Dipl.-Ing. Hannes Pessentheiner, BSc

Localization, Characterization, and Tracking of Harmonic Sources

With Applications to Speech Signal Processing

Doctoral Thesis
to achieve the university degree of
"Doktor der technischen Wissenschaften"
submitted to
Graz University of Technology

Conducted at the
Signal Processing and Speech Communication Laboratory

Thesis Advisor and Examiner:
Univ.-Prof. Dipl.-Ing. Dr.techn. Gernot Kubin
(Graz University of Technology)

External Examiner:
Prof. Dr.-Ing. Walter Kellermann
(Friedrich-Alexander-Universität Erlangen-Nürnberg)

Graz, January 23, 2017

II

Abstract

A major goal in distant-speech recognition is to transform speech signals of a target speaker into symbols in order to trigger a dialog manager. Spatio-temporal filters, so called beamformers, usually enhance the target speaker's speech signals in a noisy and reverberant environment. However, a beamformer requires information on the target speaker's position. A source localizer provides this information, which facilitates steering a beam into the direction of the target speaker. Unfortunately, the beamformer also captures noise and reverberation, especially from the target speaker's direction. To additionally reduce these artifacts, one can employ bandpass filters in order to emphasize the target speaker's harmonic components. But these bandpass filters require information on the target speaker's fundamental frequency. The problem becomes more challenging in case of two or more target speakers. This is where a joint estimator has to be used.

Two new and intuitive algorithms robustly localize and characterize simultaneously active acoustic harmonic sources intersecting in the spatial and frequency domains. They jointly determine the sources' fundamental frequencies, their respective amplitudes, and their directions of arrival based on a non-parametric signal representation. Variable-scale sampling of unbiased cross-correlation functions facilitates the representation of these three parameters in a joint parameter space. An even better solution is to employ the chirp z-transform, compute the cross-spectrum between pairs of microphone signals, and weight the cross-spectrum's magnitudes by considering a relative phase-delay mask. In both cases, a multidimensional maxima detector sparsifies the joint parameter space. In comparison to alternative approaches based on cross-correlation functions and model-based dictionaries, the new algorithms solve the issue of pitch-period doubling, they cope with one or more harmonic sources, and they associate the determined parameters to their corresponding sources in a multidimensional sparse joint parameter space. State-of-the-art multiple-target trackers, e.g., trackers based on the probability hypothesis density recursion and the multi-Bernoulli recursion, track these parameters over time.

Experiments based on synthetically generated harmonic signals, synthetically filtered speech signals under varying reverberant and noisy conditions, and real recordings yield promising results. A unique, comprehensive multi-sensor Austrian German speech corpus with moving and non-moving speakers provides recordings labeled with spatial and temporal information. This corpus facilitates the evaluation of estimators that jointly determine a speaker's spatial and temporal parameters, including fundamental frequencies.

The joint recall measure, the root-mean-square error, and the cumulative distribution function of fundamental frequencies and/or directions of arrival serve as performance measures. The optimal subpattern assignment distance and its components, e.g., the localization error and the labeling error, serve as a performance measure for the multiple-target trackers.

The evaluations show promising results: On average, both algorithms solve problems, which cannot be solved by their predecessors and other algorithms. The two algorithms outperform existing algorithms in terms of the joint recall measure and the root-mean-square error, and they achieve root-mean-square errors of one Hertz or one degree and smaller, which facilitates, e.g., distant-speech enhancement or source separation for automatic speech recognition. The optimal subpattern assignment distance as well as visualized tracks show that the sparse joint parameter space can be directly fed into a multiple-target tracker yielding smooth tracks.

Index Terms—Chirp z-transform, data association, direction of arrival, fundamental frequency, glottogram, GM-PHD, GM-CPHD, GM-CBMeMBer, joint estimation, microphone array, multiple-target tracking, optimal subpattern assignment, pitch analysis, pitch estimation, pitch-period doubling, position-pitch algorithm, POPI, probability hypothesis density filter, relative phase-delay masking, RPDM, source localization, sparse joint parameter space, speaker separation, speaker tracking, variable-scale sampling, VSS.

Kurzfassung

In einem Bereich der Spracherkennung, der sich mit aus der Distanz aufgenommenen Sprachsignalen beschäftigt, geht es um die Umwandlung von gesprochenen Worten in Symbole. Diese Symbole werden, zum Beispiel, an einen Dialog-Manager weitergeleitet. Mittels räumlich-zeitlicher Filter, sogenannter Beamformer, kann man störende Signale in solchen Aufnahmen dämpfen. Diese Filter benötigen räumliche Informationen über den Sprecher, um dessen Signale in der Aufnahme hervorzuheben. Ein Lokalisierungsalgorithmus stellt diese Informationen zur Verfügung. Ein herkömmlicher Beamformer hebt allerdings Störsignale hervor, die aus der Richtung des Zielsprechers kommen. Um diese Störsignale zu unterdrücken, bedarf es an Bandfilter. Diese Filter benötigen wiederum Informationen über die Grundfrequenz des Sprechers. Schwieriger wird es, wenn mehrere Sprecher gleichzeitig sprechen. In solchen Fällen braucht man Algorithmen, die gemeinsam die räumlichen und zeitlichen Komponenten ermitteln. In meiner Arbeit führe ich zwei Algorithmen ein, die akustische, gleichzeitig aktive, harmonische Quellen lokalisieren und charakterisieren. Diese Quellen können sich sowohl in der räumlichen Domäne als auch im Frequenzbereich überschneiden. Die beiden Algorithmen ermitteln die Grundfrequenzen, die dazugehörigen Amplituden und die Einfallsrichtungen der von den Quellen emittierten Signale bzw. Schallwellen. Diese Ermittlung beruht auf einer nicht-parametrischen Signaldarstellung. Der erste Algorithmus tastet unverzerrte Kreuzkorrelationsfunktionen ab und ermöglicht dadurch die Darstellung dieser Parameter in einem gemeinsamen Parameterraum. Der zweite Algorithmus basiert auf Chirp-z Transformationen, Kreuz-Spektren und deren Gewichtung mittels relativer Phasenlaufzeitmasken. In beiden Fällen kommt ein mehrdimensionaler Maximum-Detektor zum Einsatz, der den gemeinsamen Parameterraum in einen dünn besetzten Parameterraum umwandelt. Im Vergleich zu Modell-basierten und Korrelation-basierten Ansätzen lösen die neuen Algorithmen das Problem der Grundperiodenverdopplung. Zudem sind sie in der Lage, die Parameter von mehreren harmonischen Quellen zu ermitteln. Neueste Tracking-Algorithmen generieren mehrdimensionale, räumlich-zeitliche Trajektorien, die, zum Beispiel, einen Beamformer mit Informationen versorgen. Experimente werden in geräuschvollen, halligen Umgebungen durchgeführt und ergeben vielversprechende Ergebnisse. Ein einzigartiger, neuer Sprachkorpus stellt die für gewisse Experimente notwendigen Signale zur Verfügung. Zur Beurteilung der Algorithmen finden spezielle Maße, wie zum Beispiel die kumulative Verteilungsfunktion, das Recall-Maß, und ein Maß für die Beurteilung der Genauigkeit von Multi-Target Trackern, Verwendung. Die Evaluierungen zeigen, dass die beiden neuen Algorithmen Probleme lösen, die deren Vorgänger nicht lösen können. Zudem schneiden die beiden Algorithmen in vielerlei Hinsicht deutlich besser ab als alternative Ansätze.

Contents

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present doctoral thesis.

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wrtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Dissertation identisch.

Date

Signature

Acknowledgments

I dedicate this thesis to my family: mom and dad, my brother, and, especially, my beloved wife who supported and motivated me from the very first moment of my studies and my doctoral program. I'm thankful for being married with the best, most gifted, and most incredible woman, who literally showed me the world, who always encouraged me to follow my dreams, and who is the most valuable constant in my life.

I'm very grateful for the magnificent, constructive, and groundbreaking discussions with my mentor, Prof. Gernot Kubin, who gave me the chance to start a PhD program in his lab and to acquire indispensable skills, who encouraged me to go abroad and to work in a different lab for almost eight months, who always motivated me in the right moment, who taught me that each single minute is sometimes worth hundreds, and who gave me the opportunity to keep pushing forward some of my personal scientific interest.

"It doesn't work? Katastrophürchterlich! Never mind, let's fix it!" A phrase I was always happy to hear when visiting the lab's system administrators, Markus Köberl and Andreas Läßer, in their office. Both cheered me up when "the pipe was broken," the grid engine was down, the clustered file system failed, too many labmates ran jobs on my computer, and when I crashed my computer (which rarely happened; but when it happened it was always like an apocalypse). Thanks for introducing me to Sun's Grid Engine and Git, and thank you both for your help when I organized the lab's BBQ and Xmas parties.

I especially would like to thank my former labmate Juan Andrés Morales Cordovilla, with whom I was thoroughly working on the voice-controlled home automated system named DIRHA. He was a great collaborator who encouraged me to co-author papers in my early years as an unexperienced PhD student, who suggested questioning the functionalities of the POPI algorithm, and with whom I really enjoyed playing music together.

At this point, I would like to express my gratitude to Tobias Schrank, who taught me the meaning of free software, who encouraged me to use free software, to support free software, to think about the optimal programming language first before starting to solve problems using—guess what—unfree software, to organize code using version control systems, to use more than just one programming language, to think about GNU licenses, and to make resources publicly available. He totally improved my work flow—right, because of the things mentioned before.

A responsible-minded colleague and co-supervisor whom I am very thankful for his supporting moments is Martin Hagmüller, who initially started working on the DIRHA prototype, who initiated the CAPA project, and who mentored me in difficult times when I almost missed the forest for the trees.

I would like to express my gratitude to Prof. Bhaskar D. Rao (he is with the Department of Electrical and Computer Engineering, University of California, San Diego) who provided me an opportunity to join his team as a visiting scientist in 2013 and 2014. I would also like to give thanks to him for all his inspiring discussions and helpful words of advice.

Moreover, I would like to thank M. Gabbrielli for his assistance in analyzing, implementing, and testing the NLS and aNLS algorithms.

Besides many other people, I would also like to say thank you to my former students Fabio Perathoner, Nikolaus Fankhauser, Andreas Fuchs, Elmar Messner, Simon Wasserfall, Julian Linke, and Christoph Emlinger for assisting me in different tasks related to my projects.

Thanks to Alexander Zojer and Lukas Hutter, I figured out the true meaning of the saying "Do whatever you wanna do, as long as it's something good (for you)." and how to map its meaning to everyday life. Both are true friends and, though working in totally different fields, brilliant and marvelous companions that I never wanna miss in my life. Thanks for all our great moments.

Well, during the last few years I read a bunch of acknowledgements. However, I realized that almost all of them abruptly ended after the very last thank-you. And I ensure: this will be the case in my thesis, too! But before doing so, I'm going to share a saying that has made my life much easier, especially in the last few years:

> "There's always a good reason for that! If you don't know it, try hard to find it—even if it seems to be impossible."

And this saying brings me to its origin and my very last thank-you: I would like to thank a former schoolmate—most probably the reason why I decided to look over the rim of a tea cup after finishing business school—for telling me never to give up! *Voilà.*

Preface

As a child I was not that much into TV series; however, there was one series I used to watch with my brother: *Star Trek: The Next Generation.* Besides exploring space and fighting villains, one essential part of the series totally caught my attention: the *USS Enterprise*—a starship cruising through space. Besides splendid warp drives and photon torpedos, there was a unique feature that fascinated me: the starship's communication system. In some episodes, the captain set off the red alert in a corridor. A sweeping alarm tone drowned out the quiet but still noisy soundscape, crew members started running around and discussing with each other, or they even started screaming (depending on the current threats). However, without wearing a head-mounted microphone, the captain communicated with the starship and issued orders executed by the central computer. As a child, I never figured out how it worked. Preliminary (and probably childish) experiments together with my brother showed that even with cordless telephones in a noisy environment (loud music in our children's rooms) a smooth communication was simply impossible (at this time cell phones were far away from being market-ready). Perhaps these experiences were probably the reason why microphones and hands-free communication systems always fascinated me. Years later someone explained me how these systems theoretically work. To my utter astonishment, I realized that they did not properly work in practice; and during my studies I figured out why we are still far away from having such a system in our households. But after reading and publishing my first papers, I noticed that we are getting closer to a system capable of handling natural, interfering noise sources in a real environment. In the end I am very happy to contribute new ideas and algorithms to the field of hands-free communications; may they even be part of answers and solutions to future questions and challenges. And before you start diving into the theory of localizing, characterizing, and tracking speakers (these are typical tasks of such a system), there is just one more thing I want to tell you and which, probably, perfectly closes this preface: *Live long and prosper.*

Chapter 1

Introduction

Athletes know the risks and hazards of sports. They know their limits and how far they can go. However, accidents with disastrous consequences happen even in standard situations. Unfortunately, several athletes suffered severe accidents; they are now paralyzed. What do these athletes have in common besides their accidents and sports? They are strong characters with an iron will to improve their life. However, it is a psychic stress for all of them to have to go without things they were used to, e.g., their independence or their ability to open a window without any help. The fates of athletes that suffered severe injuries as well as the fates of elderly or physically handicapped people triggered my will to help them solving their everyday problems. And this will was one reason why I decided to work on a topic strongly linked to those fates.

As part of a project funded by the European Union, I worked in an international team on the development of a system for voice-controlled home environments and for ambient assisted living. The project named "Distant-speech interaction for Robust Home Applications" (short: DIRHA) [1] addressed the problem of natural, spontaneous communication with an automated home environment and distant-speech controlled interaction with appliances and security services. Especially for elderly or handicapped/disabled people, daily routines can be challenging. Simple tasks such as lowering the blinds of a window, switching on or turning off a room's lights, and making sure that the entrance door is locked usually stretch these people quite a bit. Based on a distant-speech interaction interface, the developed voice-controlled system assists them in everyday life by executing commands issued by an authorized person. In case of ambiguous or unclear commands, the system automatically asks further questions for clarification. This guarantees an extended period of being independent of, e.g., caretakers.

Nowadays, the voice-controlled systems for ambient assisted living—they are not market-ready, yet—feature expensive hardware and lack accuracy and reliability. For instance, reverberation and interfering sources dramatically degrade the system's performance. However, the system has to work accurately when issuing commands, even though other people in the same room acoustically interfere. In case of open windows and vehicles passing by the building, the system has to wake up when calling it. Fig. 1.1, highlights the system's complexity.

The system removes echo echo caused by acoustic feedback, it filters the sampled wave field to remove interfering sources, it classifies the acoustic event by, e.g., distinguishing between speech and noise, it transforms spoken words into symbols subsequently fed

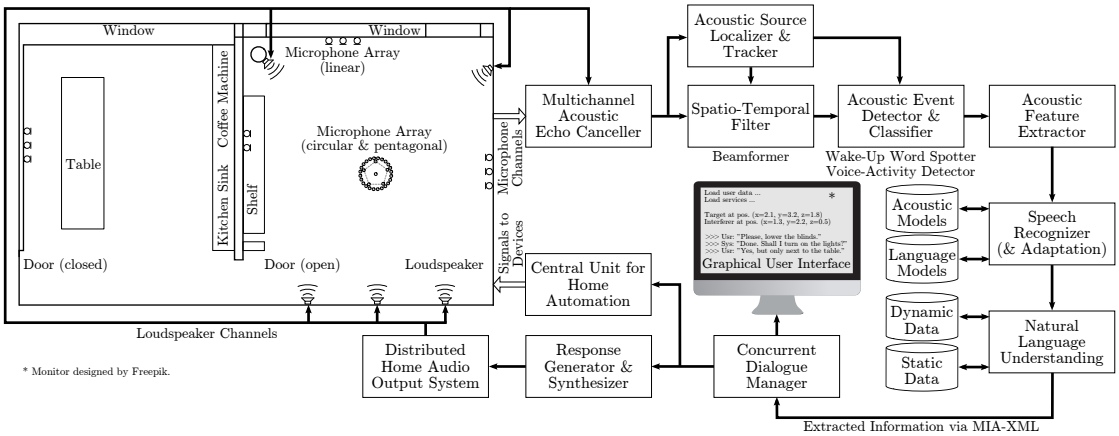


Fig. 1.1: Block diagram showing the basic components of a voice-controlled system for ambient assisted living. For instance, microphones mounted on the ceilings and the walls sample the acoustic wave field. An echo canceller removes echo produced by the system's feedback paths (e.g., the loudspeakers) and feeds the resulting signal into a spatio-temporal filter, e.g., a source localizer or a beamformer. An acoustic event detector and classifier looks out for wake-up words that initiate a dialog. Symbols returned by a speech recognizer facilitates an interaction between the target speaker and the system. And loudspeakers return the system's spoken questions and confirmations.

into a dialog manager, etc.

For sound-field analysis [2–4], sound-field coding [5], and computational auditory [6,7] or acoustic [8] scene analysis, which is part of the system's core tasks, signal parameters often need to be associated with their origin, e.g., a signal-emitting source. To describe such a scene [9–11], I need to detect, localize, characterize, separate, track, and interpret these sources [1, 12]. To localize and characterize them, I jointly estimate multiple parameters to form a joint parameter space and to avoid data association requiring additional algorithms. After tracking these sources I can feed the corresponding tracks into a spatio-temporal filter, e.g., a beamformer. A beamformer requires information on the target speaker's position. A source localizer provides this information, which facilitates steering a beam into the direction of the target speaker. Unfortunately, the beamformer also captures noise and reverberation, especially from the target speaker's direction. To additionally reduce these artifacts, one can employ bandpass filters in order to emphasize the target speaker's harmonic components. But these filters require information on the target speaker's fundamental frequency. The problem becomes more challenging in case of two or more target speakers. This is where a joint estimator—an algorithm that localizes and characterizes the target speaker—has to be used.

Localizing and characterizing harmonic sources as well as joint parameter spaces are a major issue, e.g., in the context of teleconferencing or automatic meeting transcription, or in separating instruments of an orchestral recording [13–36]. The larger the difference of the sources' parameters, the better the separator's performance. For example, if I separate two sources represented in a spectrogram, I still need to find source-independent parameters to succeed. Unfortunately, they are rare and hard to find. However, assuming the fundamental frequency (f_0) as the parameter of my choice, the overlap of a female

and a male speaker's f_0s is small due to anatomical reasons [37]. (The overlap of f_0s is limited; higher harmonics can overlap. In this thesis, I essentially focus on the f_0s.)

What if two speakers with similar f_0 talk simultaneously? Then, I face crossings in the time-frequency domain that decrease the separator's performance. Moreover, they introduce uncertainty about the f_0s' association with the correct source. Another challenging problem is crossings followed by a discontinuous change of intonation. A tracker would need to decide which f_0 corresponds to which source—an ambiguous problem without a distinct solution. But when I extend this lower-dimensional problem to a higher-dimensional one by considering the direction of arrival (DOA), I decrease the number of simultaneous crossings of both f_0 and DOA to a minimum or zero. This will increase the separator's ability to associate the f_0s to their origin.

Localizing sources spatially over time is a well known problem since decades, though, as literature points out [38], it is difficult to solve (especially in reverberant environments). There is a multitude of approaches based on a single microphone array or several distributed microphone arrays assuming plane and/or spherical wave propagation. To narrow down this topic's scope, I will mainly focus on a single uniform linear microphone array and plane wave propagation.

Since audio signals produced by a speaker [39] or a musical instrument [40] (e.g., a wind or a string instrument) feature harmonic structures, I exploit these structures to characterize their origin. Superficially speaking, the harmonic structures' energy is spread over a broad frequency band. But examining these structures in more detail, I identify multiple narrow bands containing information on the harmonic structure [27], which enables us to use, e.g., sparse estimators.

From the system's point of view, there is a multitude of tasks that need to be accomplished. To narrow down the scope, I focus on estimating DOA, f_0, the corresponding harmonics, and the respective amplitudes of harmonic sources.

1.1 Problem Statement

I address the problem of jointly estimating the f_0s, their respective amplitudes, and DOAs of moving and non-moving harmonic sources (as illustrated in Fig. 1.2) by utilizing a non-parametric signal representation; hence, I bypass an explicit statistical estimator. After determining these parameters, I employ different multiple-target trackers to produce smooth spatio-temporal tracks; one track for every harmonic component of each source.

The definition of the signal measured at the i_m-th microphone is

$$x_{i_m}[n_t] = \sum_{i_p=1}^{N_s} \mathcal{H}_{i_m}^{(i_p)}\{s_{i_p}[n_t]\} + \sum_{i_r=1}^{N_r} \mathcal{H}_{i_m}^{(i_r)}\{\nu_{i_r}[n_t]\} \tag{1.1}$$

for $i_m = 1, ..., N_m$ microphones, where $s_{i_p}[n_t]$ denotes a harmonic source's signal and where $\nu_{i_r}[n_t]$ represents the signal of an interfering noise source, which is not correlated with other sources; N_s is the number of the harmonic sources, N_r is the number of the interfering noise sources, n_t is the absolute time in samples, and i_p and i_r are the indices of harmonic sources and interfering noise sources, respectively. The system operator $\mathcal{H}$

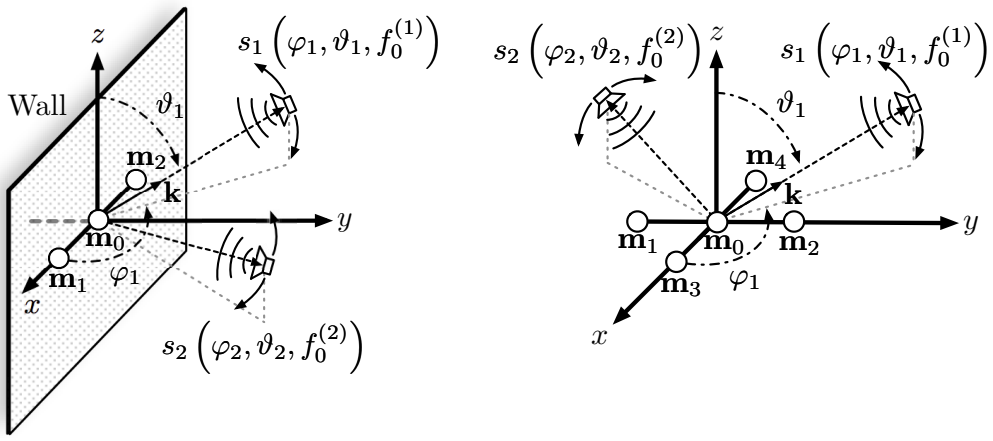


Fig. 1.2: Three-dimensional coordinate systems with two moving harmonic sources, $s_1(\varphi_1, \vartheta_1, f_0^{(1)})$ and $s_2(\varphi_2, \vartheta_2, f_0^{(2)})$, in a reverberant environment (left) and in free field (right). Variables f_0, φ, and ϑ denote the fundamental frequency, the azimuth, and the elevation of a source, respectively; $\mathbf{k}$ is the spherical unit vector, and $\mathbf{m}_i$ represents the microphone coordinates. The black, dashed lines are extensions of the spherical unit vectors. (With their origin in the array's center, these lines point to sources.) The black, dashed-dotted lines represent angles. The gray, dotted lines are projections of the spherical unit vectors' extension on the xy-plane and the xz-plane.

denotes a source's spatialization in reverberant conditions (in case of real-data experiments) and free-field conditions (in case of synthetic-data experiments). For instance, to model a source's movement in free field, I consider the two-dimensional impulse response $h_{i_m}[n_t, n_s] = \delta\left[n_s - \tau_{i_m}[n_t] f_s\right]$, where τ_{i_m} is the time difference of arrival (TDOA), n_s is the time shift, and f_s is the sampling frequency. (Considering a high resolution and rounding $\tau_{i_m}[n_t] f_s$ to an integer number, I kept the quantization error negligible small.) For a harmonic signal $s[n_t]$ sampled at frequency f_s, which sweeps its instantaneous f_0 in a linear or exponential manner from f_1 to f_2 within T_2 seconds and which consists of N_q harmonics with amplitudes α_{i_q}, I write in the linear case

$$s[n_t] = \sum_{i_q=1}^{N_q} \alpha_{i_q} \cos\left(2\pi i_q \left[f_1(n_t/f_s) + \frac{f_2 - f_1}{T_2} \frac{(n_t/f_s)^2}{2}\right]\right) \tag{1.2}$$

and in the exponential case

$$s[n_t] = \sum_{i_q=1}^{N_q} \alpha_{i_q} \cos\left(\frac{2\pi i_q f_1 T_2}{\ln(f_2/f_1)} \left[\left\{\frac{f_2}{f_1}\right\}^{\frac{n_t/f_s}{T_2}} - 1\right]\right), \tag{1.3}$$

respectively, for $0 \leq n_t \leq T_2 f_s$. I omitted the index i_p for simplicity. These practically relevant signals are reasonable models for continuous changes of f_0 in voiced speech or glissandi played by a violinist during a concert. Throughout the article, the DOA is composed of the tuple (φ, ϑ), where φ is the azimuth and ϑ is the elevation.

The questions now are: Do algorithms exist that solve or partially solve the aforementioned problem statement? Is it even possible to extract the DOAs, f_0s, and the

respective amplitudes from one or more harmonic sources from the sampled acoustic wave field?

1.2 Existing Approaches

In the past two decades, several research teams developed approaches to jointly detect or estimate DOAs and f_0s. Based on their publications, I distinguish between two groups.

The first group represents pioneering approaches, in the field of source localization, that estimate and represent both source parameters, the DOAs and the f_0s, separately. For instance, [13] and [14] presented a robust method for speech-signal time-delay estimation in reverberant environments based on estimating f_0s. A different groundbreaking approach for binaural signals builds on multi-pitch tracking [15]. To localize a speaker, the approaches presented in [16, 17] estimate time delays and frequencies of multiple sinusoids.

The second group consists of approaches that jointly estimate and represent parameters in a JPS. In this field, extensive research has been done.

Until 2012, Jesper R. Jensen and some of his colleagues published several groundbreaking papers [18–23, 26] about jointly estimating a harmonic source's DOA and f_0. He summarized most of his findings in terms of a cumulative doctoral thesis entitled "Enhancement of Periodic Signals: With Application to Speech Signals" [41] submitted at Aalborg University Denmark. Though joint estimation was just one part of his thesis, he essentially contributed to this field by introducing algorithms yielding promising results, e.g., the nonlinear least squares estimator and the two-dimensional filtering methods based on the Capon method. He wrote that the joint estimation of parameters is the key to obtain robust and accurate estimators. In fact, the joint estimation is a prerequisite to resolve the DOA and the f_0 of two sources in the following cases: first, both sources share the same DOA but a different f_0; second, both sources feature the same f_0 but a different DOA. In his thesis as well as in one of his papers [21], he proposed two-dimensional filtering methods to jointly estimate the aforementioned parameters of harmonic sources. According to [21], the two dimensions refer to two-dimensional sinusoids, i.e., a complex exponential function that depends on a spatial and a temporal parameter. The simplest method, as he stated, is the periodogram-based method. It employs two-dimensional filters, which pass a signal component with a given temporal and spatial frequency in an undistorted manner. This method assumes white Gaussian noise as an input signal. The second proposed method is based on a generalization of the two-dimensional Capon method; it relies on a single two-dimensional filter. In his filter design, he considered constraints for multiple harmonics and minimized the two-dimensional filter's output power subject to distortionless constraints on the harmonics. In another paper [23], he introduced algorithms based on nonlinear least squares methods. He stated that they are maximum likelihood estimators—they attain the Cramer-Rao bounds—when the noise is white Gaussian, the environment is anechoic, and the target emits its signals in the far-field. In both papers, he mentioned that finding the optimum parameters may imply a huge computational burden, since both algorithms rely on a grid search. A way round the problem is using a gradient search, which can dramatically decrease the computational cost. One of the most recent algorithms is based on a broadband minimum-variance distortionless response beamformer [22]. The authors applied the

algorithm to clean speech signals distorted by the signals of a non-moving interfering harmonic source with five harmonics, a white noise source, and reverberation simulated by using the image method. In [26] they employed a network of unsynchronized uniform linear arrays and maximum-likelihood TDOA-estimators, built on cone-based localization methods and inspired by [23]. Driven by his brilliant ideas, I pursued similar goals but with different approaches. However, to improve my approaches by making them more efficient and accurate in certain scenarios, I had to study and implement his methods in detail. As a result, I conducted experiments with the nonlinear least squares methods and compared their results with the results of experiments where I used my proposed algorithms as well as my algorithms' predecessor, the POPI algorithm.

Similar to Jesper R. Jensen, Ted Kronvall introduced remarkable algorithms to jointly estimate the DOA and the f_0. He summarized them in his doctoral thesis submitted in 2015 at Lund University [42]. Unlike Jesper R. Jensen, Ted Kronvall applied sparse models. All in all, it seems to me that his algorithms [24, 42] are closely related to Jesper R. Jensen's algorithms. However, Ted Kronvall's algorithms consider additional parameters making them more efficient and more accurate. For instance, in [24], they jointly estimated the TDOA and f_0 of two sources by using the alternating direction of multipliers method optimization procedure. In his thesis, he covered several topics related to a signal's spectral content, i.e., grouped line spectra. In comparison to Jesper R. Jensen's algorithms, his algorithms do not need the knowledge of the exact model order, they are based on sparse modeling, they utilize an over-complete dictionary, and they employ the alternating direction of multipliers optimization (ADMM) procedure to solve convex problems. Instead of using the well-known but resource-intensive CVX toolbox, he decided to use ADMM. It solves the problem of estimating line spectra, where the least-squares cost function additionally contains a penalty function to avoid overfitting. Using the ADMM, the algorithms estimate the f_0 by solving a convex optimization problem, then they estimate the DOA by modeling sensor and source positions in the near field and the far field. Consequently, he estimates the parameters successively (and not jointly). In the remaining part of this summary, I will focus on algorithms related to the problem of joint estimation. The first algorithm, named array DOA and pitch estimation using block sparsity (APEBS), jointly estimates the f_0 and the DOA of an unknown number of harmonic sources. It is an approach based on a dictionary learning framework. By using the ADMM, the method alternates between estimating the f_0s using an extension of the sparse group least absolute shrinkage and selection operator (SGL) and learning the phase offsets, which are related to the DOAs. Known as the harmonic audio localization using block sparsity (HALO) estimator, the second algorithm jointly estimates the f_0 and the DOAs from sources positioned in the near field and the far field. Again, he applies the ADMM, which solves a complex problem, by, first, estimating the f_0s and complex amplitudes considering a sinusoidal model, and, second, finding each source's position by utilizing the difference in phase and the relative attenuation of the magnitude estimates.

A pioneering doctoral thesis showing striking parallels with Ted Kronvall's work is the thesis written by Stefan Ingi Adalbjörnsson submitted in 2014 at Lund University [43]. He examined sparse modeling heuristics and applied sparse and robust modeling strategies to various problems related to (joint) parameter estimation, e.g., in the field of audio signal processing, deoxyribonucleic acid sequencing, and spectroscopy. To keep

this summary compact, I reduce its scope to audio signal processing with focus on joint parameter estimation. Similar to Ted Kronvall but unlike Jesper R. Jensen, Stefan I. Adalbjörnsson introduced algorithms that do not require prior knowledge of the model order (the number of a signal's harmonics; a maximum number is sufficient). To improve the efficiency of his algorithms, he assumed signal models featuring sparse properties. For instance, a small set of a large dictionary of possible candidates of Fourier kernels represented as vectors, which contain information about sinusoidal components, forms a signal. In other words, he linked the parameters of interest to a small number of a large dictionary's components. By assuming sparse signal models, he formulated efficient algorithms that accurately reconstruct a signal using non-zero components only yielding unique solutions. He minimized a cost function by employing convex optimization. As Ted Kronvall stated, convex optimization can be resource consuming. Thus, to guarantee an efficient and fast solution in case of convex optimization problems, he employed the framework of the alternating direction method of multipliers (ADMM). This framework solves large-scale optimization problems by increasing the number of variables; it splits a problem into sub-problems. Moreover, the framework utilizes the knowledge of sparse models, which increased the algorithms' efficiency in finding a solution. In his thesis as well as in [44] he presented an algorithm to estimate the f_0 by exploiting (block) sparsities based on an efficient ADMM implementation in case of one or more sources. Although the algorithm, which is known as pitch estimation using l_2-norm and block sparsity including the total variation penalty function, is not a joint estimator, it is at least an important part of the joint estimator described later on. He guaranteed that the algorithm converges and that it is robust enough to cope with the problem of pitch-period doubling. To evaluate the algorithm's performance, he conducted experiments with simulated signals and signals recorded in a real environment. In the latter case, he estimated the f_0s of a guitar recording and a recording featuring a viola and a speaker—both were active simultaneously—focusing on the speaker's voiced parts only. He showed that the algorithm still works if one or more harmonics are missing. In [27] as well as in his thesis, he presented a joint estimator for localizing one or more audio sources in a non-/reverberant environment. The algorithm is based on the aforementioned algorithm and is known as HALO. It localizes harmonic sources employing a generalization of the previously described algorithm and utilizing a limited dictionary containing components representing possible locations and frequencies. These components are necessary to model the phase differences and relative attenuations between each channel's signal. The generalization considers the measurements of all channels. The algorithm properly works for arbitrary but known array geometries, it assumes a sinusoidal signal model, and—which is remarkable—it estimates a source's f_0s and spatial parameters in the near field and the far field over time.

A different approach to jointly estimate and represent f_0s and interaural time-differences is to apply extended recurrent timing neural networks [25]. However, to narrow down the scope, I skip discussing this approach in detail.

What do all reported studies have in common?

First, they did not explain how to solve the data association problem while estimating or detecting the parameters. Although they utilize cross-correlation functions or cross-correlograms, the algorithms described in [13] and [14] estimate the TDOA of a single harmonic source only; the authors ignored estimating the frequency components.

Actually, these algorithms are incapable of estimating frequency components; thus, a data association is unfeasible and, anyway, unnecessary due to a restriction to scenarios with a single harmonic source. However, the authors highlighted the positive aspects of cross-correlation functions. The algorithms presented in [16, 17] estimate both parameters, the TDOA and the f_0. However, these algorithms cope with scenarios where a single harmonic source is active only; their TDOA-estimates depend on prior f_0-estimates (i.e., the TDOA is estimated after the f_0). In both cases, data association is feasible if a single harmonic source is active only. Despite utilizing frequency information, the approach presented in [15] estimates TDOAs only. However, it facilitates estimating the TDOAs of two sources. So far, it is unclear if all these algorithms return reasonable results in case of moving sources. The authors of [14] claimed that it is possible to develop an algorithm that tracks moving harmonic sources; they did not describe how to realize that, though. The algorithm proposed in [18] features data association, but only in case of a single harmonic source. It determines the maximum argument of a cost function based on dictionaries for single-source scenarios only. The same is true for [19, 22, 23, 26]. In [21] the authors claimed that they can estimate the parameters of two sources. They estimated the parameters of two sources by determining the two highest peaks in their JPS; however, as soon as, e.g., the second harmonic of the dominant source is larger then the first harmonic of the second source, the algorithm fails to estimate the parameters of both sources. As a consequence, the data association fails. The authors of [20] presented an ESPRIT-based algorithm that estimates the parameters associated to a single source only. They introduced another algorithm that copes with one or more sources, too, but only in a multipath-free environment. All in all, the algorithms presented in [24, 27] can estimate and associate the parameters of one or more harmonic sources. Although the algorithms determine the maximum argument only, the dictionary (or code book) in the decoding step covers multiple-source scenarios. However, if the grid in the joint parameter space spanned by the (sparsified) dictionary is not fine enough, the RMSEs and Rs dramatically increase. A finer grid means a larger dictionary and, thus, an increased computational complexity. More importantly, the authors ignored the case where the dictionary lacks certain entries; for instance, two active harmonic sources with missing second or third harmonics due to, e.g., interfering noise sources. In [25] the authors presented a different approach based on supervised learning and recurrent timing neural networks. The authors claim that their approach can estimate the parameters of two or more sources; however, they just showed that their approach works for a single harmonic source, i.e., data association is unfeasible in case of two or more sources.

Second, all proposed algorithms (except the one in [20]) did not span a joint parameter space or did not sparsify the joint parameter space to decrease the amount of data to be processed. None of the algorithms proposed in [13–17] span a parameter space, which is a major reason why these algorithms except [15] are unable to estimate the parameters of two or more harmonic sources. In [15] it is unclear how the authors extracted the arguments (i.e., the TDOAs of the impinging plane waves) of the resulting histogram contours. Neither the proposed approaches in [18, 19] nor the algorithms presented in [21–23, 25, 26] span a SJPS due to the restriction of estimating the parameters of a single harmonic source only. In [24, 27] the authors presented SJPSs of the DOAs and the magnitudes only, although their algorithms are theoretically capable of spanning a SJPS with frequency components.

Third, all studies did not consider any (multiple-target) trackers to form smooth spatio-temporal tracks of the sources' harmonic components over time. Neither the approaches of the first group, [13–17], nor the algorithms of the second group, [18–27], employ a (multiple-target) tracker to obtain smooth spatio-temporal trajectories. Tracking is important to reduce the number of clutter (caused by multipath components or interfering sources). In case of, e.g., multiple-target and sub-band beamforming, the decrease in clutter by employing trackers and feeding the smoothed tracks into a beamformer results in a reduced number of steered beams in different directions. This decrease in clutter dramatically reduces, e.g., the amount of computational resources.

Fourth, many approaches rely on estimating the global extremum of a cost function, which, in most cases, means that they are able to detect or estimate the parameters of a single source (at a certain instant of time) only. In [13] the authors compute the minimum argument, e.g., of a generic or weighted least-squares cost function, where a peak corresponds to the most likely parameters of a single harmonic source only. As a consequence, distinguishing between two simultaneously active harmonic sources or estimating the parameters of two or more sources simultaneously is impossible. The algorithms proposed in [16, 17] estimate a single TDOA per instant of time. Moreover, they fail to assign extracted parameters to a certain source. Neither [14] nor [15] describe how they estimate/extract the TDOAs in case of a multi-modal cost function. However, in [14] the authors presumably estimate the TDOAs by determining the maximum argument of a cross-correlation function. As indicated earlier, the approaches presented in [18, 19, 21–23, 26] rely on estimating the maximum argument or the minimum argument; they ignored extending their dictionaries to scenarios with two or more harmonic sources. Unlike [18, 19, 21–23, 26] the authors of [24, 27] considered extended dictionaries, and the proposed approaches presented in [20, 25] bypass estimating the maximum argument or minimum argument of a cost function.

Fifth, most of the authors focused on testing their approaches on signals produced by musical instruments or on a very small number of speech signals. For instance, the authors in [16] conducted experiments with a small set of synthetically generated sinusoidal signals and with a portion of a Cantonese word. In [17], the authors evaluated their proposed algorithm by complex sinusoids, a synthesized diphthong, and a portion of a recorded vowel. In [14] they considered a set of speech signals of non-moving sources recorded with spatially distributed microphones. The authors in [13] conducted experiments with synthetically spatially filtered segments of sampled speech. The authors in [21, 24] used synthesized harmonic signals only to evaluate their proposed algorithms. In [25] they solely conducted experiments with spoken digits. The experiments described in [19, 26, 27] covered scenarios with synthesized harmonic signals as well as a very small number of experiments with the sentence "Why were you away a year, Roy?". In [18, 20, 22, 23] the authors conducted experiments with synthesized harmonic signals and spatially filtered or recorded trumped signals. The big advantage in terms of a trumpet signal is the constant and dominant tone over a long period of time.

Bringing it all together, the existing approaches either fail to jointly estimate the DOAs, the f_0s, and the respective amplitudes of two or more harmonic targets (due to their missing data association for two or more targets) or fail to estimate the targets' higher harmonics (which are indispensable when the f_0 is distorted by an interfering source). Some approaches fail in both cases. Moreover, all authors omitted a target

tracker; as a consequence, they failed to show that they can successfully feed a tracker with the estimates provided by their estimators. All publications lack experiments with a comprehensive set of naturally spoken sentences that consist of voiced and unvoiced parts as well as breaks. The authors failed to conduct experiments with sentences spoken by a male and female speaker in a real reverberant environment. All these issues led me to innovative real-time capable solutions for two or more sources based on [45] and tested on synthetic data and speech data recorded in a real reverberant environment. The solutions are based on approaches and findings published in the recent years starting in 2007.

1.3 The Predecessors' Roadmap

Képesi et al. [45, 46] introduced the idea of jointly estimating and representing both parameters in an SJPS in 2007 by means of extracting certain features from a biased CCF using two microphones only. Until 2013 several studies extended this idea.

The doctoral thesis of Tania Habib [47] submitted in 2011 at Graz University of Technology was a corner stone of my work. She analyzed auditory inspired methods for localizing and tracking one or more speakers using a 24-element uniform circular microphone array [35, 47]. Moreover, she elaborated on a source localizer named POPI algorithm [45], which requires two or more microphones and which spans a so called POPI plane. This plane consists of three types of parameters: DOAs, frequencies, and color-coded amplitudes. As a result of elaborating on several combinations, she introduced five enhanced versions of the predecessor [45] in order to localize acoustic sources. The first one was a multi-microphone position-pitch algorithm. This was basically the algorithm presented in [45] extended by additional microphones (up to 24) spanning a uniform circular microphone array. The array's circular shape featuring an equidistant microphone spacing was necessary to arrange the microphones in pairs. Each pair shared the same reference point—the intersection of each pair's connecting line (or base line)—which was a prerequisite to combine the POPI planes of each pair. Her second algorithm employed a cepstrum-based weighting function to suppress cross-terms introduced by the cross-correlation function. However, she omitted this kind of weighting in her succeeding algorithms, because it introduced errors when two sources were active simultaneously. Extending her predecessors by a gammatone filter bank yielded the third algorithm: the multiband position-pitch (MPOPI) algorithm. It was capable of localizing one or more speakers. The full-band approaches—the previously mentioned algorithms—succeeded in scenarios with a single speaker only. She additionally introduced the frequency selection-based multiband position-pitch (MPOPI-FS) algorithm, her fourth algorithm, which only considered bands carrying a speaker's signal. Thus, she was able to improve the algorithms accuracy by reducing interfering non-harmonic artefacts. Last but not least, she extended the MPOPI-FS by an additional module [15, 48], which was working in parallel. This module computed spectro-temporal fragments used to improve the frequency selection. She called the resulting algorithm the spectro-temporal fragment-based MPOPI (MPOPI-STF) algorithm. To evaluate all these algorithms, she additionally set up a corpus containing recordings of real speakers' signals played back via loudspeakers in a reverberant room. Bringing it all together, she extended [45] by adding modules and filters inspired by auditory models of the human

inner ear. She showed that utilizing these modules and filters improve the localization accuracy by taking advantage of temporal information encoded in the cross-correlation function.

However, after reviewing them and conducting a vast number of experiments, I came to the conclusion that several modifications reported in [28–36, 45, 47] were unfavorable for the aforementioned problem scenario. For instance, they estimated DOAs only and did not exploit their algorithms' (hidden) abilities to estimate f_0s. They considered broadband analysis in case of two or more sources which yielded accurate directional information but erroneous temporal information, as shown in my experiments. They analyzed summed CCFs, which introduced pitch-period doubling. They employed biased CCFs yielding estimates with varying amplitudes for signals whose sinusoidal components exhibit the same amplitude. The application of gammatone bandpass filters caused distorted estimates due to varying gains within a band and a missing group delay compensation. They did not consider a sparse representation of their estimates, which could have been directly fed into, e.g., a tracker. They also considered spectro-temporal fragments analysis [15, 48] and combined their existing algorithm with a spectro-temporal pre-processing module yielding a dramatic increase in computational costs.

In this thesis, I introduce a subclass of the aforementioned second group which is composed of my two algorithms that jointly estimate and represent both parameters in an SJPS. I sparsify the JPS by employing a multidimensional maxima detector, which facilitates estimating two or more harmonic sources. In [45, 46], the authors suggested the idea of joint estimation and representation in an SJPS obtained by sampling a CCF. It is the cornerstone of the two proposed algorithms, i.e., the VSS-based algorithm and the RPDM-based algorithm.

1.4 Research Questions

The drawbacks and issues of the alternative approaches and the predecessors led me to the following research questions, which I will address in my thesis:

- How to redesign the POPI-algorithm in order to jointly estimate the DOAs, the f_0s, and their respective amplitudes of one or more harmonic sources? Is it necessary to start designing the algorithm from scratch?
- How to sparsify the joint parameter space of the proposed algorithms (which features more dimensions than the POPI-plane) in order to directly feed the estimates into a tracker?
- How to solve the problem of pitch-period doubling using cross-correlation functions and cross-spectra?
- How to jointly estimate the sources' parameters without requiring prior knowledge about the model order and the number of simultaneously active harmonic sources?
- How good are the estimates of the proposed estimators, the predecessor, and a state-of-the-art joint estimator in case of utterances that feature voiced and unvoiced parts as well as breaks?

- Are there any suitable speech corpora available in order to conduct experiments with joint estimators?
- Can I apply multiple-target trackers to the estimates of the proposed algorithms and, if yes, which multiple-target tracker should be preferred?
- Are there any unknown phenomena that should be considered when jointly estimating the parameters of harmonic sources in reverberant environments?

1.5 Contributions and Innovations

In this thesis, I introduce two innovative algorithms to localize and characterize one or more harmonic sources in free field and in reverberant environments. By utilizing a non-parametric signal representation, I bypass employing explicit statistical estimators. Both algorithms compute a (quasi-continuous) JPS and sparsify it, which yields a SJPS. A SJPS contains relevant information, i.e., estimates. These estimates can be directly fed into a multiple-target tracker. The proposed algorithms solve the problem of pitch-period doubling. They jointly determine the parameters of harmonic sources without requiring prior knowledge about the model order and the number of simultaneously active harmonic sources. I introduce a unique, comprehensive speech corpus that features glottograms and recordings labeled with the speakers' f_0s and spatial information. These labels are a prerequisite to conduct experiments with algorithms that jointly estimate the parameters of harmonic sources. For the very first time, I conducted experiments with recordings of spoken sentences that consist of voiced and unvoiced parts as well as breaks. And last but not least, I highlight and discuss an unexpected phenomenon in reverberant environments when employing joint estimators.

1.6 Summary by Chapters

In this thesis, I address the problem of jointly localizing, characterizing, and tracking one or more harmonic sources in different acoustic environments.

In Chapter 2 I describe my first approach based on cross-correlation functions and variable-scale sampling to solve the aforementioned problem. Moreover, I highlight its innovations and its contributions in that specific field of research. Additionally, I thoroughly explain the utilized metrics, I describe and discuss the conducted experiments and their outcomes, and I compare the results with the outcomes of alternative approaches.

In Chapter 3 I introduce another approach—I preferably call it the first approach's successor—based on cross-spectra and relative phase-delay masking. It bypasses drawbacks of its predecessor, which yields better results in terms of accuracy. I explain the experimental design and experimental results followed by a detailed discussion.

Supposing that both approaches' estimates can be directly fed into a tracker, I employ state-of-the-art multiple-target trackers to generate accurate and smooth trajectories in a parameter space. In Chapter 4 I describe the trackers, specific metrics that compare estimated tracks and ground-truth tracks, and experiments as well as their outcomes.

Chapter 5 contains information on a unique, comprehensive Austrian German multi-sensor corpus, which I had to set up to produce meaningful results for the thesis's problem

statements. Besides recording procedures, post-processing, and quality assurance, I present, e.g., a speaker's spatial trajectories and glottograms.

In Chapter 6 I briefly highlight the advantages and disadvantages of the authors' approaches followed by a discussion of their influential doctoral theses to position my work in that field of joint estimation of DOAs and f_0s. I finish this chapter by concluding my thesis and by sharing some of my ideas and open questions.

Afterwards I highlight other achievements during my doctoral program, which are related to spatio-temporal filtering methods and microphone arrays, in the appendix.

Chapter 2

Joint Estimator Based on Variable-Scale Sampling[1]

In this chapter, I thoroughly describe my first approach based on variable-scale sampling [49] to localize and characterize one or more harmonic sources; Fig. 2.1 shows the approach's block diagram. I will discuss its components, the experiments designed for evaluating the algorithm's performance, the corresponding results, as well as the utilized metrics in the remaining part of this chapter. Before doing so, I highlight the contributions and innovations introduced by this approach in the field of jointly estimating DOAs and f_0s.

2.1 Contributions and Innovations

The proposed algorithm localizes and characterizes one or more simultaneously active harmonic sources in free field and in reverberant environments. It is based on [45] and inspired by [28–36]. In contrast to its predecessor, the proposed algorithm additionally determines f_0s and their respective amplitudes but features fewer components. In comparison to [13–25, 28–36, 45], the algorithm sparsifies a (quasi-continuous) JPS and determines parameters of harmonic sources without utilizing an explicit statistical estimator. Utilizing unbiased CCFs, considering bandpass filters that feature a manageable flat passband, processing each band separately, doing narrow-band analysis, employing variable-scale sampling, and representing the estimates in a SJPS, the algorithm solves the problem of pitch-period doubling. It jointly determines the parameters of harmonic sources without requiring prior knowledge about the model order, i.e., the number of harmonics, and the number of simultaneously active harmonic sources. Beyond that, I conducted a vast number of experiments with simultaneously active, synthetically generated sources featuring non-stationary harmonic signals causing intersections in the spatial and frequency domains. To conduct experiments with recorded speech signals from male and female speakers in a real reverberant environment, I compiled a unique speech corpus [50, 51], which contains recordings of spoken sentences that contain voiced

[1] This chapter is substantially based on the journal paper [49] and was revised and adapted to the present thesis. As first author of the journal paper, I did everything on my own except the implementation of the aNLS algorithm and the NLS algorithm [23] implemented by Mattia Gabbrielli.

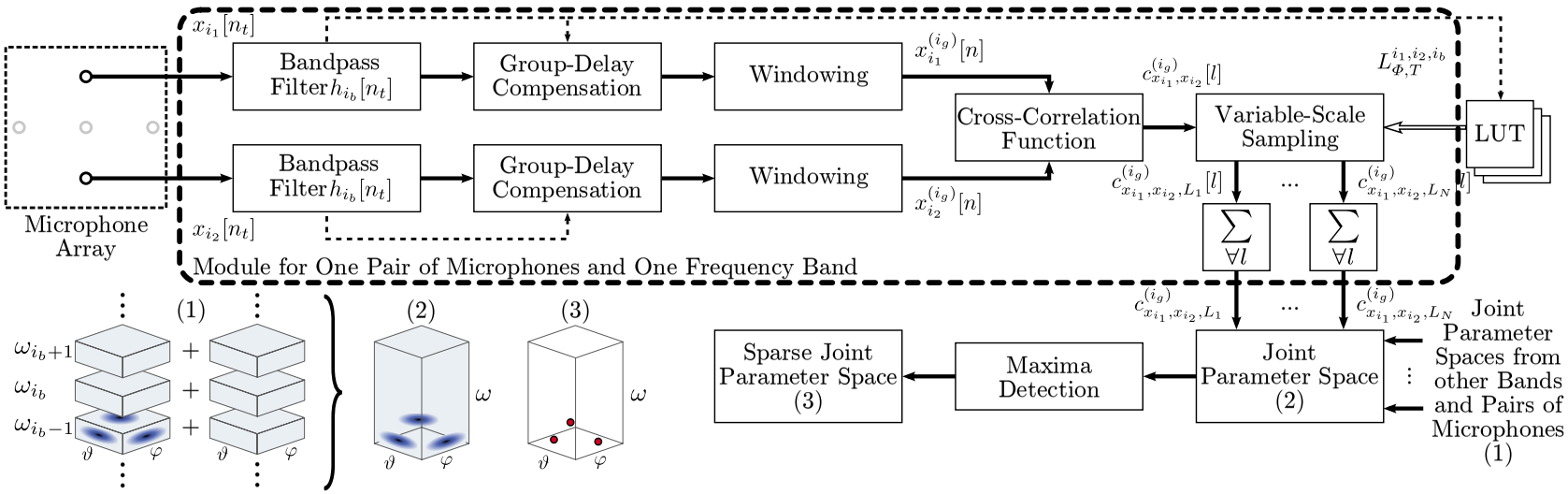


Fig. 2.1: Block diagram of the proposed algorithm that jointly estimates DOAs, f_0s, and respective amplitudes. All components inside the dashed rectangle belong to a module for one pair of microphones and one frequency band. The number of modules depends on the number of available pairs of microphones and the number of frequency bands. The components labeled with 'Windowing' split the discrete-time signals $x_{i_1}[n_t]$ and $x_{i_2}[n_t]$ from microphones with index i_1 and i_2 into frames; n_t is the sample index of the whole captured signal and n is the sample index of a windowed signal. Variable $h_{i_b}[n_t]$ is the impulse response of the i_b-th bandpass filter, $c_{x_{i_1}x_{i_2}}^{(i_g)}[l]$ is the CCF of $x_{i_1}^{(i_g)}[n]$ and $x_{i_2}^{(i_g)}[n]$ with lag index l and frame index i_g, $c_{x_{i_1}x_{i_2},L}^{(i_g)}[l]$ is the CCF sampled with a certain sampling period and sampling phase, $c_{x_{i_1}x_{i_2},L}^{(i_g)}$ is the sampled CCF summed over all lags, $L_{\Phi,T}^{(i_1,i_2,i_b)}$ denotes the subset of sampling phases and sampling periods for the i_b-th band and the microphones labeled with i_1 and i_2, φ is the azimuth, ϑ denotes the elevation, and f and ω represent the frequencies and angular frequencies, respectively. The lookup table (LUT) contains all relevant indices for variable-scale sampling. The components inside the dashed rectangle transform a frame of two microphones' time signals into (1) parameters of a certain band's (quasi-continuous) joint parameter space; merging each band's joint parameter space and detecting the local extrema of the resulting (2) merged joint parameter space yields (3) a sparse joint parameter space.

and unvoiced parts as well as breaks. In contrast with [13–25, 28–36, 45], I determined the ground truth for the recorded signals' instantaneous f_0s by analyzing their corresponding recorded glottograms, which enabled me to use a large variety of naturally produced fluent speech signals.

2.2 Parameters of Interest

The DOA is the source's spatial angle of incidence [52,53]. It is the spatialization of the relative TDOA of a propagating monochromatic plane wave observed at two different locations (or microphones). The f_0 [7] is the inverse of the fundamental period (T_0) of harmonic sources. It is the component of a harmonic structure exhibiting the lowest frequency. I do not refer to this as pitch, because pitch is a perceptual attribute and not a physical quantity [54,55]. By using f_0 as a parameter, I can improve the performance of a speech separating system [56] and efficiently employ subband beamforming. Besides, it improves parameter estimation when two or more speakers share the same DOA [18, 35, 57].

2.3 Microphone Array

To sample the acoustic wave field at specific positions in space [52], I employ an array made up of omnidirectional microphones. In case of a linear or planar array, I recommend to mount it on the enclosure of a room, e.g., the ceiling or the walls, to reduce or avoid spatial ambiguity [53]. To jointly estimate DOAs and f_0s using the proposed algorithm, the array's maximum dimension d_a has to be large enough to decrease its omnidirectional behavior at lower frequencies but short enough so that the assumption of plane wave propagation remains valid [58] and no spatial aliasing occurs [53, 58]. In case of the proposed approach, the plane wave propagation is essential in order to sample the cross-correlation functions. Thus, to ensure plane wave propagation [59], the minimum distance between a source and a uniform linear array's center, $d_{\min}$, has to be

$$d_{\min}(\gamma_i, \lambda_\omega, d_a) = d_a^2 \sin(\gamma_i)^2/(2\lambda_\omega) + d_a \mid \cos(\gamma_i)|/2 - \lambda_\omega/8, \tag{2.1}$$

where d_a is the array's maximum dimension, γ_i is the angle of incidence, and λ_ω is the wavelength of interest. Given d_a, for all angles of incidence, $0 \leq \gamma_i < 2\pi$, and wavelengths of interest, $\lambda_\omega = v/f$, $80 \leq f \leq 1000$ with f as the frequency of interest in Hz and v as the speed of sound in m/s, the minimum distance, $\hat{d}_{\min}$, is

$$\hat{d}_{\min} = \max_{\gamma_i, \lambda_\omega} d_{\min}(\gamma_i, \lambda_\omega). \tag{2.2}$$

In case of a uniform circular array (UCA), d_a is the array's diameter.

2.4 Filter Bank

In the proposed approach, a filter bank is a prerequisite to solve the multiple-source problem, to avoid pitch-period doubling [60,61] when using the CCF [62], and to reduce the influence of noise and narrow-band interfering sources.

In case of two or more harmonic sources, the CCF emphasizes the dominant source. Usually, this results in the estimation of the dominant source's parameters only. When using CCFs, a filter bank limits this nonlinear effect to a single frequency band only. The narrower the bands the better the estimation of all harmonic sources' parameters. Since voiced speech is sparse in time-frequency domain [63–66], the bandwidths can be small.

Given the CCF of a dual-channel harmonic broadband signal (i.e., a broadband signal captured by two microphones); sampling the function with all intervals of interest will result in estimated parameters unequal zero at multiples of the fundamental period. (These multiples do not physically exist.) This effect is called pitch-period doubling. Sampling the CCF of two bandpass-filtered signals with a specific set of sampling intervals solves the problem of pitch-period doubling. The set should contain sampling intervals that match the passband frequencies of the corresponding frequency band. Hence, the bandwidth must be limited (as described shortly).

A dominating interfering noise signal yields a high peak in the CCF of a dual-channel broadband signal. Depending on the noise signal's energy, the sampled CCF cause wrong estimated harmonic components in the joint parameter space. (There should be estimates for harmonic signals only.) Bandpass filters reduce the noise signal's energy in a frequency band without reducing the energy of a harmonic signal's component in the same frequency band. In such a case, the CCF will emphasize the harmonic component. If no harmonic component is present, the energy of the noise signal is small in case of narrow frequency bands and, therefore, will not be detected by the maxima detector.

