulos, "Blind digital signal separation using successive interference cancellation iterative least squares," *IEEE Transactions on Signal Processing*, vol. 48, no. 3, pp. 810-823, Apr. 2000.

[15] J. Luo, K. Pattipati, P. Willet, and F. Hasegawa, "Near-optimal multiuser detection in synchronous CDMA using probabilistic data association," *IEEE Commun. Lett.*, vol. 5, no. 8, pp. 361-363, Sep. 2001.

[16] W.-K. Ma, T.N. Davidson, K.M. Wong, Z.-Q. Luo, and P.-C. Ching, "Quasi-ML multiuser detection using semi-definite relaxation with application to synchronous CDMA," *IEEE Transactions on Signal Processing*, vol. 50, no. 4, pp. 912-922, Apr. 2002.

[17] K. Matsuoka and S. Nakashima, "Minimal distortion principle for blind source separation," in *Proceeding of ICA 2001 Conference*, pp. 722-727, Dec. 2001

[18] N. Mitianoudis and M. Davies, "Audio source separation of convolutive mixtures," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 5, pp. 489-497, Sep. 2003.

[19] L. Parra and C. Spence, "Convolutive blind separation of non-stationary sources," *IEEE Transactions on Speech and Audio Processing*, vol. 8, pp. 320-327, May 2000.

[20] PEACH multi-microphone database: *http://shine.itc.it*

[21] D.T. Pham and J. Cardoso, "Blind source separation of instantaneous mixtures of nonstationary sources," *IEEE Transactions on Signal Processing*, vol. 49, no. 9, pp. 1837-1848, Sep. 2001.

[22] D. Pham, K.R. Pattipati, P.K. Willet, and J. Luo, "A generalized probabilistic data association detector for multiple antenna systems," *IEEE Commun. Lett.*, vol. 8, no. 4, pp. 205-207, Apr. 2004.

[23] D.T. Pham, C. Serviere, and H. Boumaraf, "Blind separation of convolutive audio mixtures using nonstationarity, " in *Proceeding of ICA 2003 Conference*, Nara, Japan, Apr. 2003.

[24] K. Rahbar and J.P. Reilly, "A Frequency Domain Method for Blind Source Separation of Convolutive Audio Mixtures," *IEEE Transactions on Speech and Audio Processing*, vol. 13, pp. 832-844, Sep. 2005.

[25] Y. Rong, S.A. Vorobyov, A.B. Gershman, and N.D. Sidiropoulos, "Blind Spatial Signature Estimation via Time-Varying User Power Loading and Parallel Factor Analysis," *IEEE Transactions on Signal Processing*, vol. 53, pp. 1697-1710, May 2005.

[26] H. Sahlin and H. Broman, "MIMO signal separation for FIR channels: a criterion and performance analysis," *IEEE Transactions on Signal Processing*, vol. 48, no. 3, pp. 642-649, Mar. 2000.

[27] N.D. Sidiropoulos and R. Bro, "On the uniqueness of multilinear decomposition of N-way arrays," *J. Chemometrics*, vol. 14, pp. 229-239, 2000.

[28] N.D. Sidiropoulos, R. Bro, and G.B. Giannakis, "Parallel factor analysis in sensor array processing," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2377-2388, Aug. 2000.

[29] N.D. Sidiropoulos, G.B. Giannakis, and R. Bro, "Blind PARAFAC receivers for DS-CDMA systems," *IEEE Transactions on Signal Processing*, vol. 48, no. 3, pp. 810-823, Mar. 2000.

[30] P. Smaragdis, "Blind separation of convolved mixtures in the frequency domain," *Neurocomputing*, vol. 22, pp. 21-34, 1998.

[31] A. Stegeman, J.M.F. ten Berge, and L. De Lathauwer, "Sufficient conditions for uniqueness in CANDECOMP/PARAFAC and INDSCAL with random component matrices," *Psychometrika*, to appear.

