MULTICHANNEL SOURCE SEPARATION AND TRACKING WITH PHASE DIFFERENCES BY RANDOM SAMPLE CONSENSUS

BY

JOHANNES TRAA

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2013

Urbana, Illinois

Adviser:

Assistant Professor Paris Smaragdis

ABSTRACT

Blind audio source separation (BASS) is a fascinating problem that has been tackled from many different angles. The use case of interest in this thesis is that of multiple moving and simultaneously-active speakers in a reverberant room. This is a common situation, for example, in social gatherings. We human beings have the remarkable ability to focus attention on a particular speaker while effectively ignoring the rest. This is referred to as the "cocktail party effect" and has been the holy grail of source separation for many decades. Replicating this feat in real-time with a machine is the goal of BASS.

Single-channel methods attempt to identify the individual speakers from a single recording. However, with the advent of hand-held consumer electronics, techniques based on microphone array processing are becoming increasingly popular. Multichannel methods record a sound field from various locations to incorporate spatial information. If the speakers move over time, we need an algorithm capable of tracking their positions in the room. For compact arrays with 1-10 cm of separation between the microphones, this can be accomplished by applying a temporal filter on estimates of the directions-of-arrival (DOA) of the speakers.

In this thesis, we review recent work on BSS with inter-channel phase difference (IPD) features and provide extensions to the case of moving speakers. It is shown that IPD features compose a noisy circular-linear dataset. This data is clustered with the RANdom SAmple Consensus (RANSAC) algorithm in the presence of strong reverberation to simultaneously localize and separate speakers. The remarkable performance of RANSAC is due to its natural tendency to reject outliers. To handle the case of non-stationary speakers, a factorial wrapped Kalman filter (FWKF) and a factorial von Mises-Fisher particle filter (FvMFPF) are proposed that track source DOAs directly on the unit circle and unit sphere, respectively. These algorithms combine directional statistics, Bayesian filtering theory, and probabilistic data association techniques to track the speakers with mixtures of directional distributions.

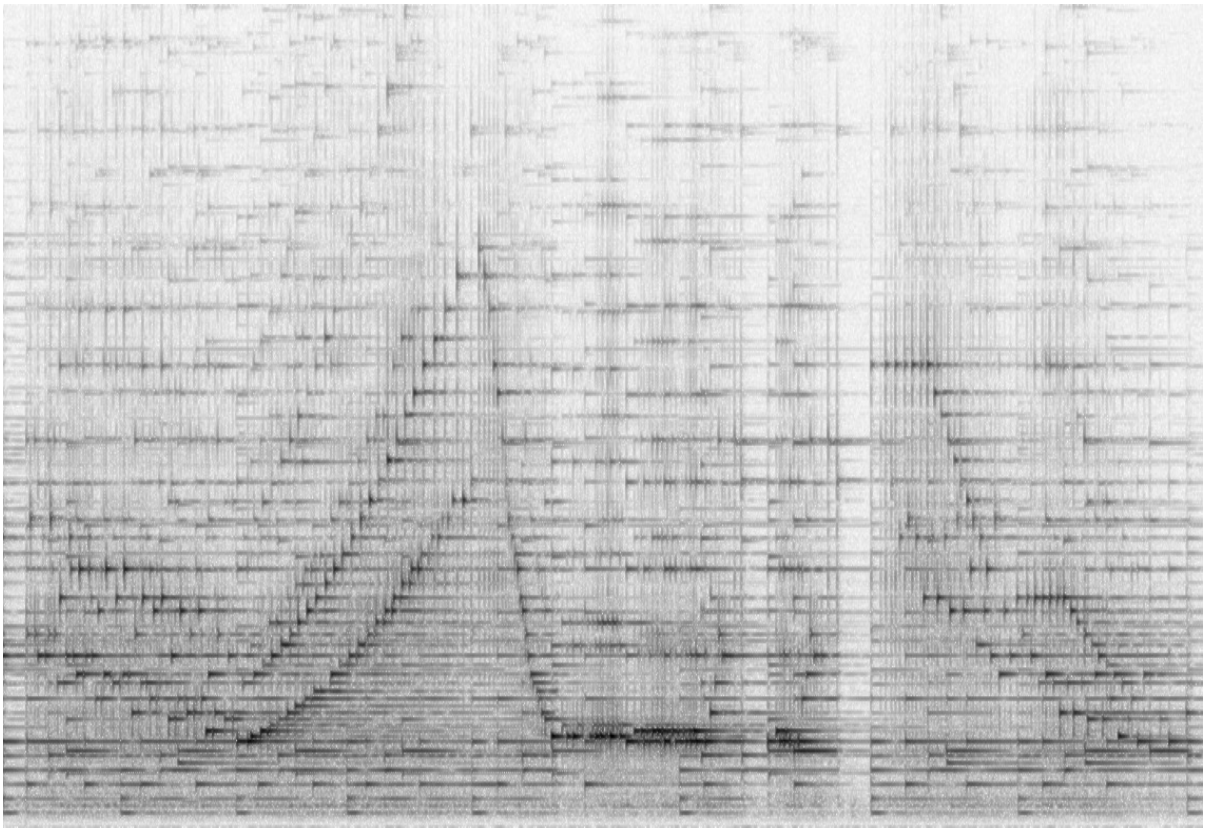
ACKNOWLEDGMENTS

The bubbling froth that is this thesis would have been rendered insipid where it not for the amiable gestures of several parties. First, I would like to thank my family for their support and prodigious tolerance over the many years. My life would most likely have taken a lesser route without them to provide its foundation. I also owe a deal of gratitude to my graduate colleagues from the past two years: Kang Kang, Minje Kim, and Nasser Mohammadiha. Minje and Nasser have contributed especially to the theoretical dissection of the tracking algorithms presented here. I can't forget the research team at Lyric Labs that has given me the valuable opportunity to work on real-world audio problems through two summer internships. And finally, I should pay gratitude to my kick-ass (in a good way) advisor, Paris, for being a near-optimal guide during my graduate education.

In addition to these fine people, I would like to thank several late contributors to my romantic wonderment: J. S., Wolfie, Ludwig, Felix, Frederic, Charles-Valentin, Franz, Alexander, Claude, Maurice, the Sergeis, the Schus, and their many messengers. And, just for the hell of it, here's a shout-out to a revolutionary comedian for putting certain ideas in just the right way, as can be observed in the following heartfelt wish:

"May the forces of evil become confused on the way to your house."

- George Carlin



Short-time Fourier transform of an excerpt from Frederic Chopin's *Berceuse*, Op. 57. Performed on a *Verdugo e Hijo* piano in Quito, Ecuador in 2009.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ALGORITHMS

CHAPTER 1

INTRODUCTION

This chapter reviews the blind source separation (BSS) and localization problems and a number of techniques that have been applied to solve them. Particular attention is given to multichannel methods that use an array of microphones to incorporate spatial information into the separation. We first describe the physical geometry involved. Then, we briefly review beamforming (the classical approach to array processing) and time-frequency (TF) masking. The latter approach is a BSS method designed specifically for the class of signals that tend to satisfy a disjointness property. Speech is conveniently a member of this class. Following this is a discussion of methods for direction-of-arrival (DOA) estimation. We also review tracking algorithms from the Bayesian filtering literature as we often cannot assume that the sources are physically stationary. Then, methods for tracking multiple sources on the unit circle and sphere with mixtures of directional distributions are introduced. Finally, we summarize the contributions of this thesis.

1.1 Geometry of source separation and localization

To understand the multichannel source separation and localization problem, we start by looking at the physical geometry involved. We have an array with C microphones placed in a room with K sound sources. This is depicted for $C = 3$, $K = 2$, and a rectangular room in Figure 1.1. In anechoic conditions, each speaker is recorded by each microphone only once with an attenuation and delay that depends solely on the distance between them. The direct-path delay from the j^{th} source to the i^{th} microphone is denoted as d_{ij}. When the room is reverberant, multiple copies of each source signal will be recorded at each microphone, where each copy is an attenuated and delayed version of the original source signal. We will not explicitly model reverberation in this thesis. Instead, we will design algorithms that are robust to its effects.

To solve the source separation problem, we need to partition one of the recordings into K parts corresponding to each of the speakers. We can use the array to estimate the directions of the sources. Methods for isolating energy from a particular direction can then be applied to do the separation. The Degenerate Unmixing Estimation Technique [1] is a famous example of this. Intuitively, we would like to design algorithms that perform better as more microphones become available and that are robust to the effects of unknown interference and complicated reverberation.

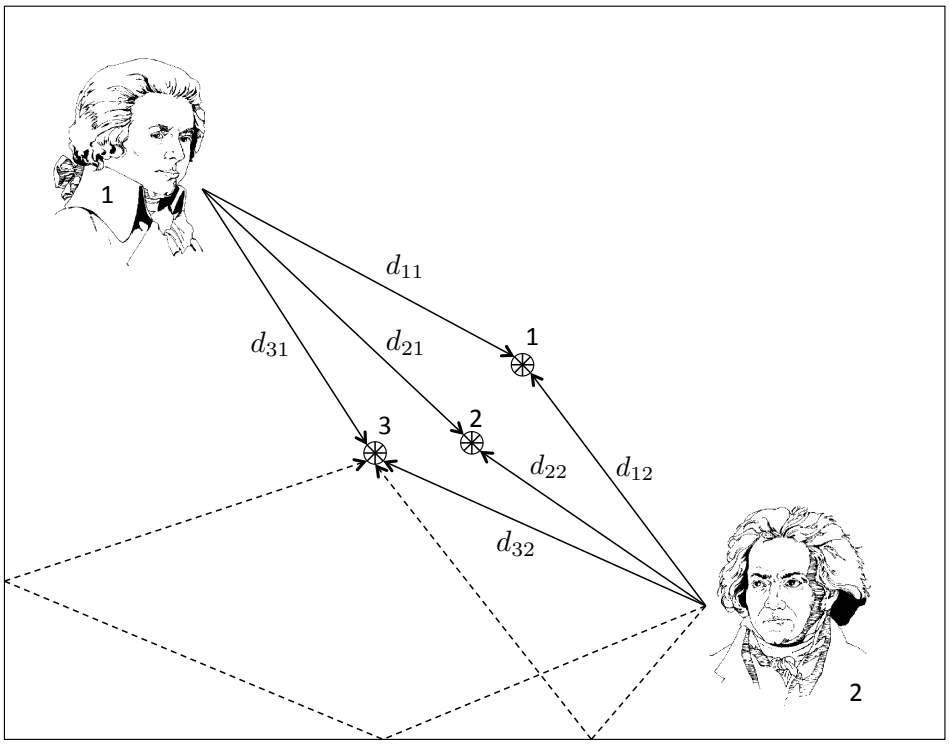


Figure 1.1: Geometry of the multichannel source separation and localization problem for $K = 2$ sources and $C = 3$ microphones (a.k.a. Wolfie and Ludwig in an argument). d_{ij} denotes the time taken for sound to propagate from source j to microphone i. Examples of first- and second-order reflections are shown as dashed lines. (The microphone separation is exaggerated for ease of interpretation.)

1.2 Multichannel blind source separation

Human beings have the remarkable ability to focus on a single speaker in a crowded, noisy environment. This is called the "cocktail party effect" and is at the center of literature on Auditory Scene Analysis (ASA) [2]. Studies in the neuroscience community on audition have revealed that the human brain does indeed perform some kind of source separation [3]. Reconstructed signals from brain scans show the salient information corresponding only to the speaker of interest, as if the other speaker(s) were not present in the experiment. This is a fascinating result that further motivates the design of automated BSS algorithms.

The literature on multichannel BSS algorithms is vast [4], [5], [6]. Independent components analysis (ICA) is one famous approach that attempts to invert a mixing matrix relating the source signals to the recorded mixtures. This matrix contains all the relevant attenuation and delay information about how the signals propagate toward the microphones. More recently, matrix decomposition techniques such as Nonnegative Matrix Factorization [7], [8] have been applied directly to the speech spectrogram. NMF has a desirable parts-based representation of the input that aligns very well with the additive nature of sounds.

In this thesis, we will focus on the time-frequency masking approach outlined in [1], which takes advantage of disjointness of speech signals in a time-frequency representation. However, a discussion of multichannel audio would not be complete without a review of the classical approach to array processing: beamforming.

1.2.1 Beamforming

Beamforming methods [9], [10] [11], [12], [13] approach source separation/enhancement from a classical array processing perspective. This involves designing a spatial filter that can be steered without moving any physical parts of the array. In general, the filter is designed to allow signals impinging on the array from particular directions to pass undistorted while blocking interfering signals incident at other angles. The filter's lobes in the desired directions are called "beams" and its zeros in the undesired directions are called "nulls," hence the terminology "beam-steering" and "null-steering."

The simplest case is that of a single directional source in ambient white noise. The optimal spatial filter, in terms of signal-to-noise ratio (SNR), is the delay-and-sum (D&S) beamformer. It delays the microphone recordings such that the source signals are aligned and then sums over the channels. This is straightforward once the DOA is known because there is a simple mapping from DOA to inter-channel delays. When the ambient noise is more structured (i.e. the noise in the channels is correlated), then we can do better with the minimum-variance distortionless response (MVDR) beamformer. For a narrowband signal, the MVDR keeps track of a channel covariance matrix and uses it to pre-whiten the inputs and further suppress noise. However, if the noise is white, the MVDR reduces to the D&S.

If multiple desired sources and directional interferers are active simultaneously, we desire a spatial filter that enforces multiple distortionless constraints and blocks the interferers. At most C such constraints can be applied at once. A flexible algorithm to achieve this is the linearly-constrained minimum-variance (LCMV) beamformer, which requires the solution to a constrained optimization problem. We can implement it efficiently by using a generalized sidelobe canceler (GSC) structure [14]. This involves separating the constraints into two orthogonal sets corresponding to beams and nulls and processing the input along two separate branches. The outputs are then combined to produce a final result.

1.2.2 Time-frequency masking

The Degenerate Unmixing Estimation Technique (DUET) [1] and its extension to more than 2 microphones [15] are based on time-frequency (TF) masking. This involves clustering of inter-channel phase and level differences (IPD, ILD) to construct a binary TF mask and is known to produce extremely clean separation of speech in non-reverberant environments. The two assumptions in DUET are that the source signals are approximately disjoint in a time-frequency representation and that at most one sample of delay is observed between the channels. Speech signals are remarkably disjoint in the short-time Fourier transform (STFT) even in the presence of strong reverberation [16]. Thus, the first assumption often holds. However, for high sampling rates or arrays with more than a few centimeters of separation between the microphones, spatial aliasing violates the second assumption. Spatial aliasing occurs when the incoming signal contains energy with a wavelength that is less than half the inter-mic spacing. Solutions include oversampling [17] and explicit modeling of phase as a wrapped quantity [18], [19], [20]. We will adopt the latter approach in this thesis.

1.2.3 Beamforming vs TF masking

A desirable property of beamforming methods is the distortionless constraint. This is important, for example, in speech enhancement algorithms where even a moderate amount of artifacts in a speaker's voice can cause discomfort to the listener. However, the drawback is that they are not designed with source separation in mind. Time-frequency (TF) masking, in contrast, was designed specifically for separating TF-disjoint signals such as speech. The ideal mask achieves near-perfect separation with only two microphones, even in the presence of strong reverberation. Another concern is that most beamforming methods assume a large number of widely-spaced channels (e.g. 10 or more) are available. This is an infeasible constraint for handheld devices. Furthermore, the criteria that beamformers typically optimize involve SNR, which is known to correlate poorly with separation quality [21]. Other metrics such as signal-to-interference ratio (SIR) are more informative. SIR, along with signal-to-distortion ratio (SDR) and signal-to-artifact ratio (SAR), have been defined in [21] for evaluating BSS algorithms.

1.3 Direction-of-arrival estimation and tracking

Array-based source localization has been an important area of research for many decades [22], [23]. The goal is to determine the position(s) of the source(s) relative to the array using only the recorded signals. For compact arrays with less than 10 cm of spacing between any pair of microphones, the far-field model is often used as a simplifying assumption. This says that the shape of the signal wavefront emitted by the source is well-approximated as planar by the time it reaches the array. For small arrays, we are better off tracking the direction-of-arrival (DOA) of a source rather than its physical position in the room as it is too difficult to estimate its distance to the array. We also need a method for following it as it moves about. Multi-target tracking algorithms generalize this idea to the case that we care about. In this thesis, algorithms for tracking multiple speakers on the unit circle and sphere are proposed that approach the problem from a Bayesian perspective.

1.3.1 DOA estimation

Consider the simple case of one source and a two-microphone array. The easiest approach to estimating the DOA is to compute the cross-correlation between the channels [24]. In the presence of moderate ambient noise, a peak will appear at the inter-channel delay corresponding to the source's position. This can be extended by applying a pre-whitening filter such as the PHAse Transform (PHAT) before the correlation. This is called the generalized cross correlation (GCC) method. A multichannel GCC (MCCC) method was developed for when more than 2 channels are available [25]. Alternatively, one can quickly compute the GCC on a grid in DOA space [26], [27]. A similar approach computes the steered response power (SRP) function, which is simply the response of your favorite beamformer as its beam is swept over DOA space [13].

Another well-known approach is the MUltiple SIgnal Classification (MUSIC) algorithm [28], which requires that more channels are available than there are sources (i.e. $C > K$). The MUSIC algorithm is based on the idea that the frequency-domain channel covariance matrix Σ contains orthogonal signal and noise subspaces. The first K eigenvectors of Σ span the space of steering vectors, $\boldsymbol{\varsigma}$, corresponding to the source DOAs θ_i

and the remaining $C - K$ eigenvectors span the noise (null) space. The norm $\|\Sigma_s \, \boldsymbol{\varsigma}(\theta)\|_2$ will be large if θ corresponds to a source direction and small otherwise, where Σ_s is the matrix whose columns span the signal subspace. We can compute this value over all directions θ to calculate a "pseudospectrum" with multiple peaks, one for each source. The pseudospectrum is usually calculated using the noise subspace matrix: $\|\Sigma_n \, \boldsymbol{\varsigma}(\theta)\|_2^{-1}$.

Many variations on the basic MUSIC algorithm exist including root-Music [29], which finds the roots of a polynomial, and Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT) [30], which takes advantage of special array structures. The DUET and ESPRIT methods were combined to form DESPRIT [31].

1.3.2 Tracking with Bayesian filters

Multichannel BSS algorithms are often made spatially adaptive to separate moving sources by tracking their directions-of-arrival (DOA) over time. This is typically done by applying the GCC or MUSIC methods adaptively on a short-term basis [32]. However, the estimates may be noisy, especially in reverberant environments, so we would ideally like to smooth them out. This motivates tracking with the Kalman filter [33], [34] and related methods.

The Bayesian filtering framework allows for sophisticated, statistically-grounded approaches to tracking. The dynamical systems involved are often non-linear, so the extended Kalman filter (EKF) [35] and unscented Kalman filter (UKF) [36] are used to linearize the dynamics. The EKF uses a first-order Taylor series expansion of the non-linearity. This may be inadequate as the approximation fails to capture higher-order effects. The UKF remedies this by choosing a number of deterministically-chosen "sigma points" that are fed through the system dynamics. This approach uses the unscented transform [37] to keep track of higher-order statistics.

For highly non-linear, non-Gaussian systems, the particle filter (PF) is a flexible alternative [38], [39], [40], [41]. More sophisticated strategies use the EKF or UKF as a sub-component in the PF to reduce the variance of the state estimate [42], [43]. This is especially effective in noisy, reverberant environments [44]. The Gaussian sum filter (GSF) [45] and its extensions [46], [47], [48], in particular, are important as they enable the tracking of non-Gaussian and possibly multimodal state distributions. An extension of GSF techniques leads to particle filters that are designed for the complicated task of tracking multiple sources simultaneously [43], [49], [50], [51].

1.3.3 Wrapped filtering

In this thesis, we are interested in tracking on the unit circle and sphere. This shows up in many applications including phase-locked loops [52], phase unwrapping [53], source localization [9], and more. Work on 2D phase unwrapping motivates use of the Kalman filter framework [54], [55] for circular data. However, deterministic methods have also been explored in [56], [57].

Our primary goal is to design Bayesian filters that operate *directly* on the unit circle and sphere. This is in contrast to many approximate methods that model these spaces indirectly by, for example, filtering in the embedding spaces: $\mathbb{R}^2$ and $\mathbb{R}^3$. We will use the wrapped Gaussian (WG) distribution [58] to derive a

wrapped Kalman filter (WKF) for tracking a wrapped dynamical system (WDS). We will show that modeling the state of the WDS explicitly with a directional distribution reduces the tracking variance significantly over 2D Gaussian methods. The WG has also been used to learn source trajectories with a wrapped-phase hidden Markov model [59]. A WG state model has the advantage of being closely related to the conventional Gaussian distribution for which optimal inference procedures exist. We will use a mixture of WGs [60] to extend the WKF to the case of multiple sources, yielding the *factorial wrapped Kalman filter* (FWKF).

The von Mises-Fisher (vMF) distribution [58] has been used to model the state distribution of a dynamical system on the unit sphere [61]. We adopt this strategy and generalize it by modeling the state distribution with a mixture of vMFs to yield the *factorial von Mises-Fisher particle filter* (FvMFPF). In speaker tracking problems with a compact array in 3 dimensions, it is infeasible to estimate range values (i.e. distance to the target). Thus, the localization problem is more appropriately posed as that of estimating directions-of-arrival (DOA) only. The FvMFPF is a natural solution since it models the source positions explicitly on the sphere.

1.3.4 Data association ambiguities in multi-source tracking

The main complication with extending filters to handle multiple sources is the data association problem. We will assume that each source evolves independently of the others and generates its own observation sequence. However, one observes the unordered set of measurements. Two famous methods for resolving this ambiguity are Probabilistic Data Association (PDA) [62] and Multiple Hypothesis Tracking (MHT) [63]. The PDA approach combines measurements in a probabilistic fashion so that all the data gets its (proportional) chance to affect the state estimate. On the other hand, MHT keeps track of several hypotheses about how measurements are associated to target tracks. These are propagated into the future in the hopes that any ambiguities will quickly be resolved with additional data. This problem has also been tackled via acceptance region methods [64], hidden Markov modeling, [65], Gaussian mixture modeling of time-delay-of-arrival data [66], recursive EM-based approaches [67], [68], and a particle filter with TF masking-based data association [69]. Particle filtering strategies for acoustic DOA tracking have also been explored in [70] and [71]. In this thesis, we will use soft assignments of measurements/particles to clusters in the FWKF and FvMFPF to effectively "integrate out" the ambiguities. This is most aligned with the PDA framework and fits naturally in the probabilistic framework of Bayesian filtering with mixtures.

1.4 Contributions

In previous work [20], the von Mises distribution [58] was used to model wrapped inter-channel phase difference (IPD) features as circular-linear data [72]. The features are modified from those presented in [1] to explicitly incorporate spatial aliasing into a statistical model. The BSS problem was reduced to one of multimodal circular-linear regression, which can be interpreted as fitting several helices to data that lies on a cylinder. The RANdom SAmple Consensus (RANSAC) algorithm [73] was applied to quickly and robustly perform the fitting.[1] The resulting wrapped lines provide a clustering of the features and thus a method for constructing time-frequency masks.

[1] A similar approach was taken in [74]. RANSAC was also used for source localization in [75].

In this thesis, we consider the use case of speaker separation with a microphone array when the sources are stationary and when they are moving [76], [77], [78]. The filters presented here require a method by which DOA votes can be extracted from the recorded speech to be used as measurements. We will apply the IPD clustering algorithm from [20] to extract short-time DOA estimates. Alternatively, a von Mises particle filter (vMPF) is proposed that uses the IPD features directly as observations. We present the WKF, FWKF, vMFPF, FvMFPF, and vMPF not as complete DOA tracking systems, which can be highly elaborate, but as potential components in such an engine.

The contributions of this thesis are:

- a probabilistic formulation for the wrapped IPD clustering problem
- a detailed account of the RANSAC-based source separation algorithm presented in [20]
- the wrapped Kalman filter (WKF) for tracking a moving source on the unit circle
- the factorial wrapped Kalman filter (FWKF), an extension of the WKF for multi-source tracking on the unit circle
- the factorial von Mises-Fisher particle filter (FvMFPF) for multi-source tracking on the unit sphere
- the von Mises particle filter (vMPF) for tracking on the unit circle with raw IPD features
- a discussion of the measurement/particle assignment ambiguities involved in multi-source tracking and how the proposed filters resolve them in a Bayesian setting
- experiments demonstrating the utility of the proposed methods for tracking and separating speakers with a microphone array

CHAPTER 2

THEORETICAL TOOLS

This chapter serves as a collage of theoretical machinery that will be used in the rest of the thesis. The short-time Fourier transform (STFT) is introduced as a means to convert time-domain signals received at the microphones into a more useful time-frequency representation. Useful theory from the directional statistics literature and EM algorithms to fit mixtures of directional distributions are reviewed. Then, inter-channel differences (IPD) are extracted from the channel STFTs. It is shown that these features compose a circular-linear dataset and that wrapped lines in IPD space correspond to speakers. The blind source separation problem is thus reduced to one of multimodal circular-linear regression. The direction-of-arrival (DOA) of a speaker is related to the slope of an IPD line, showing that solving the wrapped line-fitting problem automatically provides an estimate of the source positions. Following this is a review of the Bayesian filtering framework including particle filtering and tracking with mixture models in the presence of data association ambiguities. Finally, the RANdom SAmple Consensus (RANSAC) algorithm is presented as a fast, heuristic approach to line-fitting that is highly robust to outliers.

2.1 Short-time analysis of non-stationary signals

A discrete-time sound signal

$$\mathbf{x} = [x[0], x[1], \ldots, x[n], \ldots, x[N-2], x[N-1]] \tag{2.1}$$

is a sampled version of an acoustic waveform recorded by a microphone. In this thesis, we will work with the short-time Fourier transform (STFT) [79] of $\mathbf{x}$ as this provides a more useful and interpretable representation of its contents. A discrete-time speech signal and the magnitude portion of its STFT (also called a spectrogram) are shown in Figure 2.1. The signal's statistics across frequency and time are far more apparent in the latter figure. We can see the distinctive speech harmonics during vowels, for example, and high-frequency broadband noise bursts during "s" and "t" sounds.

2.1.1 Short-time Fourier transform

The STFT is a sequence of overlapping discrete Fourier transforms (DFT) [79]. The DFT is defined as the mapping $\mathcal{F} : \mathbb{C}^N \to \mathbb{C}^N$ such that:

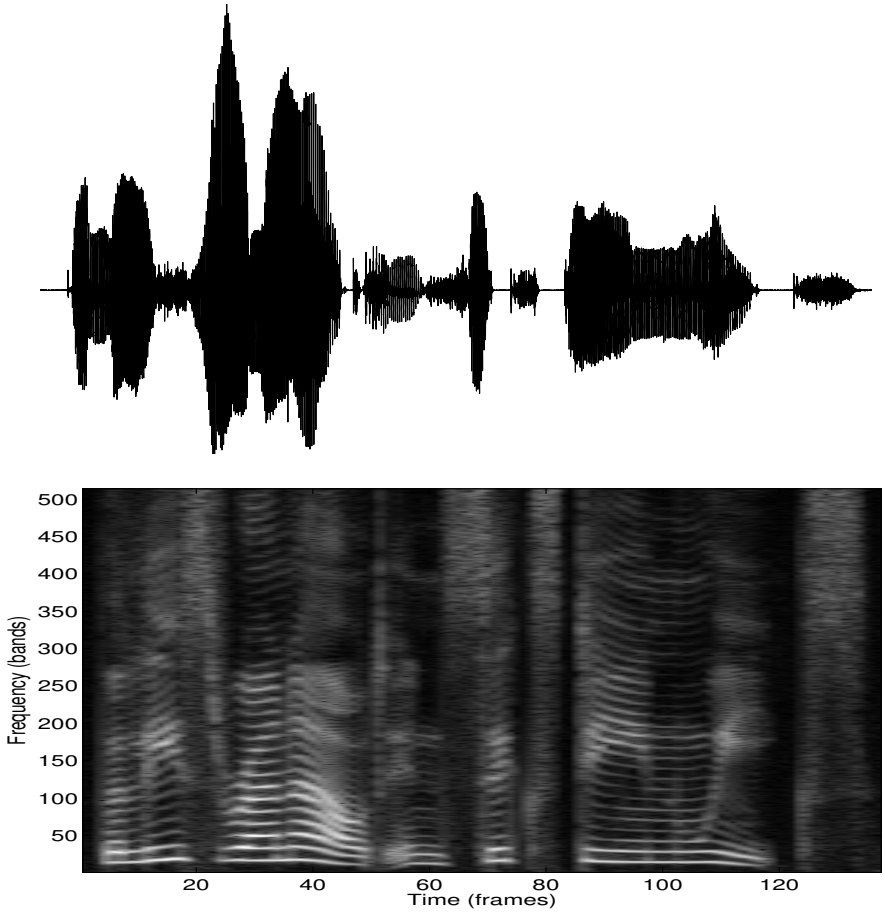


Figure 2.1: (Top) Time domain waveform of a female TSP speaker saying "the lease ran out in sixteen weeks." (Bottom) Magnitude of the corresponding short-time Fourier transform. Brightness indicates how much energy is contained in each time-frequency bin.

$$X[f] = \mathcal{F}(\mathbf{x}) = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi f}{N}n} \quad , \tag{2.2}$$

and the inverse DFT is similarly defined as the mapping $\mathcal{F}^{-1} : \mathbb{C}^N \rightarrow \mathbb{C}^N$ such that:

$$x[n] = \mathcal{F}^{-1}(\mathbf{X}) = \frac{1}{N} \sum_{f=0}^{N-1} X[f] e^{j\frac{2\pi f}{N}n} \quad . \tag{2.3}$$

Since speech is non-stationary (it changes with time), we would like to define a transformation to the Fourier domain that depends on time. The STFT is exactly what we need and is defined as the mapping $STFT(\mathbf{x}) : \mathbb{C}^M \rightarrow \mathbb{C}^{N \times T}$, where $N = 2D$ is the length of each DFT and T is the number of frames required to capture the non-zero content in the signal:

$$X_t[f] = \mathcal{F}\left(\mathbf{w} \odot \mathbf{x}_t\right) = \sum_{n=0}^{N-1} w[n] x_t[n] e^{-j\frac{2\pi f}{N}n} \quad , \quad x_t[n] = x[n + th] \ . \tag{2.4}$$

The symbol $\odot$ indicates element-wise multiplication and $\mathbf{w} \in \mathbb{R}^N$ is an analysis window function used to select and weight a portion of the time-domain signal for each DFT. The signal is shifted so that the window captures the samples required for the t^{th} DFT. The shift is parameterized by a hop size h that is typically chosen to be one quarter of the window length: $h = N/4$. Audio signals are real-valued, so the DFT will satisfy a symmetry property: the first D coefficients will be the reversed complex conjugate of the last D coefficients. Thus, we can discard the second half during processing as it provides no additional information.

The analysis window is useful to prevent artifacts in time-domain reconstructions when alterations are made to the STFT. Such alterations are common in source separation algorithms. We choose the Hanning window:

$$w[n] = \begin{cases} 0.5 - 0.5\cos\left(\frac{2\pi n}{N-1}\right) & , \quad 0 \leq n \leq N-1 \\ 0 & , \quad \text{else} \end{cases} \tag{2.5}$$

because it conveniently tapers to zero at the boundaries and is known to produce high-quality reconstructions in BSS applications compared to most other choices.