Even in case of acoustic beating caused by two superimposed signals with almost the same f_0 [67], narrow-band filters limit this effect when using a CCF and when the two f_0s are close to a band's edge.

2.4.1 Bandpass Filter

I employ Kaiser window order-estimated bandpass filters [68–71] with predefined lower and upper cut-off frequencies to attain decreased passband ripple and steep passband-stopband transitions with a manageable order. The filters exhibit impulse responses with decreasing lengths to higher bands and constant group delays. Common alternatives are, e.g., the Gammatone filter [72], the Butterworth filter [73], and the Cauer filter [73]. They feature a non-constant group delay that has to be compensated by phase reversed filtering. However, by using symmetric FIR filters, I just need to properly delay the filtered signals.

2.4.2 Group Delay Compensation

To compare the estimated f_0 with its ground-truth value and to provide time-synchronous f_0-estimates, I compensate the constant group delay. In case of symmetric FIR filters, I delay the bandpass-filtered signal of the i_b-th band by

$$\Delta N_h^{(i_b)} = (N_h^{(i_b)} - 1)/2 \tag{2.3}$$

samples, where $N_h^{(i_b)}$ is the odd number of samples of the i_b-th bandpass filter's impulse response $h_{i_b}[n]$.

2.4.3 Bandwidth

The bandwidth of each frequency band must be small enough so that the sampling intervals (which match the frequency band's periods) do not sample multiples of the harmonic signal's fundamental period. Therefore, I split the frequency range of interest into N_b bands with equal bandwidth smaller or equal than $\Delta f = f_l/2$ with f_l as the lowest f_0 of interest (which is 75 Hz in the upcoming experiments). The number of bands is

$$N_b < (f_u - f_l)/\Delta f, \tag{2.4}$$

where f_u is the highest cut-off frequency.

2.5 Unbiased Cross-Correlation Function

The CCF [7, 70, 74],

$$c_{x_{i_1}x_{i_2}}[l] = \sum_{m=-\infty}^{+\infty} x_{i_1}[m]x_{i_2}^*[l+m], \tag{2.5}$$

is a function of time lag l, where $(\cdot)^*$ denotes complex conjugation. To determine DOAs and f_0s, I calculate $c_{x_{i_1}x_{i_2}}[l]$ of $x_{i_1}[m]$ and $x_{i_2}[m]$, each with a support interval of length N_x (between $0 \leq m \leq N_x - 1$), for $-N_x + 1 \leq l \leq N_x - 1$. By considering the Wiener-Khinchin theorem [75], I speed up the computation of the cross-correlation function according to

$$c_{x_{i_1}x_{i_2}}[l] = \mathcal{F}^{-1}\{C_{x_{i_1}x_{i_2}}[k]\} = \mathcal{F}^{-1}\{X_{i_1}[k] \cdot X_{i_2}^*[k]\}, \tag{2.6}$$

where $\mathcal{F}^{-1}$ is the inverse discrete Fourier transform and $C_{x_{i_1}x_{i_2}}[k]$ is the cross spectrum of $X_{i_1}[k] = \mathcal{F}\{x_{i_1}[n]\}$ and $X_{i_2}[k] = \mathcal{F}\{x_{i_2}[n]\}$. The windowed CCF is

$$c_{x_{i_1}x_{i_2}}[l] = \begin{cases} w[l] \sum\limits_{m=0}^{N_x-1-l} x_{i_1}[m]x_{i_2}^*[m+l] & l \geq 0 \\ w[-l] \sum\limits_{m=0}^{N_x-1+l} x_{i_2}[m]x_{i_1}^*[m-l] & l < 0 \end{cases}, \tag{2.7}$$

where m denotes the time shift. Considering the window

$$w[l] = \begin{cases} \dfrac{1}{N_x - |l|} & -N_x + 1 \leq l \leq N_x - 1 \\ 0 & \text{else} \end{cases}, \tag{2.8}$$

to reduce the decrease in amplitude (for $|l| > 0$) yields the unbiased CCF [76]. Preliminary experiments showed that computing the unbiased CCF frame-wise over time with the frame size and the overlap of frames mentioned in [77] (i.e., a frame size of 0.032 s and an overlap of 0.010 s) yields the best results.

2.6 Sampling Phase and Sampling Period

The two major parameters to sample the CCF are the sampling phase and the sampling period.

The sampling phase $L_{\Phi}(\varphi, \vartheta)$ is an extrinsic parameter that is related to TDOAs,

$$\tau_{i_1,i_2}(\varphi, \vartheta) = -(\mathbf{m}_{i_1} - \mathbf{m}_{i_2})^T \mathbf{k}(\varphi, \vartheta)/v, \tag{2.9}$$

with $\mathbf{k}(\varphi, \vartheta) = (\sin(\vartheta)\cos(\varphi), \sin(\vartheta)\sin(\varphi), \cos(\vartheta))^T$ as the spherical unit vector, φ and ϑ as the azimuth and elevation of a source, $\mathbf{m}_{i_1}$ and $\mathbf{m}_{i_2}$ as the i_1-th and i_2-th microphone coordinates, and with v as the speed of sound. To sample a CCF, I transform the TDOA into the sampling phase according to

$$L_{\Phi}^{(i_1,i_2)}(\varphi, \vartheta) = \lfloor \tau_{i_1,i_2}(\varphi, \vartheta)/T_s \rceil, \tag{2.10}$$

where $T_s = f_s^{-1}$, $T_s \in \mathbb{R}$, and $\lfloor \cdot \rceil$ rounds its argument to the nearest integer to avoid fractional delays. The sampling period $L_T(T_0)$ is an intrinsic parameter related to a source's f_0:

$$L_T(T_0) = \lfloor T_0/T_s \rceil\,, \tag{2.11}$$

where $T_0 \in \mathbb{R}$. Considering a low sampling frequency ($f_s < 8000$ Hz) and a big array aperture ($d_a > 0.5$ m), errors caused by spatial aliasing, imperfectly optimized bandpass filters, and a decreasing frequency resolution to higher frequencies would predominate. Relative to those errors, rounding errors turn out to be negligible.

To localize and characterize one or more harmonic sources, I calculate sampling periods and sampling phases for all f_0s and directions of interest. I define the subset of sampling phases and sampling periods for the i_b-th band and a pair of microphones consisting of microphone i_1 and i_2 as

$$L_{\Phi,T}^{(i_1,i_2,i_b)} \subset \left(L_{\Phi}^{(i_1,i_2)}, L_T^{(i_b)}\right), \tag{2.12}$$

$$L_T^{(i_b)} = \{L_T(T_0) \mid T_0 = f_0^{-1}, i_b f_l \leq f_0 \leq (2i_b+1) f_l/2\}, \tag{2.13}$$

$$L_{\Phi}^{(i_1,i_2)} = \{L_{\Phi}^{(i_1,i_2)}(\varphi, \vartheta) \mid 0 \leq \varphi < 360, 0 \leq \vartheta \leq 180\}. \tag{2.14}$$

This yields $N_b \cdot N_g$ subsets, where N_g is the number of pairs of microphones. A single tuple of an arbitrary sampling period and sampling phase is defined as

$$L \triangleq \left(L_{\Phi}^{(i_1,i_2)}(\varphi, \vartheta), L_T^{(i_b)}(T_0)\right). \tag{2.15}$$

To save computational resources, a lookup table provides all $N_b \cdot N_g$ subsets.

2.7 Variable-Scale Sampling of Cross-Correlation Function

Sampling each CCF enables us to jointly estimate DOAs, f_0s, and the respective amplitudes of one or more harmonic sources. I sample the CCF with a limited number of sampling points at specific lags. Therefore, I define a discrete sampling function known as the Shah function [78],

$$\mathrm{III}_{i_1,i_2}^{(i_b)}[l] = \sum_{i=-N_d}^{N_d} \delta\left[l - \left(iL_T^{(i_b)}(T_0) + L_{\Phi}^{(i_1,i_2)}(\varphi,\vartheta)\right)\right], \tag{2.16}$$

where $\delta[\cdot]$ is the Kronecker delta. The number of sampling points is $2N_d + 1$, where $N_d \in \{1, 2\}$. I sample the CCF according to

$$\hat{c}_{x_{i_1}x_{i_2}}^{(i_b)}[l] = c_{x_{i_1}x_{i_2}}^{(i_b)}[l] \cdot \mathrm{III}_{i_1,i_2}^{(i_b)}[l]. \tag{2.17}$$

Inserting (2.16) into (2.17) and summing over all lags l yields

$$\hat{c}_{x_{i_1}x_{i_2}}^{(i_b)} = \frac{1}{2N_d+1} \sum_{l=-N_x+1}^{N_x-1} c_{x_{i_1}x_{i_2}}^{(i_b)}[l] \sum_{i=-N_d}^{N_d} \delta\left[l - \left(iL_T^{(i_b)}(T_0) + L_{\Phi}^{(i_1,i_2)}(\varphi,\vartheta)\right)\right], \tag{2.18}$$

for an arbitrary L. Now, I construct a 3-tuple $(L_{\Phi}^{(i_1,i_2)}(\varphi,\vartheta), L_T^{(i_b)}(T_0), \hat{c}_{x_{i_1}x_{i_2}}^{(i_b)})$ that represents a point in a 4-dimensional joint parameter space (spanned by both angles, the frequencies or periods, and the amplitudes). I compute the CCF of the signals of each pair of microphones' band and for lags l distributed symmetrically around $l = 0$. In order to justify the use of the unbiased CCF, I first analyze a variable-scale sampled, biased CCF of a periodic signal. If I compute the biased CCF of a certain frequency band (see Fig. 2.2 top), perform the variable-scale sampling, and sum over all lags, I can estimate the frequency components. If I would do this for a periodic signal with a low-frequency component, there would be a remarkable difference in amplitudes compared to the previous case. The resulting amplitude of the high-frequency component is larger than the amplitude of the low-frequency component. However, the sampled amplitudes should be identical. By using the unbiased CCF, I overcome this problem. As shown in Fig. 2.2 (middle, bottom), the peaks' amplitudes of each unbiased CCF around $l = 0$ are almost identical due to the weighting described in (2.8).

2.8 Joint Parameter Space

The JPS is a joint representation of sampling periods, sampling phases, and respective amplitudes over time. Due to the joint estimation, these signal parameters are associated with each other. Fig. 2.3 shows a three dimensional JPS representing 3-tuples $(L_{\Phi}^{(i_1,i_2)}(\varphi,\vartheta = 90°), L_T^{(i_b)}(T_0), \hat{c}_{x_{i_1}x_{i_2}}^{(i_b)})$. I set up a JPS for each pair of microphones and add them together. The parameter space still contains irrelevant information. However, I am interested in tuples, i.e., points in the JPS, representing local maxima. Therefore, I sparsify this space (see Fig. 2.3) by employing an efficient multidimensional maxima detector to obtain a sparse representation of it, i.e., an SJPS, as shown in Fig. 2.4.

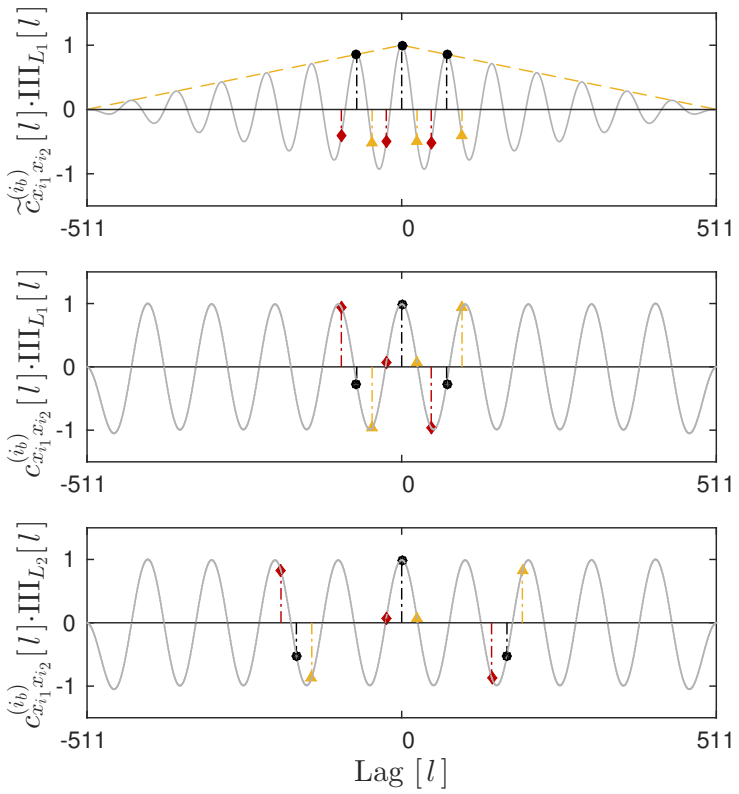


Fig. 2.2: Variable-scale sampling of the biased (top) and unbiased (middle, bottom) CCF by applying the Shah function with five ($N_d = 3$) and three ($N_d = 1$) sampling points, respectively, different sampling phases (red, black, yellow) and sampling periods based on the following frequencies: 7 Hz (top, middle) and 3 Hz (bottom). The yellow dashed lines (top) represent the decrease of amplitudes in case of the biased CCF.

2.9 Multidimensional Maxima Detector

To detect local maxima in the JPS, I apply a real-time capable multidimensional maxima detector based on Lemire's streaming maximum-minimum filter [79, 80].

Based on a sliding, hypercubic window, the detector sparsifies the JPS, which contains the associated parameters of one or more sources. If the window size is too small, the detector might detect fluctuations caused by, e.g., noise, which would introduce undesirable local maxima, i.e., clutter. If the window size is too large, the detector might fail in detecting two or more sources, whose parameters are close together in the parameter space. A fundamental problem of extrema detection in bounded spaces is the detection of endpoint or boundary extrema [81], which can be true or false extrema. To solve the problem, I extend the sampling phases' domain according to $0 - N_v \leq \varphi < 360 + N_v$ and $0 - N_v \leq \vartheta \leq 180 + N_v$, and I extend each subset of sampling periods by a single period, $N_v = 1$, at both set boundaries. Afterwards, I employ the extrema detector and eliminate those extrema detected in the extension. I sort the list of maxima according to their amplitude and select N_e maxima with the highest amplitudes. Variable N_e must be higher than the number of expected harmonic sources, N_s, times the maximum number

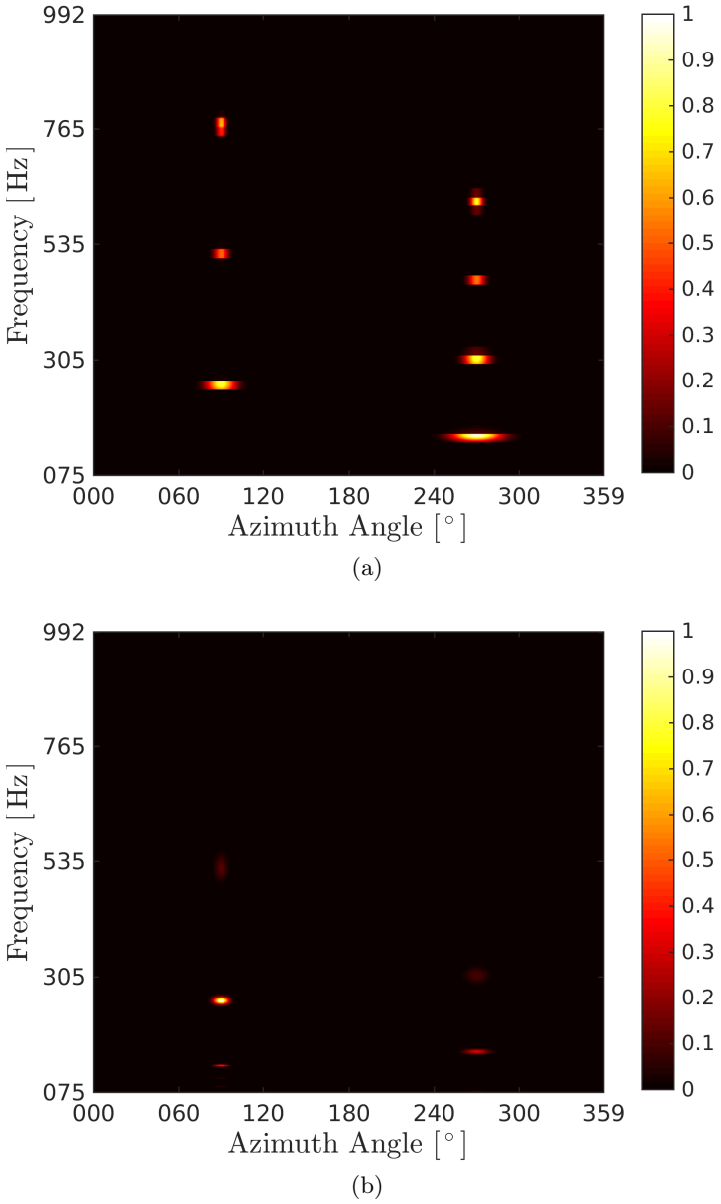
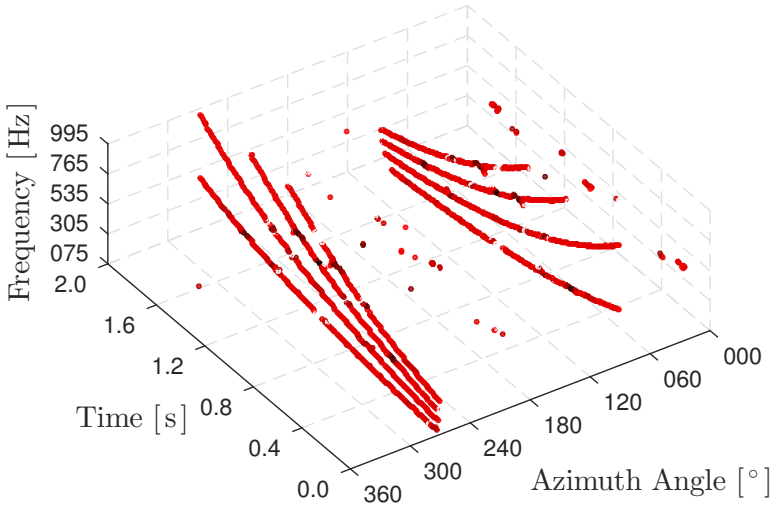
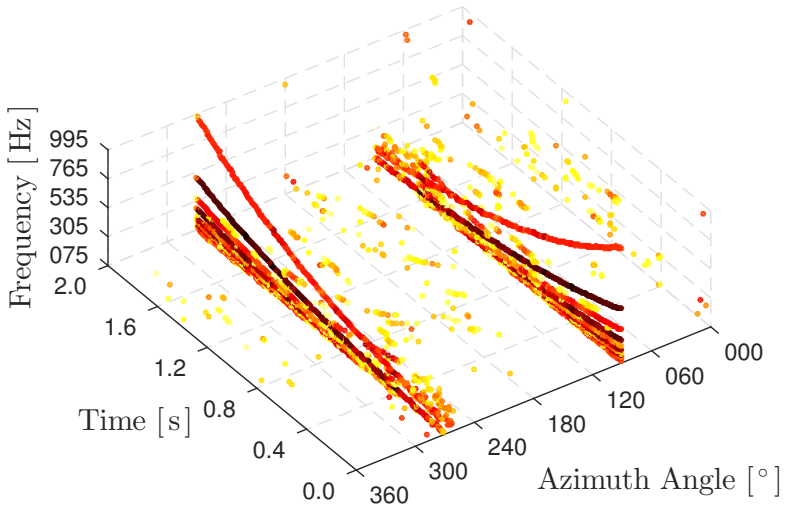


Fig. 2.3: A normalized three-dimensional JPS (similar to the POPI plane [45]) representing the jointly estimated DOAs, $f_0 s$ and their corresponding second, third, and fourth harmonics with the respective CCF-values (half-wave rectified and normalized to achieve values between zero and one) computed (a) with the proposed algorithm and (b) with its predecessor [35] without considering spectral fragments. In this figure, the sources' parameters are $\varphi_{s_1} = 90°$ and $f_0^{(s_1)} = 240$ Hz as well as $\varphi_{s_2} = 270°$ and $f_0^{(s_2)} = 160$ Hz. By comparing both planes one can see in (b) that the predecessor exhibits pitch-period doubling (at approximately 120 Hz and 90°), fewer harmonics, and harmonics with different amplitudes—all of them should be identical in this scenario. The widening of the Gaussian-like kernels to lower frequency bands in (a) is due to the increase in a band's CCF's sampling periods to lower frequency bands and the variable-scale sampling in lag domain. The higher the band, the narrower the Gaussian-like kernel and vice versa. Sampling the CCF of a harmonic dual-channel broadband signal significantly decreases the widening effect, as shown in (b).



(a)



(b)

Fig. 2.4: Trajectories obtained after joint estimation of DOAs, $f_0 s$, and second, third, and fourth harmonics with their respective amplitudes with (a) the proposed algorithm and (b) with its modified predecessor [35]. In order to generate a SJPS with [35], I avoided marginalizing over frequencies and utilized a multidimensional maxima detector. The predecessor's SJPS exhibits far more clutter, wrong amplitudes (yellow to red), missing higher harmonics, and spurious "subharmonics". In both cases, I applied Lemire's extrema detector to sparsify the JPS.

Algorithm 1: Source Localizer and Characterizer

```
   Data: Discrete-time multi-channel signals.
   Result: Sparse joint parameter spaces.
 1 initialization; // (2.3), (2.8),
 2 compute sets of indices for variable-scale sampling; // (2.9)-(2.14)
   // store sets in lookup table
   // consider extensions for maxima detector
 3 split into frequency bands by applying bandpass filters;
 4 split into frames;
 5 while getting frames do
 6     foreach pair of microphones do
 7         foreach frequency band do
 8             transform frames of both mic-channels into frequency domain;
 9             compute cross spectrum;
10             transform into time domain; // (2.6)
11             apply inverse windowing; // (2.7)
12             apply sets of indices to unbiased cross correlation;
13             sum samples related to each set's indices; // (2.18)
               // joint parameter space per pair and band
14         end
15         concatenate (≠sum) each frequency band's joint parameter space;
           // joint parameter space per microphone pair
16     end
17     sum all microphone pairs' joint parameter spaces;
18     scale joint parameter space by number of pairs of microphones;
       // joint parameter space
19     detect maxima; //  [80]
20     eliminate maxima in extension (see Section 2.9);
       // sparse joint parameter space
21 end
```

Fig. 2.5: Pseudo code of the proposed algorithm based on variable-scale sampling. Two slashes indicate a comment.

of a signal's harmonics, $\widehat{N}_q$, that can occur in the range of $[f_l, f_u]$, where $\widehat{N}_q f_l \leq f_u$.

2.10 Joint Parameter Estimation

The SJPS is a non-parametric signal representation that contains the JPS's local maxima only. To jointly estimate the parameters of one or more harmonic sources, I need to know the general signal model (1.2) or (1.3) and analyze the SJPS. As shown in Fig. 2.4(a), the f_0's, the corresponding harmonics, and their respective amplitudes at a certain DOA belong to a single harmonic source. To determine the f_0 of this specific source without using an explicit estimator or detector, I pick its lowest estimated frequency within a narrow tolerance window around a certain DOA and ignore isolated clutter. Fig. 2.5 contains the algorithm's pseudo code, which literally refers to all blocks shown in Fig. 2.1.

2.11 Metrics

To evaluate the performance of algorithms that localize and characterize harmonic sources, I employed metrics well known in the field of information retrieval, classification, or parameter estimation: the recall, the root mean square error, and the cumulative distribution function.

2.11.1 Joint Recall

The recall is the ratio of the number of correctly retrieved relevant parameters to the total number of relevant parameters; a tuple (φ, f_0) represents such a relevant parameter. Using the terminology of a confusion matrix, e.g., true positives (TP) and false negatives (FN), I defined the recall of jointly estimated DOAs and f_0s, i.e., the joint recall, as

$$\mathrm{R}_i(\varphi, f_0) = \frac{\mathrm{TP}_i(\varphi, f_0)}{\mathrm{TP}_i(\varphi, f_0) + \mathrm{FN}_i(\varphi, f_0)} \tag{2.19}$$

with i denoting the index of a Monte Carlo experiment. The average joint recall of N_c Monte Carlo experiments is

$$\overline{\mathrm{R}}(\varphi, f_0) = \frac{1}{N_c} \sum_{i=1}^{N_c} \mathrm{R}_i(\varphi, f_0). \tag{2.20}$$

A prerequisite in noisy environments or in case of estimation errors is to consider tolerance windows around ground-truth items to be able to score true positives. For instance, an estimate lying inside that region yields a true positive. If two or more estimates are inside an intersection of two or more ground-truth items' tolerance windows, I have to optimally assign the estimates to the ground-truth items. As the following example shows, this can be challenging.

Example 1. Optimal Assignment As this example shows, assigning estimates to ground-truth items is not always straight forward. Ambiguities, as described below, can cause better or worse scores.

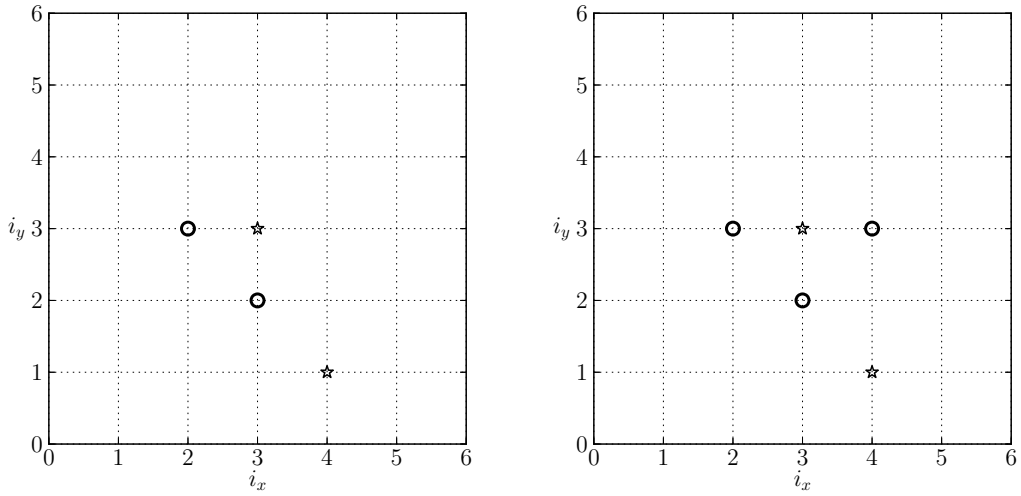


Fig. 2.6: Ambiguities in assigning ground-truth items ($\star$) to estimates ($\circ$). The tuple (i_x, i_y) represents a coordinate consisting of indices i_x and i_y.

Considering a tolerance window of $(1, 1)$, I can assign in the left figure, e.g., estimate $(3, 2)$ to item $(3, 3)$. This yields TP $= 1$, because the second estimate $(2, 3)$ cannot be assigned to item $(4, 1)$ due to the tolerance window. However, assigning estimate $(3, 2)$ to item $(4, 1)$ facilitates assigning estimate $(2, 3)$ to item $(3, 3)$, which yields TP $= 2$. In the right figure, I can assign estimate $(3, 2)$ to item $(3, 3)$, which results in TP $= 1$. This is not the global optimum; thus, I have to assign estimate $(2, 3)$ to item $(3, 3)$ before assigning estimate $(3, 2)$ to item $(4, 1)$, which is the optimal way to assign the estimates to the ground-truth items. Assigning two estimated items to one ground-truth item does not increase the joint recall; but it can decrease the joint recall when the second estimated item should be assigned to another ground-truth item in the vicinity. Thus, I assume that one ground-truth item can be assigned to a single estimated item. ■

Inspired by the optimal subpattern assignment (OSPA) distance [82,83] and its label assignment (I will describe both metrics in Chapter 4), I designed a method that assigns estimates to ground-truth items in an optimum manner regarding the number of true positives. The method is as follows: First, I identify elements of the estimates ($\mathfrak{Y}$) that lie inside the tolerance region of an element of ground-truth items ($\mathfrak{X}$) and construct a tuple, $(\mathbf{x}, \mathbf{y})$, ($\mathbf{x} \in \mathfrak{X}$, $\mathbf{y} \in \mathfrak{Y}$) which I assign to a set of tuples. Second, I eliminate the tuple whose first component, $\mathbf{x}$, occurs once in the set and increase the number of true positives by one. I repeat this procedure until each tuple with a unique first component is eliminated from the set. The next step is to decompose each tuple and construct two sets consisting of the tuples' first and second components considering that there are no duplicates in a set. After determining the cardinalities of both sets, I increase the number of true positives by the smallest set's cardinality. By considering the true positives, the number of elements in $\mathfrak{X}$, as well as the number of elements in $\mathfrak{Y}$, I am able to compute the false negatives which are required to compute the recall and precision. Figure 2.7 shows another description of the aforementioned assignment algorithm in terms of pseudo-code. I designed this algorithm to optimally assign the ground-truth items to estimated items in terms of joint recall.

Example 2. Optimal Assignment Assume that $\mathfrak{X} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}\}$ and $\mathfrak{Y} = \{\mathbf{y}^{(1)}, \mathbf{y}^{(2)}\}$ are located at indices $\{(1, 3), (3, 1)\}$ and $\{(2, 2), (4, 2)\}$, respectively. Considering a tolerance region of one index in all directions, the following assignments exist: $(\mathbf{x}^{(1)}, \mathbf{y}^{(1)})$, $(\mathbf{x}^{(2)}, \mathbf{y}^{(1)})$, and $(\mathbf{x}^{(2)}, \mathbf{y}^{(2)})$. Due to $|\mathfrak{Y}| = 2$, I can choose at most two assignments. However, if I first select $(\mathbf{x}^{(2)}, \mathbf{y}^{(1)})$, I cannot select the former or latter assignment anymore due to the tolerance region. Thus, I need to find the optimal assignment by applying the aforementioned method. Ensuring an optimal assignment, the method selects the components with indices $(2, 2)$ first, because $i_x = 2$ occurs once. Another way to show how this method works is as follows:

$$\begin{array}{c|c} i_x & i_y \\ \hline 1 & 1 \\ 1 & 2 \\ 2 & 2 \end{array} \quad \Rightarrow \quad \begin{array}{c|c} i_x & i_y \\ \hline 1 & 1 \\ 1 & 2 \end{array} \quad \Rightarrow \quad \begin{array}{l} I_x = \{1\} \\ I_y = \{1, 2\} \end{array} \quad \Rightarrow \quad \min(|I_x|, |I_y|) = 1$$

Algorithm 2: Determining Parameters of the Confusion Matrix

Data: $\mathfrak{Y}$ (estimates), $\mathfrak{X}$ (ground-truth items), $\mathbf{x} \in \mathfrak{X}$, $\mathbf{y} \in \mathfrak{Y}$
Result: TP (true positives), FP (false positives), FN (false negatives)

$i_x = i_y = 0$;
foreach $\mathbf{y} \in \mathfrak{Y}$ **do**
  $i_y \coloneqq i_y + 1$;
  foreach $\mathbf{x} \in \mathfrak{X}$ **do**
    $i_x \coloneqq i_x + 1$;
    if $(\mathbf{x} - \mathbf{w} \leqq \mathbf{y} \leqq \mathbf{x} + \mathbf{w})$ **then**
      $I_x \coloneqq I_x \cup \{i_x\}$;
      $I_y \coloneqq I_y \cup \{i_y\}$;
      $I_{xy} \coloneqq I_{xy} \cup \{(i_x, i_y)\}$;
    end
  end
end
repeat
  $I_c = \{\emptyset\}$;
  foreach $i_x \in I_x$ **do**
    $J_{xy} \coloneqq \{(i_x, j_y) \mid \forall i_y \in I_y \wedge (i_x, j_y) \in I_{xy}\}$;
    if $|J_{xy}| = 1$ **then**
      $(j_x, j_y) \coloneqq J_{xy}$;
      $I_k \coloneqq \{k_x \mid \forall k_x \in I_x \wedge (k_x, j_y) \in I_{xy}\}$;
      $I_{xy} \coloneqq I_{xy} \setminus \{(k_x, j_y) \mid \forall k_x \in I_x \wedge (k_x, j_y) \in I_{xy}\}$;
      $I_x \coloneqq I_x \setminus I_k$;
      $I_y \coloneqq I_y \setminus I_k$;
      $I_c \coloneqq I_c \cup I_k$;
    end
  end
  TP $\coloneqq$ TP+$|I_c|$;
until $|I_c| \equiv 0$;
TP $\coloneqq$ TP + $\min(|I_x|, |I_y|)$;
FN $\coloneqq |\mathfrak{X}| -$ TP;
FP $\coloneqq |\mathfrak{Y}| -$ TP;
// Note that $I_x \cup I_x = I_x$ leads to $|I_x| \equiv |I_y|$, $|I_x| \leq |I_{xy}|$, $|I_x| \leq |\mathfrak{X}|$, and $|I_y| \leq |\mathfrak{Y}|$.

Fig. 2.7: Pseudo-code representing the computation of the number of true positives, false positives, false negatives, and the assignment of estimates to ground-truth items.

First, I eliminate the row in the table with an index i_x that occurs once only; the number of true positives increases by one. Then, I assume the new table's left and right column as sets I_x and I_y, respectively, calculate their cardinalities, and select the set with the lowest cardinality yielding a total number of two true positives. ■

2.11.2 Root Mean Square Error

The root mean square error represents the difference between ground-truth items and estimated items and is defined as

$$\mathrm{RMSE}_{i_1}(\hat{\Theta}) = \sqrt{\frac{1}{N_{F,i_1}} \sum_{i_2=1}^{N_{F,i_1}} (\hat{\Theta}_{i_2} - \Theta_{i_2})^2} \tag{2.21}$$

with $\hat{\Theta}$ and Θ denoting an estimated value and a ground-truth value, respectively. For instance, $\hat{\Theta} = \varphi$ and $\Theta = \psi$, where φ is the true DOA and ψ is the estimated DOA. N_{F,i_1} is the total number of frames of a windowed signal in a single Monte Carlo experiment. It represents the RMSE of DOAs, and f_0s, where ψ and f_0 are the ground-truth parameters. The average RMSE of all Monte Carlo experiments is

$$\overline{\mathrm{RMSE}}(\hat{\Theta}) = \frac{1}{N_c} \sum_{i=1}^{N_c} \mathrm{RMSE}_i(\hat{\Theta}). \tag{2.22}$$

When considering a tolerance window (as in case of the joint recall metric), I have to optimally assign estimated items to ground-truth items regarding the number of RMSE computations per frame. The assignment algorithm has to assign as many estimates to ground-truth items as possible regardless of each pair's distance. Figure 2.8 shows the metric's pseudo-code, which is partially identical to the recall's pseudo-code. As in case of the joint recall, the algorithm first assigns estimates to ground-truth items yielding the maximum number of possible pairings. Each tuple containing a unique first ground-truth item is eliminated from the set. The algorithm computes the distances between a certain ground-truth item and each remaining estimated items, selects the estimated item featuring the smallest distance to the ground-truth item, and stores the new pairing $(\mathbf{x}^{(i_x)}, \mathbf{y}^{(i_y)})$ in a new set $\mathfrak{Z}$ until no ground-truth items are left.

2.11.3 Cumulative Distribution Function

To visualize the experimental results of a big-data problem, I employ the cumulative distribution function. It illustrates a vast number of results in terms of a monotonic increasing curve. I use X as a random variable whose individual outcomes are RMSE_i, $i = 1, ..., N_c$. Then, its cumulative distribution function $F_X(\mathrm{RMSE})$ for a given RMSE is

$$F_X(\mathrm{RMSE}) = P(X \leq \mathrm{RMSE}). \tag{2.23}$$

To compute the cumulative distribution function of recalls, I choose Y as a random variable where its outcomes are the recall values R_i, $i = 1, ..., N_c$ yielding

$$F_Y(1 - \mathrm{R}) = P(Y \leq 1 - \mathrm{R}). \tag{2.24}$$

The cumulative distribution function for recalls is

$$F_Y(\mathrm{R}) = P(Y \leq \mathrm{R}). \tag{2.25}$$

However, as I am interested in values of R close to 100%, I may redefine $\mathrm{R} = 1 - \epsilon$ to obtain

$$F_Y(1 - \epsilon) = P(Y \leq 1 - \epsilon), \tag{2.26}$$

and finally, to make the graph reflect monotonically decreasing quality in a similar way as the CDF of the RMSE, I consider

$$1 - F_Y(1 - \epsilon) = 1 - P(Y \leq 1 - \epsilon) = P(Y > 1 - \epsilon) = P(Y > \mathrm{R}). \tag{2.27}$$

Algorithm 3: Assigning Estimates to Ground-Truth Items

Data: $\mathfrak{Y}$ (estimates), $\mathfrak{X}$ (ground-truth items), $\mathbf{x} \in \mathfrak{X}$, $\mathbf{y} \in \mathfrak{Y}$
Result: $\mathfrak{Z}$ (set of paired estimates and ground-truth items)

$i_x = i_y = 0$;
foreach $\mathbf{y} \in \mathfrak{Y}$ **do**
  $i_y \coloneqq i_y + 1$;
  foreach $\mathbf{x} \in \mathfrak{X}$ **do**
    $i_x \coloneqq i_x + 1$;
    if $(\mathbf{x} - \mathbf{w} \leqq \mathbf{y} \leqq \mathbf{x} + \mathbf{w})$ **then**
      $I_x \coloneqq I_x \cup \{i_x\}$;
      $I_y \coloneqq I_y \cup \{i_y\}$;
      $I_{xy} \coloneqq I_{xy} \cup \{(i_x, i_y)\}$;
    end
  end
end
$\mathfrak{Z} = \{\emptyset\}$;
repeat
  $I_c = \{\emptyset\}$;
  foreach $i_x \in I_x$ **do**
    $J_{xy} \coloneqq \{(i_x, j_y) \mid \forall i_y \in I_y \wedge (i_x, j_y) \in I_{xy}\}$;
    if $|J_{xy}| = 1$ **then**
      $(j_x, j_y) \coloneqq J_{xy}$;
      $I_k \coloneqq \{k_x \mid \forall k_x \in I_x \wedge (k_x, j_y) \in I_{xy}\}$;
      $I_{xy} \coloneqq I_{xy} \setminus \{(k_x, j_y) \mid \forall k_x \in I_x \wedge (k_x, j_y) \in I_{xy}\}$;
      $I_x \coloneqq I_x \setminus I_k$;
      $I_y \coloneqq I_y \setminus I_k$;
      $I_c \coloneqq I_c \cup I_k$;
      $\mathfrak{Z} \coloneqq \mathfrak{Z} \cup \{(\mathbf{x}^{(j_x)}, \mathbf{y}^{(j_y)})\}$;
    end
  end
  if $(|I_c| \equiv 0 \wedge |I_x| > 0)$ **then**
    $(j_x, j_y) = \underset{(i_x, i_y) \in I_{xy}}{\arg\min} \; |\mathbf{x}^{(i_x)} - \mathbf{y}^{(i_y)}|$;
    $I_x \coloneqq I_x \setminus \{k_x \mid \forall k_x \in I_x \wedge (k_x, j_y) \in I_{xy}\}$;
    $I_y \coloneqq I_y \setminus \{k_y \mid \forall k_y \in I_y \wedge (j_x, k_y) \in I_{xy}\}$;
    $I_{xy} \coloneqq I_{xy} \setminus \{(k_x, k_y) \mid \forall k_x \in I_x \wedge (k_x, j_y) \in I_{xy}, \forall k_y \in I_y \wedge (j_x, k_y) \in I_{xy}\}$;
    $\mathfrak{Z} \coloneqq \mathfrak{Z} \cup \{(\mathbf{x}^{(j_x)}, \mathbf{y}^{(j_y)})\}$;
    $I_c = a$, $a \in \mathbb{R} \setminus 0$;
  end
until $|I_c| \equiv 0$;
// Note that $I_x \cup I_x = I_x$ leads to $|I_x| \equiv |I_y|$, $|I_x| \leq |I_{xy}|$, $|I_x| \leq |\mathfrak{X}|$, and $|I_y| \leq |\mathfrak{Y}|$.

Fig. 2.8: Pseudo-code representing the assignment of estimates to ground-truth items required to compute the root mean square error.

Table 2.1: List of all relevant parameters for generating synthetically spatialized, linearly frequency-sweeping signals. The variables denote the angular step size $\Delta\varphi$, the elevation angle ϑ, the number of microphones N_m, the array lengths d_a, the number of harmonics N_q, the sweep's start frequency and stop frequency f_1 and f_2, the sweep's duration T_2, the distance between the source and the array's center $|s|$, the signal-to-noise ratio SNR, the signal-to-interference ratio SIR, the temporal signal components' amplitude α, the uniform distribution of noise with its interval $\mathcal{U}(-0.4, 0.4)$, and the angular grid Φ.

$\Delta\varphi$	ϑ	N_m	d_a	N_q	f_1	f_2	T_2	$\lvert s\rvert$
1°	90°	$\{2, 4, 6, 8, 16\}$	$\{0.20, 0.30, 0.40\}$ m	4	80 Hz	500 Hz	2 s	3 m

SNR/SIR	α	ν	Φ
$\{-10, 0, 10, 20, 30\}$ dB	$0.4\sqrt{10^{\frac{\text{SNR}}{10}}}$	$\mathcal{U}(-0.4, 0.4)$	$\{0°, \ldots, 359°\}$

It describes the probability that a certain percentage of all experiments yields an $1 - \text{R}$, i.e.,

$$F_Y(1 - \text{R}) = P(Y \leq 1 - \text{R}), \tag{2.28}$$

of a certain value and smaller. For instance, in the latter case, $1 - \text{R} = 0.75$ equals a joint recall of 0.25 (or 25%). I estimate the CDFs, $F_X(\text{RMSE})$ and $F_Y(1 - \text{R})$, by (a) computing the total number of Monte Carlo experiments, (b) sorting all corresponding results (i.e., $\forall i : \text{RMSE}_i$ or $\forall i : 1 - \text{R}_i$) in an ascending manner, (c) defining intervals from 0 to a non-negative number unequal zero, and (d) counting the measurements lying within those intervals. Employing a CDF to visualize results yields several benefits, especially in case of Monte Carlo experiments. First, it reveals the whole range of outcomes, i.e., RMSEs and Rs or $1 - \text{Rs}$, and their corresponding probabilities. Second, the slope of a CDF tells us in which interval most of the outcomes occur.

2.12 Experimental Design

Before giving details on the experimental design, I introduce algorithmic and environmental parameters valid for all upcoming experiments. Table 2.1 lists most of the algorithmic and environmental parameters. In addition to these parameters, I set the frame size to 0.032 s, the overlap of frames to 0.010 s, and the size of the maxima detector's search window to (6×6) indices. For the evaluations, I considered a tolerance window of 10 Hz and 10° around the ground truth to define the root-mean-square errors and joint recalls, especially in case of double-source experiments. I also considered an absolute amplitude threshold of 10^{-5} in the JPS to limit the number of detected maxima; maxima below that value were omitted.

In the next subsections, I thoroughly describe the different categories of experiments with synthesized signals and experiments with synthetically spatialized real speech signals.

2.12.1 Experiments with Synthesized Signals

For these experiments, I simulated spatially non-moving and moving harmonic sources and noise sources in free field considering a uniform circular array with $N_m \geq 4$. Table 2.1 lists all relevant parameters for generating the corresponding signals described in (1.3). In some experiments, I added uniformly distributed noise (rather than Gaussian noise) to avoid clipping signals and to be able to precisely control the distribution's support. In all double-source experiments I attenuated or amplified the target source signal yielding different SNRs.

To determine the best setting of algorithmic parameters, I conducted a vast number of Monte Carlo experiments in four different categories.

In each experiment, I randomly chose the sampling frequency, $f_s \in \{16, 32, 48, 64, 96\}$ kHz, the number of the Shah function's sampling points, $N_d \in \{1, 2\}$, and the parameters mentioned before. I initialized each source with a random DOA. In case of two sources, I set the minimum initial angular difference between each source to 20°. Mobile sources were moving along circular paths with an angular velocity of 1 m/s or 3 m/s clockwise or counter-clockwise causing intersections in spatial trajectories and frequencies. If two microphones were selected only, I considered azimuth angles in the interval $[0, 180]$ degrees only due to a linear array's spatial ambiguity [53].

In each category, I carried out 10^5 Monte Carlo experiments to find the most robust setting of algorithmic parameters. After doing so, I selected the most robust setting of parameters for further experiments to determine the algorithm's performance. These experiments were, again, Monte Carlo experiments due to varying initial DOAs, velocities, directions of moving sources, SNRs, and SIRs. I conducted experiments in four different categories with different algorithms for comparison:

The first category of experiments is as follows: In the first scenario a single non-moving source (see Fig. 2.9(a)) emitted an frequency-sweeping harmonic signal at varying locations. In the second scenario a single moving source emitted an frequency-sweeping harmonic signal while moving along a circular path around the microphone array.

In the second category's first scenario a non-moving harmonic source emitted an f_0-sweeping harmonic signal at varying locations together with a non-moving noise source (see Fig. 2.9(b)) featuring a different location. In another scenario each source moved along a circular path around the array, which featured spatial intersections.

In the third category, there are again two different scenarios: two non-moving harmonic sources (see Fig. 2.9(c)) and two moving harmonic sources (see Fig. 2.10); in both scenarios the harmonic sources were emitting an f_0-sweeping harmonic signal at different locations. The goal was to estimate the parameters of both sources.

In category number four, I simulated a trumpet emitting a sequence of tones [18, 20, 23] in a noisy environment by considering a non-moving, randomly frequency-hopping harmonic source and a noise source at varying locations. The signal model was the same as described earlier, except that the f_0 changed abruptly after time intervals of 500 ms. The authors of the aforementioned articles presumably conducted experiments with a trumpet due to its distinct harmonics and constant tones over (short) time intervals. Especially constant tones are advantageous when estimating frequency components frame-wise and adaptively. Furthermore, a trumpet is usually part of a classical and contemporary orchestra and, thus, a good choice when conducting experiments in the field

of music signal processing.

Comparisons With Other Algorithms

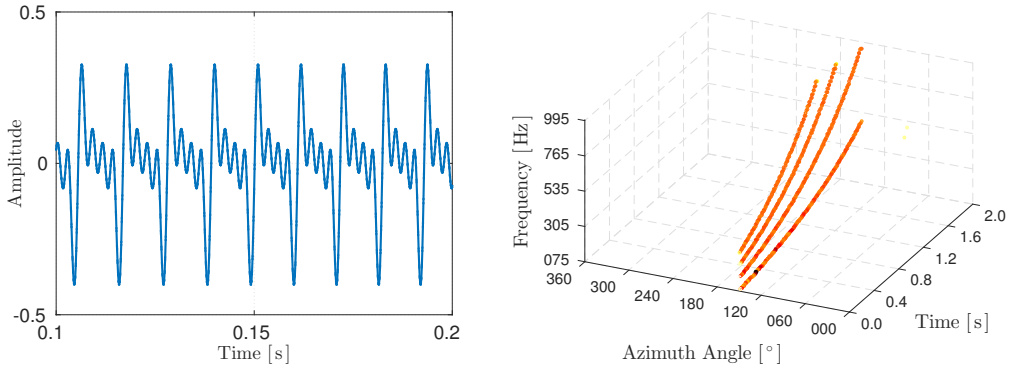
Additionally, I conducted experiments with four different algorithms denoted as VSS (variable scale sampling), POPI (position-pitch [35, 45]), NLS (nonlinear least squares [23]), and aNLS (approximate nonlinear least squares [23]). I considered experiments from the first three categories with non-moving sources.

At this point, I need to clarify some issues regarding the NLS and the aNLS published in [23]. First, one of my master students, Mattia Gabbrielli, implemented both algorithms following the description in [23]. He realized that some relevant information was missing. The authors did not specify the line search algorithm to adapt the step size over iterations. Thus, I decided to implement a backtracking line search based on the Armijo-Goldstein condition [84]. Second, they did not mention which initial values they used for their step sizes and starting points. I fixed the initial step size, $\delta^{(\text{init})} = 1$, and I randomly selected the initial parameters (the DOA and the f_0) inside the domain. Third, almost all arguments of exponential functions in [23] feature a unit; however, all units in an exponential function's argument must cancel out. I realized that they did not multiply the affected arguments with the sampling period T_s. A workaround would be defining the time instances n_t in seconds instead of samples. Regarding the use of the NLS and aNLS algorithm, I set the model order to 4, the number of time instances per frame to 80, the number of iterations to 60, the line search method's contraction factor and slope modifier to 0.5 and 10^{-5}, respectively, and the sampling frequency, as suggested in [23], to 8 kHz.

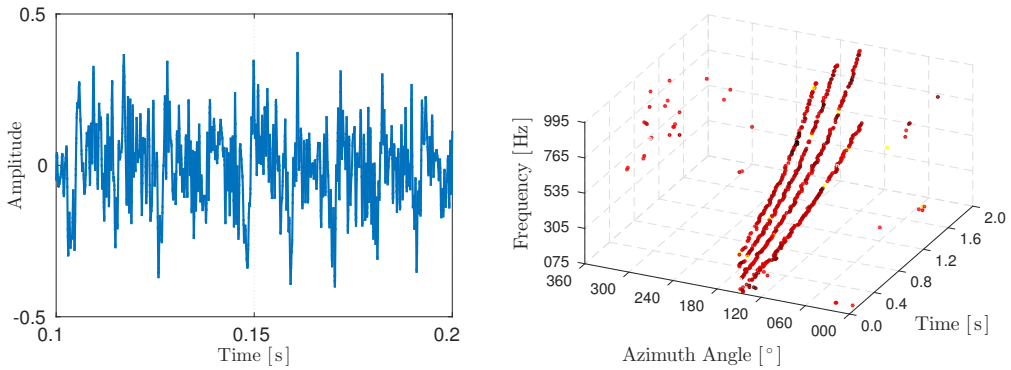
Originally, the POPI algorithm estimates DOAs only. As a consequence, determining a source's true f_0s using the POPI-algorithm presented in [35, 45] is impossible. In order to generate a SJPS with the POPI algorithm, I avoided marginalizing over frequencies and utilized a multidimensional maxima detector.

2.12.2 Experiments with Real Speech Signals

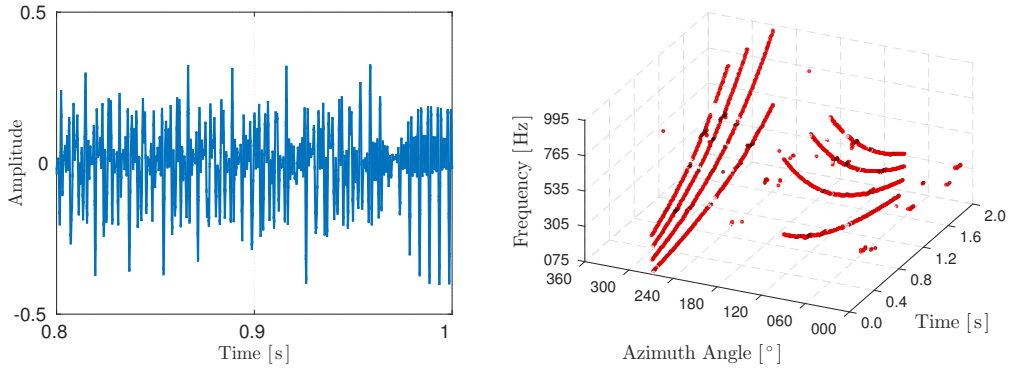
To evaluate the algorithm's performance in real environments, I especially set up an Austrian-German speech corpus (AMISCO: The Austrian German Multi-Sensor Corpus) [50, 51]. I thoroughly describe this corpus in Chapter 5. For these experiments I used recordings of read items from speakers 08 (female) and 22 (male) (see Fig. 5.3). Besides, I only focused on two- and three-channel recordings in the meeting room with arrays' maximum diameters of $d_a = 0.30$ m and $d_a = 0.60$ m, respectively. I applied a short-term power estimation utilizing a first-order infinite impulse response smoothing of the signal's instantaneous power [85] to compute the speaker's SNR. To extract f_0s from all glottograms, I first computed a one-sided unbiased auto-correlation of each glottogram's frame (with a frame length of 32 ms and a frame shift of 5 ms). Then, I employed a maximum detector to detect the lag of the auto-correlation's global maximum between lags of 2 ms and 13 ms. The inverse of the global maximum's lag corresponds to the f_0, which I used as the true f_0 [51].



(a) $f_s = 48$ kHz



(b) $f_s = 48$ kHz, $\varphi_n = 354°$, SNR $= -10$ dB



(c) $f_s = 96$ kHz, SIR $= 0$ dB

Fig. 2.9: Jointly estimated f_0s, second, third, and fourth harmonics, as well as DOAs of a non-moving frequency-sweeping source. The experimental parameters are as follows: source direction $\varphi_s = 150°$, maximum dimension of microphone array $d_a = 0.40$ m, $N_m = 8$ microphones, five sampling points ($N_d = 2$), $N_e = 8$ is the maximum number of the selected maxima, angular resolution $\Delta\varphi = 1°$, and $(3, 3)$ is the window size of the maxima detector. The frequency-sweeping harmonic signal starts at $f_0 = 75$ Hz and ends at $f_0 = 500$ Hz. All harmonics exhibit the same amplitude. The distance between the virtual microphones and the source is $|s| = 3$ m. The left column illustrates the time signals, whereas the right column shows the respective SJPSs. In (a) there is a snapshot of a non-moving frequency-sweeping harmonic source with four harmonics (left) and the corresponding SJPS (right). In (b) I considered additive white noise yielding SNR $= -10$ dB. The plots in (c) show the time signal and the SJPS of two non-moving frequency-sweeping harmonic sources with $(\varphi_s^{(1)}, \varphi_s^{(2)}) = (90°, 240°)$ and SIR $= 0$ dB.

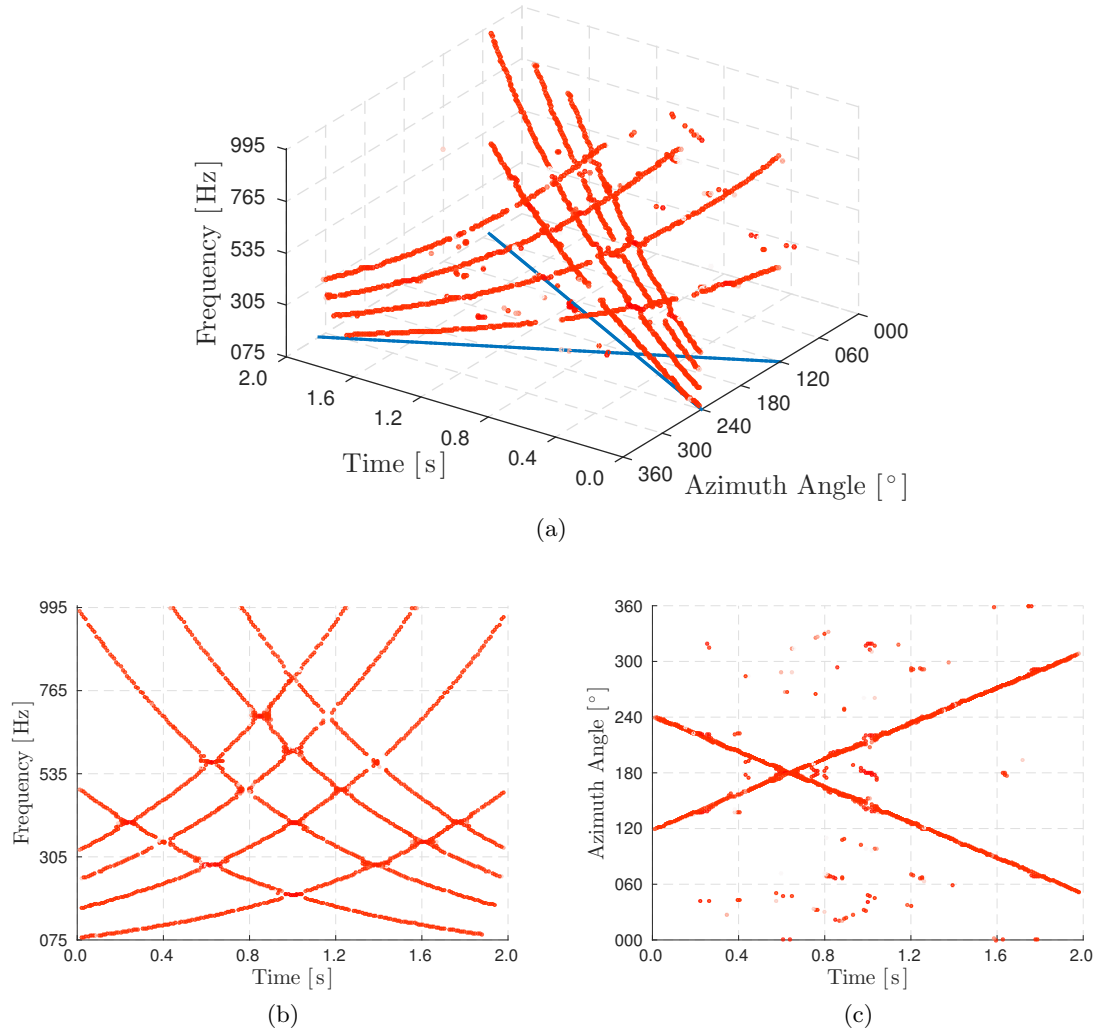


Fig. 2.10: Jointly (a) and disjointly (b,c) estimated f_0s and their second, third, and fourth harmonics, as well as DOAs of two moving, frequency-sweeping sources. The experimental parameters are as follows: initial source directions $(\varphi_s^{(1)}, \varphi_s^{(2)}) = (120°, 240°)$, source velocity $v_t = 6$ m/s, distance between the center of the microphone array and the sources $|s| = 3$ m, sampling frequency $f_s = 96$ kHz, SIR $= 0$ dB, $d_a = 0.40$ m maximum dimension of microphone array, $N_m = 8$ microphones, five sampling points ($N_d = 2$), $N_e = 8$ is the maximum number of the selected maxima, angular resolution $\Delta\varphi = 1°$, and $(3, 3)$ is the maxima detector's window size. The frequency-sweeping harmonic signals start with $f_0 = 75$ Hz and $f_0 = 500$ Hz and end at $f_0 = 500$ Hz and $f_0 = 75$ Hz, respectively. All harmonics exhibit the same amplitude. The plots in (b,c) illustrate the parameter spaces of disjoint estimates. Without prior knowledge it is impossible to associate the curves in (b) with their corresponding spatial trajectories in (c). However, in (a) these curves and trajectories are already associated to each other. The blue lines (plotted in the 75 Hz-plane) are the spatial trajectories of the angular components.