[32] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Transactions on Information Theory*, vol. 45, no. 7, pp. 1639-1642, July 1999.

[33] R. Wang and G.B. Giannakis, "Approaching MIMO capacity with reduced-complexity soft sphere-decoding," in *Proc. WCNC*, Atlanta, GA, Mar. 21-25, 2004.

[34] W. Zhao and G.B. Giannakis, "Sphere decoding algorithms with improved radius search," in *Proc. WCNC*, Atlanta, GA, Mar. 21-25, 2004.

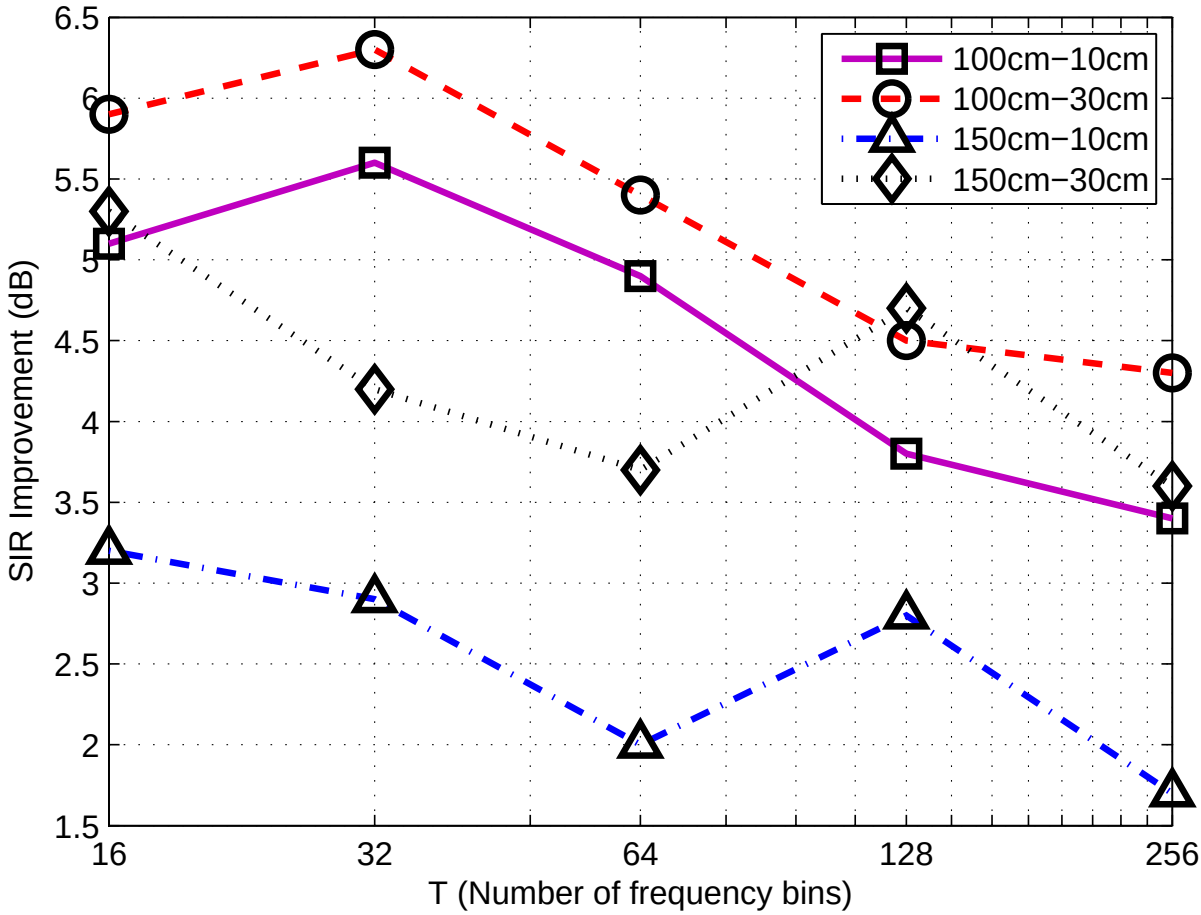

Fig. 2. Average SIR improvement versus number of frequency bins using DCE-PARAFAC-ILS.

