PHASE DIFFERENCE AND TENSOR FACTORIZATION MODELS FOR AUDIO SOURCE SEPARATION

BY

JOHANNES TRAA

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2016

Urbana, Illinois

Doctoral Committee:

Assistant Professor Paris Smaragdis, Chair
Professor Mark Hasegawa-Johnson
Professor Yoram Bresler
Dr. Noah D. Stein, Massachusetts Institute of Technology

ABSTRACT

Audio source separation is a well-known problem in the speech community. Many methods have been proposed to isolate speech signals from a multichannel mixture. In this thesis, we will explore a number of techniques involving interchannel phase difference (IPD) features within a tensor factorization framework. IPD features can be extracted on a time-frequency (TF) grid and are a function of the phase characteristics of the mixing process. Thus, the ultimate goal is to form a clustering of these features and produce TF masks that can be used to perform the separation. We discuss various non-tensor-based methods that are capable of modeling linear and nonlinear IPD trends. Then, we discuss generalizations to both nonnegative and complex tensor factorizations (NTF, CTF). We show that each method performs best in certain circumstances and we conclude by saying that more work is needed to devise a generally superior approach.

ACKNOWLEDGMENTS

To all who have contributed.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Source separation is a classic problem in signal processing that has been approached from many different angles [1, 2, 3, 4]. Some successful methods include beamforming [2], matrix factorization [5], phase difference models [6, 7], and neural networks [8]. The general problem involves the inversion of a mixing system that takes multiple signals as input and produces as output one or more mixed signals. A successful separation algorithm is able to recover the original clean signals accurately. In the case of audio signals, single-channel methods attempt to model spectrotemporal properties of the signals. When multiple mixtures are available, as when a multichannel recording is captured, spatial information can be leveraged to enhance the separation.

In this work, we focus primarily on the case of a multichannel recording of spatially separated point sources with a compact microphone array [9]. In this scenario, phase difference features extracted from pairs of microphones can be used to identify each source's activity in time and frequency. The source-specific activity patterns are used to compute time-frequency (TF) masks that separate the sources from one of the mixtures. TF masking has been shown to be very effective for general audio signals and for speech mixtures in particular [10].

Spectrotemporal factorization-based techniques can be used to introduce additional structure into a model of the mixture signals. This can be implemented in both the single-channel [11] and multichannel [12] scenarios. We will discuss extensions of spatially-informed separation methods that use phase difference features and localization cues from the beamforming literature to both nonnegative and complex tensor factorizations. In this way, we leverage the combined modeling power of various methods.

The contributions of this thesis are:

- A discussion of interchannel phase difference (IPD) features and distortions of them that limit the effectiveness of linear models

- A qualitative and quantitative evaluation of a basic linear IPD model and novel extensions that account for nonlinearities
- A comparison of existing and novel tensor factorization models that leverage localization cues to perform separation

CHAPTER 2

IPD FEATURES

2.1 Feature Extraction

Denote the short-time Fourier transform (STFT) [9] with window size N of a recorded signal as $\mathbf{X}^i, i = 1, \ldots, M$, where $\mathbf{X}^i \in \mathbb{C}^{F \times T}$. M is the number of microphones and $F = N/2 + 1$ is the number of unique coefficients per frame. Interchannel logratio features are computed as:

$$y_{ft} = \log\left(\frac{X^1_{ft}}{X^2_{ft}}\right) \tag{2.1}$$

In an ideal, anechoic setting, a single source with STFT coefficients $S_{ft} \in \mathbb{C}^{F \times T}$ is recorded at the microphones with attenuations and delays that depend on the relative positions of the array and source. The logratio can be written as:

$$\begin{aligned} y_{ft} &= \log\left(\frac{a_1 e^{-j\omega d_1} S_{ft}}{a_2 e^{-j\omega d_2} S_{ft}}\right) && (2.2) \\ &= \log\left(\frac{a_1}{a_2}\right) - j\phi\left(\omega(d_1 - d_2)\right) && (2.3) \end{aligned}$$

where $\omega = 2\pi f/N$ is the radian frequency at the f^{th} frequency band, a_i and d_i are attenuation and delay values for the i^{th} microphone, and $\phi(x)$ is a wrapping function:

$$\phi(x) = \text{mod}(x + \pi, 2\pi) - \pi \tag{2.4}$$

We will focus on the special case of a compact microphone array for which level (loudness) differences are relatively uninformative (especially in noisy

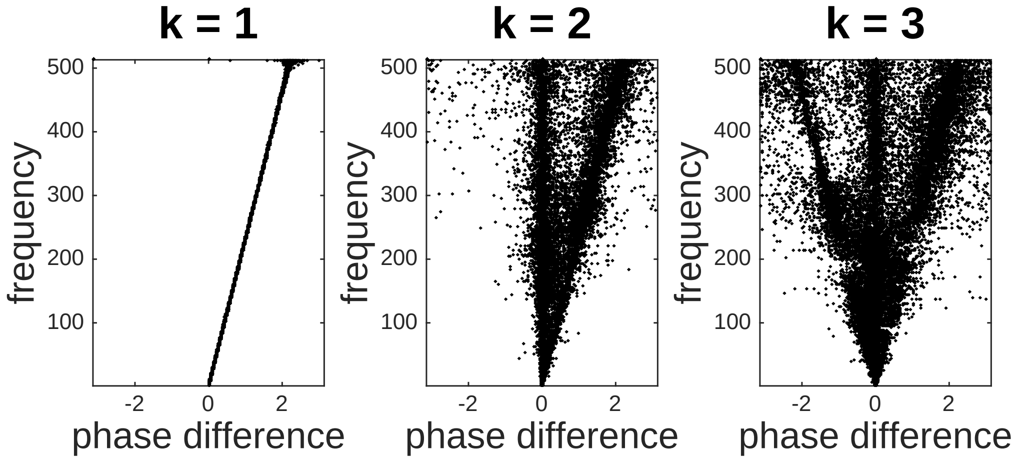

Figure 2.1: IPD feature sets for various numbers of speech sources in a simulated, two-channel, anechoic mixture. At each frequency, only the 50 features with the largest corresponding STFT magnitudes are shown. When multiple sources are present, the TF-disjointness of speech signals results in an approximate superposition of the source-specific IPD lines.

conditions). So, we define the interchannel phase difference (IPD) feature as:

$$\delta_{ft} = -\mathrm{Im}\left[y_{ft}\right] = \phi\left(\angle X^2_{ft} - \angle X^1_{ft}\right) \tag{2.5}$$

In our ideal, one-source scenario, we simply have that $\delta_{ft} = \phi\left(\omega(d_1 - d_2)\right)$, which is a wrapped-linear function of frequency. When $K > 1$ sources are present, IPD features are still meaningful as long as a TF disjointness assumption holds:

$$\forall f, t, k \neq k' \quad \left|S^k_{ft}\right| \cdot \left|S^{k'}_{ft}\right| \approx 0 \tag{2.6}$$

This says that each TF bin is occupied by at most one source. If this is the case, the IPD features associated with the k^{th} source will exhibit a unique wrapped-linear pattern as long as the interchannel delays are unique among the sources. This can typically be ensured with an appropriate array geometry and the assumption that the sources are spatially separated.

Figure 2.1 illustrates IPD feature sets for various numbers of sources mixed in a simulated environment. We can see that, in practice, perfect disjointness does not hold, but for speech signals, it holds sufficiently to be able to distinguish the source patterns. However, this may not hold quite as well for music examples where instruments play together and harmonize.

2.1.1 Effect of STFT Parameters

In this thesis, we use a window size of 1024, hop size of 256, and Hann analysis/synthesis windows. We also downsample all recordings to 16 kHz. These are fairly common choices in source separation applications [13, 14]. The main way in which these choices affect the results of the algorithms in this thesis is through the extraction of raw features. As an example, we can consider the effects on IPD features, keeping in mind that speech signals are generally non-stationary in both time and frequency.

A large window size will capture more information in a single frame, leading to high frequency resolution and low time resolution. This is beneficial for source separation when the signals are more disjoint in frequency than in time. A small window size will have the opposite effect. In terms of source separation quality, very large or very small window sizes are undesirable because they make it more difficult to reconstruct the separated sources. This is because, in either case, we are boosting the impact of errors in either frequency or time. Speech is highly variable in the TF plane, so an intermediate window size helps to strike a healthy trade-off. Similarly, a large hop size (relative to the window size) allows masking errors to have an all-or-nothing impact on the separation quality. A small relative hop size introduces unpleasant artifacts when the masking is uncertain because many frames have to cooperate to construct the signal. As with the window size, an intermediate choice is best. A hop size one quarter the size of the window is a standard choice and has various good properties when combined with a Hann window.

A non-rectangular window is used to avoid ringing artifacts from discontinuities at the boundary of each analysis frame. The Hann window strikes a balance between suppressing these artifacts and maintaining the original information in the analysis frame. It also satisfies criteria necessary for a perfect reconstruction of the mixture under no separation [15] when the window size is a power of two times the hop size. Windows that satisfy this criterion generally lead to better reconstructions after separation.

A theoretical analysis of the effects of parameter choices in the STFT on the performance of the algorithms in this thesis as well as an experimental validation of this analysis are left as an open problem for future research.

2.2 Source Localization and Separation

IPD features can be used for both localization and source separation. When one source is present, the features with nonnegligible energy tend to lie near a linear function of frequency. To localize the source, we can simply scan over a range of directions and determine which one the features agree with most. However, when K sources are present, one must check all K-tuples of directions. To avoid having to perform this exhaustive search, which can be quite expensive in 3-D localization problems and with many sources, we can interpret this search as an optimization problem to be resolved with an appropriately designed solver.

Given estimates of the source directions, there are various methods for performing source separation. One approach would be to cluster the TF bins according to how well they agree with each source's direction model. Given a clustering, we form TF masks to apply element-wise to the STFT of one of the recorded mixtures to reconstruct the individual sources.

Typically, however, the source directions are not known a priori and must be estimated jointly with the clustering. In subsequent chapters, we will discuss various methods for doing so.

2.3 Nonlinearities

Nonlinearities originate from various sources including spatial aliasing, reverberation, and channel mismatch. Each of these has a unique effect on the properties of IPD features. We will consider each in turn to better understand how to design appropriate models.

2.3.1 Spatial aliasing

In the context of array signal processing, spatial aliasing refers to the ambiguity in the direction of arrival (DOA) of a source as a result of a large microphone spacing and high sampling rate.[1] We can see the effects of spatial aliasing by noting that, upon feature extraction, IPD values can only be recovered up to the interval $[-\pi, \pi]$. Any sufficiently long delay in the arrival times of a signal at a pair of microphones will result in phase wrapping. Strategies to account for this include explicit modeling of the circular-linear nature of the data and representing IPD features as unit-norm complex values.

2.3.2 Reverberation

In an anechoic chamber, only the direct-path signal is observed. This is the signal that propagates from the source to each microphone in straight lines. Reverberation occurs when additional copies of the signal that reflect off of boundaries (walls, furniture, windows, etc.) are recorded. Each recorded reflection is a copy of the original signal after some filtering. A simple but powerful model for this filtering describes each reflection as a delayed and attenuated copy of the original signal. Thus, we can fully characterize the room impulse response (RIR) as a set of delay-attenuation pairs[2] (a, d). We observe:

$$y_{ft} = \log\left(\frac{\sum\limits_{r=1}^{R} a_{1r} e^{-j\omega d_{1r}} S_{ft}}{\sum\limits_{r=1}^{R} a_{2r} e^{-j\omega d_{2r}} S_{ft}}\right) = \log\left(\frac{a_{11} e^{-j\omega d_{11}}}{a_{21} e^{-j\omega d_{21}}}\right) + \log\left(\frac{1 + \sum\limits_{r=2}^{R} b_{1r} e^{-j\omega e_{1r}}}{1 + \sum\limits_{r=2}^{R} b_{2r} e^{-j\omega e_{2r}}}\right) \tag{2.7}$$

with relative attenuation and delay for the i^{th} microphone and r^{th} reflection defined as:

$$b_{ir} = \frac{a_{ir}}{a_{i1}} \quad , \quad e_{ir} = d_{ir} - d_{i1} \tag{2.8}$$

So we can see that the case with reverb is similar to the case with no reverb except that there is an additive perturbation that is a nonlinear function of

[1] To give some perspective, aliasing begins to occur for a signal lined up with the microphones and recorded at 16 kHz when the microphone spacing increases to 1 cm.

[2] This assumes that the longest delay is within one STFT analysis window. It holds approximately when late reflections are strongly attenuated.

the relative attenuations and delays. The result is a sinusoid-like wobble in the IPD data over frequency that depends very strongly on the room characteristics and array/source positions. This is because the attenuations and delays are heavily influenced by these factors. If the direct path has an attenuation coefficient that is much larger than that of competing arrivals, the linear term dominates and the wobble is negligible. For extremely small rooms or otherwise in situations with strong early reflections (e.g. off of an object holding the array), the nonlinearity may be quite strong.

2.3.3 Channel mismatch

Ideally, our microphones should have identical frequency responses. However, in practice, this is not the case because of many real-world factors. It is easy to see how this will affect IPD features by including additional terms $\gamma_{if} \in \mathbb{C}$ in (2.1) to account for the channel responses:

$$\delta_{ft} = -\text{Im}\left[\log\left(\frac{a_1 e^{-j\omega d_1} S_{ft}\gamma_{1f}}{a_2 e^{-j\omega d_2} S_{ft}\gamma_{2f}}\right)\right] = \phi\left(\omega(d_1 - d_2) + (\angle\gamma_{2f} - \angle\gamma_{1f})\right) \quad (2.9)$$

The phase difference between the frequency responses of the channels perturbs the feature set. In some cases, this can introduce significant nonlinearities.

2.3.4 Illustration of nonlinearities

Figure 2.2 demonstrates the effects of these nonlinearities on a simulated one-source, two-microphone mixture. As expected, early reflections introduce the largest deviations from a wrapped-line model while heavily-attenuated, late reflections introduce minor deviations that may not be distinguishable from noise in practice. Channel mismatch is particularly problematic when it introduces strong bends in the IPD function at low frequencies. This is because the most salient speech information resides in this range and a straight-line model will fail to properly capture the structure of the data.