Table 2.2: Results of synthetic-data experiments with all parameters. The letters S, $\widehat{\text{S}}$, and N denote experiments with a frequency-sweeping harmonic source, a frequency-hopping harmonic source, and a noise source, respectively. The values below $P(Y > \text{R})$ are the joint recalls, $\text{R}(\varphi, f_0)$, in %. The values below $P(X \leq \text{RMSE})$ are the root-mean-square errors, $\text{RMSE}(\varphi)$, in degrees (left) and $\text{RMSE}(f_0)$ in Hertz (right). For instance, $\text{R}(\varphi, f_0) \geq 96\%$ for $P(Y > \text{R}) \approx 90\%$ implies there is a joint recall of 96% or higher in 90% of all experiments with a harmonic source plus noise source.

	$P(Y>\text{R})\approx 100\%$	$P(Y>\text{R})\approx 90\%$	$P(Y>\text{R})\approx 80\%$	$P(X\leq\text{RMSE})\approx 100\%$	$P(X\leq\text{RMSE})\approx 90\%$	$P(X\leq\text{RMSE})\approx 80\%$
S	100	100	100	≤ 1.20 / ≤ 3.20	≤ 0.90 / ≤ 3.15	≤ 0.80 / ≤ 3.10
S+N	≥ 90	≥ 96	100	≤ 5.20 / ≤ 3.60	≤ 3.90 / ≤ 3.40	≤ 2.10 / ≤ 3.30
S+S	≥ 83	≥ 85	≥ 87	≤ 2.80 / ≤ 3.50	≤ 2.00 / ≤ 3.30	≤ 1.80 / ≤ 3.20
$\widehat{\text{S}}$+N	≥ 70	≥ 90	≥ 96	≤ 5.50 / ≤ 3.75	≤ 3.60 / ≤ 1.80	≤ 2.00 / ≤ 1.60

2.13 Experimental Results

This section summarizes the results of the aforementioned experiments. I start with the outcomes of the experiments based on fully synthesized signals and close the section with the outcomes of the experiments based on real speech signals.

2.13.1 Experiments with Synthesized Signals

In each category of experiments described in the previous section, I first conducted Monte Carlo experiments with varying parameters to describe the algorithm's robustness for different settings; Fig. 2.11–2.14 show the corresponding results for each category, and Table 2.2 summarizes and highlights the important aspects of these figures. Then, I selected the best parameters, which are $f_s = 32$ kHz, $d_a = 0.40$ m, $N_m = 8$, $N_e = 16$, and $N_d = 2$, to do further experiments; Table 2.3 lists the corresponding results. Afterwards, I conducted experiments with the POPI, the NLS, and the aNLS algorithm. Table 2.4 lists the most important outcomes. It shows that the new algorithm outperforms all the others in terms of $\text{R}(\varphi, f_0)$. Fig. 2.9 (a) and (b) illustrates the SJPS over time of a non-moving harmonic source and a non-moving harmonic source plus an interfering noise source, respectively. Fig. 2.9 (c) and Fig. 2.10 show the SJPS over time of two non-moving harmonic sources and two moving harmonic sources, respectively. As illustrated in Fig. 2.11, $P(Y > \text{R}) = 100\%$ for $\epsilon \geq 0$. This means that each experiment resulted in $\text{R}(\varphi, f_0) = 100\%$. Moreover, the $\text{RMSE}(\varphi)$ and the $\text{RMSE}(f_0)$ are around 0.8° and 3 Hz, respectively. Furthermore, I considered experiments with different SNRs and SIRs. Fig. 2.12 (a) shows that the $\text{R}(\varphi, f_0) \geq 90\%$ in 100% of all experiments; the $\text{RMSE}(f_0)$ is similar to Fig. 2.11 but the range of the $\text{RMSE}(\varphi)$ is larger. As presented in Fig. 2.12 (b), the $\text{RMSE}(\varphi)$ decreases for increasing SNR. In Fig. 2.13 (a) there is $\text{R}(\varphi, f_0) \geq 83\%$ in 100% of all experiments; the $\text{RMSE}(f_0)$ is similar to Fig. 2.11 but the range of $\text{RMSE}(\varphi)$ is larger but still smaller than in Fig. 2.12 (a). Fig. 2.13 (b) reflects these observations. The remarkable differences between Fig. 2.14 and Figs. 2.11 – 2.13 are the decreased $\text{RMSE}(f_0)$s and $\text{RMSE}(\varphi)$s.

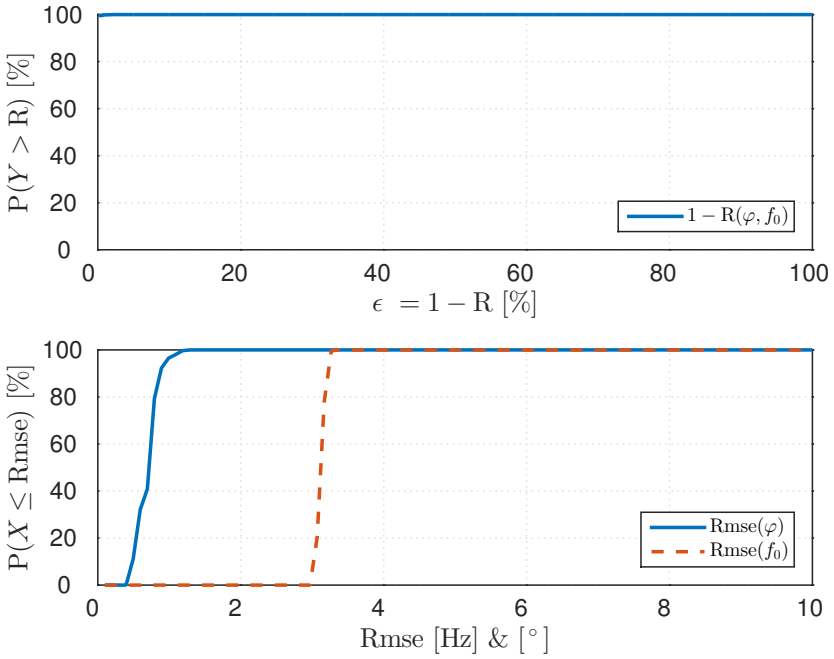


Fig. 2.11: Resulting cumulative distribution functions of an experiment with synthesized moving harmonic sources. The curves describe the probability that the opposite of R in percent (top), i.e., ($\epsilon = 1 - \mathrm{R}$), and the RMSE (bottom) of jointly estimated DOAs and f_0s has a value equal to or less than $1 - \mathrm{R}$ and RMSE.

Table 2.3: Results of synthetic-data experiments with the best parameters. The letters S, $\widehat{\mathrm{S}}$, and N denote experiments with a frequency-sweeping harmonic source, a frequency-hopping harmonic source, and a noise source, respectively. In case of the categories ($\mathrm{S} + \mathrm{N}$) or ($\hat{\mathrm{S}} + \mathrm{N}$), the first value in each column represents the averaged results of all experiments with varying SNR, the second one for SNR = 30 dB, and the third one with SNR = −10 dB. In case of category ($\mathrm{S} + \mathrm{S}$), the second value in each column represents the experimental results with SIR = 30 dB, the third with SIR = 0 dB; they are marked with a star. I set $d_a = 0.40$ m, $N_m = 8$, and $f_s = 32$ kHz.

Scenario	$\overline{\mathrm{R}}(\varphi, f_0)$ [%]			$\overline{\mathrm{RMSE}}(\varphi)$ [°]			$\overline{\mathrm{RMSE}}(f_0)$ [Hz]		
non-moving S	100			0.26			3.03		
moving S	100			0.56			3.03		
	AVG	30 dB	-10 dB	AVG	30 dB	-10 dB	AVG	30 dB	-10 dB
non-moving S+N	100	100	98	1.22	0.29	3.48	3.07	3.03	3.20
moving S+N	100	100	98	1.45	0.56	3.66	3.08	3.03	3.21
non-moving $\widehat{\mathrm{S}}$+N	97	93	97	1.17	0.11	3.23	1.17	1.08	1.58
	AVG	30 dB	0 dB	AVG	30 dB	0 dB	AVG	30 dB	0 dB
non-moving S+S	91	87*	93*	1.09	0.35*	1.56*	2.96	3.04*	3.01*
moving S+S	91	87*	94*	1.25	0.60*	1.78*	2.95	3.03*	2.98*

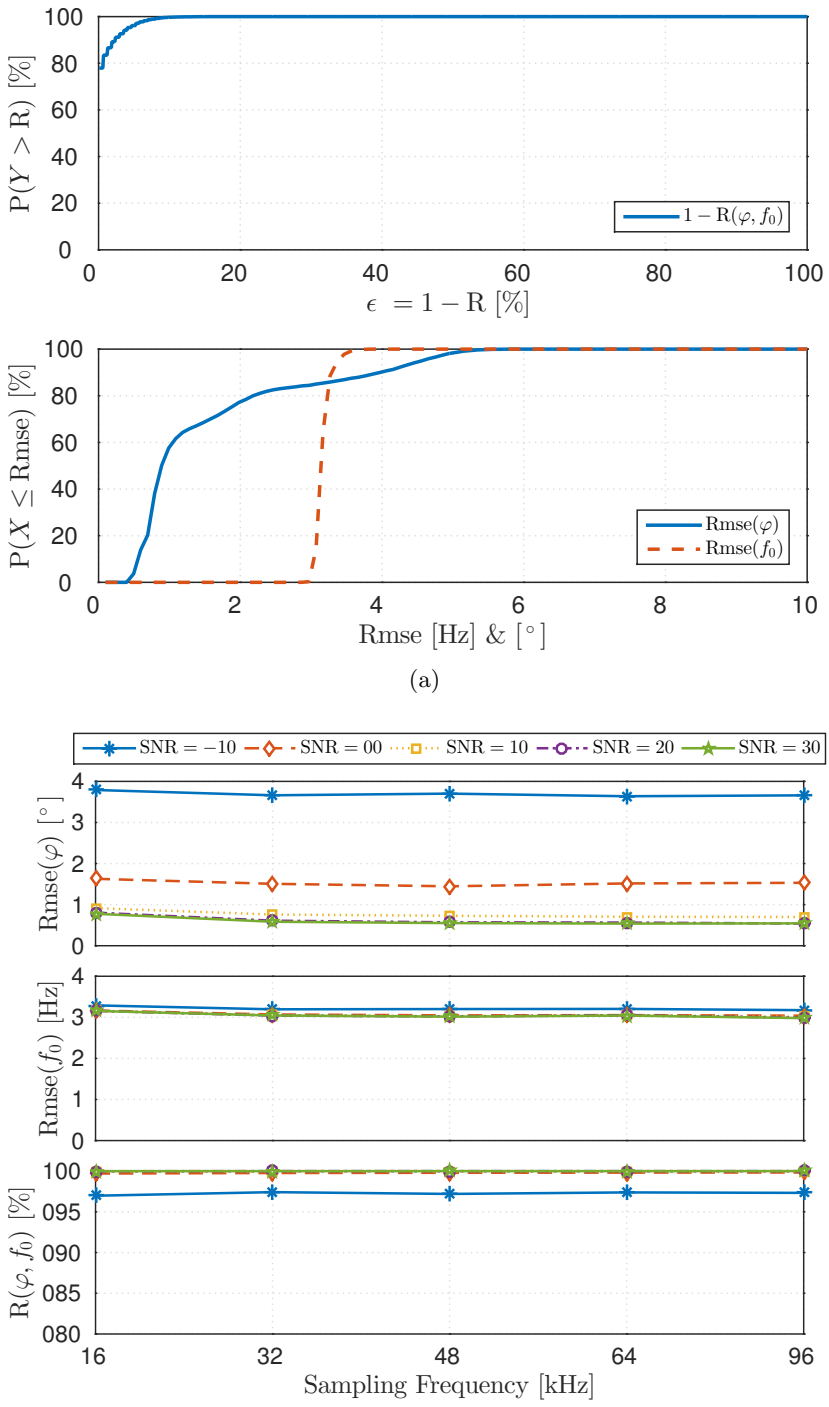


Fig. 2.12: (a) Cumulative distribution functions and (b) root-mean-square errors and joint recalls of experiments with a synthesized moving harmonic source and a noise source and with different SNRs.

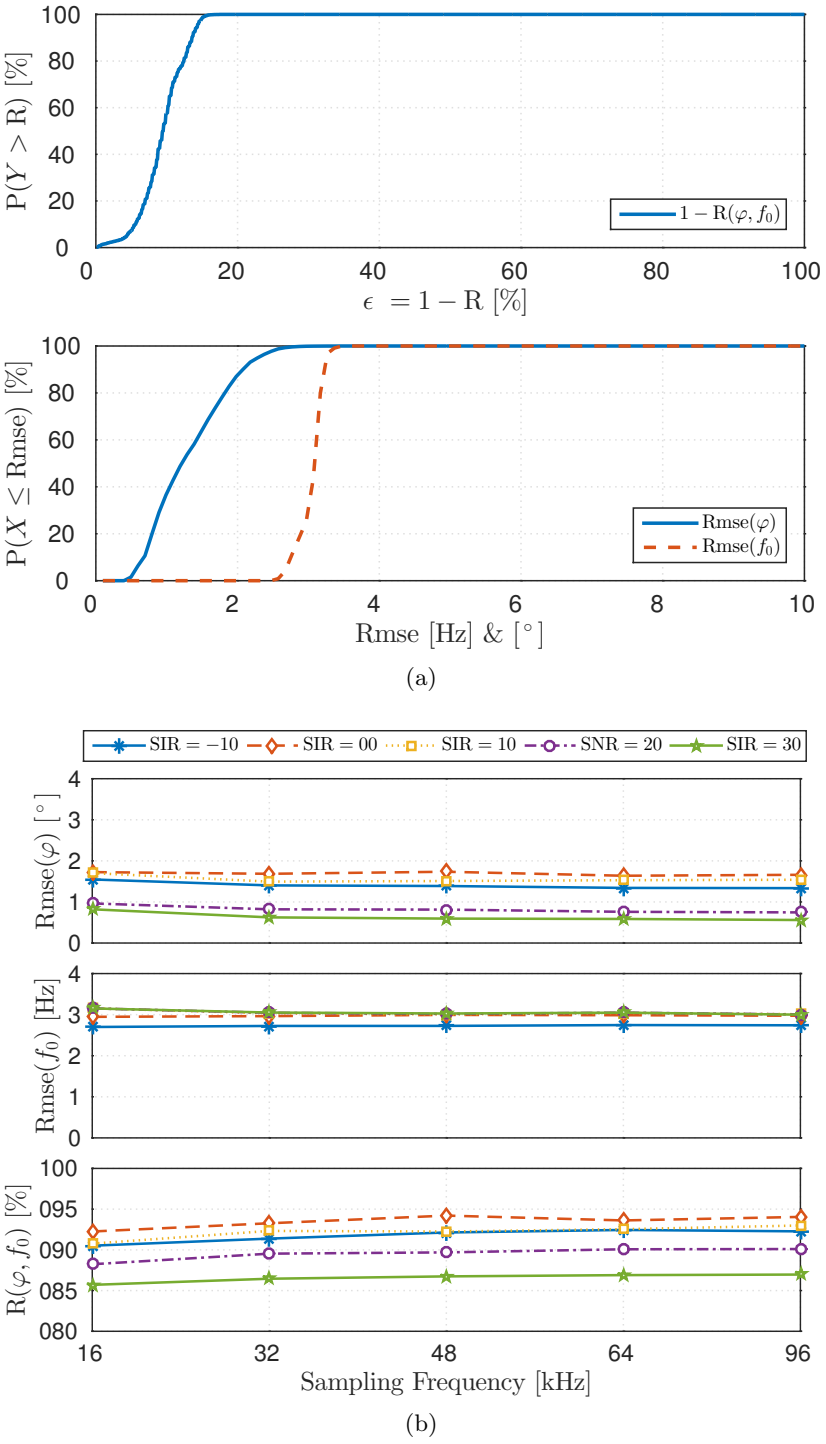
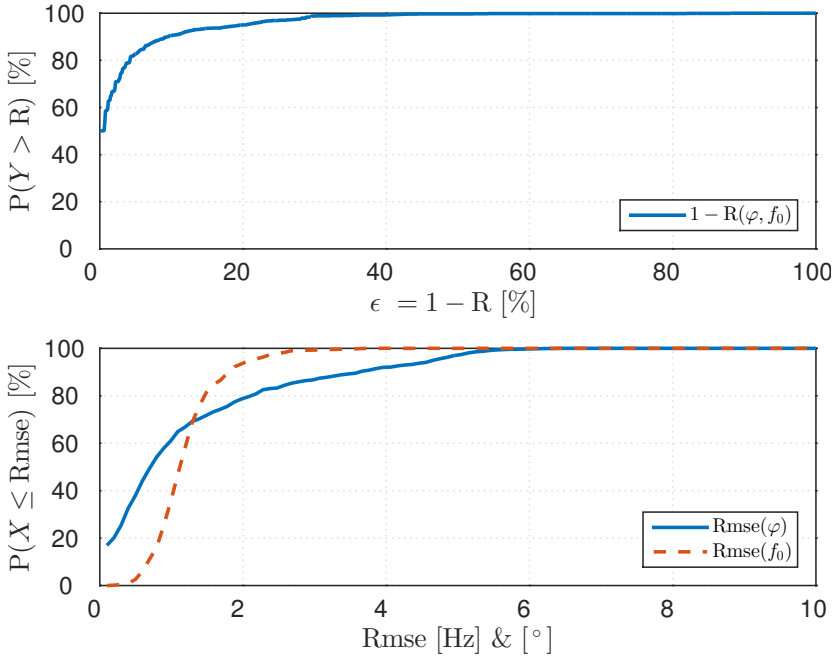
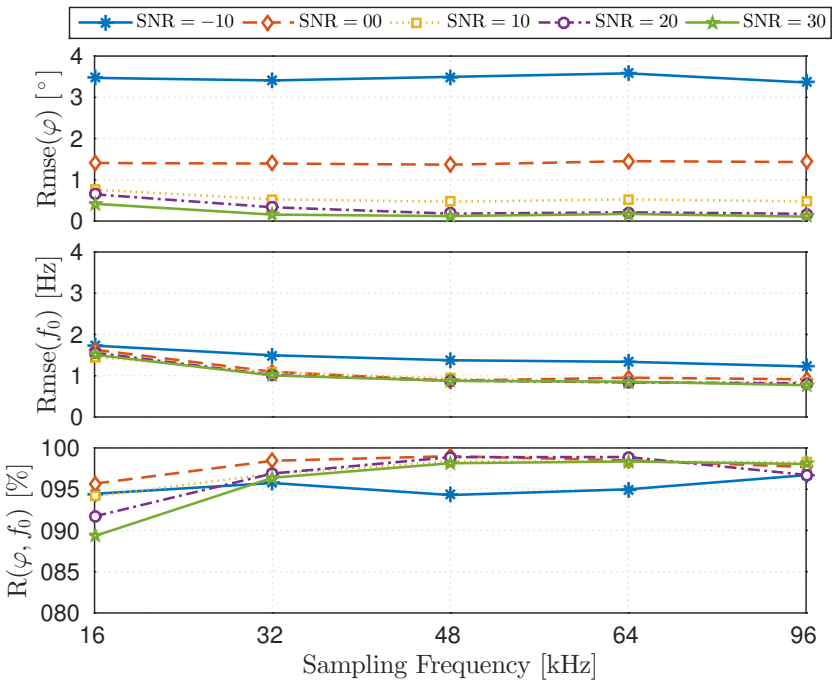


Fig. 2.13: (a) Cumulative distribution functions and (b) root-mean-square errors and joint recalls of experiments with two synthesized moving harmonic sources and with different SIRs.



(a)



(b)

Fig. 2.14: (a) Cumulative distribution functions and (b) root-mean-square errors and joint recalls of experiments with synthesized non-moving harmonic frequency-hopping sources and noise sources and with different SNRs.

Table 2.4: Results of synthetic-data experiments with different approaches: VSS (variable scale sampling), POPI (position-pitch [45]), NLS (nonlinear least squares [23]), and aNLS (approximate nonlinear least squares [23]). The table consists of three sections covering the results of experiments with a single non-moving harmonic source, a single non-moving harmonic source plus noise source, and two non-moving harmonic sources, respectively. In section two, the first column of each subsection represents the averaged results of all experiments with varying SNR, the second one with SNR = 30 dB, and the third one with SNR = −10 dB. In section three, the second column in each subsection represents the results of experiments with SIR = 30 dB, the third one with SIR = 0 dB; they are marked with a star. I set $d_a = 0.40$ m and $N_m = 8$. Note: The POPI-algorithm [35] doubles pitch periods and estimates DOAs only. As a consequence, determining a source's true fundamental frequencies using the POPI-algorithm is impossible.

Algorithm	$\overline{\mathrm{R}}(\varphi, f_0)$ [%]			$\overline{\mathrm{RMSE}}(\varphi)$ [°]			$\overline{\mathrm{RMSE}}(f_0)$ [Hz]		
VSS	**100**			0.26			3.03		
POPI	**100**			**0.01**			3.00		
NLS	51			3.80			**0.39**		
aNLS	41			3.53			1.23		
	AVG	30 dB	-10 dB	AVG	30 dB	-10 dB	AVG	30 dB	-10 dB
VSS	**100**	**100**	**98**	1.22	0.29	3.48	3.07	3.03	3.20
POPI	**100**	**100**	**98**	**0.30**	**0.02**	**1.64**	3.03	3.01	**3.05**
NLS	43	58	2	4.34	3.75	6.34	**2.05**	**0.59**	5.30
aNLS	32	44	2	4.13	3.48	5.54	2.24	1.27	5.74
	AVG	30 dB	0 dB	AVG	30 dB	0 dB	AVG	30 dB	0 dB
VSS	**91**	**87***	**93***	**1.09**	0.35*	**1.56***	2.96	3.04*	**3.01***
POPI	66	52*	92*	1.35	**0.03***	2.26*	3.32	3.00*	3.17*
NLS	25	29*	10*	4.46	3.89*	5.21*	**2.29**	**0.63***	3.56*
aNLS	19	22*	8*	4.21	3.52*	5.49*	2.33	1.23*	3.96*

2.13.2 Experiments with Real Speech Signals

I initially conducted Monte Carlo experiments to determine the best parameters, which are as follows: $f_s = 32$ kHz, $N_e = 16$, and $N_d = 2$. After selecting the best setting of parameters, I continued conducting Monte Carlo experiments by randomly selecting speech recordings. Due to a fixed setting of parameters and environmental properties, I show $P(X \leq \text{RMSE})$ and $P(Y > \text{R})$ only. In comparison to Fig. 2.11–2.14, Fig. 2.15 and Fig. 2.16 additionally feature $\text{R}(\varphi)$ and $\text{R}(f_0)$.

2.14 Discussion

In this section, I discuss the outcomes of the experiments with synthesized signals and the results of the experiments with real speech signals.

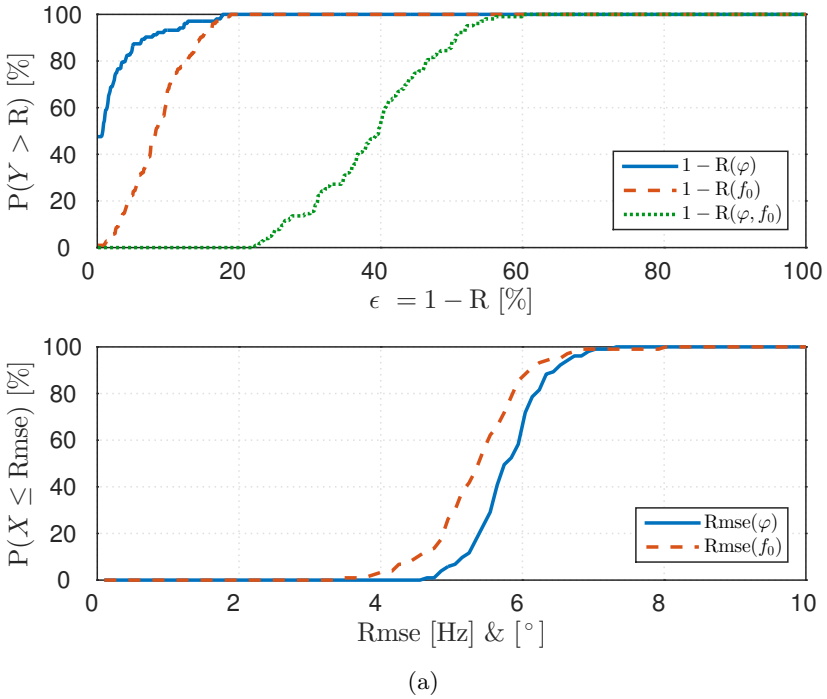
2.14.1 Synthesized Signals

In the first category (single harmonic source) the proposed algorithm achieves a $\text{R}(\varphi, f_0) = 100\%$ in each Monte Carlo experiment as shown in Fig. 2.11, Table 2.2, and Table 2.3. The algorithm perfectly solves the problem of jointly estimating the DOAs and the f_0s of a single harmonic source while keeping the $\text{RMSE}(\varphi)$ and the $\text{RMSE}(f_0)$ low.

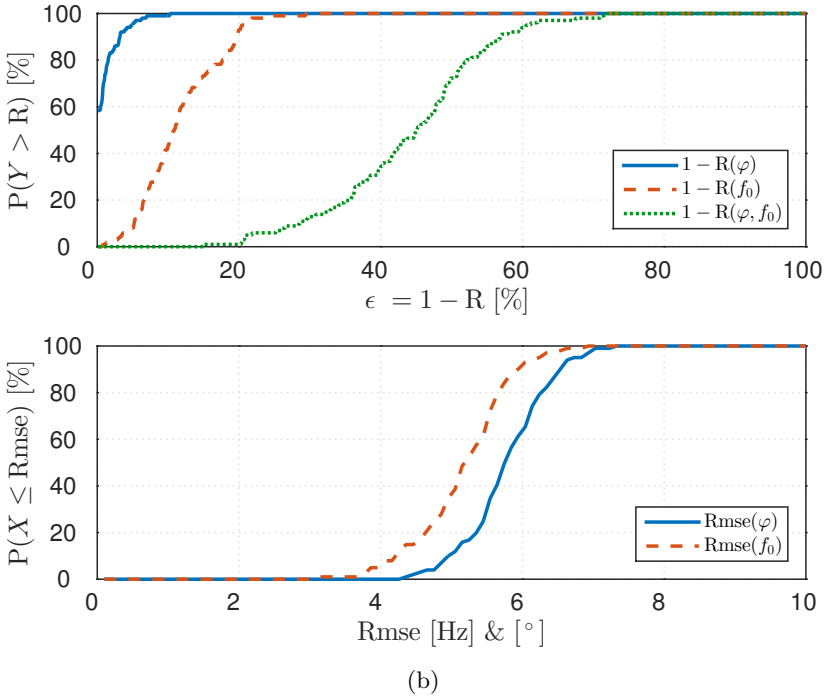
In the second category (single harmonic source plus noise source) the algorithm achieves a $\text{R}(\varphi, f_0) = 100\%$ in experiments with $\text{SNR} \geq 0$ dB. As shown in Fig. 2.12 the recall starts decreasing for $\text{SNR} < 0$ dB, which highlights the robustness against noise sources exhibiting the same or lower power as the harmonic source of interest. Table 2.2 supports this statement by showing that in 80% and 100% of all experiments the algorithm achieves an $\text{R}(\varphi, f_0) = 100\%$ and an $\text{R}(\varphi, f_0) \geq 90\%$. Table 2.2 emphasizes the algorithm's robustness for experiments with an $\text{SNR} \geq 0$ dB; the $\text{RMSE}(\varphi)$ and the $\text{RMSE}(f_0)$ are still low.

In the third category (two harmonic sources) the proposed algorithm features, as shown in Fig. 2.13, lower RMSEs than in the previous one; however, $\overline{\text{R}}(\varphi, f_0)$ is lower than in all other categories. This is due to the beating effect during crossings of frequencies shown in Fig. 2.9 (c) at 0.97 s (left) and at 240° and 0.5 s (right) as well as in Fig. 2.10 (b) at 0.4 s and 330 Hz, at 1.18 s and 730 Hz, and at 1.4 s and 550 Hz. This effect causes destructive interference of the superimposed signals. When estimating both harmonic sources, the proposed algorithm achieves the highest $\overline{\text{R}}(\varphi, f_0)$ when $\text{SIR} = 0$ dB, because the signals of both sources exhibit the same power, i.e., they are equally present. In case of $\text{SIR} = 30$ dB, one source dominates the other, which is problematic if both sources are spatially close to each other. The results in case of $\text{SIR} = \pm 10$ dB are almost identical, because one source is dominating the other one.

In comparison to the previous category and as shown in Fig. 2.14, the $\overline{\text{R}}(\varphi, f_0)$ in the fourth category (non-sweeping harmonic source plus noise source) is lower at $f_s = 16$ kHz due to the signals' characteristics and the lower frequency resolution at higher frequencies. In this category, the f_0s to-be-estimated are constant over a long period of time. The $\text{RMSE}(f_0)$s increase if the ground truth of f_0 exhibits a value at higher frequencies and if the ground-truth value is not an element of the frequency grid defined by (2.11). The RMSEs are smaller than in case of frequency-sweeping harmonic sources because the signals' f_0s are constant over certain time intervals and, though



(a)



(b)

Fig. 2.15: Cumulative distribution functions of experiments with two-channel recordings of (a) a female speaker and (b) a male speaker.

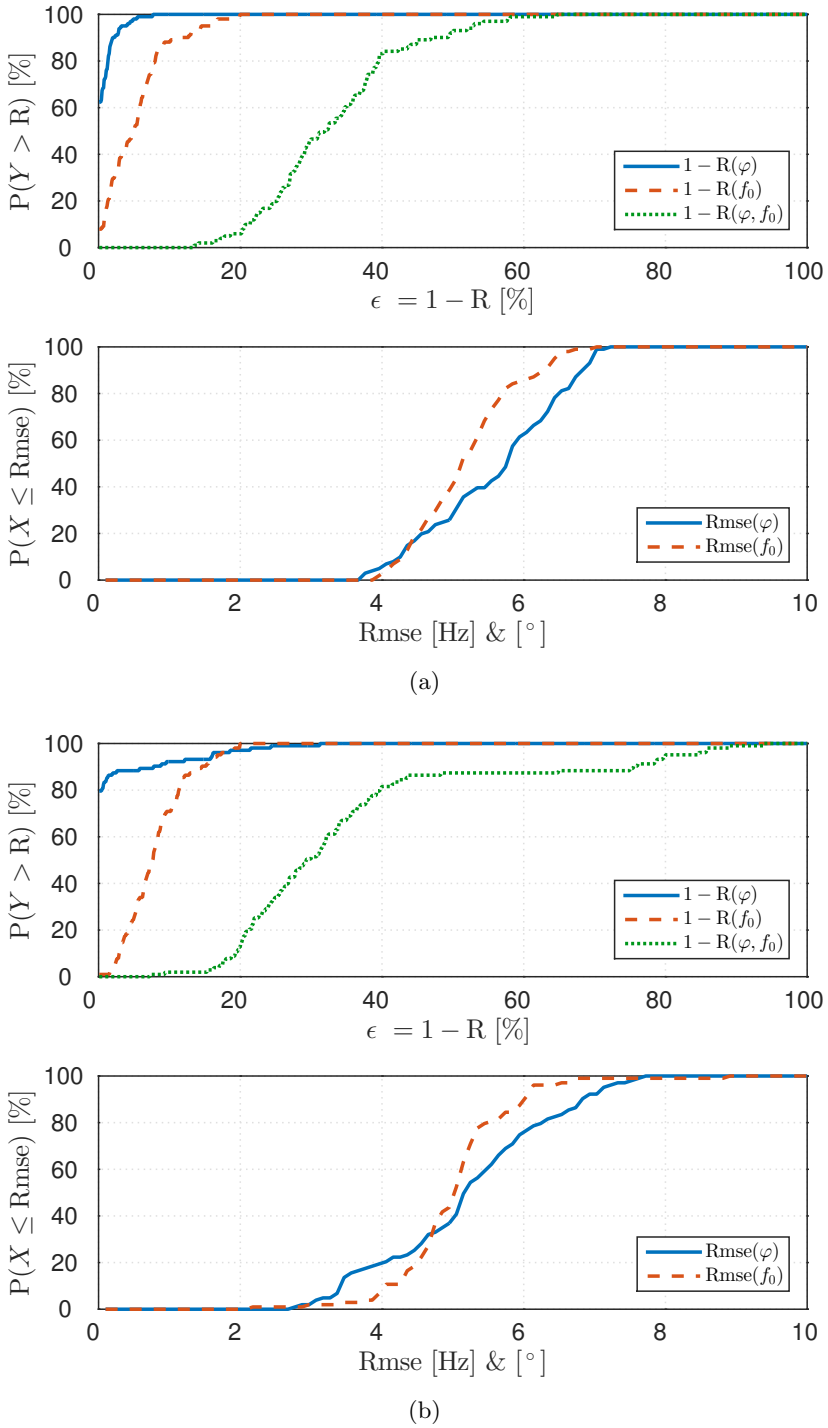


Fig. 2.16: Cumulative distribution functions of experiments with three-channel recordings of (a) a female speaker and (b) a male speaker.

being uniformly distributed in the frequency range, they occur more often in a range where the frequency resolution is approximately constant. Comparing with the results listed in [23], the proposed algorithm does not outperform the joint estimator presented in [23]. In this specific category and regarding the RMSE, I show that my algorithm works with harmonic signals based on a musical instrument's signal model.

So far, the results show that the RMSE(f_0) does not fall below 2.9 Hz in categories one, two, and three. This is due to several reasons: First, the finite number of sampling periods causes a quantization error. Second, increasing the Shah function's sampling interval linearly and sample-by-sample in the lag domain corresponds to a nonlinear decrease of the frequency interval in the frequency domain ($f = 1/T$). Thus, the quantization intervals in the frequency domain get larger to higher frequencies. Third, I generated the source signals sample-by-sample in time domain using (1.3). However, I defined the instantaneous frequency in the center of each frame as the ground-truth value, because the proposed algorithm estimates f_0s frame-by-frame. Fourth, the rounding of sampling periods to the nearest integer causes increasing errors to higher frequencies. Furthermore, the results show that noise mainly affects the estimation of DOA.

Comparisons With Other Algorithms

As listed in Table 2.4, the proposed algorithm outperforms or compares favorably with the other algorithms in all three categories in terms of $\overline{\mathrm{R}}(\varphi, f_0)$, especially in experiments with two sources. The NLS algorithm as well as the aNLS algorithm are unable to estimate parameters of two or more sources. Their accuracy decreases for low SNRs and SIRs. The modified POPI algorithm performs better, however, as soon as one source dominates the other, its estimation accuracy decreases. Focusing on $\overline{\mathrm{RMSE}}(\varphi)$, the proposed algorithm outperforms all other algorithms in experiments with two harmonic sources. In case of a single harmonic source, the modified POPI algorithm achieves the smallest $\overline{\mathrm{RMSE}}(\varphi)$, which is due to the use of a single broadband CCF; it exhibits a narrow peak at the lag corresponding to the dominant source's DOA. The NLS exhibits the smallest $\overline{\mathrm{RMSE}}(f_0)$, which corresponds to the findings reported in [23]. The proposed algorithm achieves $\overline{\mathrm{RMSE}}(f_0) \approx 3$ Hz; this is mostly due to the decreasing frequency resolution for increasing frequencies. According to [23], the ideal NLS estimator is a maximum likelihood estimator that attains the Cramér-Rao bound in case of single-source experiments with white Gaussian noise. In such experiments, it should outperform all other algorithms, but this was not the case due to the following reasons: The aforementioned statement is true if I would evaluate the cost function for all f_0 candidates and DOA candidates and search for the global maximum. However, the authors of [23] presented a version based on gradient ascent, which may converge to the global maximum (the true DOA and f_0) or to local maxima (with wrong f_0s) depending on the initial values and the employed line search algorithm. Furthermore, due to a finite number of iterations in order to compute the coefficients, the algorithm sometimes failed to reach the correct DOA. Moreover, I applied uniformly distributed white noise instead of white Gaussian noise; and, unlike [23], I employed frequency-sweeping signals. To sum it up, the proposed algorithm is able to jointly estimate the DOAs and f_0s of two or more harmonic sources, whereas the others can cope with a single source only or they focus on the dominant source.

2.14.2 Real Speech Signals

This set of experiments employs speech signals recorded in a real environment featuring, e.g., reverberation, (strong) multi-path components, and non-harmonic components like plosives, fricatives, and noise. Despite these challenging characteristics, the figures show, however, that the algorithm successfully localizes and characterizes sources using two or three microphones. Thus, the joint estimation of parameters and their representation in an SJPS introduces new possibilities to further process the parameters in a higher-dimensional sense in a real environment. Besides, evaluating DOA and f_0 disjointly by using the proposed algorithm yields even better results than estimating them jointly. However, estimating parameters disjointly requires an additional step, the data association, which in turn requires a certain amount of prior knowledge. In Fig. 2.15 and Fig. 2.16, one can see that, sometimes, $P(Y > \mathrm{R}(\varphi)) = 100\%$ and $P(Y > \mathrm{R}(f_0)) = 100\%$, but $P(Y > \mathrm{R}(\varphi, f_0)) < 100\%$. This is true because $\mathrm{R}(f_0) = \mathrm{R}(\varphi) = 100\%$ does not necessarily imply that $\mathrm{R}(\varphi, f_0) = 100\%$. For instance, if there is one reference item and if there are two estimated items, one matching the true f_0 only and the other one matching the true φ only, then $\mathrm{R}(\varphi, f_0) = 0\%$, although $\mathrm{R}(f_0) = \mathrm{R}(\varphi) = 100\%$. A captured signal's frame contained direct-path and multi-path components of a source. Due to the reverberation room's memory effect, the frequencies of the multi-path components slightly differed from the frequencies of the direct-path components within a time frame, because the true f_0 continuously changes as a function of time. Additionally, sometimes the multi-path components dominated in energy. These effects introduced small errors in f_0. Focusing on the joint estimation of DOAs and f_0s, both effects mentioned above decreased the R. Regarding RMSE one can see that $\overline{\mathrm{RMSE}}(f_0) < \overline{\mathrm{RMSE}}(\varphi)$, which is opposite to the experiments with synthetic data. This is again due to the multi-path components.

Chapter 3

Joint Estimator Based on Relative Phase-Delay Masking[1]

In this chapter, I present my second approach to localize and characterize one or more harmonic sources. It is based on the first approach, but features significant differences in the stage of jointly extracting parameters. Fig. 3.1 shows the block diagram of the new algorithm. I will discuss its components shown in Fig. 3.1 (bottom), which differ from the previously described approach based on variable-scale sampling.

3.1 Contributions and Innovations

The proposed algorithm sparsifies a (quasi-continuous) joint parameter space (JPS) by using the chirp z-transform (CZT), the relative phase-delay masking (RPDM), an optimized filter bank, and a multi-dimensional maxima detector. It is inspired by [28–36] and based on [45, 49]. As compared with my the VSS-based algorithm, the novel approach features several innovations making it more accurate and resource-efficient: First, an invariant frequency resolution guarantees the same conditions for estimating frequency components of higher-pitched and lower-pitched harmonic sources; now, on average, the DOAs and f_0s of female speakers are as accurate as the DOAs and f_0s of male speakers. Second, it employs a CZT instead of a discrete Fourier transform (DFT) which enables us to compute high-resolution cross-spectra of small frequency ranges of interest with a lower sampling frequency f_s. Third, I avoid (back-)transforming cross-spectra into CCFs in the lag domain to compute unbiased CCFs. Fourth, I circumvent setting up a comprehensive lookup table. Fifth, I introduce a technique based on a single tolerance parameter making the new algorithm more robust against small phase mismatches.

3.2 Chirp z-Transform

In the late 60s, Bluestein [87] and Rabiner et al. [88] published articles about the CZT. They describe how to compute the z-transform at N_a points lying on a contour in the

[1] This chapter is substantially based on the submitted journal paper [86] and was revised and adapted to the present thesis. As first author of the journal paper, I did everything on my own except the implementation of the aNLS algorithm and the NLS algorithm [23] implemented by Mattia Gabbrielli.

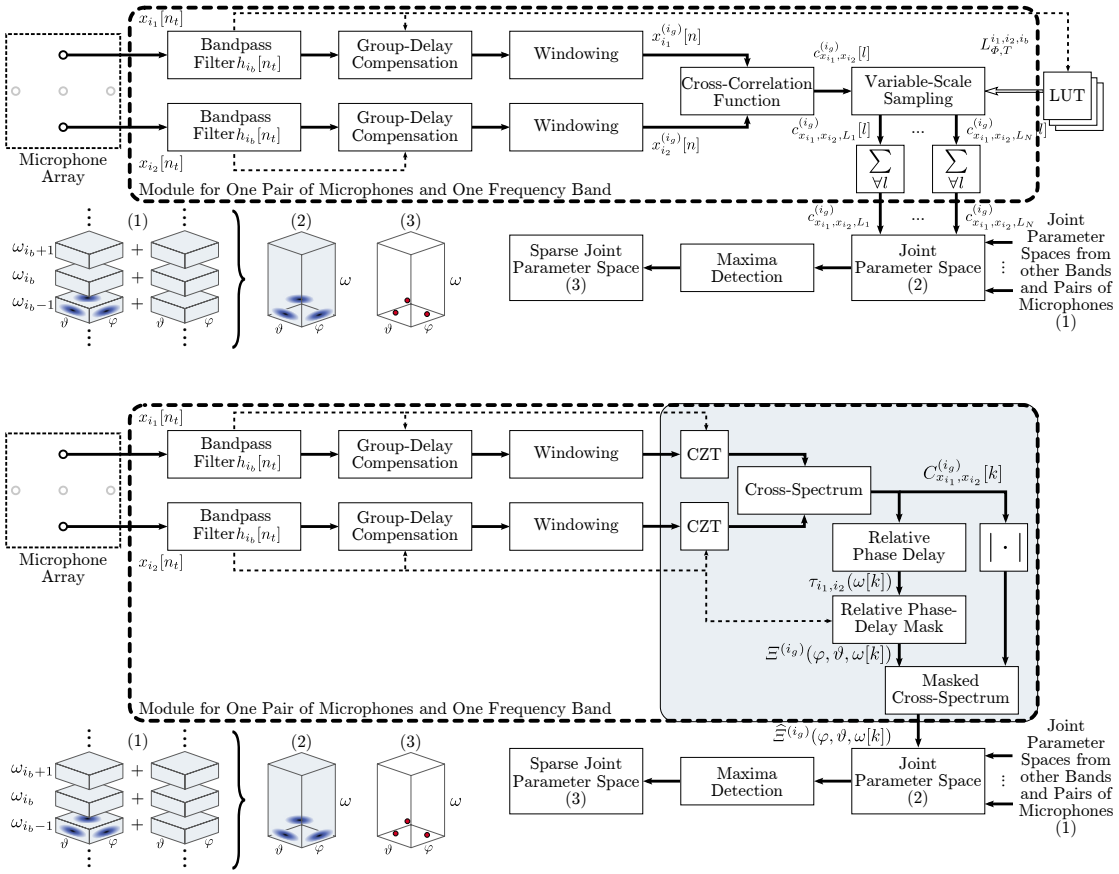


Fig. 3.1: Block-diagrams of algorithms based on (top) variable-scale sampling [49] and (bottom) the chirp z-transform (CZT) and relative phase-delay masking (RPDM). All components inside the outer dashed rectangles belong to a module for one pair of microphones and one frequency band. The number of modules depends on the number of available pairs of microphones and the number of frequency bands. The components in the filled rectangle highlight the differences between the present approach and the approach presented in Chapter 2. The components labeled with 'Windowing' split the discrete-time signals $x_{i_1}[n_t]$ and $x_{i_2}[n_t]$ from microphones with index i_1 and i_2 into frames; n_t is the sample index of the whole captured signal and n is the sample index of a windowed signal. Variable $h_{i_b}[n_t]$ is the impulse response of the i_b-th bandpass filter, φ is the azimuth, and ϑ denotes the elevation. In (top), $c^{(i_g)}_{x_{i_1}x_{i_2}}[l]$ is the cross-correlation function (CCF) of the frames $x^{(i_g)}_{i_1}[n]$ and $x^{(i_g)}_{i_2}[n]$ with lag-index l, $c^{(i_g)}_{x_{i_1}x_{i_2},L_{i_3}}[l]$ is the CCF sampled with a certain sampling period and sampling phase (both represented by L_{i_3}), $c^{(i_g)}_{x_{i_1}x_{i_2},L_{i_3}}$ is the sampled CCF summed over all lags, $L^{(i_1,i_2,i_b)}_{\Phi,T}$ denotes the subset of sampling phases and sampling periods for the i_b-th band and the microphones labeled with i_1 and i_2. The lookup table (LUT) contains all relevant indices for variable-scale sampling. In (bottom), $C^{(i_g)}_{x_{i_1}x_{i_2}}[k]$ is the cross-spectrum of the frames $x^{(i_g)}_{i_1}[n]$ and $x^{(i_g)}_{i_2}[n]$ with spectral index k, $\tau_{i_1,i_2}(\omega[k])$ is the relative phase delay with angular frequency $\omega[k]$, $\Xi^{(i_g)}(\phi,\theta,\omega[k])$ is the RPDM, and $\widehat{\Xi}^{(i_g)}(\phi,\theta,\omega[k])$ represents the masked cross-spectrum's magnitudes.

z-plane and how to analyze a narrow-band frequency spectrum with high resolution. The spiral-shaped or circular contour can start at any point, z, in the z-plane. To define contours for different bands with index i_b, I set the normalized angular starting point β_{s,i_b}, the normalized angular spacing β_{a,i_b}, and the number of points, N_a, on the contour. For a contour that starts at the angle β_{s,i_b} and ends at the angle $\beta_{s,i_b} + (N_a - 1)\beta_{a,i_b}$, I write

$$z_{k,i_b} = A_{i_b} \cdot W_{i_b}^{-k}, \tag{3.1}$$

with $k = \{0, 1, \ldots, N_a - 1\}$ as the i_b-th band's CZT-index. The parameter A_{i_b} is the complex-valued starting point in the z-plane with its radius A_0,

$$A_{i_b} = A_0 \cdot \mathrm{e}^{j2\pi\beta_{s,i_b}}, \tag{3.2}$$

and W_{i_b} defines if the contour spirals in or out with respect to the origin and depending on a parameter W_0,

$$W_{i_b} = W_0 \cdot \mathrm{e}^{-j2\pi\beta_{a,i_b}}. \tag{3.3}$$

The normalized angular starting point is

$$\beta_{s,i_b} = f_{\min}^{(i_b)}/f_s, \tag{3.4}$$

and the normalized angular spacing is

$$\beta_{a,i_b} = (f_{\max}^{(i_b)} - f_{\min}^{(i_b)})/((N_a - 1) \cdot f_s), \tag{3.5}$$

where f_s is the sampling frequency, $f_{\min}^{(i_b)}$ and $f_{\max}^{(i_b)}$ denote a band's lowest and highest f_0 of interest, respectively. For $A_{i_b} = 1$, $W_0 = 1$, $N = N_a$, and $\beta_{a,i_b} = 1/N$, where N is the number of samples of a sequence $(x[n])_{n\in N}$, the resulting CZT is identical to the DFT. The general form of the CZT is

$$X(z_{k,i_b}) = \sum_{n=0}^{N-1} x[n] z_{k,i_b}^{-n}. \tag{3.6}$$

I rewrite (3.6) by inserting (3.1) into (3.6), which yields

$$X(z_{k,i_b}) = \sum_{n=0}^{N-1} x[n] A_{i_b}^{-n} W_{i_b}^{nk}. \tag{3.7}$$

For localizing and characterizing harmonic sources, I am interested in a very small arc of the z-plane's unit circle; thus, I set $\beta_{s,i_b} > 0$, $A_0 = 1$, $N_a < N$, and consider Bluestein's substitution [87–89],

$$nk = \frac{n^2 + k^2 - (k - n)^2}{2}, \tag{3.8}$$

which results in

$$X^{(i_b)}[k] = X(z_{k,i_b})|_{|z_{k,i_b}|=1} \tag{3.9}$$

or

$$X^{(i_b)}[k] = \mathrm{e}^{j\pi\beta_{a,i_b}k^2} \sum_{n=0}^{N-1} x[n]\mathrm{e}^{-j2\pi\beta_{s,i_b}n}\mathrm{e}^{j\pi\beta_{a,i_b}n^2}\mathrm{e}^{-j\pi\beta_{a,i_b}(k-n)^2}. \tag{3.10}$$

The angular frequencies corresponding to indices k are

$$\omega_{i_b}[k] = 2\pi \left(k\frac{f_{\max}^{(i_b)} - f_{\min}^{(i_b)}}{N_a - 1} + f_{\min}^{(i_b)} \right), \tag{3.11}$$

See [71,90] for efficient implementations of the CZT based on the fast Fourier transform. The next step is to compute a cross-spectrum of two chirp-z transformed sequences.

3.3 Cross-Spectrum

For a cross-spectrum of the CZT of two sequences $(x_{i_1}[n])_{n\in N}$ and $(x_{i_2}[n])_{n\in N}$, where i_1 and i_2 are the microphone indices and $(\cdot)^*$ denotes a complex conjugation, I write

$$C_{x_{i_1}x_{i_2}}^{(i_b)}[k] = X_{i_1}^{(i_b)}[k]X_{i_2}^{*(i_b)}[k] \tag{3.12}$$

with

$$X_i^{(i_b)}[k] = \mathrm{e}^{j\pi\beta_{a,i_b}k^2} \sum_{n=0}^{N-1} x_i[n]\mathrm{e}^{-j2\pi\beta_{s,i_b}n}\mathrm{e}^{j\pi\beta_{a,i_b}n^2}\mathrm{e}^{-j\pi\beta_{a,i_b}(k-n)^2}. \tag{3.13}$$

From that, I derive the relative phase delay for the relative phase-delay masking.

3.4 Relative Phase Delay

In general, the phase delay is a measure of the time delay (in seconds) corresponding to a signal's phase shift or, in case of the cross-spectrum, the phase difference. To transform the phase difference of a cross-spectrum's complex-valued component to TDOAs in seconds, I divide the negative phase,

$$-\phi_{i_1,i_2}^{(i_b)}[k] = -\angle C_{x_{i_1}x_{i_2}}^{(i_b)}[k] \tag{3.14}$$

wrapped to $[-\pi, +\pi]$, by the angular frequency $\omega_{i_b}[k]$:

$$\tau_{i_1,i_2}(\omega_{i_b}[k]) = -\frac{\phi_{i_1,i_2}^{(i_b)}[k]}{\omega_{i_b}[k]}. \tag{3.15}$$

As described in [91], the delay in time domain, i.e., the TDOA, corresponds to a phase shift or phase rotation in frequency domain. Computing the maximum argument of the inverse discrete Fourier-transformed cross-spectrum's phase (or phase spectrum) yields the delay estimate in a single-source scenario. Thus, the cross-spectrum's phase (delay) relates to the TDOA; computing the cross-spectrum's group delay is, therefore, unnecessary.

The next step is to compute a relative phase-delay mask.

3.5 Relative Phase-Delay Mask

Before defining the RPDM, I compute frequency-independent TDOAs,

$$\bar{\tau}_{i_1,i_2}(\varphi,\vartheta) = -(\mathbf{m}_{i_1} - \mathbf{m}_{i_2})^T \mathbf{k}(\varphi,\vartheta)/v, \tag{3.16}$$

for all directions of interest, (φ,ϑ), with

$$\mathbf{k}(\varphi,\vartheta) = (\sin(\vartheta)\cos(\varphi), \sin(\vartheta)\sin(\varphi), \cos(\vartheta))^T \tag{3.17}$$

as the spherical unit vector, φ and ϑ as the azimuth and elevation, $\mathbf{m}_{i_1}$ and $\mathbf{m}_{i_2}$ as the i_1-th and i_2-th microphone coordinates, and v as the speed of sound. Due to calibration errors, spherical wave propagation (deviating from the plane wave propagation assumption), and non-ideal acoustic point sources, the estimated TDOAs, $\tau_{i_1,i_2}(\omega_{i_b}[k])$, will rarely match the exact values of the ideal TDOAs, $\bar{\tau}_{i_1,i_2}(\varphi,\vartheta)$. Thus, a lookup-table, as utilized in the predecessor, is always inaccurate. However, a mask helps us to overcome these issues, and it introduces robustness. Consequently, I have to consider intervals of TDOAs, $[\bar{\tau}^{(-)}_{i_1,i_2}(\varphi,\vartheta), \bar{\tau}^{(+)}_{i_1,i_2}(\varphi,\vartheta)]$, where

$$\bar{\tau}^{(-)}_{i_1,i_2}(\varphi,\vartheta) = \min\left[\bar{\tau}_{i_1,i_2}(\varphi-\varepsilon_\varphi,\vartheta-\varepsilon_\vartheta), \bar{\tau}_{i_1,i_2}(\varphi+\varepsilon_\varphi,\vartheta+\varepsilon_\vartheta)\right] \tag{3.18}$$

and

$$\bar{\tau}^{(+)}_{i_1,i_2}(\varphi,\vartheta) = \max\left[\bar{\tau}_{i_1,i_2}(\varphi-\varepsilon_\varphi,\vartheta-\varepsilon_\vartheta), \bar{\tau}_{i_1,i_2}(\varphi+\varepsilon_\varphi,\vartheta+\varepsilon_\vartheta)\right] \tag{3.19}$$

with ε_φ and ε_ϑ as the DOA-tolerance parameters that introduce the robustness. In practice, $0 \leq \varepsilon_\varphi, \varepsilon_\vartheta \leq \xi_r$, where ξ_r depends on the system's mismatches and the required accuracy. After defining these intervals (intervals for a certain ε are shown in Fig. 3.2), I assign triples of parameters, $(\varphi,\vartheta,\omega_{i_b}[k])$, to the estimated TDOAs, $\tau_{i_1,i_2}(\omega_{i_b}[k])$, which results in the binary RPDM:

$$\Xi^{(i_b)}_{i_1,i_2}(\varphi,\vartheta,\omega_{i_b}[k]) = \begin{cases} 1, & \bar{\tau}^{(-)}_{i_1,i_2}(\varphi,\vartheta) \leq \tau_{i_1,i_2}(\omega_{i_b}[k]) \leq \bar{\tau}^{(+)}_{i_1,i_2}(\varphi,\vartheta) \\ 0, & \text{else} \end{cases}. \tag{3.20}$$

Assuming ideal algorithmic and environmental conditions, the binary RPDM is sparse with values unequal zero at indices corresponding to $(\varphi_0,\vartheta_0,\omega_{i_q}[k])^{(i_p)}$, where φ_0, ϑ_0, and $\omega_{i_q}[k]$ is the azimuth angle, the elevation angle, and the i_q-th harmonic of the i_p-th source, respectively. However, due to, e.g., reverberant environments, I get a sparse binary RPDM with values unequal zero at indices corresponding to $(\varphi_{i_\varphi},\vartheta_{i_\vartheta},\omega_{i_\omega}[k])^{(i_p)}$, where ω_{i_ω} is a harmonic or inharmonic component, and φ_{i_φ} and ϑ_{i_ϑ} denote the azimuth and elevation of a direct path or reflected component, respectively. The captured components' indices are i_φ, i_ϑ, and i_ω. The resulting binary RPDM consists of finite regions, where each region includes the estimate of the true item. To determine this estimate, I weight the magnitudes of the cross-spectrum, $|C^{(i_b)}_{x_{i_1}x_{i_2}}[k]|$, by the elements of the binary RPDMs yielding masked cross-spectrum magnitudes.