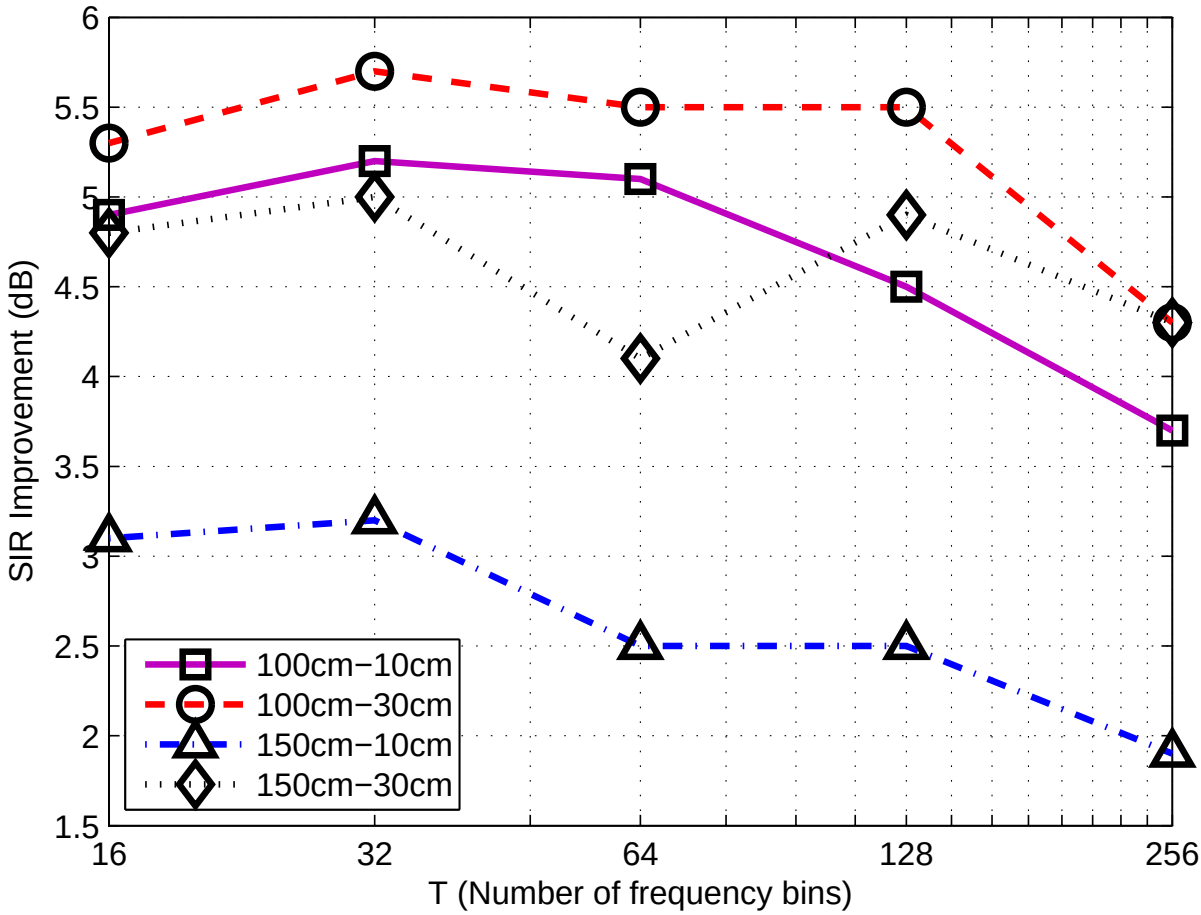

Fig. 3. Average SIR improvement versus number of frequency bins for ICE-PARAFAC-ILS.