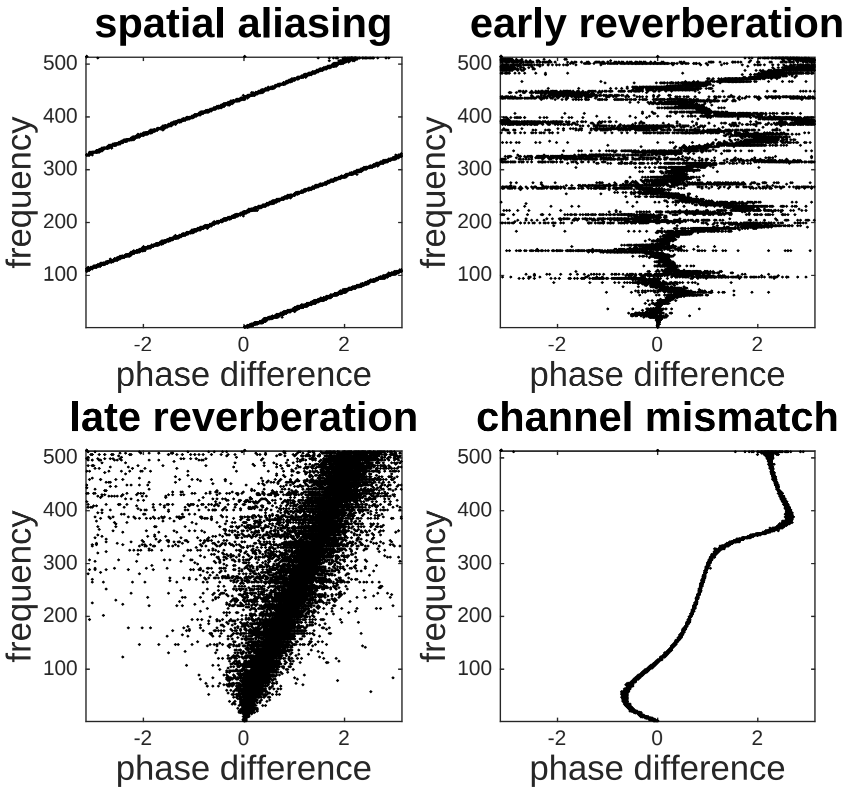

Figure 2.2: Simulated IPD feature sets exhibiting various nonlinearities. (Top left) spatial aliasing due to large microphone spacing (10 cm). (Top right) squiggles due to strong early reflections (array positioned 30 cm from corner of room). (Bottom left) noise-like pattern due to overlap of many late reflections (source near array, both far from walls). (Bottom right) squiggle due to mismatched microphone frequency responses (random IIR filters applied to either channel).

2.4 Comparison with Narrowband Beamforming

Consider a mixture of narrowband signals. This corresponds to a single frequency band f in an STFT. If the signals are perfectly disjoint in time, we will be able to isolate the IPD features corresponding to a target source and exactly separate the signals. If disjointness does not hold, we will have more difficulty. We investigate the potential of using IPD features through a comparison with traditional beamforming techniques in the narrowband case. In particular, we compare with delay-and-sum (D&S) and linearly-constrained minimum-variance (LCMV) beamformers. For simplicity, we assume the true source DOAs are known.

A beamformer is a linear spatial filter used to enhance one or more target signals in a multichannel mixture. The (single-source) D&S beamformer simply delays all the recorded signals so that the instances of the target signal in all the recordings are time-aligned and computes the sum. This will reinforce the target signal more so than other uncorrelated signals/noise. The (multiple-source) LCMV beamformer actively blocks non-target directional signals with known DOAs. We will discuss the details of these spatial filters in Chapter 4 in the context of tensor factorizations.

Generally speaking, we are interested in the signal-to-interference-and-noise-ratio (SINR):

$$SINR = 10 \log_{10} \left(\frac{\sum_t \mathbf{E}\left[|s_t^1|^2\right]}{\sum_t \mathbf{E}\left[|s_t^2 + n_t|^2\right]} \right) \tag{2.10}$$

where s_t^j is the DFT coefficient of source j at time t and n_t is the noise coefficient. We consider the following illustrative cases for a 2-channel array in ideal, anechoic conditions.

Target Signal and Uncorrelated White Gaussian Noise

In this case, the SINR reduces to an signal-to-noise (SNR) measure:

$$SNR = 10 \log_{10} \left(\frac{\sum_t |s_t^1|^2}{\sum_t \mathbf{E}\left[|n_t|^2\right]} \right) \tag{2.11}$$

Without disjointness, a D&S beamformer gives 3 dB of improvement in the SNR. We can see that this is the case by replacing s_t^1 with $2s_t^1$ in (2.11). An IPD clustering method will produce mixed results because the features are contaminated with the phase information of the noise. When perfect disjointness holds, the beamformer still achieves +3 dB, but an IPD masking procedure can give a much greater dB improvement since it can aggressively mask out noise frames.

Target Signal and Interference

In this case, the SINR reduces to an signal-to-interference (SIR) measure:

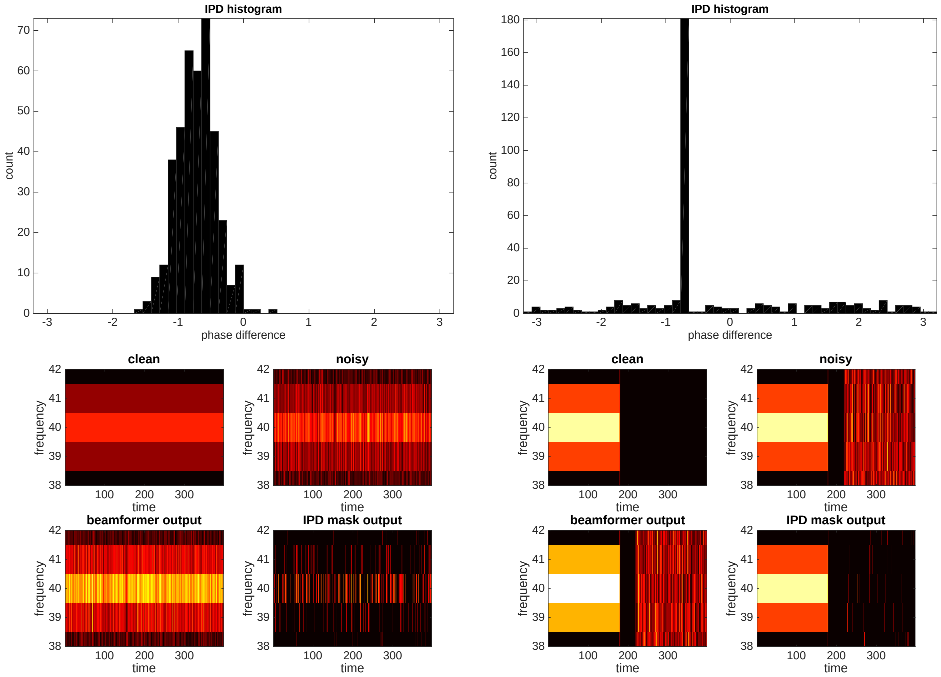

Figure 2.3: Comparison of IPD clustering/masking and beamforming for a single sinusoid in white, Gaussian noise. IPD histograms and separation results are shown for the non-disjoint (left column) and disjoint (right column) cases. In the disjoint case, the IPD-based mask aggressively blocks the noise.

$$SIR = 10\log_{10}\left(\frac{\sum_t |s_t^1|^2}{\sum_t |s_t^2|^2}\right) \tag{2.12}$$

Without disjointness, an LCMV beamformer with perfect knowledge of the source DOAs gives $+\infty$ dB. An IPD clustering method will have difficulties in bins with strong overlap. When perfect disjointness holds, both give $+\infty$. When the source DOAs are not known perfectly, the LCMV performance will reduce while the clustering result may stay very good. The rationale for this is the same as in the previous case.

Figures 2.3-2.5 illustrate various scenarios in a narrowband setting. In one method, we applied a beamformer and in the other, we created a binary mask that is 1 for any features within $2\pi/50$ of true IPD value and 0 otherwise. We can see that when disjointness does not hold, it is difficult to distinguish

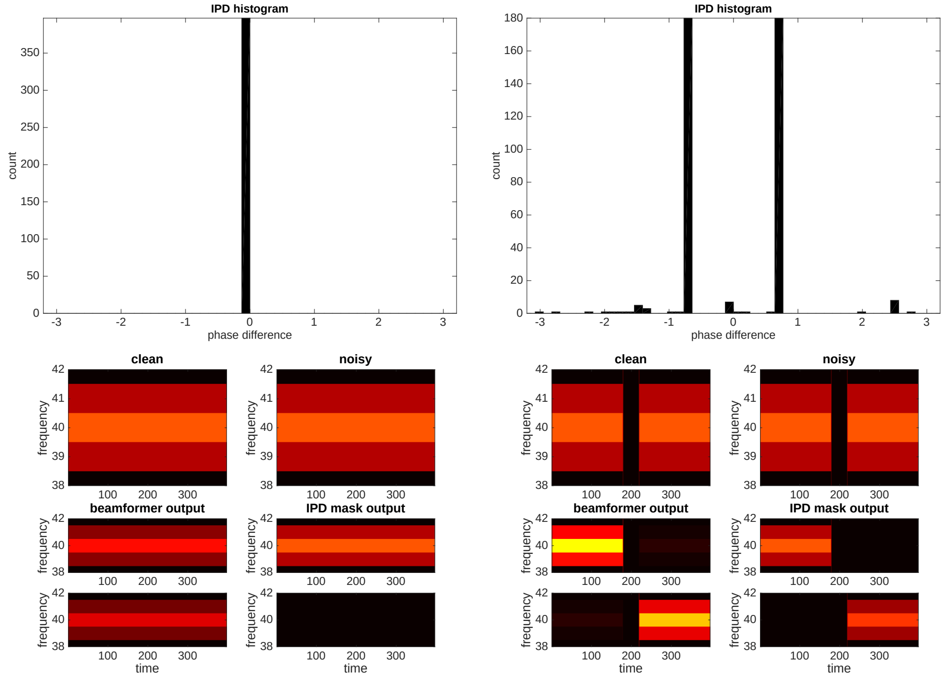

Figure 2.4: Comparison of IPD clustering/masking and beamforming for a mixture of two sinusoids. IPD histograms and separation results are shown for the non-disjoint (left column) and disjoint (right column) cases. In the disjoint case, the IPD-based mask aggressively blocks the interferer.

directional signals from each other and from noise. However, when disjointness does hold, a masking approach can be very powerful. We observe that additive noise effectively smears out the phase difference values, suggesting that an appropriate distribution can be used to model noisy IPD features in each frequency band.

One important difference between these methods is that beamforming involves linear processing while masking corresponds to nonlinear processing. Nonlinear methods are more general and can take advantage of additional knowledge such as disjointness. In speech mixtures, we often observe approximate disjointness in the TF plane. Thus, we are justified in pursuing IPD-based separation algorithms.

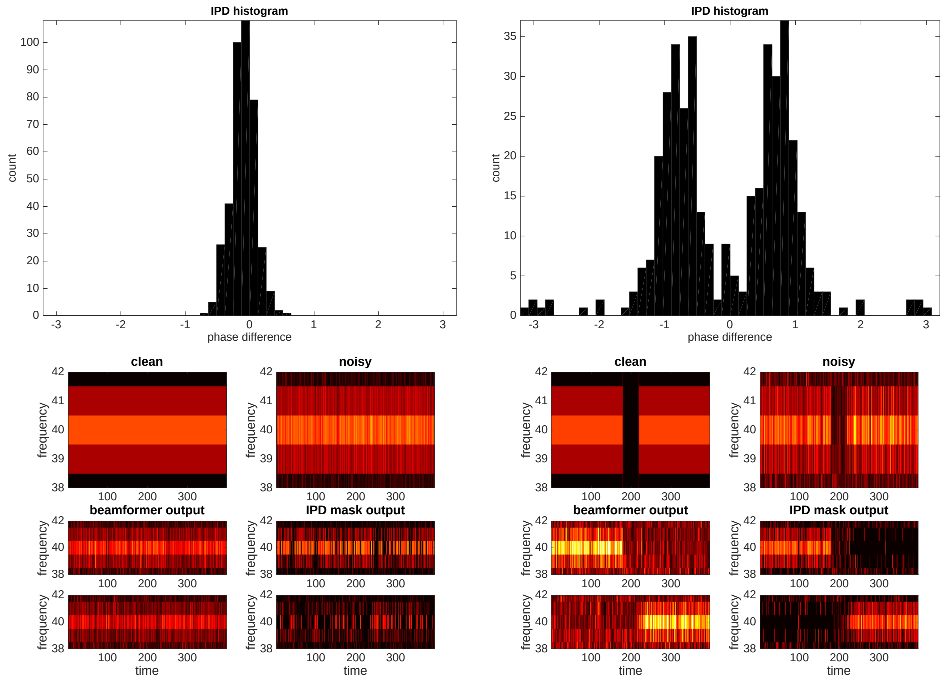

Figure 2.5: Comparison of IPD clustering/masking and beamforming for a mixture of two sinusoids in white, Gaussian noise. IPD histograms and separation results are shown for the non-disjoint (left column) and disjoint (right column) cases. In the disjoint case, the IPD-based mask aggressively blocks both the noise and interferer.

CHAPTER 3

IPD MODELS

In this chapter, we will discuss approaches to modeling IPD features for the purpose of source separation. In general, this does not necessarily imply that we must localize the sources in the process. This is because we can perform source separation using just a clustering of the features. In this chapter, we will see that we can accomplish this clustering with simple assumptions that do not depend on an explicit relationship between the features and source locations. Although the directional nature of the target signals is crucial for the clustering, we need not consider a mapping from a learned model to source directions.

The methods we will look at are the Degenerate Unmixing Estimation Technique (DUET) [6, 7], Random Sample Helix Consensus (RANSHAC) [16], the Mean-Locked Mixture of Wrapped Gaussians (ML-MoWG) model [17], and the Wrapped Cubic Regression Spline (WCRS) model [18]. The first assumes a non-wrapped IPD model, the next two generalize this to the wrapped-linear case, and the last relaxes this assumption to fit a wrapped piece-wise cubic function. We will look at them in the order of increasing complexity. As might be expected, more complex models are more difficult to fit successfully to a novel data set. Thus, in practice, it is generally useful to fit these models in order of increasing complexity, translating learned parameters appropriately at each stage. Throughout this chapter, we assume that a single feature set consisting of iPD-frequency tuples is extracted per multichannel recording. To conclude, we will compare the performance of all these models.