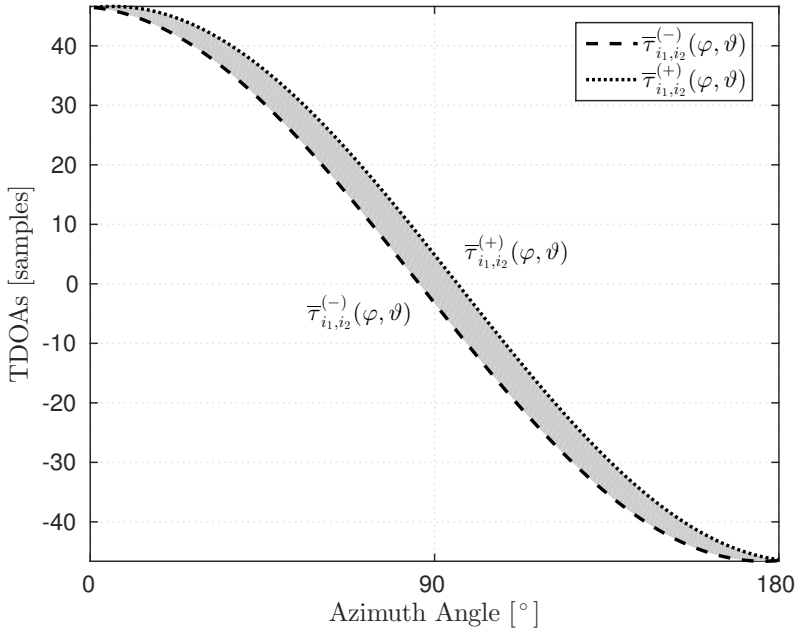


Fig. 3.2: Physically possible time delays of arrival (TDOAs) (in samples) for all azimuthal angles of interest, and a pair of omni-directional microphones placed on the y-axis of the coordinate system (shown in Fig. 1.2) with a microphone-spacing of 0.5 m, a sampling frequency of 32 kHz, and DOA-tolerances of $\varepsilon_\varphi = 5°$ and $\varepsilon_\vartheta = 0°$. The DOA-tolerance defines the size of the colored area and the robustness of the algorithm. For instance, the larger the tolerance value, the larger the colored area and the larger the robustness. If the measured TDOA for a certain frequency, ω, i.e., $\tau_{i_1,i_2}(\omega_{i_b}[k])$, is inside the colored area spanned by the two curves shown above, the relative phase-delay mask will get values equal to one at frequencies and angles which correspond to the TDOAs inside the area.

3.6 Masked Cross-Spectrum Magnitudes

To obtain the weighted RPDM, I multiply each RPDMs non-zero element by the corresponding magnitude of the cross-spectrum according to

$$|\widehat{C}|_{i_1,i_2}^{(i_b)}(\varphi,\vartheta,\omega_{i_b}[k]) = |C_{x_{i_1}x_{i_2}}^{(i_b)}[k]| \cdot \Xi_{i_1,i_2}^{(i_b)}(\varphi,\vartheta,\omega_{i_b}[k]) \tag{3.21}$$

I rewrite (3.21) by combining (3.20) and (3.21), which yields the masked cross-spectrum magnitude or weighted RPDM:

$$|\widehat{C}|_{i_1,i_2}^{(i_b)}(\varphi,\vartheta,\omega_{i_b}[k]) = \begin{cases} |C_{x_{i_1}x_{i_2}}^{(i_b)}[k]|, & \bar{\tau}_{i_1,i_2}^{(-)}(\varphi,\vartheta) \leq \tau_{i_1,i_2}(\omega_{i_b}[k]) \leq \bar{\tau}_{i_1,i_2}^{(+)}(\varphi,\vartheta) \\ 0, & \text{else} \end{cases}. \tag{3.22}$$

Computing (3.22) for all φ, ϑ, and $\omega_{i_b}[k]$ results in the JPS. Applying the multidimensional maxima detector (used in case of the VSS-based algorithm) to the JPS, I set up a SJPS.

3.7 Sparse Joint Parameter Space

The JPS is a representation of angles φ and ϑ as well as frequencies ω_{i_b} and their respective amplitudes $|\widehat{C}|_{i_1,i_2}^{(i_b)}(\varphi,\vartheta,\omega_{i_b}[k])$ over time. Thus, a point in the JPS at an arbitrary frame index is labeled as a 4-tuple $(\varphi,\vartheta,\omega_{i_b}[k],|\widehat{C}|_{i_1,i_2}^{(i_b)})$. I set up a JPS for each band and each pair of sensors. As shown in Fig. 3.3, the bands' JPSs have to be merged (and not summed), the microphones' JPSs have to be summed. In order to eliminate irrelevant information in the JPS shown in Fig. 3.4, I employ an efficient multidimensional maxima detector as described in [49] to obtain a sparse representation of the JPS: the sparse joint parameter space (SJPS) as shown in Fig. 3.5. Fig. 3.6 shows the algorithm's pseudo code, which should support the programmer when implementing the algorithm.

3.8 Experimental Design

Before giving details on the experimental design, I introduce algorithmic and environmental parameters valid for all upcoming experiments. I set the frame size to 0.032 s, the overlap of frames to 0.010 s, and the size of the maxima detector's search window to (3×3) indices. For the evaluations, I considered a tolerance window of 5 Hz and 5° around the ground truth to define the root-mean-square errors and joint recalls, especially in case of double-source experiments. I also considered an amplitude threshold of 10^{-5} in the JPS to limit the number of detected maxima; maxima below that value were omitted. Table 3.1 lists the remaining algorithmic and environmental parameters.

The upcoming subsections inform about the experiments with synthesized signals as well as the experiments with synthetically spatialized speech signals.

3.8.1 Experiments with Synthesized Signals

To determine the performance of the proposed algorithm, its predecessors, and other algorithms, I carried out experiments with non-moving harmonic sources and noise sources

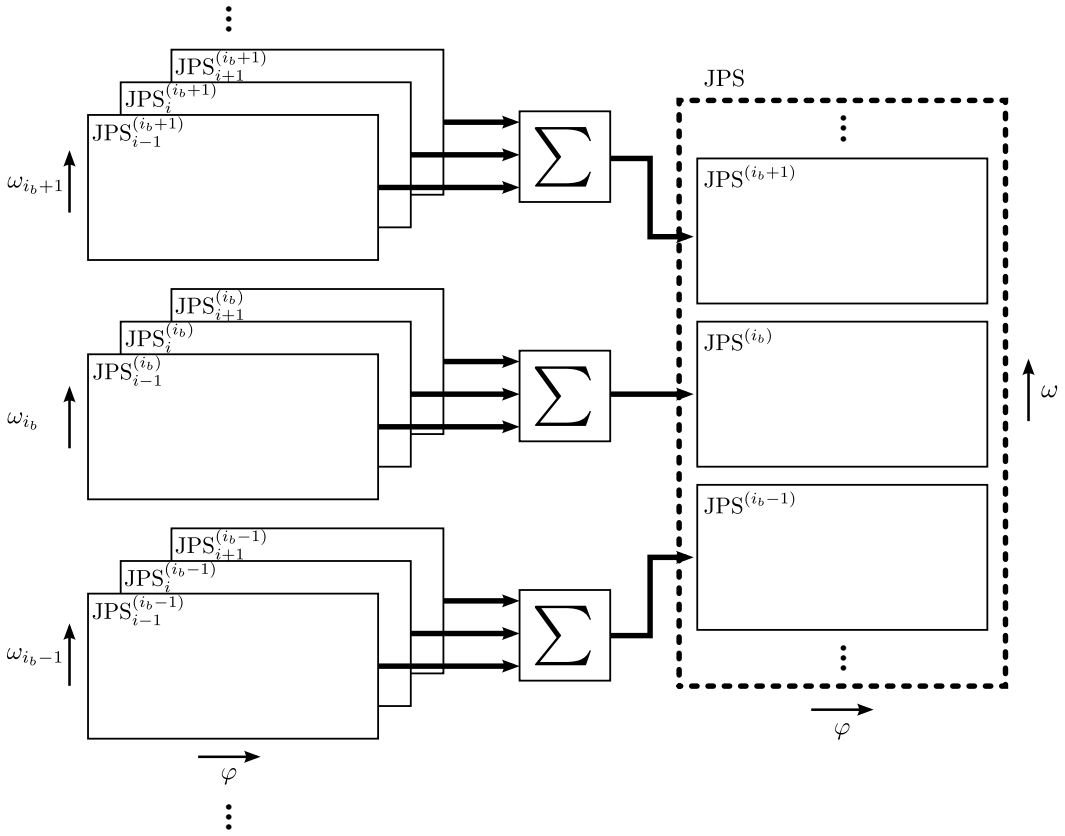
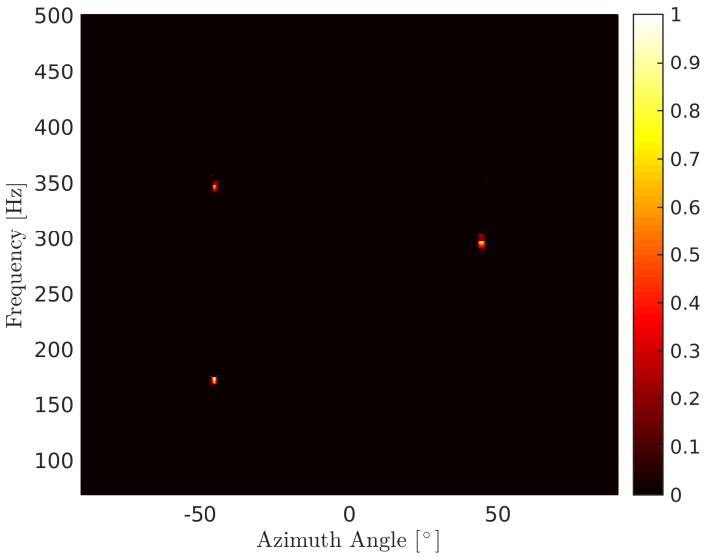


Fig. 3.3: Illustration of how to sum and merge JPSs. Summing the JPSs of all pairs of microphones per band and merging each band's JPS along the angular frequencies ω yields the overall JPS labeled with (2) in Fig. 3.1. In the figure above, i_b represents the band index, i is the microphone pair's index, φ is the azimuth angle, and ω_{i_b} denotes frequencies corresponding to a band with index i_b.

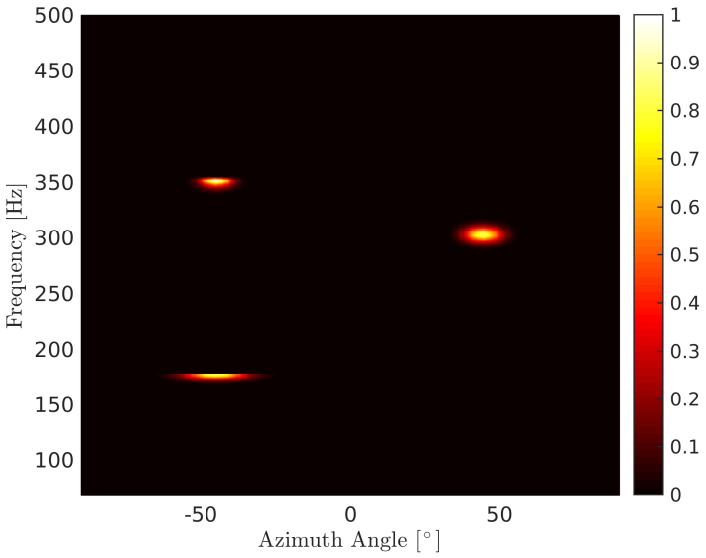
Table 3.1: Parameters of the synthetically spatialized, linearly frequency-sweeping signals. The variables denote the angular step size $\Delta\varphi$, the elevation angle ϑ, the number of microphones N_m, the array length d_a, the number of harmonics N_q, the sweep's start frequency and stop frequency f_1 and f_2, the sweep's duration T_2, the distance between the source and the array's center $|s|$, the signal to noise ratio SNR, the signal to interference ratio SIR, the temporal signal components' amplitude α, the normal distribution of noise with its parameters $\mathcal{N}(0,1)$, and the angular grid Φ.

$\Delta\varphi$	ϑ	N_m	d_a	N_q	f_1	f_2	T_2	$\lvert s\rvert$
1°	90°	8	0.50 m	4	80 Hz	500 Hz	2 s	3 m

SNR/SIR	α	ν	Φ
$\{-10, 0, 10, 20, 30\}$ dB	$0.4\sqrt{10^{\frac{\text{SNR}}{10}}}$	$\mathcal{N}(0,1)$	$\{-75°, \ldots, 75°\}$

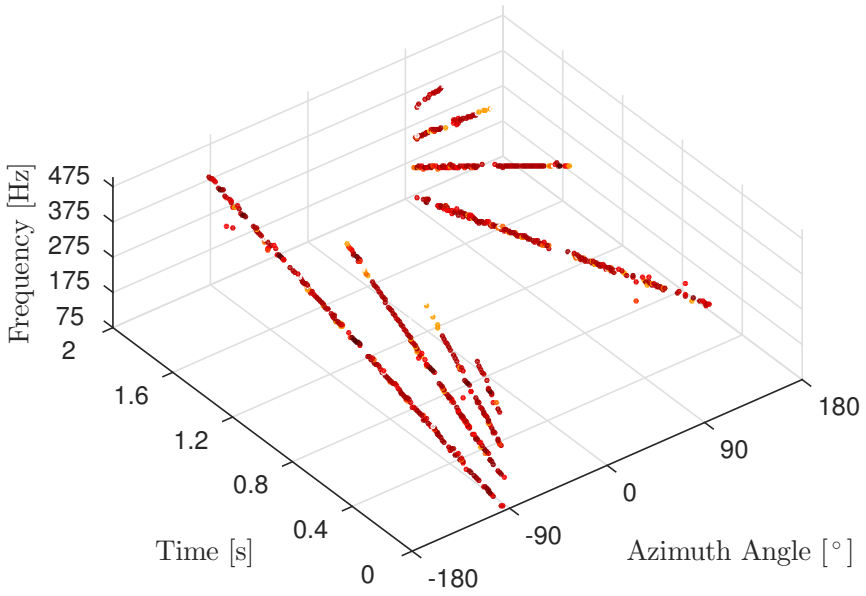


(a)

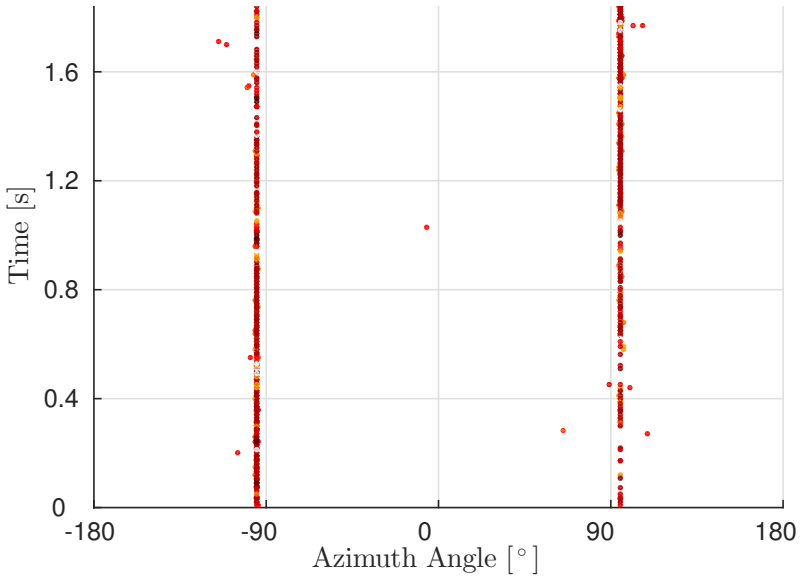


(b)

Fig. 3.4: A normalized three-dimensional JPS (similar to the POPI plane [45]) representing the jointly estimated DOAs, f_0s and their corresponding second, third, and fourth harmonics with the respective amplitudes (half-wave rectified and normalized to achieve values between zero and one) computed (a) with the proposed RPDM-based algorithm and (b) with its VSS-based predecessor [49] based on variable-scale sampling in lag domain. By comparing the planes in (a) and (b) one can see that both algorithms correctly estimate the snapshot of the two frequency-sweeping harmonic sources at $-45°$ and $45°$ with instantaneous f_0s around 180 Hz and 310 Hz. In (b), the widening of the Gaussian-like kernels to lower frequency bands is due to the increase in a band's CCF's sampling periods to lower frequency bands. However, one can avoid this widening by employing the RPDM-based algorithm, as shown in (a). In comparison to [35], there is no pitch-period doubling. To generate these plots, I set the spatial (angular) step size to $1°$, the bandwidths to 36 Hz, the sampling frequency to 32 kHz, the eight-microphone circular array's diameter to 50 cm, the frame length to 0.032 s, the frame shift to 0.010 s, the number of sampling points to 5 in case of (b), and the DOA-tolerance to $\varepsilon = 0.5°$ in case of (a).



(a)



(b)

Fig. 3.5: Resulting estimates (similar to trajectories generated by a tracker) after jointly estimating DOAs, f_0s and second, third, and fourth harmonics with the respective amplitudes. I synthetically spatialized two frequency-sweeping harmonic sources at $\phi_{s_1} = -95°$ and $\phi_{s_2} = 95°$ and simulated the sampled acoustic wave field observed by an eight-element uniform circular array with a diameter of 0.5 m.

Algorithm 4: Source Localizer and Characterizer

Data: Discrete-time multi-channel signals.
Result: Sparse joint parameter spaces.

```
initialization; // (3.1), (3.4), (3.5), (3.18), (3.19)
// design optimized bandpass filters (see [49] )
// compute intervals of time delays of arrival
// compute contours of each band's chirp z-transform
// consider extensions for maxima detector (see [49])
split into frequency bands by applying bandpass filters;
split into frames;
while getting frames do
    foreach pair of microphones do
        foreach frequency band do
            apply chirp z-transform to both mic-channels;
            // use (3.10) or consider implementation as described in [71,90] to increase
               efficiency
            compute cross-spectrum; // (3.12)
            compute magnitudes;
            compute relative phase delays; // (3.15)
            compute relative phase-delay mask; // (3.20)
            mask cross-spectrum magnitudes; // (3.22)
            // joint parameter space per pair and band
        end
        concatenate each frequency band's joint parameter space;
        // joint parameter space per pair
    end
    sum all pairs' joint parameter spaces;
    scale joint parameter space by number of pairs of microphones;
    // joint parameter space
    detect maxima; //  [80]
    eliminate maxima in extension;
    // sparse joint parameter space
end
```

Fig. 3.6: Pseudo-code of the proposed algorithm based on relative phase delay masking. Two slashes indicate a comment.

in free field. Table 3.1 lists all relevant parameters for generating the corresponding signals described in Fig. 1.2. As I showed in the previous chapter and in [49], and as I will show later, there is a negligible difference in R (joint recall) and RMSE between experiments with moving sources and experiments with non-moving sources. When I used the new approach, the difference was negligible, too, although the sources' velocity was relatively high compared to moving speakers in a real environment. Thus, I skip the discussion of experiments with moving sources. At the beginning of an experiment, I assigned a random DOA to each source without considering a minimum angular difference between two sources (in case of double-source experiments). I conducted Monte Carlo experiments because of randomly selected initial DOAs, SNRs, or SIRs, in three different categories and with different algorithms for comparisons.

In the first category, a non-moving source emitted an f_0-sweeping harmonic signal at varying locations. Fig. 3.7 (a) shows a short frame of such a signal.

In the second category, a non-moving harmonic source emitted an f_0-sweeping harmonic signal at varying locations together with a non-moving noise source featuring a different location (see Fig. 3.7 (b)).

In the third category, two non-moving harmonic sources emitted an f_0-sweeping harmonic signal at different locations (see Fig. 3.5 and Fig. 3.7 (c)). I estimated the f_0 of both sources.

Comparisons With Other Algorithms

To compare the performance of five different algorithms, I conducted experiments with the RPDM-based algorithm, its predecessor denoted as VSS (variable-scale sampling [49]), as well as POPI (position-pitch [35, 45]), the NLS (nonlinear least squares [23]) and the aNLS (approximate nonlinear least squares [23]). I used the same parameters as described in the previous chapter.

3.8.2 Experiments with Synthetically Spatialized and Reverberated Real Speech Signals

In these experiments, I used a subset of the Austrian-German speech corpus [50, 51]. Fig. 3.7 shows representative signals of this corpus. However, I synthetically spatialized the close-talking recordings and added reverberation with different reverberation times by utilizing a toolbox named image-source method for room impulse response simulation [92]. By considering the same room geometry mentioned in [51] I simulated a meeting-room with a reverberation time of $T_{60} = \{0, 0.1, 0.5\}$ s and for eight microphones representing a linear array with a maximum dimension of 0.5 m.

3.9 Experimental Results

This section summarizes the results of the aforementioned experiments. I start with the outcomes of the experiments based on fully synthesized signals and close the section with the outcomes of the experiments based on synthetically spatialized and reverberated real speech signals.

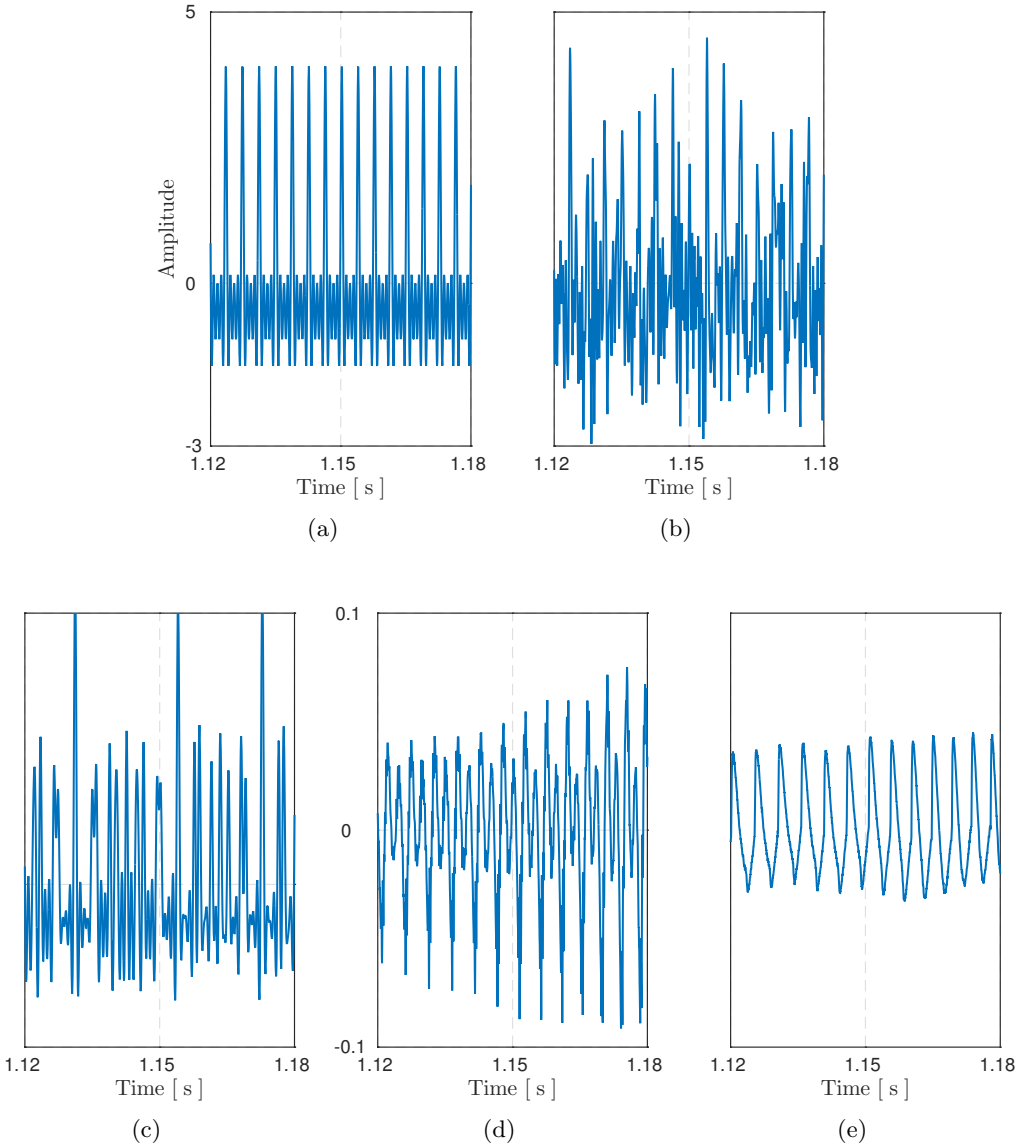


Fig. 3.7: The plots shown above represent different types of synthetically generated signals and real recorded signals; (a), (b), and (c) show a signal's snapshot of a non-moving linearly frequency-sweeping harmonic source with four harmonics (a) without any interferences, (b) with an interfering Gaussian noise source, and (c) with an interfering, linearly frequency-sweeping harmonic source. To generate these plots, I set the sampling frequency to $f_s = 32$ kHz and the initial fundamental frequency to $f_0 = 80$ Hz or $f_0 = 500$ Hz. In case of (b) and (c), a linearly frequency-sweeping source plus a white Gaussian noise source and two linearly frequency-sweeping sources, I set the initial DOAs to $(\varphi_s^{(1)}, \varphi_s^{(2)}) = (-45°, 45°)$ and SNR = SIR = 0 dB. In case of (c), I inverted the sweep of the second source yielding initial fundamental frequencies according to $(f_0^{(1)}, f_0^{(2)}) = (80\text{ Hz}, 500\text{ Hz})$. Plot (d) shows the headset microphone's, plot (e) the laryngograph's time signals of the first phoneme /eː/ of the (German-language) sentence [jeː neːa dɛɐ tsaɪgɐ aʊf axt kaːm dɛstɔ ʊnruːɪgɐ vʊɐdən diː lɔʏtə] (IPA) read by a female speaker.

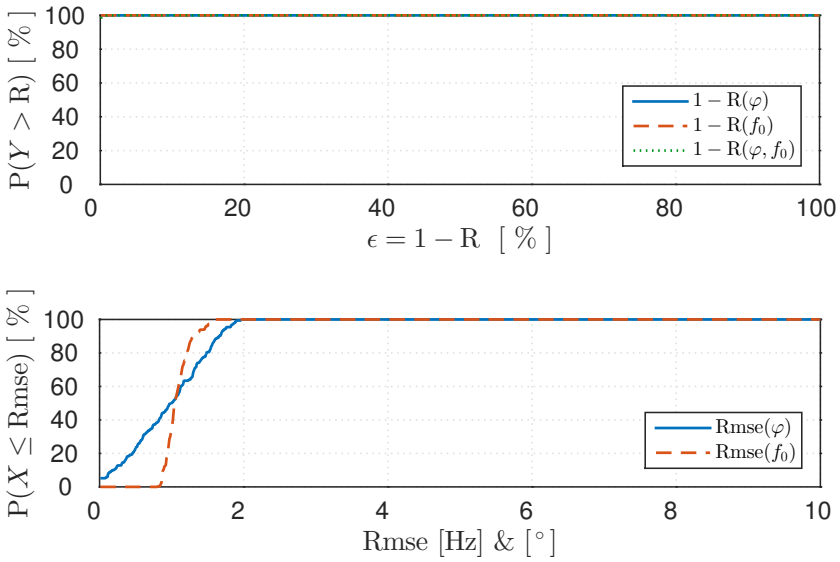


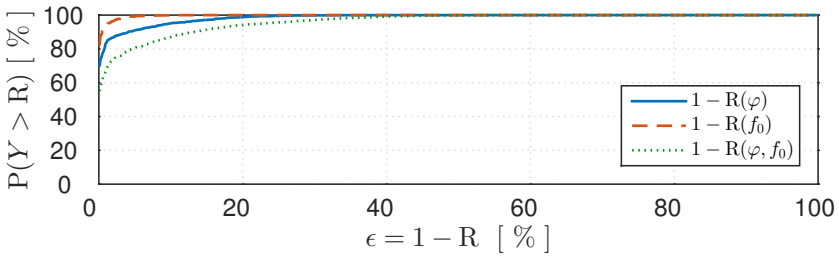
Fig. 3.8: Resulting cumulative distribution functions of an experiment with synthesized harmonic sources. The curves describe the probabilities that the opposite of R in percent (top), i.e., ($\epsilon = 1 - \mathrm{R}$), of jointly estimated DOAs and f_0s, and the RMSE (bottom) of estimated DOAs and f_0s have a value equal to or less than $1 - \mathrm{R}$ or RMSE, respectively.

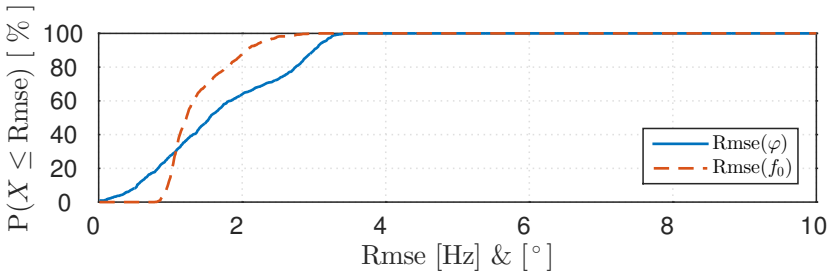
3.9.1 Experiments with Synthesized Signals

By varying algorithmic parameters and signals based on randomly generated white noise, I conducted between $1,000$ and $10,000$ Monte Carlo experiments in each category. The final number of experiments depended on the number of all possible combinations of parameters. Table 3.2 lists all three categories' results of the Monte Carlo experiments. Fig. 3.8, Fig. 3.9, and Fig. 3.10 show CDFs of category one (linearly frequency-sweeping harmonic source), category two (linearly frequency-sweeping harmonic source and noise source), as well as category three (two linearly frequency-sweeping harmonic sources), respectively. The legends' items $1 - \mathrm{R}(\varphi)$, $1 - \mathrm{R}(f_0)$, and $1 - \mathrm{R}(\varphi, f_0)$ denote one minus the recall of DOAs, f_0s, and DOAs and f_0s, respectively. The remaining legends' items, RMSE(φ) and RMSE(f_0), denote the root-mean-square error's CDFs of f_0s and DOAs, respectively.

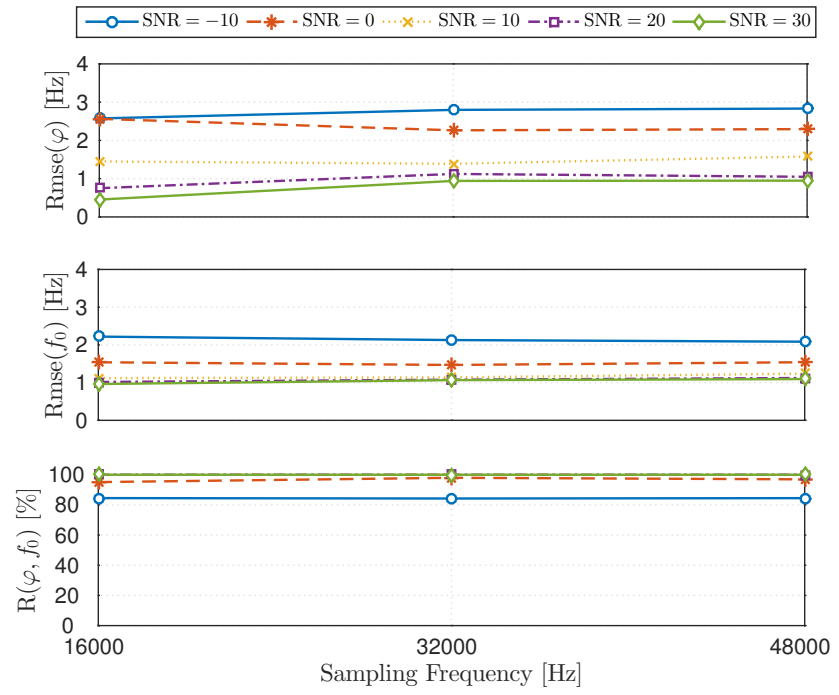
3.9.2 Experiments with Synthetically Spatialized and Reverberated Real Speech Signals

Due to a fixed set of algorithmic parameters, varying reverberant conditions ($T_{60} = \{0, 100, 500\}$ ms), and a limited number of a male and female speaker's recordings, I show results of $P(X \leq \mathrm{RMSE})$ and $P(Y > \mathrm{R})$ in Fig. 3.11 only.





(a)



(b)

Fig. 3.9: (a) Cumulative distribution functions and (b) root-mean-square errors and joint recalls for different SNRs of experiments with a synthesized harmonic source and a noise source.

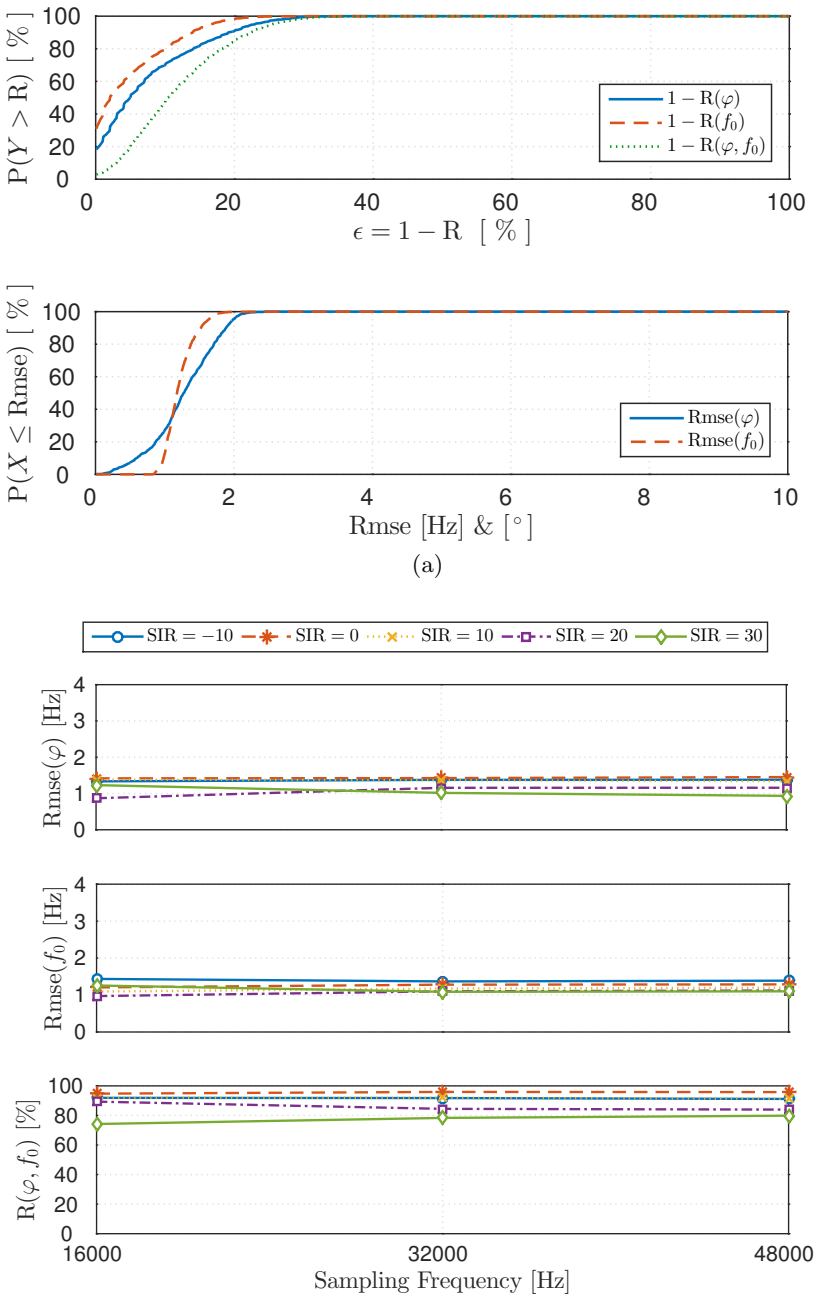


Fig. 3.10: (a) Cumulative distribution functions and (b) root-mean-square errors and joint recalls for different SIRs of experiments with two synthesized harmonic sources.

Table 3.2: Results of synthetic-data experiments with different approaches: RPDM (the new approach), VSS (variable scale sampling [49]), POPI (position-pitch [45]), NLS (nonlinear least squares [23]), and aNLS (approximate nonlinear least squares [23]). The table consists of three sections covering the results of experiments with a single non-moving harmonic source, a single non-moving harmonic source plus noise source, and two non-moving harmonic sources, respectively. In the second section, the first value in each column represents the averaged results of all experiments with varying SNR, the second one for SNR = 30 dB, and the third one for SNR = −10 dB. In the third section, the second value in each column represents the results for SIR = 30 dB, the third for SIR = 0 dB. I set $d_a = 0.50$ m, $N_m = 8$, and $f_s = 32$ kHz. Note: The POPI-algorithm [35] doubles pitch periods and estimates DOAs only. As a consequence, determining a source's true fundamental frequencies using the POPI-algorithm is impossible.

Algorithm	$\overline{\mathrm{R}}(\varphi, f_0)$ [%]			$\overline{\mathrm{RMSE}}(\varphi)$ [°]			$\overline{\mathrm{RMSE}}(f_0)$ [Hz]		
RPDM	**100**			0.92			1.05		
VSS	93			1.01			1.41		
POPI	**100**			**0.01**			3.24		
NLS	49			3.54			**0.30**		
aNLS	39			3.81			1.15		
	AVG	30 dB	-10 dB	AVG	30 dB	-10 dB	AVG	30 dB	-10 dB
RPDM	98	**100**	86	1.68	0.94	2.79	**1.36**	1.06	**2.12**
VSS	90	94	75	1.77	0.84	3.06	1.57	1.30	2.32
POPI	**100**	**100**	**98**	**0.48**	**0.02**	**1.69**	3.31	3.24	3.56
NLS	37	48	3	3.95	3.68	3.80	1.60	**0.41**	5.21
aNLS	30	39	3	4.03	3.88	3.42	1.61	1.16	4.44
	AVG	30 dB	0 dB	AVG	30 dB	0 dB	AVG	30 dB	0 dB
RPDM	**90**	80	**97**	**1.26**	1.01	**1.41**	**1.19**	1.08	**1.27**
VSS	87	**81**	90	1.34	1.04	1.43	1.41	1.31	1.49
POPI	62	52	91	1.47	**0.03**	2.06	3.71	3.24	3.70
NLS	24	28	12	4.00	3.71	4.33	1.69	**0.52**	2.65
aNLS	17	20	8	3.98	2.63	4.71	1.99	1.15	2.88

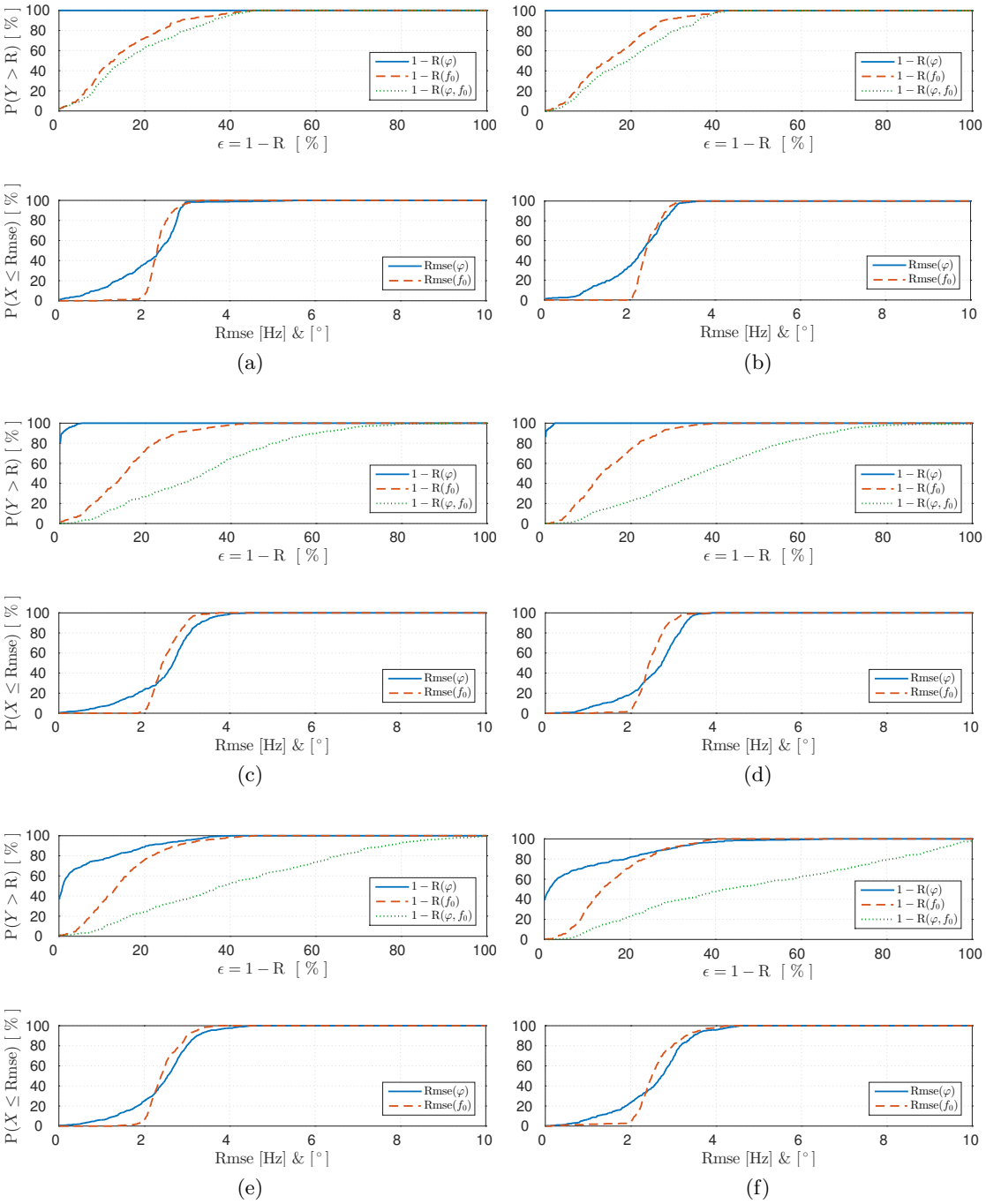


Fig. 3.11: Cumulative distribution functions of joint recalls and root-mean-square errors of experiments with synthetically spatialized and reverberated real speech signals of a female (a,c,e) and a male (b,d,f) speaker. The reverberation times are as follows: (a,b) $T_{60} = 0$ ms, (c,d) $T_{60} = 100$ ms, and (e,f) $T_{60} = 500$ ms.

3.10 Discussion

In this section, I discuss the experimental results as well as the RPDM-based algorithm's improvement in frequency resolution and the computational complexity. Additionally, I compare the results with the outcomes of the VSS-based algorithm.

3.10.1 Experiments with Synthesized Signals

In the first category (single harmonic source) the RPDM-based algorithm achieved the highest recall rates, $\mathrm{R}(\varphi) = \mathrm{R}(f_0) = \mathrm{R}(\varphi, f_0) = \overline{\mathrm{R}}(\varphi, f_0) = 100\%$, and lowest average root-mean-square errors, $\overline{\mathrm{RMSE}}(\varphi) = 0.92°$ and $\overline{\mathrm{RMSE}}(f_0) = 1.05$ Hz, as shown in Fig. 3.8 and Table 3.2. Thus, the algorithm perfectly solves the problem of localizing and characterizing a frequency-sweeping harmonic source in free field in terms of the (average joint) recall.

Regarding the second category (single harmonic source plus noise source), the CDFs shown in Fig. 3.9 (a) represent the results of all SNR-experiments. One can see that $\forall\epsilon : \mathrm{R}(f_0) \geq \mathrm{R}(\varphi) \geq \mathrm{R}(\varphi, f_0)$ and $P(Y > \mathrm{R}) \neq 100\%$ around $\epsilon = 0$ which is due to experiments with SNR ≤ 0 dB. However, for SNR > 0 dB, the algorithm achieved $P(Y > \mathrm{R}) = 100\%$. These numbers highlight the algorithm's robustness in case of white Gaussian noise sources featuring a power smaller than the power of harmonic sources. Fig. 3.9 (b) (top) shows a small increase in $\mathrm{RMSE}(\varphi)$, which is attributable to experiments with SNR ≤ 0, as Fig. 3.9 (b) (bottom) confirms. Table 3.2 emphasizes the algorithm's robustness for experiments with SNR $= \{-10, \ldots, 30\}$ dB, because $\overline{\mathrm{R}}(\varphi, f_0) = 98\%$. The average RMSEs, $\overline{\mathrm{RMSE}}(f_0) \approx 1$ Hz and $\overline{\mathrm{RMSE}}(\varphi) \approx 1°$, are still low.

Fig. 3.10 (a) illustrates the results of all SIR-experiments in category three (two harmonic sources). Similar to Fig. 3.9 (a), Fig. 3.10 shows that $\forall\epsilon : \mathrm{R}(f_0) \geq \mathrm{R}(\varphi) \geq \mathrm{R}(\varphi, f_0)$ and $P(Y > \mathrm{R}) \neq 100\%$ for $\epsilon \leq 29\%$. Again, this is due to experiments with SIR ≤ 0 dB. There are two major reasons why I cannot achieve $\mathrm{R}(\cdot) = 100\%$ in any experiment: First, the beating effect during crossings of frequencies [67]. Superimposed signals might cancel out each other at frequency crossings; these crossings cause destructive interference. Though having ground-truth data at frequency-crossings, there is no estimate at all due to destructive interference resulting in missing estimates. Second, the cross-spectrum (and the CCF) emphasizes the dominating source due to its nonlinear characteristics [93, 94]. Thus, if one source dominates, the other one is literally suppressed. If both sources feature the same power, i.e., SIR $= 0$ dB, I achieve the highest $\overline{\mathrm{R}}(\varphi, f_0)$; they are equally present. If one source dominates the other, which is true in case of SIR $= \{-10, 10, 20, 30\}$ dB, the recall is lower. If SIR $= \pm 10$ dB, the results are identical because one source, no matter which one, dominated the other source.

Comparisons with other Algorithms

In the following lines, I will discuss the results listed in Table 3.2 of the RPDM-based algorithm in relation to the VSS-based algorithm, the modified POPI algorithm, and the NLS approach as well as the aNLS approach.

On average, the RPDM-based algorithm outperforms its predecessor, the VSS-based algorithm, in each category, especially in terms of $\overline{\mathrm{RMSE}}(f_0)$ and $\overline{\mathrm{R}}(\varphi, f_0)$, which is

3-8% higher. These results indicate that the VSS-based algorithm's nonlinear frequency resolution to higher frequencies affects its performance. There is also a decrease in $\overline{\mathrm{RMSE}}(\varphi)$ in case of the RPDM-based algorithm. The increasing number of grid-points to higher frequencies leads to a higher number of estimates that are closer to the ground truth. As a consequence, this yields an increased $\overline{\mathrm{R}}(\varphi, f_0)$.

On average, the modified POPI algorithm is at least as good or slightly better than the RPDM-based algorithm in terms of $\overline{\mathrm{R}}(\varphi, f_0)$ and $\mathrm{RMSE}(\varphi)$ in experiments with a single source and a noise. However, once a second harmonic source interferes, the performance of POPI dramatically decreases and the RPDM-based algorithm outperforms POPI for each measure. This is because POPI is based on the sum of CCFs yielding a single CCF that carries information of two sources, where the dominating source is emphasized and the other one literally suppressed. Moreover, POPI exhibits pitch-period doubling and the worst $\overline{\mathrm{RMSE}}(f_0)$s. However, it features the best $\overline{\mathrm{RMSE}}(\varphi)$ due to the use of a summed CCF, and, as shown in the table, it can successfully cope with white Gaussian noise.

In general, the RPDM-based algorithm yields very promising results especially in noisy experiments and experiments with two harmonic sources. The invariant frequency resolution leads to an $\overline{\mathrm{RMSE}}(f_0)$, which is always best except in the single-source experiments, where the NLS performs best.

3.10.2 Experiments with Synthetically Spatialized and Reverberated Real Speech Signals

Fig. 3.11 (a) and (b) illustrate the CDFs of experiments with a female speaker and a male speaker in free field. In both cases, the RPDM-based algorithm perfectly estimated the speakers' DOAs, i.e. $\forall \epsilon : P(Y > \mathrm{R}(\varphi)) = 100\%$. Consequently, the joint recall mostly depends on the capability of estimating the frequency components.

Fig. 3.11 (c) and (d) show the CDFs of experiments featuring a reverberant environment with a reverberation time of $T_{60} = 100$ ms. In comparison to the previous figures, there is almost no change in $P(Y > \mathrm{R}(\varphi))$ and $P(Y > \mathrm{R}(f_0)$. However, there is a decrease in the number of experiments exhibiting a high joint recall, $\mathrm{R}(\varphi, f_0)$, in case of the male speaker and the female speakers.

Fig. 3.11 (e) and (f) illustrate the CDFs of experiments with a reverberation time of $T_{60} = 500$ ms. Compared with Fig. 3.11 (c) and (d), there is a decrease in $P(Y > \mathrm{R}(\varphi))$ but no noticeable decrease in $P(Y > \mathrm{R}(f_0))$. Reverberation causes a decrease in localization accuracy, as stated in many papers about source localization. However, it does not affect the signals' frequency components except in the presence of resonance frequencies depending on the room geometry. There is a small decrease in $P(Y > \mathrm{R}(\varphi, f_0))$, which shows, again, that the joint recall rather depends on the frequency components than the DOAs.

Compared with the results of real-data experiments carried out with the VSS-based algorithm, the new approach shows a reduction in RMSE (up to 4 Hz and 4°). However, the experiments with real signals in the previous chapter and in [49] were conducted with two and three microphones only. According to Fig. 3.11 (a-f) (bottom), the RMSEs slightly change for increasing reverberation time T_{60}, and their CDFs are almost identical for experiments with male and female speakers, especially in case of $\mathrm{RMSE}(f_0)$.

3.10.3 The Use of Linear Sweeps

In the previous chapter and in [49] I evaluated the algorithms' performance by using exponential sweeps in synthetic-data experiments. After conducting experiments with the RPDM-based approach, I noticed that this kind of sweeps were in favor of the predecessor's major drawback—the nonlinear frequency resolution. However, this time I applied linear sweeps to the VSS-based algorithm, too. According to the results, I realized that it featured a decreased performance due to the discussed drawback. Table 3.2 shows that by employing the CZT and the RPDM I can solve this issue. Thus, the RPDM-based approach outperforms the VSS-based approach in terms of $\overline{\mathrm{RMSE}}(\varphi)$, $\overline{\mathrm{RMSE}}(f_0)$, and $\overline{\mathrm{R}}(\varphi, f_0)$ in all experimental categories. These results highlight one novelty of the new approach: the invariant frequency resolution.

3.10.4 Improvements in Frequency Resolution

In comparison to the RPDM-based algorithm, the frequency resolution of the VSS-based algorithm is nonlinearly decreasing to higher frequencies. The predecessor uniformly samples the cross-correlation function around lag zero with different sampling intervals. It increases the intervals lag-wise, which yields a linear increase in sampling period but a non-linear decrease in frequency, $f_0 = T_0^{-1}$. As a consequence, the predecessor features a non-linearly decreasing frequency resolution for lag-wise decreasing periods,

$$f_0[l \cdot T_s] = (l \cdot T_s)^{-1}, \tag{3.23}$$

$$f_0[l \cdot T_s + T_s] = (l \cdot T_s + T_s)^{-1} \tag{3.24}$$

with l as the lag-index and T_s as the sampling period. This decrease in frequency resolution causes an increasing RMSE to higher frequencies. I can decrease the RMSE by increasing f_s; however, I cannot eliminate the nonlinear resolution. The proposed algorithm samples a unit circle's arc uniformly, which yields uniformly spaced frequencies according to

$$f_0[k] = k \cdot (f_{\max} - f_{\min})/(M-1) + f_{\min} \tag{3.25}$$

with $f_{\min}$ and $f_{\max}$ as the bounding frequencies, M as the number of the chirp's points, and k as the frequency index.

3.10.5 Computational Complexity

It is difficult to clearly determine the computational complexity of an algorithm in terms of the big O-notation when using unfree software, e.g., MATLAB. Most built-in functions are not accessible. Thus, I failed to investigate the function's implementation and, as a consequence, its exact complexity. (For instance: A trigonometric function, e.g., `sin`, either returns values stored in a lookup table or returns values based on a computation with order $\mathcal{O}(M(n)\log_2(n))$, where $M(n)$ depends on the computer number format.) However, in Table 3.3 I list variables denoting the number of a module's application for a single frame. At first glance, the predecessor denoted as VSS requires fewer computations, because there are fewer variables in the last three rows. Taking a closer look at the variables, one can see that $N_\Phi \approx N_\varphi \cdot N_\vartheta$, $N_T \approx N_b \cdot N_k$, and

Table 3.3: Computational complexity of the VSS-based approach and the RPDM-based approach. This table lists variables denoting the number of a module's application per frame. For instance, $N_{\mathrm{BPF}} = N_m \cdot N_b$ means that a bandpass filter has to be applied $N_m \cdot N_b$ times, where N_m and N_b is the number of microphones and the number of bands, respectively. The remaining variables are as follows: N_{BPF} is the number of bandpass filters, N_{JPS} is the number of joint parameter spaces, N_{SJPS} is the number of sparse joint parameter spaces, N_{MAX} is the number of maxima detections, N_{DFT} is the number of discrete Fourier transforms, N_{CSP} is the number of computing the cross-spectrum, N_{IDFT} is the number of inverse discrete Fourier transforms, N_{VSS} is the number of variable-scale sampling procedures, N_{SUM} is the number of summations, N_{CZT} is the number of chirp z-transforms, N_{RPD} is the number of relative phase delays, N_{RPDM} is the number of applying relative phase-delay masking, and N_{WGT} is the number of weightings. Moreover, N_p is the number of microphone pairs, N_T is the number of sampling periods, N_Φ is the number of sampling phases, N_k is the number of CZT-indices per band, N_φ is the number of azimuth angles, and N_ϑ is the number of elevation angles.

# Applications	VSS	RPDM	# Applications
N_{BPF}	$N_m \cdot N_b$	$N_m \cdot N_b$	N_{BPF}
N_{JPS}	1	1	N_{JPS}
N_{SJPS}	1	1	N_{SJPS}
N_{MAX}	1	1	N_{MAX}
N_{DFT}	$N_m \cdot N_b$	$N_m \cdot N_b$	N_{CZT}
N_{CSP}	$N_p \cdot N_b$	$N_p \cdot N_b$	N_{CSP}
N_{IDFT}	$N_p \cdot N_b$	$N_p \cdot N_b \cdot N_k$	N_{RPD}
N_{VSS}	$N_p \cdot N_T \cdot N_\Phi$	$N_p \cdot N_b \cdot N_k \cdot N_\varphi \cdot N_\vartheta$	N_{RPDM}
N_{SUM}	$N_p \cdot N_T \cdot N_\Phi$	$N_p \cdot N_b \cdot N_k \cdot N_\varphi \cdot N_\vartheta$	N_{WGT}

$N_T \cdot N_\Phi \approx N_b \cdot N_k \cdot N_\varphi \cdot N_\vartheta$. This implies that the number of each component's application per frame is approximately the same in case of the RPDM-based approach and its predecessor. Both algorithms are real-time capable for a manageable number of bands and microphones, even when a single core for computations is used only.

Chapter 4

Bayesian Multiple-Target Trackers[1]

Albeit there is a large number of publications about multiple-target tracking, I usually stick to literature published by those who originally invented/introduced new algorithms. For instance, Ronald P. S. Mahler thoroughly described the random finite sets (RFS), the finite set statistics (FISST), and Bayesian multiple-target tracking in [95,96]. Apart from Mahler, the Vo Brothers, Ba-Ngu Vo and Ba-Tuong Vo, carefully described multiple-target trackers, e.g., the Gaussian mixture probability hypothesis density (GM-PHD) filter [97], the Gaussian mixture cardinalized probability hypothesis density filter (GM-CPHD) filter [98], and the Gaussian mixture cardinality-balanced multi-Bernoulli multi-target (GM-CBMeMBer) filter [99]. Besides, Daniel E. Clark published literature about the basics of random-set filtering [100].

The upcoming part summarizes all the aforementioned references about multiple-target tracking. But before going into detail, I summarize and explain its origin: single-target filtering and Bayesian multiple-target tracking.

The goal in the former field is to estimate a system's state that changes over time by utilizing a sequence of noisy observations. This state, for instance, contains kinematic characteristics, i.e., its position in space and its velocity, etc. In practice, I have access to observations representing noise-corrupted sensor measurements. To estimate a target's state after receiving a new observation and without reprocessing the preceding observations, I apply a recursive filter.

In a single-target environment, I usually represent a state and an observation as a vector. However, in a multiple-target environment, a state and an observation are finite sets (or collections) of vectors. An observation is a set of various elements, e.g., observed states distorted by noise as well as spurious target-independent observations, known as clutter. Targets appear and disappear in a scene (e.g., the surveillance region); the number of (observed) states vary with time.

The whole concept of multiple-target tracking is based on finite sets. They represent

[1]This chapter is based on research initiated during my stay at the DSP-Lab at University of California, San Diego (UCSD). Collaboration with Bhaskar D. Rao is gratefully acknowledged. My personal contributions are the application of FISST-based multiple-target trackers (GM-PHD, GM-CPHD, GM-CBMeMBer) and the application of the optimal subpattern assignment (OSPA) distance in the field of joint parameter estimation and parameter tracking of harmonic sources.

multiple-target states in all variations, e.g., an empty set when no target is active or a set consisting of three targets. In contrast, it is impossible to describe a target-less scene by vector analysis alone.

Bringing it all together, the objective of multiple-target tracking is to jointly estimate the number of targets and their states at each instant of time from a sequence of noisy and/or cluttered observations.

To associate observations with their corresponding targets in multiple-target tracking, I require a huge amount of computational resources. However, utilizing RFS [101] bypasses the association problem. In theory, a certain number of targets is a set-valued state; a certain number of observations is a set-valued observation. Considering this specific formulation, I can dynamically estimate the states of multiple targets in a noisy and cluttered environment using a Bayesian filtering framework [101]. Moreover, this formulation results in a generalization of the single-target Bayesian filter. In the classical single-target (Bayesian) framework, a state as well as an observation is a (vector-valued) random variable's realization. In the multiple-target framework, a (multiple-target) state and (multiple-target) observation is a finite set (of vectors). As a consequence, I need the concept of RFS to apply the Bayesian framework to this specific (multiple-target) tracking problem.