To reconstruct a time-domain signal from a (possibly modified) STFT $\tilde{\mathbf{X}}$, we apply the inverse STFT using the analysis window as a synthesis window:

$$\tilde{\mathbf{x}} = \sum_{t=0}^{T-1} \left[\mathbf{w} \odot \mathcal{F}^{-1}\left(\tilde{\mathbf{X}}_t\right)\right] * \delta[n - th] \ , \tag{2.6}$$

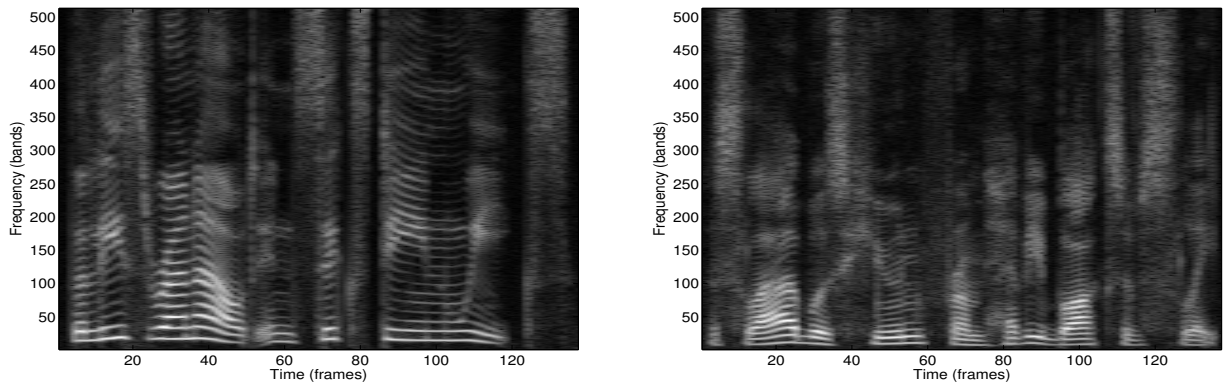
where $*$ denotes convolution and $\delta[n-k]$ is the Kronecker delta shifted by k samples. The convolution operation shifts the windowed inverse DFTs into place. A condition for perfect reconstruction from an unaltered STFT is that the sum of squared windows:

$$w_s[n] = \sum_{t=0}^{T-1} w[n - th]^2 \ , \tag{2.7}$$

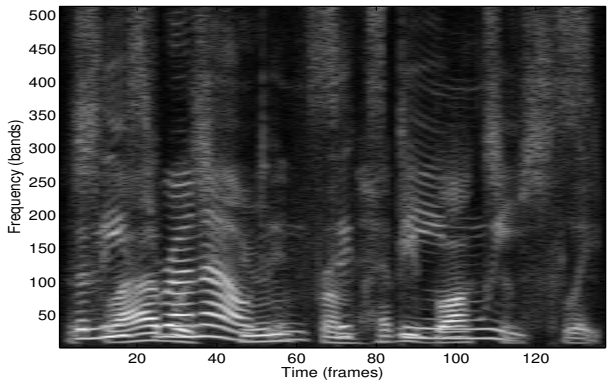
remain constant over the domain where the signal is non-zero [80]. If it does not, we should divide the reconstruction element-wise by this sum. However, if we choose the hop size to be a quarter of the window length and we use a Hanning window, the optimality condition is satisfied and no division is necessary.

2.1.2 Time-frequency masking

A special property of speech signals is that they tend not to overlap in the STFT. This is known as W-disjoint orthogonality [16], or simply disjointness, and is very useful for source separation using time-frequency (TF) masks [1]. Consider the case of two speaker signals $\mathbf{s}^{(1)}$ and $\mathbf{s}^{(2)}$ with STFTs $\mathbf{S}^{(1)}$ and $\mathbf{S}^{(2)}$ that are active simultaneously (they are talking over each other). Spectrograms of the clean speech signals from the TSP



(a)



(b)

Figure 2.2: Clean and mixed speech spectrograms. (a) Spectrograms of one female (left) and one male (right) TSP speaker. The sentences are "the lease ran out in sixteen weeks" and "the slab was hewn from heavy blocks of slate." (b) Spectrogram of two TSP speakers after mixing. Disjointness of speech in the STFT domain allows us to approximate the mixture spectrogram as the sum of the clean speech spectrograms.

database [81] and their mixture are shown in Figure 2.2(a) and Figure 2.2(b). The signals are considered to be disjoint if

$$\forall \ t, f \quad S_{t,f}^{(1)} \cdot S_{t,f}^{(2)} = 0 \ . \tag{2.8}$$

The ideal binary mask (IBM) needed to reconstruct approximations of the individual speakers is a matrix $\mathbf{M}$ of ones and zeros such that:

$$M_{t,f} = \begin{cases} 1 & , \quad |S_{t,f}^{(1)}|^2 > |S_{t,f}^{(2)}|^2 \\ 0 & , \quad \text{otherwise} \end{cases} \ . \tag{2.9}$$

Applying this mask element-wise to the mixture, $\mathbf{X} = STFT\left(\mathbf{s}^{(1)} + \mathbf{s}^{(2)}\right)$, will isolate the energy from the first speaker. Similarly, applying the inverse mask, $1 - \mathbf{M}$, will isolate the energy from the second speaker. IBMs for the example signals are shown in Figure 2.3. The inverse STFT can then be applied to the masked STFTs to reconstruct separated time-domain signals. Spectrograms of the separated signals are shown in

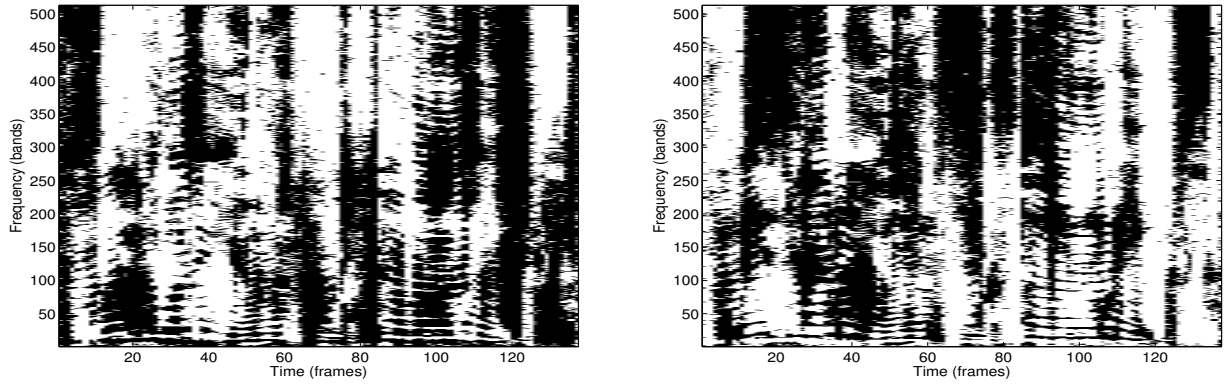


Figure 2.3: Ideal binary masks (IBM) for separating the two speakers from the mixture. The IBM assigns time-frequency bins according to which of the two speakers has the most energy.

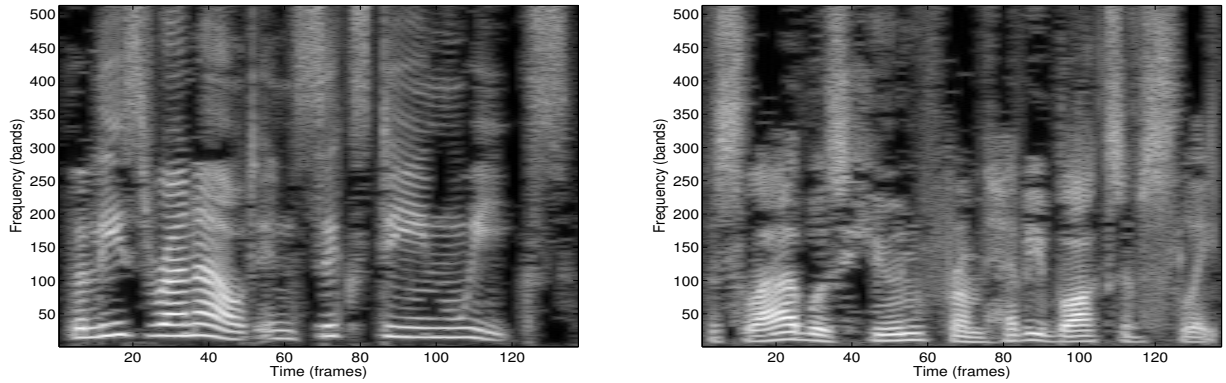


Figure 2.4: Spectrograms of separated speech after applying time-frequency masks. Comparing this to the original spectrograms shows that masking works extremely well for speech.

Figure 2.4. The IBM is considered optimal in the sense that it maximizes the signal-to-noise ratio (SNR) of either reconstruction [82]. Binary TF masking has been shown to give very good separation results in anechoic conditions precisely because of the disjointness of speech signals in the STFT [1]. One drawback is that a poorly-estimated mask can introduce distracting musical noise. In practice, further processing is needed to clean up the separation.

2.2 Directional statistics

In this section, we review material from the directional statistics literature [58], [83], [84] that will be useful for modeling phase in the STFT and the direction-of-arrival (DOA) of a sound source. This includes the wrapped Gaussian (WG) and von Mises (vM) distributions on the unit circle and the von Mises-Fisher (vMF) distribution on the unit sphere as well as algorithms for sampling from them. Rotations on the unit sphere are also discussed as this will be useful for speaker tracking.

2.2.1 Directional distributions

We will find use for statistical models of circular quantities. Classic examples are time of day, wind direction, and sinusoidal phase. Such quantities cannot be modeled with a distribution on the real line, $\mathbb{R}^1$, because they actually lie on a sub-interval of $\mathbb{R}^1$, i.e. $[0, 12]$ or $[-\pi, \pi]$. The latter set defines the wrapped interval

$$\mathbb{S}^1 = \{\theta : \theta \in [-\pi, \pi]\} \ . \tag{2.10}$$

Alternatively, we can represent each scalar angle with a unit vector in $\mathbb{R}^2$:

$$\mathbb{S}^1 = \{\mathbf{x} : \|\mathbf{x}\|_2 = 1 \ , \ \mathbf{x} \in \mathbb{R}^2\} \ . \tag{2.11}$$

Both define a valid wrapped sample space since the endpoints represent the same index, but we will find the former representation more useful.

An intuitive way to understand the nature of $\mathbb{S}^1$ is to consider an appropriate measure of centrality. On the real line, the arithmetic mean of a dataset $\mathbf{X} = \{x_i\}$, $i = 1, \ldots, N$, is calculated as

$$\widehat{\mu} = \frac{1}{N} \sum_{i=1}^{N} x_i \ . \tag{2.12}$$

This is the maximum-likelihood estimator (MLE) of the mean for data that is assumed to have been sampled from a Gaussian distribution. However, consider a dataset of angle measurements $\boldsymbol{\theta} = \{\theta_i\}$, $i = 1, \ldots, N$ that lie on the interval $[-\pi, \pi]$. If they are concentrated near the boundaries $-\pi$ and π, the arithmetic mean will return a value near 0, which is incorrect. Instead, we should calculate a circular measure of centrality:

$$\widehat{\mu} = \angle \sum_{i=1}^{N} e^{j\theta_i} = \angle \left(\sum_{i=1}^{N} \cos(\theta_i) + j \sin(\theta_i) \right) = \tan^{-1} \left(\frac{\sum_{i=1}^{N} \sin(\theta_i)}{\sum_{i=1}^{N} \cos(\theta_i)} \right) \ . \tag{2.13}$$

This is an unbiased estimator for the mean of a wrapped Gaussian (WG) distribution and the MLE for the mean of a von Mises (vM) distribution.

Wrapped Gaussian distribution

The probability density function (pdf) of the WG is given as:

$$P\left(\theta\,;\, \mu, \sigma^2\right) = \sum_{l=-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\theta - \mu + 2\pi l)^2}{2\sigma^2}} \quad , \quad -\pi \leq \theta \leq \pi \ , \tag{2.14}$$

and is the result of transforming a Gaussian random variable x via the mapping $\psi : \mathbb{R}^1 \to \mathbb{S}^1$:

$$\theta = \psi(x) = \mathrm{mod}(x + \pi, 2\pi) - \pi \ . \tag{2.15}$$

We can visualize the WG on the unit circle in $\mathbb{R}^2$ (left panel of Figure 2.5) or directly in $\mathbb{S}^1$ (right panel

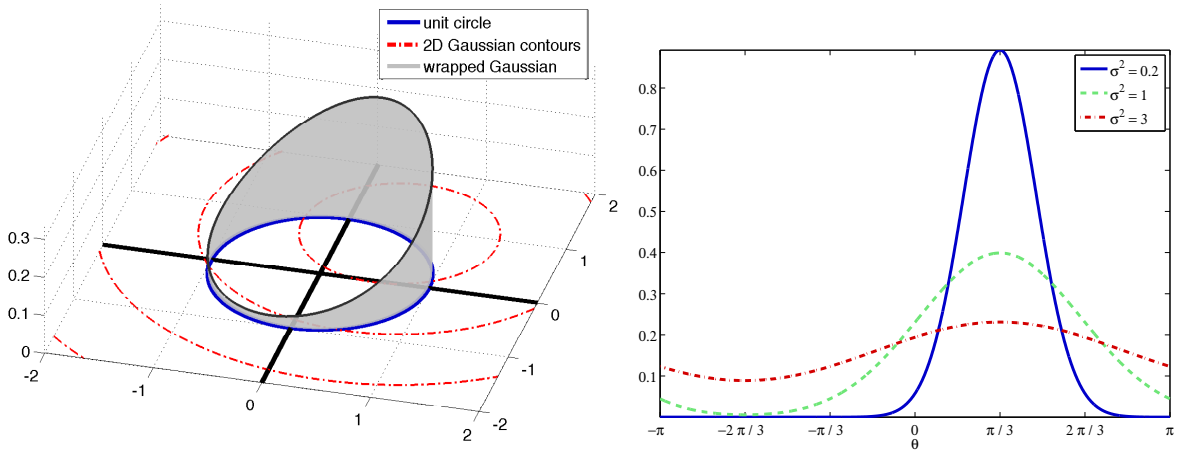


Figure 2.5: (Left) Wrapped Gaussian pdf ($\mu = \frac{\pi}{3}$) on the unit circle in $\mathbb{R}^2$ shown with 2D Gaussian contours ($\sigma^2 = 0.8$). (Right) WG pdf in $[-\pi, \pi]$ ($\mu = \frac{\pi}{3}$ and varying σ^2). The θ axis is the unit circle, unfolded.

of Figure 2.5). Its close relationship with the conventional univariate Gaussian distribution will make it possible to derive approximate closed-form expressions for several algorithms in this thesis.

von Mises distribution

We will also find use for the von Mises (vM) distribution. The pdf of the vM is given as

$$P(\theta\,;\,\mu,\kappa) = \frac{1}{2\pi I_0(\kappa)} e^{\kappa\cos(\theta-\mu)} \quad , \quad -\pi \le \theta \le \pi \quad , \tag{2.16}$$

and is the result of conditioning a 2D Gaussian, $\mathcal{N}\left(\boldsymbol{\mu}, \sigma^2\mathbf{I}\right)$, $\|\boldsymbol{\mu}\|_2 = 1$, on the unit circle and converting from Cartesian to polar coordinates (see Appendix A). The conditioning results in that $\kappa = 1/\sigma^2$. The vM may be more convenient than the WG when we only care to evaluate the pdf rather than derive an algorithm. The vM becomes a Dirac delta at μ for $\kappa \to \infty$ and the uniform distribution on $\mathbb{S}^1$ for $\kappa \to 0$. It looks very similar to the WG.

von Mises-Fisher distribution

We can define a sample space on the unit sphere:

$$\mathbb{S}^2 = \{\mathbf{x} : \|\mathbf{x}\|_2 = 1\,,\, \mathbf{x} \in \mathbb{R}^3\} \quad . \tag{2.17}$$

This representation turns out to be most convenient, although an equivalent parameterization in terms of spherical coordinates (azimuth θ and zenith ϕ) is:

$$\mathbb{S}^2 = \{[\theta, \phi] : \theta \in [-\pi, \pi]\,,\, \phi \in [0, \pi]\} \quad . \tag{2.18}$$

The latter definition is less useful in general because it defines a 2D rectangle that results from applying a

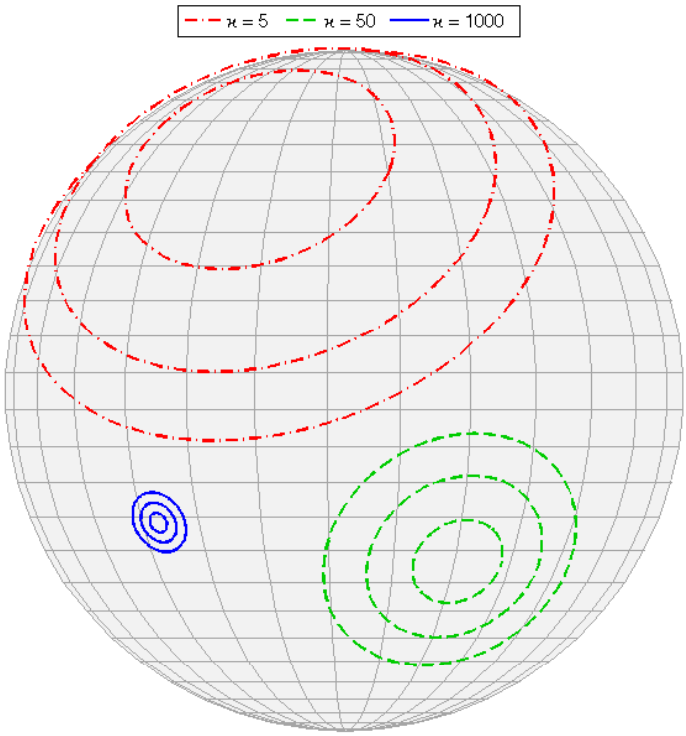


Figure 2.6: Contours of the von Mises-Fisher distribution on the unit sphere for three values of the concentration parameter κ.

highly non-linear mapping to the former (degenerate) 3D set. This complication arises from the fact that it is impossible to gracefully wrap a rectangle around a sphere.

The classic distribution on the unit sphere is a generalization of the von Mises called the von Mises-Fisher (vMF). It is parameterized by a mean direction $\boldsymbol{\mu}$, $\|\boldsymbol{\mu}\|_2 = 1$, and concentration κ:

$$P(\mathbf{x}\,;\,\boldsymbol{\mu},\kappa) = \frac{\kappa}{2\pi\left(e^{\kappa} - e^{-\kappa}\right)} e^{\kappa \mathbf{x}^T \boldsymbol{\mu}} \quad , \quad \mathbf{x} \in \mathbb{S}^2 \quad . \tag{2.19}$$

The contours of various vMFs are depicted in Figure 2.6. The vMF can be derived by conditioning a 3D Gaussian, $\mathcal{N}\left(\boldsymbol{\mu}, \sigma^2 \mathbf{I}\right)$, $\|\boldsymbol{\mu}\|_2 = 1$, on the unit sphere (see Appendix A). It becomes a Dirac delta at $\boldsymbol{\mu}$ for $\kappa \to \infty$ and the uniform distribution on $\mathbb{S}^2$ for $\kappa \to 0$. The normalization constant may be unstable for large κ because of the $(e^{\kappa} - e^{-\kappa})$ term. We can prevent this by working in the log domain[1] [85]:

$$\log\left(e^{\kappa} - e^{-\kappa}\right) = \log\left(e^{\kappa} - e^{-\kappa}\right) + \kappa - \kappa = \log\left(\left[e^{\kappa} - e^{-\kappa}\right] e^{-\kappa}\right) + \kappa = \log\left(1 - e^{-2\kappa}\right) + \kappa \quad . \tag{2.20}$$

2.2.2 Sampling from directional distributions

In this thesis, we will need to sample from all three directional distributions described in the previous section.

[1] Thanks to Jeff Bernstein for pointing this out.

Sampling from a wrapped Gaussian distribution

The WG is easy since we can sample from a conventional Gaussian on $\mathbb{R}^1$ and apply the wrap mapping in Equation (2.15).

Sampling from a von Mises distribution

It is not known how to sample from the vM directly. However, it was shown in [86] that this can be done with rejection sampling [87] using a wrapped Cauchy (WC) envelope. The envelope upper-bounds the vM distribution so that we can sample directly from the WC and accept with a probability equal to the ratio of the vM pdf and WC envelope.

Alternatively, we could use importance sampling [87]. A convenient choice for the proposal distribution is the wrapped Gaussian. We simply need to choose the standard deviation to minimize the variance of the importance weights. An easy way to do this is to choose the σ that minimizes the KL divergence between the target vM and the proposal wG. Without loss of generality, we can consider the case of $\mu = 0$. The KL divergence is:

$$D(p\|q) = \int_{-\pi}^{\pi} p(x\,;\,0,\kappa) \log\left(\frac{p(x\,;\,0,\kappa)}{q(x\,;\,0,\sigma^2)}\right)\, dx \tag{2.21}$$

$$\propto -\int_{-\pi}^{\pi} \frac{1}{2\pi I_0(\kappa)} e^{\kappa\cos(x)} \log\left(\sum_{l=-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x+2\pi l)^2}{2\sigma^2}}\right)\, dx \tag{2.22}$$

This expression is analytically intractable, so the integration must be performed numerically [88] (truncating the infinite sum, of course). The relationship between κ and the optimally matched σ is shown in Figure 2.7 along with the KL divergence as a function of κ. The worst fit occurs at $\kappa \approx 2$. We will see later that we only care about the case when $\kappa \gg 2$, so the mismatch is not a problem.

Sampling from a von Mises-Fisher distribution

The von Mises-Fisher is more straightforward. It is described in [85], [89] that for $\boldsymbol{\mu} = [1, 0, 0]^T$,

$$\mathbf{x} = \begin{bmatrix} W & \sqrt{1 - W^2}\mathbf{V}^T \end{bmatrix}^T \quad , \tag{2.23}$$

is vMF-distributed, where

$$\mathbf{V} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \end{bmatrix}^T \quad , \quad \theta \sim U(0, 2\pi) \ , \tag{2.24}$$

and the pdf of W is

$$f_W(w) = \frac{\kappa}{2\sinh(\kappa)} e^{\kappa w} \quad , \quad w \in [-1, 1] \ . \tag{2.25}$$

The inverse CDF method is used to simulate W:

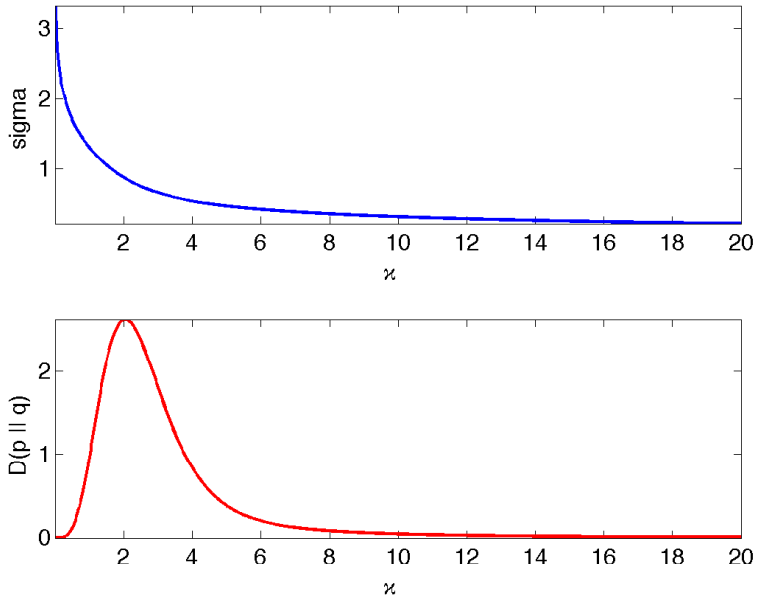


Figure 2.7: (Top) Optimal relationship between von Mises concentration κ and wrapped Gaussian σ for importance sampling from the vM. (Bottom) Corresponding KL divergence between the distributions.

$$W \sim F_W^{-1}(u) = \frac{1}{\kappa} \log\left(e^{-\kappa} + 2\sinh(\kappa)\, u\right) \quad , \quad u \sim U(0,1) \ . \tag{2.26}$$

For $\boldsymbol{\mu} \neq [1,0,0]^T$, a rotation is applied.

2.2.3 Rotations on the unit sphere

There are several methods for rotating vectors in $\mathbb{S}^2$. We will use rotations about an axis and rotations by an azimuth and elevation pair. One or the other might be more convenient depending on the situation.

Rotation about an axis

To rotate a vector $\mathbf{x} \in \mathbb{S}^2$ about an axis $\mathbf{r}$ by an angle ν, we premultiply it by the following matrix:

$$\mathbf{R}(\mathbf{r}, \nu) = \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix} \sin(\nu) + \left(\mathbf{I} - \mathbf{r}\mathbf{r}^T\right) \cos(\nu) + \mathbf{r}\mathbf{r}^T \ . \tag{2.27}$$

Rotation by azimuth and elevation angles

To rotate a vector $\mathbf{x} \in \mathbb{S}^2$ by azimuth and elevation angles $[\nu, \eta]$, we perform two rotations back to back. First, we rotate about the z-axis to change the azimuthal orientation:

$$\mathbf{y} = \mathbf{R}\left(\begin{bmatrix}0 & 0 & 1\end{bmatrix}^T, \nu\right)\mathbf{x} = \begin{bmatrix}\cos(\nu) & -\sin(\nu) & 0\\ \sin(\nu) & \cos(\nu) & 0\\ 0 & 0 & 1\end{bmatrix}\mathbf{x} \ . \tag{2.28}$$

Applying the same rotation to the y-axis gives a vector $\mathbf{r}$ about which we can rotate by η to change the elevation:

$$\mathbf{z} = \mathbf{R}(\mathbf{r}, \eta)\ \mathbf{y} \ . \tag{2.29}$$

2.3 Fitting mixtures of directional distributions

We will find it useful to be able to fit mixtures of WGs (MoWG), mixtures of vMs (MovM), and mixtures of vMFs (MovMF) to directional datasets. This can all be accomplished by the Expectation-Maximization (EM) algorithm [87], [90], [91].

2.3.1 Expectation-Maximization

EM is a learning algorithm for maximum-likelihood problems with hidden variables. In the case of a mixture model, we have observed variables $\mathbf{x}$, unobserved variables $\mathbf{z}$, and parameters $\boldsymbol{\Theta}$ (to be learned). $\boldsymbol{\Theta}$ includes parameters of the components in the mixture as well as the component weights $\boldsymbol{\pi}$. An underlying generative model is assumed in which the data is drawn i.i.d. from the mixture. The hidden variables serve to indicate what component each data point was sampled from. The pdf of the mixture is given as:

$$P(\mathbf{x}\,;\,\boldsymbol{\Theta}) = \sum_{j=1}^{K} \pi_j\, P(\mathbf{x}\,;\,\boldsymbol{\Theta}_j) \ , \tag{2.30}$$

where $P(\mathbf{x}\,;\,\boldsymbol{\Theta}_j)$ is the probability model (pdf) of the j^{th} component evaluated at $\mathbf{x}$. If we incorporate the unknown assignments $\mathbf{z}$, the result is the complete data likelihood:

$$\mathcal{L} = \prod_{i=1}^{N} P(\mathbf{x}_i, \mathbf{z}_i\,;\,\boldsymbol{\Theta}) \tag{2.31}$$

$$= \prod_{i=1}^{N}\sum_{j=1}^{K} P(\mathbf{x}_i|z_{ij}\,;\,\boldsymbol{\Theta}_j)P(z_{ij}\,;\,\boldsymbol{\Theta}_j) \tag{2.32}$$

$$= \prod_{i=1}^{N}\prod_{j=1}^{K} [P(\mathbf{x}_i\,;\,\boldsymbol{\Theta}_j)P(z_{ij}\,;\,\boldsymbol{\Theta}_j)]^{z_{ij}} \tag{2.33}$$

$$= \prod_{i=1}^{N}\prod_{j=1}^{K} [P(\mathbf{x}_i\,;\,\boldsymbol{\Theta}_j)\ \pi_j]^{z_{ij}} \ . \tag{2.34}$$

The quantity $P(\mathbf{x}_i, \mathbf{z}_i\,;\,\boldsymbol{\Theta})$ is the complete data likelihood for the i^{th} observation $\mathbf{x}_i$ and $\pi_j = P(z_{ij}\,;\,\boldsymbol{\Theta}_j)$

is the mixing weight of the j^{th} component. The hidden variables z_{ij} are treated as indicator variables for each i in the above notation. So, for the i^{th} observation $\mathbf{x}_i$, z_{ij} takes the value 1 for a single index j and 0 for all others, which means that the j^{th} component generated the i^{th} data point. This has the effect of selecting one term in the product over j for each i. It is easier to work with the log likelihood:

$$\log \mathcal{L} = \sum_{i=1}^{N} \sum_{j=1}^{K} \log \left[P(\mathbf{x}_i \,;\, \boldsymbol{\Theta}_j)\ \pi_j \right] z_{ij} \quad . \tag{2.35}$$

This is easy to maximize with respect to the parameters $\boldsymbol{\Theta}_j$ if we know the values of the indicator variables z_{ij}. In that case, we can just estimate the parameters for the j^{th} component using all the data whose indicator is active for that component (i.e. $z_{ij} = 1$). Although we do not know the z_{ij}'s, we can derive a tractable lower-bound on the log likelihood by taking its expected value with respect to the hidden variables. This gives what is known as the "Q function":

$$Q = \mathrm{E}_{\mathbf{z}|\mathbf{x},\boldsymbol{\Theta}^{\text{old}}} \left[\log \mathcal{L} \right] = \sum_{i=1}^{N} \sum_{j=1}^{K} \log \left[P(\mathbf{x}_i \,;\, \boldsymbol{\Theta}_j)\ \pi_j \right] \eta_{ij} \quad , \tag{2.36}$$

where the posterior probabilities

$$\eta_{ij} = \mathrm{E}_{\mathbf{z}|\mathbf{x},\boldsymbol{\Theta}^{\text{old}}}[z_{ij}] = P(z_{ij} \mid \mathbf{x}_i \,;\, \boldsymbol{\Theta}^{\text{old}}) = \frac{P(\mathbf{x}_i \mid z_{ij} \,;\, \boldsymbol{\Theta}^{\text{old}})\ P(z_{ij} \,;\, \boldsymbol{\Theta}^{\text{old}})}{\sum_{j=1}^{K} P(\mathbf{x}_i \mid z_{ij} \,;\, \boldsymbol{\Theta}^{\text{old}})\ P(z_{ij} \,;\, \boldsymbol{\Theta}^{\text{old}})} = \frac{P(\mathbf{x}_i \,;\, \boldsymbol{\Theta}_j^{\text{old}})\ \pi_j}{\sum_{j=1}^{K} P(\mathbf{x}_i \,;\, \boldsymbol{\Theta}_j^{\text{old}})\ \pi_j} \quad , \tag{2.37}$$

represents how likely it is that the j^{th} component in the mixture is responsible for generating the i^{th} observation. This follows since the expectation of an indicator variable is equal to its probability of being 1.

The Q function is easier to maximize and leads to the EM algorithm. In the E step, we fix the current estimate of the parameters $\boldsymbol{\Theta}$ and calculate the posterior probabilities $\boldsymbol{\eta}$. This captures how much each data point $\mathbf{x}_i$ contributes to estimating the parameters of each component $\boldsymbol{\Theta}_j$. Then, in the M step, we use these posteriors as *soft weights* to update the model parameters via maximization of Equation (2.36). Data points with higher weights for a specific value of j will exert more influence on the update of the j^{th} component's parameters. After the M step, $\boldsymbol{\Theta}$ has changed, so $\boldsymbol{\eta}$ has changed. We re-estimate $\boldsymbol{\eta}$, update $\boldsymbol{\Theta}$, and repeat until convergence.

2.3.2 Fitting a mixture of wrapped Gaussian distributions

The mixture of wrapped Gaussians (MoWG) [60] is given as

$$P\left(x \,;\, \boldsymbol{\mu}, \boldsymbol{\sigma}^2, \boldsymbol{\pi}\right) = \sum_{j=1}^{K} \pi_j \sum_{l=-\infty}^{\infty} \mathcal{N}\left(x \,;\, \mu_j + 2\pi l, \sigma_j^2\right) \quad , \quad -\pi \leq x \leq \pi \quad , \tag{2.38}$$

and the log likelihood function is

$$\log \mathcal{L} = \sum_{i=1}^{N} \log \sum_{j=1}^{K} \pi_j \sum_{l=-\infty}^{\infty} \mathcal{N}(x_i \,;\, \mu_j + 2\pi l, \sigma_j^2) \quad . \tag{2.39}$$

Algorithm 1 EM for fitting a mixture of wrapped Gaussian distributions

E step

$$\eta_{ijl} = \frac{\mathcal{N}\big(x_i\,;\widehat{\mu}_j{+}2\pi l,\widehat{\sigma}_j^2\big)\;\widehat{\pi}_j}{\sum_{j=1}^{K}\sum_{l=-\infty}^{\infty}\mathcal{N}\big(x_i\,;\widehat{\mu}_j{+}2\pi l,\widehat{\sigma}_j^2\big)\;\widehat{\pi}_j}$$

M step

$$\widehat{\mu}_j = \frac{\sum_{i=1}^{N}\sum_{l=-\infty}^{\infty}(x_i - 2\pi l)\;\eta_{ijl}}{\sum_{i=1}^{N}\sum_{l=-\infty}^{\infty}\eta_{ijl}}$$

$$\widehat{\sigma}_j^2 = \frac{\sum_{i=1}^{N}\sum_{l=-\infty}^{\infty}(x_i - \widehat{\mu}_j - 2\pi l)^2\;\eta_{ijl}}{\sum_{i=1}^{N}\sum_{l=-\infty}^{\infty}\eta_{ijl}}$$

$$\widehat{\pi}_j = \tfrac{1}{N}\sum_{i=1}^{N}\sum_{l=-\infty}^{\infty}\eta_{ijl}$$

The Q function is then given by

$$Q = \sum_{i=1}^{N}\sum_{j=1}^{K}\sum_{l=-\infty}^{\infty}\Big(\log\big[\pi_j\,\mathcal{N}(x_i\,;\,\mu_j + 2\pi l, \sigma_j^2)\big]\Big)\,\eta_{ijl} \tag{2.40}$$

$$= \sum_{i=1}^{N}\sum_{j=1}^{K}\sum_{l=-\infty}^{\infty}\left(\log(\pi_j) - \frac{1}{2}\log(2\pi) - \frac{1}{2}\log(\sigma_j^2) - \frac{(x_i - \mu_j - 2\pi l)^2}{2\sigma_j^2}\right)\eta_{ijl}\ , \tag{2.41}$$

where the posterior probabilities

$$\eta_{ijl} = P(z_{jl} \mid x_i\,;\,\mu_j, \sigma_j^2, \pi_j) \tag{2.42}$$

are constrained to sum to 1 for each data point:

$$\forall\, i \quad \sum_{j=1}^{K}\sum_{l=-\infty}^{\infty}\eta_{ijl} = 1 \tag{2.43}$$

Taking partial derivatives with respect to each parameter, setting the result equal to zero, and solving for the parameters, we get the M step update rules. The EM procedure is summarized in Algorithm 1. In practice, we cannot evaluate expressions with an infinite number of terms numerically, so the WGs need to be truncated after a sufficient number of terms. This involves replacing all $\sum_{l=-\infty}^{\infty}(-)$ with $\sum_{l=-L}^{L}(-)$.