3.1 DUET

One significant contribution in the field of source separation is the Degenerate Unmixing Estimation Technique (DUET) [6, 7]. In this approach, inter-channel phase and level difference (IPD, ILD) features extracted from a pair of microphones are clustered to construct binary time-frequency masks. If no spatial aliasing occurs, the phase difference features can be normalized by ω and clustered using, for example, k-means.

In the absence of reverberation and source overlap in the time-frequency plane, this approach has been shown to be very successful. However, it (1) fails to leverage a wealth of information present in the magnitude spectrograms of the mixtures and (2) does not accurately represent the data when reverberation, aliasing, microphone mismatch, and other effects are present. Generally speaking, these factors produce nonlinearities in the features as a function of frequency. We seek to generalize this approach with more expressive and robust modeling techniques.

3.2 Random Sample Helix Consensus

One extension of the DUET algorithm is a combination of the Random Sample Consensus (RANSAC) [19] algorithm and DUET. RANSAC was first proposed in the context of computer vision where the problem is to identify the parameters of a simple model in the presence of many outliers. Groups of feature vectors are sampled at random from a data set and each group is used to propose a possible fit of the model. Each such candidate is compared with the entire data set to verify a good fit and the best model is reported. For example, if a line is to be fit, each group contains two data points. It can be shown that even in the presence of a large proportion of outliers, a relatively small number of groups must be sampled to learn the correct model with a high probability of success.

In the presence of aliasing, the un-normalized phase difference features δ_{ft} associated with a single source lie near a line that has been wrapped to the interval $[-\pi, \pi]$. Thus, source separation is apparently reduced to a problem of multimodal circular-linear regression. RANSHAC[1] [16] iteratively applies

[1] The 'H' stands for helix. When circular-linear data is visualized on a cylinder, the

Algorithm 1 RANSHAC: RANSAC for fitting multiple wrapped lines

```
Inputs:  $\mathbf{\Delta} = \{\boldsymbol{\delta}_i\}$ : $N$ IPD data points
         $K$ : number of wrapped lines to fit
Outputs: $\widehat{\boldsymbol{\alpha}} = \{\widehat{\alpha}_j\}$ : $K$ slopes
    $\mathbf{Y} = M$ samples from $\mathbf{\Delta}$ selected uniformly at random
    $\mathbf{I} = \mathbf{0}^{N \times M}$
    for $m = 1 : M$ do
        Fit line with slope $\alpha_m$ to $Y_m$
        $\mathbf{I}(i, m) = 1$ , $\forall i$ s.t. $\boldsymbol{\delta}_i$ is inlier of line with slope $\alpha_m$
    end for
    $\widehat{\boldsymbol{\alpha}} = \{\}$
    $A = \{1, \ldots, N\}$
    for $j = 1 : K$ do
        $\widehat{m} = \operatorname*{argmax}_m \sum_{i \in A} \mathbf{I}(i, m)$
        $\widehat{\boldsymbol{\alpha}} = \widehat{\boldsymbol{\alpha}} \cup \alpha_{\widehat{m}}$
        $A = A \setminus \{i : \mathbf{I}(i, \widehat{m}) = 1\}$
    end for
    return $\widehat{\boldsymbol{\alpha}}$
```

the RANSAC algorithm to this problem. This is computationally efficient and is capable of handling spatial aliasing. It has also been extended to larger arrays that can make use of ILD features [20].

The pseudocode for RANSHAC is given in Algorithm 1 and an illustration is shown in Figure 3.1. Only a single data point is required to propose a candidate wrapped line. At the beginning of the algorithm, a number of IPD features are sampled uniformly at random from the data set. Wrapped-line candidates are fit through these points and the origin. Then, an inlier count is computed for each candidate based on how many points are within a window of constant width across frequency. The highest-scoring candidate is chosen as the first line and its inliers are removed from the dataset. This process is repeated until k lines have been chosen. This procedure has been shown to be successful even in the presence of many outliers. Figure 3.2 shows examples of real-world 2-channel recordings where RANSHAC works very well.

wrapped lines form helices.

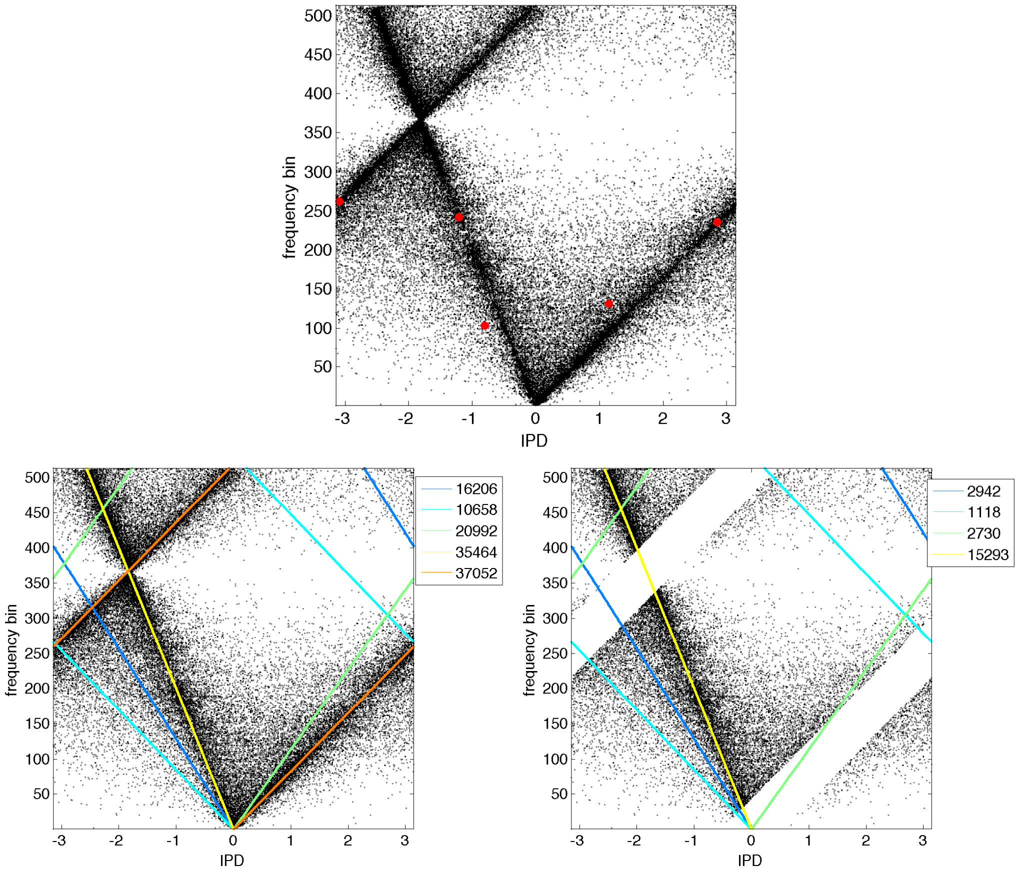

Figure 3.1: Example of sequential RANSAC for wrapped line-fitting. (Top) IPD data with 5 RANSAC samples overlaid. (Bottom left) First iteration showing candidate wrapped lines and their inlier counts. (Bottom right) Second iteration after removal of the inliers of the first model.

3.3 Mean-Locked Mixture of Wrapped Gaussians

RANSHAC has more modeling power than the original DUET approach, but it relies on random sampling and is not guaranteed to find a statistically optimal solution. So, we turn to a more principled probabilistic formulation. In the Mean-Locked Mixture of Wrapped Gaussians (ML-MoWG) model [17], we assume that the observed data is generated by a mixture of wrapped Gaussians in each frequency band. However, we introduce the constraint that the means corresponding to each source are tied across frequency via a wrapped linear function. In other words, each source is represented by a distribution with a wrapped line mean (parameterized by a scalar slope value α_k) and frequency-dependent variance and mixing weight parameters σ_{kf}^2 and π_{kf}.

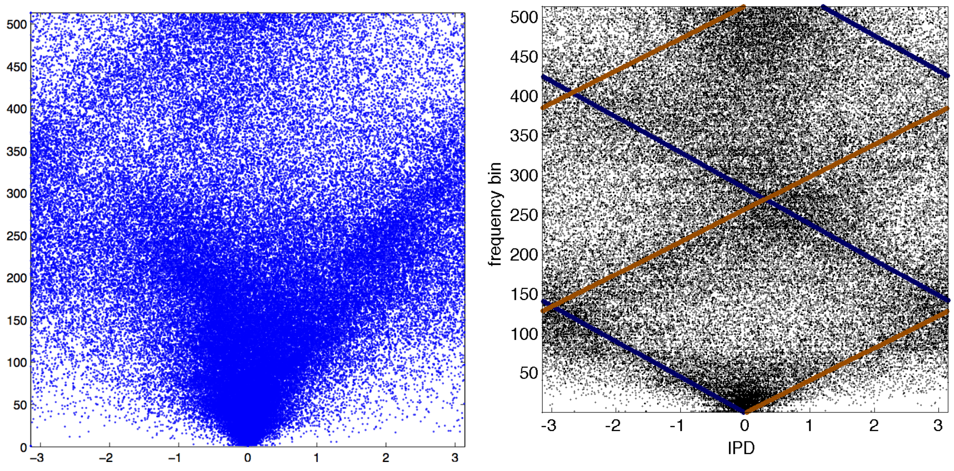

Figure 3.2: IPD datasets extracted from real-world stereo mixtures of two speakers. (Left) recording with a hearing aid in an office. (Right) recording with a low-quality microphone array in a stairwell with wrapped-line fits overlaid.

The ML-MoWG pdf for a single frame of a two-channel mixture is:

$$p\left(\boldsymbol{\delta}\,;\,\boldsymbol{\alpha},\boldsymbol{\sigma}^2,\boldsymbol{\pi}\right)=\prod_{f=1}^{F}\sum_{k=1}^{K}\pi_{kf}\,WN\left(\delta_f\,;\,\alpha_{kf}\,,\,\sigma_{kf}^2\right) \tag{3.1}$$

where the wrapped Gaussian distribution [21] is given as:

$$\mathcal{WN}\left(x\,;\,\mu,\sigma^2\right)=\sum_{l=-\infty}^{\infty}\mathcal{N}\left(x\,;\,\mu+2\pi l,\sigma^2\right)\ ,\ x\in[-\pi,\pi] \tag{3.2}$$

and arises from applying (2.4) to $x\sim\mathcal{N}\left(\mu,\sigma^2\right)$. We assume that the IPD features are independent across STFT frames to write the associated likelihood over an entire data set as:

$$\mathcal{L}\left(\boldsymbol{\delta}_{1:T}\,;\,\boldsymbol{\alpha},\boldsymbol{\sigma}^2,\boldsymbol{\pi}\right)=\prod_{t=1}^{T}\prod_{f=1}^{F}\sum_{k=1}^{K}\pi_{kf}\,WN\left(\delta_{ft}\,;\,\alpha_{kf}\,,\,\sigma_{kf}^2\right) \tag{3.3}$$

The EM algorithm is applied to iteratively learn the parameters (see Algorithm 2). Although this is guaranteed to converge to a local optimum of the likelihood, the noisy and wrapped nature of the data results in the presence of many local optima. To ensure that we find a good solution, the RAN-

Algorithm 2 EM for fitting a mixture of mean-locked wrapped Gaussians

E step

$$\eta_{tjfl} = \frac{\mathcal{N}\left(\delta_{f,t}\,;\widehat{\alpha}_j f+2\pi l\,,\widehat{\sigma}^2_{jf}\right)\,\widehat{\pi}_j}{\sum\limits_{j=1}^{K}\sum\limits_{f=1}^{D}\sum\limits_{l=-\infty}^{\infty}\mathcal{N}\left(\delta_{f,t}\,;\widehat{\alpha}_j f+2\pi l\,,\widehat{\sigma}^2_{jf}\right)\,\widehat{\pi}_j}$$

M step

$$\widehat{\alpha}_j = \frac{\sum\limits_{t=1}^{T}\sum\limits_{f=1}^{D}\sum\limits_{l=-\infty}^{\infty}\frac{f\left(\delta_{f,t}-2\pi l\right)}{\widehat{\sigma}^2_{jf}}\,\eta_{tjfl}}{\sum\limits_{t=1}^{T}\sum\limits_{f=1}^{D}\sum\limits_{l=-\infty}^{\infty}\frac{f^2}{\widehat{\sigma}^2_{jf}}\,\eta_{tjfl}}$$

$$\widehat{\sigma}^2_{jf} = \frac{\sum\limits_{t=1}^{T}\sum\limits_{l=-\infty}^{\infty}\left(\delta_{f,t}-\widehat{\alpha}_j f-2\pi l\right)^2\eta_{tjfl}}{\sum\limits_{t=1}^{T}\sum\limits_{l=-\infty}^{\infty}\eta_{tjfl}}$$

$$\widehat{\pi}_j = \frac{1}{T}\sum_{t=1}^{T}\sum_{f=1}^{D}\sum_{l=-\infty}^{\infty}\eta_{tjfl}$$

SHAC algorithm can be used to quickly initialize EM. Figure 3.3 illustrates an example of an ML-MoWG fit in this way.

3.4 Wrapped Cubic Regression Spline

All the methods so far failed to address the presence of nonlinearities other than wrapping due to aliasing. To account for this, one can fit a Wrapped Cubic Regression Spline (WCRS) [18] to the IPD features. This is a convenient approach because splines are fairly general and simply parameterized. We first show how a spline is fit to a non-wrapped dataset and then extend this to the wrapped case.