For the first time in 1996, Mahler used the RFS theory in the field of multiple-sensor and multiple-target filtering [102]. In the years that followed, the RFS theory evolved into the FISST theory. Unfortunately, the resulting FISST-based Bayesian multiple-target recursion was intractable. In 2000, Mahler approximated the recursion by employing the probability hypothesis density (PHD) filter. Described in [97, 101], this filter propagates the first-order statistical moment (or intensity) of the states' RFS. In contrast to the previous approaches, the PHD recursion operates on a single-target state space (though being a multiple-target tracking algorithm). As a consequence, it circumvents the combinatorial increase of computational resources that would have been caused by data association. However, at this point the PHD recursion contains integrals which lack closed-form solutions.

In 2005, Vo et al. introduced a closed-form solution to the PHD recursion for linear Gaussian and mildly nonlinear multiple-target models. In [97], they present an analytic solution to the PHD recursion for linear Gaussian target dynamics and a Gaussian birth model based on Kalman filtering, i.e., by propagating Gaussian components: the means, the weights, and the covariances of states. By using the GM-PHD filter, I can extract the state estimates from the posterior intensity more efficiently compared with clustering in a particle-based approach [97] (an approach based on sequential Monte Carlo techniques). (Integrating a density function over its entire space yields one; in comparison to that, integrating an intensity function over its entire space can yield values smaller or larger than one.) A GM-PHD filter's drawback is the increasing number of Gaussian components over time. To mitigate this problem, I can employ pruning and merging, which I will describe later.

As mentioned before, multiple-target filtering has its origin in single-target filtering. To understand the link between both, I first have to focus on single-target filtering as described below.

4.1 Single-Target Filtering

In a hidden Markov model, a state $\mathbf{x}_k \in \mathcal{X}$ with $\mathcal{X}$ as a set of vectors at time index k is not directly observable. However, observing it yields an observation $\mathbf{z}_k \in \mathcal{Z}$, with $\mathcal{Z}$ as a set of vectors, which represents a state distorted by noise and/or interferences. Given a state $\mathbf{x}_{k-1}$ at time index $k-1$, the density function of translating a state from time index $k-1$ to time index k, i.e., from $\mathbf{x}_{k-1}$ to $\mathbf{x}_k$, is $f_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$. Given a state $\mathbf{x}_k$, the density function of receiving the observation $\mathbf{z}_k$ is $g_k(\mathbf{z}_k|\mathbf{x}_k)$. Given all observations $\mathbf{z}_{1:k} = (\mathbf{z}_1, \ldots, \mathbf{z}_k)$ with time indices $\{1, \ldots, k\}$, a state's probability density at time index k is $p_k(\mathbf{x}_k|\mathbf{z}_{1:k})$, where $p_k(\cdot)$ is the posterior density at time index k. Knowing the initial density $p_0(\cdot)$, the posterior density is the likelihood times the prior density divided by the likelihood marginalized over the states according to

$$p_k(\mathbf{x}_k|\mathbf{z}_{1:k}) = \frac{g_k(\mathbf{z}_k|\mathbf{x}_k)p_{k|k-1}(\mathbf{x}_k|\mathbf{z}_{1:k-1})}{\int g_k(\mathbf{z}_k|\mathbf{x})p_{k|k-1}(\mathbf{x}|\mathbf{z}_{1:k-1})d\mathbf{x}}, \tag{4.1}$$

with the prior density

$$p_{k|k-1}(\mathbf{x}_k|\mathbf{z}_{1:k-1}) = \int f_{k|k-1}(\mathbf{x}_k|\mathbf{x})p_{k-1}(\mathbf{x}|\mathbf{z}_{1:k-1})d\mathbf{x} \tag{4.2}$$

where $p_k(\cdot|\mathbf{z}_{1:k})$ contains the entire information about $\mathbf{x}_k$. By using the minimum mean squared error (MMSE) criterion or the maximum a posteriori (MAP) criterion [97], I can estimate $\mathbf{x}_k$.

When I apply single-target filtering (as described before) in a multiple-target scenario, I fail in determining which target generated which observation. There are methods based on single-target filtering that handle multiple targets and associate observations to their corresponding states, but these methods are usually computationally expensive [95]. However, a simple and clear solution to multiple target tracking and association is employing multiple-target filtering based on RFS and FISST.

4.2 Multiple-Target Filtering

The objective of multiple-target filtering is to jointly estimate the number of targets and their states at each instant of time from a sequence of noisy and/or cluttered observations.

Given the number of targets at time index $k-1$, $N_{\mathbf{x},k-1}$, the target states at time index $k-1$ are $\{\mathbf{x}_{k-1,1}, \ldots, \mathbf{x}_{k-1,N_{\mathbf{x},k-1}}\}$, where $\forall i : \mathbf{x}_{k-1,i} \in \mathcal{X}$ The states' order in a set is irrelevant for RFSs. Given the number of observations at time index k, $N_{\mathbf{x},k}$, the observations at time index k are $\{\mathbf{z}_{k,1}, \ldots, \mathbf{z}_{k,N_{\mathbf{z},k}}\}$, where $\forall i : \mathbf{z}_{k,i} \in \mathcal{Z}$. These observations may originate from targets and clutter.

In multiple-target filtering, states and observations are finite sets of subsets, $X_k \in \mathcal{F}(\mathcal{X})$ and $Z_k \in \mathcal{F}(\mathcal{Z})$, respectively, where

$$X_k = \left\{\mathbf{x}_{k,1}, \ldots, \mathbf{x}_{k,N_{\mathbf{x},k}}\right\} \tag{4.3}$$

and

$$Z_k = \left\{\mathbf{z}_{k,1}, \ldots, \mathbf{z}_{k,N_{\mathbf{z},k}}\right\} \tag{4.4}$$

with $\mathcal{F}(\mathcal{X}) \subseteq \mathcal{X}$ and $\mathcal{F}(\mathcal{Z}) \subseteq \mathcal{Z}$. Moreover, each set of states and observations, X_k and Z_k, is a multiple-target state and multiple-target observation, respectively, modeled as an RFS. While being a set, X_k is a finite-set-valued random variable featuring a (discrete) probability mass function and a joint density function; i.e. for a given cardinality of X_k, $|X_k|$, the set's probability density function represents the joint distribution of each element in X_k, whereas the probability mass function of X_k characterizes its cardinality.

In case of multiple-target filtering based on RFS and FISST, targets may die, survive (evolve), or appear at time index k yielding $N_{\mathbf{x},k}$ new states. Given a multiple-target state X_{k-1}, each $\mathbf{x}_{k-1} \in X_{k-1}$ survives and continues existing at time index k with probability $p_{S,k}(\mathbf{x}_{k-1})$ before it will be translated into $\mathbf{x}_k$ according to $f_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$. Considering the terminology of RFSs, the aforementioned statement is equal to the RFS

$$S_{k|k-1}(\mathbf{x}_{k-1}) = \begin{cases} \{\mathbf{x}_k\} & \text{if target survives} \\ \emptyset & \text{if target dies} \end{cases}. \tag{4.5}$$

It is a non-empty set, if the target survives; otherwise it is empty.

A new target begin to exist after birth (which is independent of any existing target) or spawning (which depends on existing targets). For instance, if a sensor (array) fails to resolve a group of closely spaced targets, this group will be represented as a single state. Once the targets diverge from each other and the sensor resolves each target, the group will suddenly be represented as multiple (spawned) targets. Assuming random finite sets for birth, Γ_k, death and surviving, $S_{k|k-1}(\mathbf{x}_{k-1})$, and spawning, $B_{k|k-1}(\mathbf{x}_{k-1})$, the multiple-target state $X_k = \{\mathbf{x}_{k,1}, \ldots, \mathbf{x}_{k,N_{\mathbf{x},k}}\}$ is

$$X_k = \left\{ \bigcup_{i \in X_{k-1}} S_{k|k-1}(i) \right\} \cup \left\{ \bigcup_{i \in X_{k-1}} B_{k|k-1}(i) \right\} \cup \Gamma_k. \tag{4.6}$$

Given a state $\mathbf{x}_k$, a sensor detects it with probability $p_{D,k}(\mathbf{x}_k)$ or misses it with probability $1 - p_{D,k}(\mathbf{x}_k)$ and obtains a state-related observation $\mathbf{z}_k$ according to $g_k(\mathbf{z}_k|\mathbf{x}_k)$. (The detection probability, $p_{D,k}(\mathbf{x}_k)$, affects the multiple-target tracker's posterior intensity; I will come back to it in (4.13)). Considering the terminology of RFSs, this is equal to

$$O_k(\mathbf{x}_k) = \begin{cases} \{\mathbf{z}_k\} & \text{if measured} \\ \emptyset & \text{if missed} \end{cases} \tag{4.7}$$

Considering clutter as an independent RFS, K_k, the RFS representing the received observations at the sensor is

$$Z_k = \left\{ \bigcup_{i \in X_k} O_k(i) \right\} \cup K_k \tag{4.8}$$

To describe the translation of a multiple-target state or a multiple-target observation, I employ the multiple-target transition density $f_{k|k-1}(X_k|X_{k-1})$ and the multiple-target likelihood $g_k(Z_k|X_k)$, which were explicitly derived in [101, 103, 104]. Similar to (4.1), the optimal multiple-target Bayes filter's multiple-target posterior density is

$$p_k(X_k|Z_{1:k}) = \frac{g_k(Z_k|X_k)p_{k|k-1}(X_k|Z_{1:k-1})}{\int g_k(Z_k|X)p_{k|k-1}(X|Z_{1:k-1})\mu_s(dX)}, \tag{4.9}$$

with its prior density

$$p_{k|k-1}(X_k|Z_{1:k-1}) = \int f_{k|k-1}(X_k|X)p_{k-1}(X|Z_{1:k-1})\mu_s(dX) \tag{4.10}$$

where $p_k(\cdot|Z_{1:k})$ contains the entire information about multiple-target state X_k and where μ_s is an appropriate reference measure on $\mathcal{F}(\mathcal{X})$ [104]. Solving the integrals in (4.9) is intractable; thus, I need an approximation to make multiple-target tracking feasible. For instance, the PHD filter bypasses this intractability.

4.3 PHD Filtering

By using the PHD filter to approximate the optimal multiple-target Bayes filter recursion, I propagate the FISST-based first moment of the multiple-target posterior density, also known as the posterior intensity or probability hypothesis density [101]. In comparison to the posterior density, integrating the posterior intensity does not yield a unit value. The intensity D_X is a nonnegative function. Integrating D_X in a region $\mathcal{S}$ yields the expected number of targets in that region, i.e.,

$$\int_{\mathcal{S}} D_X(\mathbf{x})d\mathbf{x} = \mathbb{E}\{|\mathcal{S} \cap X|\}\,. \tag{4.11}$$

For any state $\mathbf{x}$, its intensity at this specific point, $\mathbf{x}$, is $D_X(\mathbf{x})$. The intensity's local maxima are vectors in $\mathcal{X}$ and, thus, potential target states. The total number of targets, $\widehat{N}_{\mathbf{x}}$, in a scene is

$$\int D_X(\mathbf{x})d\mathbf{x} = \widehat{N}_{\mathbf{x}}. \tag{4.12}$$

To estimate the elements in X, I choose the $\lfloor \widehat{N}_{\mathbf{x}} \rceil$ highest peaks in $D_X(\mathbf{x})$, where $\lfloor\cdot\rceil$ denotes rounding to the nearest integer.

As described in [97], considering the following assumptions enables us to successfully employ the PHD filter. First, each target evolves independently. Second, each target generates observations independently. Third, spurious observations (or clutter) are Poisson distributed and independent of observations generated by targets. Fourth, the multiple-target predicted density $p_{k|k-1}$ is Poisson-distributed. Furthermore, assume that the densities are approximately intensities, $p_k \sim D_k$ and $p_{k|k-1} \sim D_{k|k-1}$; then, after applying FISST [101] to (4.9) and (4.10), the posterior intensity is

$$D_k(\mathbf{x}) = [1 - p_{D,k}(\mathbf{x}))]\, D_{k|k-1}(\mathbf{x}) + \sum_{\mathbf{z}\in Z_k} \frac{p_{D,k}(\mathbf{x})g_k(\mathbf{z}|\mathbf{x})D_{k|k-1}(\mathbf{x})}{\kappa_k(\mathbf{z}) + \int p_{D,k}(\xi)g_k(\mathbf{z}|\xi)D_{k|k-1}(\xi)d\xi} \tag{4.13}$$

with its prior intensity

$$D_{k|k-1}(\mathbf{x}) = \int p_{S,k}f_{k|k-1}(\mathbf{x}|\xi)D_{k-1}(\xi)d\xi + \int \beta_{k|k-1}(\mathbf{x}|\xi)D_{k-1}(\xi)d\xi + \gamma_k(\mathbf{x}), \tag{4.14}$$

where $\beta_{k|k-1}$ is the intensity of $B_{k|k-1}$, γ_k is the intensity of Γ_k, and κ_k is the clutter intensity of K_k. For instance, the clutter intensity can be modeled as follows:

$$\kappa_k(\mathbf{z}_k) = \lambda_c \cdot c_k(\mathbf{z}_k) \tag{4.15}$$

with λ_c as the average number of Poisson-distributed false alarms and $c_k(\mathbf{z}_k)$ as the clutter distribution. The intensities, (4.13) and (4.14), are related to (4.6) and (4.8).

To sum it up, the PHD filter operates on the single-target state space, whereas the multiple-target Bayes filter operates on the multiple-target state space, as formulated in (4.9) and (4.10). However, the PHD recursion, it involves multi-dimensional integrals, lacks a closed-form solution in general [105]. Assuming linear Gaussian multiple-target models bypasses this problem; it yields a closed-form solution of the PHD recursion: the Gaussian mixture probability hypothesis density (GM-PHD) filter.

4.4 GM-PHD Filtering

When employing the GM-PHD filter, I additionally have to consider the following assumptions: First, each target moves according to a linear Gaussian dynamical model with state transition matrix F_{k-1}, mean $F_{k-1}\xi$, and process noise covariance Q_{k-1}:

$$f_{k|k-1}(\mathbf{x}|\xi) = \mathcal{N}\left(\mathbf{x}; F_{k-1}\xi, Q_{k-1}\right). \tag{4.16}$$

Second, each sensor observes a scene according to a linear Gaussian observation model

$$g_k(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mathbf{z}; H_k\mathbf{x}, R_k\right), \tag{4.17}$$

where H_k is the observation matrix, $H_k\mathbf{x}$ is the mean, and R_k is the observation noise covariance matrix. Third, the survival probabilities and detection probabilities are independent of the states, i.e., $p_{S,k}(\mathbf{x}) = p_{S,k}$ and $p_{D,k}(\mathbf{x}) = p_{D,k}$. Fourth, the birth intensity as well as the spawning intensity are based on Gaussian mixtures according to

$$\gamma_k(\mathbf{x}) = \sum_{i=1}^{N_{\gamma,k}} w_{\gamma,k}^{(i)} \mathcal{N}\left(\mathbf{x}; m_{\gamma,k}^{(i)}, P_{\gamma,k}^{(i)}\right) \tag{4.18}$$

and

$$\beta_{k|k-1}(\mathbf{x}|\xi) = \sum_{i=1}^{N_{\beta,k}} w_{\beta,k}^{(i)} \mathcal{N}(\mathbf{x}; F_{\beta,k-1}^{(i)}\xi + d_{\beta,k-1}^{(i)}, Q_{\beta,k-1}^{(i)}), \tag{4.19}$$

where $N_{\gamma,k}$ is the number of born targets, $N_{\beta,k}$ denotes the number of spawned targets, $w_{\gamma,k}$ and $w_{\beta,k}$ are the born and spawned targets' weights, respectively, $P_{\gamma,k}$ represents the born target's covariance matrix, and where $d_{\beta,k-1}$ defines the difference between the spawned target and its parent target. The means m represent the intensities' peaks.

4.4.1 The Previous Posterior Intensity

The posterior intensity at time index $k-1$ is a Gaussian mixture:

$$D_{k-1}(\mathbf{x}) = \sum_{i=1}^{N_{k-1}} w_{k-1}^{(i)} \mathcal{N}\left(\mathbf{x}; m_{k-1}^{(i)}, P_{k-1}^{(i)}\right), \tag{4.20}$$

where N_{k-1} is the previous posterior intensity's number of Gaussian components at time index $k-1$ (i.e., after predicting and updating the states in the previous instant

of time). Its corresponding track table, $\mathcal{L}_{k-1}$, consists of a set of quadruples (4-tuples), which represent the weights, the means, the covariances, and the labels, $l_{k-1}^{(i)}$, of each state:

$$\mathcal{L}_{k-1} = \left\{ \left(l_{k-1}^{(i)}, w_{k-1}^{(i)}, m_{k-1}^{(i)}, P_{k-1}^{(i)} \right) \right\}_{i=1}^{N_{k-1}}. \tag{4.21}$$

4.4.2 The Prediction Intensity

The predicted intensity at time index k is a Gaussian mixture consisting of the surviving intensity $D_{S,k|k-1}(\mathbf{x})$, the spawning intensity $D_{\beta,k|k-1}(\mathbf{x})$, and the birth intensity $\gamma_k(\mathbf{x})$ according to

$$D_{k|k-1}(\mathbf{x}) = D_{S,k|k-1}(\mathbf{x}) + D_{\beta,k|k-1}(\mathbf{x}) + \gamma_k(\mathbf{x}) \tag{4.22}$$

with the surviving intensity

$$D_{S,k|k-1}(\mathbf{x}) = \sum_{i=1}^{N_{k-1}} w_{S,k|k-1}^{(i)} \mathcal{N}\left(\mathbf{x}; m_{S,k|k-1}^{(i)}, P_{S,k|k-1}^{(i)}\right), \tag{4.23}$$

where the weights, the means, and the covariances of the surviving targets are

$$w_{S,k|k-1}^{(i)} = p_{S,k} w_{k-1}^{(i)}, \tag{4.24}$$

$$m_{S,k|k-1}^{(i)} = F_{k-1} m_{k-1}^{(i)}, \tag{4.25}$$

$$P_{S,k|k-1}^{(i)} = Q_{k-1} + F_{k-1} P_{k-1}^{(i)} F_{k-1}^{T}, \tag{4.26}$$

and where the spawning intensity is

$$D_{\beta,k|k-1}(\mathbf{x}) = \sum_{i_1=1}^{N_{k-1}} \sum_{i_2=1}^{N_{\beta,k}} w_{\beta,k|k-1}^{(i_1,i_2)} \mathcal{N}\left(\mathbf{x}; m_{\beta,k|k-1}^{(i_1,i_2)}, P_{\beta,k|k-1}^{(i_1,i_2)}\right) \tag{4.27}$$

with the means and the covariances of the spawned targets

$$w_{\beta,k|k-1}^{(i_1,i_2)} = w_{k-1}^{(i_1)} w_{\beta,k}^{(i_2)}, \tag{4.28}$$

$$m_{\beta,k|k-1}^{(i_1,i_2)} = F_{\beta,k-1}^{(i_2)} m_{k-1}^{(i_1)} + d_{\beta,k-1}^{(i_2)}, \tag{4.29}$$

$$P_{\beta,k|k-1}^{(i_1,i_2)} = Q_{\beta,k-1}^{(i_2)} + F_{\beta,k-1}^{(i_2)} P_{\beta,k-1}^{(i_1)} \left(F_{\beta,k-1}^{(i_2)}\right)^{T}. \tag{4.30}$$

The track table corresponding to the predicted intensity is

$$\mathcal{L}_{k|k-1} = \left\{ \left(l_{S,k|k-1}^{(i)}, w_{S,k|k-1}^{(i)}, m_{S,k|k-1}^{(i)}, P_{S,k|k-1}^{(i)} \right) \right\}_{i=1}^{N_{k-1}} \tag{4.31}$$

$$\cup \left\{ \left(l_{\beta,k|k-1}^{(i_1,i_2)}, w_{\beta,k|k-1}^{(i_1,i_2)}, m_{\beta,k|k-1}^{(i_1,i_2)}, P_{\beta,k|k-1}^{(i_1)} \right) \right\}_{i_1=1,\ldots,N_{k-1}}^{i_2=1,\ldots,N_{\beta,k}} \tag{4.32}$$

$$\cup \left\{ \left(l_{\gamma,k}^{(i)}, w_{\gamma,k}^{(i)}, m_{\gamma,k}^{(i)}, P_{\gamma,k}^{(i)} \right) \right\}_{i=1}^{N_{\gamma,k}}. \tag{4.33}$$

I assign randomly generated labels, $l_{\gamma,k^{(i)}}$ and $l^{(i_1,i_2)}_{\beta,k|k-1}$, to the born components and the spawned components, respectively. The surviving components maintain their labels, i.e., $l^{(i)}_{S,k|k-1} = l^{(i)}_{k-1}$.

Considering that all intensities are Gaussian mixtures, I rewrite (4.22) according to

$$D_{k|k-1}(\mathbf{x}) = \sum_{i=1}^{N_{k|k-1}} w^{(i)}_{k|k-1} \mathcal{N}\left(\mathbf{x}; m^{(i)}_{k|k-1}, P^{(i)}_{k|k-1}\right), \tag{4.34}$$

where $N_{k|k-1}$ is the prediction intensity's number of the Gaussian components.

4.4.3 The Posterior Intensity

As in case of the previous posterior intensity and the prior intensity, the posterior intensity is a Gaussian mixture given by

$$D_k(\mathbf{x}) = (1 - p_{D,k}) D_{k|k-1}(\mathbf{x}) + \sum_{\mathbf{z} \in Z_k} D_{D,k}(\mathbf{x}, \mathbf{z}) \tag{4.35}$$

with

$$D_{D,k}(\mathbf{x}, \mathbf{z}) = \sum_{i=1}^{N_{k|k-1}} w^{(i)}_k(\mathbf{z}) \mathcal{N}\left(\mathbf{x}; m^{(i)}_k(\mathbf{z}), P^{(i)}_k\right), \tag{4.36}$$

where the weights, the means, and the covariances of the detected targets are

$$w^{(i)}_k(\mathbf{z}) = \frac{p_{D,k} w^{(i)}_{k|k-1} q^{(i)}_k(\mathbf{z})}{\kappa_k(\mathbf{z}) + p_{D,k} \sum_{\iota=1}^{N_{k|k-1}} w^{(\iota)}_{k|k-1} q^{(\iota)}_k(\mathbf{z})}, \tag{4.37}$$

$$m^{(i)}_k(\mathbf{z}) = m^{(i)}_{k|k-1} + K^{(i)}_k \cdot \left(\mathbf{z} - H_k m^{(i)}_{k|k-1}\right), \tag{4.38}$$

$$P^{(i)}_k = \left(I - K^{(i)}_k H_k\right) P^{(i)}_{k|k-1} \tag{4.39}$$

with

$$K^{(i)}_k = P^{(i)}_{k|k-1} H^T_k \cdot \left(H_k P^{(i)}_{k|k-1} H^T_k + R_k\right)^{-1}, \tag{4.40}$$

$$q^{(i)}_k(\mathbf{z}) = \mathcal{N}\left(\mathbf{z}; H_k m^{(i)}_{k|k-1}, R_k + H_k P^{(i)}_{k|k-1} H^T_k\right). \tag{4.41}$$

The the updated components' track table consists of the predicted components' labels and the labels of the observation-corrected components:

$$\mathcal{L}_k = \left\{\left(l^{(i)}_k, w^{(i)}_k, m^{(i)}_k, P^{(i)}_k\right)\right\}_{i=1}^{N_{k|k-1}} \tag{4.42}$$

$$\cup \left\{\left(l^{(i)}_k(\mathbf{z}), w^{(i)}_k(\mathbf{z}), m^{(i)}_k(\mathbf{z}), P^{(i)}_k(\mathbf{z})\right)\right\}_{\mathbf{z} \in Z_k}^{i=1,\ldots,N_{k|k-1}} \tag{4.43}$$

The components which are independent of $\mathbf{z}$ maintain their preceding labels; each observation-dependent component retain the same label as its predecessor causing multiple components with the same label. Pruning and merging eliminate those multiples which feature low weights.

4.4.4 The Implementation

The relevant equations in order to implement the GM-PHD filter are (4.24)–(4.26), as well as (4.28)–(4.30) and (4.37)–(4.41). In [97, 106], the authors listed pseudo-codes and/or examples. When using the GM-PHD filter, the posterior intensities' number of Gaussian components increases without bounds. To limit this number, I prune components with small weights and merge components which are close together utilizing, e.g., the Mahalanobis distance. After updating all components, I select those with weights above a certain threshold and consider them as extracted multiple-target state estimates. For the next iteration, consider the remaining components and extracted components. See [97] for details on pruning and merging.

The PHD filter as well as the GM-PHD filter are both special cases of a more complex filter, the cardinalized probability hypothesis density (CPHD) filter, which also propagates probability mass functions of the number of targets.

4.5 CPHD Filtering

In 2007, Mahler derived a generalization of the PHD recursion: the cardinalized probability hypothesis density recursion [107]. It jointly propagates the posterior intensity and the posterior cardinality (i.e., the probability mass function of the targets' number) [98,108]. In general, this recursion is intractable; however, there exists a closed-form solution in case of linear Gaussian dynamics and birth processes. In comparison to the PHD filter, it improves the accuracy of estimating states, and it decreases the variance of the estimated number of targets.

The PHD filter models the cardinality of targets by employing the Poisson distribution. The distribution's mean is equal to its variance; thus, for a high number of targets, the PHD filter propagates the cardinality information with a high variance. Stated differently, the PHD recursion's principal weakness is the loss of higher order cardinality information [98, 109].

In comparison to the PHD filter, the CPHD filter additionally propagates the posterior cardinality mass function; the function depends on the posterior intensity. The principal weakness of the CPHD filter is the filter's complex intensity functions. The assumptions that need to be considered to employ the CPHD filter are similar to the PHD filter's assumptions [98]. The only difference is the clutter process's RFS, which is independent and identically distributed.

The following equations are a prerequisite to analyze the intensity functions and the cardinality mass functions. I denote the binomial coefficient as $C_{i_2}^{i_1} = i_1!/(i_2!(i_1 - i_2)!)$, the permutation coefficient as $P_{i_2}^{i_1} = i_1!/(i_1 - i_2)!$, the inner product of two real sequences as $\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=0}^{\infty} x[i]y[i]$, and I represent the elementary symmetric function as

$$e_i(Z) = \sum_{S \subseteq Z, |S|=i} \left(\prod_{\zeta \in S} \zeta \right)$$

with $e_i(0) = 1$.

The posterior intensity and the posterior cardinality mass function at time index $k-1$ are D_{k-1} and p_{k-1}, respectively. Considering $p_{\Gamma,k}$ as the cardinality mass function

of births at time index k, the predicted cardinality mass function is

$$p_{k|k-1}[n] = \sum_{i_2=0}^{n} p_{\Gamma,k}[n-i_2] \sum_{i_1=i_2}^{\infty} C_{i_2}^{i_1}\, p_{k-1}[i_1] \frac{\langle p_{S,k}, D_{k-1}\rangle^{i_2} \langle 1-p_{S,k}, D_{k-1}\rangle^{i_1-i_2}}{\langle 1, D_{k-1}\rangle^{i_1}}, \quad (4.44)$$

which is the convolution of the cardinality mass functions of the born targets and the surviving targets. The predicted intensity is

$$D_{k|k-1}(\mathbf{x}) = \int p_{S,k}(\zeta) f_{k|k-1}(\mathbf{x}|\zeta) D_{k-1}(\zeta) d\zeta + \gamma_k(\mathbf{x}). \quad (4.45)$$

Updating the predicted intensity $D_{k|k-1}$ and the predicted cardinality mass function $p_{k|k-1}$ yields the updated cardinality mass function,

$$p_k[n] = \frac{\Upsilon_k^0[D_{k|k-1}, Z_k](n) \cdot p_{k|k-1}[n]}{\langle \Upsilon_k^0[D_{k|k-1}, Z_k], p_{k|k-1}\rangle}, \quad (4.46)$$

and the updated intensity,

$$\begin{aligned} D_k(\mathbf{x}) = {} & \frac{\langle \Upsilon_k^1[D_{k|k-1}, Z_k], p_{k|k-1}\rangle}{\langle \Upsilon_k^0[D_{k|k-1}, Z_k], p_{k|k-1}\rangle} \big(1 - p_{D,k}(\mathbf{x})\big) D_{k|k-1}(\mathbf{x}) \\ & + \sum_{\mathbf{z}\in Z_k} \frac{\langle \Upsilon_k^1[D_{k|k-1}, Z_k \setminus \{\mathbf{z}\}], p_{k|k-1}\rangle}{\langle \Upsilon_k^0[D_{k|k-1}, Z_k], p_{k|k-1}\rangle} \psi_{k,\mathbf{z}}(\mathbf{x}) D_{k|k-1}(\mathbf{x}), \end{aligned} \quad (4.47)$$

with

$$\Upsilon_k^u[D, Z](n) = \sum_{i=0}^{\min(|Z|, n-u)} (|Z|-i)!\, p_{K,k}(|Z|-i) P_{i+u}^n \frac{\langle 1-p_{D,k}, D\rangle^{n-i-u}}{\langle 1, D\rangle^n} e_i\left(\Lambda_k(D, Z)\right), \quad (4.48)$$

as well as

$$\psi_{k,\mathbf{z}}(\mathbf{x}) = \frac{\langle 1, \kappa_k\rangle}{\kappa_k(\mathbf{z})} g_k(\mathbf{z}|\mathbf{x}) p_{D,k}(\mathbf{x}) \quad (4.49)$$

and

$$\Lambda_k(D, Z) = \{\langle D, \psi_{k,\mathbf{z}}\rangle : \mathbf{z} \in Z\}; \quad (4.50)$$

the expression $Z_k \setminus \{\mathbf{z}\}$ denotes the set of all observations at time index k without the observation $\mathbf{z}$.

4.6 GM-CPHD Filtering

For a special class of linear Gaussian multiple-target models, there exists a closed-form solution of the CPHD recursion. The assumptions made in case of the GM-PHD filter also hold in case of the GM-CPHD filter. Considering all those assumptions, I can propagate the posterior intensity and the posterior cardinality mass function over time.

4.6.1 The Previous Posterior Intensity

The posterior intensity at time index $k-1$ is a Gaussian mixture of the form

$$D_{k-1}(\mathbf{x}) = \sum_{i=1}^{N_k-1} w_{k-1}^{(i)} \mathcal{N}\left(\mathbf{x}; m_{k-1}^{(i)}, P_{k-1}^{(i)}\right). \tag{4.51}$$

Its corresponding track table is identical to the GM-PHD filter's track table:

$$\mathcal{L}_{k-1} = \left\{\left(l_{k-1}^{(i)}, w_{k-1}^{(i)}, m_{k-1}^{(i)}, P_{k-1}^{(i)}\right)\right\}_{i=1}^{N_{k-1}}. \tag{4.52}$$

4.6.2 The Prediction Intensity and Cardinality Mass Function

The predicted cardinality mass function, which consists of the cardinality mass function of births, $p_{\Gamma,k}$, and a combinatorial term, is

$$p_{k|k-1}[n] = \sum_{i_2=0}^{n} p_{\Gamma,k}[n-i_2] \sum_{i_1=i_2}^{\infty} C_{i_2}^{i_1}\, p_{k-1}[i_1] p_{S,k}^{i_2} (1-p_{S,k})^{i_1-i_2}, \tag{4.53}$$

and the predicted intensity at time index k is a Gaussian mixture according to

$$D_{k|k-1}(\mathbf{x}) = D_{S,k|k-1}(\mathbf{x}) + \gamma_k(\mathbf{x}) \tag{4.54}$$

with the same surviving intensity as in (4.23). Considering that all intensities are Gaussian mixtures, I rewrite (4.54) as follows:

$$D_{k|k-1}(\mathbf{x}) = \sum_{i=1}^{N_{k|k-1}} w_{k|k-1}^{(i)} \mathcal{N}\left(\mathbf{x}; m_{k|k-1}^{(i)}, P_{k|k-1}^{(i)}\right). \tag{4.55}$$

Due to simplicity, I omit the spawned components, which yields the following track table:

$$\begin{aligned}\mathcal{L}_{k|k-1} = &\left\{\left(l_{S,k|k-1}^{(i)}, w_{S,k|k-1}^{(i)}, m_{S,k|k-1}^{(i)}, P_{S,k|k-1}^{(i)}\right)\right\}_{i=1}^{N_{k-1}} \\ &\cup \left\{\left(l_{\gamma,k}^{(i)}, w_{\gamma,k}^{(i)}, m_{\gamma,k}^{(i)}, P_{\gamma,k}^{(i)}\right)\right\}_{i=1}^{N_{\gamma,k}}.\end{aligned} \tag{4.56}$$

4.6.3 The Posterior Intensity and Cardinality Mass Function

The posterior cardinality mass function is

$$p_k[n] = \frac{\Psi_k^0[w_{k|k-1}, Z_k](n) p_{k|k-1}[n]}{\langle \Psi_k^0[w_{k|k-1}, Z_k], p_{k|k-1}\rangle}, \tag{4.57}$$

whereas the posterior intensity is a Gaussian mixture given by

$$\begin{aligned}D_k(\mathbf{x}) = &\frac{\Psi_k^1[w_{k|k-1}, Z_k] p_{k|k-1}[n]}{\langle \Psi_k^0[w_{k|k-1}, Z_k], p_{k|k-1}\rangle}(1-p_{D,k}) D_{k|k-1}(\mathbf{x}) \\ &+ \sum_{\mathbf{z}\in Z_k} \sum_{i=1}^{N_{k|k-1}} w_k^{(i)}(\mathbf{z}) \mathcal{N}\left(\mathbf{x}; m_k^{(i)}(\mathbf{z}), P_k^{(i)}\right),\end{aligned} \tag{4.58}$$

where

$$\Psi_k^u[w,Z](n) = \sum_{i=0}^{\min(|Z|,n-u)} (|Z|-i)!\, p_{K,k}(|Z|-i) P_{i+u}^n \frac{(1-p_{D,k})^{n-i-u}}{\langle 1, w\rangle^{i+u}} e_i\left(\Lambda_k(w,Z)\right), \tag{4.59}$$

$$\Lambda_k(w,Z) = \left\{ \frac{\langle 1, \kappa_k\rangle}{\kappa_k(\mathbf{z})} p_{D,k} w^T q_k(\mathbf{z}) : \mathbf{z} \in Z \right\}, \tag{4.60}$$

$$w_{k|k-1} = \left(w_{k|k-1}^{(1)}, \ldots, w_{k|k-1}^{(N_{k|k-1})} \right)^T, \tag{4.61}$$

$$q_k(\mathbf{z}) = \left(q_k^{(1)}(\mathbf{z}), \ldots, q_k^{(N_{k|k-1})}(\mathbf{z}) \right)^T. \tag{4.62}$$

(Note that there are typographical errors in several publications of Vo et al.; however, in my thesis I have already considered the corrected equations listed in [108] on page 5816. I verified this by reimplementing the algorithm in Julia; experiments with the corrected equations yielded plausible results.) The parameters of the corresponding means and covariances of the detected targets are identical to the PHD filter's means and covariances. However, the weights differ from (4.37) as follows:

$$w_k^{(i)}(\mathbf{z}) = p_{D,k} w_{k|k-1}^{(i)} q_k^{(i)}(\mathbf{z}) \frac{\langle \Psi_k^1[w_{k|k-1}, Z_k \setminus \{\mathbf{z}\}], p_{k|k-1}\rangle}{\langle \Psi_k^0[w_{k|k-1}, Z_k], p_{k|k-1}\rangle} \frac{\langle 1, \kappa_k\rangle}{\kappa_k(\mathbf{z})}. \tag{4.63}$$

The track table of the updated components consists of the predicted components' labels and the labels of the observation-corrected components:

$$\mathcal{L}_k = \left\{ \left(l_k^{(i)}, w_k^{(i)}, m_k^{(i)}, P_k^{(i)} \right) \right\}_{i=1}^{N_{k|k-1}} \tag{4.64}$$

$$\cup \left\{ \left(l_k^{(i)}(\mathbf{z}), w_k^{(i)}(\mathbf{z}), m_k^{(i)}(\mathbf{z}), P_k^{(i)}(\mathbf{z}) \right) \right\}_{\mathbf{z}\in Z_k}^{i=1,\ldots,N_{k|k-1}}. \tag{4.65}$$

This track table is identical to the GM-PHD filter's track table.

4.6.4 The Implementation

The relevant equations in order to implement the GM-CPHD filter are listed in the following. To compute the means and the covariances, I consider (4.24)–(4.26) as well as (4.38)–(4.41). For the weights, (4.63), I compute all cardinality mass functions, i.e., (4.53) and (4.57) for all possible numbers of targets. Even in case of the GM-CPHD filter, I have to limit the number of the posterior intensity's components; thus, I employ pruning and merging. Due to the cardinality mass function's infinite number of components, I have to set a maximum number of possible targets, which is larger than the number of targets on the scene at any time. A resource-consuming step is the computation of the elementary symmetric function for a large number of states. To save resources, I utilize Newton-Girard formulas or employ Vieta's theorem [110]. To extract components representing possible targets, I apply the same procedure as explained in the chapter about the GM-PHD filter. To estimate the number of targets, I employ an expected a posteriori (EAP) estimator, $N_{\mathrm{EAP},k} = \mathbb{E}\{|X_k|\}$ or a maximum a posteriori (MAP) estimator, $N_{\mathrm{MAP},k} = \arg\max p_k(\cdot)$.

Until now, I explained filters based on Poisson RFSs. However, there is another one that assumes Bernoulli RFSs: the cardinality-balanced multi-target multi-Bernoulli (CBMeMBer) filter. It requires less computational resources than the CPHD filter.

4.7 CBMeMBer Filtering

The multi-target multi-Bernoulli (MeMBer) recursion proposed by Mahler approximates the Bayes multiple-target recursion if the clutter density is low [95]. Instead of propagating moments and/or cardinality mass functions as in case of the PHD recursion and the CPHD recursion, the MeMBer recursion approximately propagates the multiple-target posterior density (whereas the PHD recursion and the CPHD recursion operate on the single-target state space). Although the MeMBer filter propagates the multiple-target posterior density, its performance is similar to the performance of the PHD filter. More precisely, the recursion propagates a multi-Bernoulli RFS's parameters, which approximates the posterior multiple-target RFS. However, Mahler's MeMBer recursion described in [95] features a bias in cardinality when updating the predicted tracks, as described in [99]. To bypass this bias, Vo et al. introduced the cardinality-balanced MeMBer filter, which propagates a set of multi-Bernoulli parameters that characterize the posterior multiple-target RFS.

To highlight the differences between the MeMBer recursion and the CPHD and PHD recursions, I first need to recap the differences between a random vector and an RFS. A random vector generates a fixed number and order of randomly sampled points, whereas the RFS's number of sampled points and the order of these points are random. In other words, an RFS is a finite-set valued random variable, and its probability mass function or probability density function describes its randomness.

In case of the PHD filter and the CPHD filter, I assumed a Poisson RFS characterized by its intensity function D, where $\widehat{N}_{\mathbf{x}} = \int D(\mathbf{x})d\mathbf{x}$ is its cardinality and $D(\cdot)/\widehat{N}_{\mathbf{x}}$ is its density

In contrast to the PHD filter and the CPHD filter (both are based on Poisson RFSs) I now have to consider Bernoulli RFSs. A Bernoulli RFS is empty with a probability of $1-r$ and is a singleton with probability r. A probability density p distributes the set's element according to

$$\pi(X) = \begin{cases} 1-r & X = \emptyset \\ r \cdot p(\mathbf{x}) & X = \{\mathbf{x}\} \end{cases} \tag{4.66}$$

A random finite set of multiple targets requires a union of a fixed number of independent Bernoulli RFSs, $X = \bigcup_{i=1}^{M_{\mathbf{x}}} X^{(i)}$. A set's existence probability is $r^{(i)} \in (0,1)\}$, i.e., it describes the probability that the i-th hypothesized track is a true track. A set's probability density is $p^{(i)}$, which describes the estimated current state of the track [99]. The parameter set $\{(r^{(i)}, p^{(i)})\}_{i=1}^{M_{\mathbf{x}}}$ describes a multi-Bernoulli RFS with $M_{\mathbf{x}}$ as the number of tracks. The multi-Bernoulli RFS's probability density is

$$\pi(\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}) = \pi(\emptyset) \sum_{1 \leq i_1 \neq \cdots \neq i_n \leq M_{\mathbf{x}}} \prod_{j=1}^{n} \frac{r^{(i_j)} p^{(i_j)}(\mathbf{x}_j)}{1 - r^{(i_j)}} \tag{4.67}$$

with $\pi(\emptyset) = \prod_{j=1}^{M_{\mathbf{x}}}(1 - r^{(j)})$; its abbreviated form is $\pi = \{(r^{(i)}, p^{(i)})\}_{i=1}^{M_{\mathbf{x}}}$. The Bernoulli RFS $S_{k|k-1}(\mathbf{x}_{k-1})$ with $r = p_{S,k}(\mathbf{x}_{k-1})$ and $p(\cdot) = f_{k|k-1}(\cdot|\mathbf{x}_{k-1})$ models the translation of state $\mathbf{x}_{k-1} \in X_{k-1}$ from time index $k-1$ to time index k. The Bernoulli RFS $O_k(\mathbf{x}_k)$ with $r = p_{D,k}(\mathbf{x}_k)$ and $p(\cdot) = g_k(\cdot|\mathbf{x}_k)$ describes how a target $\mathbf{x}_k \in X_k$ generates an observation $\mathbf{z}_k$.

The original MeMBer recursion described in [95] approximates the multiple-target Bayes recursion by employing multi-Bernoulli RFSs. By propagating a finite, time-varying number of hypothesized tracks, the recursion propagates the multiple-target posterior probability density over time. A probability of existence and a probability density of the current hypothesized state characterize each track.

The assumptions that need to be considered to employ the MeMBer filter are similar to those of the PHD filter and the CPHD filter except that the target births follow a multi-Bernoulli RFS, which is independent of the target survivals.

Given the number of persistent tracks $M_{\mathbf{x},k-1}$ and the multi-Bernoulli posterior multiple-target density at time index $k-1$,

$$\pi_{k-1} = \left\{ \left(r_{k-1}^{(i)}, p_{k-1}^{(i)} \right) \right\}_{i=1}^{M_{k-1}}, \tag{4.68}$$

the multi-Bernoulli predicted multiple-target density is the union of the multi-Bernoulli parameter sets of the surviving targets and target births,

$$\pi_{k|k-1} = \left\{ \left(r_{P,k|k-1}^{(i)}, p_{P,k|k-1}^{(i)} \right) \right\}_{i=1}^{M_{k-1}} \cup \left\{ \left(r_{\gamma,k}^{(i)}, p_{\gamma,k}^{(i)} \right) \right\}_{i=1}^{M_{\gamma,k}} \tag{4.69}$$

with the number of born tracks $M_{\gamma,k}$, the existence probability $r_{P,k|k-1}^{(i)} = r_{k-1}^{(i)} \langle p_{k-1}^{(i)}, p_{S,k} \rangle$, and the probability density $p_{P,k|k-1}^{(i)}(\mathbf{x}) = \langle f_{k|k-1}(\mathbf{x} \mid \cdot), p_{k-1}^{(i)} p_{S,k} \rangle / \langle p_{k-1}^{(i)}, p_{S,k} \rangle$, and with the parameters of the multi-Bernoulli RFS of births, $\{(r_{\gamma,k}^{(i)}, p_{\gamma,k}^{(i)})\}_{i=1}^{M_{\gamma,k}}$. Given the number of legacy tracks $M_{k|k-1}$ and the predicted multi-Bernoulli multiple-target density,

$$\pi_{k|k-1} = \left\{ \left(r_{k|k-1}^{(i)}, p_{k|k-1}^{(i)} \right) \right\}_{i=1}^{M_{k|k-1}}, \tag{4.70}$$

the approximated multi-Bernoulli posterior multiple-target density is the union of the multi-Bernoulli parameter sets of the legacy tracks and the observation-corrected tracks,

$$\pi_k \approx \left\{ \left(r_{L,k}^{(i)}, p_{L,k}^{(i)} \right) \right\}_{i=1}^{M_{k|k-1}} \cup \left\{ \left(r_{U,k}(\mathbf{z}), p_{U,k}(\cdot; \mathbf{z}) \right) \right\}_{\mathbf{z} \in Z_k} \tag{4.71}$$

with the existence probability of the legacy tracks

$$r_{L,k}^{(i)} = r_{k|k-1}^{(i)} \frac{1 - \langle p_{k|k-1}^{(i)}, p_{D,k} \rangle}{1 - r_{k|k-1}^{(i)} \langle p_{k|k-1}^{(i)}, p_{D,k} \rangle},$$

the probability density of the legacy tracks

$$p_{L,k}^{(i)}(\mathbf{x}) = p_{k|k-1}^{(i)}(\mathbf{x}) \frac{1 - p_{D,k}(\mathbf{x})}{1 - \langle p_{k|k-1}^{(i)}, p_{D,k} \rangle},$$

the existence probability of the observation-corrected tracks

$$r_{U,k}(\mathbf{z}) = \frac{\sum_{i=1}^{M_{k|k-1}} \frac{r_{k|k-1}^{(i)} \langle p_{k|k-1}^{(i)}, \psi_{k,\mathbf{z}} \rangle}{1 - r_{k|k-1}^{(i)} \langle p_{k|k-1}^{(i)}, p_{D,k} \rangle}}{\kappa_k(\mathbf{z}) + \sum_{i=1}^{M_{k|k-1}} \frac{r_{k|k-1}^{(i)} \langle p_{k|k-1}^{(i)}, \psi_{k,\mathbf{z}} \rangle}{1 - r_{k|k-1}^{(i)} \langle p_{k|k-1}^{(i)}, p_{D,k} \rangle}},$$

and the probability density of the observation-corrected tracks

$$p_{U,k}(\mathbf{x};\mathbf{z}) = \frac{\sum_{i=1}^{M_{k|k-1}} \frac{r_{k|k-1}^{(i)} p_{k|k-1}^{(i)} \psi_{k,\mathbf{z}}(\mathbf{x})}{1-r_{k|k-1}^{(i)}\langle p_{k|k-1}^{(i)}, p_{D,k}\rangle}}{\sum_{i=1}^{M_{k|k-1}} \frac{r_{k|k-1}^{(i)}\langle p_{k|k-1}^{(i)}, \psi_{k,\mathbf{z}}\rangle}{1-r_{k|k-1}^{(i)}\langle p_{k|k-1}^{(i)}, p_{D,k}\rangle}},$$

where $\psi_{k,\mathbf{z}}(\mathbf{x}) = g_k(\mathbf{z}|\mathbf{x})p_{D,k}(\mathbf{x})$. Without going into detail, I would like to highlight that Mahler [95] (unwittingly) introduced a cardinality bias. To significantly reduce this bias, Vo et al. [99] proposed a modified existence probability,

$$r_{U,k}^{*}(\mathbf{z}) = \frac{\sum_{i=1}^{M_{k|k-1}} \frac{r_{k|k-1}^{(i)}(1-r_{k|k-1}^{(i)})\langle p_{k|k-1}^{(i)}, \psi_{k,\mathbf{z}}\rangle}{(1-r_{k|k-1}^{(i)}\langle p_{k|k-1}^{(i)}, p_{D,k}\rangle)^2}}{\kappa_k(\mathbf{z}) + \sum_{i=1}^{M_{k|k-1}} \frac{r_{k|k-1}^{(i)}\langle p_{k|k-1}^{(i)}, \psi_{k,\mathbf{z}}\rangle}{1-r_{k|k-1}^{(i)}\langle p_{k|k-1}^{(i)}, p_{D,k}\rangle}}, \tag{4.72}$$

and a modified multi-target multi-Bernoulli posterior density according to

$$\pi_k \approx \left\{\left(r_{L,k}^{(i)}, p_{L,k}^{(i)}\right)\right\}_{i=1}^{M_{k|k-1}} \cup \left\{\left(r_{U,k}^{*}(\mathbf{z}), p_{U,k}^{*}(\cdot;\mathbf{z})\right)\right\}_{\mathbf{z}\in Z_k} \tag{4.73}$$

with a new probability density

$$p_{U,k}^{*}(\mathbf{x};\mathbf{z}) = \frac{\sum_{i=1}^{M_{k|k-1}} \frac{r_{k|k-1}^{(i)}}{1-r_{k|k-1}^{(i)}} p_{k|k-1}^{(i)} \psi_{k,\mathbf{z}}(\mathbf{x})}{\sum_{i=1}^{M_{k|k-1}} \frac{r_{k|k-1}^{(i)}}{1-r_{k|k-1}^{(i)}} \langle p_{k|k-1}^{(i)}, \psi_{k,\mathbf{z}}\rangle}, \tag{4.74}$$

which yields the cardinality-balanced multi-target multi-Bernoulli (CBMeMBer) filter. It features a similar complexity as the PHD filter and a lower complexity as the CPHD filter [95, 96, 99].

4.8 GM-CBMeMBer Filtering

As in case of the GM-CPHD filter, all but one assumption made in case of the GM-PHD filter also hold for the GM-CBMeMBer filter. The only difference is the birth model represented by a multi-Bernoulli parameter set $\{(r_{\gamma,k}^{(i)}, p_{\gamma,k}^{(i)})\}_{i=1}^{M_{\gamma,k}}$ with

$$p_{\gamma,k}^{(i_1)}(\mathbf{x}) = \sum_{i_2=1}^{N_{\gamma,k}^{(i_1)}} w_{\gamma,k}^{(i_1,i_2)} \mathcal{N}\left(\mathbf{x}; m_{\gamma,k}^{(i_1,i_2)}, P_{\gamma,k}^{(i_1,i_2)}\right). \tag{4.75}$$

4.8.1 The Previous Posterior Multiple-Target Density

The multi-Bernoulli posterior multiple-target density at time index $k-1$ is

$$\pi_{k-1} = \left\{\left(r_{k-1}^{(i)}, p_{k-1}^{(i)}\right)\right\}_{i=1}^{M_{k-1}}. \tag{4.76}$$

Its corresponding track table is

$$\mathcal{L}_{k-1} = \left\{\left(l_{k-1}^{(i)}, r_{k-1}^{(i)}, p_{k-1}^{(i)}\right)\right\}_{i=1}^{M_{k-1}}. \tag{4.77}$$

4.8.2 The Predicted Multiple-Target Density

The predicted multi-Bernoulli multiple-target density of the form

$$\pi_{k|k-1} = \left\{ \left(r_{P,k|k-1}^{(i)}, p_{P,k|k-1}^{(i)} \right) \right\}_{i=1}^{M_{k-1}} \cup \left\{ \left(r_{\gamma,k}^{(i)}, p_{\gamma,k}^{(i)} \right) \right\}_{i=1}^{M_{\gamma,k}} \tag{4.78}$$

as well as the predicted track table,

$$\mathcal{L}_{k|k-1} = \left\{ \left(l_{P,k|k-1}^{(i)}, r_{P,k|k-1}^{(i)}, p_{P,k|k-1}^{(i)} \right) \right\}_{i=1}^{M_{k-1}} \cup \left\{ \left(l_{\gamma,k}^{(i)}, r_{\gamma,k}^{(i)}, p_{\gamma,k}^{(i)} \right) \right\}_{i=1}^{M_{\gamma,k}}, \tag{4.79}$$

consist of the existence probability

$$r_{P,k|k-1}^{(i)} = r_{k-1}^{(i)} p_{S,k}, \tag{4.80}$$

the probability density

$$p_{P,k|k-1}^{(i_1)}(\mathbf{x}) = \sum_{i_2=1}^{N_{k-1}^{(i_1)}} w_{k-1}^{(i_1,i_2)} \mathcal{N}\left(\mathbf{x}; m_{P,k|k-1}^{(i_1,i_2)}, P_{P,k|k-1}^{(i_1,i_2)} \right), \tag{4.81}$$

and the existence probabilities and probability densities of the birth model. The survived targets' labels are $l_{P,k|k-1}^{(i)} = l_{k-1}^{(i)}$, i.e., they maintain their labels, whereas the born targets' labels, $l_{\gamma,k}^{(i)}$, are randomly generated.

4.8.3 The Posterior Multiple-Target Density

The posterior multi-Bernoulli multiple-target density of the form

$$\pi_k = \left\{ \left(r_{L,k}^{(i)}, p_{L,k}^{(i)} \right) \right\}_{i=1}^{M_{k|k-1}} \cup \left\{ \left(r_{U,k}^{*}(\mathbf{z}), p_{U,k}^{*}(\cdot;\mathbf{z}) \right) \right\}_{\mathbf{z} \in Z_k} \tag{4.82}$$

as well as its corresponding track table,

$$\mathcal{L}_k = \left\{ \left(l_{L,k}^{(i)}, r_{L,k}^{(i)}, p_{L,k}^{(i)} \right) \right\}_{i=1}^{M_{k|k-1}} \cup \left\{ \left(l_{U,k}(\mathbf{z}), r_{U,k}^{*}(\mathbf{z}), p_{U,k}^{*}(\cdot;\mathbf{z}) \right) \right\}_{\mathbf{z} \in Z_k}, \tag{4.83}$$

consist of the existence probability of the legacy tracks,

$$r_{L,k}^{(i)} = r_{k|k-1}^{(i)} \frac{1 - p_{D,k}}{1 - r_{k|k-1}^{(i)} p_{D,k}}, \tag{4.84}$$

the probability density of the legacy tracks,

$$p_{L,k}^{(i)}(\mathbf{x}) = p_{k|k-1}^{(i)}(\mathbf{x}), \tag{4.85}$$

the existence probability of the observation-corrected tracks

$$r_{U,k}^{*}(\mathbf{z}) = \frac{\sum_{i=1}^{M_{k|k-1}} \frac{r_{k|k-1}^{(i)} \left(1 - r_{k|k-1}^{(i)}\right) \varrho_{U,k}^{(i)}(\mathbf{z})}{\left(1 - r_{k|k-1}^{(i)} p_{D,k}\right)^2}}{\kappa_k(\mathbf{z}) + \sum_{i=1}^{M_{k|k-1}} \frac{r_{k|k-1}^{(i)} \varrho_{U,k}^{(i)}(\mathbf{z})}{1 - r_{k|k-1}^{(i)} p_{D,k}}}, \tag{4.86}$$

and the probability density of the observation-corrected tracks

$$p^*_{U,k}(\mathbf{x};\mathbf{z}) = \frac{\sum_{i_1=1}^{M_{k|k-1}} \sum_{i_2=1}^{N_{k|k-1}^{(i_1)}} w_{U,k}^{(i_1,i_2)}(\mathbf{z}) \mathcal{N}\left(\mathbf{x}; m_{U,k}^{(i_1,i_2)}, P_{U,k}^{(i_1,i_2)}\right)}{\sum_{i_1=1}^{M_{k|k-1}} \sum_{i_2=1}^{N_{k|k-1}^{(i_1)}} w_{U,k}^{(i_1,i_2)}(\mathbf{z})}, \tag{4.87}$$

where

$$\varrho_{U,k}^{(i_1)}(\mathbf{z}) = p_{D,k} \sum_{i_2=1}^{N_{k|k-1}^{(i_1)}} w_{k|k-1}^{(i_1,i_2)} q_k^{(i_1,i_2)}(\mathbf{z}), \tag{4.88}$$

$$q_k^{(i_1,i_2)}(\mathbf{z}) = \mathcal{N}\left(\mathbf{z}; H_k m_{k|k-1}^{(i_1,i_2)}, H_k P_{k|k-1}^{(i_1,i_2)} H_k^T + R_k\right), \tag{4.89}$$

and

$$w_{U,k}^{(i_1,i_2)}(\mathbf{z}) = \frac{r_{k|k-1}^{(i_1)}}{1 - r_{k|k-1}^{(i_1)}} p_{D,k} w_{k|k-1}^{(i_1,i_2)} q_k^{(i_1,i_2)}(\mathbf{z}). \tag{4.90}$$

The legacy components' labels are $l_{L,k}^{(i)} = l_{k|k-1}^{(i)}$, i.e., they retain their labels, whereas the observation-updated components' labels are $l_{U,k}(\mathbf{z}) = l_{k|k-1}^{(i_1)}$, where

$$i_1 = \arg\max_{i_2} r_{k|k-1}^{(i_2)} (1 - r_{k|k-1}^{(i_2)}) p_{U,k}^{(i_2)}(\mathbf{z}) / (1 - r_{k|k-1}^{(i_2)} p_{D,k})^2 \,. \tag{4.91}$$

As explained in [99], an observation-updated component's label is the label of the predicted track which has the largest contribution to the current observation-updated probability of existence.

4.8.4 The Implementation

As in case of the GM-PHD filter and the GM-CPHD filter, the GM-CBMeMBer filter's number of Gaussian components increases without bound. Thus, I prune hypothesized tracks by rejecting those tracks with an existence probability below a certain threshold. To discard Gaussian components of the remaining tracks, I reject those components with weights below a certain threshold and merge those components featuring a distance to each other which is smaller than a certain threshold. The relevant equations in order to implement the GM-CBMeMBer filter are those used in case of the GM-PHD filter as well as (4.80), (4.84), (4.86), (4.88), (4.89), (4.90) instead of (4.37), and (4.91).