2.3.3 Fitting a mixture of von Mises distributions

We will need an algorithm to update the parameters of a MovM [92], whose pdf is:

$$P(\theta\,;\,\boldsymbol{\mu},\boldsymbol{\kappa},\boldsymbol{\pi}) = \sum_{j=1}^{K}\pi_j\;vM(\theta\,;\,\mu_j,\kappa_j) \quad , \quad -\pi \le \theta \le \pi\ . \tag{2.44}$$

Algorithm 2 EM for fitting a mixture of von Mises distributions

E step

$$\eta_{ij} = \frac{vM\left(\theta_i\,;\widehat{\mu}_j,\widehat{\kappa}_j\right)\,\widehat{\pi}_j}{\sum\limits_{j=1}^{K} vM\left(\theta_i\,;\widehat{\mu}_j,\widehat{\kappa}_j\right)\,\widehat{\pi}_j}$$

M step

$$\widehat{\mu}_j = \tan^{-1}\left(\frac{\sum\limits_{i=1}^{N}\sin(\theta_i)\,\eta_{ij}}{\sum\limits_{i=1}^{N}\cos(\theta_i)\,\eta_{ij}}\right)$$

$$A\left(\widehat{\kappa}_j\right) = \frac{I_1(\widehat{\kappa}_j)}{I_0(\widehat{\kappa}_j)} = \frac{\sum\limits_{i=1}^{N}\cos(\theta_i-\widehat{\mu}_j)\,\eta_{ij}}{\sum\limits_{i=1}^{N}\eta_{ij}}$$

$$\widehat{\pi}_j = \frac{1}{N}\sum_{i=1}^{N}\eta_{ij}$$

Because of the $\cos(-)$ term in the vM pdf, we will have to numerically update the concentration parameter κ. The log likelihood is:

$$\log\mathcal{L} = \sum_{i=1}^{N}\log\sum_{j=1}^{K}\pi_j\;vM(\theta_i\,;\,\mu_j,\kappa_j)\quad, \tag{2.45}$$

and the Q function is:

$$Q = \sum_{i=1}^{N}\sum_{j=1}^{K}\log\left[\pi_j\;vM(\theta_i\,;\,\mu_j,\kappa_j)\right]\;\eta_{ij} \tag{2.46}$$

$$= \sum_{i=1}^{N}\sum_{j=1}^{K}\left[\log(\pi_j) - \log(2\pi) - \log(I_0(\kappa_j)) + \kappa_j\cos(\theta_i-\mu_j)\right]\;\eta_{ij}\quad, \tag{2.47}$$

where the posterior probabilities

$$\eta_{ij} = P(z_j \mid \theta_i\,;\,\mu_j,\kappa_j,\pi_j)\quad, \tag{2.48}$$

are constrained to sum to 1 for each data point:

$$\forall\, i \quad \sum_{j=1}^{K}\eta_{ij} = 1\quad. \tag{2.49}$$

Taking partial derivatives with respect to each parameter and setting the results to zero gives the M step update rules. The EM procedure is summarized in Algorithm 2. We can solve for κ_j with a zero-finder (e.g. bisection search [88]) using the old estimate as a starting point.

2.3.4 Fitting a mixture of von Mises-Fisher distributions

Details about the EM algorithm for fitting a MovMF are discussed at length in [93]. There is also a k-means algorithm for clustering on the sphere [94]. The pdf of a MovMF is given as:

$$P(\mathbf{x}\,;\,\boldsymbol{\mu},\boldsymbol{\kappa},\boldsymbol{\pi}) = \sum_{j=1}^{K} \pi_j \; vMF(\mathbf{x}\,;\,\boldsymbol{\mu}_j,\kappa_j) \quad , \quad \|\mathbf{x}\|_2 = 1 \;\; . \tag{2.50}$$

The log likelihood is:

$$\log \mathcal{L} = \sum_{i=1}^{N} \log \sum_{j=1}^{K} \pi_j \; vMF(\mathbf{x}_i\,;\,\boldsymbol{\mu}_j,\kappa_j) \;\; , \tag{2.51}$$

and the Q function is:

$$Q = \sum_{i=1}^{N} \sum_{j=1}^{K} \log \left[\pi_j \; vMF(\mathbf{x}_i\,;\,\boldsymbol{\mu}_j,\kappa_j) \right] \; \eta_{ij} \tag{2.52}$$

$$= \sum_{i=1}^{N} \sum_{j=1}^{K} \left[\log(\pi_j) + \log(\kappa_j) - \log(2\pi) - \log(e^{\kappa_j} - e^{-\kappa_j}) + \kappa_j \boldsymbol{\mu}_j^T \mathbf{x}_i \right] \; \eta_{ij} \;\; , \tag{2.53}$$

where the posterior probabilities

$$\eta_{ij} = P(z_j \mid \mathbf{x}_i\,;\,\boldsymbol{\mu},\boldsymbol{\kappa},\boldsymbol{\pi}) \;\; , \tag{2.54}$$

are constrained to sum to 1 for each data point:

$$\forall\, i \quad \sum_{j=1}^{K} \eta_{ij} = 1 \;\; . \tag{2.55}$$

Taking partial derivatives, setting the results to zero and solving for the new parameters gives the M step update rules. The EM procedure is summarized in Algorithm 3. The update of the concentration parameters can be numerically unstable, but there are good approximations. For $\kappa \gg 3$, we can drop the $e^{-\widehat{\kappa}_j}$ terms and use the following approximation [58]:

$$\widehat{\kappa}_j \approx \frac{1}{1 - A(\widehat{\kappa}_j)} \;\; . \tag{2.56}$$

Even when the conditions for this approximation are not met, we find empirically that spherical data can still be successfully clustered.

2.4 Interchannel phase difference features

We will use inter-channel phase differences (IPD) as a raw feature to perform multi-source separation and tracking. The IPD representation is modified from that of [1] so as to incorporate spatial aliasing explicitly in a statistical model. We show that these wrapped features compose a circular-linear dataset.

Algorithm 3 EM for fitting a mixture of von-Mises Fisher distributions

E Step

$$\eta_{ij} = \frac{vMF\left(\mathbf{x}_i\,;\widehat{\boldsymbol{\mu}}_j,\widehat{\kappa}_j\right)\widehat{\pi}_j}{\sum\limits_{j=1}^{K} vMF\left(\mathbf{x}_i\,;\widehat{\boldsymbol{\mu}}_j,\widehat{\kappa}_j\right)\widehat{\pi}_j}$$

M step

$$\widehat{\boldsymbol{\mu}}_j = \frac{\sum\limits_{i=1}^{N}\mathbf{x}_i\,\eta_{ij}}{\left\|\sum\limits_{i=1}^{N}\mathbf{x}_i\,\eta_{ij}\right\|_2}$$

$$A\left(\widehat{\kappa}_j\right) = \frac{e^{\widehat{\kappa}_j}+e^{-\widehat{\kappa}_j}}{e^{\widehat{\kappa}_j}-e^{-\widehat{\kappa}_j}} - \frac{1}{\widehat{\kappa}_j} = \frac{\left\|\sum\limits_{i=1}^{N}\mathbf{x}_i\,\eta_{ij}\right\|_2}{\sum\limits_{i=1}^{N}\eta_{ij}}$$

$$\widehat{\pi}_j = \frac{1}{N}\sum_{i=1}^{N}\eta_{ij}$$

2.4.1 Feature extraction

A microphone array with C channels captures C time-domain signals $\mathbf{x}_i$, $i = 1, \ldots, C$. These signals are converted to a time-frequency representation using the short-time Fourier transform (STFT) (see Section 2.1.1):

$$X^{(i)} = STFT(\mathbf{x}_i) \in \mathbb{C}^{D\times T} \quad , \tag{2.57}$$

where D denotes the coefficient index in the DFT corresponding to half the sampling rate. We ignore the second half of the DFT because it contains the same information as the first half. Since the Fourier transform is a linear operation, we have that the DFT coefficient at each time-frequency bin is approximately equal to the sum of the contributions from the sources. In the absence of reverberation, this gives

$$X_{f,t}^{(i)} = \sum_{j=1}^{K} S_{f,t}^{(j)} \cdot a_{ij}\, e^{-j\omega d_{ij}} \quad , \quad \omega = \frac{\pi f}{D} \quad , \tag{2.58}$$

where $S_{f,t}^{(j)}$ is the DFT coefficient of the j^{th} source, a_{ij} and d_{ij} are the attenuation and delay for the direct path between the i^{th} microphone and the j^{th} source, and ω is the digital frequency corresponding to the f^{th} frequency band.

We compute element-wise logratios to consolidate the STFT information across channels. For $C = K = 2$, we have

$$F_{f,t} = \log\left(\frac{X_{f,t}^{(1)}}{X_{f,t}^{(2)}}\right) = \log\left(\frac{S_{f,t}^{(1)}\cdot a_{11}e^{-j\omega d_{11}} + S_{f,t}^{(2)}\cdot a_{12}e^{-j\omega d_{12}}}{S_{f,t}^{(1)}\cdot a_{21}e^{-j\omega d_{21}} + S_{f,t}^{(2)}\cdot a_{22}e^{-j\omega d_{22}}}\right) \quad . \tag{2.59}$$

If we assume that the signals are approximately disjoint in the STFT (see Equation (2.8)), then we can simplify Equation (2.59) to the one-source case in each TF bin:

$$F_{f,t} \approx \log\left(\frac{S_{f,t}\cdot a_1 e^{-j\omega d_1}}{S_{f,t}\cdot a_2 e^{-j\omega d_2}}\right) = \log\left(\frac{a_1}{a_2}\right) - j\omega\left(d_1 - d_2\right) \quad . \tag{2.60}$$

The negative imaginary part of this logratio yields the IPD:

$$\delta_{f,t} \;=\; -\operatorname{Im}(F_{f,t}) = \omega\,(d_1 - d_2) = \angle X^{(2)}_{f,t} - \angle X^{(1)}_{f,t} \quad , \tag{2.61}$$

which is a (wrapped) linear function of frequency. For a fixed source position, we expect these features to lie on a wrapped line in a plot of frequency vs. phase difference:

$$\delta_{f,t} = \psi(\alpha\, f) \quad , \quad \alpha = \frac{\pi}{D}(d_1 - d_2) \quad , \tag{2.62}$$

where $\psi(-)$ is the wrap mapping in Equation (2.15). To make the dependence on frequency explicit, we can form the following IPD feature vector:

$$\boldsymbol{\delta}_{f,t} = \begin{bmatrix} \delta_{f,t} & f \end{bmatrix} \quad . \tag{2.63}$$

Now it is clear that a collection of these vectors composes a circular-linear dataset. The case of three or more sources ($K \geq 3$) is no different as long as the disjointness property (Equation (2.8)) holds for all source pairs. For three or more microphones ($C \geq 3$), the IPD feature vector contains $C - 1$ phase differences:

$$\boldsymbol{\delta}_{f,t} = \begin{bmatrix} \delta_{f,t}(1,2) & \cdots & \delta_{f,t}(1,C) & f \end{bmatrix} \quad , \tag{2.64}$$

where $\delta_{f,t}(1,i)$ is the phase difference calculated from the 1^{st} and i^{th} channels. This feature representation is similar to that of MENUET [15].

2.4.2 Effect of reverberation on IPD features

The impact of reverberation on IPD features depends primarily on the physical arrangement of the array, the sources, and other objects in the room. Consider a source signal $s[n]$. The recorded (reverby) signals are

$$x_1[n] = \sum_{r=1}^{R} a_1^r\, s[n - d_1^r] \quad , \quad x_2[n] = \sum_{r=1}^{R} a_2^r\, s[n - d_2^r] \quad , \tag{2.65}$$

where a_i^r and d_i^r are the attenuation and delay values for the r^{th} reflection at the i^{th} microphone. Now the IPD features are given as

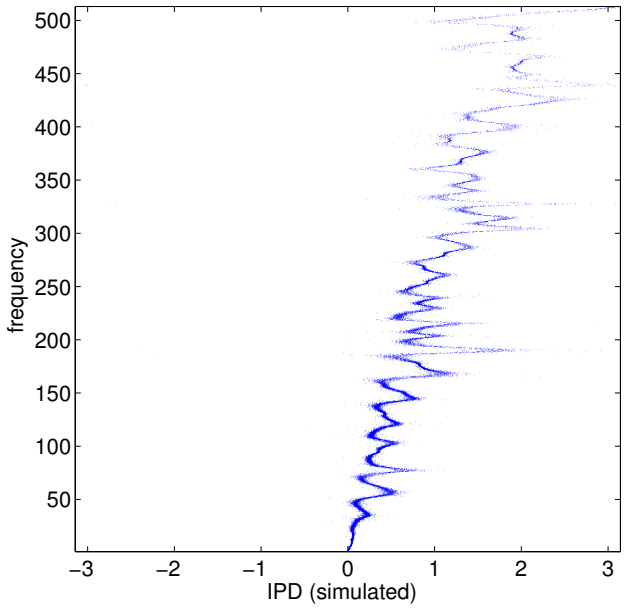


Figure 2.8: Moderately distorted IPD plot for a synthetic, echoic recording of speech. Reflections off of the walls in the (small) room cause sinusoid-like perturbations in the IPD line.

$$F = \angle STFT(\mathbf{x}_2) - \angle STFT(\mathbf{x}_1) \tag{2.66}$$

$$= \angle \left(\sum_{r=1}^{R} S(\omega) a_2^r e^{-j\omega d_2^r} \right) - \angle \left(\sum_{r=1}^{R} S(\omega) a_1^r e^{-j\omega d_1^r} \right) \tag{2.67}$$

$$= \angle \left(\sum_{r=1}^{R} S(\omega) a_2^r \left[\cos(-\omega d_2^r) + j \sin(-\omega d_2^r) \right] \right) - \angle \left(\sum_{r=1}^{R} S(\omega) a_1^r \left[\cos(-\omega d_1^r) + j \sin(-\omega d_1^r) \right] \right) \tag{2.68}$$

$$= \angle \left(\sum_{r=1}^{R} S(\omega) a_2^r \cos(\omega d_2^r) - j \sum_{r=1}^{R} S(\omega) a_2^r \sin(\omega d_2^r) \right) - \angle \left(\sum_{r=1}^{R} S(\omega) a_1^r \cos(\omega d_1^r) - j \sum_{r=1}^{R} S(\omega) a_1^r \sin(\omega d_1^r) \right) \tag{2.69}$$

$$= \tan^{-1} \left(\frac{-\sum_{r=1}^{R} S(\omega) a_2^r \sin(\omega d_2^r)}{\sum_{r=1}^{R} S(\omega) a_2^r \cos(\omega d_2^r)} \right) - \tan^{-1} \left(\frac{-\sum_{r=1}^{R} S(\omega) a_1^r \sin(\omega d_1^r)}{\sum_{r=1}^{R} S(\omega) a_1^r \cos(\omega d_1^r)} \right) \tag{2.70}$$

$$= \tan^{-1} \left(\frac{\sum_{r=1}^{R} a_1^r \sin(\omega d_1^r)}{\sum_{r=1}^{R} a_1^r \cos(\omega d_1^r)} \right) - \tan^{-1} \left(\frac{\sum_{r=1}^{R} a_2^r \sin(\omega d_2^r)}{\sum_{r=1}^{R} a_2^r \cos(\omega d_2^r)} \right) \tag{2.71}$$

Thus, reverb results in 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, that term dominates the argument of $\tan^{-1}(-)$, which approximately reduces

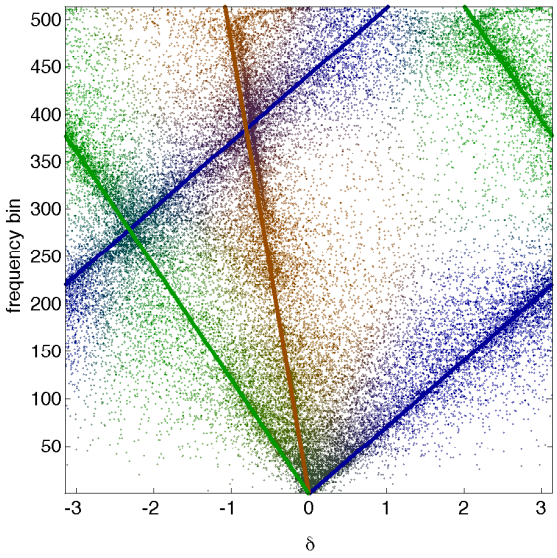


Figure 2.9: IPD plot for synthetic mixture of three sources, colored according to likelihood probability. (This figure appears in [20].)

to the case of no reverb, 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 non-linearity may be significant. Figure 2.8 depicts a moderate case.

2.5 Circular-linear regression

A circular-linear dataset is one that contains vectors with two values: one linear and one circular [72]. That is, one component describes a value that lies on the real line, $\mathbb{R}^1$, and the other describes a value that lies on the unit circle, $\mathbb{S}^1$ (generalizations of this allow for more than two components). Circular-linear regression is the problem of fitting a wrapped line[2] to such a dataset. This is nothing but linear regression modulo 2π in one of the variables.[3] Each IPD feature vector corresponds to a TF bin, so we can form a binary mask by partitioning the vectors by some clustering algorithm that fits multiple wrapped lines to the data. We can view this as a multimodal circular-linear regression problem. We will see that source separation and localization require only the slopes of the wrapped lines corresponding to the speakers.

2.5.1 IPDs as circular-linear data

An acoustic wavefront that arrives at a microphone array at an angle incurs a particular delay between the microphones. By the delay property of the Fourier transform, this corresponds to a phase shift in the frequency domain. More shift will exist at higher frequencies, resulting in data that lies along a wrapped line. When multiple speakers are present and they are disjoint in the STFT, we observe data that traces out multiple wrapped lines. An example of this for a synthetic, anechoic mixture of three sources captured by a two-microphone array is shown in Figure 2.9. To perform IPD-based BSS, we will cluster the feature vectors in Equation (2.63) and partition the mixture STFT accordingly. This is equivalent to the problem of multimodal circular-linear regression, namely, recovering the underlying wrapped linear models.

2.5.2 Probabilistic model for circular-linear regression

Consider the case of fitting a single wrapped line of the form in Equation (2.62) to an IPD dataset $\boldsymbol{\Delta} = \{\boldsymbol{\delta}_{f,t}\}$ derived from a stereo recording. We can measure the goodness-of-fit of a wrapped line with slope α by a likelihood criterion such as:

$$\mathcal{L}(\boldsymbol{\Delta}\,;\,\alpha) = \prod_{f=1}^{D}\prod_{t=1}^{T} P(\boldsymbol{\delta}_{f,t}\,;\,\psi(\alpha f),\kappa) \quad , \tag{2.72}$$

where the probability distribution is arbitrarily chosen to be vM for simplicity (we could also choose it to be WG). Extending this to the case of multiple lines, we have:

$$\mathcal{L}(\boldsymbol{\Delta}\,;\,\boldsymbol{\alpha}) = \prod_{f=1}^{D}\prod_{t=1}^{T}\sum_{j=1}^{K} P(\boldsymbol{\delta}_{f,t}\,;\,\psi(\alpha_j f),\kappa) \quad . \tag{2.73}$$

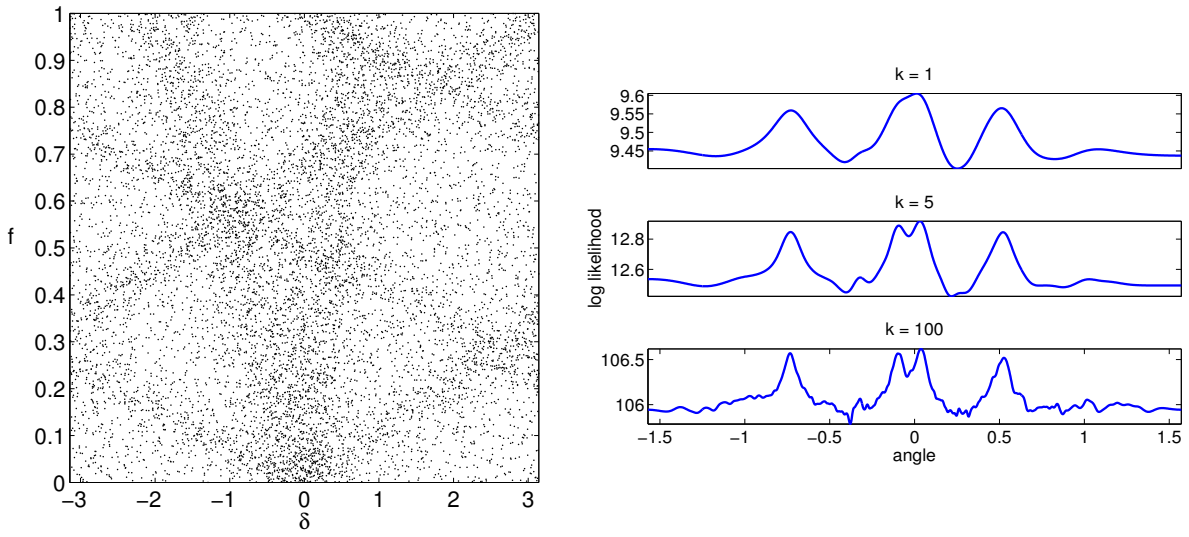
An example of multimodal circular-linear data generated from this model along with outliers sampled from the uniform distribution is shown in Figure 2.10(a). The corresponding likelihood functions for three values of the von Mises concentration parameter κ are shown in Figure 2.10(b). It is important to include the uniform noise as IPD features tend to look this way in practice. Roughly half of the data in Figure 2.10(a) can be considered outliers. Nevertheless, it is clear that peaks in the likelihood function correspond to the slopes of the wrapped lines. For this dataset, a higher κ is necessary to differentiate the two lines with slopes near 0.

2.6 Direction-of-arrival estimation

We will use 2-, 3-, and 4-microphone arrays to localize and track multiple speakers on the unit circle and unit sphere. Thus, we will need a method to estimate the directions-of-arrival (DOA) of the sound sources.

[2]This has also been called a "barber pole regression curve" in the directional statistics literature [72] because it can be visualized as a helix on the surface of a cylinder. A less intuitive but more general and correct interpretation would be that IPD features in a single frequency band lie on a torus. The wrapped IPD lines are then visualized as spirals on a disk for the case of 2 channels, where radius corresponds to frequency. Each concentric circle on the disk corresponds to a single frequency band. For the case of three channels, IPD features lie on a torus (i.e. a donut) whose thickness changes with frequency.

[3]The case without spatial aliasing was discussed in [95].



(a) (b)

Figure 2.10: (a) 10,000 data points showing several wrapped-line trends in the presence of outliers. (b) Log likelihood as a function of DOA. The concentration parameter κ of the von Mises distribution determines how strongly outliers are penalized.

Many methods exist for DOA estimation [9], [13], [22]. The most common technique is to search for peak(s) in the inter-channel cross-correlation function over a grid in DOA space [26], [27]. When implemented with a noise-shaping filter, this is called the generalized cross correlation (GCC) method [24], [25]. For localizing multiple sources simultaneously, the steered response power (SRP) of a beamformer can be calculated over DOA space. There are also well-known subspace methods like MUSIC [28] and ESPRIT [30] that use eigendecompositions to identify signal and noise subspaces of the channel correlation matrix. In MUSIC, a "pseudospectrum" is calculated that has peaks corresponding to the DOAs, and in ESPRIT, estimates are calculated by solving a total least squares problem. Unfortunately, MUSIC requires at least as many microphones as there are sources and ESPRIT requires at least twice as many.

In this thesis, we are concerned with estimating DOAs from inter-channel delays. The easiest approach is to solve a least-squares problem to recover a direction vector [23]. Another method involves trigonometric arguments. The only information we need for either is the relative positions of the microphones and an estimate of the inter-channel delays. We review both of these in this section.

It will be important to relate the slope α of a wrapped line calculated from IPD data to the DOA of a source. This can be done by first relating slopes to inter-channel delays. The slope α of a wrapped line is related to the delay e_{12} (*samples*) from microphone 1 to microphone 2 by

$$\alpha = -\frac{\pi}{D}\, e_{12} \quad . \tag{2.74}$$

2.6.1 Least-squares DOA estimation

One method to solve for the DOA of a sound source reduces to solving a system of linear equations. We can derive it by considering the physical arrangement of the microphones and the source. This is depicted

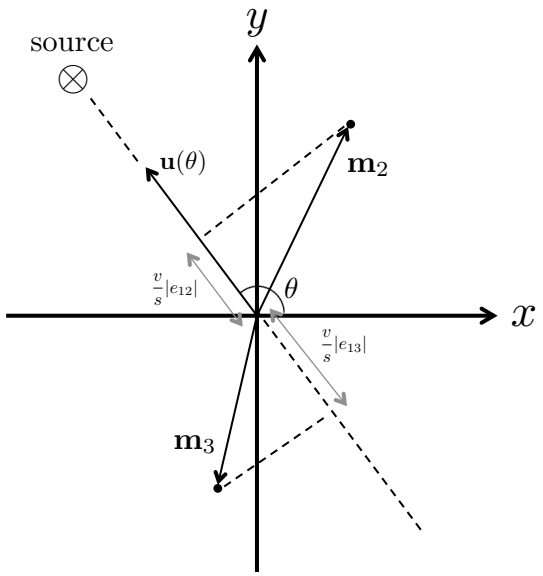


Figure 2.11: Geometry of least-squares DOA estimation. The first microphone is located at the origin.

in Figure 2.11 for a 3-mic array in 2D space. The position of the i^{th} microphone is

$$\mathbf{m}_i = \begin{bmatrix} m_{i1} & m_{i2} \end{bmatrix}^T \quad , \tag{2.75}$$

and the vector connecting the i^{th} and j^{th} microphones is

$$\mathbf{q}_{ij} = \mathbf{m}_i - \mathbf{m}_j \quad . \tag{2.76}$$

We denote a unit vector oriented in the direction θ as

$$\mathbf{u}(\theta) = \begin{bmatrix} u_1(\theta) & u_2(\theta) \end{bmatrix}^T \quad , \tag{2.77}$$

and note that an inner product relates this vector to the inter-channel delay e_{ij} for each mic pair:

$$\mathbf{q}_{ij}^T \mathbf{u}(\theta) = \frac{v}{s} e_{ij} \quad , \tag{2.78}$$

where v is the speed of sound ($meters\ sec^{-1}$) and s is the sampling rate ($samples\ sec^{-1}$). This defines a linear system of equations over all microphone pairs:

$$\begin{bmatrix} \mathbf{q}_{12} & \mathbf{q}_{13} & \mathbf{q}_{23} \end{bmatrix}^T \mathbf{u}(\theta) = \frac{v}{s} \mathbf{e} \quad , \tag{2.79}$$

where the vector of delays is

$$\mathbf{e} = \begin{bmatrix} e_{12} & e_{13} & e_{23} \end{bmatrix}^T \quad . \tag{2.80}$$

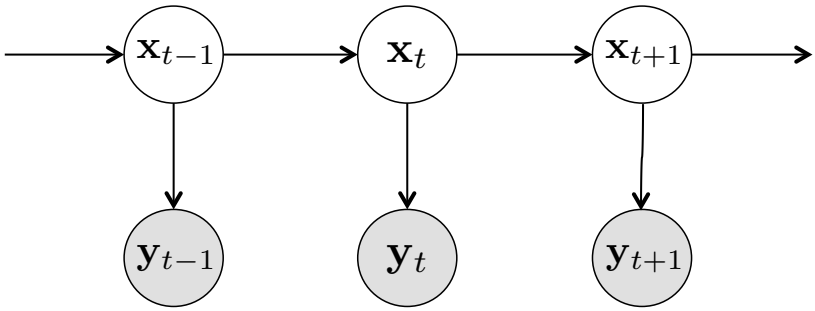


Figure 2.12: Graphical model of dynamic Bayesian network (DBN). The hidden state $\mathbf{x}_t$ evolves over time and emits an observed measurement $\mathbf{y}_t$. The goal of Bayesian filtering is to recursively infer the state from only the observation sequence $\mathbf{y}_{1:t}$ and knowledge of the system dynamics.

Solving this system gives the least-squares solution for the source direction.

2.6.2 Trigonometric DOA estimation

We can also derive DOA estimators through trigonometric arguments. We start with the case of 1D localization on a semicircle with 2 microphones. The angle of arrival θ can be related to the delay e_{12} and therefore to the slope α as follows:

$$\cos(\theta) = -\frac{v\,\Delta t}{d} = -\frac{v}{d\,s}\,e_{12} = \frac{v\,D}{\pi\,d\,s}\,\alpha \quad , \tag{2.81}$$

where v is the speed of sound ($meters\ sec^{-1}$), D is the maximum frequency bin value, d is the distance between the microphones ($meters$) and s is the sampling rate ($samples\ sec^{-1}$). We can also derive DOA estimators for localizing on the unit circle and hemisphere with 3 microphones and on the unit sphere with 4 microphones. These estimators are given in Appendix B.

2.7 Recursive Bayesian filtering

This section serves to summarize the Bayesian filtering framework for tracking the hidden state of a dynamical system. We will use wrapped filters to track speakers on the unit circle and sphere. The DOA tracking algorithms presented in this thesis require one or more observations (also called measurements) at each time step. We can use raw IPD features from Section 2.4 as measurements or we can transform these features into DOA estimates. This choice affects the system model. In this section, we review the basic approach to Bayesian filtering for tracking one or more sources.

2.7.1 Bayesian filtering equations

The Bayesian filtering equations [38] describe the recursive inference procedure for dynamic Bayesian networks (DBN) [96], [97]. We will derive them here. The graphical model of the DBN is shown in Figure 2.12. The (hidden) state $\mathbf{x}_t$ evolves according to a transition distribution in Equation (2.82) and the observation $\mathbf{y}_t$ is emitted according to a measurement distribution in Equation (2.83):

$$\mathbf{x}_t \sim P(\mathbf{x}_t \,|\, \mathbf{x}_{t-1}) \ , \tag{2.82}$$

$$\mathbf{y}_t \sim P(\mathbf{y}_t \,|\, \mathbf{x}_t) \ . \tag{2.83}$$

The system dynamics are fully described by these two equations. Equivalently, we may use the notation:

$$\mathbf{x}_t = f\left(\mathbf{x}_{t-1}, \mathbf{v}_t\right) \ , \tag{2.84}$$

$$\mathbf{y}_t = g\left(\mathbf{x}_t, \mathbf{w}_t\right) \ , \tag{2.85}$$

where $\mathbf{v}_t$ and $\mathbf{w}_t$ are process and measurement noise variables with known statistics and the (possibly nonlinear) functions $f(-)$ and $g(-)$ dictate how the state and measurement are generated at time t. This representation is often referred to as the state space model (SSM) of the system.

The filtered state distribution $P(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})$ evolves according to two repeated steps, *predict* and *correct*, that propagate it from time $t-1$ to time t. This density accumulates information about the entire observation sequence $\mathbf{y}_{1:t-1}$ without explicitly storing its values. In the *predict* step, the filtered distribution is propagated forward via (2.82):

$$P(\mathbf{x}_t|\mathbf{y}_{1:t-1}) = \int P(\mathbf{x}_t, \mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})\, d\mathbf{x}_{t-1} \tag{2.86}$$

$$= \int P(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{y}_{1:t-1}) P(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})\, d\mathbf{x}_{t-1} \tag{2.87}$$

$$= \int P(\mathbf{x}_t|\mathbf{x}_{t-1}) P(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})\, d\mathbf{x}_{t-1} \ . \tag{2.88}$$

The prediction is then updated in the *correct* step via Equation (2.83):

$$P(\mathbf{x}_t|\mathbf{y}_{1:t}) = P(\mathbf{x}_t|\mathbf{y}_t, \mathbf{y}_{1:t-1}) \tag{2.89}$$

$$= \frac{P(\mathbf{y}_t|\mathbf{x}_t, \mathbf{y}_{1:t-1}) P(\mathbf{x}_t|\mathbf{y}_{1:t-1})}{P(\mathbf{y}_t|\mathbf{y}_{1:t-1})} \tag{2.90}$$

$$= \frac{P(\mathbf{y}_t|\mathbf{x}_t) P(\mathbf{x}_t|\mathbf{y}_{1:t-1})}{P(\mathbf{y}_t|\mathbf{y}_{1:t-1})} \tag{2.91}$$

$$\propto P(\mathbf{y}_t|\mathbf{x}_t) P(\mathbf{x}_t|\mathbf{y}_{1:t-1}) \ . \tag{2.92}$$

The resulting algorithm depends on the filtered, state transition, and observation densities. The Kalman filter [33], [34] is optimal for a linear-Gaussian DBN. However, we will work with the WG, vM, vMF, and mixtures of each one. Thus, the goal is to derive filters that approximate Equations (2.88) and (2.92) as closely as possible.