3.4.1 Regression spline

A cubic spline is a twice-differentiable, piece-wise polynomial defined with respect to anchor points x_m , $m = 0, \ldots, M-1$. Each polynomial section is defined as:

$$y\left(f\,;\,\mathbf{a}_m\right) = a_{m0}\left(f-x_m\right)^3 + a_{m1}\left(f-x_m\right)^2 + a_{m2}\left(f-x_m\right) + a_{m3} \tag{3.4}$$

$$x_m \leq f \leq x_{m+1} \tag{3.5}$$

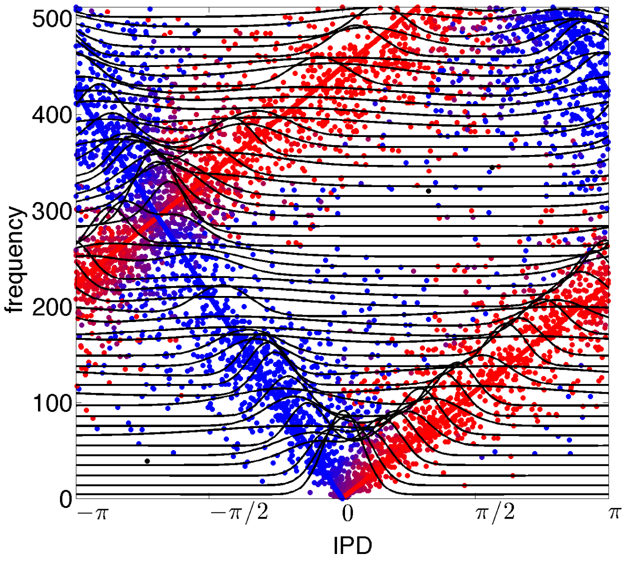

Figure 3.3: Two-component, mean-locked mixtures of wrapped Gaussians fit to IPD data with EM. The data is colored according to its posterior probability and 50 of the mixtures are superimposed.

where $\mathbf{a}_m \in \mathbb{R}^{4\times 1}$, $m = 0, \ldots, M-2$, denotes the parameters for the m^{th} section. We also have smoothness constraints at each anchor point to ensure that the values and first two derivatives of adjacent sections are equal:

$$y\left(f\,;\,\mathbf{a}_m\right)\Big|_{f=x_{m+1}} = y\left(f\,;\,\mathbf{a}_{m+1}\right)\Big|_{f=x_{m+1}} \tag{3.6}$$

$$\frac{\partial y\left(f\,;\,\mathbf{a}_m\right)}{\partial f}\Big|_{f=x_{m+1}} = \frac{\partial y\left(f\,;\,\mathbf{a}_{m+1}\right)}{\partial f}\Big|_{f=x_{m+1}} \tag{3.7}$$

$$\frac{\partial^2 y\left(f\,;\,\mathbf{a}_m\right)}{\partial f^2}\Big|_{f=x_{m+1}} = \frac{\partial^2 y\left(f\,;\,\mathbf{a}_{m+1}\right)}{\partial f^2}\Big|_{f=x_{m+1}} \tag{3.8}$$

We also enforce derivative constraints at the spline endpoints for stability:

$$\frac{\partial y\left(f\,;\,\mathbf{a}_0\right)}{\partial f}\Big|_{f=x_0} = 0 \tag{3.9}$$

$$\frac{\partial y\left(f\,;\,\mathbf{a}_{M-2}\right)}{\partial f}\Big|_{f=x_{M-1}} = 0 \tag{3.10}$$

We can solve for the parameters via the linearly-constrained quadratic optimization problem:

$$\min_{\mathbf{a}_{(0)},\ldots,\mathbf{a}_{(M-2)}} \quad \sum_{i=0}^{N-1} \left(\delta_i - y\left(f_i\,;\,\mathbf{a}_{(i)}\right)\right)^2 \tag{3.11}$$

$$s.t. \quad (3.6) - (3.10) \tag{3.12}$$

where $\mathbf{a}_{(i)}$ denotes the parameters of the spline section satisfying (3.5) for f_i. In matrix-vector form, we have:

$$\min_{\mathbf{a}} \quad (\boldsymbol{\delta} - \mathbf{X}\,\mathbf{a})^\top (\boldsymbol{\delta} - \mathbf{X}\,\mathbf{a}) \tag{3.13}$$

$$s.t. \quad \mathbf{G}\,\mathbf{a} = \mathbf{0} \tag{3.14}$$

where

$$\mathbf{a} = \begin{bmatrix} \mathbf{a}_0^\top & \mathbf{a}_1^\top & \cdots & \mathbf{a}_{M-2}^\top \end{bmatrix}^\top \in \mathbb{R}^{4(M-1)\times 1} \tag{3.15}$$

$$\boldsymbol{\delta} = \begin{bmatrix} \delta_0 & \delta_1 & \cdots & \delta_{N-1} \end{bmatrix}^\top \in \mathbb{R}^{N\times 1} \tag{3.16}$$

$\mathbf{X} \in \mathbb{R}^{N\times 4(M-1)}$ allows us to evaluate (3.4) for the dataset via $\mathbf{X}\,\mathbf{a}$, and $\mathbf{G} \in \mathbb{R}^{3(M-2)+2\times 4(M-1)}$ allows the constraints to be expressed via (3.14). The solution is found with vector calculus and the method of Lagrange multipliers [22]:

$$\widehat{\mathbf{a}} = \left(\mathbf{X}^\top\mathbf{X}\right)^{-1} (\mathbf{I} - \mathbf{H})\,\mathbf{X}^\top\boldsymbol{\delta} \tag{3.17}$$

where

$$\mathbf{H} = \mathbf{G}^\top \left[\mathbf{G}\left(\mathbf{X}^\top\mathbf{X}\right)^{-1}\mathbf{G}^\top\right]^{-1} \mathbf{G}\left(\mathbf{X}^\top\mathbf{X}\right)^{-1} \tag{3.18}$$

We require at least 4 unique data points in the domain of each polynomial section. This ensures that $\mathbf{X}$ is full column rank so that $\mathbf{X}^\top\mathbf{X}$ is invertible.

3.4.2 Wrapped regression spline

We now assume that the data is wrapped Gaussian-distributed and express the optimization as a weighted least squares problem:

$$\min_{\mathbf{a}_{(0)},\ldots,\mathbf{a}_{(M-2)}} \quad \sum_{i=0}^{N-1} \sum_{l=-\infty}^{\infty} w_{il} \left(\delta_i - \left[y\left(f_i\,;\,\mathbf{a}_{(i)}\right) + 2\pi l\right]\right)^2 \tag{3.19}$$

$$s.t. \quad (3.6) - (3.10) \tag{3.20}$$

where we choose the weights to be:

$$w_{il} = \frac{\mathcal{N}\left(\delta_i\,;\,y\left(f_i\,;\,\mathbf{a}_{(i)}\right) + 2\pi l, \sigma^2\right)}{\sum\limits_{n=-\infty}^{\infty} \mathcal{N}\left(\delta_i\,;\,y\left(f_i\,;\,\mathbf{a}_{(i)}\right) + 2\pi n, \sigma^2\right)} \tag{3.21}$$

We write this more compactly as:

$$\min_{\mathbf{a}} \quad \sum_{l=-\infty}^{\infty} \left(\boldsymbol{\delta} - \left(\mathbf{X}\,\mathbf{a} - 2\pi l \mathbf{1}\right)\right)^\top \mathbf{W}_l \left(\boldsymbol{\delta} - \left(\mathbf{X}\,\mathbf{a} - 2\pi l \mathbf{1}\right)\right) \tag{3.22}$$

$$s.t. \;\; \mathbf{G}\,\mathbf{a} = \mathbf{0} \tag{3.23}$$

where $\mathbf{W}_l = diag\left(\mathbf{w}_l\right)$ contains the weights and $\sum\limits_{l=-\infty}^{\infty} \mathbf{W}_l = \mathbf{I}$.

For fixed $\mathbf{W}$, the solution is given as:

$$\widehat{\mathbf{a}} = \left(\mathbf{X}^\top\mathbf{X}\right)^{-1} \left(\mathbf{I} - \mathbf{H}\right) \mathbf{X}^\top \left(\boldsymbol{\delta} - 2\pi \sum_{l=-\infty}^{\infty} \mathbf{w}_l\, l\right) \tag{3.24}$$

We typically truncate the infinite summation to 5 terms centered at $l = 0$. This incurs very little error.

The weights $\mathbf{w}$ and parameters $\mathbf{a}$ are coupled, so we must iterate between them until convergence. This procedure is actually an EM algorithm. We can see this by recognizing (3.19) as the negative of the Q function for this problem where the weights are posterior probabilities. In the E step, we calculate the posteriors via (3.21) and in the M step, we update the parameters via (3.24). This will converge to a feasible stationary point of the likelihood function associated with this problem.

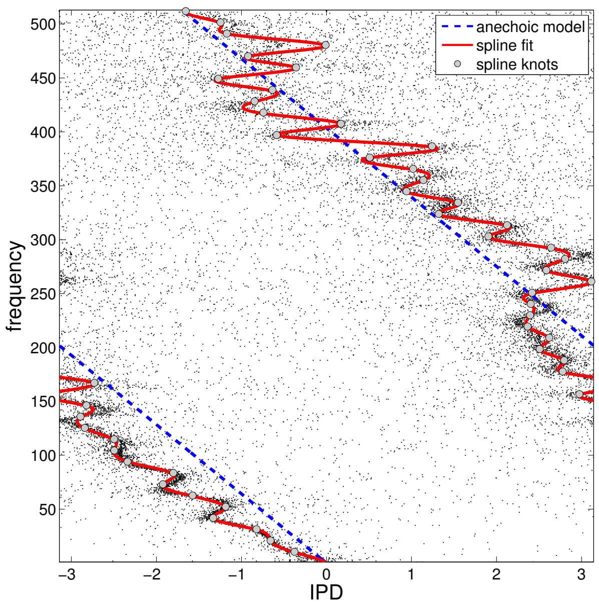

Figure 3.4: Phase difference scatterplot showing nonlinearities due to reverberation and microphone mismatch. The anechoic model and a 50-knot, wrapped cubic spline fit are overlaid.

An attractive feature of this model is its generality. If we constrain the 2^{nd}- and 3^{rd}-order spline parameters to be zero and further constrain all of the linear parameters to be equal, the WCRS reduces to a wrapped line. An unattractive aspect is the computational complexity. Although the large matrix inversion $\left(\mathbf{X}^\top\mathbf{X}\right)^{-1}$ can be broken up into $M-1$ small inversions of size 4×4, these must be computed at every iteration. Another issue is the generalization to multiple sources. This is mathematically straightforward because the only difference is that the posterior probabilities are evaluated over all wrapping indices l *and* source indices k (the spline parameters are updated on a source-specific basis). However, in practice, it is difficult to fit the splines so that they properly handle cross-overs between the individual sources' IPD functions (see Figure 3.8). This suggests that an additional cue is required to distinguish between features that belong to each source.

Figure 3.4 shows an example of an IPD dataset perturbed by noticeable nonlinearities and the spline fit. This data is from a simulation in a reverberant room with randomized IIR filtering at either microphone. We see that the flexibility to adapt to bends in the IPD function allows the spline to correctly model the data. This is especially important at low frequencies where the majority of important speech information lies.

3.5 Other IPD Clustering Methods

The authors in [23] proposed a RANSAC-based solution similar to the one discussed here. However, it involves constructing IPD histograms in each frequency band after replicating the feature values over all physically realizable multiples of 2π. This becomes exponentially computationally expensive as the number of channels increases. RANSHAC avoids this by using the raw IPD values.

Model-Based EM Source Separation and Localization (MESSL) [24] uses an IPD-ILD clustering approach to separate speech mixtures. Gaussian distributions are assumed for both features and an EM algorithm is derived that initially attempts to fit wrapped-line IPD functions and slowly relaxes this to capture general trends. Although this is an interesting approach, it may be difficult to implement in practice for compact arrays in real-world noisy conditions. In this case, ILD features tend to either be uninformative or actively disturb the clustering process.

The beamforming literature [1, 2, 3, 4] consists of an entirely different class of approaches that use phase cues. Beamformers are often used for localization, tracking, and denoising of moving sources, but they can also be applied in general source separation. In the next chapter, we will discuss these methods further and incorporate them into several matrix and tensor factorization algorithms.

3.6 Experiments

To illustrate the differences among the approaches discussed in this chapter, we ran a number of experiments. Random 3-second speech signals from the TSP corpus [25] were mixed in a 5×5 meter room simulator with a 3-channel, right-angle array positioned in the middle of the room with two sources positioned 1 meter away from it on opposite sides. To simulate early reverberation, the source-array configuration was shrunk in size by a factor of 2 and positioned 1 meter from the corner of the room. In the RANSHAC algorithm, the expected fraction of outliers was set to 0.1 and the inlier threshold was set to $\pi/8$. The ML-MoWG slope parameters were initialized with those of RANSHAC and the variance parameters were bounded after

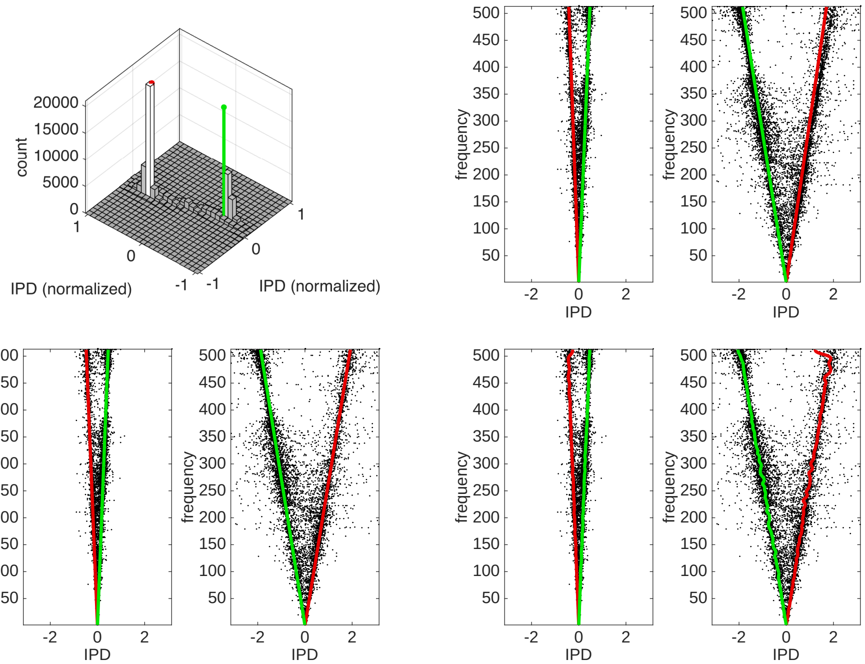

Figure 3.5: IPD modeling results for a simulated mixture of two speakers captured with a three-channel array with no sources of nonlinearity. In each frequency band, only the 50 IPD features with largest corresponding STFT magnitude are shown. (Top left) DUET histogram and estimated source means. (Top right) IPD data and RANSHAC fits. (Bottom left) IPD data and ML-MoWG fits (initialized with RANSHAC). (Bottom right) IPD data and spline fits (initialized with RANSHAC).

each iteration within $[0.1, 1]$. In the spline model, we used a wrapped Gaussian truncation order of 4, 100 spline knots, and an assumed data variance of 0.05.