Given the multi-Bernoulli posterior multiple-target density after pruning and merging:

$$\pi_k = \left\{ \left(r_k^{(i)}, p_k^{(i)} \right) \right\}_{i=1}^{M_k}.$$

As described in [107] and in the code related to [99], I estimate the cardinality of this density (after considering pruning and merging) by utilizing the expected a posteriori estimator according to

$$M_{\mathrm{EAP},k} = \sum_{i=1}^{M_k} r_k^{(i)}, \tag{4.92}$$

or by employing the maximum a posteriori estimator according to

$$M_{\mathrm{MAP},k} = \arg\max_{i_2} e_{i_2}\left(\left\{r_k^{(i_1)}/(1-r_k^{(i_1)})\right\}_{i_1=1}^{M_k}\right) \tag{4.93}$$

where $i_1 = 0, \ldots, M_k$ and $e_{i_2}(\cdot)$ is the elementary symmetric function, or by computing the minimum of the number of remaining tracks and the maximum-a-posteriori-based cardinality,

$$M_{\mathrm{MIN},k} = \min\left(M_k, M_{\mathrm{MAP}}\right). \tag{4.94}$$

The cardinality's variance is

$$\sigma_{M,k}^2 = \sum_{i=1}^{M_k} r_k^{(i)} \cdot (1 - r_k^{(i)}). \tag{4.95}$$

4.9 Metrics

To evaluate the algorithms' performance, I employed metrics well known in the field of multiple-target tracking: the optimal subpattern assignment and label assignment for (multiple) tracks [82, 83].

4.9.1 Optimal Subpattern Assignment for Tracks

I originally wanted to apply the metrics precision and recall as well as the root mean square error to evaluate the trackers' performance. However, I wondered if there is a metric that incorporates the aforementioned ones. Searching for alternatives, I soon realized that publications (co-)authored by Ba-Ngu and Ba-Tuong Vo as well as Daniel Clark and Ronald Mahler will lead me to what I was looking for.

A common metric to describe a multiple-target filter's performance is the Hausdorff metric [111]; however, it does not take differences in cardinality, i.e., a set's number of elements, into account. To bypass this drawback, Hoffman and Mahler [112] proposed a new metric based on the Wasserstein distance. Unfortunately, it features other shortcomings listed in [111]. Years later, Schuhmacher et al. [111] evaluated several multiple-target filters by utilizing the OSPA distance, a metric that did not feature any shortcomings. Still, the metric was applicable to filtering algorithms only. (A multiple-target filtering algorithm sequentially estimates the number of states and their position in the state space; a multiple-target tracking algorithm outputs tracks: estimated temporal sequences of labeled states associated with targets.) To overcome this drawback, Ristic et al. [82, 83] proposed the OSPA distance for tracks: a metric defined on the space of finite sets of tracks, where each track is a labeled as a temporal sequence of states varying in length. It measures the distance between the set of ground-truth tracks, known a priori, and the set of estimated tracks. The metric's scores are typically averaged over independent Monte Carlo experiments. It combines various aspects of a multiple-target tracker's performance: the timeliness, the track accuracy, the continuity, the data association, the false tracks, etc. The metric is defined on the space of finite

sets of objects, i.e., tracks, over discrete-time support points $\mathcal{T} = (t_1, t_2, \ldots, t_k, \ldots, t_K)$. A track is a labeled sequence of objects, X_k, for time indices $k = \{1, 2, \ldots, K\}$,

$$X = (X_1, X_2, \ldots, X_K), \tag{4.96}$$

where X_k is either an empty set or a tuple consisting of a label l, the track's label, and a state vector $\mathbf{x}_k$, $(l, \mathbf{x}_k)$, $l \in \mathbb{N}$:

$$X_k = \begin{cases} \emptyset & \text{if } \mathbf{1}_k = 0 \\ \{(l, \mathbf{x}_k)\} & \text{if } \mathbf{1}_k = 1 \end{cases}; \tag{4.97}$$

variable $\mathbf{1}_k$ indicates a track's existence at time index k. A track's label does not change with time, but its state vector evolves in a state space. This space consists of the angular domain and the frequency domain as well as their corresponding velocity domains in order to represent a state, $\mathbf{x}_k = (\varphi_k, f_k, \dot{\varphi}_k, \dot{f}_k)^T$, where $\dot{\varphi}_k$ and $\dot{f}_k$ denote the derivatives, i.e., velocities, of the angular and frequency component, respectively. Ristic et al. defined the proposed metric at one particular discrete-time support point t_k (or time index k), where the set of all tracks is $\mathbb{X}_k$ ($X_k \in \mathbb{X}_k$) and where the set of finite subsets of $\mathbb{X}$ is $\mathcal{X}_k$. If $\mathcal{X}_k$ is an arbitrary non-empty set, then the metric maps a multi-dimensional argument to a real number: $\mathcal{D} : \mathcal{X}_k \times \mathcal{X}_k \to \mathbb{R}_+ = [0, \infty)$. According to [83], the proposed metric satisfies three axioms for all $\mathfrak{X}_k, \mathfrak{Z}_k, \mathfrak{W}_k \in \mathcal{X}_k$, where $\mathfrak{X}_k$ is the set of tracks representing the ground truth at time index k, $\mathfrak{Z}_k$ is the set of estimated tracks at time index k, and where $\mathfrak{W}_k$ is a non-empty set:

1. identity: $\mathcal{D}(\mathfrak{X}_k, \mathfrak{Z}_k) = 0$ in case of $\mathfrak{X}_k = \mathfrak{Z}_k$ only,
2. symmetry: $\mathcal{D}(\mathfrak{X}_k, \mathfrak{Z}_k) = \mathcal{D}(\mathfrak{Z}_k, \mathfrak{X}_k)$,
3. triangle inequality: $\mathcal{D}(\mathfrak{X}_k, \mathfrak{Z}_k) \leq \mathcal{D}(\mathfrak{X}_k, \mathfrak{W}_k) + \mathcal{D}(\mathfrak{W}_k, \mathfrak{Z}_k)$.

The OSPA distance, $\mathcal{D}(\mathfrak{X}_k, \mathfrak{Z}_k)$, between any two sets of tracks at time index k is a metric on $\mathcal{X}_k$ [82, 111],

$$\mathfrak{X}_k = \{(l_1, \mathbf{x}_{k,1}), \ldots, (l_{N_{M_k}}, \mathbf{x}_{k,N_{M_k}})\}, \tag{4.98}$$

$$\mathfrak{Z}_k = \{(\hat{l}_1, \mathbf{z}_{k,1}), \ldots, (\hat{l}_{N_{N_k}}, \mathbf{z}_{k,N_{N_k}})\}, \tag{4.99}$$

where N_{M_k} and N_{N_k} are the cardinalities of the sets $\mathfrak{X}_k$ and $\mathfrak{Z}_k$, respectively, and where $\hat{l}$ denotes an estimated track's label.

Due to a fixed-length permutation of a certain number of elements, I distinguish between two different cases: $N_{M_k} \leq N_{N_k}$ and $N_{M_k} > N_{N_k}$; i.e., the case where the number of elements of ground-truth tracks is equal or smaller than the number of estimated tracks, and the case where the number of ground-truth tracks is larger than the number of estimated tracks.

Given $N_{M_k} \leq N_{N_k}$, an OSPA order $1 \leq \eta_o < \infty$, a cutoff parameter a, and a set of permutations $\Pi_{N_{N_k}}$, where each permutation b consists of N_{M_k} elements taken from $\{1, 2, \ldots, N_{N_k}\}$, the OSPA distance between $\mathfrak{X}_k$ and $\mathfrak{Z}_k$ is defined as

$$\mathcal{D}_{\eta_o,a}(\mathfrak{X}_k, \mathfrak{Z}_k) = \left[\frac{1}{N_{N_k}} \left(\min_{b \in \Pi_{N_{N_k}}} \sum_{i=1}^{N_{M_k}} (\mathcal{A}_{\eta_o,a}(\bar{\mathbf{x}}_{k,i}, \bar{\mathbf{z}}_{k,b(i)}) + \mathcal{B}_{\eta_o,a}(N_{N_k}, N_{M_k})) \right) \right]^{\frac{1}{\eta_o}}, \tag{4.100}$$

with $\bar{\mathbf{x}}_{k,i} = (l_i, \mathbf{x}_{k.i})$ and $\bar{\mathbf{z}}_{k,i} = (\hat{l}_{b(i)}, \mathbf{z}_{k,b(i)})$. Variable a denotes the exponentiated scaling factor of the cardinality error component,

$$\mathcal{B}_{\eta_o,a}(N_{N_k}, N_{M_k}) = (N_{N_k} - N_{M_k}) \cdot a^{\eta_o}, \tag{4.101}$$

the limit of the base distance error component for a certain η_o,

$$\mathcal{A}_{\eta_o,a}(\bar{\mathbf{x}}_{k,i}, \bar{\mathbf{z}}_{k,b(i)}) = d_a\left(\bar{\mathbf{x}}_{k,i}, \bar{\mathbf{z}}_{k,b(i)}\right)^{\eta_o}, \tag{4.102}$$

and the upper limit of the cutoff distance

$$d_a(\bar{\mathbf{x}}_{k,i}, \bar{\mathbf{z}}_{k,b(i)}) = \min\left(a, d(\bar{\mathbf{x}}_{k,i}, \bar{\mathbf{z}}_{k,b(i)})\right) \tag{4.103}$$

between two tracks at time index k with $a > 0$. The base distance—as part of the cutoff distance—is

$$d(\bar{\mathbf{x}}_{k,i}, \bar{\mathbf{z}}_{k,b(i)}) = \sqrt[\eta_b]{d(\mathbf{x}_{k,i}, \mathbf{z}_{k,b(i)})^{\eta_b} + d(l_i, \hat{l}_{b(i)})^{\eta_b}} \tag{4.104}$$

with $1 \leq \eta_b \leq \infty$ as the base distance's order. The base distance consists of the localization base distance and the labeling error,

$$d(\mathbf{x}_{k,i}, \mathbf{z}_{k,b(i)}) = ||\mathbf{x}_{k,i} - \mathbf{z}_{k,b(i)}||_{\eta_b} \tag{4.105}$$

and

$$d(l_i, \hat{l}_{b(i)}) = c \cdot \left(1 - \delta[l_i, \hat{l}_{b(i)}]\right), \tag{4.106}$$

respectively, where $\delta[l_i, \hat{l}_{b(i)}]$ is the Kronecker delta yielding $\delta[l_i, \hat{l}_{b(i)}] = 1$ if $l_i = \hat{l}_{b(i)}$ and $\delta[l_i, \hat{l}_{b(i)}] = 0$ if $l_i \neq \hat{l}_{b(i)}$. Variable $c \in [0, a]$ represents a penalty assigned to the labeling error; there is no penalty if $c = 0$, whereas $c = a$ assigns the maximum penalty. As η_o increases, the OSPA distance between a ground-truth item and an estimated item increases, too. An increasing cutoff parameter a results in an increasing penalty for cardinality errors, i.e., it is the assigned error if an item is assumed to be unassignable.

If the number of ground-truth tracks is larger than the number of estimated tracks, i.e., $N_{M_k} > N_{N_k}$, I need to interchange the metric's arguments according to

$$\mathcal{D}_{\eta_o,a}(\mathfrak{X}_k, \mathfrak{Z}_k) \triangleq \mathcal{D}_{\eta_o,a}(\mathfrak{Z}_k, \mathfrak{X}_k). \tag{4.107}$$

If I would compute $\mathcal{D}_{\eta_o,a}(\mathfrak{X}_k, \mathfrak{Z}_k)$ in case of $N_{M_k} > N_{N_k}$, I would not be able to calculate the set of permutations, $\Pi_{N_{N_k}}$, of length N_{M_k} with elements $\{1, 2, \ldots, N_{N_k}\}$ due to missing elements. However, I solve this problem by interchanging the metric's arguments. The interchange of arguments does not change the OSPA distance because of the symmetry axiom.

The base distances requires the labels of estimated tracks and ground-truth tracks. Thus, I assign the labels of ground-truth tracks to the estimated tracks.

4.9.2 Optimal Label Assignment

To compute the aforementioned base distance (and the labeling error), I assign the labels of the ground-truth tracks to the estimated tracks in a globally optimum manner. As mentioned in [95], there are several approaches to assign labels to tracks. However, I decided to use the one mentioned in [83] and covered below: By minimizing the global OSPA distance between pairs of tracks over time, I assign labels of ground-truth tracks, $\{X^{(1)}, \ldots, X^{(N_{M_k})}\}$, to estimated tracks, $\{Z^{(1)}, \ldots, Z^{(N_{N_k})}\}$, with $X_k^{(l)}$ as a track at time index k and where N_{M_k} and N_{N_k} are the total numbers of ground-truth tracks and estimated tracks, respectively. As in case of the OSPA distance, I need to distinguish between two cases: the case where $N_{M_k} \leq N_{N_k}$ and the case where $N_{M_k} > N_{N_k}$.

The goal is to determine an optimal global assignment b^*, which optimally assigns the ground-truth tracks to a certain set of estimated tracks. I determine this assignment by minimizing the global OSPA distance between each discrete-time support point's pairs of tracks.

For $N_{M_k} \leq N_{N_k}$ and a set of permutations $\Pi_{N_{N_k}}$, where $b \in \Pi_{N_{N_k}}$, and where each permutation b consists of N_{M_k} elements taken from $\{1, 2, \ldots N_{N_k}\}$, the assignment is as follows:

$$b^* = \arg \min_{b \in \Pi_{N_{N_k}}} \sum_{l=1}^{N_{M_k}} \sum_{k=1}^{K} \Big(\mathbf{1}_k^l \mathbf{1}_k^{b(l)} \min \Big(a, \|\mathbf{x}_k^l - \mathbf{z}_k^{b(l)}\|_2 \Big) + (1 - \mathbf{1}_k^l)\mathbf{1}_k^{b(l)} a + \mathbf{1}_k^l (1 - \mathbf{1}_k^{b(l)}) a \Big). \tag{4.108}$$

Variable $\mathbf{1}_k^l$ indicates whether a ground-truth track exists, whereas $\mathbf{1}_k^{b(l)}$ indicates that a b-assigned estimated track exists. If one track exists and another one does not at index k, the label assignment imposes a penalty, a. The higher a the more the label assignment favors the assignment of longer-duration estimated tracks to ground-truth tracks. If b^* indicates that an estimated track is assigned to a ground-truth track with label l_k, then I need to set the estimated track's label to l_k, too. Each unassigned estimated track receives a label different from all ground-truth tracks' labels.

In case of $N_{M_k} > N_{N_k}$, I interchange the ground-truth tracks with the estimated tracks, as done in case of the OSPA distance, due to the computation of permutations with length N_{M_k}.

To evaluate a multiple-target tracking algorithm's performance, it is sufficient to compute the OSPA distance. However, to obtain more insight into certain types of errors, I compute some OSPA components separately.

Implementing (4.108) is unfeasible for scenarios with a high number of tracks; it takes too much time to compute (4.108) for all permutations $b \in \Pi_{N_{N_k}}$. Instead of implementing the aforementioned equation, I employed the Munkres method [113] (known as Kuhn-Munkres algorithm or Hungarian algorithm) to speed up optimally assigning labels to the estimated tracks. It is a combinatorial optimization algorithm that solves the label assignment in polynomial time.

4.9.3 Components of Optimal Subpattern Assignment

The OSPA distance consists of two major components: the cardinality error component (4.101) and the base distance error component (4.102) [82, 111]. They account for localization errors and cardinality errors, respectively.

For $N_{M_k} \leq N_{N_k}$, I compute the base distance error component and the cardinality error component separately according to

$$\mathcal{Q}_{\eta_o,a}(\mathfrak{X}_k, \mathfrak{Z}_k) = \left[\frac{1}{N_{N_k}} \min_{b \in \Pi_{N_{N_k}}} \sum_{i=1}^{N_{M_k}} \mathcal{A}_{\eta_o,a}(\bar{\mathbf{x}}_{k,i}, \bar{\mathbf{z}}_{k,b(i)}) \right]^{\frac{1}{\eta_o}}, \tag{4.109}$$

and

$$\mathcal{R}_{\eta_o,a}(N_{N_k}, N_{M_k}) = \left(\frac{(N_{N_k} - N_{M_k}) a^{\eta_o}}{N_{N_k}} \right)^{\frac{1}{\eta_o}} \tag{4.110}$$

to obtain valuable additional information, though separation is not necessary when calculating the OSPA distance.

In case of $N_{M_k} > N_{N_k}$, $\mathcal{Q}_{\eta_o,a}(\mathfrak{X}_k, \mathfrak{Z}_k) \triangleq \mathcal{Q}_{\eta_o,a}(\mathfrak{Z}_k, \mathfrak{X}_k)$ which is necessary due to the computation of permutations of length N_{M_k}. Moreover, $\mathcal{R}_{\eta_o,a}(N_{N_k}, N_{M_k}) \triangleq \mathcal{R}_{\eta_o,a}(N_{M_k}, N_{N_k})$, because in [111] they assume $\mathcal{R}_{\eta_o,a}$ as a metric on the space of nonnegative integers. Beyond that, $\mathcal{R}_{\eta_o,a}(N_{N_k}, N_{M_k})$ would yield a complex value which violates $\mathcal{D} : \mathcal{X}_k \times \mathcal{X}_k \to \mathbb{N}_+ = [0, \infty)$.

Example 3. Cardinality Error Component The cardinality error is a nonlinear measure. It is difficult to interpret this error without knowing the value of the cutoff parameter and the OSPA order.

For instance, a cardinality error with value three does not necessarily mean that there are N_N estimated tracks and $N_M = N_N - 3$ ground-truth tracks. Given the OSPA order, $\eta_o = 1$, the cutoff parameter, $a = 4$, the number of estimated tracks, $N_N = 4$, and the number of ground-truth tracks, $N_M = 1$, then $\mathcal{R}_{\eta_o,a}(N_{N_k}, N_{M_k}) = 3$. However, if $N_N = 3$, then $\mathcal{R}_{\eta_o,a}(N_{N_k}, N_{M_k}) = 8/3 \neq 2$. As this example shows, $\mathcal{R}_{\eta_o,a}(N_{N_k}, N_{M_k}) \in \mathbb{R}$. ■

4.9.4 Averaged Metrics

For my evaluations, I introduced two versions of the averaged OSPA distance, the averaged cardinality error, and the averaged localization error. The first version yields the average over all Monte Carlo experiments for a single frame index. The second version computes the mean value of the first one.

The OSPA distance averaged over (all) N_c Monte Carlo experiments is

$$\overline{\mathcal{D}}_{\eta_o,a}(\mathfrak{X}_k, \mathfrak{Z}_k) = \frac{1}{N_c} \sum_{i=1}^{N_c} \mathcal{D}_{\eta_o,a,i}(\mathfrak{X}_k, \mathfrak{Z}_k) \tag{4.111}$$

where i is the Monte Carlo experiment's index. Similar to (4.111), the averaged localization error is as follows:

$$\overline{\mathcal{Q}}_{\eta_o,a}(\mathfrak{X}_k, \mathfrak{Z}_k) = \frac{1}{N_c} \sum_{i=1}^{N_c} \mathcal{Q}_{\eta_o,a,i}(\mathfrak{X}_k, \mathfrak{Z}_k). \tag{4.112}$$

To compute the averaged cardinality error, I consider

$$\overline{\mathcal{R}}_{\eta_o,a}(N_{N_k}, N_{M_k}) = \frac{1}{N_c} \sum_{i=1}^{N_c} \mathcal{R}_{\eta_o,a,i}(N_{N_k}, N_{M_k}). \tag{4.113}$$

In the experiments, I additionally computed the mean values of the OSPA distance,

$$\overline{\mathcal{D}}_{\eta_o,a} = \frac{1}{N_F} \sum_{k=1}^{N_F} \overline{\mathcal{D}}_{\eta_o,a}(\mathfrak{X}_k, \mathfrak{Z}_k), \tag{4.114}$$

the localization error,

$$\overline{\mathcal{Q}}_{\eta_o,a} = \frac{1}{N_F} \sum_{k=1}^{N_F} \overline{\mathcal{Q}}_{\eta_o,a}(\mathfrak{X}_k, \mathfrak{Z}_k), \tag{4.115}$$

and the cardinality error,

$$\overline{\mathcal{R}}_{\eta_o,a} = \frac{1}{N_F} \sum_{k=1}^{N_F} \overline{\mathcal{R}}_{\eta_o,a}(N_{N_k}, N_{M_k}). \tag{4.116}$$

4.10 Experimental Design

To show that I can feed multiple-target trackers with the estimates of the proposed joint estimators, I conducted experiments with the GM-PHD filter, the GM-CPHD filter, and the GM-CBMeMBer filter. Instead of considering both joint estimators, I focused on experiments with the RPDM-based algorithm, the VSS-based algorithm's successor, due to its invariant frequency intervals. These invariant intervals ensure an improved observation-based correction of estimated states at higher frequencies.

In total I conducted experiments in three different categories. In the first category, I employed signals of synthetically generated and spatialized linearly frequency-sweeping harmonic sources. In the second category, I used signals of synthetically spatialized sources based on close-talking speech recordings. In the last category, I conducted experiments with signals of speakers recorded in a reverberant environment.

In the first category, I considered linearly frequency-sweeping harmonic sources mixed with spatially filtered white Gaussian noise. Table 4.1 lists all necessary algorithmic parameters for the RPDM-based algorithm, and Table 4.2 covers all relevant parameters in order to generate and spatialize the sources. The azimuth domain spanned a grid of angles between $-90°$ and $+90°$, the size of the maxima detector's search window was (3×3) indices. The experiments' goal was to track the harmonic signal's f_0 as well as the second, third, and fourth harmonic.

In the second category's experiments, I utilized the same subset of the Austrian German speech corpus [51] as mentioned in the previous chapters. Considering the parameters mentioned in Table 4.2, I synthetically spatialized the close-talking speech recordings. I used the f_0s provided by the Austrian German speech corpus. Table 4.1, lists all necessary algorithmic parameters for the RPDM-based algorithm. The goal of the experiments with speech recordings was to successfully track the speaker's f_0. I did not synthetically add any reverberation or interferences to keep the experiments feasible without algorithmically manipulating any multiple-target tracker. The main purpose of these experiments is to present a working cascade of joint estimator and multiple-target tracker for the very first time.

In the third category, I conducted experiments with real speech signals recorded in a meeting room to localize, characterize, and track a male speaker. Details on that room can be found in Chapter 5. As shown in Fig. 4.1 and Fig. 4.2, I sampled the meeting room's acoustic wave field by using two microphone arrays: one above the window (labeled as `A1` in Fig. 4.2) and one at the whiteboard (labeled as `A2` in Fig. 4.2). For the upcoming evaluations, I used the outermost microphones of `A2` only. (Chapter 5 includes information on the arrays' geometries.) A male speaker uttered vowels and the sentence "Why were you away a year, Roy?" at six different positions on two arcs. The positions' labels at the window's arc are `L1`, `M1`, and `R1`; those at the whiteboard's arc are `L2`, `M2`, and `R2`. The arcs' radii are 2.42 m measured from the arrays' central microphone. To extract the speaker's f_0s of each utterance, I utilized a laryngograph. The laryngograph returns glottograms used to extract those frequencies. (Chapter 5 contains detailed information on how to record and process glottograms in order to extract f_0s.) I considered the same procedures described in Chapter 5 to prepare the speaker, to calibrate the hardware, and to post-process the recordings. Table 4.1 lists the algorithmic parameters for the RPDM-based algorithm. To highlight a certain phenomenon unknown in the field of joint estimation, I employed a set of recordings that slightly differed from the recordings of the Austrian German speech corpus [50]. I required recordings of a speaker, which looked exactly toward the center of the microphone array and which uttered voiced sounds with an invariant pitch, i.e., vowels or the sentences "Why were you away a year, Roy?". However, the aforementioned speech corpus lacks such voiced utterances.

To compute the tracks, I applied the GM-PHD filter, the GM-CPHD filter, and the GM-CBMeMBer filter to the observations generated by the RPDM-based estimator. Table 4.3, Table 4.4, and Table 4.5 list the algorithmic parameters of each tracker used in these experiments. I additionally considered a uniform clutter distribution, a uniform birth distribution, and a normal distribution for updating and spawning the states. In case of the GM-CPHD filter, I utilized the Poisson distribution to model the cardinality mass function for bearing states and for clutter as well as the Binomial distribution for computing the cardinality's masses of surviving states.

In all experiments, the multiple-target trackers assumed linear Gaussian target dynamics according to

$$f_{k|k-1}(m_{k|k-1} \mid m_{k-1}) = \mathcal{N}\left(m_{k|k-1}; F_{k-1}m_{k-1}, Q_{k-1}\right) \tag{4.117}$$

and a linear Gaussian observation model,

$$g_k(z_k \mid m_{k|k-1}) = \mathcal{N}\left(z_k; H_k m_{k|k-1}, R_k\right), \tag{4.118}$$

Table 4.1: Algorithmic and environmental parameters of the RPDM-based estimator. The variables denote the speed of sound v, the sampling frequency f_s, the lowest and highest fundamental frequency of interest f_l and f_u, the desired amplitudes of the stop band and the pass band a_s and a_p, the desired ripples of the stop band and the pass band b_s and b_p, the number of extrema N_e, and the DOA-tolerance parameter ε. The first (left) value of f_u and N_e is used in experiments with simulated data, whereas the second value is used in experiments with spatially filtered speech recordings. The third value of N_e is used in experiments with real speech signals recorded in a meeting room.

v	f_s	f_l	f_u	a_s	a_p	b_s	b_p	N_e	ε
343.2 m/s	48 kHz	70 Hz	1610 Hz | 400 Hz	0	1	0.01	0.05	16 | 2 | 4	0.5

Table 4.2: Parameters of the synthetically spatialized, linearly frequency-sweeping sources. The variables denote the angular step size $\Delta\varphi$, the elevation angle ϑ, the number of microphones N_m, the array length d_a, the number of harmonics N_q, the sweep's start frequency and stop frequency f_1 and f_2, the sweep's duration T_2, the signal to noise ratio SNR, the temporal signal components' amplitude α, the normal distribution of noise with its parameters $\mathcal{N}(0,1)$, and the angular grid Φ.

$\Delta\varphi$	ϑ	N_m	d_a	N_q	f_1	f_2	T_2
1°	90°	8	0.5 m	4	80 Hz	400 Hz	2 s

SNR	α	ν	Φ
$\{-10, 0, 10, 20, 30\}$ dB	$0.4\sqrt{10^{\frac{\text{SNR}}{10}}}$	$\mathcal{N}(0,1)$	$\{-65°, \ldots, +65°\}$

with state transition matrix and process noise covariance

$$F_k = \begin{bmatrix} I_2 & I_2 \\ 0_2 & I_2 \end{bmatrix}, \quad Q_k = \sigma_Q^2 \begin{bmatrix} I_2/4 & I_2/2 \\ I_2/2 & I_2 \end{bmatrix}, \tag{4.119}$$

respectively, where I_n is the $(n \times n)$ identity matrix and 0_n is the $(n \times n)$ zero matrix.

In order to compute the OSPA distance as well as its components, the cardinality error component and the base distance error component, I set the OSPA parameters in all experiments as described in Table 4.6.

4.11 Experimental Results

In this section, I present the results of experiments with multiple-target trackers applied to estimates computed by the RPDM-based algorithm. The results comprise visualizations of computed tracks and metrics, i.e., the OSPA distance, the cardinality error, and the localization error over frame indices. In the experiments, I employed signals of synthetically generated and spatialized linearly frequency-sweeping harmonic sources, signals of synthetically spatialized sources based on close-talking speech recordings, and signals of speakers recorded in a reverberant environment.

Table 4.3: Algorithmic parameters of the GM-PHD filter. The variables denote the surviving probability p_S, the detection probability p_D, the spawning probability p_β, the birth probability p_γ, the spawned target's covariance matrix P_γ, the born target's covariance matrix P_γ, the clutter rate λ_c, the standard deviation of observation noise σ_R, the standard deviation of process noise σ_Q, the threshold ξ_v for pruning states according to their velocity, the threshold ξ_m for merging states, the threshold ξ_p for pruning states according to their weights, and the threshold ξ_e for selecting states as final estimates. The left value of a pair of values refers to experiments with sweeping signals, whereas the right value refers to experiments with speech signals.

p_S	p_D	p_β	p_γ	P_β	P_γ	λ_c	σ_R	σ_Q	ξ_v	ξ_m	ξ_p	ξ_e
0.99 | 0.95	0.9	0.01	0.1 | 0.35	diag(25)	diag(25)	8 | 1	10	0.1	10	10	10^{-4} | 10^{-6}	10^{-3}

Table 4.4: Algorithmic parameters of the GM-CPHD filter. The variables denote the surviving probability p_S, the detection probability p_D, the birth probability p_γ, the born target's covariance matrix P_γ, the clutter rate λ_c, the standard deviation of observation noise σ_R, the standard deviation of process noise σ_Q, the threshold ξ_v for pruning states according to their velocity, the threshold ξ_m for merging states, the threshold ξ_p for pruning states according to their weights, the threshold ξ_e for selecting states as final estimates, the mean cardinality of born states $\mathbb{E}\{|X_\gamma|\}$, and the mean cardinality of clutter $\mathbb{E}\{|X_K|\}$. Both mean cardinalities are the mean values of Poisson distributions.

p_S	p_D	p_γ	P_γ	λ_c	σ_R	σ_Q	ξ_v	ξ_m	ξ_p	ξ_e	$\mathbb{E}\{\|X_\gamma\|\}$	$\mathbb{E}\{\|X_K\|\}$
0.99	0.85	0.10	diag(25)	1	10	0.1	10	10	10^{-4}	0.1	0.5	0.5

Table 4.5: Algorithmic parameters of the GM-CBMeMBer filter. The variables denote the surviving probability p_S, the detection probability p_D, the birth probability p_γ, the born target's covariance matrix P_γ, the clutter rate λ_c, the standard deviation of observation noise σ_R, the standard deviation of process noise σ_Q, the threshold ξ_v for pruning states according to their velocity, the threshold ξ_m for merging states, the threshold ξ_p for pruning states according to their weights, the threshold ξ_e for selecting states as final estimates, and the threshold ξ_t for pruning tracks according to their probabilities.

p_S	p_D	p_γ	P_γ	λ_c	σ_R	σ_Q	ξ_v	ξ_m	ξ_p	ξ_e	ξ_t
0.99	0.85	0.10	diag(25)	1	10	0.1	10	10	10^{-4}	0.01	0.01

Table 4.6: Parameters of the optimal subpattern assignment metric. The variables denote the cutoff parameter a, the labeling-error penalty c, the OSPA order η_o, and the base distance order η_b.

a	c	η_o	η_b
10	1	1	1

Table 4.7: Measured reverberation times T_{30} and corresponding standard deviations $\sigma_{T_{30}}$ in seconds for several frequencies f in the meeting room labeled with `CPR`. A detailed description of the methods used to measure the reverberation time can be found in [114].

f [Hz]	63	125	250	500	1000	2000	4000	8000	16000
T_{30} [s]	0.90	0.59	0.54	0.51	0.51	0.50	0.48	0.44	0.41
$\sigma_{T_{30}}$ [s]	± 0.11	± 0.03	± 0.03	± 0.01	± 0.01	± 0.01	± 0.02	± 0.01	± 0.01

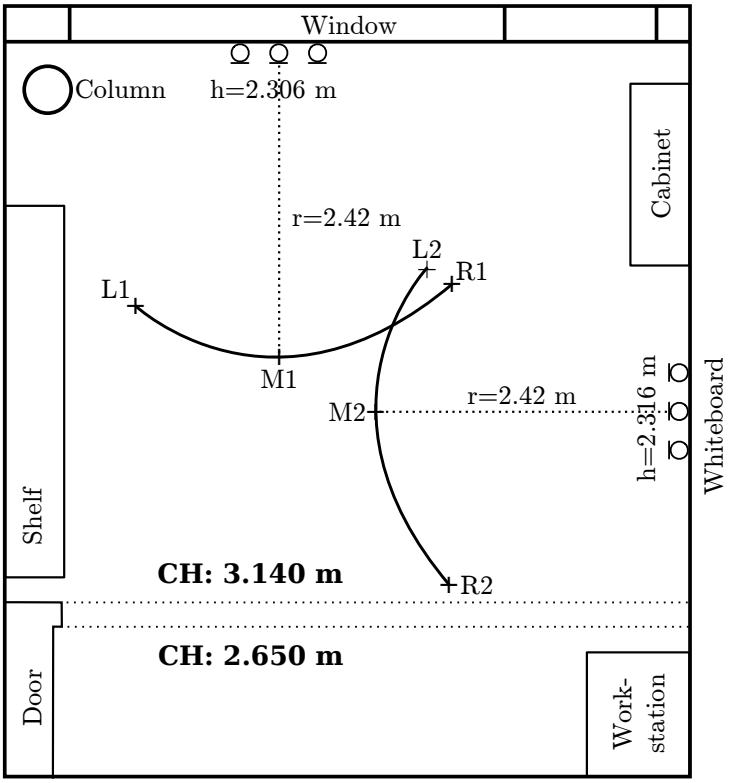


Fig. 4.1: Floor plan of the recording environment. The meeting room features six microphones. Both microphone arrays, the array above the window and the array above the whiteboard, consist of three uniformly spaced microphones. The label `CH` denotes the ceiling height. The labels `L1`, `M1`, and `R1` as well as `L2`, `M2`, and `R2` represent the speaker's positions on two different black arcs. The arcs guarantee a constant distance between the source and the central microphone of each array; thus, the direct paths' attenuation (the paths between the speaker and the array's central microphone) is the same at each position. The speaker always looked into the direction of an array's central microphone.

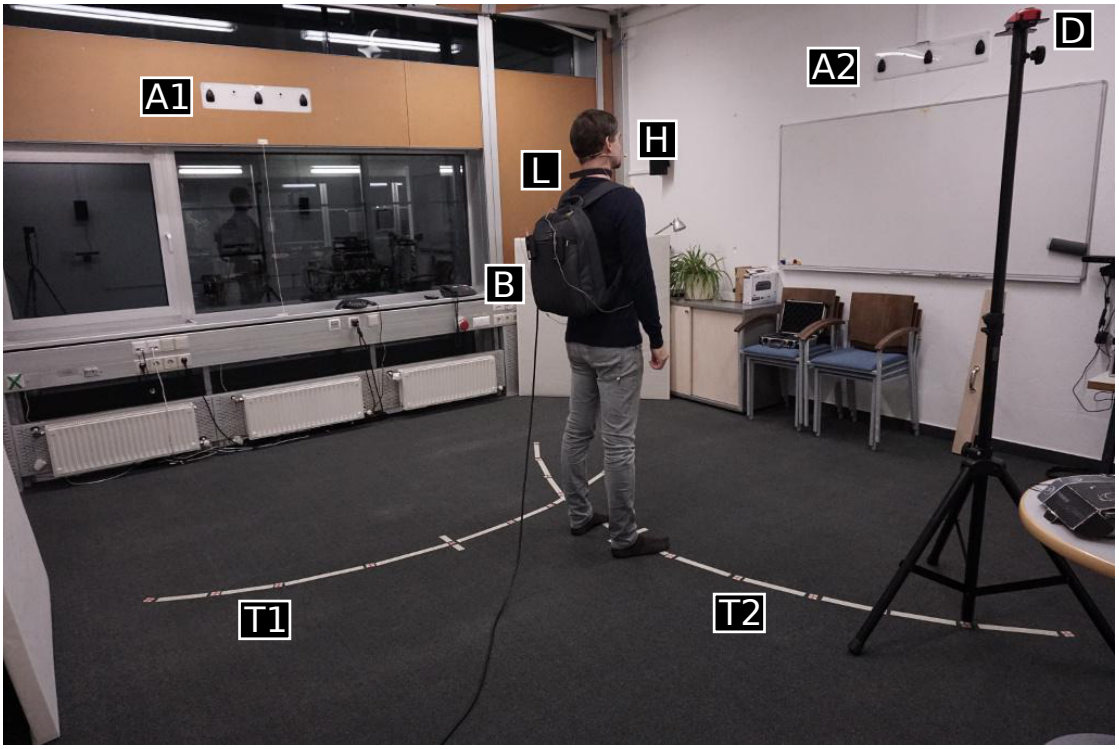


Fig. 4.2: A speaker utters vowels and sentences at different positions on an arc (`T2`) facing east in the meeting room. Two arrays (`A1` and `A2`) sample the room's acoustic wave field. There are marked trajectories (`T1` and `T2`) on the floor. The speaker wears a backpack (`B`) which contains a laryngograph and wireless transmitters. The speaker wears the laryngograph's sensors (`L`) on the neck and a head-mounted microphone (`H`). A laser distance meter (`D`) measures the distance between the array's central microphone and several points in the room to be able to draw an arc with a constant radius.

Table 4.8: Values representing the OSPA distance, the cardinality error, and the localization error averaged over all frames of all Monte Carlo experiments with synthetically generated sources.

	GM-PHD	GM-CPHD	GM-CBMeMBer
Averaged OSPA $\overline{\mathcal{D}}_{\eta_o,a}$	9.20	12.03	26.77
Averaged Localization Error $\overline{\mathcal{Q}}_{\eta_o,a}$	4.56	3.45	1.84
Averaged Cardinality Error $\overline{\mathcal{R}}_{\eta_o,a}$	1.62	2.69	6.40

4.11.1 Experiments with Synthesized Signals

Fig. 4.3, Fig. 4.5, and Fig. 4.7 show three-dimensionally visualized (a) RPDM-based frame-by-frame estimates and (b) original tracks marginalized over frequencies or frame indices (gray) and estimated tracks (black) of synthetically generated and spatialized linearly frequency-sweeping harmonic sources superimposed by spatially filtered noise yielding an SNR = 10 dB. Additionally, these figures contain estimated and original tracks marginalized over (c) frequencies or (d) angles as well as (e) their corresponding OSPA distances, (f) cardinality errors, and (g) localization errors of the GM-PHD filter (Fig. 4.3), the GM-CPHD filter (Fig. 4.5), and the GM-CBMeMBer filter (Fig. 4.7). Fig. 4.4, Fig. 4.6, and Fig. 4.8 present metrics averaged over all Monte Carlo experiments and for each multiple-target tracker: (a) the OSPA distance, (b) the OSPA distance of experiments with SNR = $\{-10, 0, 10, 20, 30\}$ dB, (c) the cardinality error and (d) the cardinality errors of experiments with the aforementioned SNRs, (e) the localization error considering all SNRs and (f) the localization errors of experiments with certain SNRs. I applied a moving average filter with weights $(0.25, 0.5, 0.25)$ to improve the SNR-dependent curves' readability. Table 4.8 lists values representing the OSPA distance, the cardinality error, and the localization error averaged over all Monte Carlo experiments and over all frames. Stated differently, these values represent the means of the plotted values in the left column of Fig. 4.4, Fig. 4.6, and Fig. 4.8.

4.11.2 Experiments with Synthetically Spatialized Real Speech Signals

Fig. 4.9, Fig. 4.11, and Fig. 4.13 show (a-b) observations, (c-d) original tracks and estimated tracks, and (e-h) marginalized tracks, while Fig. 4.10, Fig. 4.12, and Fig. 4.14 show (a-b) OSPA distances, (c-d) cardinality errors, (e-f) and localization errors. The plots in the left columns of Fig. 4.9–4.14 show the visualized results based on recordings of a female speaker, whereas the plots in the right columns feature results based on recordings of a male speaker. To emphasize the differences between the trackers' resulting trajectories, I selected a specific set of experiments. In case of the experiments based on the male speaker's recordings, I show the results of each tracker based on the same recording. In case of the experiments with the female speaker, I present results based on different recordings. However, in comparison to Fig. 4.3–4.8, I did not visualize the averaged metrics due to varying speech activity over time; the number of non-zero items for averaging would highly vary for each frame index. Table 4.9 lists values representing the localization error averaged over all Monte Carlo experiments and over all frames.

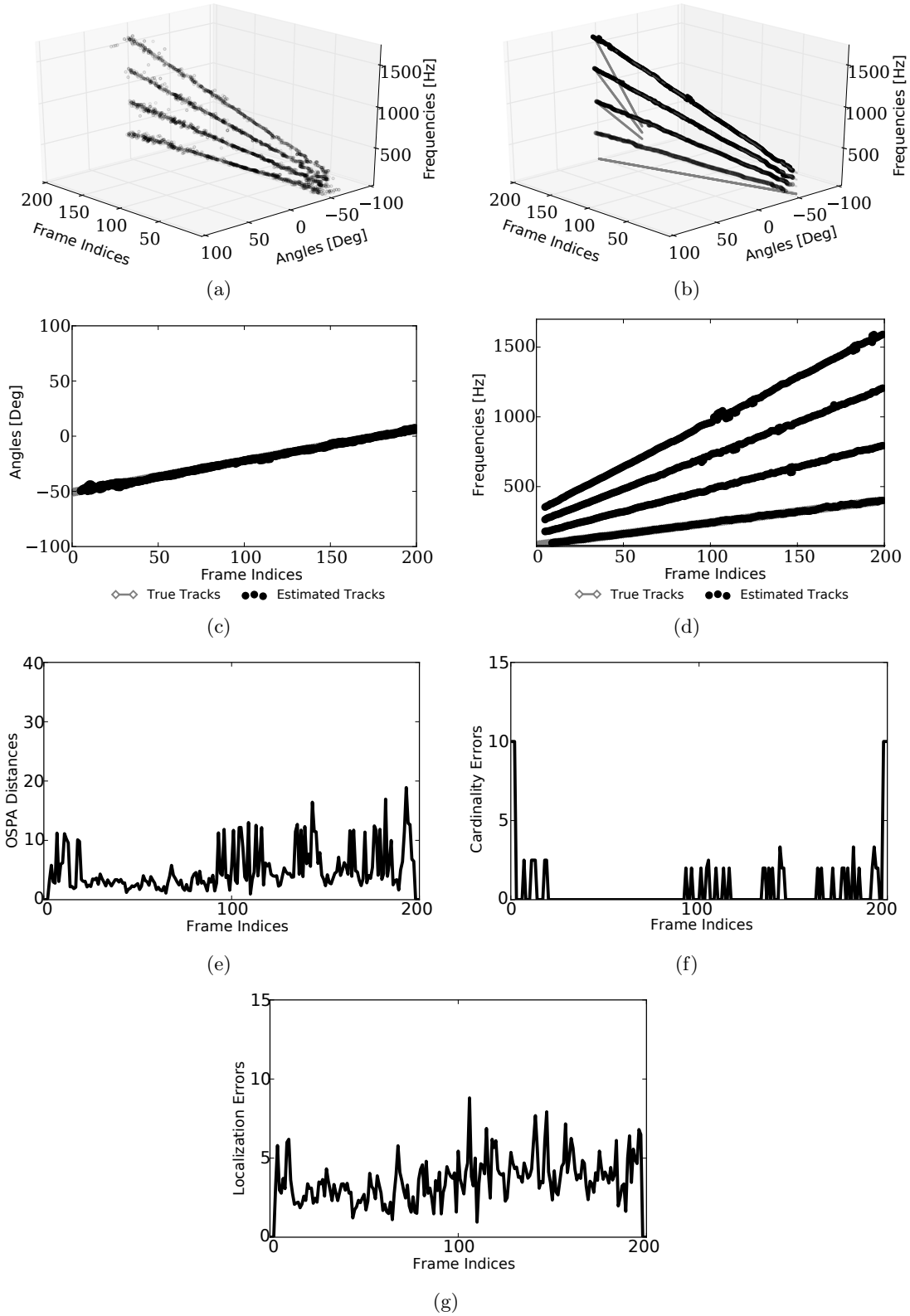


Fig. 4.3: (a) Frame-by-frame estimates, (b) original tracks marginalized over frequencies or frame indices (gray) and estimated tracks (black) of the f_0s, their second, third, and fourth harmonic's components (of a noisy signal with SNR = 10 dB) produced by the RPDM-based algorithm and the GM-PHD filter; (c) original tracks and estimated tracks of the DOAs in spatial domain after marginalizing over the frequency components, and (d) original tracks and estimated tracks in frequency domain after marginalizing over the spatial components. The plots in (e-g) show the OSPA distance, the cardinality error, and the localization error, respectively.

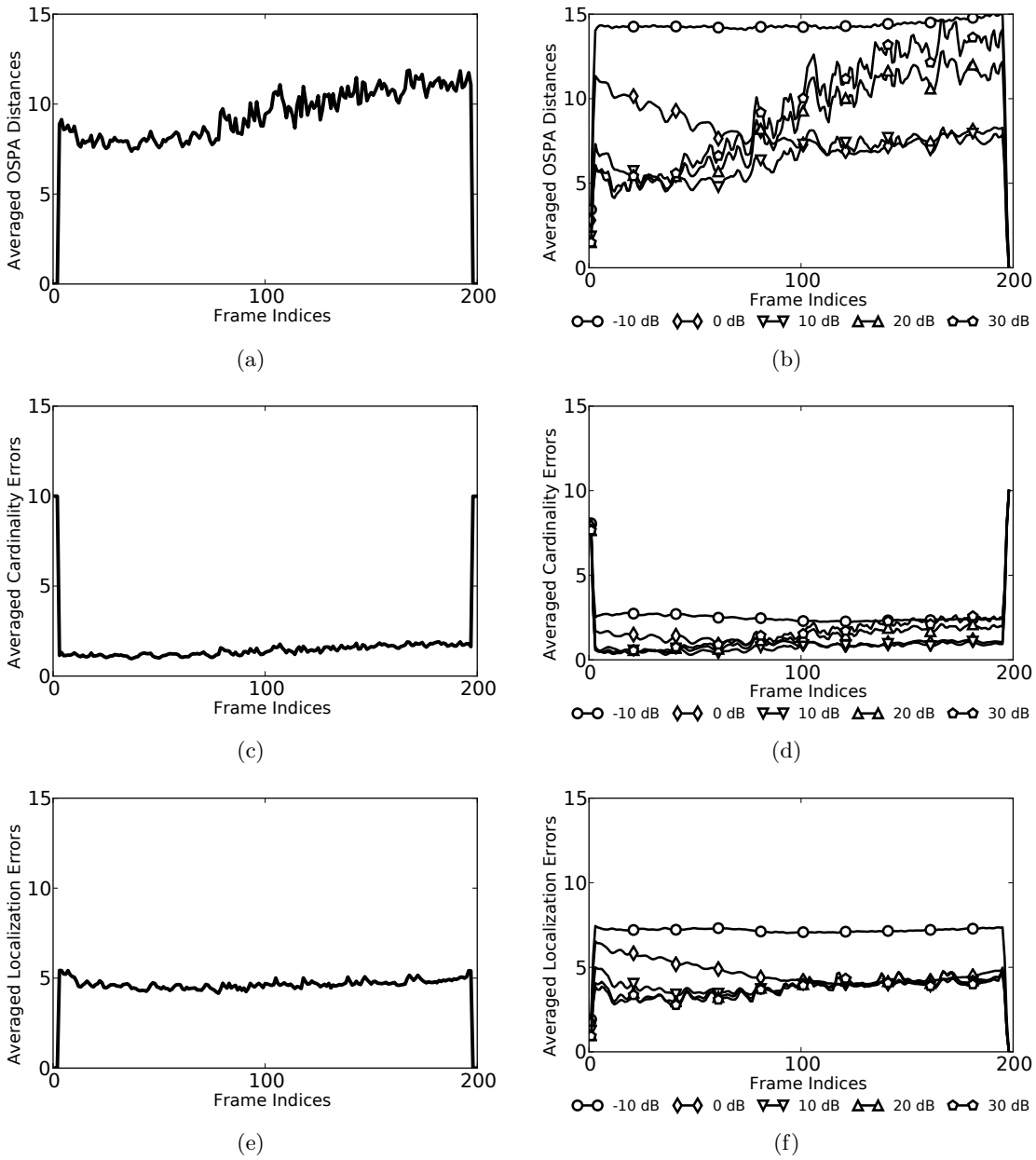


Fig. 4.4: (a) OSPA distances and (b) SNR-dependent OSPA distances averaged over all experiments, (c) cardinality errors and (d) SNR-dependent cardinality errors averaged over all experiments, (e) localization errors and (f) SNR-dependent localization errors averaged over all experiments of the GM-PHD filter's tracks. The RPDM-based algorithm estimated the f_0s, their second, third, and fourth harmonic's components of linearly frequency-sweeping harmonic sources superimposed by filtered noise. A weighted moving average filter with weights $(0.25, 0.5, 0.25)$ smoothed the curves of the SNR-dependent metrics.

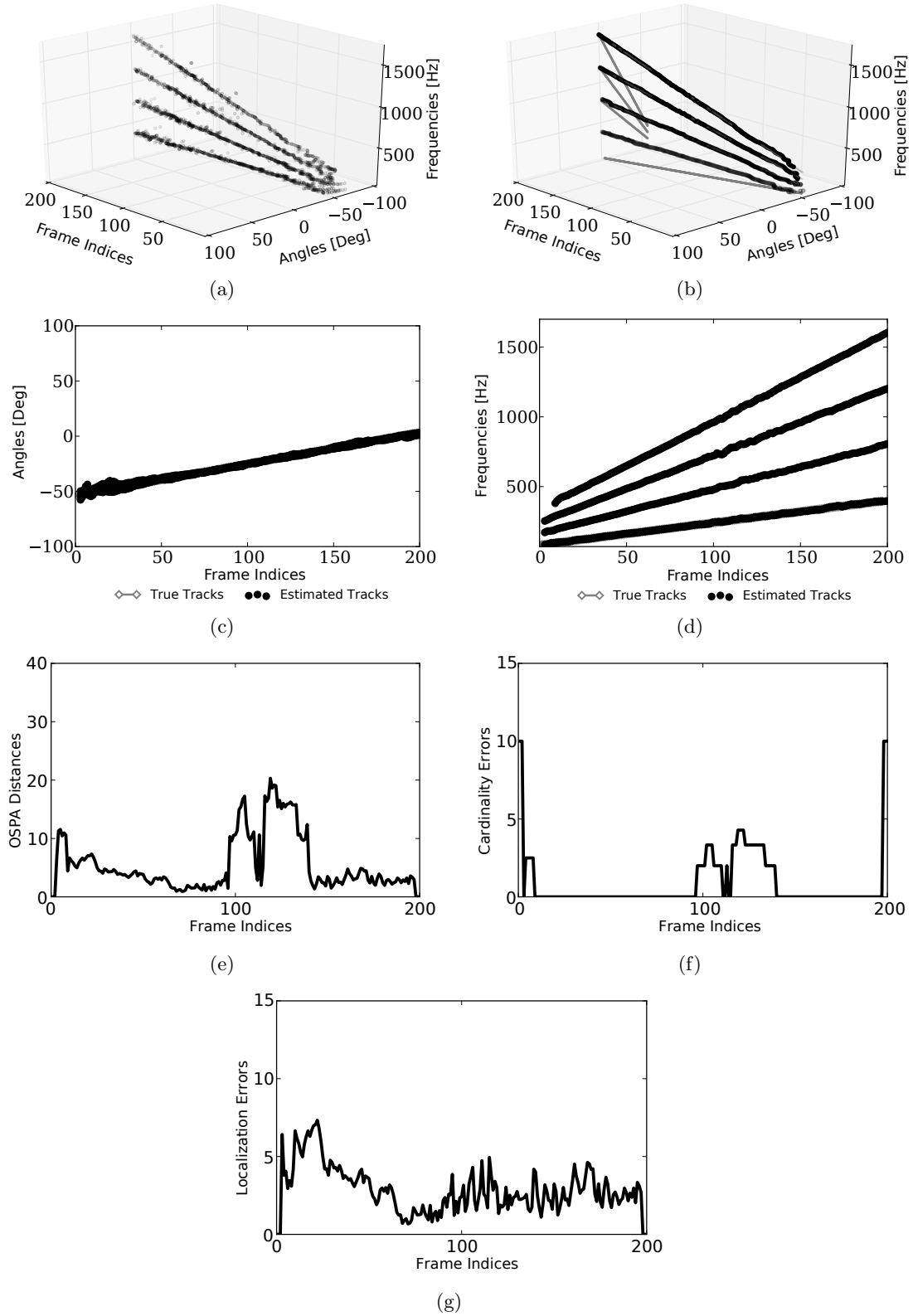


Fig. 4.5: (a) Frame-by-frame estimates, (b) original tracks marginalized over frequencies or frame indices (gray) and estimated tracks (black) of the f_0s, their second, third, and fourth harmonic's components (of a noisy signal with SNR = 10 dB) produced by the RPDM-based algorithm and the GM-CPHD filter; (c) original tracks and estimated tracks of the DOAs in spatial domain after marginalizing over the frequency components, and (d) original tracks and estimated tracks in frequency domain after marginalizing over the spatial components. The plots in (e-g) show the OSPA distance, the cardinality error, and the localization error, respectively.

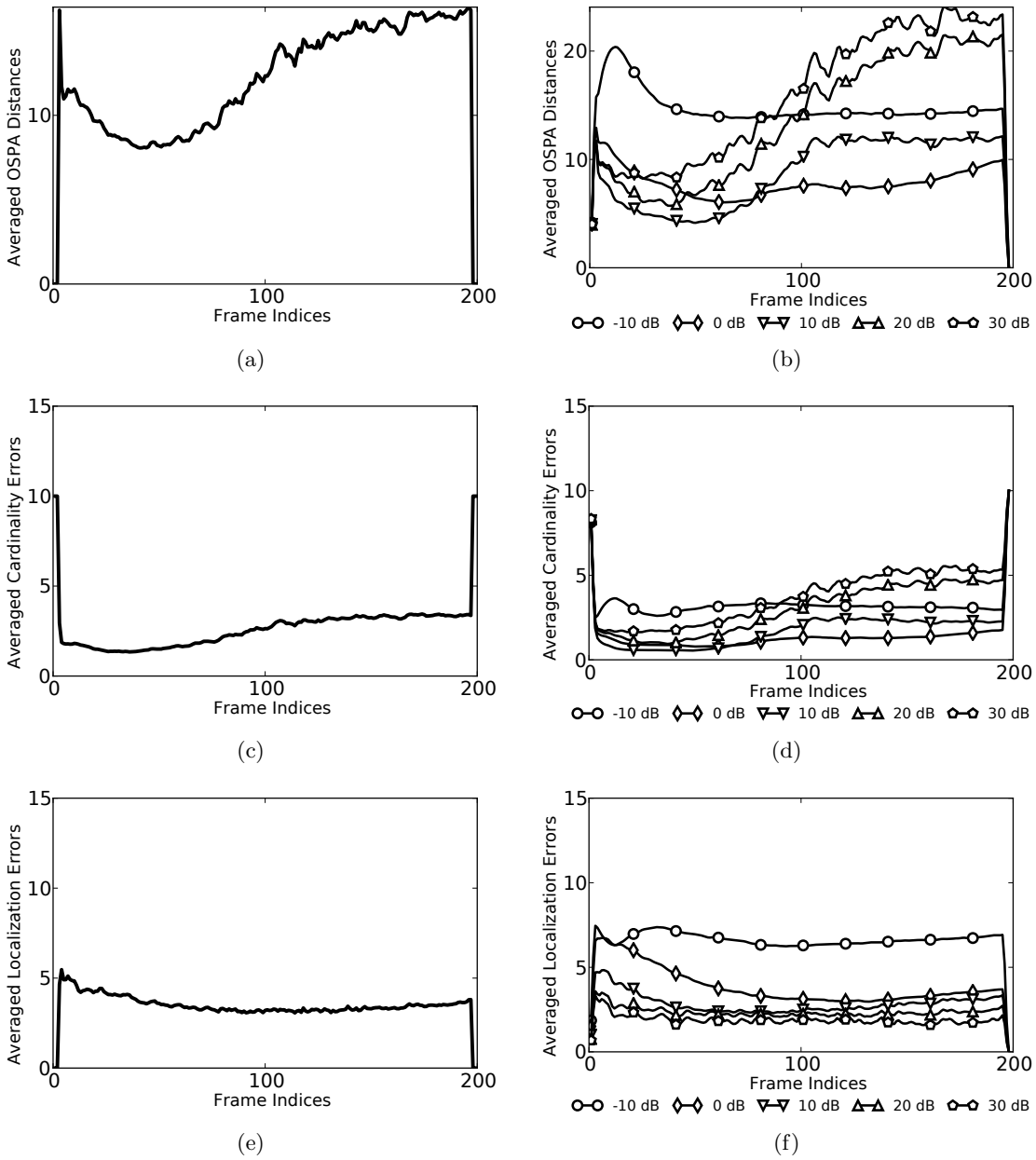


Fig. 4.6: (a) OSPA distances and (b) SNR-dependent OSPA distances averaged over all experiments, (c) cardinality errors and (d) SNR-dependent cardinality errors averaged over all experiments, (e) localization errors and (f) SNR-dependent localization errors averaged over all experiments of the GM-CPHD filter's tracks. The RPDM-based algorithm estimated the f_0s, their second, third, and fourth harmonic's components of linearly frequency-sweeping harmonic sources superimposed by filtered noise. A weighted moving average filter with weights $(0.25, 0.5, 0.25)$ smoothed the curves of the SNR-dependent metrics.