Algorithm 4 Kalman filter

Predict

$\widehat{\mathbf{x}}_t^- = A\widehat{\mathbf{x}}_{t-1}$

$\widehat{\Sigma}_t^- = A\widehat{\Sigma}_{t-1}A^T + \Sigma_v$

Correct

$K_t = \frac{\widehat{\Sigma}_t^- B^T}{B\widehat{\Sigma}_t^- B^T + \Sigma_w}$

$\widehat{\mathbf{x}}_t = \widehat{\mathbf{x}}_t^- + K_t\left(\mathbf{y}_t - B\widehat{\mathbf{x}}_t^-\right)$

$\widehat{\Sigma}_t = \left(\mathbf{I} - K_t B\right)\widehat{\Sigma}_t^-$

2.7.2 Kalman filter

If the state space model (SSM) is linear-Gaussian:

$$\mathbf{x}_t = A\mathbf{x}_{t-1} + \mathbf{v}_t \quad , \quad \mathbf{v}_t \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_v\right) \quad , \tag{2.93}$$

$$\mathbf{y}_t = B\mathbf{x}_t + \mathbf{w}_t \quad , \quad \mathbf{w}_t \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_w\right) \quad , \tag{2.94}$$

where A and B are real-valued matrices, then the Bayesian filtering equations (2.88) and (2.92) can be solved in closed form to yield the well-known Kalman filter [34]. The filtered state distribution $P(\mathbf{x}_t|\mathbf{y}_{1:t})$ remains Gaussian from one time step to the next, so we need only estimate its mean $\widehat{\mathbf{x}}_t$ and covariance $\widehat{\Sigma}_t$. The filter equations can take various forms, one of which is given in Algorithm 4. The innovation term, $\mathbf{y}_t - B\widehat{\mathbf{x}}_t^-$, is important because it captures how well the anticipated observation, $B\widehat{\mathbf{x}}_t^-$, matches the true observation, $\mathbf{y}_t$. It also serves to inform the Kalman filter on how to adapt the state estimate to the new measurement.

Approximate Kalman filtering

Filtering schemes for a SSM that is not linear-Gaussian often involve a linear-Gaussian approximation of the SSM. Although the Kalman filter is applicable to the new system, one must take care to account for the inaccuracies introduced. The extended Kalman filter (EKF) [34] uses a first-order Taylor series expansion to linearize non-linearities in the system dynamics. The unscented Kalman filter (UKF) [36], in contrast, approximates the state distribution as a Gaussian and explicitly models the non-linearities via the unscented transform [37]. A number of deterministically-chosen "sigma points" are calculated that capture the mean and covariance of the filtered distribution. They are fed through the SSM to give transformed points that are used to construct a Gaussian approximation of the filtered distribution in the next time step. It can be shown that the UKF achieves a higher-order approximation and tends not to diverge in the presence of strong non-linearities where the EKF does.

2.7.3 Particle filter

There are cases where the Bayesian filter equations cannot be approximated in some convenient closed form. When all else fails, we can always attempt to represent the state distribution with a weighted set of

particles. These particles are propagated forward through time via the Bayesian filtering equations, which involves sampling from the state transition distribution in Equation (2.82). If this distribution is difficult to sample from, we can apply importance sampling techniques [87], [97].

Importance sampling

Suppose we would like to draw independent samples from a target distribution $P(\mathbf{x})$, but we cannot do so directly. We can side-step the issue by gathering samples, $\mathbf{x}^{(l)}, l = 1, \ldots, L$, from a suitably-chosen proposal distribution $Q(\mathbf{x})$ and assigning them weights:

$$w^{(l)} \propto \frac{P(\mathbf{x}^{(l)})}{Q(\mathbf{x}^{(l)})} \quad . \tag{2.95}$$

The importance weights compensate for any mismatch between $P(\mathbf{x})$ and $Q(\mathbf{x})$. We can assume that $P(\mathbf{x})$ (and even $Q(\mathbf{x})$) is known only up to a normalization constant. A well-matched proposal distribution will ensure that the weights are spread near $1/L$. If $Q(\mathbf{x})$ is poorly-matched, many of the weights will be nearly zero, giving bad results or an inefficient approximation scheme.

Sampling importance resampling

The samples $\mathbf{x}^{(l)} \sim Q(\mathbf{x})$ and their weights $w^{(l)}$ approximate the distribution $P(\mathbf{x})$ as a weighted set of point estimates. If we would like an unweighted sample set, we can resample by treating the weights vector $\mathbf{w}$ as a categorical distribution. In other words, the new unweighted sample set consists of L i.i.d. draws from the weighted approximation of $P(\mathbf{x})$:

$$\mathbf{x} \sim \sum_{l=1}^{L} w^{(l)} \, \delta\left(\mathbf{x} - \mathbf{x}^{(l)}\right) \quad . \tag{2.96}$$

One potential drawback is that heavily-weighted samples will be repeated, having been drawn with replacement.

Sequential importance resampling (SIR)

The most straightforward technique for particle filtering is a repeated application of sampling importance resampling to approximate the recursive scheme from Section 2.7.1. This is alternatively called sequential importance resampling (SIR), bootstrap filtering, the condensation algorithm, and survival of the fittest.

Instead of keeping track of the entire state distribution in each time step, we approximate it with a set of weighted Dirac deltas, referred to as particles. The filtered state distribution at time $t-1$ is approximated with a particle set

$$\mathbf{X}_{t-1} = \left\{\mathbf{x}_{t-1}^{(l)}, w_{t-1}^{(l)}\right\} \quad , \quad l = 1, \ldots, L \ , \tag{2.97}$$

by:

Algorithm 5 Particle filter - sequential importance resampling

Predict

$\mathbf{x}_t^{(l),-} \sim P\left(\mathbf{x}_t \middle| \mathbf{x}_{t-1}^{(l)}\right)$

Correct

$w_t^{(l),-} \propto P\left(\mathbf{y}_t \middle| \mathbf{x}_t^{(l),-}\right)\ w_{t-1}^{(l)}$

Resample

$\mathbf{x}_t^{(l)} \sim \sum_{m=1}^{L} w_t^{(m),-}\ \delta\left(\mathbf{x}_t - \mathbf{x}_t^{(m),-}\right)$

$w_t^{(l)} = \frac{1}{L}$

$$P(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}) \approx \sum_{l=1}^{L} w_{t-1}^{(l)}\ \delta\left(\mathbf{x}_{t-1} - \mathbf{x}_{t-1}^{(l)}\right) \quad . \tag{2.98}$$

When we propagate this particle set through the SSM in Equations (2.82) and (2.83) to the next time step, we will have an updated particle set

$$\mathbf{X}_t = \left\{\mathbf{x}_t^{(l)}, w_t^{(l)}\right\} \quad , \quad l = 1, \ldots, L \ , \tag{2.99}$$

that approximates the filtered distribution at time t:

$$P(\mathbf{x}_t|\mathbf{y}_{1:t}) \approx \sum_{l=1}^{L} w_t^{(l)}\ \delta\left(\mathbf{x}_t - \mathbf{x}_t^{(l)}\right) \quad . \tag{2.100}$$

The set $\mathbf{X}_t$ is determined by combining Equation (2.98) and the Bayesian filtering Equations (2.88) and (2.92):

$$P(\mathbf{x}_t|\mathbf{y}_{1:t}) \propto P(\mathbf{y}_t|\mathbf{x}_t) \sum_{l=1}^{L} w_{t-1}^{(l)}\ \delta\left(\mathbf{x}_t - \mathbf{x}_t^{(l),-}\right) \tag{2.101}$$

$$\propto \sum_{l=1}^{L} P\left(\mathbf{y}_t \middle| \mathbf{x}_t^{(l),-}\right) w_{t-1}^{(l)}\ \delta\left(\mathbf{x}_t - \mathbf{x}_t^{(l),-}\right) \quad , \tag{2.102}$$

where the predicted particles are sampled from the transition distribution:

$$\mathbf{x}_t^{(l),-} \sim P\left(\mathbf{x}_t \middle| \mathbf{x}_{t-1}^{(l)}\right) \quad . \tag{2.103}$$

Comparing Equations (2.100) and (2.102), we can say that the updated weights should be:

$$w_t^{(l),-} \propto P\left(\mathbf{y}_t \middle| \mathbf{x}_t^{(l),-}\right)\ w_{t-1}^{(l)} \ . \tag{2.104}$$

The last step is to resample from the updated particle set to prevent degeneracy:

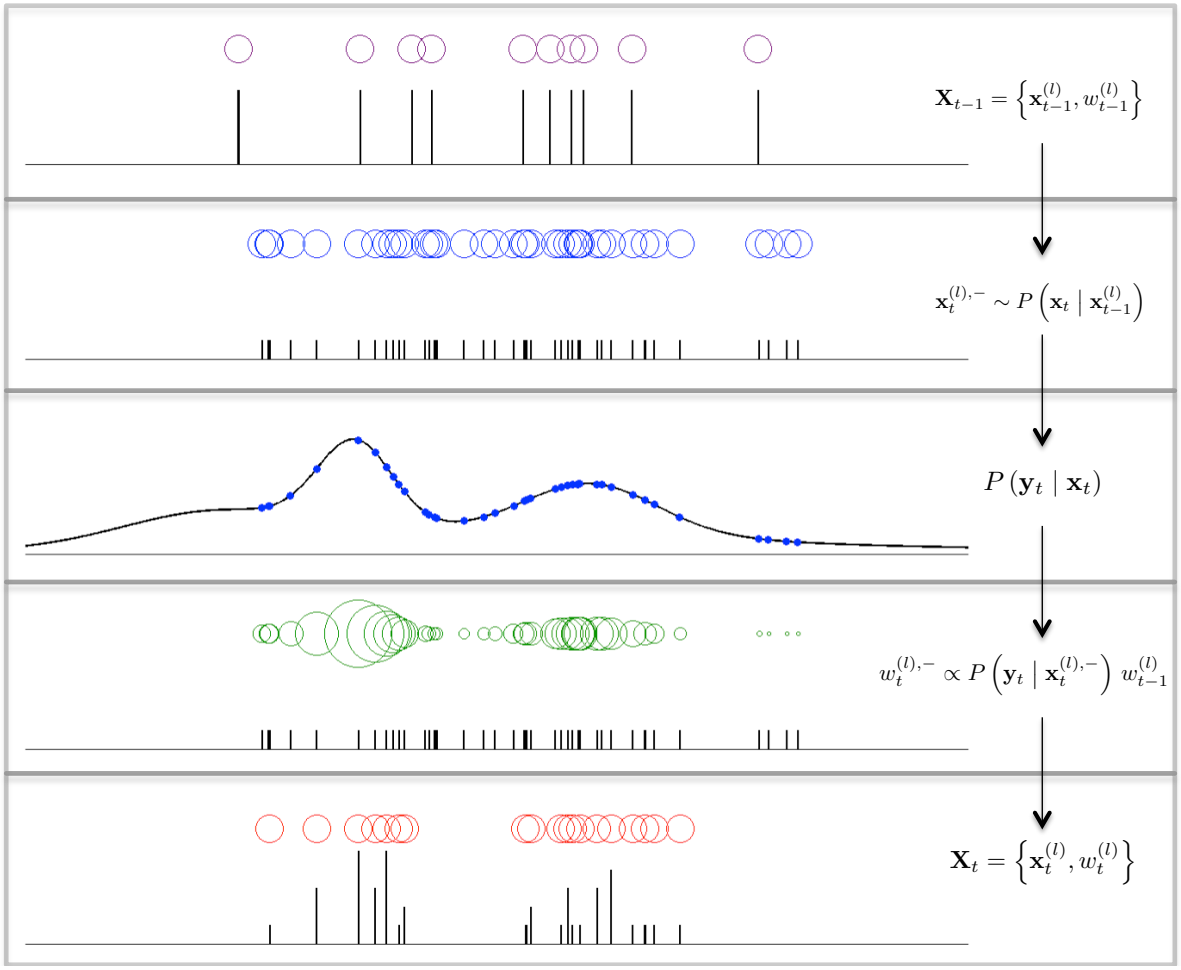


Figure 2.13: Sequential importance resampling for particle filtering. Circles represent particles (size indicates weight) and bars indicate the number of copies. (Top) Particle set at time $t-1$ with four copies of each. (Top middle) *Predict*: propagate via transition distribution. (Middle) Meausurement likelihood. (Bottom middle) *Correct*: re-weight via likelihood. (Bottom) *Resample*: sample i.i.d. from updated set.

$$\mathbf{x}_t^{(l)} \sim \sum_{m=1}^{L} w_t^{(m),-} \, \delta\left(\mathbf{x}_t - \mathbf{x}_t^{(m),-}\right) \tag{2.105}$$

$$w_t^{(l)} = \frac{1}{L} \ . \tag{2.106}$$

The SIR procedure is depicted in Figure 2.13 and is summarized in Algorithm 5. To recap, we started at time $t-1$ with a set of weighted particles and fed this through our SSM to get a sum-of-deltas approximation of the posterior distribution at time t. Without resampling, the weights quickly degenerate (i.e. within a few iterations) such that a single particle is left to represent the entire state distribution. In fact, it can be shown that the variance of the weights increases exponentially with time [97]. Although many resampling techniques exist including stratified sampling, systematic resampling, and Markov chain Monte Carlo methods, we will

use the multinomial resampling method described above.

SIR in context

SIR derives from a general formulation in which the new particles are sampled from a proposal distribution:

$$\mathbf{x}_t^{(l)} \sim Q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{y}_t) \ , \tag{2.107}$$

and the updated weights are given via importance sampling as:

$$w_t^{(l)} \propto w_{t-1}^{(l)} \ \frac{P\left(\mathbf{x}_t^{(l)}|\mathbf{x}_{t-1}^{(l)}\right) P\left(\mathbf{y}_t|\mathbf{x}_t^{(l)}\right)}{Q\left(\mathbf{x}_t^{(l)}|\mathbf{x}_{t-1}^{(l)}, \mathbf{y}_t\right)} \ . \tag{2.108}$$

If we assume that the proposal is equal to the state transition distribution:

$$Q\left(\mathbf{x}_t^{(l)}|\mathbf{x}_{t-1}^{(l)}, \mathbf{y}_t\right) = P\left(\mathbf{x}_t^{(l)}|\mathbf{x}_{t-1}^{(l)}\right) \ , \tag{2.109}$$

then the weight update reduces to that of the SIR algorithm. More clever proposals can improve the accuracy of a particle filter when the state transition distribution and observation likelihood are very different. However, we will only use the popular SIR algorithm in this thesis as it is conceptually simple, easy to implement, and works quite well in practice.

Estimating statistics from a particle set

Statistics of the state distribution can be calculated from the particle set. This is best done before resampling as it gives estimates with lower variance. For example, if the state distribution can be reasonably approximated as a Gaussian, its mean and covariance can be estimated as:

$$\widehat{\mathbf{x}}_t = \sum_{l=1}^{L} w_t^{(l)} \ \widehat{\mathbf{x}}_t^{(l)} \ , \tag{2.110}$$

$$\widehat{\Sigma}_t = \sum_{l=1}^{L} w_t^{(l)} \left(\widehat{\mathbf{x}}_t^{(l)} - \widehat{\mathbf{x}}_t\right)\left(\widehat{\mathbf{x}}_t^{(l)} - \widehat{\mathbf{x}}_t\right)^T \ . \tag{2.111}$$

These may not always be appropriate statistics to evaluate. For example, if the SSM describes the evolution of a system on the unit circle, we should calculate circular statistics instead.

2.7.4 Multi-source tracking with mixture models

In this thesis, we will derive algorithms for tracking K simultaneously active sources. We choose the state space model to be a factorial DBN where multiple state chains $\mathbf{x}_{t,j}$, $j = 1, \ldots, K$ evolve independently, each of which produces an observation $\mathbf{y}_{t,j}$. However, we observe the *unordered* set of measurements $\mathbf{y}_t = \{\mathbf{y}_{t,m}\}$, $m = 1, \ldots, K$:

$$\forall\, j \quad \mathbf{x}_{t,j} \sim P\left(\mathbf{x}_{t,j} \,|\, \mathbf{x}_{t-1,j}\right) \tag{2.112}$$

$$\forall\, j \quad \mathbf{y}_{t,j} \sim P\left(\mathbf{y}_{t,j} \,|\, \mathbf{x}_{t,j}\right) \tag{2.113}$$

$$\mathbf{y}_t = \{\mathbf{y}_{t,m}\} \quad . \tag{2.114}$$

A data association ambiguity exists because we do not know what observation was generated by what source (i.e. the bipartite matching between j and m indices is hidden). Exact inference requires an exponentially-growing model for the filtered state distribution. A well-known approximation is the switching Kalman filter (SKF) [98], which collapses the filtered distribution in each time step such that its size remains fixed.

We will model $P(\mathbf{x}_t|\mathbf{y}_{1:t})$ as a mixture and consider probabilistic assignments of observations/particles to mixture components. These are incorporated as weights in the correct step of the filter, effectively integrating out the unknown assignments. The reader is referred to the extensive literature on filtering with mixtures [46], [47], [48] [66].

2.8 RANdom SAmple Consensus (RANSAC)

RANSAC is a hugely important method in the computer vision community for estimating a simple model from a dataset with a large proportion of outliers [73]. If the model can be fully described by a small set of points, then one simply needs to find such a set in the data to recover the parameters of the model. If we wish to fit a line in $\mathbb{R}^2$, then only two points must be sampled. Each draw of a data pair is known as a "RANSAC sample" and the line connecting them is known as a "candidate." The number of inliers of each candidate is calculated and the one with the most inliers is chosen as an estimate of the true model. Other simple models such as circles and ellipses can also be found with RANSAC.

To ensure a good fit, a sufficiently high number M of candidates is collected. This is given by the expected number of trials $\mathrm{E}[t]$ until an inlier is chosen. If the proportion of inliers in the dataset is p and we need to sample n data points to fit a model, it can be shown that $\mathrm{E}[t] = p^{-n}$. In practice, M is overestimated to ensure a good fit. A final detail is that we must specify an inlier criterion (typically a thresholded distance function). The overall procedure is summarized in Algorithm 6.

RANSAC has a marked advantage for line-fitting in the presence of outliers compared to standard linear regression by least-squares. Consider the dataset in Figure 2.14 in which half of the data are sampled uniformly at random and half are sampled from a line with Gaussian-distributed error. Standard linear regression is thrown off by the outliers, whereas RANSAC completely ignores them. This is because a candidate line not aligned with the inlier set receives far fewer votes than one that is.

In this thesis, we are interested in fitting a wrapped line that passes through the origin in IPD space. Thus, we only need one point to fully specify the model. Because of this, a RANSAC-based approach will be very fast and robust to the effects of unknown interference and reverberation.

Algorithm 6 RANSAC

Inputs: $\mathbf{X} = \{\mathbf{x}_i\}$: N data points

Outputs: $\widehat{\boldsymbol{\alpha}}$: parameter estimates

$\mathbf{Y} = M$ RANSAC samples drawn uniformly at random from $\mathbf{X}$

$\mathbf{I} = \mathbf{0}^{N \times M}$

for $m = 1 : M$ **do**

Fit model with parameters $\boldsymbol{\alpha}_m$ to $\mathbf{Y}_m$

$\mathbf{I}(i, m) = 1$, $\forall i$ s.t. $\mathbf{x}_i$ is inlier of m^{th} model

end for

$\widehat{m} = \underset{m}{\operatorname{argmax}} \sum_i \mathbf{I}(i, m)$

return $\boldsymbol{\alpha}_{\widehat{m}}$

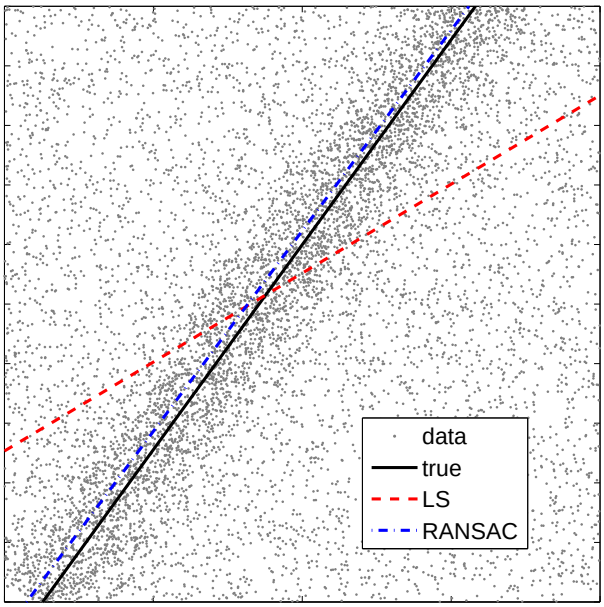


Figure 2.14: Line-fitting in uniform noise with least-squares regression (LS) and RANSAC. Half the data consists of outliers chosen uniformly at random in the square.

CHAPTER 3

BSS AND DOA ESTIMATION FOR MULTIPLE STATIONARY SPEAKERS

This chapter discusses algorithms for blind source separation (BSS) and direction-of-arrival (DOA) estimation of multiple, physically stationary speakers using wrapped IPD features. The basic idea is to cluster the features (see Section 2.4) according to a probabilistic circular-linear model. This takes spatial aliasing into account and does away with a permutation ambiguity across frequencies [4].[1] Two clustering methods are proposed: one uses EM and the other uses a sequential variant of RANSAC [73]. The latter is shown to be more efficient and accurate. It is also shown that the results of the clustering provide an estimator of the DOAs of the speakers. We will use this fact in Chapter 4 to track moving speakers over time.

3.1 EM for fitting a mixture of wrapped lines

The source separation problem can be reduced to one of multimodal circular-linear regression (see Section 2.5). It is instructive to view this as a parameter estimation problem with latent variables. Then, we can apply the Expectation-Maximization (EM) framework (see Section 2.3.1). The observed variables are the IPD data, the hidden variables are the TF bin labels (i.e. which source is active), and the unknown parameters are the IPD line slopes (or, equivalently, the source DOAs). The well-known graphical model for this sort of clustering problem is shown in Figure 3.1. We first consider the approach in a single frequency band and then extend this to combine information across frequencies.

3.1.1 Clustering in each frequency band individually

Consider the IPD clustering problem in a single frequency band. The observation model consists of a mixture of wrapped distributions on the interval $[-\pi, \pi]$, one for each source. The generative process involves choosing one component from the mixture with some probability and sampling a data point from that distribution. This is repeated until N i.i.d. samples are collected. Having observed this data, we would like to discover the parameters of the underlying mixture.

We can model each source distribution as a wrapped Gaussian (WG) and run the EM algorithm to fit a MoWG (see Section 2.3). The posterior probabilities from EM indicate how to construct a binary mask. However, clustering in each frequency band individually fails to capture the wrapped linear structure of the IPD lines. This results in a permutation ambiguity [100] where it is unclear how to group the clusters

[1] A related approach called Model-based Expectation-maximization Source Separation and Localization (MESSL) incorporates IPDs and inter-channel level differences (ILD) [99]. However, in real-world scenarios with compact arrays, ILD features typically provide little to no information and may significantly reduce performance.

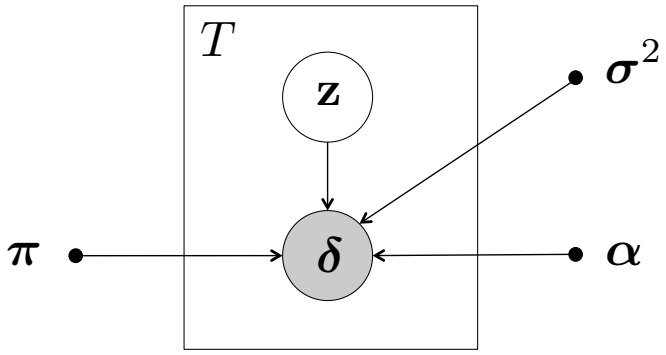


Figure 3.1: Graphical model for wrapped-line fitting. Observed IPD vectors y are generated by sampling from a mixture of wrapped lines with parameters $\{\boldsymbol{\alpha}, \boldsymbol{\sigma}^2, \boldsymbol{\pi}\}$. Hidden variables z associate each TF bin with a line.

across frequencies such that each group contains the WGs corresponding to a single source (this is a famous problem for ICA-based BSS [4]). In the next section, we modify this approach to perform the clustering jointly across all frequencies.

3.1.2 Clustering across frequencies

To incorporate the linear trend of IPD data across frequency, we reparameterize the model from the previous section in terms of the slopes α_j:

$$\mu_{jf} = \alpha_j f \quad . \tag{3.1}$$

This locks the means together for each line, resulting in what appears to be a multivariate WG. However, the pdf this distribution does *not* factorize into a product of densities, but is modeled as a sum of univariate WGs. This is because the frequency bands are completely separated in the generative model. We call this separated, wrapped, and mean-locked distribution a *multi-band wrapped Gaussian* (MWG).

The EM algorithm to learn a mixture of MWGs (MoMWG) proceeds as follows. We have a dataset of T vectors $\boldsymbol{\Delta} = \{\boldsymbol{\delta}_t\}$, $\boldsymbol{\delta}_t \in \mathbb{S}^{T\times 1}$, sampled i.i.d. from a mixture of multi-band wrapped Gaussians. The pdf of this distribution is

$$P(\boldsymbol{\delta}\,;\,\boldsymbol{\alpha}, \boldsymbol{\sigma}^2, \boldsymbol{\pi}) = \sum_{j=1}^{K} \pi_j \sum_{f=1}^{D} \frac{1}{D} \sum_{l=-\infty}^{\infty} \mathcal{N}(\delta_f\,;\,\alpha_j f + 2\pi l\,,\,\sigma_{jf}^2) \quad , \tag{3.2}$$

the log likelihood function for parameter estimation is

$$\log\,\mathcal{L} \propto \sum_{t=1}^{T} \log \sum_{j=1}^{K} \sum_{f=1}^{D} \sum_{l=-\infty}^{\infty} \pi_j\, \mathcal{N}\left(\delta_{f,t}\,;\,\alpha_j f + 2\pi l\,,\,\sigma_{jf}^2\right) \quad , \tag{3.3}$$

and the Q function is

Algorithm 7 EM for fitting a mixture of multi-band wrapped Gaussian distributions

E step

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

M step

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

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

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

$$Q = \sum_{t=1}^{T}\sum_{j=1}^{K}\sum_{f=1}^{D}\sum_{l=-\infty}^{\infty}\left(\log\left[\pi_j\,\mathcal{N}(\delta_{f,t}\,;\,\alpha_j f+2\pi l\,,\,\sigma^2_{jf})\right]\right)\eta_{tjfl} \tag{3.4}$$

$$= \sum_{t=1}^{T}\sum_{j=1}^{K}\sum_{f=1}^{D}\sum_{l=-\infty}^{\infty}\left(\log(\pi_j)-\frac{1}{2}\log(2\pi)-\frac{1}{2}\log(\sigma^2_{jf})-\frac{(\delta_{f,t}-\alpha_j f-2\pi l)^2}{2\sigma^2_{jf}}\right)\eta_{tjfl}\ . \tag{3.5}$$

The posterior probabilities

$$\eta_{tjfl} = P(z_{j,l}\,|\,\delta_{f,t}\,;\,\alpha_j,\sigma^2_{jf},\pi_j)\ , \tag{3.6}$$

are constrained to sum to 1 for each (multivariate) data point:

$$\forall\, t \quad \sum_{j=1}^{K}\sum_{f=1}^{D}\sum_{l=-\infty}^{\infty}\eta_{tjfl} = 1\ . \tag{3.7}$$

A standard derivation leads to the EM equations summarized in Algorithm 7. To avoid over-fitting, we can further constrain the model to have one variance parameter for each band or for the entire mixture (i.e. a scalar). This simply adds summations to the numerator and denominator of the variance update rule.

3.1.3 Drawbacks of EM

The issues with the EM algorithm for this problem are three-fold. First, this approach is slow. A 4-D array of posterior probabilities must be calculated in each E step. Second, this method is complicated relative to the inherent simplicity of the problem. The main complication arises from the wrapping of the linear models. However, this intuitively should not cause a large increase in algorithm complexity. And third, the wrapping and noisy nature of IPD data leads to the existence of many local maxima in the log likelihood.

Algorithm 8 Sequential RANSAC for fitting multiple wrapped lines

Inputs: $\boldsymbol{\Delta} = \{\boldsymbol{\delta}_i\} : N$ IPD data points

K : number of wrapped lines to fit

Outputs: $\widehat{\boldsymbol{\alpha}} = \{\widehat{\alpha}_j\} : K$ slopes

$\mathbf{Y} = M$ samples from $\boldsymbol{\Delta}$ selected uniformly at random

$\mathbf{I} = \mathbf{0}^{N \times M}$

for $m = 1 : M$ **do**

Fit line with slope α_m to Y_m

$\mathbf{I}(i, m) = 1$, $\forall i$ s.t. $\boldsymbol{\delta}_i$ is inlier of line with slope α_m

end for

$\widehat{\boldsymbol{\alpha}} = \{\}$

$A = \{1, \ldots, N\}$

for $j = 1 : K$ **do**

$\widehat{m} = \underset{m}{\operatorname{argmax}} \sum_{i \in A} \mathbf{I}(i, m)$

$\widehat{\boldsymbol{\alpha}} = \widehat{\boldsymbol{\alpha}} \cup \alpha_{\widehat{m}}$

$A = A \setminus \{i : \mathbf{I}(i, \widehat{m}) = 1\}$

end for

return $\widehat{\boldsymbol{\alpha}}$

Since EM performs a local optimization, it may converge to a solution that does not correspond to the true source DOAs.

However, this approach has the advantage of explicitly modeling the wrapped nature of the data. We can expect a good model to be fit as long as the initial conditions are sufficiently close to the correct solution. In the next section, we present a much faster heuristic method to perform the clustering.

3.2 Circular-linear regression by random sampling

We describe a fast method for clustering IPD data based on the RANdom SAmple Consensus (RANSAC) algorithm. RANSAC [73] was introduced in Section 2.8. We summarize a sequential approach [20] to fit multiple wrapped lines to a circular-linear dataset.

3.2.1 Sequential RANSAC

Multi-model variants of RANSAC have been proposed for stereo imaging applications [101], [102], [103]. As discussed in [20], we can apply a sequential approach to cluster IPD data efficiently and robustly. The generative model involves sampling i.i.d. from a mixture of multi-band wrapped distributions. We arbitrarily choose the von Mises distribution in this context for its simplicity. The procedure is summarized in Algorithm 8, where the data is indexed by i rather than the pair $\{f, t\}$ for clarity. M is scaled proportionally with the number of sources K.

Consider the following example. Figure 3.2(a) shows the IPD data for an anechoic mixture of two TSP

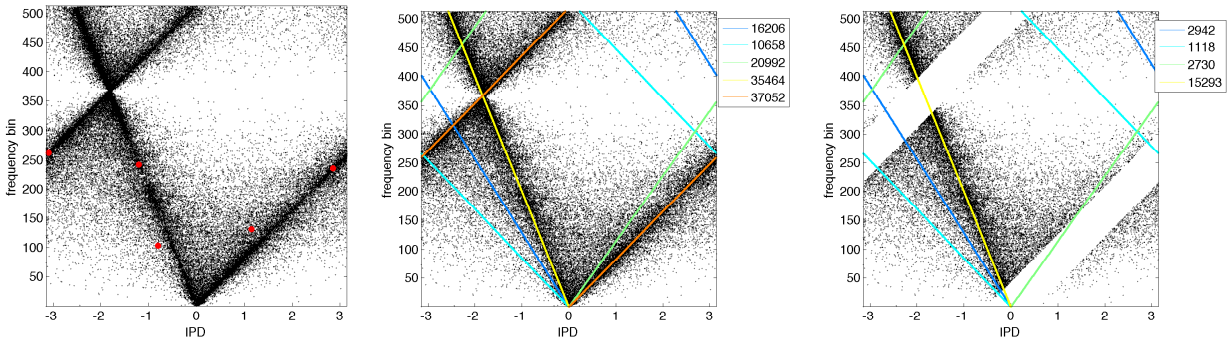


Figure 3.2: Example of sequential RANSAC for wrapped line-fitting. (a) IPD data with 5 RANSAC samples overlaid. (b) First iteration of sequential RANSAC showing candidate wrapped lines and their inlier counts. (c) Second iteration of sequential RANSAC after removal of the inliers of the first model.