The model-fitting results are shown in Figures 3.5-3.8 for various types of nonlinearity in the IPD feature set. We observe that as the IPD trends deviate from a linear model, the very flexible spline becomes more appropriate. However, it can be difficult to control the spline precisely because of its flexibility. This is evident from the spurious bend in the spline observable in Figure 3.8. In a noisy data set, the wrapped-line models may perform better because they are more constrained.

The corresponding source separation results are given terms of SIR in Figure 3.9. To perform the separation, binary masks were constructed with a nearest neighbor rule. Each TF bin is assigned to the source whose model value is closest to the corresponding feature value. These quantitative results

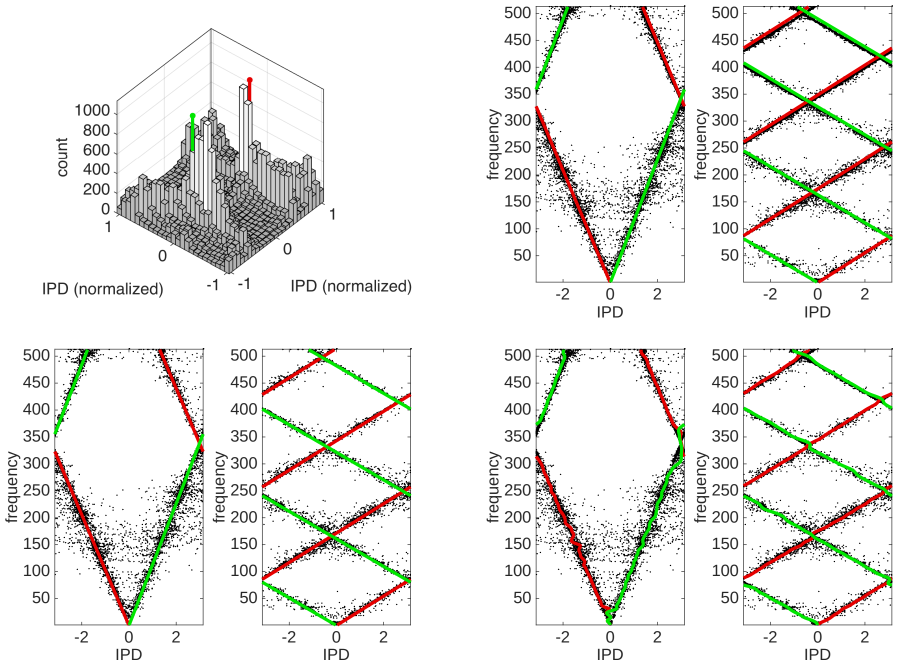

Figure 3.6: IPD modeling results as in Figure 3.5 but with spatial aliasing.

mirror the qualitative results. Although the RANSHAC and ML-MoWG methods show nearly identical results, the latter has the distinct advantage of a principled probabilistic model that can be adapted to other situations (e.g. moving sources [17]).

Finally, we compare their computation time. Given our particular algorithm parameter settings, the average run times for the four approaches were 0.1503, 0.7931, 18.8176, and 29.5018 seconds. We can easily see that increased modeling power comes with longer computation times. However, the source separation performance is potentially much greater.

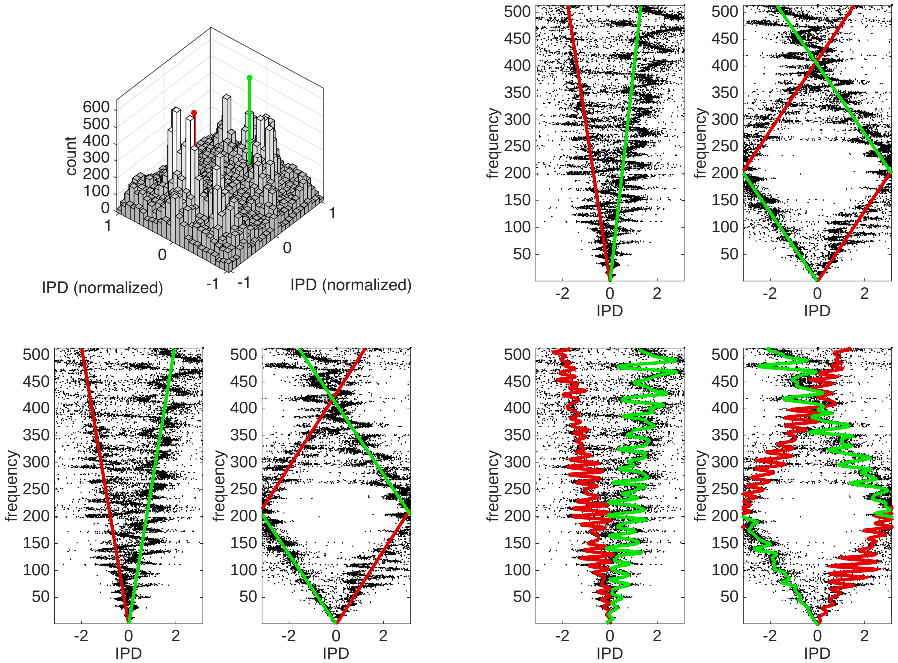

Figure 3.7: IPD modeling results as in Figure 3.5 but with early reverberation.

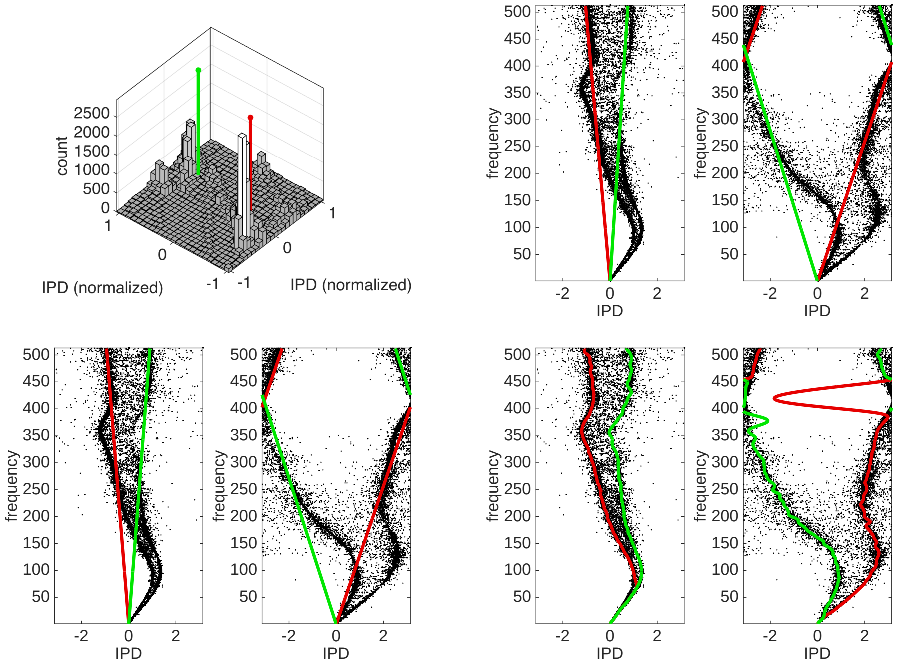

Figure 3.8: IPD modeling results as in Figure 3.5 but with channel mismatch. The flexible spline may have difficulty disambiguating at cross-overs.

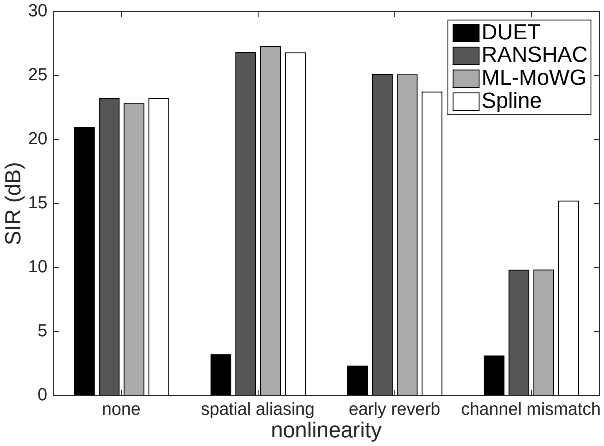

Figure 3.9: Source separation results corresponding to Figures 3.5-3.8.

CHAPTER 4

MATRIX AND TENSOR FACTORIZATION MODELS

All of the previous methods focused on modeling the IPD features exclusively. In a more general approach, we would like to be able to incorporate spectrotemporal information.

A multichannel NMF [26] formulation extends a single-channel model that assumes i.i.d. complex Gaussian STFT coefficients with variances that factor in a two-term NMF form. This is shown to be an exponential-family distribution and an appropriate EM algorithm is derived.

The CMF [27] model was proposed to extend single-channel NMF to incorporate complex values and escape the assumption of disjointness. This model assumes that an STFT matrix factorises into a sum of products of magnitude and exponentiated phase terms. One significant drawback is that each term in the factorization contains its own F-by-T matrix of phase information. This results in a drastic over-parameterization. There is also the additional complication of not knowing how the basis vectors are grouped by source index. These issues were fixed in [28] by assuming one phase matrix per source rather than per basis element. An extension of CMF was proposed to handle the multichannel case [29]. However, this has the same drawbacks as the original single-channel CMF.

Multichannel extensions [30] were proposed that factorize a block matrix of rank-one outer products of complex TF vectors into TF activations and positive semidefinite frequency-dependent matrices that characterize the spatial information in the mixture (gains and delays between sources and microphones). This involves assuming a zero-mean complex Gaussian distribution for each TF bin whose covariance matrix is assumed to factorize. The authors in [31] also used a zero-mean Gaussian model for each TF bin in a somewhat different approach, finding that a full-rank covariance performed best.

The authors in [32] convert the complex matrix factorization problem into a real-valued one by appropriately placing real and imaginary components in a block-wise matrix to be factored into a pair of block-wise matrices.

There are clearly many matrix and tensor factorization approaches to audio source separation. In this chapter, we will focus on the extension of IPD feature and beamformer localization cue modeling to the factorization framework. This will incorporate both spatial and spectrotemporal cues into the separation process.

4.1 Localization Cues

Classical array processing techniques use spatial information to distinguish between sources near the array. The standard approach is to assume an additive Gaussian model for the observed DFT coefficients at each TF bin:

$$\mathbf{x}_{ft} = \mathbf{A}_f\left(\mathbf{\Phi}\right)\ \mathbf{s}_{ft} + \mathbf{n}_{ft} \quad , \quad \mathbf{n}_{ft} \sim \mathcal{N}\left(\mathbf{0}, \sigma_f^2\,\mathbf{I}\right) \tag{4.1}$$

where $\mathbf{s}_{ft} \in \mathbb{C}^K$ is a vector of source DFT coefficients, $\mathbf{n}_{ft} \in \mathbb{C}^M$ is a noise vector, and the steering matrix:

$$\mathbf{A}_f\left(\mathbf{\Phi}\right) = \frac{1}{\sqrt{M}} \exp\left(j\frac{2\pi l_f}{u}\mathbf{m}^\top\mathbf{\Phi}\right) \tag{4.2}$$

relates the source DOAs (in the columns of $\mathbf{\Phi}$) and M microphone locations (in the columns of $\mathbf{m}$) to the array's phase response at frequency band f. The constants l_f and u denote frequency in Hertz at the f^{th} band and the speed of sound, respectively.

When the true DOAs are known, we can apply beamforming to isolate and enhance each source signal. A beamformer is a linear filter $\mathbf{w}$ that can be applied to recover an estimate of a source coefficient via $\widehat{s}_{ft} = \mathbf{w}^H\mathbf{x}_{ft}$. One typically seeks to minimize the expected output power of the beamformer while maintaining certain constraints. In a source separation context, this involves solving the following optimization problem:

$$\min_{\mathbf{w}} \quad \mathbf{w}^H\mathbf{R}\mathbf{w} \tag{4.3}$$

$$\text{s.t.} \;\; \mathbf{A}_f^H\mathbf{w} = \mathbf{u} \tag{4.4}$$

where $\mathbf{R} = \mathbf{E}\left[\mathbf{x}_{ft}\mathbf{x}_{ft}^H\right]$ and $\mathbf{u} \in \mathbb{C}^K$ is a vector of desired gains. Enforcing the constraints makes sure that the energy corresponding to specific DOAs is emphasized or suppressed, while minimizing the objective ensures that as much residual energy as possible is removed. For example, if we wanted to isolate a signal at DOA $\boldsymbol{\phi}_1$ and suppress a signal at DOA $\boldsymbol{\phi}_2$, the constraint would be given as $[\mathbf{a}_f(\boldsymbol{\phi}_1), \mathbf{a}_f(\boldsymbol{\phi}_2)]^H\mathbf{w} = [1, 0]^\top$.

The solution, known as the linearly-constrained minimum-variance (LCMV) beamformer [33], is found with the method of Lagrange multipliers:

$$\widehat{\mathbf{w}} = \mathbf{R}^{-1}\mathbf{A}_f\left(\mathbf{A}_f^H\mathbf{R}^{-1}\mathbf{A}_f\right)^{-1}\mathbf{u} \tag{4.5}$$

and is often simplified to the data-independent form:

$$\widehat{\mathbf{w}} = \mathbf{A}_f\left(\mathbf{A}_f^H\mathbf{A}_f\right)^{-1}\mathbf{u} \tag{4.6}$$

When only one directional source is present, this reduces to the well-known delay-and-sum (D&S) beamformer:

$$\widehat{\mathbf{w}} = \mathbf{a}_f \tag{4.7}$$

A typical beamforming approach to locating the sources, called steered response power (SRP) [34] localization, scans each feasible DOA with a beamformer (typically D&S) and computes the output power of the filtered signal:

$$P_f(\boldsymbol{\theta}) = \sum_t \left|\mathbf{a}_f^H(\boldsymbol{\theta})\,\mathbf{x}_{ft}\right|^2 \tag{4.8}$$

Directions exhibiting salient peaks indicate the presence of a directional source. The peaks in this SRP function can be sharpened by applying the phase transform (PHAT) [35], which simply sets all of the STFT coefficients' magnitudes to 1.