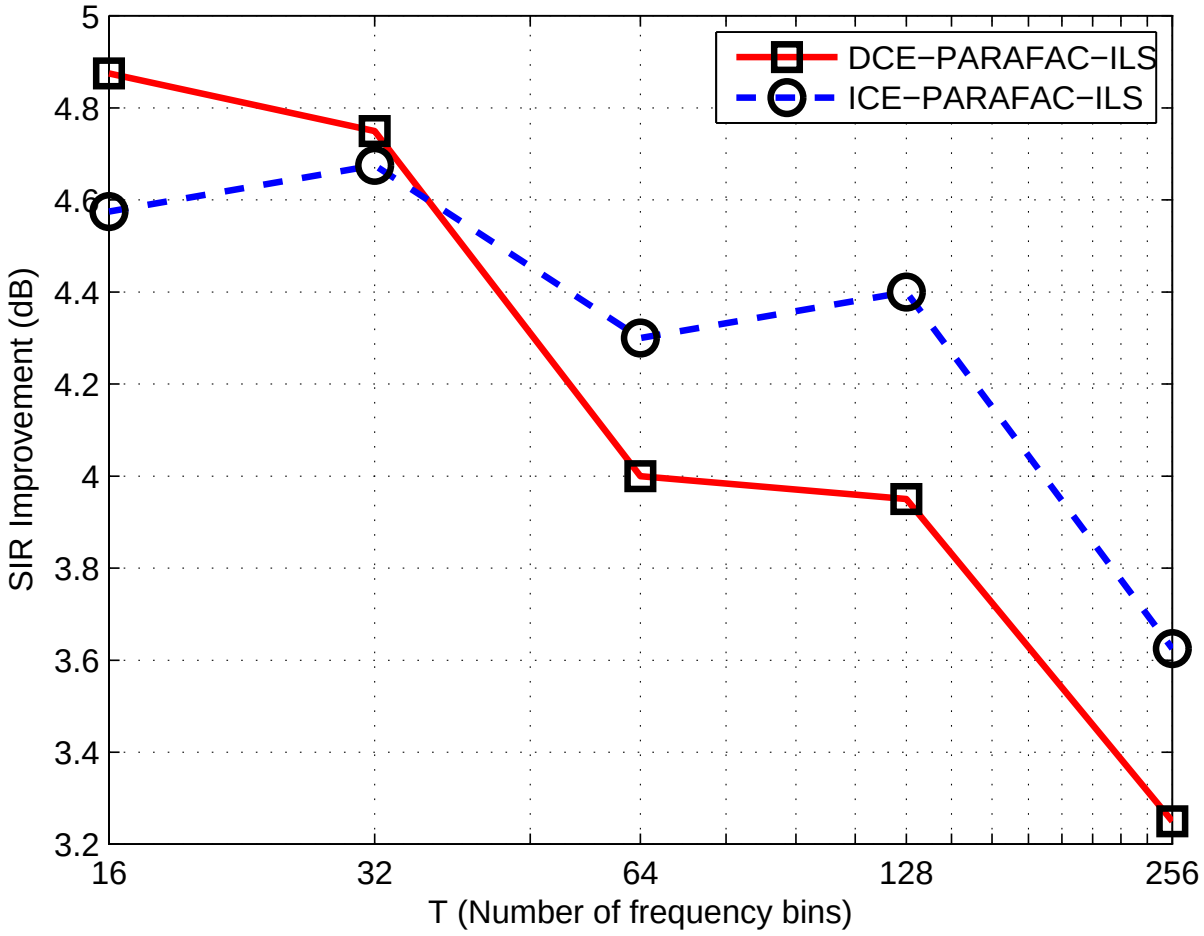

Fig. 4. Separation performance of our algorithms versus number of frequency bins, averaged over all geometric configurations