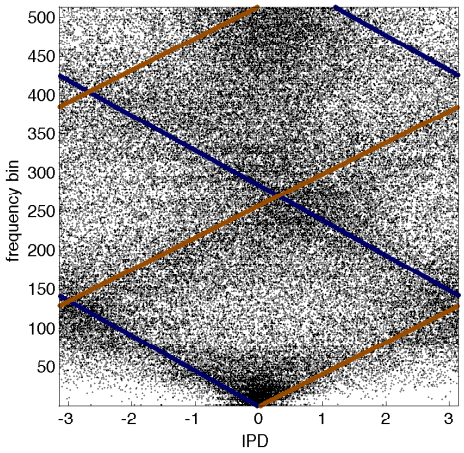


Figure 3.3: IPD plot of highly reverberant ($T_{60} = 1.5$ seconds), 2-speaker stairwell recording with wrapped lines fit by sequential RANSAC overlaid. Sequential RANSAC succeeds despite the fact that 65% of the data consists of outliers. (This figure appears in [20].)

speakers. Five RANSAC samples have been chosen uniformly at random. Figure 3.2(b) shows the corresponding wrapped line candidates and their inlier counts. The orange line is chosen and removed along with its inliers. This process is repeated to find the next best candidate, the yellow line, as shown in Figure 3.2(c).

3.2.2 Why sequential RANSAC works

We have found that sequential RANSAC as applied to the IPD line fitting problem works very well for a wide range of conditions. In [20], the example was given of a stereo recording in a stairwell whose T_{60} reverberation time was 1.5 seconds. The IPD plot for this recording is shown in Figure 3.3. Roughly 65% of the data consists of outliers, yet the line-fitting is still successful. We can understand this by considering the original probabilistic model for circular-linear regression presented in Section 2.5.2.

When RANSAC samples are drawn from the dataset $\mathbf{\Delta}$, they are effectively sampled from the likelihood

function shown in Figure 2.10(b).[2] We expect to draw more candidates from high-density regions in the likelihood function, so it makes intuitive sense that relatively few samples are required to fit the IPD lines. The speed and simplicity of sequential RANSAC make it the author's method of choice for clustering noisy, wrapped IPD data.

3.3 Blind source separation and DOA estimation

We will now see how the BSS and DOA estimation problems can be solved with sequential RANSAC.

3.3.1 Blind source separation

We review the BSS method proposed in [20] for the stereo unmixing case and elaborate on how this is extended to handle 3 or more microphones. Phase differences are calculated according to Equation (2.63), resulting in a dataset $\boldsymbol{\Delta} = \{\boldsymbol{\delta}_{f,t}\}$. Sequential RANSAC is then applied to fit K wrapped lines to this data. The $(f,t)^{\text{th}}$ data point is considered an inlier of the j^{th} line if $\delta_{f,t}$ is within $\pm\frac{\pi}{8}$ of the mean $\mu_{jf} = \psi(\alpha_j f)$. This is equivalent to the criterion:

$$\cos(\delta_{f,t} - \mu_{jf}) \geq \cos\left(\frac{\pi}{8}\right) \quad . \tag{3.8}$$

To recover the K source signals, we apply time-frequency (TF) masks to one of the mixture STFTs and apply the inverse STFT (see Section 2.1.1). The mask weights are given by the posterior probabilities:

$$w_{ftj} = \frac{P(\boldsymbol{\delta}_{f,t}\,;\,\mu_{jf},\kappa)}{\sum_{j=1}^{K} P(\boldsymbol{\delta}_{f,t}\,;\,\mu_{jf},\kappa)} = \frac{e^{\kappa\cos(\delta_{f,t}-\alpha_j f)}}{\sum_{j=1}^{K} e^{\kappa\cos(\delta_{f,t}-\alpha_j f)}} \quad . \tag{3.9}$$

This probability represents the soft assignment of the $(f,t)^{\text{th}}$ bin to the j^{th} source. Increasing κ results in a more aggressive separation. In the limit as $\kappa \to \infty$, Equation (3.9) reduces to a maximum-likelihood binary mask where each bin contributes to the reconstruction of only one source:

$$\forall\, f,t \quad w^b_{ftj} = \begin{cases} 1 & \text{if} \quad w_{ftj} = \max\limits_{l}\ w_{ftl} \\ 0 & \text{else} \end{cases} \quad . \tag{3.10}$$

We can extend the IPD feature vectors as in Equation (2.64) to take advantage of additional mics. The higher-dimensional data has multiple circular variables, increasing the inter-cluster distances. This leads to a better clustering and, therefore, separation. We calculate inliers by expanding the criterion in Equation (3.8):

$$\sum_{i=1}^{C-1} \cos(\delta_{f,t}(1,i+1) - \mu_{jfi}) \geq (C-1)\cos\left(\frac{\pi}{8}\right) \quad , \tag{3.11}$$

[2] Actually, this is not correct because multiple wrapped lines can pass through any data point. We can compensate by only sampling from low-frequency data. This is acceptable for speech separation since the most reliable TF bins are in this range. High-frequency bins are more noisy but still help to validate the fitting.

where $\mu_{jfi} = \psi(\alpha_{ij} f)$ is the value of the j^{th} wrapped line in the f^{th} frequency band and i^{th} circular axis. α_{ij} denotes the j^{th} slope in the i^{th} circular axis. This criterion implicitly assumes that regression error is measured with a multivariate von Mises (vM) distribution [104] whose dimensions are independent. The mask weights are also generalized via the multivariate vM:

$$w_{ftj} = \frac{P(\boldsymbol{\delta}_{f,t}\,;\,\boldsymbol{\mu}_{jf}, \kappa)}{\sum_{j=1}^{K} P(\boldsymbol{\delta}_{f,t}\,;\,\boldsymbol{\mu}_{jf}, \kappa)} = \frac{\prod_{i=1}^{C-1} e^{\kappa \cos(\delta_{f,t}(1,i+1) - \alpha_{ij} f)}}{\sum_{j=1}^{K} \prod_{i=1}^{C-1} e^{\kappa \cos(\delta_{f,t}(1,i+1) - \alpha_{ij} f)}} \quad . \tag{3.12}$$

3.3.2 Direction-of-arrival estimation

We can use sequential RANSAC to estimate the directions-of-arrival (DOAs) of multiple sound sources in a reverberant environment. Sequential RANSAC is run on an IPD dataset to estimate a set of wrapped lines. The slopes of the IPD lines are converted to inter-channel delays according to Equation (2.74). Finally, a DOA estimator from Section 2.6 can then be used to convert these delays into estimates of the source directions.

Least squares vs trigonometry

The two DOA estimators presented in Section 2.6 give essentially identical answers with non-degenerate arrays. However, in certain cases, such as localization on a hemisphere with 3 or more coplanar mics, the least-squares method requires the inversion of a rank-deficient matrix. This is remedied by first solving a rank-2 LS problem for the source direction within the plane of the array and then solving for the missing component perpendicular to the plane.

3.4 Experiments

In this section, we describe several experiments with blind source separation and DOA estimation for the case of stationary speakers.

3.4.1 Synthetic multimodal circular-linear data

We first investigate the performance of sequential RANSAC. Synthetic data was generated from a mixture of wrapped lines with moderate noise and uniform mixing weights. The slopes $\boldsymbol{\alpha}$ were sampled such that the corresponding DOAs θ were uniform in the range $[-\frac{\pi}{2}, \frac{\pi}{2}]$ with the condition that the angles differ by at least $\frac{\pi}{32}$. The linear variable ranges from 0 to 1 and the circular variable ranges from $-\pi$ to π. At most 3 full wrap-arounds were allowed. To test the robustness of the method to outliers, a portion of the data was sampled uniformly at random over the circular variable. Data points are counted as inliers if they are within $\pm\frac{\pi}{8}$ of the line.[3] During the fitting, all lines with slopes in the range $[-6\pi, 6\pi]$ that passed through a point were considered. This is necessary in the general case because multiple lines can be fit to a single sample.

[3] This is equivalent to the vM likelihood criterion: $P(\boldsymbol{\delta}\,;\,\mu\,,\,\kappa) > \frac{1}{2} P(\mu\,;\,\mu\,,\,\kappa)$, $\mu = \psi(\alpha f)$, $\kappa = 9.1$.

Table 3.1: Accuracy scores for sequential RANSAC with synthetic mixtures of K wrapped lines. A score above 75 indicates a nearly perfect fit (shaded in gray).

K \ % outliers	0	20	40	60	80
1	96.18	96.58	96.98	96.47	95.72
2	93.90	92.81	92.63	93.18	85.39
3	90.70	90.11	88.09	84.00	61.98
4	86.96	84.84	82.48	72.61	33.70
5	83.03	79.62	72.51	53.41	20.14

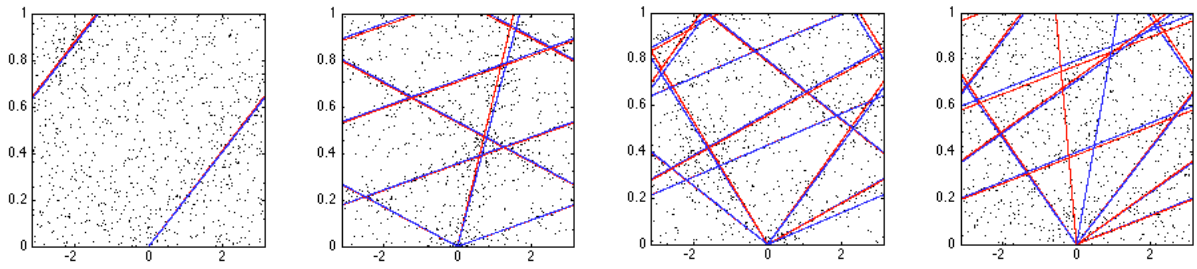


Figure 3.4: Models fit by sequential RANSAC. The red lines are the ground truth and the blue lines are inferred. (Far left) $K = 1$, % outliers = 80. (Middle left) $K = 3$, % outliers = 40. (Middle right) $K = 5$, % outliers = 40. (Far right) $K = 5$, % outliers = 80.

To quantify performance, a mixture of Laplacian distributions with scale parameter $b = 0.2$ and location parameters set to the true DOAs $\boldsymbol{\theta}$ was constructed. The following ratio of likelihoods was evaluated to generate an accuracy score between 0 and 100:

$$\mathcal{A}(\boldsymbol{\phi}\,;\,\boldsymbol{\theta}\,,\,b) = 100\,\frac{P(\boldsymbol{\phi}\,;\,\boldsymbol{\theta}\,,\,b)}{P(\boldsymbol{\theta}\,;\,\boldsymbol{\theta}\,,\,b)} = 100\,\frac{\prod_{i=1}^{K}\sum_{j=1}^{K} e^{-\frac{|\phi_i-\theta_j|}{b}}}{\prod_{i=1}^{K}\sum_{j=1}^{K} e^{-\frac{|\theta_i-\theta_j|}{b}}}\quad, \tag{3.13}$$

where ϕ_i is the i^{th} DOA found by sequential RANSAC. This accuracy function captures the peaked nature of the underlying likelihood surface shown in Figure 2.10(b) and so can be considered an appropriate measure of success. Inspecting the results, one can say that an accuracy score above 75 indicates an excellent fit.

100 trials were run for each model order-% outliers pair and 1000 data points were generated in each trial from a new mixture of wrapped lines. The average accuracy scores for these toy experiments are summarized in Table 3.1. Figure 3.4 shows examples of estimates returned by sequential RANSAC. It does an excellent job of inferring the wrapped lines in the presence of many outliers.

3.4.2 Blind source separation

To test sequential RANSAC's viability for BSS, we simulated a reverberant room with an array of omnidirectional microphones placed at its center. We mixed two-second sentences by five speakers from the TSP corpus [81]. They were positioned at random, distinct angles on the unit semicircle, circle, and sphere for 2-, 3-, and 4-channel unmixing. The microphones were positioned 5 cm apart in a right-angle configuration. We

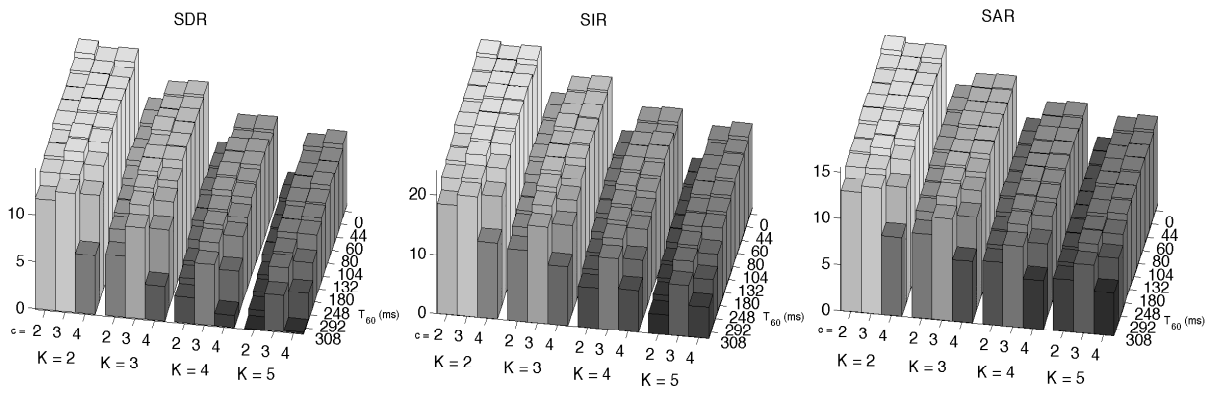


Figure 3.5: Signal-to-distortion, signal-to-interference, and signal-to-artifact ratios for 2- and 3-channel source separation in a 2D room and 4-channel source separation in a 3D room. C is the number of channels and K is the number of sources. (Figure appears in [20].)

ran 100 trials for $K = \{2, 3, 4, 5\}$ speakers with sentences downsampled to 16 kHz. STFTs were calculated with a window size of 1024 and an overlap factor of $\frac{3}{4}$.

We simulated reverberation with the image method [23], [105], with the T_{60} time[4] of the room varying from 0 to 308 milliseconds. We used a 2D room (5×5 m) for testing 2- and 3-channel separation and a 3D room ($5 \times 5 \times 5$ m) for testing 4-channel separation. We evaluated the separation performance with the BSS Eval toolbox [21]. This requires that we provide reference signals for each speaker. Because we are testing for source separation and not de-reverberation, the reference is chosen to be the convolution of the individual speaker signals with the appropriate room impulse responses. De-reverberation can be viewed as a post-processing step.

Figure 3.5 summarizes the performance with a binary mask. We can see that, in general, the separation quality improves with more microphones. However, there is a decrease in performance from the 2D to the 3D rooms. This is due to the increased amount of reverberation. Because speech signals are quite sparse, the loudest 10-20% of the TF bins should be used for clustering. Figure 3.3 depicts the fitting for a highly-reverberant, real-world case with two male speakers in a stairwell. Despite a T_{60} time of 1.5 seconds, sequential RANSAC succeeds in fitting the correct wrapped lines. A subjective assessment of the output confirmed that the speakers were mostly separated. This is remarkable given how harsh the conditions are for separation (and localization).

3.4.3 Direction-of-arrival estimation

Direction-of-arrival estimation in the context of sequential RANSAC is no more difficult than the source separation problem. This is due to the simple mapping from DOA to IPD line slopes (see Section 2.6). In other words, both the BSS and DOA estimation problems are solved simultaneously by sequential RANSAC. A likelihood surface over the DOA space can be calculated from Equation (2.72) via the mapping. This surface very closely resembles the histogram of inliers calculated by RANSAC. An example of this for DOA estimation on the unit sphere in an anechoic 3D room with a 4-microphone array is shown in Figure 3.6(a).

[4]The T_{60} time of a room is the time required for the reverberation energy to drop by 60 dB below the direct path energy.

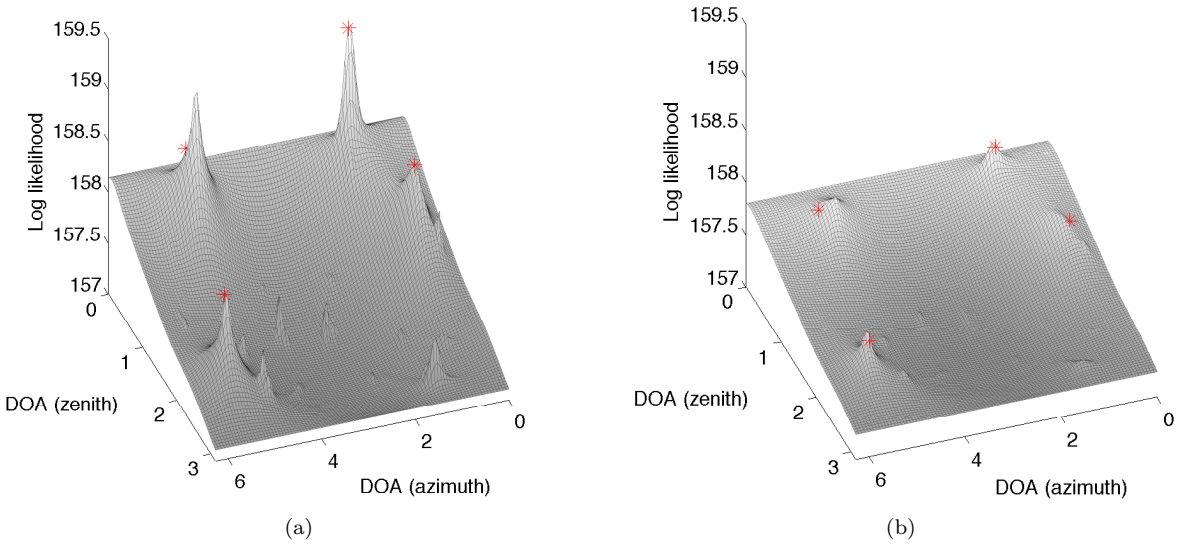


Figure 3.6: Likelihood surfaces for synthetic DOA estimation with 4 sources and a 4-microphone array. The speakers recite TSP sentences 2.5 seconds long. Every azimuth-zenith pair corresponds to a DOA on the unit sphere. Sequential RANSAC is successful in identifying the speaker directions (marked with red *'s). (a) Anechoic conditions. (b) Reverberant conditions with $T_{60} = 0.3$ seconds.

Reverberation reduces the sharpness of the surface, as shown in Figure 3.6(b), but even in these conditions, the sources are still located accurately.

The likelihood for a more complicated example with 6 sources is depicted in Figure 3.7 at each iteration of sequential RANSAC. We can see that each peak corresponds to a speaker direction and that the algorithm is able to successfully remove them one by one without propagating significant errors from one iteration to the next. It is interesting to note that the localization is accomplished just by random sampling from the dataset. This is much faster than scanning over all of DOA space, as is done in steered response power (SRP) methods [13]. We also observe that no TF bin selectivity was applied in calculating the likelihood surfaces. We can expect the peaks to be much more salient if only the TF bins with significant speech energy are fed to sequential RANSAC. This requires a pre-processing step that could, for example, find peaks in the magnitude STFT corresponding to the harmonics of speech [23].

3.4.4 Comparison with Bartlett beamformer and MUSIC

In this section, we compare the steered response power (SRP) of the Bartlett beamformer [13], the MUSIC pseudospectrum [28], and the IPD likelihood function. The Bartlett beamformer measures how much power is present at each DOA, frequency, and time by computing the following SRP function:

$$P_{f,t}(\theta) = |\boldsymbol{\varsigma}_f(\theta)^H \, \mathbf{X}_{f,t}|^2 \quad , \tag{3.14}$$

where

$$\mathbf{X}_{f,t} = \begin{bmatrix} X_{f,t}^{(1)} & \cdots & X_{f,t}^{(C)} \end{bmatrix}^T \quad , \tag{3.15}$$

is the vector of DFT coefficients from all the channels and

$$\boldsymbol{\varsigma}_f(\theta) = \begin{bmatrix} 1 & e^{-j\frac{2\pi f}{N}e_{12}} & \cdots & e^{-j\frac{2\pi f}{N}e_{1C}} \end{bmatrix}^T \quad , \tag{3.16}$$

is known as the "steering vector." It encodes the phase information that we expect to see for a signal arriving at frequency f and DOA θ and depends on the inter-channel delays e_{1i} , $i = 1, \ldots, C$. Accumulating over all time-frequency bins gives the SRP for the Bartlett beamformer:

$$P(\theta) = \sum_{t=1}^{T} \sum_{f=1}^{D} P_{f,t}(\theta) \quad . \tag{3.17}$$

This is equivalent to scanning DOA space with a delay-and-sum beamformer. We apply the PHAse Transform [24] (PHAT) to the inputs before calculating the SRP. This involves setting all the magnitudes of the DFT coefficients to 1, improving the response function by retaining only phase information.

The MUSIC spectrum is given by

$$R_{f,t}(\theta) = \frac{1}{\sum\limits_{j=2}^{C} |\boldsymbol{\varsigma}_f^H(\theta)\mathbf{b}_{f,t}^{(j)}|^2} \quad , \tag{3.18}$$

where $\mathbf{b}_{f,t}^{(j)}$ is the j^{th} eigenvector of the spatial correlation matrix:

$$\boldsymbol{\Sigma}_{f,t} = \mathrm{E}\left[\mathbf{X}_{f,t}\, \mathbf{X}_{f,t}^H\right] = \mathbf{B}_{f,t}\, \boldsymbol{\Lambda}_{f,t}\, \mathbf{B}_{f,t}^H \quad , \tag{3.19}$$

$$\mathbf{B}_{f,t} = \begin{bmatrix} \mathbf{b}_{f,t}^{(1)} & \cdots & \mathbf{b}_{f,t}^{(C)} \end{bmatrix} \quad , \quad \boldsymbol{\Lambda}_{f,t} = diag\left(\begin{bmatrix} \lambda_{f,t}^{(1)} & \cdots & \lambda_{f,t}^{(C)} \end{bmatrix}\right) \quad , \tag{3.20}$$

and the eigenvalues are sorted in non-ascending order. Under additive white-noise assumptions, peaks will be present in the accumulated MUSIC spectrum:

$$R(\theta) = \sum_{t=1}^{T} \sum_{f=1}^{D} R_{f,t}(\theta) \quad . \tag{3.21}$$

We can compare the functions $P(\theta)$ and $R(\theta)$ with the von Mises (vM) likelihood in Equation (2.72) over DOA space. A hard-thresholded alternative is calculated by accepting features within some distance of the IPD line. Figure 3.8 shows these functions for a 3-mic array arranged in an equilateral triangle with 3 centimeters of spacing. Three TSP speakers are located at $\theta = -\frac{\pi}{3}, 0, \frac{\pi}{3}$. The vM concentration and inlier threshold are set to $\{1000, \frac{\pi}{16}\}$ and $\{100, \frac{\pi}{8}\}$ for the anechoic and echoic cases. The T_{60} time for the echoic case is 30 milliseconds. The IPD likelihood surface retains its peaks in a reverberant environment while the MUSIC spectrum does not. It also requires fewer computations since no eigendecompositions are needed. The SRP-PHAT function, on the other hand, can only be used to estimate two of the speakers' locations.

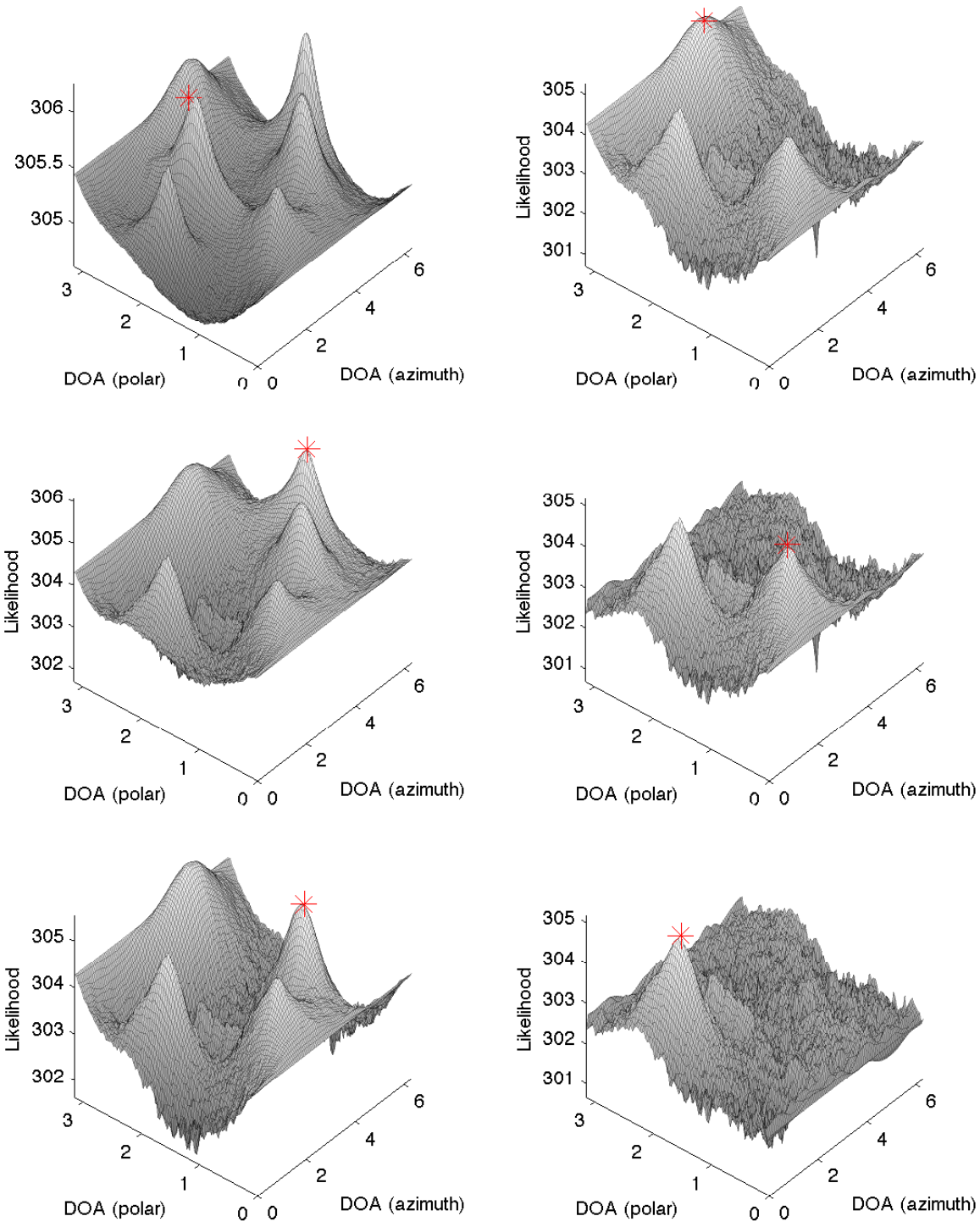


Figure 3.7: Likelihood surfaces for 4-microphone localization of 6 speakers in anechoic conditions. Sequential RANSAC is able to incrementally identify the DOAs of all 6 speakers by, in effect, sampling from the likelihood surface and removing inliers. The figures are ordered downwards with the left column first.

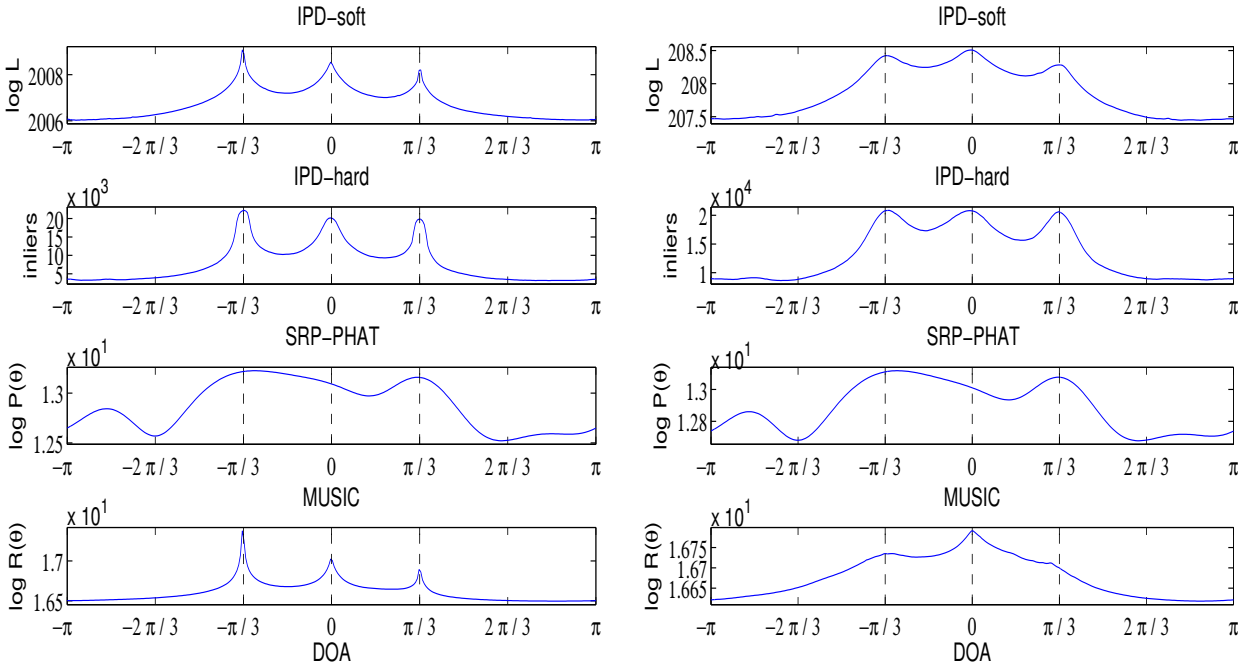


Figure 3.8: SRP-PHAT response power, MUSIC spectrum, and IPD likelihood for a simulated, 3-microphone, 3-speaker experiment. Dashed lines indicate the source positions. (Left) Anechoic conditions. (Right) Echoic conditions. The T_{60} reverberation time is 30 milliseconds and the room dimensions are 5×5 meters. The peaks corresponding to the speakers can be seen under both conditions only in the IPD-based function.

CHAPTER 4

DIRECTION-OF-ARRIVAL TRACKING WITH IPD FEATURES

In the previous chapter, we saw how IPD features can be clustered to perform source separation and localization. However, it was assumed that the speakers were physically stationary. This is often not the case in a real-world environment. If we allow the speakers to move, the IPD line slopes become dependent on time. This can be more appropriately expressed as a tracking problem in direction-of-arrival (DOA) space. If we are given the sources' DOA paths over time, the separation problem reduces to that of the previous chapter. Thus, we will focus solely on the problem of multiple DOA tracking.

We will apply Bayesian filtering techniques to estimate the unobserved DOA paths (see Section 2.7 for an overview). The observation can be DOA votes calculated by applying sequential RANSAC to a few STFT frames worth of IPD data or the raw IPD features themselves. In either case, we will use directional distributions to track speakers directly on the unit circle and sphere (see Section 2.2 for a review of directional statistics). The Kalman filter is modified to handle tracking on the unit circle, $\mathbb{S}^1$, by modeling the state distribution of a wrapped dynamical system with a wrapped Gaussian (WG). This new algorithm is called the wrapped Kalman filter (WKF). A mixture of WGs is then used to track multiple sources on $\mathbb{S}^1$ with the factorial WKF (FWKF). We will see that the WKF and FWKF can be interpreted as good approximations of switching Kalman filter [98] procedures.

The von Mises-Fisher (vMF) distribution is then used to model the state distribution of a spherical dynamical system. The tracking is performed with a von Mises-Fisher particle filter (vMFPF) [61]. Finally, this is expanded to the multiple-source case with a mixture of vMFs to yield the factorial von Mises-Fisher particle filter (FvMFPF). When multiple speakers are present, there is a data association ambiguity because we do not know what state generated what observation (see Section 2.7.4 for a summary of the problem). To remedy this, we will incorporate posterior probabilities into the filter equations that capture the probabilistic assignments of observations to source clusters.

4.1 Bayesian tracking on the unit circle

We now describe algorithms for tracking directly on the unit circle with the wrapped Gaussian (WG) distribution. The WG contains an infinite sum of periodic Gaussians, so the filtered state distribution can only be approximated as WG from one time to the next. However, for low to moderate noise levels, this is acceptable.

4.1.1 State space models for wrapped filtering

In this section, we present two dynamical systems that model the evolution of a circular state variable: a rotating vector model and the wrapped dynamical system.