4.2 Directional NMF

Directional NMF (DNMF) [36] involves factorizing a matrix of steered response power (SRP) features into terms that describe the spatial and spectrotemporal properties of the source signals. Rather than accumulate SRP values across frames as in (4.8), we evaluate this function for a discrete set of DOAs at each TF bin and interpret it as a feature vector. This model assumes TF disjointness, which typically holds for speech mixtures, but can handle moderate overlap fairly well.

The single-source version of (4.1) corresponds to the Gaussian likelihood:

$$\mathcal{L}_{ft}\left(\boldsymbol{\theta}\right) = \mathcal{N}\left(\mathbf{x}_{ft}\,;\,\boldsymbol{\mu}_{ft}\,,\,\sigma_f^2\mathbf{I}\right) \tag{4.9}$$

where:

$$\boldsymbol{\mu}_{ft} = \mathrm{E}\left[\mathbf{x}_{ft}\right] = \mathbf{a}_f\left(\boldsymbol{\theta}\right)\mathrm{E}\left[s_{ft}\right] \tag{4.10}$$

Since the source coefficients are unavailable (we are trying to recover them), we replace the expectation in (4.10) with the least-squares estimate and write:

$$\widehat{\boldsymbol{\mu}}_{ft} = \mathbf{a}_f\left(\boldsymbol{\theta}\right)\widehat{s}_{ft} = \mathbf{a}_f\left(\boldsymbol{\theta}\right)\mathbf{a}_f^H\left(\boldsymbol{\theta}\right)\,\mathbf{x}_{ft} \tag{4.11}$$

Substituting (4.11) into (4.9) and expanding, we can write:

$$\log\mathcal{L}_{ft}\left(\boldsymbol{\theta}\right) \propto -\frac{1}{2\sigma_f^2}\,\left(\|\mathbf{x}_{ft}\|_2^2 - |\mathbf{a}_f^H\left(\boldsymbol{\theta}\right)\,\mathbf{x}_{ft}|^2\right) \tag{4.12}$$

This log likelihood is simply an affine transformation of the output power of a delay-and-sum (D&S) beamformer. The variances σ_f^2 can be adjusted to minimize the mismatch in the shape of these functions across frequency, effectively implementing a broadband beamformer. We can concatenate the "likelihood" feature vectors evaluated over D look directions in a nonnegative matrix $\mathbf{L} \in \mathbb{R}^{D\times FT}$ and assume the factorization:

$$\begin{aligned} & \mathbf{L} = \mathbf{DGV} && (4.13) \\ \text{s.t.} \quad & \mathbf{D}, \mathbf{G}, \mathbf{V} \geq 0 && (4.14) \\ & \mathbf{1}_D^\top \mathbf{D} = \mathbf{1}_K^\top \ , \ \mathbf{V} \mathbf{1}_{FT} = \mathbf{1}_K && (4.15) \\ & \mathbf{D} \in \mathbf{R}^{D \times K}, \mathbf{G} \in \mathbf{R}^{K \times K}, \mathbf{V} \in \mathbf{R}^{K \times FT} && (4.16) \end{aligned}$$

where **D** contains SRP basis vectors in the columns, **G** contains mixing weights on the diagonal, and **V** contains TF mask values in the rows. Recall that K indicates the number of sources.

We minimize the Kullback-Liebler divergence $\mathrm{KL}\left(\mathbf{L} \| \mathbf{DGV}\right)$ via multiplicative updates like those proposed in [11] to iteratively solve for the factors:

$$\mathbf{D} \leftarrow \mathbf{D} \odot \frac{\left(\mathbf{L} \oslash \widehat{\mathbf{L}}\right) \mathbf{V}^\top \mathbf{G}^\top}{\mathbf{J} \mathbf{V}^\top \mathbf{G}^\top} \tag{4.17}$$

$$\mathbf{G} \leftarrow \mathbf{G} \odot \frac{\mathbf{D}^\top \left(\mathbf{L} \oslash \widehat{\mathbf{L}}\right) \mathbf{H}^\top}{\mathbf{D}^\top \mathbf{J} \mathbf{V}^\top} \tag{4.18}$$

$$\mathbf{V} \leftarrow \mathbf{V} \odot \frac{\mathbf{G}^\top \mathbf{D}^\top \left(\mathbf{L} \oslash \widehat{\mathbf{L}}\right)}{\mathbf{G}^\top \mathbf{D}^\top \mathbf{J}} \tag{4.19}$$

where $\odot$ and $\oslash$ denote element-wise multiplication and division, **J** is a $D \times FT$ matrix of ones, and $\widehat{\mathbf{L}} = \mathbf{DGV}$ is a reconstruction of the SRP matrix. To avoid scale ambiguities, we normalize the columns of **D** and the rows of **V**:

$$\mathbf{G} \leftarrow \mathrm{diag}\left(\mathbf{D}^\top \mathbf{1}_D\right) \mathbf{G} \, \mathrm{diag}\left(\mathbf{V} \, \mathbf{1}_{FT}\right) \tag{4.20}$$

$$\mathbf{D} \leftarrow \mathbf{D} \, \mathrm{diag}\left(\mathbf{D}^\top \mathbf{1}_D\right)^{-1} \tag{4.21}$$

$$\mathbf{V} \leftarrow \mathrm{diag}\left(\mathbf{V} \, \mathbf{1}_{FT}\right)^{-1} \mathbf{V} \tag{4.22}$$

A derivation of these updates is given in Appendix A. We can interpret the columns of **D** as distributions over DOAs $p\left(\boldsymbol{\theta}_k\right)$ and the rows of **V** as time-frequency distributions $p\left(f, t | k\right)$. Figure 4.1 shows two SRP distributions found by NMF for a mixture of two speakers.

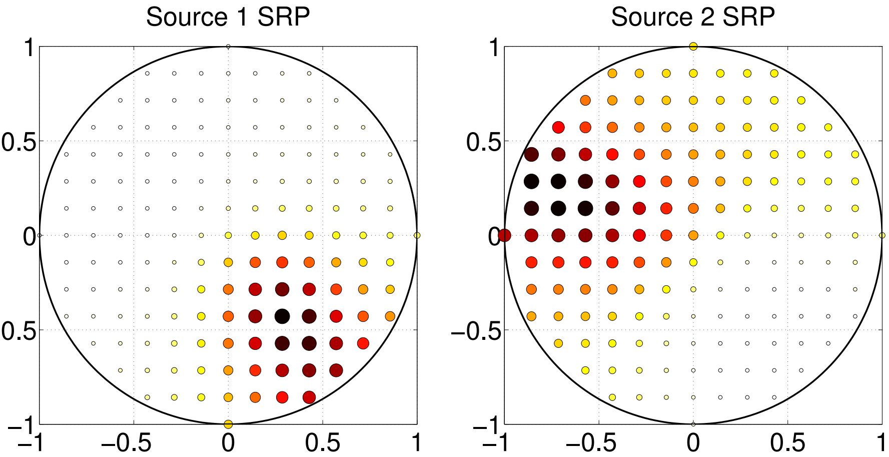

Figure 4.1: SRP distributions (from $\mathbf{W}_{:j}$) for $K = 2$ two sources located on the DOA hemisphere. The hemisphere is flattened such that (azimuth, zenith) points map to (argument, modulus) points. Larger/darker circles denote areas of higher probability mass. The grid has 147 points.

4.3 Nonnegative Tensor Factorization

Directional NMF is generalized by arranging the SRP feature vectors in an $F \times T \times D$ tensor and assuming the following factorization:

$$\mathbf{L} = \sum_{k=1}^{K} (\mathbf{W}_k \mathbf{H}_k) \otimes \mathbf{d}_k \tag{4.23}$$

where $\otimes$ is a tensor outer product, $\mathbf{W} = [\mathbf{W}_1, \ldots, \mathbf{W}_K] \in \mathbb{R}^{F \times Z}$ is a spectral dictionary, $\mathbf{H} = \left[\mathbf{H}_1^\top, \ldots, \mathbf{H}_K^\top\right]^\top \in \mathbb{R}^{Z \times T}$ is a temporal activation matrix, and $\mathbf{d}_k \in \mathbb{R}^D$ is the SRP basis vector for the k^{th} source. This factorization incorporates the often-applied assumption that the mask parameters (i.e. $\mathbf{V}$ in DNMF) are well modeled with a low-rank, two-term factorization. Like DNMF, NTF assumes TF disjointness, but can handle moderate overlap fairly well.

The multiplicative updates can be written as:

$$
\begin{aligned}
\mathbf{W}_k &\leftarrow \mathbf{W}_k \odot \frac{\left\langle \mathbf{L} \oslash \widehat{\mathbf{L}}\,, \mathbf{d}_k \right\rangle \mathbf{H}_k^\top}{\mathbf{1}_{F\times T}\mathbf{H}_k^\top} && (4.24)\\
\mathbf{H}_k &\leftarrow \mathbf{H}_k \odot \left(\mathbf{W}_k^\top \left\langle \mathbf{L} \oslash \widehat{\mathbf{L}}\,, \mathbf{d}_k \right\rangle \right) && (4.25)\\
\mathbf{d}_k &\leftarrow \mathbf{d}_k \odot \left\langle \mathbf{L} \oslash \widehat{\mathbf{L}}\,, \mathbf{W}_k\mathbf{H}_k \right\rangle && (4.26)
\end{aligned}
$$

where $\langle \mathbf{X}\,, \mathbf{y} \rangle_{ij} = \sum\limits_k X_{ijk} y_k$ denotes a tensor inner product. We enforce normalization constraints via:

$$
\begin{aligned}
\mathbf{W} &\leftarrow \mathbf{W}\,\mathrm{diag}\left(\mathbf{W}^\top \mathbf{1}_F\right)^{-1} && (4.27)\\
\mathbf{H}_k &\leftarrow \frac{1}{\mathbf{1}_z^\top \mathbf{H}_k \mathbf{1}_T}\mathbf{H}_k && (4.28)\\
\mathbf{D} &\leftarrow \frac{1}{\mathbf{1}_D^\top \mathbf{D} \mathbf{1}_K}\mathbf{D} && (4.29)
\end{aligned}
$$

A derivation of these updates is given in Appendix B. NTF was shown to significantly outperform DNMF in source separation experiments [12].

4.3.1 Explicit factorial formulation

We can also consider the more computationally burdensome generalization of DNMF where K $F \times T$ SRP matrices are evaluated for every unique source direction K-tuple using a data-independent LCMV beamformer. This beamformer is characterized by a weight matrix whose columns are steering vectors corresponding to each source direction. Thus, the magnitude squared of each LCMV output coefficient gives the SRP values used to construct the tensors. However, it turns out that if we write out the math for the corresponding NTF problem, a closed-form expression results for the direction distributions.

The factorization for each LCMV output is:

$$
\mathbf{L}_k = (\mathbf{W}_k \mathbf{H}_k) \otimes \left(\bigotimes_{k=1}^{K} \mathbf{d}_k \right) \tag{4.30}
$$

where $\bigotimes$ represents a vector Kronecker product. The optimization procedure attempts to factorize all K $F \times T \times O(D^K)$ tensors. Thus, source separation in this case can be seen as a complicated version of a two-step procedure in which an LCMV beamformer is swept through all DOA K-tuples and masks are constructed from the LCMV output with the largest total power. In other words, this is fundamentally no different from a standard localize-then-separate approach and has an exponential run-time as a function of K. The only notable difference is that all DOA K-tuples are considered by weighting all LCMV output matrices by the DOA parameters $\mathbf{d}_k$ before updating the NMF parameters $\mathbf{W}_k, \mathbf{H}_k$. Details of the multiplicative updates are given in Appendix C.

4.4 Complex Tensor Factorization

Complex tensor factorizations have been shown to be promising for blind audio source separation [37].

Although NTF is a fairly powerful model with many opportunities for regularization and generalization, it assumes a particular array configuration, no channel mismatch, no reverberation, and no spatial aliasing. We can reformulate our description of the data in a way that leverages spectrotemporal factorization and raw IPD feature modeling simultaneously. We do this by arranging the IPD features in a tensor as in NTF:

$$L_{ftd} = \left|X^*_{ft}\right| e^{j\angle\left(X^{I_1(d)}_{ft} \big/ X^{I_2(d)}_{ft}\right)} \tag{4.31}$$

where $I_1(d), I_2(d)$ are indexing operators to select distinct channel pairs, $d \in [1, D]$ denotes a pair index, and the asterisk in X^*_{ft} indicates that any channel can be chosen. We then assume the factorization:

$$\mathbf{L} = \sum_{k=1}^{K} \left(\mathbf{W}^k \mathbf{H}^k\right) \otimes \mathbf{M}^k \tag{4.32}$$

$$\text{s.t.} \quad \forall\, k \;\; \mathbf{W}^k, \mathbf{H}^k \geq 0 \tag{4.33}$$

$$\forall\, k \;\; \mathbf{1}_F^\top \mathbf{W}^k = \mathbf{1}_z^\top \tag{4.34}$$

$$\forall\, f, d, k \;\; \left|M_{fd}^k\right| = 1 \tag{4.35}$$

$$\forall\, k \;\; \mathbf{W}^k \in \mathbb{R}^{F \times z}, \mathbf{H}^k \in \mathbb{R}^{z \times T}, \mathbf{M}_k \in \mathbb{C}^{F \times d} \tag{4.36}$$

In this model, which we will refer to as CTF-IPD, the matrix $\mathbf{M}^k$ contains complex-valued mean functions that can represent any nonlinear pattern in the IPD features. In this sense, it generalizes all of the other models discussed so far.

Assuming a complex Gaussian error, we solve iteratively for the parameters via projected gradient descent on the error function:

$$e = \sum_{f,t,d} \left\| L_{ftd} - \sum_k \left(\sum_z W_{fz}^k H_{zt}^k \right) M_{fd}^k \right\|_2^2 \tag{4.37}$$

This model is highly expressive and therefore must be constrained appropriately. For example, we may want to impose smoothness in the mean functions across frequency and enforce that the dictionaries learn speech-like spectra. One interesting approach is that taken in the MESSL algorithm [24]. The IPD functions are constrained to be circular-linear at first and are allowed to be increasingly unconstrained as the learning progresses. The optimization details are given in Appendix D.