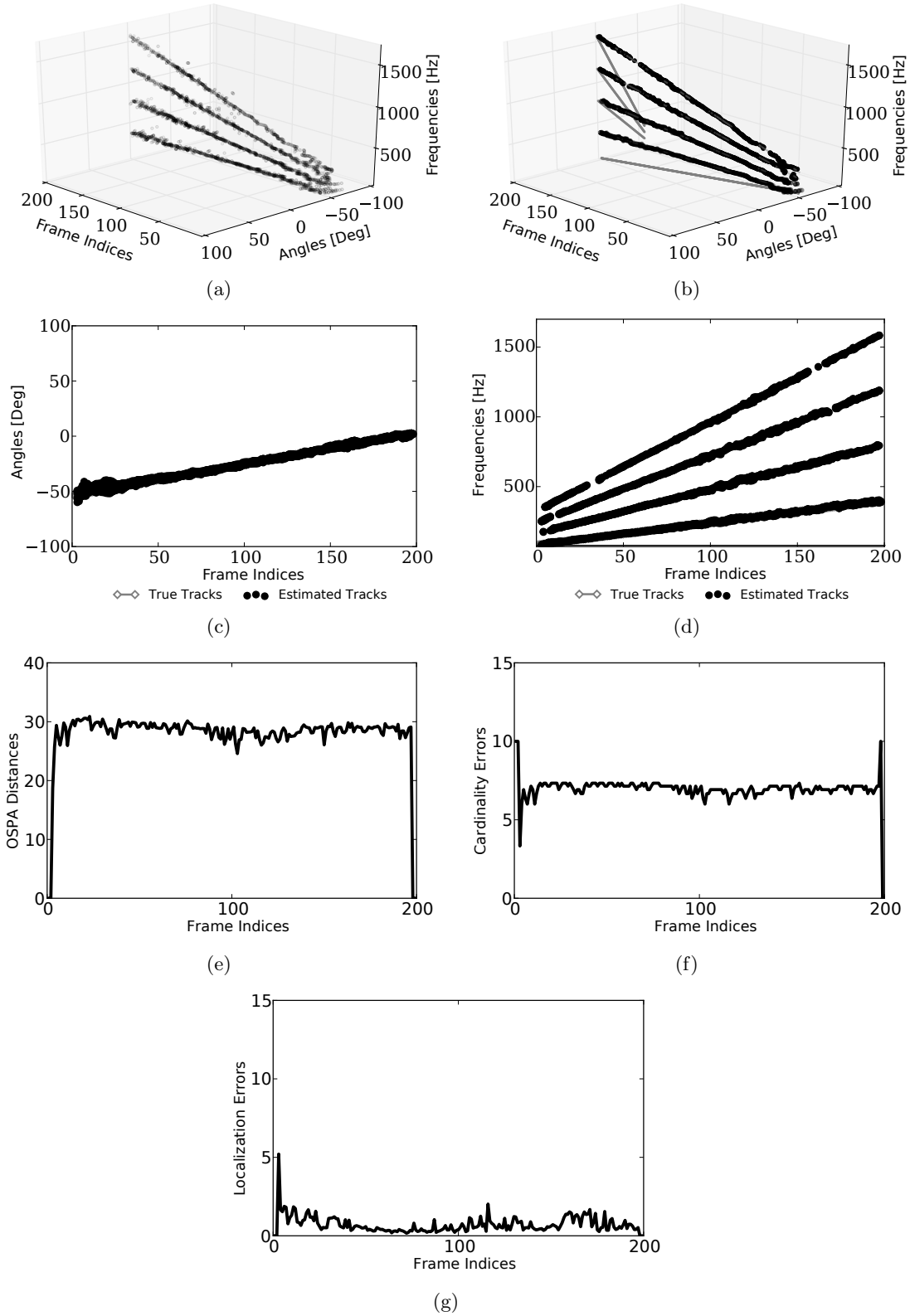


Fig. 4.7: (a) Frame-by-frame estimates, (b) original tracks marginalized over frequencies or frame indices (gray) and estimated tracks (black) of the f_0s, their second, third, and fourth harmonic's components (of a noisy signal with SNR = 10 dB) produced by the RPDM-based algorithm and the GM-CBMeMBer filter; (c) original tracks and estimated tracks of the DOAs in spatial domain after marginalizing over the frequency components, and (d) original tracks and estimated tracks in frequency domain after marginalizing over the spatial components. The plots in (e-g) show the OSPA distance, the cardinality error, and the localization error, respectively.

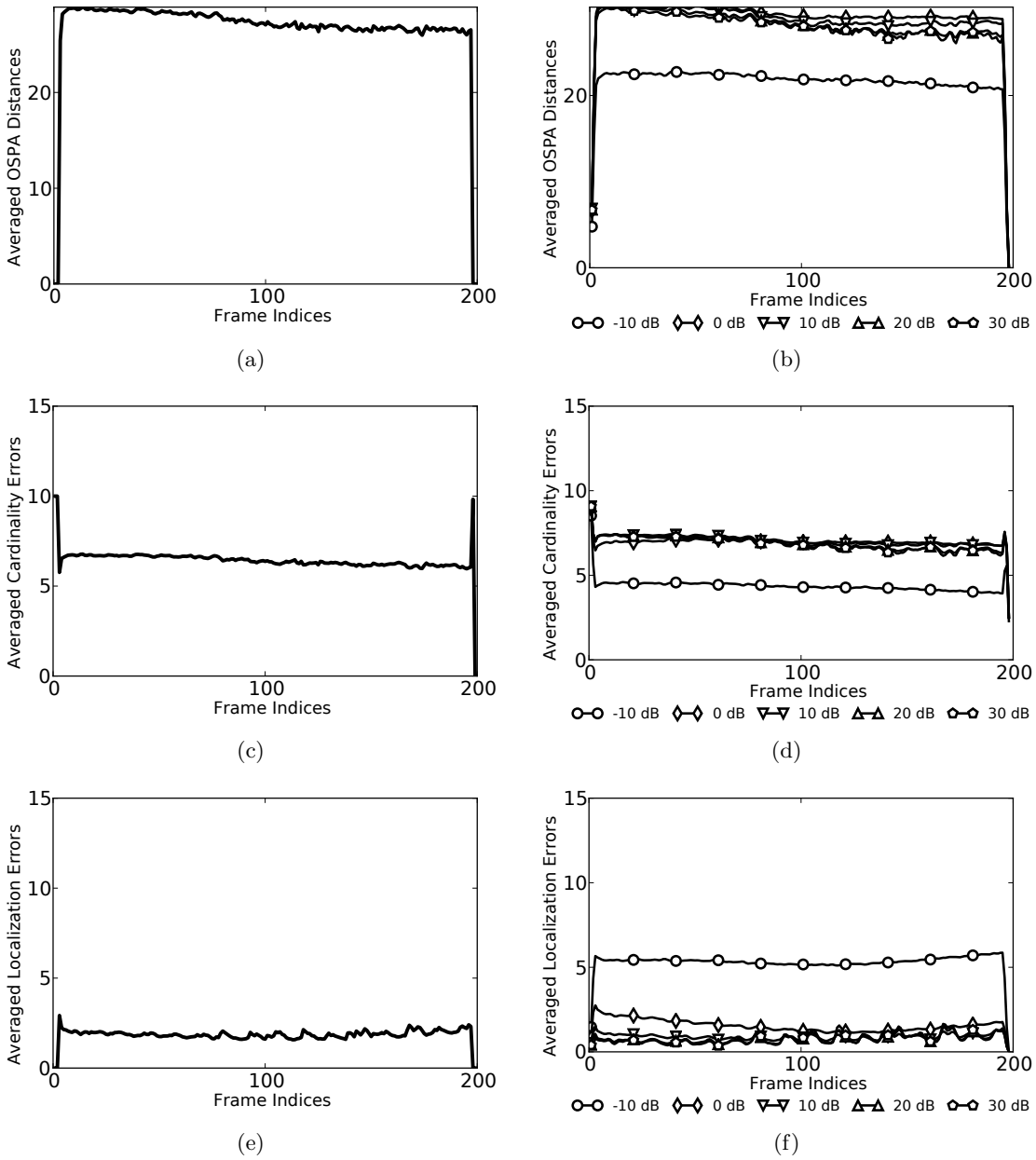


Fig. 4.8: (a) OSPA distances and (b) SNR-dependent OSPA distances averaged over all experiments, (c) cardinality errors and (d) SNR-dependent cardinality errors averaged over all experiments, (e) localization errors and (f) SNR-dependent localization errors averaged over all experiments of the GM-CBMeMBer filter's tracks. The RPDM-based algorithm estimated the f_0s, their second, third, and fourth harmonic's components of linearly frequency-sweeping harmonic sources superimposed by filtered noise. A weighted moving average filter with weights $(0.25, 0.5, 0.25)$ smoothed the curves of the SNR-dependent metrics.

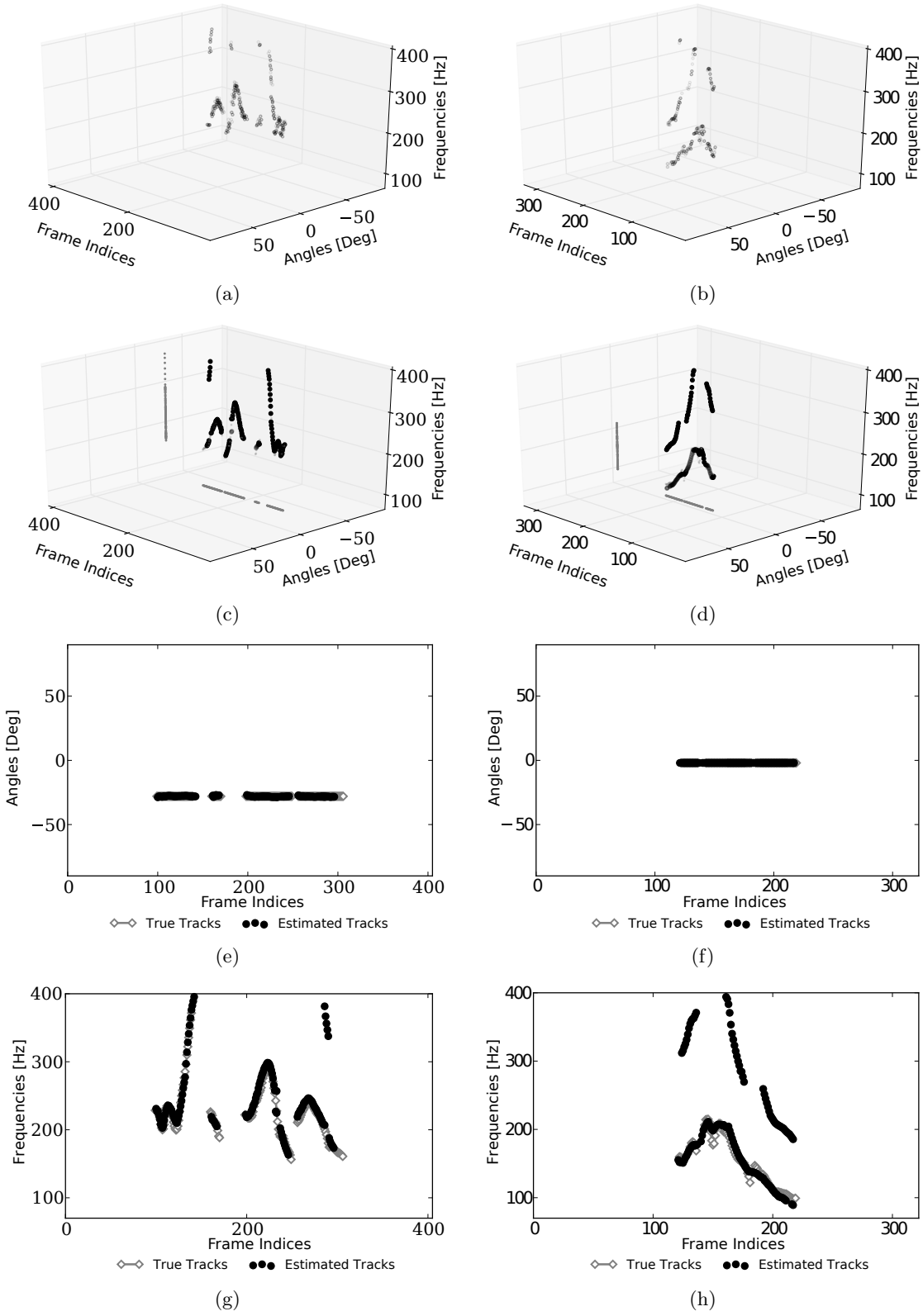


Fig. 4.9: (a,b) Frame-by-frame estimates and (c,d) original tracks marginalized over frequencies or frame indices (gray) and estimated tracks (black) of the fundamental frequencies and the second harmonic's components produced by the RPDM-based algorithm and the GM-PHD filter, respectively, (e,f) original tracks and estimated tracks of the DOAs in spatial domain after marginalizing over the frequency components, and (g,h) original tracks and estimated tracks in frequency domain after marginalizing over the spatial components of spatially filtered signals of a female speaker (left column) and a male speaker (right column).

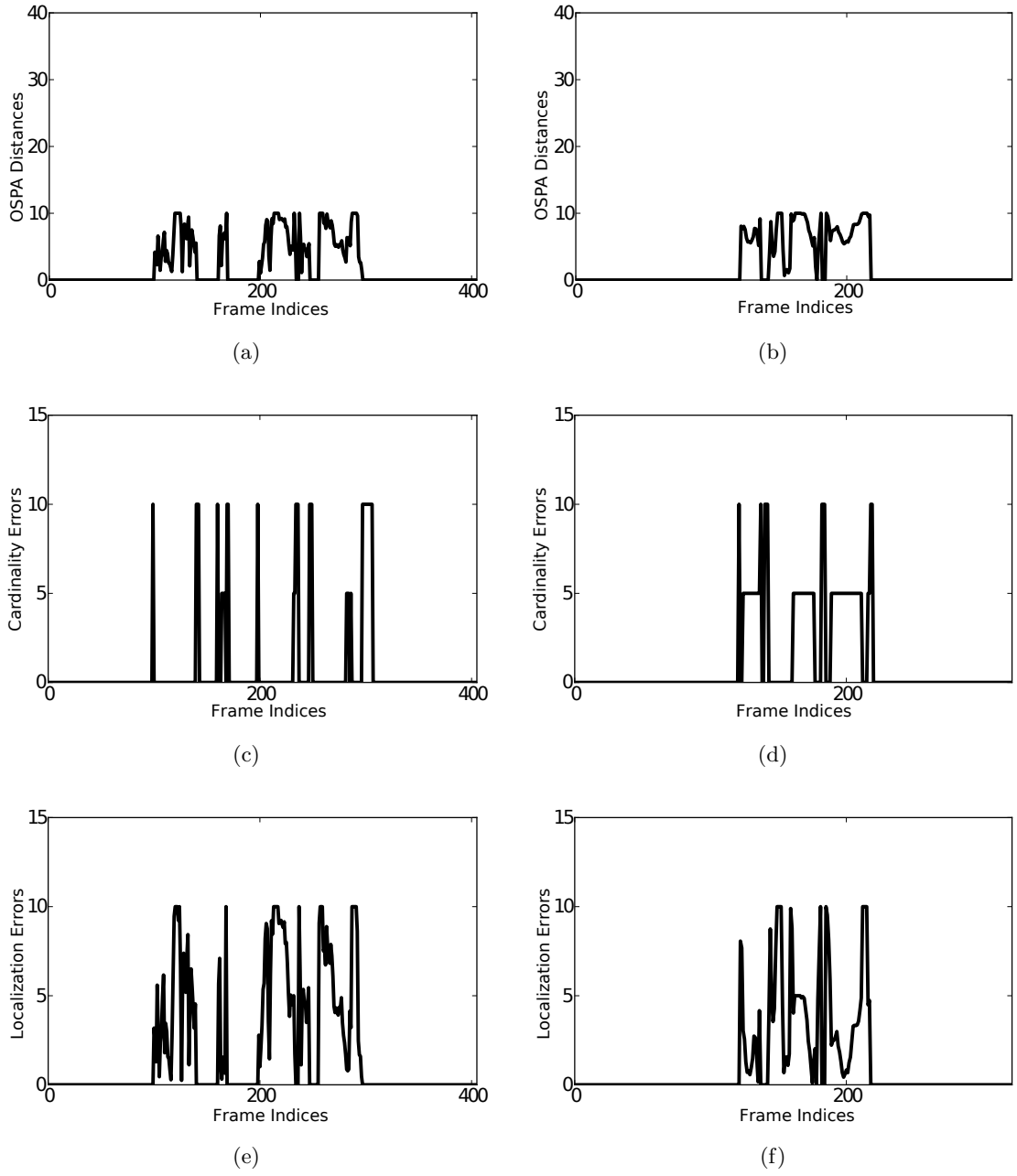


Fig. 4.10: (a,b) OSPA distances, (c,d) cardinality errors, and (e,f) localization errors of GM-PHD filtered tracks based on spatially filtered speech recordings of a female speaker (left column) and a male speaker (right column).

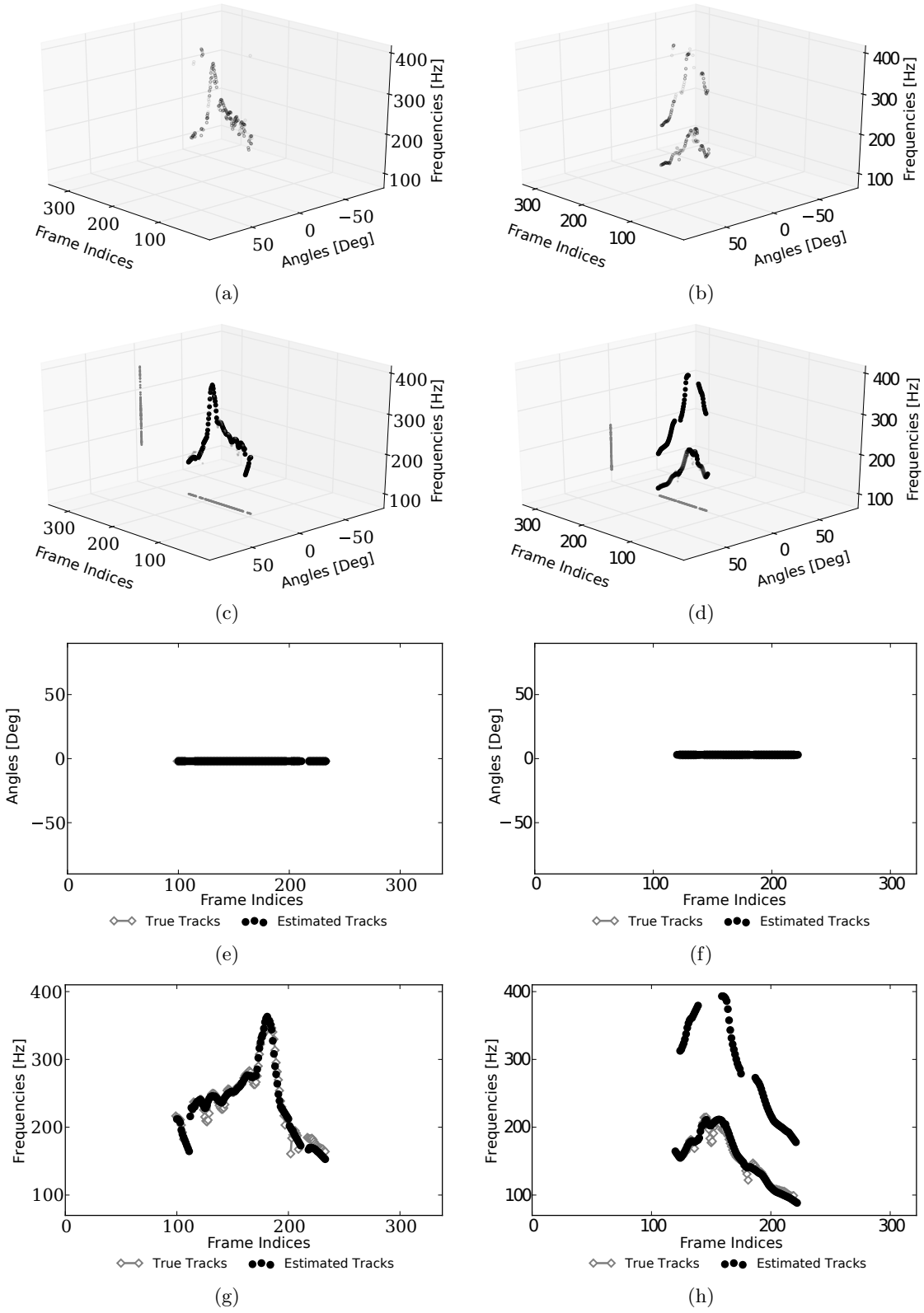


Fig. 4.11: (a,b) Frame-by-frame estimates and (c,d) original tracks marginalized over frequencies or frame indices (gray) and estimated tracks (black) of the fundamental frequencies and the second harmonic's components produced by the RPDM-based algorithm and the GM-CPHD filter, respectively, (e,f) original tracks and estimated tracks of the DOAs in spatial domain after marginalizing over the frequency components, and (g,h) original tracks and estimated tracks in frequency domain after marginalizing over the spatial components of spatially filtered signals of a female speaker (left column) and a male speaker (right column).

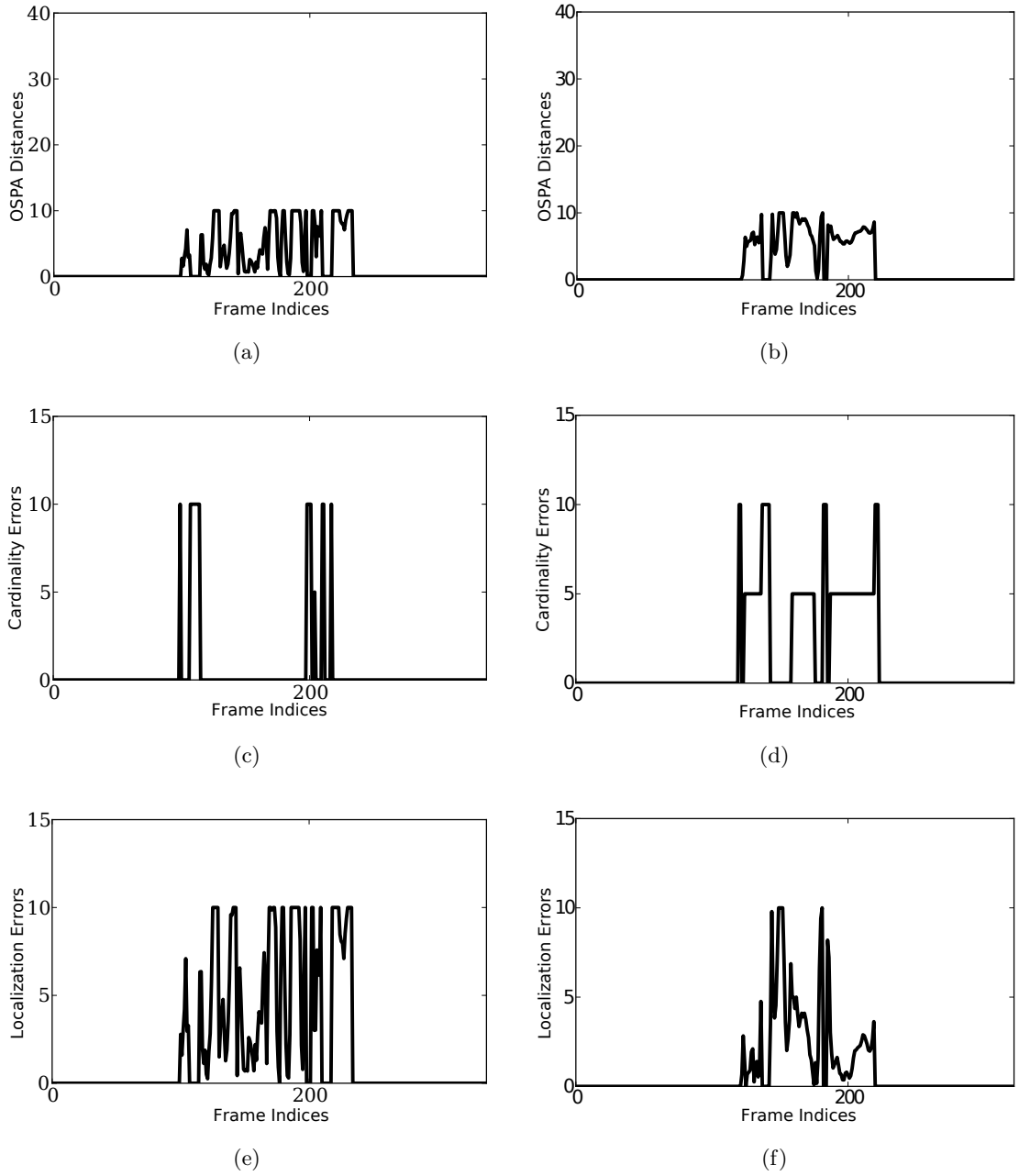


Fig. 4.12: (a,b) OSPA distances, (c,d) cardinality errors, and (e,f) localization errors of GM-CPHD filtered tracks based on spatially filtered speech recordings of a female speaker (left column) and a male speaker (right column).

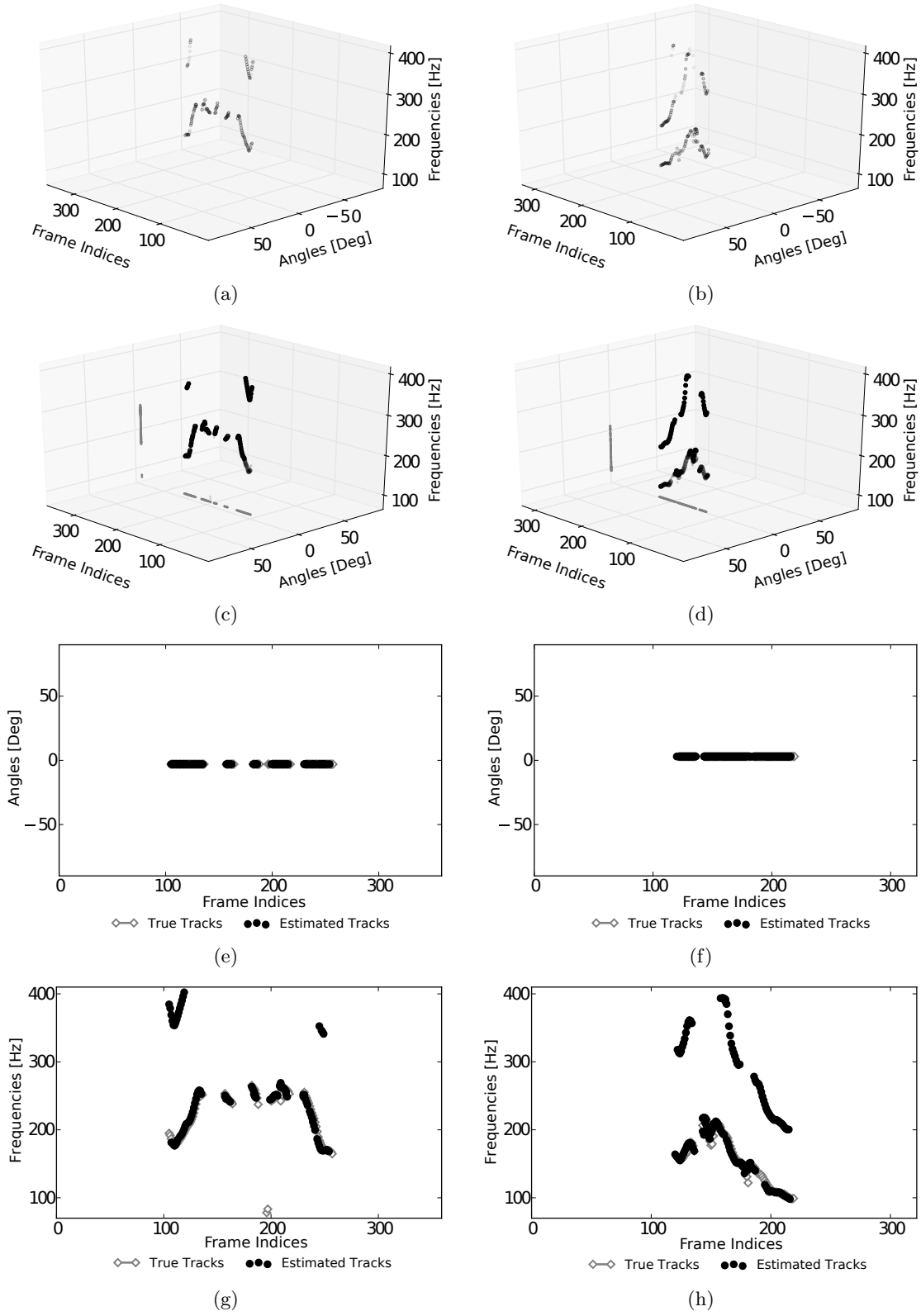


Fig. 4.13: (a,b) Frame-by-frame estimates and (c,d) original tracks marginalized over frequencies or frame indices (gray) and estimated tracks (black) of the fundamental frequencies and the second harmonic's components produced by the RPDM-based algorithm and the GM-CBMeMBer filter, respectively, (e,f) original tracks and estimated tracks of the DOAs in spatial domain after marginalizing over the frequency components, and (g,h) original tracks and estimated tracks in frequency domain after marginalizing over the spatial components of spatially filtered signals of a female speaker (left column) and a male speaker (right column).

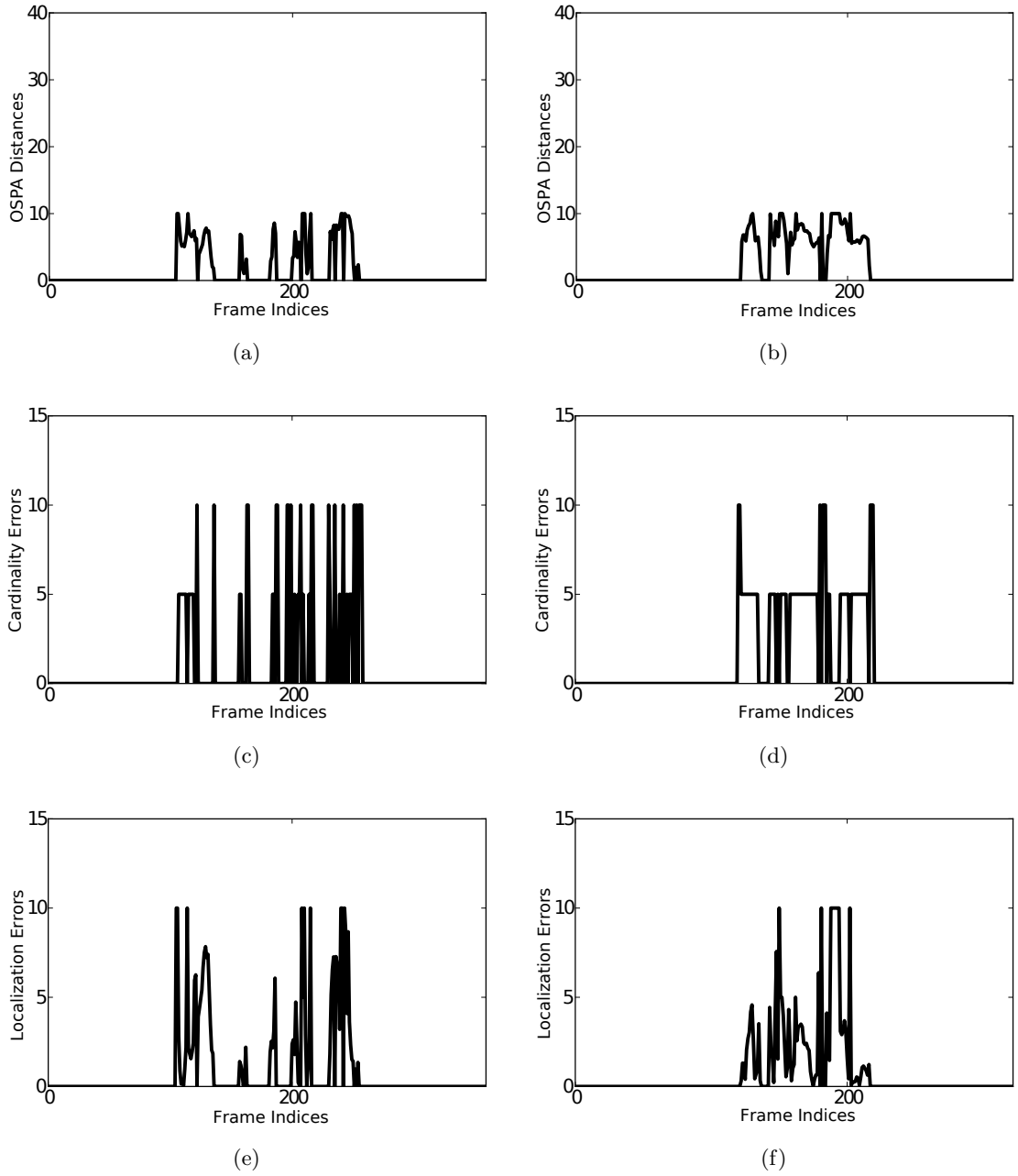


Fig. 4.14: (a,b) OSPA distances, (c,d) cardinality errors, and (e,f) localization errors of GM-CBMeMBer filtered tracks based on spatially filtered speech recordings of a female speaker (left column) and a male speaker (right column).

Table 4.9: Values representing the localization error averaged over over all frames of all Monte Carlo experiments with real speech recordings from a female (left value) and male (right value) speaker.

	GM-PHD	GM-CPHD	GM-CBMeMBer
Averaged Localization Error $\overline{\mathcal{Q}}_{\eta_o,a}$	9.53 / 11.91	11.02 / 12.60	4.93 / 6.80

4.11.3 Experiments with Real Reverberant Speech Recordings

In case of experiments with real reverberant speech signals recorded in a meeting room, I present visualizations of estimates and tracks shown in Fig. 4.15–4.17. These figures illustrate (a-b) the observations and (c-d) the corresponding trajectories in a three-dimensional space; additionally, they include plots marginalized over (e-f) frequencies or (g-h) angles. The left column of each figure represents the estimates and tracks of a male speaker uttering a vowel. The right column shows estimates and tracks of a male speaker uttering the sentence "Why were you away a year, Roy?" with almost constant pitch. I decided to avoid varying the pitch to demonstrate a fascinating phenomenon in reverberant environments when using frame-based joint estimators. The outermost microphones of the whiteboard's microphone array (`A2`) provided the corresponding signals. The speaker stood at position `M2`. I excluded visualizations of (averaged) metrics due to phenomena described in the corresponding discussion.

4.12 Discussion

A thorough evaluation of the results yielded an overview of each multiple-target tracker's accuracy and new findings on the RPDM-based algorithm's behavior in experiments described before. For instance, the estimator produces more clutter around the ground-truth f_0's and ground-truth harmonics at higher frequencies. One reason for that are the ripples around the bandpass filters' cutoff frequencies. Though using optimized Kaiser window-order estimated bandpass filters, the ripple around these frequencies increases towards higher frequencies. This might cause wrong estimates in a neighboring band's edge, if the harmonic source's frequency is close to that edge. From a multiple-target tracker's point of view, this increase of (correlated) clutter to higher frequencies has to be considered (in the future) in terms of an increasing variance of observation noise to higher frequencies when predicting and updating states.

The structure of this section's remaining part is as follows: First, I discuss each multiple-target tracker's results of experiments with synthesized signals. Second, I discuss the outcomes of experiments with synthetically spatialized real speech signals. And third, I discuss the results of experiments with speech signals recorded in a reverberant environment.

4.12.1 Experiments with Synthesized Signals

The ground-truth signals in the experiments started with a delay of three frames and stopped three frames before the end, which ensured a silence at the beginning and the

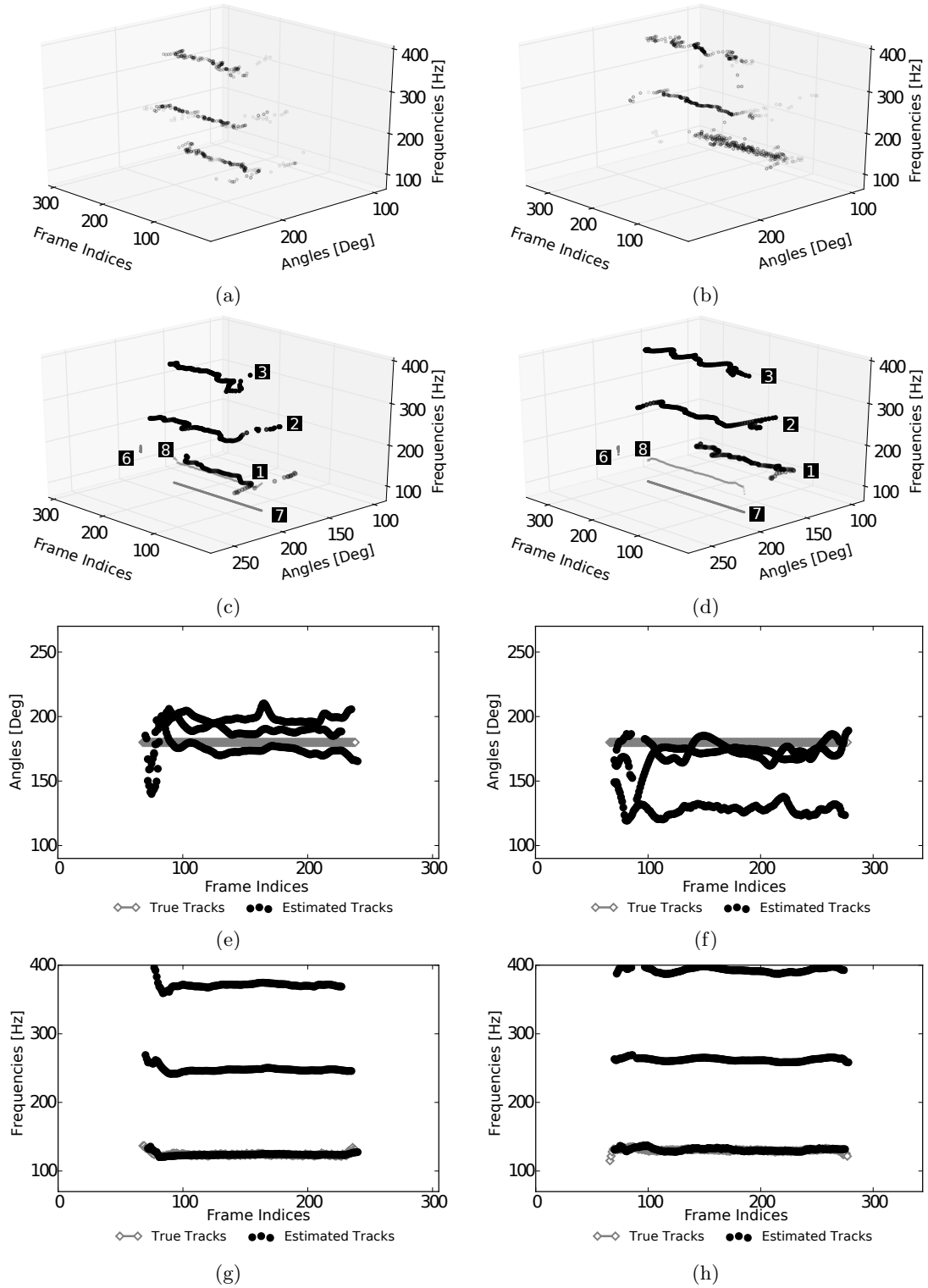


Fig. 4.15: (a,b) Frame-by-frame estimates, (c,d) estimated tracks (black) and (marginalized) ground-truth tracks (gray) of the f_0s, the second and the third harmonic's components produced by the RPDM-based algorithm and the GM-PHD filter, (e,f) tracks of the DOAs in spatial domain after marginalizing over the frequency components, and (g,h) tracks in frequency domain after marginalizing over the spatial components of a speaker uttering a vowel (left column) and a sentence (right column) with almost constant pitch. The labels denote (1) the f_0-trajectory, (2,3) the second and third harmonics' trajectories, (6) the f_0s marginalized over time at the ground-truth DOAs, (7) the DOAs over time marginalized over the ground-truth f_0s, and (8) the true joint parameters over time.

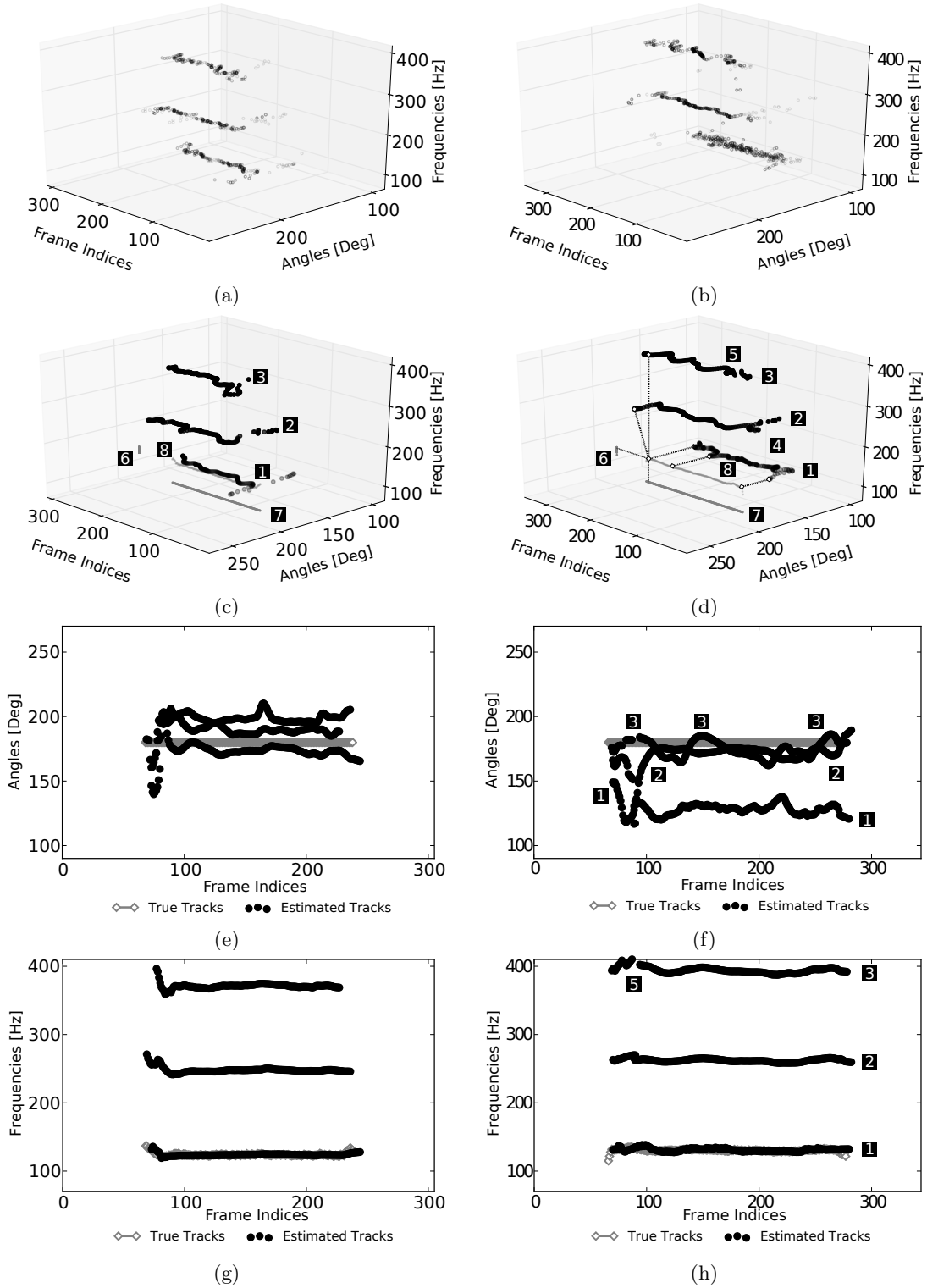


Fig. 4.16: (a,b) Frame-by-frame estimates, (c,d) estimated tracks (black) and (marginalized) ground-truth tracks (gray) of the f_0s, the second and the third harmonic's components produced by the RPDM-based algorithm and the GM-CPHD filter, (e,f) tracks of the DOAs in spatial domain after marginalizing over the frequency components, and (g,h) tracks in frequency domain after marginalizing over the spatial components of a speaker uttering a vowel (left column) and a sentence (right column) with almost constant pitch. The labels denote (1) the f_0-trajectory, (2,3) the second and third harmonics' trajectories, (4) a birth process, (5) a death/birth process, (6) the f_0s marginalized over time at the ground-truth DOAs, (7) the DOAs over time marginalized over the ground-truth f_0s, and (8) the true joint parameters over time.

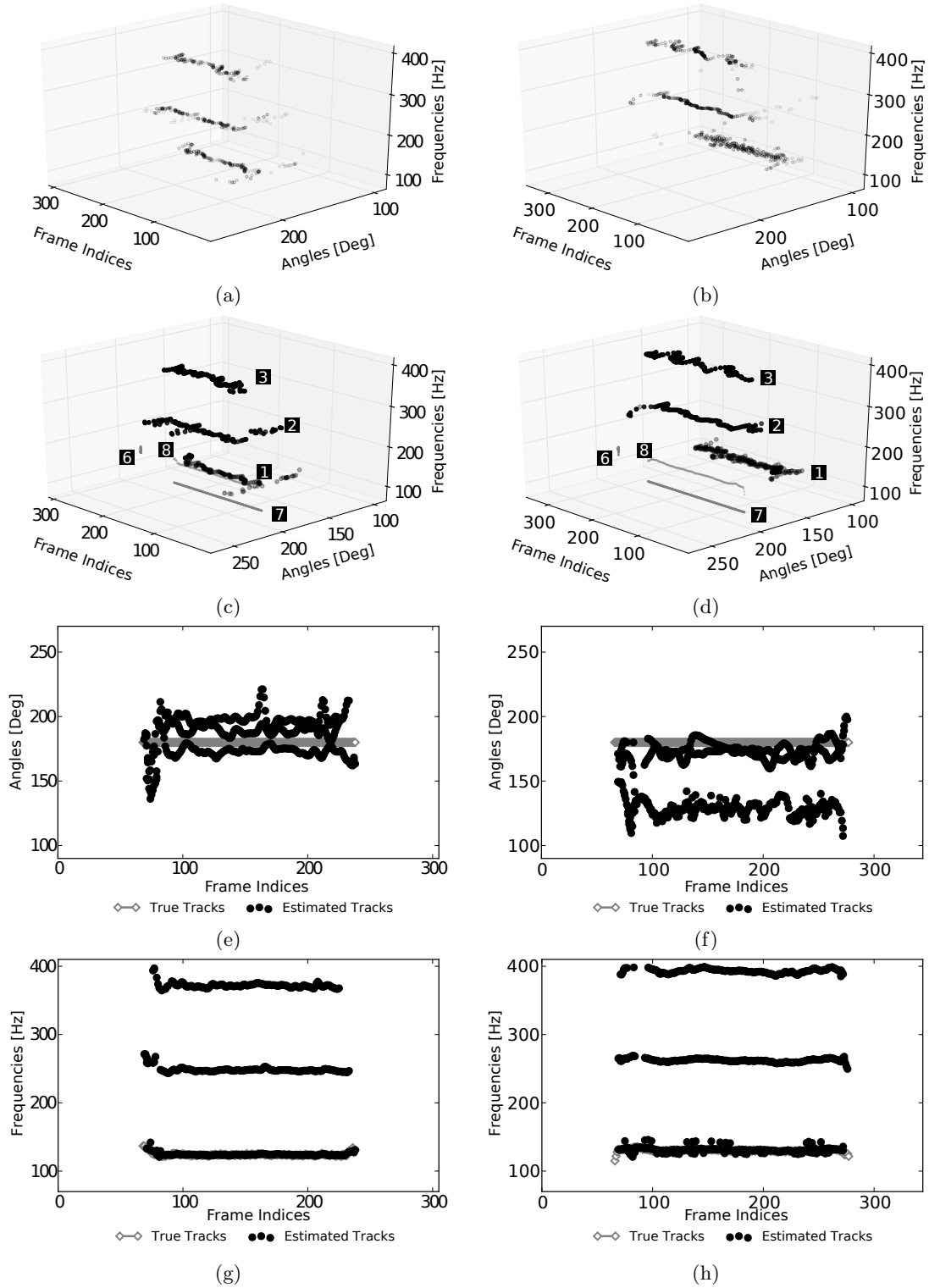


Fig. 4.17: (a,b) Frame-by-frame estimates, (c,d) estimated tracks (black) and (marginalized) ground-truth tracks (gray) of the f_0s, the second and the third harmonic's components produced by the RPDM-based algorithm and the GM-CBMeMBer filter, (e,f) tracks of the DOAs in spatial domain after marginalizing over the frequency components, and (g,h) tracks in frequency domain after marginalizing over the spatial components of a speaker uttering a vowel (left column) and a sentence (right column) with almost constant pitch. The labels denote (1) the f_0-trajectory, (2,3) the second and third harmonics' trajectories, (6) the f_0s marginalized over time at the ground-truth DOAs, (7) the DOAs over time marginalized over the ground-truth f_0s, and (8) the true joint parameters over time.

end of each signal. This was necessary to observe each tracker's birth processes and death processes at the beginning and the end of each experiment.

Depending on the tracker's performance, the cardinality errors were high at the beginning and the end of every experiment; each tracker required two to three frames to successfully start and stop tracking. Hence, the localization errors are zero at these frames, because there is a ground-truth track but no estimated track to compare with or vice versa.

Focusing on the GM-PHD filter's results shown in Fig. 4.3, (c) presents the smoothest tracks in terms of DOA in comparison to all other multiple-target trackers, especially at lower frequencies. In contrast to the other trackers, the GM-PHD filter requires two to three additional frames to start a track as shown in (d). Although (e-g) are snapshots representing the metrics of a single experiment only, they reveal interesting details hidden in the average metric's results. According to (f) there are sudden changes in cardinality, i.e., in the number of tracks. Fig. 4.4 shows the averaged metrics of all experiments (left column) and of experiments with different SNRs (right column). In addition, Table 4.8 presents the means of the averaged metrics of experiments with all SNRs. Comparing with all other trackers' visualized results, Fig. 4.4 (a,c,e), presents the smallest averaged OSPA distances, the smallest averaged cardinality errors, and the highest averaged localization errors making the GM-PHD filter an ideal tracker if a small OSPA distance and a low cardinality error is more important than a high localization accuracy. Distinguishing between different SNRs, Fig. 4.4 (b,d,f) shows the smallest peak value in OSPA distance, the smallest values in cardinality error, but the highest values in localization error over time. This is consistent with the snapshots' evaluations. The cardinality error is high for a low SNR and vice versa, but the error slowly increases towards higher frequencies. As expected, the localization error is high for a low SNR and small for a high SNR. As in case of the cardinality error, the error increases towards higher frequencies. The OSPA distance combines the cardinality error, the localization error, and the labeling error. For SNR $= -10$ dB the OSPA distances are high due to a large number of (spatially correlated) clutter causing deviations in tracks. Contrary to expectations, for SNR $= 10$ dB rather than SNR $= 30$ dB the OSPA distances are the smallest. The higher the SNR, the lower the amount of clutter. The first column of Table 4.8 reflects all these findings in terms of single values.

As shown in Fig. 4.5 (d), the GM-CPHD filter features the smoothest tracks in frequency domain. The filter starts tracking from the beginning; thus, it requires fewer frames to give birth to and to kill tracks—an advantage of balancing the cardinality. However, the higher the ground-truth's frequency at the beginning (e.g., in case of the fourth harmonic), the longer it takes to give birth to a track. In terms of DOA, the track shows deviations at its beginning, which is linked to the estimator's decreased accuracy for low-pitched sources; the tracker cannot produce smooth tracks under such conditions. The snapshot of the GM-CPHD filter's cardinality error in (f) shows the smallest number of cardinality errors and the smallest number of changes. Comparing the averaged metrics in Fig. 4.6 (a,c,e) with those of the GM-PHD filter, the GM-CPHD filter's averaged OSPA distance and averaged cardinality error is higher, but its localization error is smaller. Distinguishing between different SNRs, Fig. 4.6 (b,d,f) shows increasing averaged OSPA distances for increasing frequencies, especially for SNR $= \{10, 20, 30\}$ dB. Only for SNR $= -10$ dB, the curve maintains its OSPA distance after 25 frames. This

is also true for the cardinality errors. However, the localization errors are as expected. The higher the SNR, the larger the error and vice versa. In comparison to the GM-PHD filter, the GM-CPHD filter features smaller localization errors, as reflected in Table 4.8.

Looking at Fig. 4.5 (b,g), there are two striking properties that characterizes the GM-CBMeMBer filter: The first one is the small localization error. In comparison to the GM-PHD filter and the GM-CPHD filter, the GM-CBMeMBer filter features the smallest localization errors over time. Solely at the beginning there is a peak at the first few frames after giving birth to the tracks. Even this peak is much smaller compared to the peaks of the other trackers. However, the second striking property is the number of dropouts, e.g., killed tracks followed by born tracks after a short break. This is due to the pruning of tracks. One major feature of the GM-CBMeMBer filter is that the filter prunes states and tracks; however, to keep the localization error small, I prune more tracks, which causes these dropouts. Considering the snapshots, the GM-CBMeMBer filter exhibits the highest OSPA distances (d) and highest cardinality errors (e), but its localization errors (f) are significantly smaller than in case of the GM-PHD filter and GM-CPHD filter. Comparing the averaged metrics in Fig. 4.8 (a,c,e) with those of the GM-PHD filter and GM-CPHD filter, the GM-CBMeMBer filter's averaged OSPA distance and averaged cardinality error is the highest, but its localization error is the smallest. Distinguishing between different SNRs, Fig. 4.8 (b,d,f) shows the largest peak value in OSPA distance; interestingly, it features the smallest distances for SNR $= -10$ dB. A plausible explanation is that the (correlated) noise causes more tracks close to the ground-truth trajectories. The tracks closer to the ground-truth trajectories feature higher weights and, thus, won't be pruned. Though being a cardinality balanced tracker, it features the largest cardinality errors, but, again, the smallest localization errors, as reflected in Table 4.8. It is interesting to see that the metrics stay almost constant over time.

To sum it up, for the very first time I showed that feeding multiple-target trackers with jointly estimated parameters yields smooth spatio-temporal trajectories. If the goal is to achieve the smoothest tracks in spatial domain, the smallest OSPA distances, and the smallest cardinality errors, then the GM-PHD filter is the best choice. If smooth tracks in frequency domain and small localization errors are important, then the GM-CPHD filter is the best choice. However, if the localization error should be as small as possible, then the GM-CBMeMBer filter should be the first choice.

4.12.2 Experiments with Synthetically Spatialized Real Speech Signals

The main purpose of the experiments with spatially filtered speech signals was to show that the cascade of the RPDM-based algorithm combined with multiple-target trackers yields promising results in the field of localizing, characterizing, and tracking of harmonic sources. The experiments' goal was to successfully produce tracks that correspond to the ground-truth tracks of a speaker representing its current DOA and its f_0. In these experiments I focused on the localization error. It takes the distance between the estimated track and the ground-truth track of the f_0 and DOA into account only, i.e., it ignores the cardinality error and labeling error. According to (c-h) of Fig. 4.9, Fig. 4.11, and Fig. 4.13, one can see that each multiple-target tracker successfully tracks the ground-truth parameters of the male speaker and the female speaker. Table 4.9 confirms that the

GM-CBMeMBer filter yields the smallest localization errors, even in case of real speech. This makes it an ideal tracker if localization accuracy and cardinality play a major and minor role, respectively. For the sake of completeness, I added Fig. 4.10, Fig. 4.12, and Fig. 4.14, showing the OSPA distance, the cardinality error, and the localization error.

The table and the figures show that the GM-PHD filter and the GM-CPHD filter are the best choices in order to compute smooth tracks, whereas the GM-CBMeMBer filter is the best choice when a small localization error is desired.

4.12.3 Experiments with Real Reverberant Speech Recordings

In all experiments with real reverberant speech recordings, the speaker was positioned perpendicular to the array's baseline (axis), as shown in Fig. 4.1 and Fig. 4.2. The estimator as well as the tracker should theoretically return estimates or tracks that correspond to the angle perpendicular to the linear array's axis. The two microphones should capture signals that feature no phase difference to each other. Interestingly, Fig. 4.20 shows something different, so do Fig. 4.15, Fig. 4.16, and Fig. 4.17. Fig. 4.20 (a,b) illustrate the captured signals of the phoneme /j/ of the sentence "Why were you away a year, Roy?". Plot (a) represents broadband signals, plot (b) shows bandpass-filtered signals with lower and upper cutoff frequency of 240 Hz and 280 Hz, respectively. As one can see in (b), the microphone signals feature a phase difference; however, it should be zero. There are several reasons that caused this phase difference. As shown in Table 4.7 the reverberation time T_{30} of the meeting room is larger than 0.5 s in the frequency range of interest. The reverberation and head movements while speaking additionally affects the propagating waves.

The signals' waveforms of impulses generated by clapping hands in front of the array showed that both signals are in phase, i.e., there is no phase difference. For instance, Fig. 4.20 (c,d) depicts the captured signals of the phoneme /w/, which is the first phoneme of the aforementioned sentence. Silence preceded this phoneme; thus, during this time interval the microphones captured direct-path components only. Although the two microphone signals differ in amplitude, they feature a phase difference of zero.

Given a wave field sampled by a two-element microphone array during a time frame, where a non-moving harmonic source positioned perpendicular to the array axis emitted a sinusoidal signal in a small reverberant room. Assuming a frame length of 32 ms and a small array diameter, the estimator will neither estimate the exact ground-truth angle of the speaker nor the exact angle of the reflections. In fact, it estimates the angle of the impinging acoustic wave composed of the source's direct-path component and reflections, whereas the direct-path component features the same frequency as the reflections. As a consequence, the estimated DOA of a time frame is the mean of the captured direct-path component and the captured reflections. Due to varying textured surfaces, furniture, room modes, etc., the paths of the reflections vary with frequency. Therefore, the harmonics' DOAs at a certain instant of time are different, as shown in Fig. 4.16. Example 4 describes this phenomenon in terms of equations and visualizations. Bringing it all together, this phenomenon is the reason why I loose spatial information on reflections and true sources. However, the estimator follows each harmonic's impinging waves that feature the highest energy. This maximizes the SNR. Nonetheless, more studies are required to further investigate this phenomenon.