Rotating vector model

The following rotating-vector state space model [54] is often used for filtering with circular data:

$$\mathbf{x}_t = \begin{bmatrix} 1 & dt \\ 0 & 1 \end{bmatrix} \mathbf{x}_{t-1} + \mathbf{v}_t \ , \ \ \mathbf{v}_t \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \sigma_{v,1}^2 & 0 \\ 0 & \sigma_{v,2}^2 \end{bmatrix}\right) \tag{4.1}$$

$$\mathbf{y}_t = \begin{bmatrix} \cos(x_{t,1}) \\ \sin(x_{t,1}) \end{bmatrix} + \mathbf{w}_t \ , \ \ \mathbf{w}_t \sim \mathcal{N}\left(\mathbf{0}, \sigma_w^2 \mathbf{I}\right) \ . \tag{4.2}$$

The state vector $\mathbf{x}_t \in \mathbb{R}^2$ consists of position and velocity, $\mathbf{y}_t \in \mathbb{R}^2$ is the observation vector, dt is the time increment, and $\mathbf{v}_t \in \mathbb{R}^2$ and $\mathbf{w}_t \in \mathbb{R}^2$ are the process and measurement noise, respectively. The measurement in Equation (4.2) involves a non-linear transformation of the state. The extended Kalman filter (EKF) [54] approximates this by a first-order linearization, whereas the unscented Kalman filter (UKF) [55] does so with the unscented transform.

The drawback of this model is that it regards the observation as a 2D vector when the state is truly 1D (and can be inferred via $\angle \mathbf{y}_t$). This introduces additional noise to the system that limits the tracking capabilities of the filters. Contours of the distribution of $\mathbf{y}_t$ are shown in the top panel of Figure 2.5.

Wrapped dynamical system

The wrapped dynamical system (WDS) is described by the following state space model:

$$\theta_t = \psi\left(\theta_{t-1} + v_t\right) \ , \ \ v_t \sim \mathcal{N}\left(0, \sigma_v^2\right) \tag{4.3}$$

$$y_t = \psi\left(\theta_t + w_t\right) \ , \ \ w_t \sim \mathcal{N}(0, \sigma_w^2) \tag{4.4}$$

where $\theta_t, y_t \in \mathbb{S}^1$ and $v_t, w_t \in \mathbb{R}^1$. The wrapping in the state space is not crucial to the operation of the WKF, but we include it here (in the generative model) for completeness. Additional state information can easily be included by extending the state vector as it is done in the traditional Kalman filter since these quantities are not wrapped and do not appear in Equation (4.4). For simplicity, we will only consider position in the derivation of the WKF. A typical sample path for the WDS is shown in Figure 4.1 along with an observation sequence and the state path estimated by the WKF.

The advantage of the WDS over the rotating vector model is that the observations are treated as 1D quantities. We can expect that filtering in this model will be more accurate since it is easier to infer the hidden state sequence $\theta_{1:T}$ from lower-dimensional measurements. This fact is used to reduce the variance of particle filtering schemes and is mathematically formalized in the Rao-Blackwell theorem [39]. Conceptually, the WDS is an application of this approach to the rotating vector model in which the radius of $\mathbf{y}_t$ in Equation (4.2) is marginalized out (see top panel of Figure 2.5).

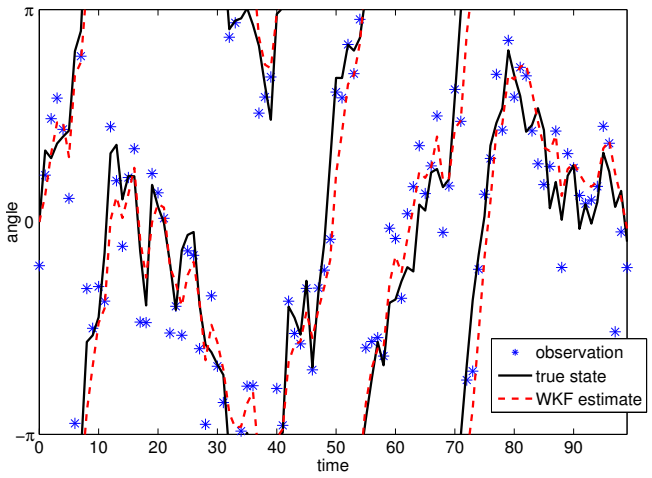


Figure 4.1: Sample path and observation sequence for the wrapped dynamical system with position and velocity state components. The WKF tracks the WDS despite wrapping effects at $-\pi$ and π. ($\sigma^2_{v,1} = \sigma^2_w = 0.5, \sigma^2_{v,2} = 0.001$)

4.1.2 Wrapped Kalman filter (WKF)

Now we derive the wrapped Kalman filter (WKF) and show that the *correct* step can be interpreted in two equivalent ways. In one, the WG state distribution is updated using a single observation, and in the other, a representative component of the state distribution is updated via 2π-periodic copies of the observation. The latter interpretation suggests a measurement fusion strategy that leads to the WKF algorithms.

The WG allows us to model the wrapping function $\psi(-)$ in Equations (4.3) and (4.4). The filtered state distribution at time $t-1$ is:

$$P(\theta_{t-1}|y_{1:t-1}) = \sum_{l=-\infty}^{\infty} P_l(\theta_{t-1}|y_{1:t-1}) \tag{4.5}$$

$$= \sum_{l=-\infty}^{\infty} \mathcal{N}(\theta_{t-1}\,;\, \mu_{t-1} + 2\pi l, \sigma^2_{t-1}) \ . \tag{4.6}$$

At each step, we first *predict* the next state distribution:

$$P(\theta_t|y_{1:t-1}) = \int P(\theta_t|\theta_{t-1})\ P(\theta_{t-1}|y_{1:t-1})\, d\theta_{t-1} \tag{4.7}$$

$$= \int P(\theta_t|\theta_{t-1}) \sum_{l=-\infty}^{\infty} P_l(\theta_{t-1}|y_{1:t-1})\, d\theta_{t-1} \tag{4.8}$$

$$= \sum_{l=-\infty}^{\infty} \int P(\theta_t|\theta_{t-1})\ P_l(\theta_{t-1}|y_{1:t-1})\, d\theta_{t-1} \tag{4.9}$$

$$= \sum_{l=-\infty}^{\infty} P_l(\theta_t|y_{1:t-1}) \ , \tag{4.10}$$

where it is assumed that the state space is not wrapped (i.e. the wrap function $\psi(-)$ is not present in

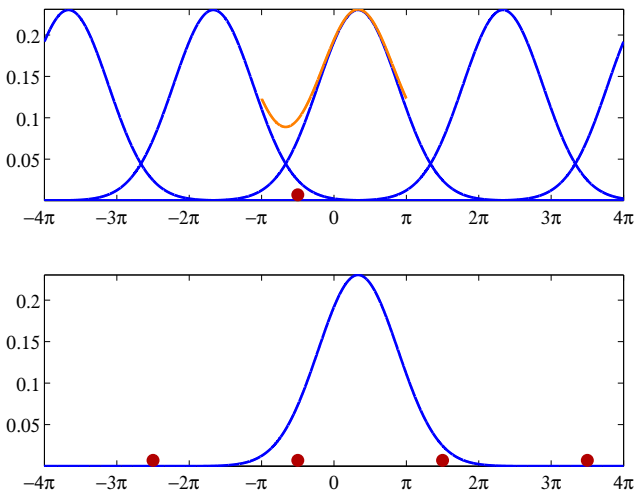


Figure 4.2: Two interpretations of the *correct* step in the wrapped Kalman filter. (Top) Single observation and periodic Gaussians (wrapped Gaussian in $[-\pi, \pi]$ overlaid). (Bottom) Single Gaussian and periodic observations. ($\mu = \frac{\pi}{3}$, $\sigma^2 = 3$)

Equation (4.3)). This does not introduce any issues because the wrapping is modeled in Equation (4.4) anyway. After propagating the state distribution forward, we *correct* the prediction:

$$P(\theta_t|y_{1:t}) \propto P(y_t|\theta_t)P(\theta_t|y_{1:t-1}) \tag{4.11}$$

$$\propto \left[\sum_{m=-\infty}^{\infty} P_m(y_t|\theta_t)\right]\left[\sum_{l=-\infty}^{\infty} P_l(\theta_t|y_{1:t-1})\right] \tag{4.12}$$

$$\propto \sum_{m=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} P_m(y_t|\theta_t)\ P_l(\theta_t|y_{1:t-1}) \ . \tag{4.13}$$

The true posterior is an exponentially-growing sum of increasingly differing Gaussian components. We can approximate it at time t with a WG by considering a single term of the predicted density and interpreting the observation as being replicated (see Figure 4.2):

$$P(\theta_t|y_{1:t}) = \sum_{l=-\infty}^{\infty} P_l(\theta_t|y_{1:t}) \tag{4.14}$$

$$\approx \sum_{l=-\infty}^{\infty} P_0(\theta_t, z_t = l|y_{1:t}) \tag{4.15}$$

$$= \sum_{l=-\infty}^{\infty} P_0(\theta_t|z_t = l, y_{1:t})\ \eta_{t,l} \tag{4.16}$$

$$\propto \sum_{l=-\infty}^{\infty} P(y_t|\theta_t, z_t = l, y_{1:t-1})\ P_0(\theta_t|z_t = l, y_{1:t-1})\ \eta_{t,l} \tag{4.17}$$

$$= \sum_{l=-\infty}^{\infty} P(y_t + 2\pi l|\theta_t)\ P_0(\theta_t|y_{1:t-1})\ \eta_{t,l} \ , \tag{4.18}$$

Algorithm 9 Wrapped Kalman filter

Predict

$\widehat{\mu}_t^- = \widehat{\mu}_{t-1}$

$\widehat{\sigma}_t^{2,-} = \widehat{\sigma}_{t-1}^2 + \sigma_v^2$

Correct

$K_t = \frac{\widehat{\sigma}_t^{2,-}}{\widehat{\sigma}_t^{2,-} + \sigma_w^2}$

$\bar{y}_t = 2\pi \,\text{round}\left(\frac{\widehat{\mu}_t^-}{2\pi}\right) + y_t$

$\eta_{t,l} = \frac{\mathcal{N}\left(\bar{y}_t + 2\pi l \,;\, \widehat{\mu}_t^-, \sigma_w^2\right)}{\sum_{l=-\infty}^{\infty} \mathcal{N}\left(\bar{y}_t + 2\pi l \,;\, \widehat{\mu}_t^-, \sigma_w^2\right)}$

$g_t = \sum_{l=-\infty}^{\infty} \left((\bar{y}_t + 2\pi l) - \widehat{\mu}_t^-\right) \eta_{t,l}$

$\widehat{\mu}_t = \widehat{\mu}_t^- + K_t \, g_t$

$\widehat{\sigma}_t^2 = (1 - K_t)\, \widehat{\sigma}_t^{2,-}$

where

$$\eta_{t,l} = P\left(z_t = l \,|\, y_t \,;\, \mu_t, \sigma_w^2\right) \quad , \tag{4.19}$$

represents the probability of a replicate. The posterior at time t is approximated by finding the closest Gaussian distribution to Equation (4.18) via moment-matching and then repeating it every 2π to form a WG. Thus, the *correct* step is equivalent to a measurement fusion step [106]. In the WKF, we form a weighted average of the innovations due to the copies of y_t. The resulting composite innovation g_t is used to correct the state estimate.

The filtering procedure is summarized in Algorithm 9. A hat over a variable indicates an estimate and a minus sign superscript indicates a prediction. $\widehat{\mu}_t$ is the estimated state, $\widehat{\sigma}_t^2$ is the corresponding variance, and K_t is the Kalman gain. The WKF works with the shifted observation $\bar{y}_t$ to account for the drift of $\widehat{\mu}_t$ out of $[-\pi, \pi]$. This allows us to truncate the WG to 3 terms ($l = -1, 0, 1$) in practice since we only care about wrapping effects in $[\widehat{\mu}_t - 2\pi, \widehat{\mu}_t + 2\pi]$ at time t. Thus, we only need to consider 3 replicates of y_t. For very high noise levels (e.g. $\sigma_w^2 > 2$), this degree of truncation may be inadequate. However, we cannot expect to track the state with any confidence in such harsh conditions, so we will ultimately only be interested in cases where 3 terms suffice.

4.1.3 WKF as an approximation of a switching Kalman filter

Modeling the state distribution as a set of periodic Gaussians implies a generative model where we sample the observation from a single WG component. We can incorporate a hidden indicator variable z_t that selects what component is active at time t. The result is that we have a switching measurement equation. The state is a vector $\boldsymbol{\theta}_t$ of the WG component means and the observation y_t is a selected mean plus noise. The corresponding (switching) state space model is given as:

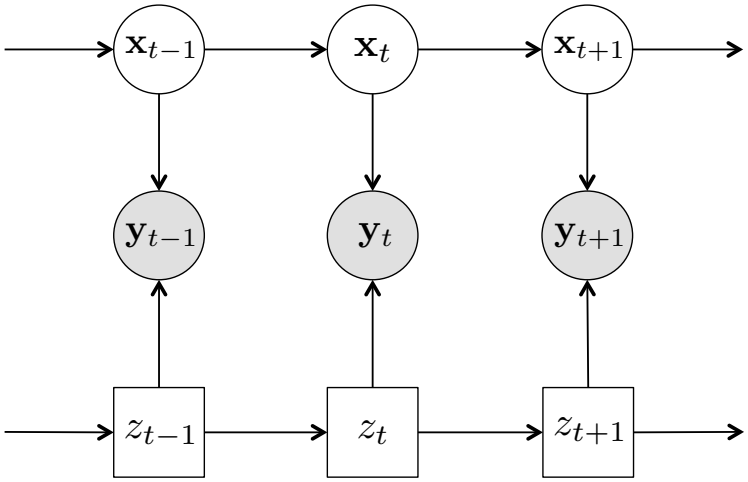


Figure 4.3: Graphical model of a switching dynamic Bayesian network. The switch variable z_t selects the measurement model that is active at time t.

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} + \mathbf{v}_t \quad , \quad \mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, \sigma_v^2 \mathbf{J}) \ , \tag{4.20}$$

$$y_t = B_{z_t} \boldsymbol{\theta}_t + w_t \ , \ w_t \sim \mathcal{N}(0, \sigma_w^2) \quad , \tag{4.21}$$

where $\mathbf{J}$ is the matrix of ones and z_t selects a measurement matrix with a single 1 in the position of the active WG component:

$$B_{z_t} = \begin{bmatrix} \cdots & 0 & 0 & 1 & 0 & 0 & \cdots \end{bmatrix} \ . \tag{4.22}$$

Implicitly, there is also a Markovian dynamics model, $P(z_t|z_{t-1})$, for the switch variable that describes how its statistics change over time. Figure 4.3 depicts the graphical model for a generic switching dynamical system.

The state distribution has a rank-1 covariance matrix $\Sigma_t = \sigma_t^2 \mathbf{J}$ because the WG means (components of $\boldsymbol{\theta}_t$) are locked together. The typical approach to inference in this model is the switching Kalman filter (SKF) [98]. The WKF is equivalent to the SKF when the transition distribution $P(z_t|z_{t-1})$ is modeled as uniform. We find that this is not a significant drawback as the weights $\eta_{t,l}$ are sufficient to capture transitions between neighboring WG components. Forming the composite innovation g_t is analogous to the "collapse" operation of the SKF.

4.1.4 Discussion of the WKF

In practice, one only needs to evaluate 3 terms ($l = -1, 0, 1$) in the infinite summation in line 6 of Algorithm 9 because the WKF operates with the shifted observation $\bar{y}_t$. If we could implement the infinite summation, a shift would be unnecessary. If all we care about is tracking the WDS in $\mathbb{S}^1$, the state estimate $\widehat{\mu}_t$ can be

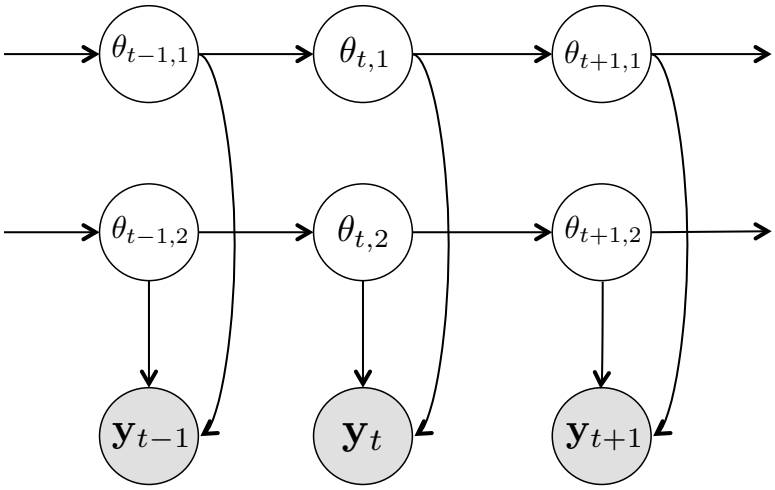


Figure 4.4: Graphical model of the factorial wrapped dynamical system for multi-source tracking on the unit circle where the number of sources $K = 2$.

wrapped to $[-\pi, \pi]$ after the *correct* step to avoid computing line 4. However, for an application such as 1D phase unwrapping, $\widehat{\mu}_t$ should be allowed to drift out of $\mathbb{S}^1$. As they are presented, the WKF equations can be applied directly for either task.

4.1.5 Factorial wrapped Kalman filter (FWKF)

When multiple sources are active, we have a factorial WDS (FWDS) with identical dynamics in all the systems. The graphical model corresponding to the FWDS is shown in Figure 4.4. The state-space model can be written as:

$$\boldsymbol{\theta}_t = \psi\left(\boldsymbol{\theta}_{t-1} + \mathbf{v}_t\right) \quad , \quad \mathbf{v}_t \sim \mathcal{N}\left(\mathbf{0}, \sigma_v^2 \mathbf{I}\right) \quad , \tag{4.23}$$

$$\mathbf{y}_t = \psi\left(\boldsymbol{\theta}_t + \mathbf{w}_t\right) \quad , \quad \mathbf{w}_t \sim \mathcal{N}\left(\mathbf{0}, \sigma_w^2 \mathbf{I}\right) \quad , \tag{4.24}$$

where the state and measurement are both vectors of length K.

A complication arises because we do not observe the measurements in any particular order. We cannot tell what system generated what measurement. The data association ambiguity is resolved by modeling the state distribution with a mixture of wrapped Gaussians (MoWG) [60] (see Section 2.3). Thus, we can derive a factorial variant of the WKF, summarized in Algorithm 10. The posterior probabilities

$$\eta_{t,jlm} = P\left(z_{t,m} = \{j, l\} \,|\, \bar{y}_{t,jm}\,;\, \widehat{\mu}_{t,j}^{-}, \sigma_w^2, \widehat{\pi}_{t,j}^{-}\right) \quad , \tag{4.25}$$

represent how likely it is that the l^{th} component of the j^{th} WG generated the m^{th} observation and are constrained as follows:

$$\forall\, t, m \quad \sum_{j=1}^{K} \sum_{l=-\infty}^{\infty} \eta_{t,jlm} = 1 \quad . \tag{4.26}$$

Algorithm 10 Factorial wrapped Kalman filter

Predict

$\widehat{\mu}_{t,j}^{-} = \widehat{\mu}_{t-1,j}$

$\widehat{\sigma}_{t,j}^{2,-} = \widehat{\sigma}_{t-1,j}^{2} + \sigma_v^2$

$\widehat{\pi}_{t,j}^{-} = \widehat{\pi}_{t-1,j}$

Correct

$K_{t,j} = \frac{\widehat{\sigma}_{t,j}^{2,-}}{\widehat{\sigma}_{t,j}^{2,-} + \sigma_w^2}$

$\bar{y}_{t,jm} = 2\pi \,\mathrm{round}\left(\frac{\widehat{\mu}_{t,j}^{-}}{2\pi}\right) + y_{t,m}$

$\eta_{t,jlm} = \frac{\mathcal{N}\left(\bar{y}_{t,jm}+2\pi l \,;\, \widehat{\mu}_{t,j}^{-}, \sigma_w^2\right) \widehat{\pi}_{t,j}^{-}}{\sum_{j=1}^{K}\sum_{l=-\infty}^{\infty} \mathcal{N}\left(\bar{y}_{t,jm}+2\pi l \,;\, \widehat{\mu}_{t,j}^{-}, \sigma_w^2\right) \widehat{\pi}_{t,j}^{-}}$

$g_{t,j} = \frac{\sum_{m=1}^{K}\sum_{l=-\infty}^{\infty} \left((\bar{y}_{t,jm}+2\pi l) - \widehat{\mu}_{t,j}^{-}\right) \eta_{t,jlm}}{\sum_{m=1}^{K}\sum_{l=-\infty}^{\infty} \eta_{t,jlm}}$

$\widehat{\mu}_{t,j} = \widehat{\mu}_{t,j}^{-} + K_{t,j}\, g_{t,j}$

$\widehat{\sigma}_{t,j}^{2} = (1 - K_{t,j})\, \widehat{\sigma}_{t,j}^{2,-}$

$\widehat{\pi}_{t,j} = \frac{1}{K} \sum_{m=1}^{K}\sum_{l=-\infty}^{\infty} \eta_{t,jlm}$

The FWKF reduces to the WKF when $K = 1$. As with the WKF, we can incorporate other state components like velocity since they are not wrapped and do not enter into the measurement equation.

4.1.6 Derivation of FWKF

The FWKF is derived by applying the Bayesian filtering Equations (2.88) and (2.92). The filtered state distribution at time $t-1$ is modeled as a MoWG:

$$P\left(\theta_{t-1}|\mathbf{y}_{1:t-1}\right) = \sum_{j=1}^{K} \pi_{t-1,j} \sum_{l=-\infty}^{\infty} P_l\left(\theta_{t-1,j}|\mathbf{y}_{1:t-1}\right) \quad , \tag{4.27}$$

$$= \sum_{j=1}^{K} \pi_{t-1,j} \sum_{l=-\infty}^{\infty} \mathcal{N}\left(\theta_{t-1,j}|\mu_{t-1,j} + 2\pi l, \sigma_{t-1,j}^2\right) \quad . \tag{4.28}$$

The *predict* step is like that of the WKF:

$$P\left(\theta_t|\mathbf{y}_{1:t-1}\right) = \int P\left(\theta_t|\theta_{t-1}\right)\ P\left(\theta_{t-1}|\mathbf{y}_{t-1}\right) d\theta_{t-1} \tag{4.29}$$

$$= \int P\left(\theta_t|\theta_{t-1}\right)\ \sum_{j=1}^{K} \pi_{t-1,j} \sum_{l=-\infty}^{\infty} P_l\left(\theta_{t-1,j}|\mathbf{y}_{1:t-1}\right) d\theta_{t-1} \tag{4.30}$$

$$= \sum_{j=1}^{K} \pi_{t-1,j} \sum_{l=-\infty}^{\infty} \int P\left(\theta_t|\theta_{t-1}\right)\ P_l\left(\theta_{t-1,j}|\mathbf{y}_{1:t-1}\right) d\theta_{t-1} \tag{4.31}$$

$$= \sum_{j=1}^{K} \pi_{t-1,j} \sum_{l=-\infty}^{\infty} P_l\left(\theta_{t,j}|\mathbf{y}_{1:t-1}\right) \quad . \tag{4.32}$$

The *correct* step is nontrivial in that the filtered state distribution does not remain a MoWG over time. But we can approximate it as such with one component from each WG:

$$P\left(\theta_t|\mathbf{y}_{1:t}\right) = \sum_{j=1}^{K} \pi_{t,j} \sum_{l=-\infty}^{\infty} P_l\left(\theta_{t,j}|\mathbf{y}_{1:t}\right) \tag{4.33}$$

$$\approx \sum_{j=1}^{K} \pi_{t,j} \sum_{l=-\infty}^{\infty} \sum_{m=1}^{K} P_0\left(\theta_{t,j}, z_{t,m} = \{j,l\}|\mathbf{y}_{1:t}\right) \tag{4.34}$$

$$= \sum_{j=1}^{K} \pi_{t,j} \sum_{l=-\infty}^{\infty} \sum_{m=1}^{K} P_0\left(\theta_{t,j}|z_{t,m} = \{j,l\}, \mathbf{y}_{1:t}\right) \eta_{t,jlm} \tag{4.35}$$

$$\propto \sum_{j=1}^{K} \pi_{t,j} \sum_{l=-\infty}^{\infty} \sum_{m=1}^{K} P\left(\mathbf{y}_t|\theta_{t,j}, z_{t,m} = \{j,l\}, \mathbf{y}_{1:t-1}\right)\ P_0\left(\theta_{t,j}|z_{t,m} = \{j,l\}, \mathbf{y}_{1:t-1}\right) \eta_{t,jlm} \tag{4.36}$$

$$= \sum_{j=1}^{K} \pi_{t,j} \sum_{l=-\infty}^{\infty} \sum_{m=1}^{K} P\left(y_{t,m} + 2\pi l|\theta_{t,j}\right)\ P_0\left(\theta_{t,j}|\mathbf{y}_{1:t-1}\right) \eta_{t,jlm} \tag{4.37}$$

The posterior means of each WG are found by moment-matching as in the WKF. The summations over m and l consolidate information due to the observations $\mathbf{y}_t$. This makes it possible to construct composite innovations $g_{t,j}$ that are used to update the cluster means $\widehat{\mu}_{t,j}$. The variances $\widehat{\sigma}^2_{t,j}$ are updated as in the usual Kalman filter. Finally, the mixture weights $\widehat{\pi}_{t,j}$ are updated as in the EM algorithm for learning a MoWG [60].

4.1.7 FWKF as an approximation of a switching Kalman filter

The FWKF raises the same conceptual concern as the WKF. Modeling the state distribution as a wrapped mixture implies a generative model where, in each time step, we sample from a MoWG K times to get a set of observations. This is technically incorrect because we actually want to draw only once from each mixture component. Thus, the observations are sampled in a highly non-i.i.d. fashion since, once we sample from one WG component, we will not sample from it again until the next time step.

If we ignore the wrapping complication for a moment, we can represent the non-i.i.d. sampling with a hidden indicator variable z_t that represents the permutation of the samples at time t. Thus, the FWDS

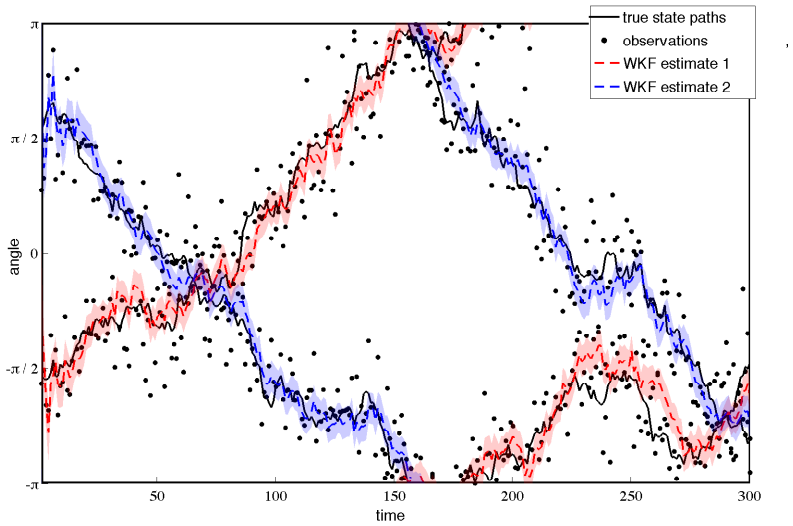


Figure 4.5: Sample paths and observation sequences for factorial wrapped dynamical system with position and velocity in the state. Mean paths estimated with the FWKF are shown with their 1-σ contours.

requires a switching measurement equation. The state is comprised of a K-component vector and the observations are a permutation of the state vector plus noise:

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} + \mathbf{v}_t \quad , \quad \mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, \sigma_v^2 \mathbf{I}) \quad , \tag{4.38}$$

$$\mathbf{y}_t = B_{z_t}\, \boldsymbol{\theta}_t + \mathbf{w}_t \quad , \quad \mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, \sigma_w^2 \mathbf{I}) \quad , \tag{4.39}$$

where z_t selects one of $K!$ permutation matrices B. This implies that we could apply a standard switching Kalman filter (SKF) [98] to perform inference in this model. Unfortunately, once the wrapping nature of the FWDS is taken into account, the number of switch states becomes prohibitively large. The FWKF considers only $\mathcal{O}(K^2)$ terms and is conceptually equivalent to a SKF with uniform transition probabilities $P(z_t|z_{t-1})$. As in the WKF, forming the composite innovations is a measurement fusion step [106] analogous to the "collapse" operation in the SKF.

4.1.8 Discussion of FWKF

Figure 4.5 shows typical sample paths and observation sequences from the FWDS with K=2 sources. The FWKF successfully tracks the sources despite data association ambiguities, cross-over of the state paths, and wrapping. We make several remarks concerning this approach.

Estimation with more than K observations

We may consider the case where *at least* one observation is generated from each source, i.e. the index m ranges in $[1, M]$ for $M > K$. Since the observations are assigned probabilistically, more data can be incorporated without additional design effort to improve the tracking. For large M, the FWKF has a substantial computational advantage over the SKF.

Over-estimating K

The MoWG weights $\boldsymbol{\pi}$ should remain uniform to conform to the switching formulation. Nevertheless, they are useful when we are not sure how many states are active. Overestimating K cannot hurt since a redundant WG component will tend to have low weight in the mixture.

Collisions

Clusters interact gracefully because of the soft assignments. Including velocity in the state helps to disambiguate cross-overs by tracking the movement of either of the colliding clusters. It is not clear that this is always the ideal behavior. For example, it may be that the two states actually "bounce" rather than pass by each other. However, this departs from the assumption in the FWDS that the state chains do not interact. Additional dynamics information is needed to handle the "bounce" behavior. This may be resolved with a method for matching the clusters before and after the collision. In the case of tracking speakers, we can classify their identities with simple speaker-dependent models that match them to clusters.

Coalescence

A drawback of soft assignments is that clusters tend to "coalesce" when they adapt to the same observations [63] (e.g. during a cross-over). This can be observed, for example, at time step 230 in Figure 4.5. A proper statistical solution would be to regularize the FWKF updates with a peaky Gaussian penalty term to push the clusters apart when their means are similar. A simple heuristic is to add a small amount of noise, e.g. $\mathcal{N}(0, 1/50)$, to the cluster means when $\cos(\widehat{\mu}_j - \widehat{\mu}_{j'}) > \cos(\pi/10)$.

Convergence of filter parameters

As in the Kalman filter, the variance parameters are independent of the observed data. All variances converge to a single steady state value, as do the Kalman gains $K_{t,j}$. As discussed in [46], a Gaussian sum filter (GSF) may converge to a steady state (i.e. fixed $\boldsymbol{\pi}$). We find this to be true of the FWKF as well when K is over-estimated and the state paths are well-separated.

4.2 Bayesian tracking on the unit sphere

We can adapt the WKF/FWKF design procedure to derive Bayesian filters for tracking on $\mathbb{S}^2$. This is important when the azimuth angle alone is insufficient and we would like to track in a half-sphere or sphere. The WKF/FWKF were straightforward to derive because the distributions involved were composed of 2π-periodic Gaussians. Unfortunately, wrapping a 2D distribution on the unit sphere is nontrivial due to the unique topology. So we turn to sequential Monte Carlo methods, also called particle filters. A review of particle filtering can be found in Section 2.7.3 or [38]. In this section, we will describe two algorithms: the von Mises-Fisher particle filter (vMFPF) and a generalization to the multi-source case called the factorial vMFPF (FvMFPF).

Algorithm 11 Von Mises-Fisher particle filter

Predict

$\mathbf{x}_t^{(l),-} \sim vMF\left(\mathbf{x}_{t-1}^{(l)}, \kappa_v\right)$

Correct

$\gamma_t^{(l)} = vMF\left(\mathbf{y}_t\,;\, \mathbf{x}_t^{(l),-}, \kappa_w\right)$

$w_t^{(l),-} \propto w_{t-1}^{(l)}\, \gamma_t^{(l)}$

Resample

$\mathbf{x}_t^{(l)} \sim \sum_{m=1}^{L} w_t^{(m),-}\, \delta\left(\mathbf{x} - \mathbf{x}_t^{(m),-}\right)$

$w_t^{(l)} = 1\,/\,L$

4.2.1 Von Mises-Fisher particle filter (vMFPF)

We can define a state space model for single-source tracking on the unit sphere referred to as a spherical dynamical system (SDS). The state transition and measurement equations are given as:

$$\mathbf{x}_t \sim vMF(\mathbf{x}_{t-1}, \kappa_v) \ , \tag{4.40}$$

$$\mathbf{y}_t \sim vMF(\mathbf{x}_t, \kappa_w) \ . \tag{4.41}$$

The filtered distribution does not maintain a closed form representation, so we will use a set of weighted particles:

$$\mathbf{X}_t = \left\{\mathbf{x}_t^{(l)}, w_t^{(l)}\right\} \quad , \quad l = 1, \ldots, L \ , \tag{4.42}$$

to approximate the underlying state distribution as:

$$P\left(\mathbf{x}_t | \mathbf{y}_{1:t}\right) \approx \sum_{l=1}^{L} w_t^{(l)}\, \delta\left(\mathbf{x} - \mathbf{x}_t^{(l)}\right) \ . \tag{4.43}$$

The particles $\mathbf{x}_t^{(l)}$ serve as point estimates of the state distribution and the importance weights $w_t^{(l)}$ reflect the value of the state distribution at the l^{th} particle's location. An estimate of the state at time t is given by:

$$\mathrm{E}\left[\mathbf{x}_t\right] \approx \frac{\sum_{l=1}^{L} w_t^{(l)}\, \mathbf{x}_t^{(l)}}{\left\| \sum_{l=1}^{L} w_t^{(l)}\, \mathbf{x}_t^{(l)} \right\|_2} \ . \tag{4.44}$$

This is the weighted maximum-likelihood estimator for the mean of a vMF distribution and appears in the EM algorithm for fitting a mixture of vMFs (see Section 2.3.4). Equation 4.44 is best calculated before resampling as this gives a lower variance.[1]

[1] Resampling can only increase uncertainty in the particle set.