4.5 CTF of Raw STFT Matrices

One drawback that limits the expressivity of these models is the assumption that only one source is strongly activated in each TF bin. Although this is approximately the case for speech signals, it is clearly suboptimal. A better model, CTF-Raw, should represent overlap between the sources in the TF plane and therefore additivity of the STFT coefficients. A straightforward adaptation of the CTF-IPD model accomplishes this:

$$\mathbf{L} = \sum_{k=1}^{K} \left[\left(\mathbf{W}^k \mathbf{H}^k \right) \odot \mathbf{P}^k \right] \otimes \mathbf{M}^k \tag{4.38}$$

$$\text{s.t.} \quad \forall\, k \;\; \mathbf{W}^k, \mathbf{H}^k \geq 0 \tag{4.39}$$

$$\forall\, k \;\; \mathbf{1}_F^\top \mathbf{W}^k = \mathbf{1}_z^\top \tag{4.40}$$

$$\forall\, f, d, k \;\; \left| M_{fd}^k \right| = 1 \tag{4.41}$$

$$\forall\, f, t, k \;\; \left| P_{ft}^k \right| = 1 \tag{4.42}$$

$$\forall\, k \;\; \mathbf{W}^k \in \mathbb{R}^{F \times z}, \mathbf{H}^k \in \mathbb{R}^{z \times T}, \mathbf{P}^k \in \mathbb{C}^{F \times T}, \mathbf{M}^k \in \mathbb{C}^{F \times d} \tag{4.43}$$

where $\mathbf{P}^k$ is a matrix of unit complex numbers that represents the estimated phase of the k^{th} source's STFT. Now, the $\mathbf{M}^k$ parameter represents the frequency-dependent phase response for the k^{th} source. The optimization procedure is analogous to that of CTF-IPD. However, this model does not assume TF disjointness. It only assumes that the spectrogram of each source is accurately represented with a low-rank two-term factorization.

4.6 Experiments

In this section, we will compare all five methods discussed in this chapter, three of which involve nonnegative factorizations (DNMF, NTF, Factorial NTF) and two of which involve complex factorizations (CTF-IPD, CTF-Raw). For reference, we will also include a single-channel supervised NMF algorithm and a standard classical array processing approach that first applies SRP-PHAT for localization and then LCMV beamforming for separation.

The NMF algorithm involves first learning speaker-specific dictionaries for each speaker and then concatenating them to form a dictionary for the mixture. The mixture spectrogram is used to learn the activation matrix at test time and source-specific reconstructions are used to perform the separation via masking. The SRP-PHAT + LCMV approach implements a standard SRP localization scheme on a grid over the DOA space that sequentially identifies peaks. These peak locations are used to implement LCMV beamformers to isolate the sources. These are very standard procedures in the NMF and beamforming literatures.

The experimental setup was as follows. In each of 20 trials, 2- to 3-second sentences for $K = 2$ speakers were selected uniformly at random from the TSP corpus [25]. These were emitted from randomly chosen locations in a ring centered at a 4-channel, square microphone array placed in a 2-dimensional room simulator of size 5×5 meters. The speaker locations were chosen to ensure that they were separated by at least $2\pi/(K+1)$ radians relative to the array.[1] Six different scenarios were used to test the algorithms and the scenario-specific settings are given in Table 4.1. All NMF and NTF algorithms were run for 50 iterations. When nonlinearities are expected, the CTF-IPD model is fit with a linear phase difference function constraint for 50 iterations and then allowed to fit unconstrained for another 50 iterations. 50 basis vectors were used in all spectral source dictionaries. The DOAs learned in NTF where used to initialize the CTF models.

All matrix factorization-based algorithms used masking to reconstruct the separated sources. This involves the standard procedure of estimating the magnitude portion of the source spectrograms with the learned parameters and forming soft TF masks [11] to be applied element-wise to the first channel's mixture STFT. The SRP-PHAT + LCMV method automatically produces estimated STFTs, one for each separate source.

Separation performance results are shown in terms of signal-to-interference Ratio (SIR) in Figure 4.2. What we see is that, in simpler cases (e.g. ideal set-up), methods based on beamforming, directional NMF, and NTF perform better than other methods (sometimes including supervised NMF). However, as the experimental circumstances become more difficult to handle, a more expressive nonlinear model like CTF-IPD performs best. CTF-Raw has a surprisingly poor performance in almost all cases. However, superior performance is observed in specific trials. The average performance suffers when a poor initialization is used and the optimization gets stuck in a poor local optimum. A complicating factor in the CTF-Raw model is the large number of parameters. The performance of the IPD-based methods depends strongly on the optimization procedure used.[2] Thus, improving it beyond the adaptive gradient descent scheme used here may lead to better results. As expected, the more expressive CTF-IPD algorithm outperforms the other unsupervised

[1] Ensuring robustness to small angles of separation (in terms of DOA) is not a point of focus in this thesis.

[2] Take special note of the mixed constraints in the CTF-IPD model.

Table 4.1: Details of experimental setup. For each scenario, the variable settings are indicated as follows. Reverb: if moderate reverberation was applied. IIR: if IIR filtering was applied to the recorded mixtures (to simulate channel mismatch). CTF iters: the iteration counts used for the CTF-IPD and CTF-raw algorithms during linear and nonlinear learning stages. Mic spacing: length of microphone array square sides. Array center: location of center of microphone array as fraction of room size. Source radius: radius of circle centered at microphone array on which sources are located.

Scenario	reverb	IIR	CTF iters	mic spacing (cm)	array center	source radius (m)
ideal	no	no	75,0	2	1/2	1
alias	no	no	75,0	10	1/2	1
alias + l_rev	yes	no	75,75	10	1/2	1
alias + e_rev	yes	no	75,75	7.5	1/4	0.5
alias + IIR	no	yes	75,75	10	1/2	1
alias + IIR + l_rev	yes	yes	75,75	10	1/2	1

methods when channel mismatch (simulated with IIR filtering) is present.

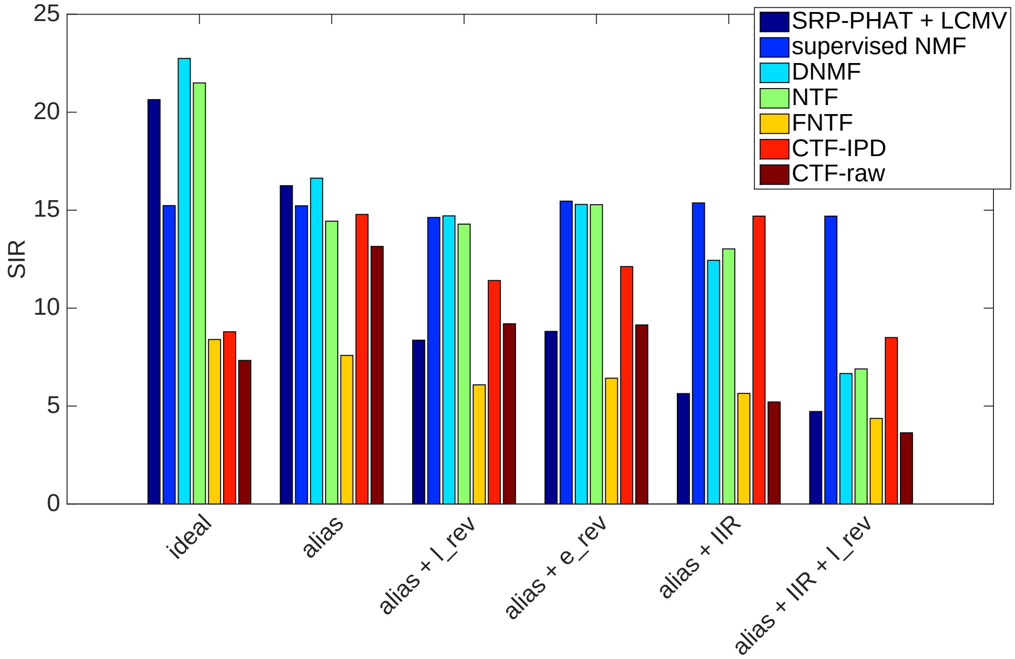

Figure 4.2: Average SIR values for various algorithms in a two-source separation experiment. The specifics of the experimental setups are given in the text and Table 4.1.

CHAPTER 5

CONCLUSION AND DISCUSSION

Phase difference modeling, matrix and tensor factorizations, and TF masking have been shown to be effective in source separation applications. We can apply these tools simultaneously in models that are capable of describing observed spatial mixtures in the presence of noise, reverberation, channel mismatch, etc. One promising direction described in this thesis is the CTF model, which can represent an arbitrary nonlinear phase difference function as well as spectrotemporal characteristics for each source. Extending this model to be robust in challenging real-world scenarios is the next step in this line of research. This may involve adaptations to the model such as regularization and task-specific prior knowledge.

In this thesis, it was assumed that all sources are stationary. However, this is not necessary. The ML-MoWG and RANSHAC approaches have been extended to the case where the sources are moving by tracking the source directions-of-arrival (DOAs) with a factorial wrapped Kalman filter (FWKF). The WKF [38] was proposed separately to treat the problem of tracking on the unit circle. In this context, the IPD features are transformed to DOA space to provide observations for the filters using a RANSAC-like sampling scheme. A directional filter was also developed for tracking on the sphere: von Mises-Fisher Filter (vMFF) [39]. Both make use of deterministic approximations to solve the Bayesian filtering equations more efficiently than particle filtering and with greater accuracy than extended (EKF) and unscented (UKF) Kalman filters. Finally, an explicit multiple-source SRP likelihood has been derived and used to perform simultaneous localization of speech sources [40].

We conclude by considering the relationship between the IPD and factorization models. IPD features and beamformer localization cues (as used in DNMF and NTF) both derive from spatial information through the time-delay-of-arrival (TDOA) of sound waves impinging on the microphone array.

This relationship is clear for the anechoic case in which a single TDOA is active per source. For each source, the IPD line slope is linearly proportional to the TDOA and the SRP feature vectors all share a dominant peak. We can even draw an analogy between an SRP profile and the function resulting from evaluating the "likelihood" of feasible wrapped lines for an IPD dataset [17]. The resulting feature sets differ mathematically, but they both derive from the same time delays.

When nonlinearities (e.g. due to reverberation and channel mismatch) are present, the connection between TDOAs and feature values is significantly more complicated. The non-linear IPD and CTF models both attempt to gracefully handle this complication in a general way. However, we can see how difficult this is by observing that an ideal fit to the data effectively recovers the room impulse responses. And this, in turn, implies de-reverberation and channel equalization. Thus, in practice, a balance must be struck between generalization and modeling precision. A crucial factor in the success of these methods is an excellent match between feature representation and model. In this thesis, we have seen several of these pairings, but there are likely others that perform better in some way. Exploring this possibility is left for future work.

Also left as an open problem for future research is a theoretical analysis of the effects of parameter choices in the STFT on the performance of the algorithms in this thesis as well as an experimental validation of this analysis.

APPENDIX A

DNMF OPTIMIZATION

The optimization problem we are trying to solve is:

$$\begin{aligned}
\min_{\mathbf{D},\mathbf{G},\mathbf{V}} \quad & \mathrm{KL}\left(\mathbf{L} \,\|\, \mathbf{DGV}\right) && \text{(A.1)}\\
\text{s.t.} \quad & \mathbf{D}, \mathbf{G}, \mathbf{V} \geq 0 && \text{(A.2)}\\
& \mathbf{1}_D^\top \mathbf{D} = \mathbf{1}_K^\top \ , \ \mathbf{V}\mathbf{1}_{FT} = \mathbf{1}_K && \text{(A.3)}\\
& \mathbf{D} \in \mathbf{R}^{D\times K}, \mathbf{G} \in \mathbf{R}^{K\times K}, \mathbf{V} \in \mathbf{R}^{K\times FT} && \text{(A.4)}
\end{aligned}$$

where:

$$\begin{aligned}
\mathrm{KL}\left(\mathbf{X} \,\|\, \mathbf{Y}\right) &= \mathrm{tr}\left[\mathbf{X}^\top \log\left(\mathbf{X} \oslash \mathbf{Y}\right)\right] - \mathrm{tr}\left[\mathbf{1}_{D\times FT}^\top \mathbf{X}\right] + \mathrm{tr}\left[\mathbf{1}_{D\times FT}^\top \mathbf{Y}\right] && \text{(A.5)}\\
&= \sum_{i,j} X_{ij} \log\left(\frac{X_{ij}}{Y_{ij}}\right) - X_{ij} + Y_{ij} && \text{(A.6)}
\end{aligned}$$

To derive the multiplicative update for a factor $\mathbf{Q}$, we compute the partial derivative of the objective, which is always of the form:

$$\nabla_{\mathbf{Q}} = \nabla_{\mathbf{Q}}^{+} - \nabla_{\mathbf{Q}}^{-} \qquad \text{(A.7)}$$

with positive and negative parts $\nabla_{\mathbf{Q}}^{+}, \nabla_{\mathbf{Q}}^{-}$ and apply gradient descent:

$$\mathbf{Q} \longleftarrow \mathbf{Q} - \boldsymbol{\eta} \odot \nabla_{\mathbf{Q}} \qquad \text{(A.8)}$$

where the step size is chosen as $\boldsymbol{\eta} = \mathbf{Q} \oslash \nabla_{\mathbf{Q}}^{+}$. Thus, we have:

$$\mathbf{Q} \longleftarrow \mathbf{Q} \odot \frac{\nabla_{\mathbf{Q}}^{-}}{\nabla_{\mathbf{Q}}^{+}} \qquad \text{(A.9)}$$

To enforce the constraints, we normalize the factors appropriately after each iteration. This procedure inherits the local convergence properties of two-term NMF [11].