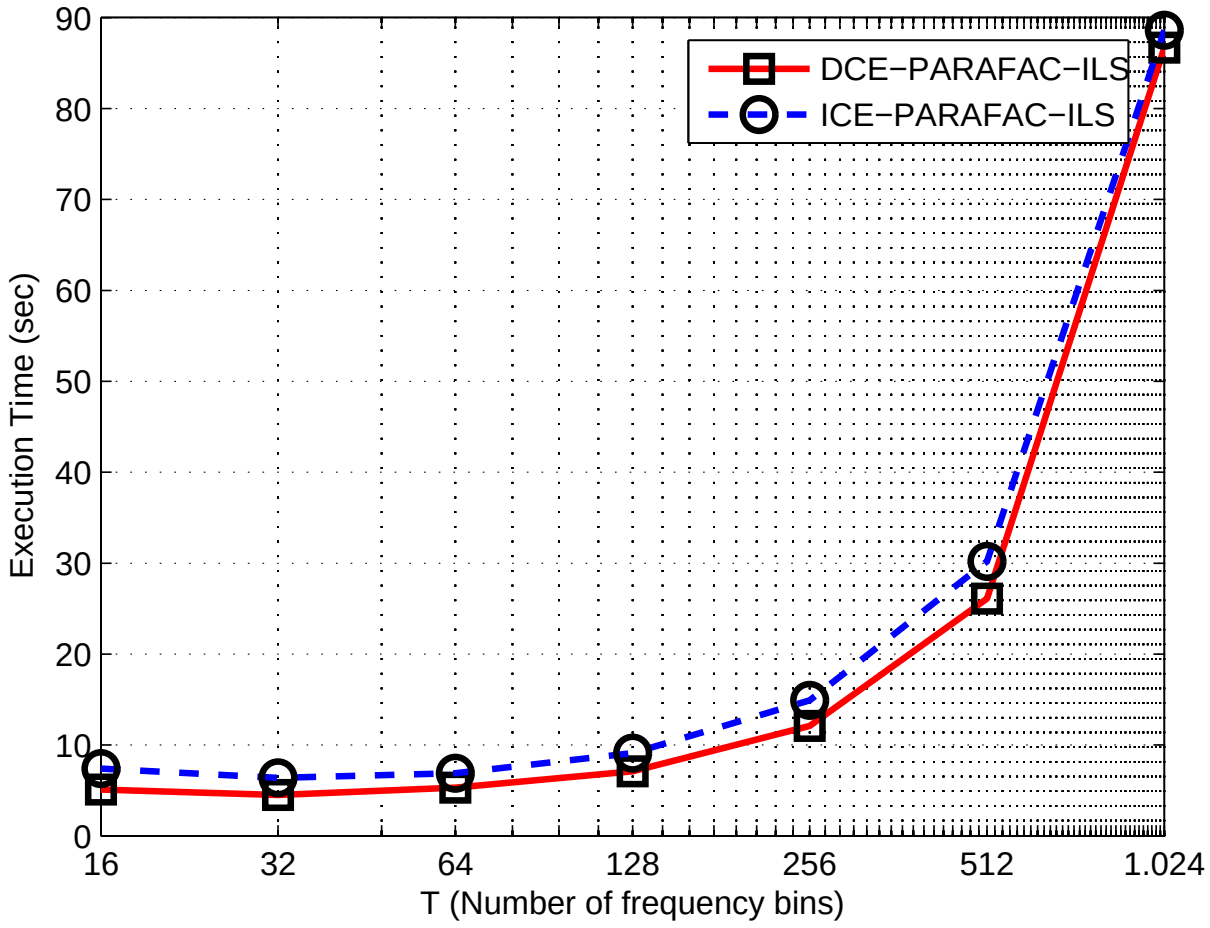

Fig. 5. Average execution time of our algorithms. The average length of the speech mixtures processed is about 300,000 samples.