The vMFPF [61] is summarized in Algorithm 11 and proceeds as follows. At time $t-1$, we have a weighted set of L particles $\mathbf{X}_{t-1}$. The particles are propagated to time t by exact sampling from the state transition distribution in Equation (4.40). Following this, we evaluate the likelihood according to Equation (4.41) that each new particle explains the observed data and update the particle weights. Finally, a new batch of L particles $\mathbf{X}_t$ is sampled i.i.d. from the propagated, re-weighted set to avoid degeneration. This replicates particles that represent high-density regions of the filtered state distribution.

4.2.2 Factorial von Mises-Fisher particle filter (FvMFPF)

We now extend the vMFPF to track multiple sources assuming, as in the FWKF, that the dynamics are identical across all the systems. The major difference is that we are keeping track of a particle-based representation of the state distribution rather than means and covariances. In each time step, we monitor a MovMF and update its parameters as the particle set evolves. Because of this, there is are *two* assignment ambiguities to deal with. The first comes from the fact that we observe the measurements as an unordered set. And the second arises because it is unclear how the particles are associated to clusters in the MovMF. Both ambiguities are death with in a probabilistic fashion by introducing posterior assignment probabilities into the filter equations. Rather than derive the filter equations at length, we simply present and describe them here and refer the reader to the literature on filtering with mixtures [46], [47], [48], [66].

The FvMFPF is summarized in Algorithm 12 and proceeds as follows. We first propagate the particles *and* clusters forward via Equation (4.40). It is infeasible to propagate the state distributions exactly, but we can approximate this step by updating the mixture parameters with one iteration of EM. Parameter estimates from the previous time step are used to initialize EM and the propagated particles are treated as data. This is enough to update the MovMF. The E step provides posterior probabilities

$$\eta_j^{(l)} = P\left(z^{(l)} = j \,|\, \mathbf{x}^{(l)}\,;\, \boldsymbol{\mu}_j^-, \kappa_j^-, \pi_j^-\right) \quad , \tag{4.45}$$

that indicate the degree to which the particles are responsible for representing the clusters. A second set of posterior probabilities captures how each observation is associated to each cluster:

$$\lambda_{jm} = P\left(z_m = j \,|\, \mathbf{y}_m\,;\, \boldsymbol{\mu}_j^-, \kappa_w, \pi_j^-\right) \quad . \tag{4.46}$$

The following steps use the two sets of posteriors to derive a composite likelihood $\beta^{(l)}$ that each particle can explain the entire set of observations. We first evaluate the likelihood $\gamma_m^{(l)}$ that each particle explains each observation. Then, a weighted geometric mean $\xi_j^{(l)}$ of the likelihoods is formed with weights λ_{jm} for each particle-cluster pair. This ensures that only the observations most associated with a cluster are used to update the weights of similarly associated particles. Equivalently, we are calculating a weighted arithmetic mean of the log likelihoods, $\ln(\gamma_m^{(l)})$.

The weighted geometric means are then normalized to derive the probabilities $\bar{\xi}_j^{(l)}$. This ensures that each cluster has a chance to "vote" for the particles that represent it. Without this step, the particle set immediately degenerates towards explaining only one cluster. Finally, we form a weighted arithmetic mean of the $\bar{\xi}_j^{(l)}$'s, where the weights are the posteriors $\eta_j^{(l)}$, to yield the composite likelihood $\beta^{(l)}$. These probabilities are used to update the particle weights as in the vMFPF.

Algorithm 12 Factorial von Mises-Fisher particle filter

Predict

$\mathbf{x}_t^{(l),-} \sim vMF\left(\mathbf{x}_{t-1}^{(l)},\, \kappa_v\right)$

EM update

$\left[\boldsymbol{\mu}_t^-, \boldsymbol{\kappa}_t^-, \boldsymbol{\pi}_t^-, \boldsymbol{\eta}_t\right] = \text{EM\_vMF}\left(\mathbf{x}_t^{(:),-}, \boldsymbol{\mu}_{t-1}, \boldsymbol{\kappa}_{t-1}, \boldsymbol{\pi}_{t-1}\right)$

Correct

$\lambda_{t,jm} = \frac{vMF\left(\mathbf{y}_{t,m}\,;\, \boldsymbol{\mu}_{t,j}^-, \kappa_w\right)\pi_{t,j}^-}{\sum_{j=1}^{K} vMF\left(\mathbf{y}_{t,m}\,;\, \boldsymbol{\mu}_{t,j}^-, \kappa_w\right)\pi_{t,j}^-}$

$\gamma_{t,m}^{(l)} = vMF\left(\mathbf{y}_{t,m}\,;\, \mathbf{x}_t^{(l),-}, \kappa_w\right)$

$\xi_{t,j}^{(l)} = \left[\prod_{m=1}^{K}\left(\gamma_{t,m}^{(l)}\right)^{\lambda_{t,jm}}\right]^{\frac{1}{\sum_{m=1}^{K}\lambda_{t,jm}}}$

$\bar{\xi}_{t,j}^{(l)} = \xi_{t,j}^{(l)} \,\Big/\, \sum_{l=1}^{L} \xi_{t,j}^{(l)}$

$\beta_t^{(l)} = \sum_{j=1}^{K} \eta_{t,j}^{(l)}\, \bar{\xi}_{t,j}^{(l)} \,\Big/\, \sum_{j=1}^{K} \eta_{t,j}^{(l)}$

$w_t^{(l),-} \propto w_{t-1}^{(l)}\, \beta_t^{(l)}$

Resample

$\mathbf{x}_t^{(l)} \sim \sum_{m=1}^{L} w_t^{(m),-}\, \delta\left(\mathbf{x} - \mathbf{x}_t^{(m),-}\right)$

$w_t^{(l)} = 1\,/\,L$

EM update

$[\boldsymbol{\mu}_t, \boldsymbol{\kappa}_t, \boldsymbol{\pi}_t] = \text{EM\_vMF}\left(\mathbf{x}_t^{(:)}, \boldsymbol{\mu}_t^-, \boldsymbol{\kappa}_t^-, \boldsymbol{\pi}_t^-\right)$

From the propagated and re-weighted particle set, we resample a fresh batch $\mathbf{X}_t$ of L particles. And finally, one more iteration of EM adapts the mixture parameters.[2]

4.2.3 Discussion of FvMFPF

In this section, we make several observations about the behavior of the FvMFPF. A convenient theoretical result is that it reduces to the vMFPF when $K = 1$ and we track with a single vMF. We can verify this by examining the filter equations in Algorithm 12.

Coalesence

As in the FWKF, we can prevent the clusters from sticking together by incorporating a small amount of noise, e.g. $\tilde{\boldsymbol{\mu}}_j \sim vMF\left(\widehat{\boldsymbol{\mu}}_j, 1000\right)$, when $\widehat{\boldsymbol{\mu}}_j^T \widehat{\boldsymbol{\mu}}_{j'} > 0.95$. This threshold corresponds to a rotation of $\pi/10$ in any direction. On the other hand, if the state transition noise is sufficiently large, the inherent randomness built into the particle filter may be enough to prevent coalescence.

[2] This step may be unnecessary in practice since we also run an iteration of EM at the beginning of each iteration.

Hard vs soft assignments

It is instructive to consider the special cases of the FvMFPF when the posterior probabilities $\eta_j^{(l)}$ and λ_{jm} are thresholded with a winner-take-all strategy. In that case, we have assignments of particles to clusters and of observations to clusters, respectively. For a certain application, it may be beneficial to tune the assignment "hardness" by exponentiating the posteriors with a number in $[1, \infty]$ and re-normalizing. This reduces the computational complexity of the filter, but may sacrifice stability. Instability may arise when the clusters collide because the hard assignments tend to be too aggressive. This same effect is observed when the posteriors in EM for fitting a mixture model are thresholded.

Extension to other manifolds

A very convenient aspect of the approach taken in the FvMFPF is that Algorithm 12 can easily be adapted for tracking on a manifold other than $\mathbb{S}^2$. This is done simply by choosing an appropriate mixture model since a particle set can represent any distribution. For example, rotation information can be tracked with a mixture of Gaussians (described next).

Velocity as a rotation

We can include velocity information by tracking the axis about which the state is rotating (see Section 2.2.3 for a review of rotations on the unit sphere). This adds 3 components to the state vector that we can think of as a unit vector scaled by the rotation angle: $\mathbf{r}_t \in \mathbb{R}^3$. If we assume that the position and rotation in the state are independent, we can track the rotation vectors separately with an adaptation of Algorithm 12 that models the state distribution as a mixture of Gaussians (MoG):

$$P\left(\mathbf{r}_t \,|\, \mathbf{y}_{1:t-1}\right) = \sum_{j=1}^{M} \pi_{t,j}\, \mathcal{N}\left(\mathbf{r}\,;\, \boldsymbol{\mu}_{t,j}^r, \Sigma_{t,j}^r\right) \quad . \tag{4.47}$$

The particles $\mathbf{r}_t^{(l)}$ and the estimated rotation states $\widehat{\boldsymbol{\mu}}_{t,j}^r$ dictate how to rotate the particle and cluster mean positions in the *predict* step, respectively. This may lead to better tracking performance in cases where a speaker is moving at an appreciable speed across DOA space, a segment of data is missing, etc. However, this comes at the cost of an increase in state space dimensionality. An alternative approach involves tracking quaternions with a Bingham distribution [107].

4.3 Bayesian tracking with raw IPD features

One issue with the WKF is that it relies heavily on sequential RANSAC to provide high-quality observations. It would be more direct to use the raw IPD data as the observation and deal explicitly with the non-linear transformation between the DOA and IPD line slope spaces via particle filtering. In this section, we will consider the case of tracking on the unit circle, although this approach is easily extended to the unit sphere. We will represent the DOA state with a convenient wrapped distribution such as the von Mises (vM) or wrapped Gaussian (WG). The vM has a simple pdf, but may be more difficult to sample from. On the other

hand, the WG has a more complicated pdf, but is easy to sample from. We (somewhat arbitrarily) choose the vM to derive a von Mises particle filter (vMFPF).

4.3.1 State space model

We can use the IPD data directly as our observation by considering a non-linearity in the measurement equation. The state space model in this case is given as:

$$\theta_t \sim vM(\theta_{t-1}, \kappa_v) \quad , \tag{4.48}$$

$$\delta_{t,fi} \sim vM(\psi(\alpha_i(\theta_t)f), \kappa_w) \quad , \tag{4.49}$$

where the observation $\mathbf{\Delta}_t = [\delta_{t,fi}]$ is a collection of C-dimensional IPD vectors, one for each frequency band, and $\alpha_i(\theta_t)$ is the IPD slope corresponding to the i^{th} microphone pair for a signal at DOA θ_t. With a 2-mic array, $\mathbf{\Delta}_t$ is just a $D \times 1$ vector.

4.3.2 Tracking on the unit circle with a von Mises particle filter (vMPF)

To track a single source, we represent the state (DOA) distribution with a collection of particles. It is worthwhile to note that each DOA particle corresponds to an IPD line in the observation space. So, to some degree, they serve the same purpose as the RANSAC samples (i.e. candidate lines) in the sequential RANSAC approach.

Importance sampling for the von Mises distribution

To perform particle filtering with von Mises-distributed noise, we will need an importance distribution that is sufficiently close to a von Mises. The case of a wrapped Gaussian proposal was discussed in Section 2.2.2. For many practical DOA tracking problems, the sources will not change position too quickly from one frame to the next. This will imply that the values of κ that we care about are large enough that we can sample from the proposal and assume the importance weights are uniform. This already becomes reasonable at $\kappa = 10$ and higher, while the values of κ that we are generally interested in are much higher than this (e.g. 50 and above). We conclude that sampling from a von Mises in this high-κ region for the purposes of particle filtering is as straightforward as sampling from a Gaussian distribution.

von Mises particle filter (vMPF)

The vMPF is summarized in Algorithm 13 and proceeds as follows. At time $t-1$, we have a weighted set of L particles:

$$\mathbf{X}_{t-1} = \left\{ x_{t-1}^{(l)}, w_{t-1}^{(l)} \right\} \quad , \quad l = 1, \ldots, L \ . \tag{4.50}$$

The particles are propagated to the next time step by sampling from the state transition distribution in Equation (4.48). This is accomplished indirectly by importance sampling, from which we also get importance

Algorithm 13 Von Mises particle filter with raw IPD features

Predict

$$\left[x_t^{(l),-}, \zeta_t^{(l)}\right] \sim vM\left(x_{t-1}^{(l)}, \kappa_v\right)$$

Correct

$$\gamma_t^{(l)} = \sum_{f=1}^{D} \prod_{i=1}^{C-1} P\left(\delta_{t,fi}\,;\, \psi\left(\alpha_i\left(x_t^{(l),-}\right) f\right), \kappa_w\right)$$

$$w_t^{(l),-} \propto w_{t-1}^{(l)}\, \gamma_t^{(l)}\, \zeta_t^{(l)}$$

Resample

$$x_t^{(l)} \sim \sum_{m=1}^{L} w_t^{(m),-}\, \delta\left(x - x_t^{(m),-}\right)$$

$$w_t^{(l)} = \tfrac{1}{L}$$

weights:

$$\zeta_t^{(l)} = \frac{P\left(x_t^{(l),-}\,;\, x_{t-1}^{(l)}, \kappa_v\right)}{Q\left(x_t^{(l),-} \,\middle|\, x_{t-1}^{(l)}\right)} \quad , \tag{4.51}$$

where $P(-)$ denotes the true (von Mises) distribution and $Q(-)$ denotes the proposal distribution. Following this, we use Equation (4.49) to evaluate the likelihood:

$$\gamma_t^{(l)} = P\left(\boldsymbol{\Delta}_t \,\middle|\, x_t^{(l),-}\right) \quad , \tag{4.52}$$

that each particle $x_t^{(l),-}$ explains the observed data $\boldsymbol{\Delta}_t$. The particle weights are updated using the importance weights and the observation likelihoods. Finally, we sample a new batch $\mathbf{X}_t$ of L particles i.i.d. from the propagated, re-weighted set.

Tracking multiple sources with a factorial von Mises particle filter (FvMPF)

To track multiple sources, we can extend the vMPF in the same way that we extended the vMFPF to the FvMFPF. In each time step, we monitor a mixture of von Mises distributions and update its parameters as the particle set evolves. This simply involves replacing the distributions and EM updates in Algorithm 12 with the corresponding vM forms.

4.4 Experiments

In this section, we assess the performance of the proposed filtering algorithms. In the case of multiple speakers, we will assume that the sources are moving relatively slowly and that the measurement noise is low enough that the source paths can be differentiated.

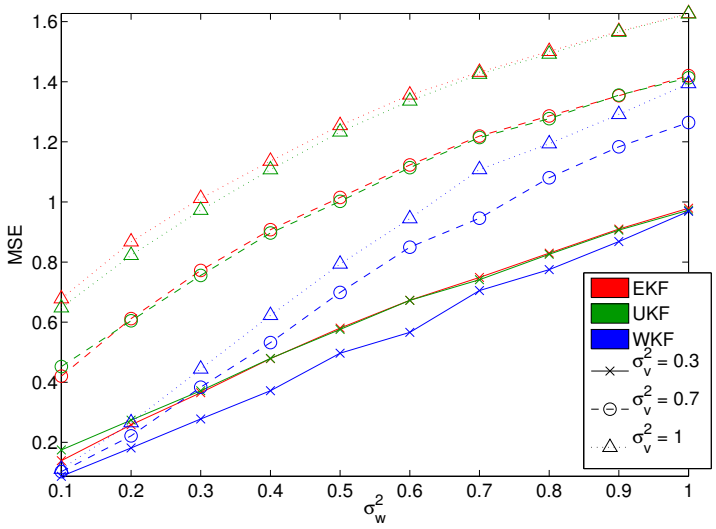


Figure 4.6: MSE for EKF, UKF, and WKF over 5×10^3 length-100 sequences for tracking on the unit circle. The variance of the velocity component was fixed at $\sigma_{v,2}^2 = 0.001$.

4.4.1 Single source tracking on the unit circle

We compared the tracking capabilities of the EKF, UKF, and WKF on the unit circle, estimating position and velocity for all three methods starting from zero initial conditions. Length $T = 100$ state sequences $\mathbf{x}_{1:T}$ were generated according to Equation (4.1) and observation sequences $\mathbf{y}_{1:T}$ for the EKF and UKF were generated according to Equation (4.2).

To ensure a fair comparison, the observation sequences $y_{1:T}$ for the WDS were generated according to Equation (4.4) with σ_w^2 chosen so that the noise levels in Equation (4.2) and Equation (4.4) would be as similar as possible on the unit circle. We did this by fitting a wrapped Gaussian to data that was sampled from a 2D Gaussian with variance $\sigma_{v,1}^2$ and projected to the unit circle. The resulting WG variance was used to generate observations for the WDS.

The mean squared error (MSE) was calculated so as to account for wrapping in the state:

$$\text{MSE} = \frac{1}{T} \sum_{t=1}^{T} \min_{l \in [-\infty, \infty]} \left(\widehat{\mu}_t - \theta_t + 2\pi l \right)^2 \quad . \tag{4.53}$$

The MSEs for several values of the noise parameters are shown in Figure 4.6. If we compare the curves corresponding to $\sigma_v^2 = 0.3$ and $\sigma_v^2 = 0.7$, the WKF's robustness to increasing uncertainty in the state is clear. The WKF suffers a much smaller increase in MSE than the EKF and UKF, especially for low state noise levels.

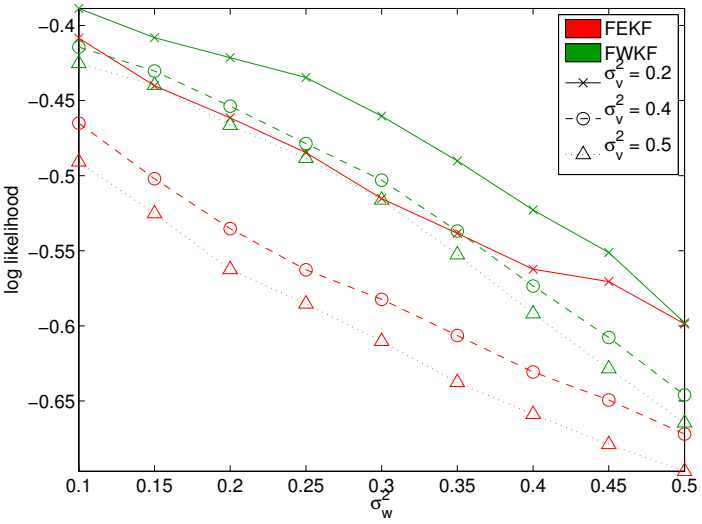


Figure 4.7: Normalized log likelihoods of a factorial EKF method and the FWKF for 2-source tracking on the unit circle. 10^3 length-100 trials were averaged for each noise level pair. The FWKF is more accurate because it infers the state paths via 1D rather than 2D observations. The variance of the velocity component was fixed at $\sigma^2_{v,2} = 0.001$.

4.4.2 Multi-source tracking on the unit circle

Synthetic experiments

This analysis of the previous section is repeated for the FWKF and a multi-source extension of the EKF. The extension is performed using exactly the same measurement association strategy as in the FWKF. A minimum cluster weight of $\frac{1}{10K}$ should be enforced for stability. As with the WKF, we ensure that the noise characteristics are as similar as possible on the unit circle by projecting the measurement distribution in Equation (4.2) to the unit circle and fitting a WG to the result. We can evaluate the tracking performance with a MoWG likelihood for simplicity and to account for wrapping in the state estimates:

$$\mathcal{L} = \left[\prod_{t=1}^{T} \prod_{m=1}^{K} \sum_{j=1}^{K} \frac{1}{K} w\mathcal{N}\left(\widehat{\mu}_{t,j} \,|\, \theta_{t,m}\,,\, 1\right) \right]^{\frac{1}{TK}} . \tag{4.54}$$

Results are summarized in Figure 4.7 and show, once again, that the 1D approach with wrapped Gaussians gives higher accuracy than the 2D Gaussian method.

Azimuthal speaker tracking

Consider the task of tracking multiple speakers on the unit circle with a 3-microphone array. We will use sequential RANSAC to extract the observations in $\mathbb{S}^1$ needed to run the FWKF. These DOA votes are shown in Figure 4.8 for a synthetic mixture of two speakers from the TSP speaker database [81] along with the output of the FWKF. The speakers are moving in opposite directions around the array at a radius of 3.35 meters and with constant speed. The microphones are placed in an equilateral triangle with sides of length 3

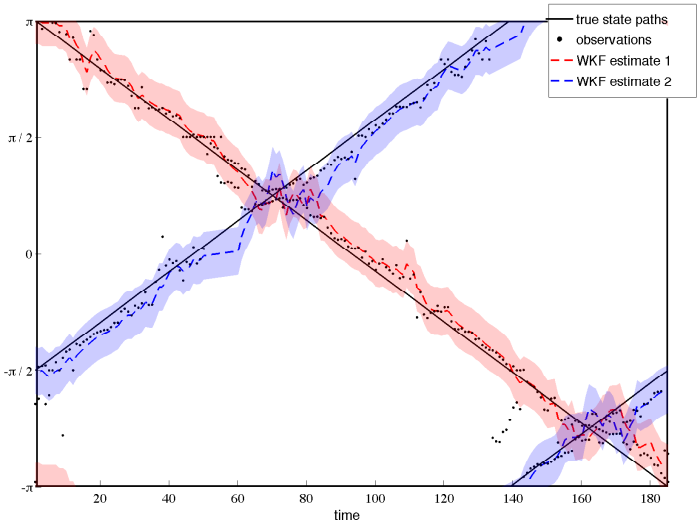


Figure 4.8: DOA tracking with the factorial wrapped Kalman filter (FWKF) for 2 speakers. Adding a small amount of noise during cross-overs prevents coalescence. The FWKF is able to accurately track both speakers.

centimeters. We find empirically that the following parameter settings work well in this case: $\sigma^2_{v,1} = 5 \times 10^{-2}$, $\sigma^2_{v,2} = 10^{-5}$, $\sigma^2_w = 3 \times 10^{-1}$.

We can observe that mistakes are made when a speaker pauses or a DOA estimate is inaccurate (or missing). A gating procedure can be applied to alleviate this [108]. The correct step for the j^{th} cluster is only carried out with data that falls within $\frac{\pi}{5}$ of the cluster mean. If no such data exists, the estimate is only predicted during that time step and not corrected. We refer the reader to the relevant literature for more advanced methods [50], [62], [70], [78].

4.4.3 Multiple speaker tracking on the unit sphere

We can track the DOAs of two speakers on a hemisphere with the FvMFPF and a 3-microphone array placed in the middle of a simulated room. The same array configuration and DOA estimator were used as in Section 4.4.2. As before, gating is applied to reject outliers. Observations not within a rotation of $\frac{\pi}{8}$ in any direction of any cluster mean were considered outliers. The speakers maintained a distance of 3.35 meters from the array. For this case, we find that the following parameter settings work well: $\kappa_v = 1000$, $\kappa_w = 200$, $L = 100$, $\Sigma^r = 10^{-5}\,\mathbf{I}$.

Figure 4.9 depicts a typical output of the FvMFPF. As expected, the tracking results are similar to those observed with the FWKF. Soft assignments of observations and particles to clusters help to spread uncertainty over the clusters during cross-over periods. In addition, the built-in randomness of the SIR procedure prevents coalescence.

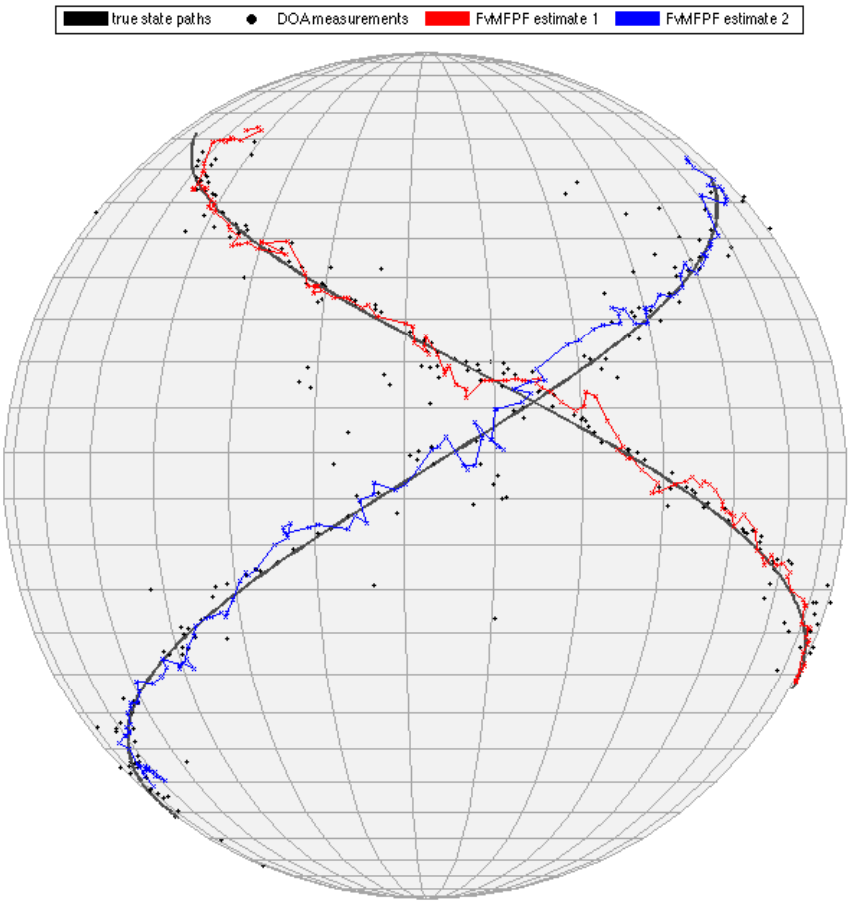


Figure 4.9: DOA tracking with factorial von Mises-Fisher particle filter (FvMFPF) for 2 speakers. Probabilistic assignments of observations/particles to clusters encourages stability in the cross-over region. Coalescence is avoided through the inherent randomness injected by the particle filter.

CHAPTER 5

CONCLUDING THOUGHTS

In this thesis, we developed strategies for tracking and separating multiple speakers with a compact microphone array. The microphones were assumed to be spaced between 1 and 10 centimeters apart. We used the short-time Fourier transform (STFT) to extract inter-channel phase differences (IPD) features from the microphone recordings. These features formed a circular-linear dataset with data scattered around a wrapped line or helix for each speaker. Thus, the blind source separation (BSS) and direction-of-arrival (DOA) estimation problems were reduced to one of multimodal circular-linear regression. A formal approach based on the Expectation-Maximization (EM) algorithm was developed for clustering the IPD features. A faster, more accurate, and more robust solution based on the RANdom SAmple Consensus (RANSAC) algorithm was then introduced.

Wrapped Bayesian filters were developed for tracking the DOAs of the speakers on the unit circle and the unit sphere. The sequential RANSAC algorithm was used to extract short-time DOA votes that served as measurements in the filters. The wrapped Gaussian (WG) distribution was used for tracking a single source on the unit circle with the wrapped Kalman filter (WKF) and the von Mises-Fisher (vMF) distribution was used for tracking on the unit sphere with the von Mises-Fisher particle filter (vMFPF). These were then extended to the multi-source case by modeling the filtered state distribution in DOA space as a mixture. This led to the factorial WKF (FWKF) and factorial vMFPF (FvMFPF). A particle filtering scheme that used raw IPD features as its measurement was also introduced.

There are a number of observations we can make about these techniques. First, the IPD features are a variation on those proposed for the DUET and MENUET algorithms in that they do not break down in the presence of **spatial aliasing**. This occurs when the sampling rate is too high or the inter-mic spacing is too great or both. A stereo set-up with a spacing of 2 centimeters and a sampling rate of 16 kHz will just avoid spatial aliasing. However, for mobile phones, for example, the spacing is typically around 8-10 cm. Thus, it is very important to have a method that can gracefully handle wrapping in the IPD data. That was accomplished in this thesis by using a wrapped linear model of the phase differences across all the frequency bands.

The Bayesian filters proposed here are not complete DOA tracking systems. They are statistically-grounded theoretical methods for tracking directly on wrapped state spaces: the unit circle and sphere. Further steps are required if they are to be integrated into a real-world system. For example, in the experiments with the FWKF and FvMFPF, a **gating** procedure was applied to avoid adapting to nonsensical measurements. In very noisy, real-world conditions, completely erroneous measurements can be fed to the filters that, ideally, should just be ignored. A common way of enforcing this is to attach a confidence level to each measurement. If these are expressed in terms of probabilities, it is straightforward to integrate them

directly into the filter equations. Thus, measurements with high confidence would have the greatest impact on the estimate of the state distribution. A related, conceptually equivalent technique called "soft gating" is the probabilistic extension of the regular gating procedure.

Another issue that is not directly addressed in this thesis is the complicated structure of human conversations. In the real world, speakers typically pause and take turns. This poses an interesting design problem. What sort of generative model could capture this **turn-taking** behavior? We are already taking advantage of a kind of "turn-taking" by assuming that speech does not overlap in a time-frequency representation. But a more sophisticated model could improve the separation and tracking performance by following when the speakers are active or inactive. The classical approach to this, called Multiple Hypothesis Tracking (MHT), maintains several hypotheses about how measurements are associated to target tracks. In this thesis, we borrowed from the Probabilistic Data Association (PDA) literature, which combines observations in a probabilistic way to track multiple states. We could also incorporate birth-death detection techniques so that we can start a new track when a speaker begins talking and delete the track when they finish. The single-sentence speech examples used in the experiments of Chapter 4 mostly avoided all of these real-world issues. However, the probabilistic data associations used in the proposed Bayesian filters allow for a great deal of flexibility in handling noisy data. So, we can imagine that it would not be a great burden to incorporate turn-taking and birth-death models.

Reverberation is yet another complication that was only indirectly addressed in this thesis. It is known that de-reverberation is an extremely challenging problem in real-world scenarios. Furthermore, it has quite a destructive effect on IPD features. This is because phase information is very sensitive to changes in the time-domain signal. RANSAC was able to handle reverb with ease because of its natural tendency to reject a large amount of outliers. This makes DOA estimation with IPD features robust to reverberation. However, to form time-frequency masks, we still need to decide what wrapped line (i.e. source) each TF bin belongs to based on its IPD value. Thus, even if we can easily localize the speakers, we might not be able to separate them to an acceptable degree. Some pre- and/or post-processing is necessary in a real-world application.

The methods described in this thesis constitute a step towards a computational auditory scene analysis (CASA) engine. So, their areas of application stretch far beyond the speech-centered use case explored here. Source separation and tracking techniques are often used in robots so that they can understand and interact with their environment. Marine biologists and naval engineers also make use of these techniques, but underwater. We should observe that speech is not the only class of signals that satisfies the disjointness property. For example, animal vocalizations and many mechanical sounds also tend to be separable with time-frequency masking. DOA tracking of multiple targets is a common problem in surveillance, navigation, communications, and many other fields. It would be interesting to see how the methods proposed in this thesis can be applied in all of these areas.

APPENDIX A

VON MISES/VON MISES-FISHER AS A CONDITIONED GAUSSIAN

Von Mises

We can derive the univariate von Mises distribution from a 2-dimensional spherical Gaussian distribution centered on the unit circle (see Figure A.1). To do this, we condition on the unit circle. This requires a change of variables from Cartesian coordinates to polar coordinates. The Gaussian pdf is

$$P_N(\mathbf{x}\,;\,\boldsymbol{\mu},\sigma^2) \propto \exp\left(-\frac{1}{2\sigma^2}\left[(x_1-\mu_1)^2+(x_2-\mu_2)^2\right]\right) \quad ,$$

and the change of variables is

$$x_1=\cos(\theta) \quad , \quad \mu_1=\cos(\nu) \quad , \quad x_2=\sin(\theta) \quad , \quad \mu_2=\sin(\nu)$$

where we have defined θ as the new random variable and ν as its mean direction. Thus,

$$\begin{aligned}
P_N(\mathbf{x} \mid x_1^2+x_2^2=1) &\propto \exp\left(-\frac{1}{2\sigma^2}\left[\left(\cos(\theta)-\cos(\nu)\right)^2+\left(\sin(\theta)-\sin(\nu)\right)^2\right]\right) \\
&\propto \exp\left(-\frac{1}{2\sigma^2}\left[\cos(\theta)^2-2\cos(\theta)\cos(\nu)+\cos(\nu)^2+\sin(\theta)^2-2\sin(\theta)\sin(\nu)+\sin(\nu)^2\right]\right) \\
&\propto \exp\left(-\frac{1}{2\sigma^2}\left[2-2\cos(\theta)\cos(\nu)-2\sin(\theta)\sin(\nu)\right]\right) \\
&\propto \exp\left(\frac{1}{\sigma^2}\cos(\theta-\nu)\right) \\
&= \frac{1}{C}\exp\left(\kappa\cos(\theta-\nu)\right) \quad .
\end{aligned}$$

This is in the form of the von Mises distribution. The concentration parameter κ replaces the Gaussian's inverse variance. Solving for the normalization constant, the von Mises pdf is given as

$$P_{vM}(\theta;\nu,\kappa) = \frac{1}{2\pi I_0(\kappa)}e^{\kappa\cos(\theta-\nu)} \quad ,$$

where $I_0(-)$ is the 0^{th}-order modified Bessel function of the first kind.