APPENDIX B

NTF OPTIMIZATION

We could view NTF optimization problem directly in terms of linear algebra. However, an equivalent probabilistic formulation allows for greater generalization. We will derive multiplicative updates via a setup akin to PLSI [41]. The assumed factorization is:

$$p(f,t,d) = \sum_{s,z} p(f|s,z)p(t|s,z)p(z|s)p(d|s)p(s) = \sum_{s,z} p(f|s,z)p(t,z|s)p(d,s) \tag{B.1}$$

where s and z denote source and dictionary element indices. We seek to maximize the negative cross entropy:

$$\sum_{f,t,d} L(f,t,d) \log p(f,t,d) \tag{B.2}$$

Applying the EM framework, we define the auxiliary Q function:

$$Q = \sum_{f,t,d,s,z} L(f,t,d) \log p(f,t,d,s,z) \tag{B.3}$$

Computing partial derivatives with appropriate Lagrange multiplier terms to ensure normalization, we have the following EM update equations:

$$p(s,z|f,t,d) = \frac{p(f|s,z)p(t,z|s)p(d,s)}{p(f,t,d)} \tag{B.4}$$

$$p(f|s,z) = \frac{\sum\limits_{t,d} L(f,t,d)p(s,z|f,t,d)}{\sum\limits_{f,t,d} L(f,t,d)p(s,z|f,t,d)} = \frac{\sum\limits_{t,d} L(f,t,d)p(s,z|f,t,d)}{\sum\limits_{t} p(t,z|s)\sum_d p(d,s)} \quad \text{(B.5)}$$

$$p(t,z|s) = \frac{\sum\limits_{f,d} L(f,t,d)p(s,z|f,t,d)}{\sum\limits_{f,t,d,z} L(f,t,d)p(s,z|f,t,d)} = \frac{\sum\limits_{f,d} L(f,t,d)p(s,z|f,t,d)}{\sum\limits_{d} p(d,s)} \quad \text{(B.6)}$$

$$p(d,s) = \frac{\sum\limits_{f,t,z} L(f,t,d)p(s,z|f,t,d)}{\sum\limits_{f,t,d,s,z} L(f,t,d)p(s,z|f,t,d)} = \sum_{f,t,z} L(f,t,d)p(s,z|f,t,d) \quad \text{(B.7)}$$

Plugging the E step into the M step and simplifying, we have multiplicative updates:

$$p(f|s,z) \longleftarrow p(f|s,z)\frac{\sum\limits_{t} p(t,z|s)\sum\limits_{d} \bar{L}(f,t,d)p(d|s)}{\sum\limits_{t} p(t,z|s)} \quad \text{(B.8)}$$

$$p(t,z|s) \longleftarrow p(t,z|s)\sum_{f} p(f|s,z)\sum_{d} \bar{L}(f,t,d)p(d|s) \quad \text{(B.9)}$$

$$p(d,s) \longleftarrow p(d,s)\sum_{f,t} \bar{L}(f,t,d)\sum_{z} p(f|s,z)p(t,z|s) \quad \text{(B.10)}$$

where $\bar{L}(f,t,d) = L(f,t,d)/p(f,t,d)$ and $p(d|s) = p(d,s)/\sum\limits_{d} p(d,s)$. If we enforce normalization constraints after each iteration, it suffices to replace $p(d|s)$ with $p(d,s)$ in the first two updates.

The striking similarity of these updates to standard NMF updates is explained by the equivalence of PLSI and NMF [42].

APPENDIX C

FACTORIAL NTF OPTIMIZATION

In the factorial variant of NTF, we make use of the same probabilistic formulation as in NTF. The assumed factorization is:

$$p(f, t, \bar{d}, s) = \sum_{z} p(f|s, z)p(t, s, z)p(\bar{d}) \tag{C.1}$$

where $\bar{d}$ is the index into the product distribution $\bar{\mathbf{d}} = \bigotimes_{k=1}^{K} \mathbf{d}_k$ that captures the probability that each DOA K-tuple is the true one.

Following the derivation procedure for NTF, we have multiplicative updates:

$$p(f|s, z) \longleftarrow p(f|s, z)\frac{\sum_t p(t, z, s) \sum_{\bar{d}} \bar{L}(f, t, \bar{d}, s)p(\bar{d})}{\sum_t p(t, s, z)} \tag{C.2}$$

$$p(t, s, z) \longleftarrow p(t, s, z)\sum_{f} p(f|s, z) \sum_{\bar{d}} \bar{L}(f, t, \bar{d}, s)p(\bar{d}) \tag{C.3}$$

$$p(d_k) \longleftarrow \sum_{f,t,\bar{d}_{\neg k},s} L(f, t, \bar{d}_{\neg k}, s) \tag{C.4}$$

where the DOA distribution update is in closed-form for each component k of the product distribution. The notation $\bar{d}_{\neg k}$ denotes all indices in the product distribution that include d_k.

APPENDIX D

CTF OPTIMIZATION

The optimization problem we are trying to solve is:

$$\min_{\mathbf{W},\mathbf{H},\mathbf{M}} \quad \left\| \mathbf{L} - \sum_{k=1}^{K} \left(\mathbf{W}^k \mathbf{H}^k\right) \otimes \mathbf{M}^k \right\|_2^2 \tag{D.1}$$

$$\text{s.t.} \quad \forall\, k \;\; \mathbf{W}^k, \mathbf{H}^k \geq 0 \tag{D.2}$$

$$\forall\, k \;\; \mathbf{1}_F^\top \mathbf{W}^k = \mathbf{1}_z^\top \tag{D.3}$$

$$\forall\, f, d, k \;\; \left| M_{fd}^k \right| = 1 \tag{D.4}$$

$$\forall\, k \;\; \mathbf{W}^k \in \mathbb{R}^{F \times z}, \mathbf{H}^k \in \mathbb{R}^{z \times T}, \mathbf{M}_k \in \mathbb{C}^{F \times d} \tag{D.5}$$

In light of the fact that the constraints are fairly prohibitive, we apply alternating projected gradient descent to optimize the parameters. The objective can be written as:

$$e = \sum_{f,t,d} \left| L_{f,t,d} - \sum_k \Gamma_{ft}^k M_{fd}^k \right|^2 \quad , \quad \Gamma_{ft}^k = \sum_i W_{fi}^k H_{it}^k \tag{D.6}$$

The gradients are:

$$\frac{\partial\, e}{\partial\, M_{f,d}^k} = -2 \sum_t \left(L_{f,t,d} - \sum_{k'} \Gamma_{f,t}^{k'} M_{f,d}^{k'} \right) \Gamma_{f,t}^k \tag{D.7}$$

$$\frac{\partial\, e}{\partial\, W_{f,i}^k} = -2 \sum_{t,d} \left(L_{f,t,d} - \sum_{k'} \Gamma_{f,t}^{k'} M_{f,d}^{k'} \right) {M_{f,d}^k}^* H_{i,t}^k \tag{D.8}$$

$$\frac{\partial\, e}{\partial\, H_{i,t}^k} = -2 \sum_{f,d} \left(L_{f,t,d} - \sum_{k'} \Gamma_{f,t}^{k'} M_{f,d}^{k'} \right) {M_{f,d}^k}^* W_{f,i}^k \tag{D.9}$$

If we constrain the means to have a particular parameterized form, we can use the chain rule to include the contribution of this parameterization to the gradient. Suppose we assume the standard wrapped-linear form characterized by steering vectors:

$$M^k_{f,d} = e^{j\frac{2\pi\omega_f}{v_s}\mathbf{m}_d^\top \boldsymbol{\theta}^k} \tag{D.10}$$

where $\mathbf{m}$ is the matrix of microphone location differences and $\boldsymbol{\theta} \in \mathbb{R}^3$ is a DOA vector. Then, the chain rule gives:

$$\frac{\partial M^k_{f,d}}{\partial \theta^k_i} = M^k_{f,d}\, j\frac{2\pi\omega_f}{v_s} m_{i,d} \tag{D.11}$$

The full gradient for this DOA parameter is:

$$\frac{\partial\, e}{\partial\, \theta^k_i} = -2\sum_{f,t,d}\left(L_{f,t,d} - \sum_{k'}\Gamma^{k'}_{f,t}M^{k'}_{f,d}\right)\Gamma^k_{f,t}\frac{\partial M^k_{f,d}}{\partial \theta^k_i} \tag{D.12}$$

Iterating gradient descent updates using (D.7)-(D.9) and projections to ensure the constraints are satisfied, using an adaptive step size, and ensuring that the objective function decreases at each step lead to convergence to a local solution.

REFERENCES

[1] J. Benesty, J. Chen, and Y. Huang, *Topics in Signal Processing: Microphone Array Signal Processing.* Springer, 2008, vol. 1.

[2] H. K. van Trees, *Detection, Estimation, and Modulation Theory: Optimum Array Processing (Part IV).* Wiley, 2002.

[3] W. Herbordt, *Sound Capture for Human-Machine Interfaces.* Springer, 2005.

[4] I. J. Tashev, *Sound Capture and Processing: Practical Approaches.* Wiley, 2009.

[5] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2003, pp. 177–180.

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

[7] S. Araki, H. Sawada, R. Mukai, and S. Makino, "Underdetermined blind sparse source separation for arbitrarily arranged multiple sensors," *Signal Processing*, vol. 87, no. 8, pp. 1833 – 1847, 2007.

[8] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 65–68, 2014.

[9] J. Traa, "Multichannel source separation and tracking with phase differences by random sample consensus," M.S. thesis, University of Illinois at Urbana-Champaign, 2013.

[10] Y. Li and D. Wang, "On the optimality of ideal binary time-frequency masks," *Elsevier - Speech Communication*, vol. 51, pp. 230–239, 2009.

[11] D. L. Daniel and H. S. Seung, "Algorithms for non-negative matrix factorization," *Conference on Advances in Neural Information Processing Systems*, pp. 556–562, 2001.

[12] N. D. Stein, "Nonnegative tensor factorization for directional blind audio source separation," *arXiv:1411.5010 [stat.ML]*, 2014.

[13] G. J. Mysore, P. Smaragdis, and B. Raj, *Non-negative Hidden Markov Modeling of Audio with Application to Source Separation.* Springer Berlin Heidelberg, 2010, pp. 140–148.

[14] G. J. Mysore and P. Smaragdis, "A non-negative approach to semi-supervised separation of speech from noise with the use of temporal dynamics," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2011, pp. 17–20.

[15] D. W. Griffin and J. S. Lim, "Signal estimation from modified short-time Fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 32, no. 2, pp. 236–243, 1984.

[16] J. Traa and P. Smaragdis, "Blind multi-channel source separation by circular-linear statistical modeling of phase differences," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013.

[17] J. Traa and P. Smaragdis, "Multichannel source separation and tracking with ransac and directional statistics," *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)*, vol. 22, pp. 2233–2243, 2014.

[18] J. Traa and P. Smaragdis, "Robust interchannel phase difference modeling with wrapped regression splines," *IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM)*, 2014.

[19] M. A. Fischler and R. C. Bolles, "Random Sample Consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[20] J. Traa, M. Kim, and P. Smaragdis, "Phase and level difference fusion for robust multichannel source separation," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2014.

[21] K. V. Mardia, "Statistics of directional data (with discussion)," *J. R. Statist. Soc.*, vol. B 37, pp. 349–393, 1975.

[22] J. R. Magnus and H. Neudecker, *Matrix Differential Calculus with Applications in Statistics and Econometrics*, 2nd ed. Wiley, 1999.

[23] L. Litwic and P. Jackson, "Source localization and separation using random sample consensus with phase cues," in *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2011 IEEE Workshop on*, Oct 2011, pp. 337–340.

[24] M. I. Mandel, R. J. Weiss, and D. P. W. Ellis, "Model-based expectation-maximization source separation and localization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 2, pp. 382–394, 2010.

[25] P. Kabal, "TSP speech database," 2002, Telecommunications and Signal Processing Lab, McGill University.

[26] A. Ozerov and C. Fvotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 18, no. 3, pp. 550–563, 2010.

[27] H. Kameoka, N. Ono, K. Kashino, and S. Sagayama, "Complex NMF: A new sparse representation for acoustic signals," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, 2009, pp. 3437–3440.

[28] B. J. King, "New methods of complex matrix factorization for single-channel source separation and analysis," Ph.D. dissertation, University of Washington, 2013.

[29] H. Sawada, H. Kameoka, S. Araki, and N. Ueda, "Formulations and algorithms for multichannel complex NMF," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011, pp. 229–232.

[30] H. Sawada, H. Kameoka, S. Araki, and N. Ueda, "Multichannel extensions of non-negative matrix factorization with complex-valued data," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 21, no. 5, pp. 971–982, 2013.

[31] N. Duong, E. Vincent, and R. Gribonval, "Under-determined reverberant audio source separation using a full-rank spatial covariance model," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 18, no. 7, pp. 1830–1840, 2010.

[32] C. Ahuja, K. Nathwani, and R. M. Hegde, "A complex matrix factorization approach to joint modeling of magnitude and phase for source separation," *CoRR*, vol. abs/1411.6741, 2014. [Online]. Available: http://arxiv.org/abs/1411.6741

[33] L. J. Griffiths and C. W. Jim, “An alternative approach to linearly constrained adaptive beamforming,” *IEEE Transactions on Antennas and Propagation*, vol. 30, no. 1, pp. 27–34, 1982.

[34] S. T. Birchfield and D. K. Gillmor, “Fast Bayesian acoustic localization,” *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, pp. 1793–1796, 2002.

[35] C. H. Knapp and G. C. Carter, “The generalized correlation method for estimation of time delay,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 4, pp. 320–327, 1976.

[36] J. Traa, N. D. Stein, D. Wingate, and P. Smaragdis, “Directional NMF for joint source localization and separation,” *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2015.

[37] A. Ozerov and C. Fvotte, “Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation,” *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 18, no. 3, pp. 550–563, 2010.

[38] J. Traa and P. Smaragdis, “A wrapped Kalman filter for azimuthal speaker tracking,” *IEEE Signal Processing Letters (SPL)*, vol. 20, pp. 1257–1260, 2013.

[39] J. Traa and P. Smaragdis, “Multiple speaker tracking with the factorial von mises-fisher filter,” *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2014.

[40] J. Traa, N. D. Stein, D. Wingate, and P. Smaragdis, “Robust source localization and enhancement with a probabilistic steered response power model,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)*, (in review).

[41] T. Hofmann, “Probabilistic latent semantic indexing,” in *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR ’99, 1999, pp. 50–57.

[42] C. Ding, T. Li, and W. Peng, “On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing,” *Computational Statistics and Data Analysis*, vol. 52, no. 8, pp. 3913–3927, Apr. 2008.