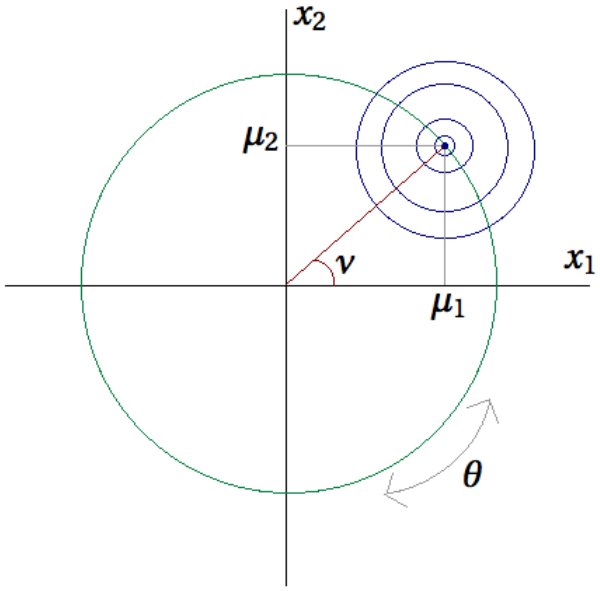


Figure A.1: 2D Gaussian on the unit circle. Conditioning on the unit circle results in the von Mises distribution.

Von Mises-Fisher

We can also derive the von Mises-Fisher distribution, which is the analogue of the vM on the sphere. The Gaussian pdf is

$$P_N(\mathbf{x}\,;\,\boldsymbol{\mu},\sigma^2) \propto \exp\left(-\frac{1}{2\sigma^2}\left[(x_1-\mu_1)^2+(x_2-\mu_2)^2+(x_3-\mu_3)^2\right]\right) \quad .$$

We derive the von Mises-Fisher by conditioning on the sphere:

$$\begin{aligned} P_N(\mathbf{x} \mid x_1^2+x_2^2+x_3^2=1) &\propto \exp\left(-\frac{1}{2\sigma^2}\left[x_1^2-2x_1\mu_1+\mu_1^2+x_2^2-2x_2\mu_2+\mu_2^2+x_3^2-2x_3\mu_3+\mu_3^2\right]\right) \\ &\propto \exp\left(-\frac{1}{2\sigma^2}\left[\left(x_1^2+x_2^2+x_3^2\right)-2\mathbf{x}^T\mu+\left(\mu_1^2+\mu_2^2+\mu_3^2\right)\right]\right) \\ &\propto \exp\left(\frac{1}{\sigma^2}\mathbf{x}^T\mu\right) \\ &= \frac{1}{C}\exp\left(\kappa\mathbf{x}^T\mu\right) \quad . \end{aligned}$$

Solving for the normalization constant, we have that the von Mises-Fisher pdf is

$$P_{vMF}(\mathbf{x}\,;\,\boldsymbol{\mu},\kappa) = \frac{\kappa}{2\pi\left(e^{\kappa}-e^{-\kappa}\right)}\exp\left(\kappa\mathbf{x}^T\mu\right) \quad .$$

APPENDIX B

DERIVATION OF TRIGONOMETRIC DOA ESTIMATORS

In this section, direction-of-arrival estimators are derived using the far-field model of wave propagation. The wavefronts are approximated as planar at the microphone array to simplify the math (this is a common assumption for compact arrays). Microphone 1 is taken as a reference point and is located at the origin. The position of the i^{th} microphone is given as:

$$\mathbf{p}_i = [x_i, y_i, z_i] \quad , \tag{B.1}$$

the distance between mic 1 and mic i is:

$$d_i = \|\mathbf{p}_i\|_2 = \sqrt{x_i^2 + y_i^2 + z_i^2} \quad , \tag{B.2}$$

and the inter-channel time delay (ITD) between mic 1 and mic i is:

$$\Delta t_{1i} = t_i - t_1 \quad , \tag{B.3}$$

where t_i is the time-of-arrival (TOA) from the source to the i^{th} mic. We can convert from a delay in seconds to a delay in samples e_{1i} as follows:

$$\Delta t_{1i} = \frac{e_{1i}}{s} \quad , \tag{B.4}$$

where s is the sampling rate in Hertz.

Localizing on a semicircle with 2 microphones

A source can be localized within π radians on the unit circle. An ambiguity will exist about the axis of the array. From the geometry in Figure B.1(a), we know that:

$$\sin\left(\theta - \frac{\pi}{2}\right) = -\cos(\theta) = \frac{v\Delta t_{12}}{d_2} = \frac{ve_{12}}{d_2 s} \quad , \tag{B.5}$$

so the DOA is given as:

$$\theta = \cos^{-1}\left(-\frac{ve_{12}}{d_2 s}\right) \quad . \tag{B.6}$$

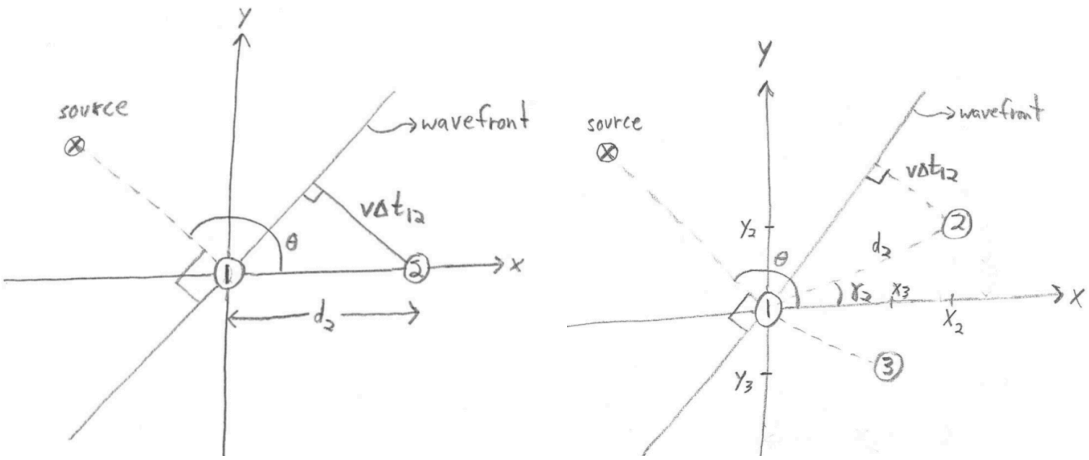


Figure B.1: Geometry of localization on a (a) semicircle and (b) circle.

Localizing on a circle with 3 microphones

We can localize without ambiguities on the unit circle in 2D space with three non-colinear microphones. From the geometry in Figure B.1(b), we know that:

$$\sin\left(\theta - \frac{\pi}{2} - \gamma_2\right) = -\cos(\theta - \gamma_2) = \frac{ve_{12}}{d_2 s} \quad , \tag{B.7}$$

$$\sin\left(\theta - \frac{\pi}{2} - \gamma_3\right) = -\cos(\theta - \gamma_3) = \frac{ve_{13}}{d_3 s} \quad . \tag{B.8}$$

where

$$\tan(\gamma_i) = \frac{y_i}{x_i} \quad . \tag{B.9}$$

Dividing Equation (B.7) by Equation (B.8) and using the identity

$$\cos(\theta - \gamma) = \cos(\theta)\cos(\gamma) + \sin(\theta)\sin(\gamma) \quad , \tag{B.10}$$

we can solve for θ up to a $\pm\pi$ ambiguity:

$$\tan(\theta) = \frac{d_2 e_{13}\cos(\gamma_2) - d_3 e_{12}\cos(\gamma_3)}{d_3 e_{12}\sin(\gamma_3) - d_2 e_{13}\sin(\gamma_2)} \quad . \tag{B.11}$$

To resolve the ambiguity, we add π to θ if the left and right sides of Equation (B.7) disagree in sign.

Localizing on a hemisphere with 3 microphones

We can use an array of 3 microphones to localize in a half-sphere. Consider the configuration in Figure B.2(a). The mics are necessarily coplanar and we choose them arbitrarily to be in the x-z plane. Figure B.2(b) depicts

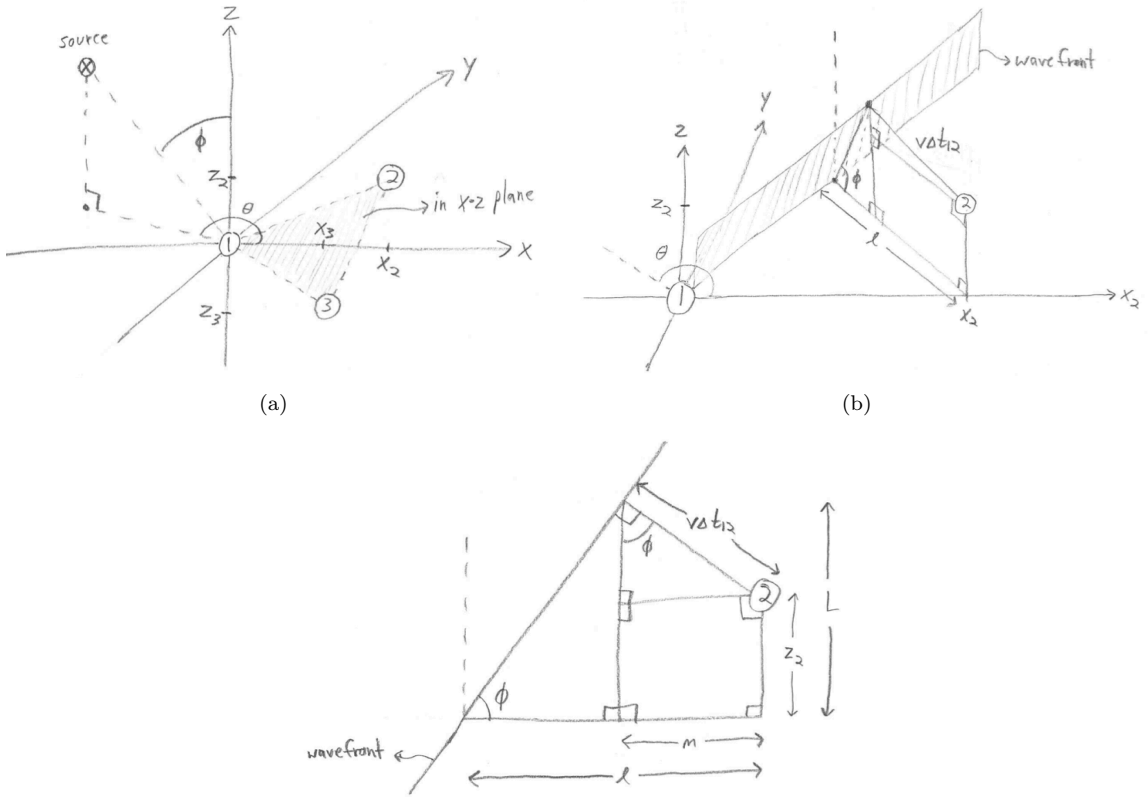


Figure B.2: Geometry of localization on a hemisphere. (a) 3-microphone array in x-z plane and source DOA angles. (b) Geometry of planar wavefront and a pair of microphones. (c) Detail of trigonometry in plane perpendicular to wavefront.

the geometry of the wavefront and one microphone pair while Figure B.2(c) shows relevant detail in the plane perpendicular to the wavefront. From the latter 2 figures, we know that:

$$\sin\left(\theta - \frac{\pi}{2}\right) = -\cos(\theta) = \frac{l}{x_2} \quad , \tag{B.12}$$

$$\cos(\phi) = \frac{L - z_2}{v\Delta t_{12}} \quad , \tag{B.13}$$

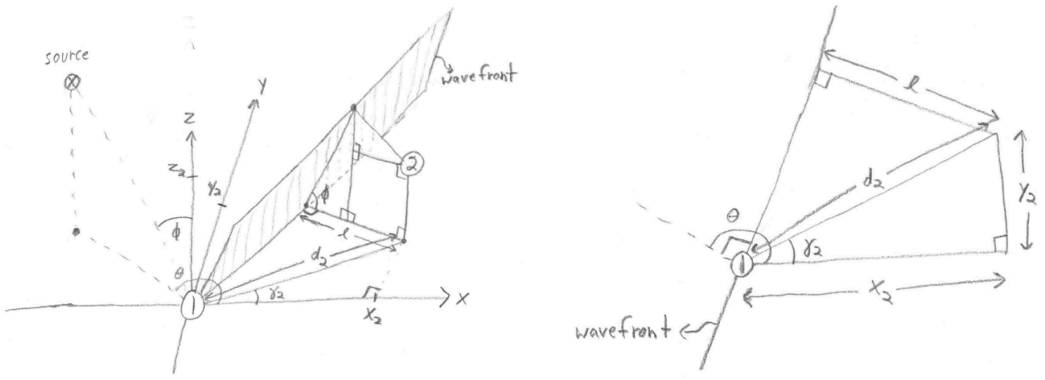
$$\sin(\phi) = \frac{m}{v\Delta t_{12}} \quad , \tag{B.14}$$

$$\tan(\phi) = \frac{L}{l - m} \quad . \tag{B.15}$$

Combining these four equations into one to eliminate l, L, and m, and repeating the process for mic 3, we have that:

$$ve_{12} + sz_2\cos(\phi) = -sx_2\cos(\theta)\sin(\phi) \quad , \tag{B.16}$$

$$ve_{13} + sz_3\cos(\phi) = -sx_3\cos(\theta)\sin(\phi) \quad . \tag{B.17}$$



(a) (b)

Figure B.3: Geometry of localization on the unit sphere. (a) Geometry of planar wavefront and a pair of microphones. (b) Detail of trigonometry in x-y plane.

Dividing Equation (B.16) by Equation (B.17) and solving for ϕ, we have:

$$\cos(\phi) = \frac{v}{s}\frac{x_3 e_{12} - x_2 e_{13}}{x_2 z_3 - x_3 z_2} \quad . \tag{B.18}$$

Substituting Equation (B.18) into Equation (B.17) and solving for θ, we have that:

$$\cos(\theta) = -\frac{v}{s}\frac{z_3 e_{12} - z_2 e_{13}}{x_2 z_3 - x_3 z_2}\frac{1}{\sin(\phi)} \quad . \tag{B.19}$$

Thus, we can solve for both angles by first calculating ϕ and then θ.

Localizing on a sphere with 4 microphones

The most complicated case involves localizing on the unit sphere with 4 non-coplanar microphones. Figure B.3(a) depicts the geometry of the wavefront and a microphone pair. Figures B.2(c) and B.3(b) give a detailed view of the trigonometry in the plane perpendicular to the wavefront and in the x-y plane. From these figures, we know that:

$$\tan(\phi) = \frac{L}{l - m} \quad , \tag{B.20}$$

$$\sin(\phi) = \frac{m}{v\Delta t_{12}} \quad , \tag{B.21}$$

$$\cos(\phi) = \frac{L - z_2}{v\Delta t_{12}} \quad , \tag{B.22}$$

$$\sin\left(\theta - \frac{\pi}{2} - \gamma_2\right) = -\cos(\theta - \gamma_2) = \frac{l}{d_2} \quad . \tag{B.23}$$

where

$$\tan(\gamma_i) = \frac{y_i}{x_i} \quad . \tag{B.24}$$

Combining these four equations into one to eliminate l, L, and m, and repeating the process for mics 3 and 4, we have that:

$$-ve_{12} = d_2 s\cos(\theta-\gamma_2)\sin(\phi) + z_2 s\cos(\phi) \ , \tag{B.25}$$
$$-ve_{13} = d_3 s\cos(\theta-\gamma_3)\sin(\phi) + z_3 s\cos(\phi) \ , \tag{B.26}$$
$$-ve_{14} = d_4 s\cos(\theta-\gamma_4)\sin(\phi) + z_4 s\cos(\phi) \ . \tag{B.27}$$

Dividing Equation (B.25) by Equation (B.26), dividing Equation (B.25) by Equation (B.27), using the property:

$$\cos(\theta-\gamma) = \cos(\theta)\cos(\gamma) + \sin(\theta)\sin(\gamma) \ , \tag{B.28}$$

and re-arranging, we get:

$$\tan(\phi) = \frac{k_1}{k_2\cos(\theta) + k_3\sin(\theta)} \ , \tag{B.29}$$
$$\tan(\phi) = \frac{k_4}{k_5\cos(\theta) + k_6\sin(\theta)} \ , \tag{B.30}$$

where

$$k_1 = z_2 e_{13} - z_3 e_{12} \ , \tag{B.31}$$
$$k_2 = d_3 e_{12}\cos(\gamma_3) - d_2 e_{13}\cos(\gamma_2) \ , \tag{B.32}$$
$$k_3 = d_3 e_{12}\sin(\gamma_3) - d_2 e_{13}\sin(\gamma_2) \ , \tag{B.33}$$
$$k_4 = z_2 e_{14} - z_4 e_{12} \ , \tag{B.34}$$
$$k_5 = d_4 e_{12}\cos(\gamma_4) - d_2 e_{14}\cos(\gamma_2) \ , \tag{B.35}$$
$$k_6 = d_4 e_{12}\sin(\gamma_4) - d_2 e_{14}\sin(\gamma_2) \ . \tag{B.36}$$

Equating Equations (B.29) and (B.30) and solving for θ, we get:

$$\tan(\theta) = \frac{k_2 k_4 - k_1 k_5}{k_1 k_6 - k_3 k_4} \ . \tag{B.37}$$

The resulting value of θ can be substituted into Equation (B.29) to solve for ϕ.

REFERENCES

[1] 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.

[2] A. S. Bregman, *Auditory Scene Analysis.* MIT Press, 1990.

[3] N. Mesgarani and E. F. Chang, "Selective cortical representation of attended speaker in multi-talker speech perception," *Nature*, vol. 485, pp. 233–236, 2012.

[4] P. Common and C. Jutten, *Handbook of Blind Source Separation: Independent Component Analysis and Applications*, 1st ed. Academic Press, 2010.

[5] G. R. Naik, Ed., *Independent Component Analysis for Audio and Biosignal Applications.* InTech, 2012.

[6] P. Smaragdis, "Blind separation of convolved mixtures in the frequency domain," *International Workshop on Independence and Artificial Neural Networks*, 1998.

[7] 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.

[8] 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.

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

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

[11] S. Haykin, *Adaptive Filter Theory*, 3rd ed. Prentice Hall, 1995.

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

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

[14] 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.

[15] 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.

[16] S. Rickard and O. Yilmaz, "On the approximate W-disjoint orthogonality of speech," *IEEE Conference on Acoustics, Speech, and Signal Processing (CASSP)*, vol. 1, pp. 529–532, 2002.

[17] Y. Wang, O. Yilmaz, and Z. Zhou, "Phase aliasing correction for robust blind source separation using DUET," *IEEE Transactions on Signal Processing*, vol. 35, no. 32, pp. 341–349, 2011.

[18] N. Mitianoudis, "A generalized directional Laplacian distribution: Estimation, mixture models and audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 9, pp. 2397–2408, 2012.

[19] C. Kim, C. Khawand, and R. M. Stern, "Two-microphone source separation algorithm based on statistical modeling of angular distributions," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4629–4632, 2012.

[20] 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.

[21] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462 –1469, 2006.

[22] Z. Chen, G. K. Gokeda, and Y. Yu, *Introduction to Direction-of-Arrival Estimation.* Google eBook, 2010.

[23] K. M. Varma, "Time delay estimate based direction of arrival estimation for speech in reverberant environments," Ph.D. dissertation, Virginia Polytechnic Institute and State University, 2007.

[24] 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.

[25] J. Chen, J. Benesty, and Y. Huang, "Robust time delay estimation exploiting redundancy among multiple microphones," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 6, pp. 549–557, 2003.

[26] 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.

[27] S. T. Birchfield and D. K. Gillmor, "Acoustic source direction by hemisphere sampling," *IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 3053–3056, 2001.

[28] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[29] R. Kumaresan and D. W. Tufts, "Estimating the angles of arrival of multiple plane waves," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 19, no. 1, pp. 134–139, 1983.

[30] R. Roy and T. Kailath, "ESPRIT - estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 7, pp. 984–995, 1989.

[31] S. Rickard, T. Melia, and C. Fearon, "DESPRIT - histogram based blind source separation of more sources than sensors using subspace methods," *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 5–8, 2005.

[32] D. Tsuji and K. Suyama, "A moving sound source tracking based on two successive algorithms," *IEEE Symposium on Circuits and Systems (ISCAS)*, pp. 2577–2580, 2009.

[33] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[34] G. Welch and G. Bishop, "An introduction to the Kalman filter," University of North Carolina at Chapel Hill, Tech. Rep., 2006.

[35] S. Gannot and T. G. Dvorkind, "Microphone array speaker localizers using spatio-temporal information," *EURASIP Journal on Advances in Signal Processing*, vol. 2006, pp. 1–17, 2006.

[36] S. Julier and J. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[37] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Transactions on Automatic Control*, vol. 45, no. 3, pp. 477–482, 2000.

[38] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[39] A. Doucet, N. de Freitas, K. Murphy, and S. Russell, "Rao-Blackwellised particle filtering for dynamic bayesian networks," *Uncertainty in AI*, 2000.

[40] X. Zhon, A. B. Premkumar, and A. S. Madhukumar, "Particle filtering for 2-D direction of arrival tracking using an acoustic vector sensor," *19th European Signal Processing Conference (EUSIPCO)*, 2011.

[41] Y. Zhai and M. Yeary, "A new particle filter tracking algorithm for DOA sensor systems," *Proceedings of IEEE Instrumentation and Measurement Technology Conference (IMTC)*, vol. 1, no. 4, pp. 1–3, 2007.

[42] R. van der Merwe and E. Wan, "Sigma-point Kalman filters for probabilistic inference in dynamic state-space models," in *Proceedings of the Workshop on Advances in Machine Learning*, 2003.

[43] X. Zhong and J. R. Hopgood, "Nonconcurrent multiple speakers tracking based on extended Kalman particle filter," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 293–296, 2008.

[44] D. B. Ward, E. A. Lehmann, and R. C. Williamson, "Particle filtering algorithms for tracking an acoustic source in a reverberant environment," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 6, pp. 826–836, 2003.

[45] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Dover Books on Engineering, 2005.

[46] D. L. Alspach and H. W. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximations," *IEEE Transactions on Automatic Control*, vol. 17, no. 4, pp. 439–448, 1972.

[47] R. van der Merwe and E. Wan, "Gaussian mixture sigma-point particle filters for sequential probabilistic inference in dynamic state-space models," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 6, pp. 701–704, 2003.

[48] J. Kotecha and P. M. Djuric, "Gaussian sum particle filtering," *IEEE Transactions on Signal Processing*, vol. 51, no. 10, pp. 2602–2612, 2003.

[49] S. Sarkka, A. Vehtari, and H. Lampinen, "Rao-Blackwellized particle filter for multiple target tracking," *Information Fusion*, vol. 8, no. 1, pp. 2–15, 2007.

[50] W.-K. Ma, B.-N. Vo, S. S. Singh, and A. Baddeley, "Tracking an unknown time-varying number of speakers using TDOA measurements: A random finite set approach," *IEEE Transactions on Signal Processing*, vol. 54, no. 9, pp. 3291–3304, 2006.

[51] V. Cevher, F. Shah, R. Velmurugan, and J. H. McClellan, "A multi target bearing tracking system using random sampling consensus," *IEEE Aerospace Conference*, vol. 1, no. 15, pp. 3–10, 2007.

[52] G.-C. Hsieh and J. Hung, "Phase-locked loop techniques: A survey," *IEEE Transactions on Industrial Electronics*, vol. 43, no. 6, pp. 609–615, 1996.

[53] J. Estrada, M. Servin, and J. Quiroga, "Noise robust linear dynamic system for phase unwrapping and smoothing," *Optics Express*, vol. 19, no. 6, 2011.

[54] H. Nies, O. Loffeld, and R. Wang, "Phase unwrapping using 2D-Kalman filter - potential and limitations," *IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2008*, vol. 4, pp. 1213–1216, 2008.

[55] X. Xie and Y. Pi, "Phase noise filtering and phase unwrapping method based on unscented Kalman filter," *Journal of Systems Engineering and Electronics*, vol. 22, no. 3, pp. 365–372, 2011.

[56] M. Zamani, M.-D. Hua, J. Trumpf, and R. Mahony, "Minimum-energy filtering on the unit circle using velocity measurements with bias and vectorial state measurements," *Australian Control Conference*, 2012.

[57] P. Coote, J. Trumpf, R. Mahony, and J. Willems, "Near-optimal deterministic filtering on the unit circle," *IEEE Conference on Design and Control*, 2009.

[58] K. Mardia and P. Jupp, *Directional Statistics*. Wiley, 1999.

[59] P. Smaragdis and P. Boufounos, "Learning source trajectories using wrapped-phase hidden Markov models," *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 114–117, 2005.

[60] Y. Agiomyrgiannakis and Y. Stylianou, "Wrapped Gaussian mixture models for modeling and high-rate quantization of phase data of speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 4, pp. 775–786, 2009.

[61] F. Zhang, E. R. Hancock, C. Goodlett, and G. Gerig, "Probabilistic white matter fiber tracking using particle filtering and von Mises-Fisher sampling," *Medical Image Analysis*, vol. 13, no. 1, pp. 5–18, 2009.

[62] T. Kirubarajan and Y. Bar-Shalom, "Probabilistic data association techniques for target tracking in clutter," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 536–557, 2004.

[63] S. S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *IEEE Aerospace and Electronic Systems Magazine*, vol. 19, no. 1, pp. 5–18, 2004.

[64] D. E. Sturim, M. S. Brandstein, and H. F. Silverman, "Tracking multiple talkers using microphone-array measurements," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 1, pp. 371–374, 1997.

[65] N. Roman and D. Wang, "Binaural tracking of multiple moving sources," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 4, pp. 728–739, 2008.

[66] Y. Oualil, F. Faubel, M. M. Doss, and D. Klakow, "A TDOA Gaussian mixture model for improving acoustic source tracking," *20th European Signal Processing Conference (EUSIPCO)*, pp. 1339–1343, 2012.

[67] K. J. Molnar and J. W. Modestino, "Application of the em algorithm for the multitarget/multisensor tracking problem," *IEEE Transactions on Signal Processing*, vol. 46, no. 1, pp. 115–129, 1998.

[68] H. Gauvrit, J.-P. L. Cadre, and C. Jauffret, "A formulation of multitarget tracking as an incomplete data problem," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 4, pp. 1242–1257, 1997.

[69] X. Zhong and J. R. Hopgood, "Time-frequency masking based multiple acoustic source tracking applying Rao-Blackwellized Monte Carlo data association," *IEEE 15th Workshop on Statistical Signal Processing*, pp. 253–256, 2009.

[70] J.-R. Larocque, J. P. Reilly, and W. Ng, "Particle filters for tracking an unknown number of sources," *IEEE Transactions on Signal Processing*, vol. 50, no. 12, pp. 2926–2937, 2002.

[71] W. Ng, J.-R. Larocque, and J. P. Reilly, "On the implementation of particle filters for DOA tracking," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSSP)*, vol. 5, pp. 2821–2824, 2001.

[72] T. D. Downs and K. V. Mardia, "Circular regression," *Biometrika*, vol. 89, no. 3, 2002.

[73] 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.

[74] L. Litwic and P. J. Jackson, "Source localization and separation using Random Sample Consensus with phase cues," *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 337–340, 2011.

[75] P. Li and X. Ma, "Robust acoustic source localization with TDOA based RANSAC algorithm," *Emerging Intelligent Computing Technology and Applications: Lecture Notes in Computer Science*, vol. 5754, pp. 222–227, 2009.

[76] C. R. Rao, C. R. Sastry, and B. Zhou, "Tracking the direction of arrival of multiple moving targets," *IEEE Transactions on Signal Processing*, vol. 45, no. 5, pp. 1133–1144, 1994.

[77] A. Srivastava, M. I. Miller, and U. Grenander, "Multiple target direction of arrival tracking," *IEEE Transactions on Signal Processing*, vol. 43, no. 5, pp. 1282–1285, 1995.

[78] M. R. Azimi-Sadjadi, A. Pezeshki, and L. L. Scharf, "Wideband DOA estimation algorithms for multiple target detection and tracking using unattended acoustic sensors," in *Proceedings of the Society of Photographic Instrumentation Engineers*, vol. 5417, 2004, pp. 1–11.

[79] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, 1st ed. Prentice Hall Signal Processing Series, 1989.

[80] 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.

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

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

[83] N. I. Fisher, *Statistical Analysis of Circular Data*. Cambridge University Press, 1993.

[84] N. I. Fisher, T. Lewis, and B. J. J. Embleton, *Statistical Analysis of Spherical Data*. Cambridge University Press, 1993.

[85] W. Jakob, "Numerically stable sampling of the von Mises-Fisher distribution on S2 (and other tricks)," Cornel University, Tech. Rep., 2012.

[86] D. J. Best and N. I. Fisher, "Efficient simulation of the von Mises distribution," *Journal of the Royal Statistical Society, Series C*, vol. 28, no. 2, pp. 152–157, 1979.

[87] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[88] M. T. Heath, *Scientific Computing: An Introductory Survey*, 2nd ed. McGraw Hill, 2002.

[89] G. Ulrich, "Computer generation of distributions on the m-sphere," *Journal of the Royal Statistical Society, Series C*, vol. 33, no. 2, pp. 158–163, 1984.

[90] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[91] D. A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*. Prentice Hall, 2003.

[92] S. Claderara, A. Prati, and R. Cucchiara, "Mixtures of von Mises distribution for people trajectory shape analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 4, pp. 457–471, 2011.

[93] A. Banerjee, I. S. Dhillon, J. Ghosh, and S. Sra, "Clustering on the unit hypersphere using von Mises-Fisher distributions," *Journal of Machine Learning Research*, vol. 6, pp. 1345–1382, 2005.

[94] K. Hornik, I. Feinerer, M. Kober, and C. Buchta, "Spherical k-means clustering," *Journal of Statistical Software*, vol. 50, no. 10, 2012.

[95] M. S. Brandstein and H. F. Silverman, "A robust method for speech signal time-delay estimation in reverberant rooms," *IEEE Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 1, pp. 375–378, 1997.

[96] K. P. Murphy, "Dynamic bayesian networks: Representation, inference and learning," Ph.D. dissertation, UC Berkeley, Computer Science Division, 2002.

[97] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

[98] K. P. Murphy, "Switching Kalman filters," University of British Columbia, Tech. Rep., 1998.

[99] 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.

[100] M. Z. Ikram and D. R. Morgan, "A beamforming approach to permutation alignment for multichannel frequency-domain blind speech separation," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 1, pp. 881–884, 2002.

[101] E. Vincent and R. Laganiere, "Detecting planar homographies in an image pair," *2nd International Symposium on Image and Signal Processing and Analysis*, pp. 182–187, 2001.

[102] Y. Kanazawa and H. Kawakami, "Detection of planar homographies with uncalibrated stereo using distribution of feature points," *British Machine Vision Conference*, vol. 1, pp. 247–256, 2004.

[103] M. Zuliani, C. S. Kenney, and B. S. Manjunath, "The multiRANSAC algorithm and its application to detect planar homographies," *IEEE Conference on Image Processing (ICIP)*, vol. 3, pp. 153–156, 2005.

[104] K. V. Mardia, G. Hughes, C. C. Taylor, and H. Singh, "A multivariate von Mises distribution with applications to bioinformatics," University of Leeds, Tech. Rep., 2007.

[105] J. Allen and D. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Am.*, vol. 65, no. 4, pp. 943–950, 1979.

[106] Q. Gan and C. Harris, "Comparison of two measurement fusion methods for Kalman-filter-based multisensor data fusion," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 1, pp. 273–279, 2001.

[107] J. Glover and L. P. Kaelbling, "Tracking 3-D rotations with the quaternion Bingham filter," MIT - Computer Science and Artificial Intelligence Laboratory (CSAIL), Tech. Rep., 2013.

[108] V. Cevher, R. Velmurugan, and J. H. McClellan, "Acoustic multitarget tracking using direction-of-arrival batches," *IEEE Transactions on Signal Processing*, vol. 55, no. 6, pp. 2810–2825, 2007.