Example 4. Given a wave field sampled by a two-element microphone array during a time frame, where a non-moving harmonic source positioned perpendicular to the array axis emitted a sinusoidal signal in a small reverberant room. Assuming a frame length of 32 ms, a small array diameter of 0.5 m, a distance between the source and the center of the array of 2 m, and the path lengths of two reflections, which are 3.5 m and 3.6 m, as illustrated in Fig. 4.18 (left). As the following calculation in this simple example will show, the estimated TDOA will neither correspond to the source-related TDOA nor to the TDOA of a reflection.

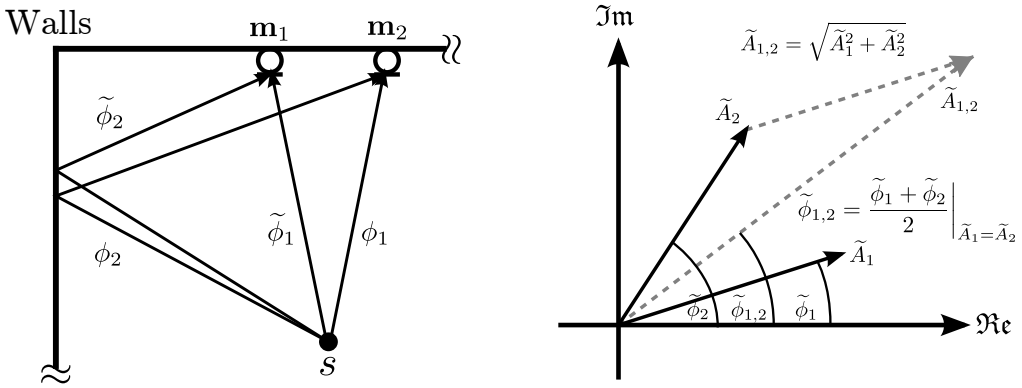


Fig. 4.18: Left: A part of a reverberant room featuring a harmonic source s, two microphones $\mathbf{m}_1$ and $\mathbf{m_2}$ representing the two-element microphone array, two arrows representing the direct-path components labeled as ϕ_1 and $\widetilde{\phi}_1$, and two arrows representing reflections labeled as ϕ_2 and $\widetilde{\phi}_2$. The symbol ϕ denotes the phase of the captured direct-path components and reflections. Right: Complex plane with a vector of length $\widetilde{A}_1$ and phase angle $\widetilde{\phi}_1$ representing the direct-path component captured at microphone $\mathbf{m}_1$, a vector of length $\widetilde{A}_2$ and phase angle $\widetilde{\phi}_2$ representing the reflection captured at microphone $\mathbf{m}_1$, and a vector of length $\widetilde{A}_{1,2}$ and phase angle $\widetilde{\phi}_{1,2}$ representing the sum of the aforementioned vectors. The symbols $\mathfrak{Re}$ and $\mathfrak{Im}$ denote the real axis and the imaginary axis, respectively.

Given the source signal

$$s(t) = A \sin(\omega t)\,, \tag{4.120}$$

the signals captured by microphone $\mathbf{m}_1$ and $\mathbf{m}_2$,

$$x_1(t) = \widetilde{A}_1 \sin\left(\omega t + \widetilde{\phi}_1\right) + \widetilde{A}_2 \sin\left(\omega t + \widetilde{\phi}_2\right) \tag{4.121}$$

and

$$x_2(t) = A_1 \sin(\omega t + \phi_1) + A_2 \sin(\omega t + \phi_2)\,. \tag{4.122}$$

If there is no attenuation, then $A_1 = \widetilde{A}_1$, $A_2 = \widetilde{A}_2$, and $A_1 = A_2$.

According to (3.15) the TDOA is

$$\tau_{1,2}(\omega) = -\frac{\Delta\phi_{1,2}}{\omega} \tag{4.123}$$

with

$$\Delta\phi_{1,2} = \frac{\widetilde{\phi}_1 + \widetilde{\phi}_2}{2} - \frac{\phi_1 + \phi_2}{2}. \tag{4.124}$$

I derived $(\widetilde{\phi}_1 + \widetilde{\phi}_2)/2$ in Fig. 4.18 (right). Rewriting (4.123) yields

$$\tau_{1,2}(\omega) = -\frac{\frac{\widetilde{\phi}_1 + \widetilde{\phi}_2}{2} - \frac{\phi_1 + \phi_2}{2}}{\omega} = \frac{\phi_1 + \phi_2 - \widetilde{\phi}_1 - \widetilde{\phi}_2}{2\omega}. \tag{4.125}$$

Since the source is perpendicular to the microphone array's center, $\widetilde{\phi}_1 = \phi_1$. This results in

$$\tau_{1,2}(\omega)\big|_{\widetilde{\phi}_1=\phi_1} = \frac{1}{\omega} \cdot \frac{\phi_2 - \widetilde{\phi}_2}{2}. \tag{4.126}$$

The resulting TDOA neither refers to the TDOA of the direct-path component nor to the TDOAs of both reflections. It refers to the mean of both reflections' phases. ■

In Fig. 4.16 (d,f,h) there are numbered labels. Number one, two, and three label the trajectory of the f_0 and the trajectories of the second and third harmonics, respectively. Number six labels the utterance's true f_0s marginalized over time at the ground-truth DOAs. Number seven labels the speaker's DOA marginalized over the ground-truth f_0s. The label with number eight marks the true joint parameters over time.

In plot (d) one can see that the estimated trajectories feature different and varying DOAs. The first harmonic's trajectory features different DOAs than the second harmonic's trajectory. Plot (f) emphasizes this phenomenon. Against my expectations, the trajectories do not permanently overlap when marginalizing over the frequencies. The second and third harmonic's trajectory is close to the ground-truth, i.e., 180°. But the first harmonic features significant deviations of up to 50°. However, all trajectories are smooth, the estimates in (b) are closely distributed around the estimated trajectories. The deviations are high at the beginning of the tracks. This is due to early, strong reflections in the acoustic wave field. Plot (h) shows trajectories of the f_0s and components of the second and third harmonic without significant deviations. These trajectories are as expected. As shown at the beginning of the track labeled with one, the tracker requires two to three frames of estimates before starting a track; this corresponds to the theory. Label five marks a death/birth process although the signal is still present. This is due to the frequency domain's upper limit, which is obviously exceeded at this point.

Prima facie, the data association seems to fail when trying to assign a trajectory of f_0s and trajectories of the corresponding harmonic components to a source. Given such a challenging environment as described before, the spatial components of a source's trajectories feature different DOAs at a certain instant of time. However, I can still assign

Algorithm 5: Iterative Grouping of Trajectories

Data: $\mathfrak{G}$ (set of trajectories of harmonic sources),
G, H (trajectories), V_G, V_H (trajectories' support points);
Result: $\mathfrak{H}_{i_s}$ (set of grouped trajectories assigned to a source);

$V_{G,H} \equiv V_G \cap V_H$;
$d\left(G^{(i_g)}, H^{(i_g)}\right) \equiv \ln\left(G^{(i_g)}\right) - \ln\left(H^{(i_g)}\right)$;
$\Delta d\left(G^{(i_g)}, H^{(i_g)}\right) \equiv d\left(G^{(i_g)}, H^{(i_g)}\right) - d\left(G^{(i_g-1)}, H^{(i_g-1)}\right)$;
$i_s = 1$;
repeat
  $G \in \mathfrak{G}$;
  $\mathfrak{H}_{i_s} = \left\{ H \middle| H \in \mathfrak{G} \wedge |V_{G,H}|^{-1} \sum_{i_g \in V_{G,H}} \Delta d\left(G^{(i_g)}, H^{(i_g)}\right) \leq \varepsilon \right\}$;
  $\mathfrak{G} = \mathfrak{G} \setminus \mathfrak{H}_{i_s}$
  $i_s = i_s + 1$;
until $\mathfrak{G} = \emptyset$;

Fig. 4.19: Pseudo-code for grouping the trajectories of a harmonic source. In case of a real reverberant environment, the harmonic source's trajectories rarely overlap in spatial domain; however, in frequency domain they are integer multiples of the trajectory representing the fundamental frequencies. Thus, one can group the trajectories of a source assuming that, e.g., speech is sparse in temporal-frequency domain. In the pseudo-code, G and H are trajectories, V_G and V_H are sets containing the support points of the trajectories G and H, respectively, $V_{G,H}$ is a set of support points of the intersection of set V_G and V_z, $d(\cdot)$ is a distance measure between two values, $\Delta d(\cdot)$ is a finite difference, i_g is the frame index, i_s is the source index, and ε is a very small value. For simplicity, the trajectories G and H contain frequency components only; in practice, consider the frequency components of G and H only. Ideally, $\varepsilon = 0$; however, due to mismatches and finite precision, $\varepsilon > 0$.

trajectories to their corresponding source, because, first, the spatial components are still associated with the frequency components in the SJPS, second, the sources' trajectories are integer multiples of their trajectories containing the f_0s. To group trajectories that correspond to a certain source, I assume that speech is sparse in the time-frequency domain [63–66]. Fig. 4.19 presents an algorithm that groups each source's trajectories and allows to estimate the number of harmonic sources. Additionally, it assigns fragments of a trajectory (caused by unexpected death-and-birth processes) to the correct group of trajectories.

The existing approaches as well as the POPI-based approaches described in the introduction expect the harmonics of a harmonic source's signal to feature the same DOAs per frame. This is why they will fail to find the speaker's exact DOA and f_0 in such challenging acoustic environments. A solution is to extend their dictionaries so that they cover the cases where a source's harmonics feature different DOAs at a certain instant of time. However, this would dramatically increase the processing time (i.e., the time to find the maximum argument).

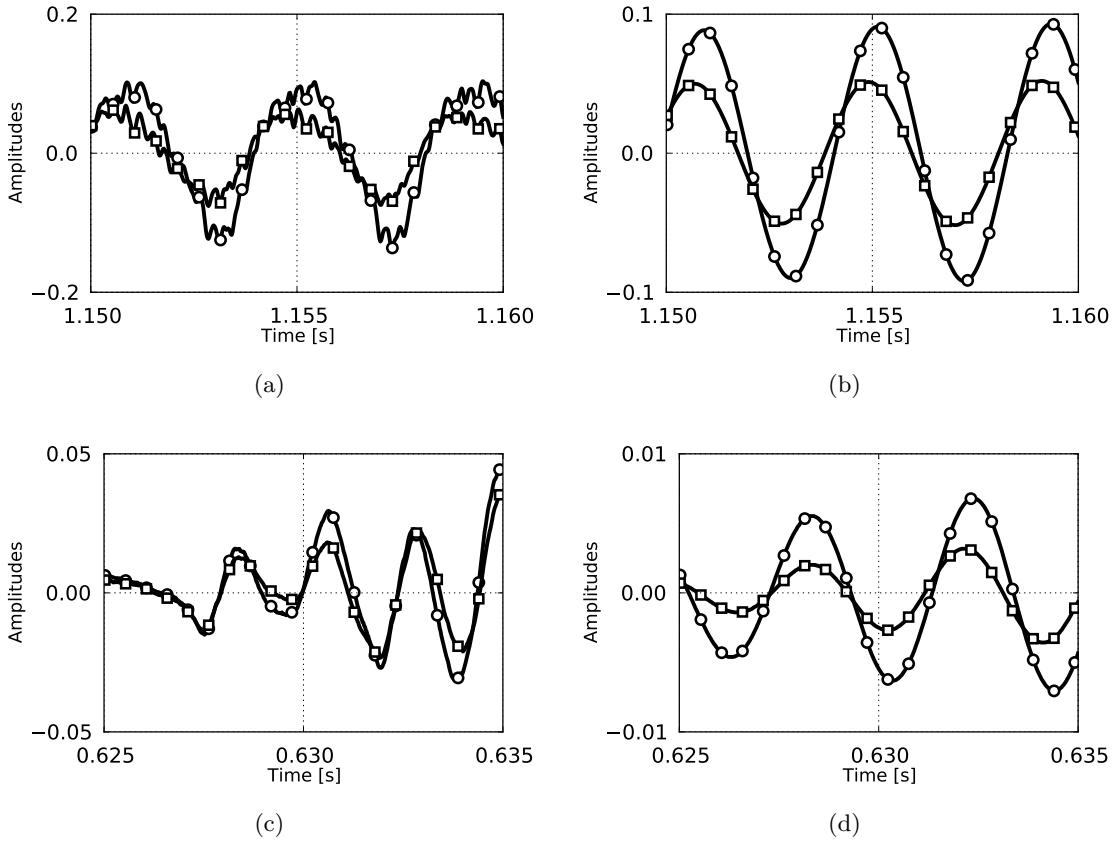


Fig. 4.20: Time signals of two microphones representing a snapshot of the sentence "Why were you away a year, Roy?" spoken by a speaker positioned perpendicular to the microphone array's baseline. Plots (a,b) show time signals of the phoneme /j/, whereas plots (c,d) show time signals of the phoneme /w/. Each signal's curve features a certain marker. Plots (a,c) show snapshots of the broadband signals, plots (b,d) show snapshots of bandpass-filtered signals with lower and upper cutoff frequency 240 Hz and 280 Hz, respectively.

Chapter 5

AMISCO: The Austrian German Multi-Sensor Corpus[1]

In this chapter, I finally introduce the aforementioned corpus entitled AMISCO: the Austrian German multi-sensor corpus; it is the first corpus of its sort. I discuss existing corpora, compare some of them with the proposed corpus, thoroughly describe the data collection, the editing, the post-processing, as well as the quality assurance and the validation.

5.1 Contributions and Innovations

This unique, comprehensive Austrian German speech corpus features glottograms and recordings labeled with the speakers' f_0s and spatial information. These special labels are a prerequisite to conduct experiments with algorithms that jointly estimate the DOAs and the f_0s harmonic sources. It is a collection of two-room and 43-channel close-talking and distant-talking high-quality speech recordings from 24 moving and non-moving single speakers, balanced male and female. It contains around 8.2 hours of read speech, 53,000 word tokens based on 2,700 unique word types.

5.2 The First Corpus of Its Sort

In the field of distant speech enhancement [115, 116], several research teams dedicated their time to jointly detect or estimate a source's DOA and f_0 with two or more microphones. Finding these parameters is a prerequisite to improve, e.g., the word accuracy rate of a speech recognizer by applying beamforming or source separation algorithms. They applied their algorithms to synthesized harmonic signals [21, 24], signals from musical instruments [18, 23], snapshots of filtered clean-speech signals [17, 22], synthetically spatialized signals [22], or speech signals without having a reliable ground truth of the

[1]This chapter is substantially based on the conference paper [51] and was revised and adapted to the present thesis. As first author of the conference paper, I did the planning and the post-processing of the corpus on my own, whereas colleagues (Martin Hagmüller and Thomas C. Pichler) and I did the recording as well as the quality assurance (and some parts explicitly stated in the chapter.) Moreover, I wrote the conference paper all by myself.

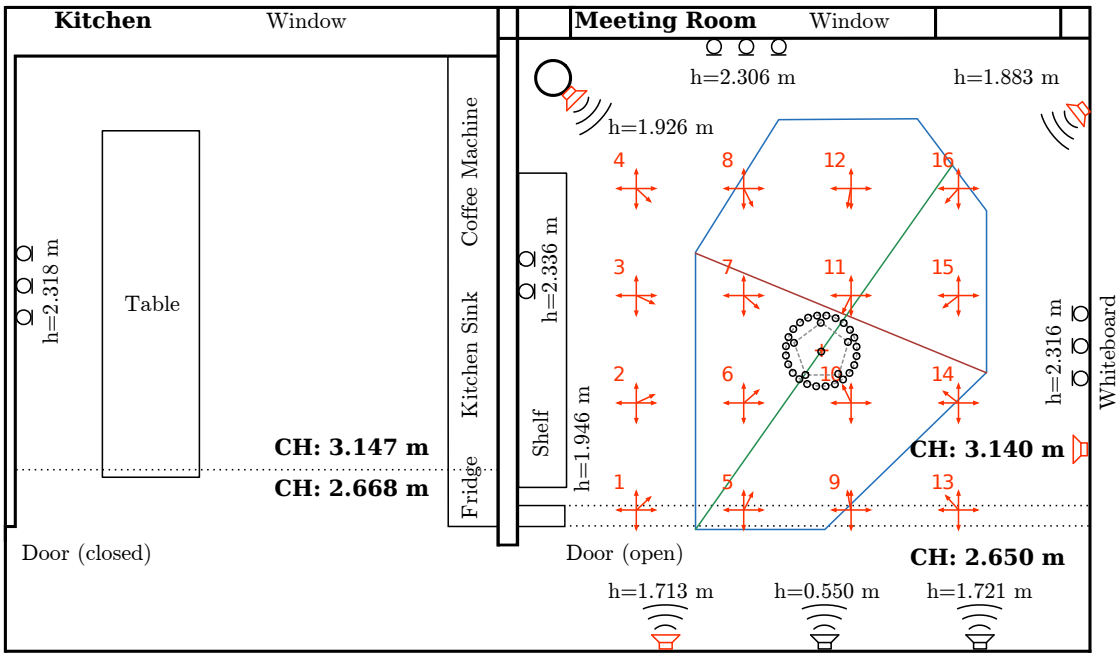


Fig. 5.1: Floor plan of the recording environment. The kitchen on the left features three, the meeting room on the right 38 microphones. All microphones but the microphones in the meeting room's center are mounted on the wall; the remaining microphones, i.e., the microphones of the uniform circular array and the pentagonal array) are mounted on the ceiling. Additionally, the meeting room features a video camera and four Kinects (highlighted as red loudspeaker symbols: top right, center right, bottom left, top left). There are five loudspeakers (illustrated as red or black loudspeaker symbols with three arcs: bottom left to bottom right, top left, top right) mounted on the walls. A red loudspeaker symbol featuring three arcs denotes a loudspeaker and a Kinect or a camera. The crosses' center represents the speakers' positions (labeled with numbers from 1 to 16), whereas the arrows signify the orientation of the speakers. The blue, green, and red lines represent the trajectories. The label `CH` denotes the ceiling height.

f_0 [35]. One thing they all had in common was no access to multi-channel speech signals recorded in real environments and labeled with f_0s and DOAs or positions in space, because such data did not exist.

To make such data publicly available, I, therefore, set up a corpus containing Austrian German multi-sensor, multi-channel speech recordings in a real environment, shown in Fig. 5.1 and Fig. 5.2, labeled with a speaker's f_0s, positions, orientations, and other parameters. The new corpus offers glottograms that can be used in prosody analysis, speech coding, speaker identification, as well as speech recognition. They are a prerequisite to evaluate pitch-detectors, e.g., YIN [117] and RAPT [118], and pitch-trackers [119]. Furthermore, the corpus is suitable for linguistic studies, various machine learning and (multi-modal and multi-channel) signal processing tasks, and studies related to a speaker's f_0.

There are outstanding corpora available; however, they do not meet my requirements to jointly detect and estimate DOAs and f_0s.

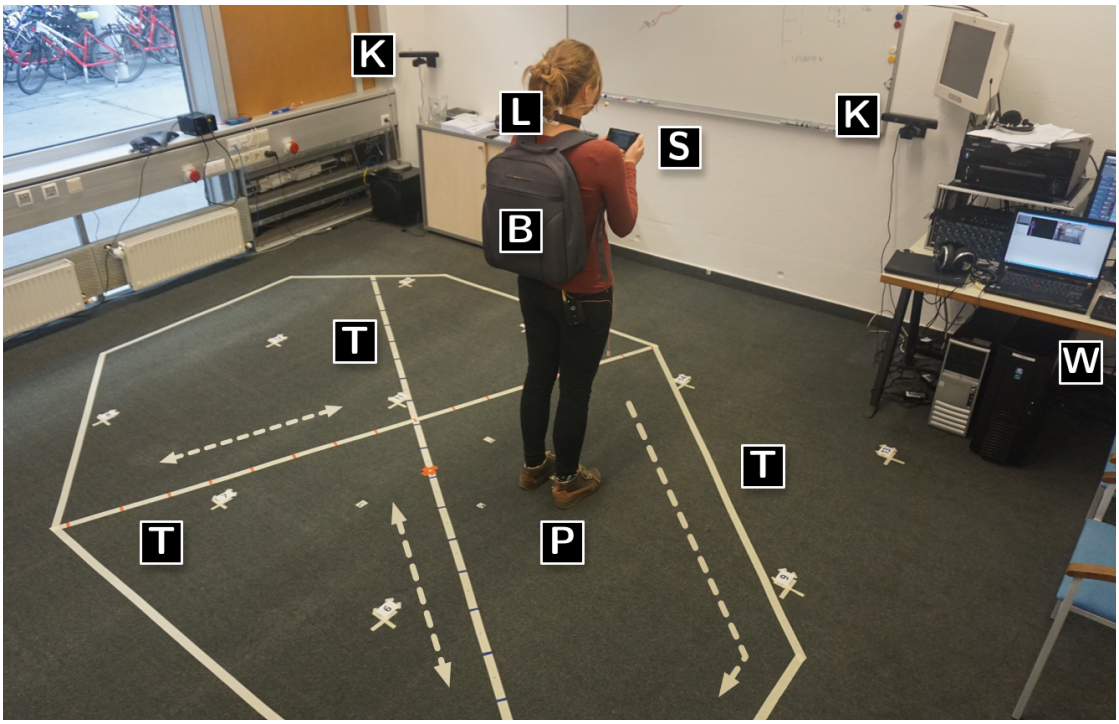


Fig. 5.2: A speaker reads sentences shown on the smart-phone's screen (S) at position ten (P) facing east in the meeting room. There are two Kinects (K) next to the white-board, a work station (W) on the right-hand side, and marked trajectories (T) and positions (P) on the floor. The speaker is wearing a backpack (B) containing a battery-driven laryngograph and transmitters. The laryngograph's sensor (L) is mounted on the speaker's neck. The bright arrows on the floor mark the directions of movement, the small cross-shaped markers represent the positions with four orientations. The red spot on the left-hand side of the speaker marks the pentagonal array's center (mounted on the ceiling).

On the one hand, there are four corpora that include glottograms: the Mocha-TIMIT database [120], the Keele corpus [121], the PTDB-TUG corpus [122], and the GRASS corpus, [123]. The Mocha-TIMIT [120] includes single-channel recordings from a male and a female speaker sampled at 16 kHz. The PTDB-TUG [122] and the Keele corpus [121] contain single-channel and close-talking speech recordings, which cannot be used for multi-channel experiments. The GRASS corpus [123], provides glottograms that are highly distorted.

On the other hand, there are corpora containing multi-room and multi-channel recordings: the ATHENA corpus [124] and the DIRHA-GRID corpus [125].

However, none of these corpora contains moving speakers, and none of them but [123] contains any (Austrian) German speech recordings, which are indispensable for experiments with (Austrian) German voice-controlled systems. Almost all corpora mentioned before lack video recordings, which are important for audio-visual experiments and quality assurance. In [126], Le Roux et al. summarized the properties of over 40 data sets; however, neither ShATR [127], AV16.3 [128], ICSI Meet [129], nor NIST meet [130], CHIL [131], AMI [132], ETAPE [133], just to mention a few, feature ground-truth data on f_0s. Thus, these data sets are useless for jointly estimating the f_0s and the DOAs of harmonic sources.

All these lacks led me to set up a new, unique, and comprehensive corpus called "AMISCO: The Austrian German Multi-Sensor Corpus."

5.3 Data Collection and Editing

5.3.1 Speakers

The corpus contains read-speech produced by 24 speakers, balanced male and female, aged between 25 and 52, 23 of them with Austrian German, and one with German German accent to be able to draw a rough comparison between both variants' pronunciations. Since discussing the differences between Austrian German and German German is out of this article's scope, I refer to [123, 134, 135] for more information about this topic. All speakers but one were born in Austria; one was born in Germany. Those who were born in Austria grew up in non-western provinces, which ensures reduced dialectal variations. At the time of recording, all of them lived in Graz, Austria, and exhibited a higher education with at least a university degree. I asked each speaker to fill in a short questionnaire to get an overview of his/her language skills, education, diseases of the vocal tract (including the larynx), and other personal parameters. Additionally, I measured each speaker's body height which is a prerequisite to approximate the position of the signal-emitting head in space.

5.3.2 Equipment

To guarantee high-quality recordings, two colleagues, Thomas C. Pichler and Martin Hagmüller, and I employed the high-end RME Hammerfall HDSPe RayDAT sound card with a HDSP 9632 word clock module to connect the computer with three Focusrite OctoPre MkII Dynamic pre-amplifiers and three audio interfaces. Additionally, we utilized an AKG CMS 70 dual station for wireless recordings, an Intel Xeon CPU E3-1275L v3 (4 cores, 8 threads) with 32 GB EEC RAM, 120 GB and 250 GB SSD, and Debian Linux

7.8 with Kernel 3.14. We sampled the acoustic wave field by employing different types of arrays: a wall-mounted 2-element linear array with a length of 30 cm, three wall-mounted 3-element linear arrays with a length of 60 cm, and a ceiling-mounted 5-element pentagonal array with a diameter of 54.44 cm and a microphone in its center. These microphone arrays featured SHURE MX391 Microflex Boundary omni-directional boundary microphones. We also used 24 MP34DT01 MEMS omni-directional microphones connected to three AST-Robotics STM32F407 micro-controller-boards to set up a 24-element (3×8 MEMS microphones) circular microphone array with a diameter of 61.90 cm; it surrounded the pentagonal array. The micro-controller-boards were connected to the server via USB. To facilitate close-talking and laryngograph recordings, we employed an AKG HC577 L headset microphone, a portable laryngograph, and two wireless transmitters (of the AKG CMS 70 dual station) connected to these devices. The transmitters and the receivers were synchronized with the central word clock. Additionally, we used four Kinects (featuring the Microsoft Kinect skeleton tracker based on SDK v1.8) for skeleton tracking and a video camera; the Kinects and the camera captured 30 frames per second. A multi-core computer operated the Kinects and transmitted the captured data via TCP/IP to the main server. For recording and post-processing, particularly for synchronizing audio with video data, we employed the following digital audio workstations: Ardour 3.5.403 and Reaper 4.77. In total, we applied 43 acoustic sensors (including the laryngograph's sensor which measures the laryngeal transconductance), four Kinects (for skeleton tracking), and a video camera.

5.3.3 Recording Environment

We did the recordings in rooms that are characteristic for ambient assisted living and staff meetings [1, 136], i.e., a kitchen at home and a meeting room in a company. In these rooms, distant speech enhancement, e.g., localization and characterization of multiple sources [50], has to be applied for successful speech recognition. Fig. 5.2 and Fig. 5.1 show the recording environment consisting of a meeting room with dimensions $(5.3 \times 5.8 \times 3.1)$ m and a kitchen with dimensions $(4.0 \times 5.7 \times 3.1)$ m. The reverberation time in the meeting room is around $T_{60,c} \approx 500$ ms, whereas the reverberation time in the kitchen is about $T_{60,k} \approx 700$ ms when placing sound-emitting sources in the meeting room and keeping the connecting door open.

5.3.4 Calibration

To guarantee a well-calibrated recording system, we generated a diffuse noise field, measured the average captured power (over all frequencies) of each channel, and adjusted each channel's gain to obtain the same average captured power for each channel. We employed five pre-installed hi-fi loudspeakers in the meeting room to play back white noise. To verify a diffuse noise field, we measured the A-weighted equivalent sound pressure level in front of each microphone by applying a sound level meter. Measurements revealed level differences between ± 1 dB. To calibrate the Kinects, we selected four position markers in the center of the meeting room (at position 6, 7, 10, and 11), which were in the visual field of all Kinects. We computed the markers' coordinates by using a laser distance meter and walked within the area spanned by those markers on well-defined paths to determine the deviations. Then, we adjusted the Kinects' positions

and orientations by hand and measured both parameters by applying a laser distance meter. After repeating the measurement procedure several times, we averaged all measurements and entered the resulting coordinates in the Kinects' config-files to improve their accuracy.

5.3.5 Recording Procedure

Each speaker read items that appeared on a smart-phone's screen. At the beginning, I informed the speaker about the purpose of the recording, and he/she signed a statement of agreement (e.g., including our commitment to preserve the speaker's anonymity). Afterwards, I instructed the speaker about the overall procedure and equipped him/her with a headset, a backpack containing a battery-driven laryngograph and wireless transmitters. On the neck close to the larynx we mounted the laryngograph's sensors. We recorded the speaker in one session (50-60 min) composed of three sub-sessions (10-12 min) and short breaks where we informed the speaker about the upcoming tasks. In sub-session 1, the speaker read 104 short items at positions, which were selected uniformly at random, with 5 different orientations per position: north, east, south, west, and center of the pentagonal array. In sub-session 2, the speaker walked at constant speed along predefined trajectories marked at the floor. We split this sub-session into three parts: (1) reading 24 long sentences and walking along the heptagon-shaped trajectory clockwise, walking along the inner, straight trajectories from (2) west to east and (3) north to south, and vice versa, and reading in total 40 long sentences. Sub-session 3 was identical to sub-session 1, except that the speaker read 64 long items. During the whole session, a colleague and I (we were sitting in the same room) supervised the speaker by verifying the read items, the positions and orientations, and by visually examining the speaker's gait velocity in sub-session 2. Furthermore, we triggered the change of sentences-to-read by manually changing the slides (shown on the smart phone) using TeamViewer. In case of any bloopers, interferences, or other problems, we told the speaker to stop, to wait, and to read the item again.

5.3.6 Acquisition Data

The recorded utterances represent read speech, phonetically balanced sentences, commands, and digits. The sentences were identical to those used in the GRASS corpus [123], and the orthographic transcriptions include around 53,000 word tokens and 2,070 unique word types. We recorded the utterances with a sampling frequency of 48 kHz and encoded them with PCM S24 LE (araw). After synchronizing the audio recordings with the video data we set markers ~ 0.5 s before and after each utterance by hand. Then, we exported the markers as a text file (one file per session) and split the original multi-channel files and the Kinects' skeleton tracks into chunks. Table 5.1 shows the minimum, maximum, and average segmental SNR over all speakers for one microphone of each microphone array and the headset in dB. (I describe the segmental SNR's computation in the next section.) The varying SNR-values are due to varying speech levels and distances between each speaker and the corresponding microphone. Apart from speech recordings (see Fig. 5.3 and Fig. 5.4), we provide estimated f_0s, glottograms, positions and orientations of each speaker, files containing additional information about the speaker and the scenarios, as well as orthographic transcriptions. Moreover, we provide different noise

Table 5.1: Minimum, maximum, and average signal-to-noise ratio over all speakers in dB for one microphone of each microphone array and the headset. CPR denotes the meeting room.

	Min-SNR [dB]	Max-SNR [dB]	Avg-SNR [dB]
Headset	23.58	52.23	38.67
CPR 1-2	18.97	36.43	24.73
CPR 3-5	19.14	35.75	24.84
CPR 6-8	19.32	36.66	25.09
CPR 9-14	19.06	37.14	24.33
Kitchen 15-17	17.27	31.54	21.35
MEMS M1	20.68	42.14	27.46
MEMS M2	21.97	45.09	29.62
MEMS M3	21.50	44.53	29.15

recordings in the meeting room and kitchen: moving chairs, smashing and closing doors, running water-tap, etc.

5.4 Post-Processing

5.4.1 Signal-to-Noise Ratio

I determined segmental SNRs of each speaker's recorded utterance by applying a short-term power estimation utilizing a first-order IIR smoothing of the signal's instantaneous power [85] of a microphone signal $x[n]$ according to

$$P_s[n] = (1 - \gamma_s[n]) \cdot x^2[n] + \gamma_s[n] \cdot P_s[n-1] \tag{5.1}$$

with

$$\gamma_s[n] = \begin{cases} \gamma_r & \text{if } x^2[n] > P_s[n-1] \\ \gamma_f & \text{otherwise} \end{cases}. \tag{5.2}$$

Variables γ_r and γ_f are smoothing constants for rising and falling signal edges, $x[n]$ is the input signal at index n, and $P_s[n]$ is the smoothed instantaneous power of the signal. Then, I estimated the local background noise power as follows:

$$P_b[n] = (1 + \epsilon_s) \cdot \min\left(P_s[n], P_b[n-1]\right), \tag{5.3}$$

where ϵ_s is a small positive constant, which controls the maximum speed for increasing the estimated noise level. After this, I computed the power ratio,

$$P[n] = 10 \cdot \log_{10}\left(\{P_s[n] - P_b[n]\}/P_b[n]\right), \tag{5.4}$$

and averaged all values of $P[n]$ above a certain threshold μ_{SNR} yielding the average SNR per audio file:

$$\text{SNR}^{(i_2)} = \frac{1}{|\mathcal{K}|} \sum_{i_1 \in \mathcal{K}} P[i_1] \tag{5.5}$$

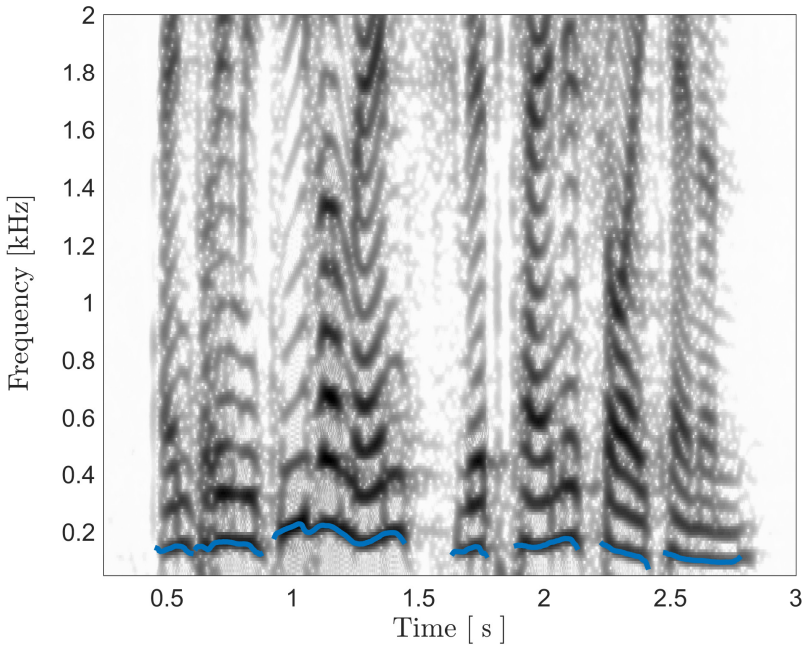
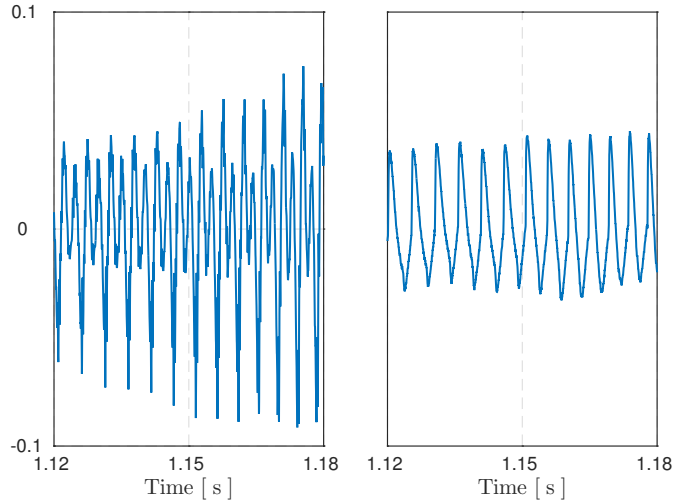
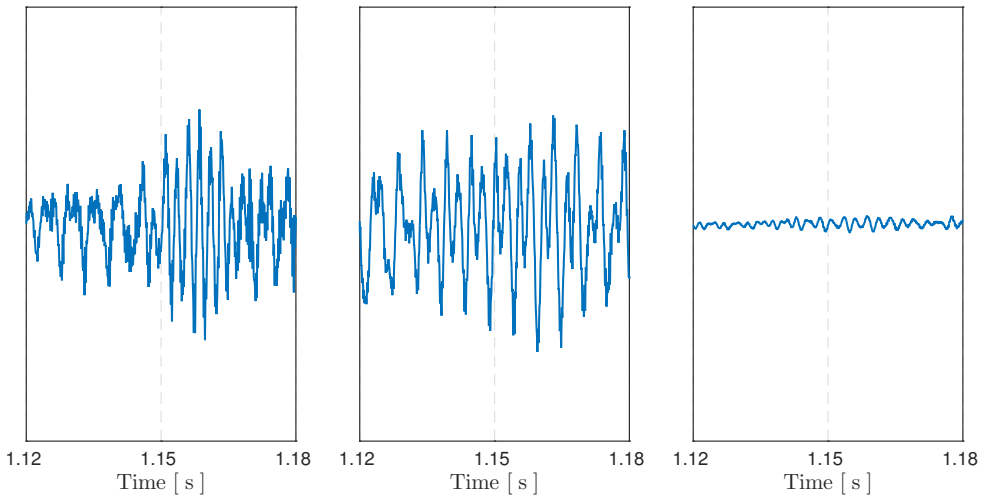


Fig. 5.3: Spectrogram of a male speaker's speech signal recorded with a headset. The (German-language) utterance is [aɪn priːmiːtiːvɐ mɛnʃ vɪrt kaɪmə ʃɔʏ kɛnnən] (IPA). The f_0's ground-truth values are marked with a blue line. The corresponding audio file used for this figure is `22_m_short_1_wireless_042_9_1.wav`, where `22` denotes the speaker's id-number, `m` is the gender, `short_1` is the session, `wireless` denotes the headset's recording, `042` is the id-number of the spoken item, `9` is the position, and `1` is the orientation, respectively.



(a) Headset. (b) Glottogram.



(c) CPR 1. (d) MEMS M1/1. (e) Kitchen 15.

Fig. 5.4: Time signals of the first phoneme /eː/ of the (German-language) sentence [jeː neːa dɛʁ tsaɪgʁ aʊf axt kaːm dɛstɔ ʊnruːɪgʁ vʊʁdən diː lɔʏtə] (IPA) read by speaker two. To plot these figures, I used the file named `02_f_long_2_<LABEL>_028_6_2.wav`, where `<LABEL>` is a wildcard used for the room and the microphone number. For instance, the used audio file in plot (c) is `02_f_long_2_cpr1-2_028_6_2.wav`, where `02` denotes the speaker's id-number, `f` is the gender, `long_2` is the session, `cpr1-2` represents the room's label (cpr) and the used microphones (1 and 2), `028` denotes the id-number of the spoken item, `6` is the position, and `2` is the orientation, respectively. The figures show the signals of (a) the headset microphone, (b) the laryngograph (represented as a glottogram), (c) the first microphone of the linear array labeled as CPR1-2, (d) a MEMS microphone of the circular array's first MEMS module, and (e) the first microphone of the kitchen's linear array. In comparison to (a), the signals in (c-e) are time-shifted and filtered due to the time-differences of arrival and the influence of the room.

with $\mathcal{K} = \{i_1 \mid \forall i_1 : P[i_1] \geq \mu_{\text{SNR}}\}$, where $|\mathcal{K}|$ is the cardinality of set $\mathcal{K}$. Averaging the SNRs of all speakers' utterances yields the overall SNR per microphone:

$$\text{SNR} = \frac{1}{N_u} \sum_{i_2=1}^{N_u} \text{SNR}^{(i_2)}, \tag{5.6}$$

where N_u is the number of all utterances of all speakers. I chose a sampling frequency of $f_s = 48$ kHz, $\gamma_r = 0.99$, $\gamma_f = 0.97$, $\epsilon_s = 2 \cdot 10^{-5}$, $\mu_{\text{SNR}} = 15$ dB, and $P_b[0] = P_s[0] = x[0]^2$ as initial values.

5.4.2 Resampling & Filtering Skeleton Tracks

Since the Kinects delivered unequally spaced detections in time (they lacked a clock to control the frame rate), the data points had to be resampled with equally spaced 30 fps. Assuming that the speakers had a constant gait velocity (which we verified by visually examining their velocity), I resampled the resulting skeleton tracks (provided by the Kinect skeleton tracker) by considering linear interpolation, which yielded data points with equally spaced time-intervals. The measurement of the Kinects' positions with a laser distance meter by hand introduced a small systematic error. Thus, I decided to make use of some prior knowledge: all speakers were walking on marked trajectories. Moreover, the visual evaluations of the videos revealed that all speakers were walking on the trajectories, which were marked on the floor, without (visually) noticeable deviations (e.g., deviations smaller than ±15 cm). Therefore, I computed the squared error between each detection and a fine grid of points on the trajectories. Then I determined the point on the trajectories where the squared error of a detection exhibited the global minimum, and mapped the detection to this point of the trajectory (see Fig. 5.5). The corpus provides the original and modified skeleton tracks as text files.

5.4.3 Estimating Fundamental Frequency

First, I upsampled the glottogram from $fs = 48$ kHz to $fs = 96$ kHz. Then, I filtered the signal with a Kaiser window order-estimated bandpass filter with a lower and upper cut-off frequency of 70 Hz and 8000 Hz. After compensating the introduced group-delay, I split the whole signal into frames with a frame length of 32 ms and a frame shift of 5 ms. I computed the one-sided unbiased auto-correlation of each frame and applied a maximum detector based on the Lemire-algorithm [79] with a sliding-window size of 10 samples to each frame. After eliminating maxima with a lag below 2 ms and above 13 ms, I selected the global maximum of all remaining maxima. To eliminate outliers (e.g., caused by the speaker's act of swallowing), I computed the first derivative of the f_0-trajectory and eliminated sudden jumps with at least a minor sixth—this corresponds to a pitch ratio of $8 : 5$—upwards and downwards. The corpus provides the glottograms as wav-files and the trajectories of the estimated f_0s (see Fig. 5.6) as text files.

5.4.4 Orthographic Transcription

To generate accurate transcriptions of the recorded utterances, I followed the transcription guidelines mentioned in [123], which lists all symbols used for the orthographic transcription.

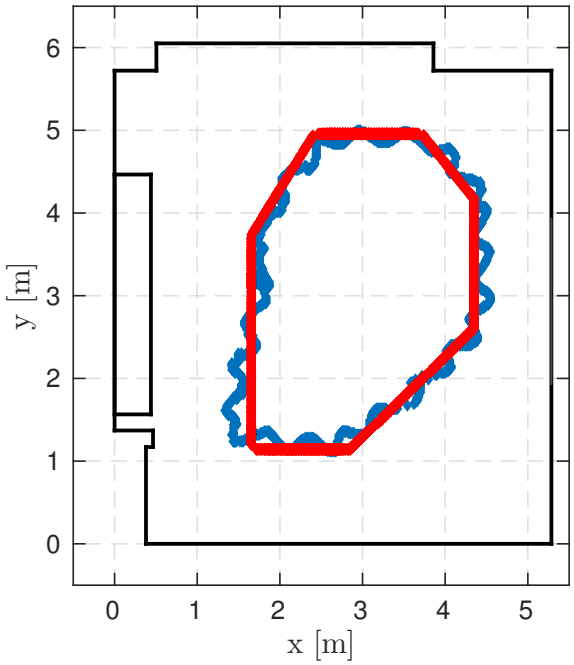


Fig. 5.5: The original (blue and curvy) and modified (red and straight) skeleton tracks represented as trajectories in the meeting room's floor plan. These trajectories correspond to a snapshot of a speaker's movements in sub-session 2.

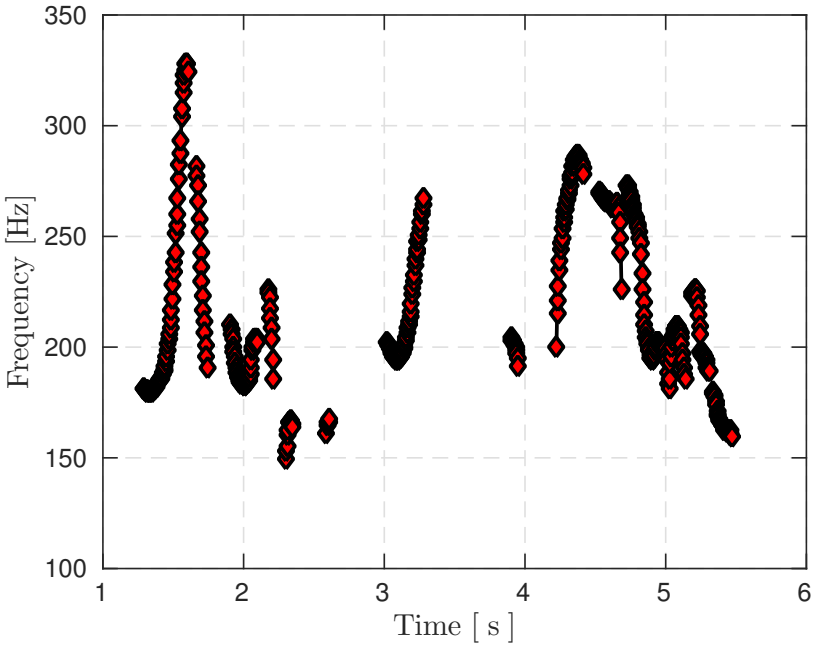


Fig. 5.6: The estimated f_0-trajectory of speaker 2 uttering the (German-language) sentence [jeː neːa dɛʁ tsaɪgʁ aʊf axt kaːm dɛstɔ ʊnruːgʁ vʊʁdən diː lɔʏtə] (IPA).

5.5 Quality Assurance & Validation

I prepared protocols for each speaker's (sub-)session beforehand; these protocols defined what to say and where to go. They contained all selected sentences, positions, as well as orientations. During each (sub-)session, two assistants supervised the speaker by verifying the read items, the positions and orientations, and the speaker's gait velocity. I used the protocols to create a first transcription of the utterances. Afterwards, two colleagues and I checked all recorded utterances, video tracks, and text files, and made corrections if required.

5.6 Results & Discussion on AMISCO

I evaluated the experiments of the joint DOA and f_0 estimators in terms of recalls and root-mean-square errors by using a subset of the AMISCO's recordings. From a set of 24 speakers, I randomly selected one male speaker and one female speaker. The evaluation's results are listed in [50], and they are shown as cumulative distribution functions of recalls and root-mean-square errors. The author of [137] analyzed the laryngograph (electroglottograph) recordings focusing on gender differences and speaker identity for excitation signal synthesis—the synthesis of the vocal folds' movements. The author of [138] used parts of the corpus to evaluate the performance of differential microphone arrays for speaker localization and speaker separation.

During the recording and the post-processing, my colleagues and I encountered three problems. First, not being able to connect the camera and the Kinects to the word clock used for the audio recordings, we noticed varying delays between the starting-point of the audio and video recordings. To overcome this problem, a person clapped his hands once in the middle of the room at the beginning and the end of each sub-session. Captured by the audio and video devices, I was able to synchronize the audio recordings with the video data during post-processing by acoustically and visually aligning the moment of clapping in the audio and video tracks. Doing so for each recording, I realized that there was no significant rate drift between those two devices. Second, I had to split the 24 MEMS microphones into three groups due to the fixed number of eight microphone-connections on the microcontroller-boards. I knew that there will be clock-drifts and synchronization problems between the boards, because they were not connected to a central word clock (this was a hardware-restriction). Thus, I set up the 24-element circular array in a way that eight MEMS microphones connected to one board represent an 8-element circular array with constant angular interval of 45°; considering all three circular arrays, the interleaved 24-element circular array exhibits an interval of 15°. Third, due to an undetectable and unpredictable problem with the internal power supply of the laryngograph, speaker 1 exhibits distorted glottograms that should not be used. Speaker 24 doesn't include any skeleton tracks due to undetected communication problems between the Kinects and the computer during recording.

The corpus provides single-speaker recordings due to a limitation in equipment. For instance, I had access to a single laryngograph; as a consequence, I could simultaneously measure the laryngeal transconductance of one speaker only. Moreover, the Kinects solely tracked the skeleton of a single person.

To highlight a certain phenomenon unknown in the field of joint estimation and

mentioned in Chapter 4, I had to employ a different set of recordings that slightly differed from the recordings of the Austrian German Multi-Sensor corpus. I required recordings of a speaker, which looked exactly toward the center of the microphone array and which uttered voiced sounds with an invariant pitch, i.e., vowels or the sentences Why were you away a year, Roy?. However, the Austrian German speech corpus lacks such voiced utterances.

5.7 Conclusion

The Austrian German multi-sensor corpus (AMISCO) is a collection of two-room and 43-channel close- and distant-talking Austrian German high-quality speech recordings from 24 moving and non-moving speakers, balanced male and female. It contains around 8.2 hours of read speech, 53,000 word tokens based on 2,070 unique word types. Furthermore, this corpus features orthographic transcriptions, glottograms, fundamental frequencies, and positions and orientations of speakers located at certain positions or walking along pre-defined trajectories. The new corpus offers glottograms that can be used in prosody analysis, speech coding, speaker identification, as well as speech recognition. The synergy of all these components yields a unique and comprehensive corpus that can be used in several fields of research, e.g., signal processing, linguistic studies, or machine learning.

In comparison to all datasets listed in [126], the new corpus additionally features ground-truth data on fundamental frequencies. The authors of [126] reported that the perfect data set is out of reach when they aim at automatic speech recognition research using microphone arrays. However, they insufficiently described the features of a perfect data set. Thus, according to [126], it is unclear if the Austrian German Multi-Sensor corpus meets all criteria that define a perfect data set. For challenging experiments in the field of joint parameter estimation, it definitely is a valuable data set.

The website of the corpus [139] provides audio samples along with further information on the corpus. It provides information on how to obtain a copy of the corpus and scripts to extract the f_0 of the glottogram, and how to process the raw skeleton tracks (in case you want to apply different algorithms to these data). This website also provides the symbols for the orthographic transcriptions.

Chapter 6

Conclusion

To position my work in the field of joint DOA and f_0 estimation, I did a literature research with focus on doctoral theses. I thoroughly described four of the most inspiring theses in Section 1.2 and Section 1.3 to cast light on this field of research. The authors of these theses are Tania Habib, Jesper R. Jensen, Ted Kronvall, and Stefan I. Adalbjörnsson. This chapter consists of discussions of the aforementioned theses, a conclusion of my thesis, and an outlook for research of future PhD students in this area.

6.1 Discussion of Related Doctoral Theses

My belief in more efficient realizations of an algorithm regarding accuracy, computational resources, and the horizon of undreamt-of possibilities always inspired me to focus on the principles and properties of fundamental operations and basic filters first. Thus, after carefully analyzing Tania Habib's work [35, 47] I realized that the module that computed the spectro-temporal fragments was actually one of my major concerns. Though improving the accuracy in localizing speakers, I believed that this extension was redundant. In other words, overloading the direct predecessor with such a resource-consuming module was unnecessary. And this certain belief literally fired the starting pistol for a revealing journey in the field of joint parameter estimation extending. After analyzing Tania Habib's and her former colleague's publication about localization utilizing spatial and temporal information of a cross-correlation function [45] and thinking about how to improve these algorithms, I was able to introduce two new, innovative, and more powerful algorithms. For instance, elaborating on the fundamental principles of her algorithms [35, 47] and their corresponding predecessor [45], I figured out how to overcome their fundamental problems, e.g., the pitch-period doubling, the cross-correlation function's limitations (it emphasizes the features of the dominating source), the use of a biased cross-correlation function (it estimated harmonics with different amplitudes although they should be identical), the summation (marginalization) over frequencies that caused a loss of information in the frequency domain, the algorithms' decreasing frequency resolution towards higher frequencies, and the lack of a joint representation in a sparse joint parameter space. Moreover, her database presented in [47] was insufficient for evaluating my new algorithms because it lacked ground-truth frequency information. The database did not provide any information about the speakers' fundamental periods or fundamental frequencies, which can be determined by using a laryngograph that

records glottal activities and returns glottograms. This lack inspired me to set up an even more comprehensive corpus featuring additional data, e.g., glottograms of a speaker giving access to a speaker's fundamental periods during voiced parts of speech.

In case of the nonlinear least squares methods introduced by Jesper R. Jensen, I realized that employing the gradient search results in a decrease in accuracy. Without knowing anything about the parameters' ideal initial values and with a lack of information on how to update the step-size, it is difficult to achieve good results, even when I updated the step sizes by utilizing the Armijo-Goldstein condition—a line search method. Both issues are neither covered in his thesis nor in his papers. The problems are as follows: The algorithm often converges to local maxima unless the parameters' initial values are close to the global maximum. Since employing the Armijo-Goldstein condition improves the algorithms' behavior of convergence, I should obtain accurate estimates of the parameters. However, I realized that line search methods require a sufficiently high number of iterations, which was neither mentioned in the papers nor in the thesis. If this number is not high enough, the estimate in terms of the DOA might feature a significant error. Besides, there are many other parameters that have to be selected carefully when applying algorithms based on gradient search and line search. Nevertheless, all these issues triggered an avalanche full of questions; most of these questions were even related to my algorithms I was working on. I am glad that I found many answers. For instance, I realized that the signal model for the whole optimization procedure has to cover situations where a speaker's harmonics feature different DOAs at an instant of time in a reverberant environment; as shown in Fig. 4.16 and Fig. 4.20 in Chapter 4, the DOAs of a speaker's harmonics at a certain instant of time are rarely the same. These issues led me to the algorithms presented in my thesis. Thus, especially the inspiring work of Jesper R. Jensen positively influenced my work. Comparing his work with mine answers the question, which algorithm performs better in reverberant environments. His approach assumes a signal model where the DOAs of a speaker's harmonics are identical at a certain instant of time—it will fail in challenging reverberant environments—, whereas my approach also estimates the speaker's harmonics featuring different DOAs at a certain instant of time. Additionally, the comparison reveals that employing gradient search requires accurate prior knowledge to achieve the same accuracy as a grid search-based approach.

I soon realized that the theses of Stefan I. Adalbjörnsson and Ted Kronvall were strongly related to each other. Indeed, both were working in parallel at the same department. Although Stefan I. Adalbjörnsson finished almost one year before Ted Kronvall, it seemed to me that Ted Kronvall's intention was to build on the algorithms and findings of Stefan I. Adalbjörnsson. However, Ted Kronvall solely focused on applications related to audio signal processing, whereas Stefan I. Adalbjörnsson applied his algorithms to problems in different fields of research, e.g., spectroscopy. Interestingly, both obviously learned from the shortcomings of Jesper R. Jensen's proposed algorithms. I felt vindicated when I read that, regarding the a-/NLS algorithm, they rejected the idea of utilizing a gradient method due to the high number of local maxima; their statement was totally in line with my findings. Unlike Jesper R. Jensen, Stefan I. Adalbjörnsson approximated the non-linear model by a linear one employing convex relaxations, convex optimization, as well as the framework of alternating direction method of multipliers (ADMM). Stefan I. Adalbjörnsson and Ted Kronvall additionally focused on sparse mod-

els. In comparison to Jesper R. Jensen's algorithms, theirs rely on the prior knowledge of the highest possible model order, and some of them can cope with inharmonicities. I read Ted Kronvall's thesis after analyzing the outcomes of experiments with the non-linear least squares estimators proposed by Jesper R. Jensen. I was glad to see that Ted Kronvall listed several points regarding the NLS's drawbacks that coincided with my experiments' findings. He claimed that the NLS works poorly in practice because the resulting grid after determining the maximum arguments was highly multimodal, the optimization needs to be well initialized, i.e., the starting parameters have to be close to the global maximum, the evaluation grid must feature closely spaced grid points, and the sources' frequencies must be sufficiently separated in order to achieve good estimates. He also claimed that the global maximum is very sharp; however, I disagree because the size of the grid's Gaussian-kernel like global maximum—let's call it variance—depends on, e.g., the sensor spacing. Choosing a realistic spacing in hand-held devices causes the kernel's variance to increase. Regarding Ted Kronvall's thesis, I realized that he worked with real signals, too. However, these signals represented the spoken sentence "Why were you away a year, Roy", which contains voiced parts only. In his thesis, the algorithms' behavior in case of fricatives, unvoiced speech, and silence is missing. In contrast, I considered sentences containing all of these properties when I evaluated the VSS-based and RPDM-based approaches. However, in Chapter 4 I considered voiced speech only to highlight that a speaker's harmonics feature different DOAs at a certain instant of time in a challenging reverberant environment. In their experiments, they evaluated their algorithms with synthetically generated harmonic signals in the near field and the far field. In case of real signals, they placed two loudspeakers in a room playing back the sentence "Why were you away a year, Roy?". Unfortunately, there are no experiments with sentences containing voiced and unvoiced parts. Thus, the approaches behavior in case of natural speech is unclear. In many experiments, e.g., [27], the distance between the array and the source was relatively small, i.e., approximately 50 cm. Considering distances around 50 cm and large arrays, as utilized in his experiments, a near-field assumption is necessary for successful source localization. However, as shown in his thesis and in his publications, the spatial resolution towards larger distances becomes non-linearly smaller. Stated differently, the intervals between the grid points towards larger distances become larger. This yields a high spatial resolution in the array's vicinity, but a decreasing spatial resolution towards larger distances. Thus, the near-field property increases the algorithm's complexity and decreases the spatial resolution towards larger distances. In distant-speech enhancement, the distance between the arrays and the speakers are usually large: the arrays are mounted on the ceilings and the walls, whereas the speakers move, sit, or lie somewhere in the room. Thus, it is questionable if near-field assumptions are necessary. Due to these reasons I focused on far-field assumptions only. Besides, assuming near-field wave propagation requires knowledge about the distance between the array and the source. Relying on the relative attenuation of magnitude estimates (as done in Ted Kronvall's experiments) in real reverberant environments is questionable; more studies are required.

At this point, I would like to highlight the advantages of my proposed algorithms.

In contrast to Jesper R. Jensen's approaches, my approaches, first, do not need any prior knowledge about the model order. As in case of Ted Kronvall's and Stefan I. Adalbjörnsson's approaches, the model order has to be set to a maximum only in order

to limit the dictionaries or lookup tables.

Second, my approaches do not need information about the number of active speakers, which is a prerequisite in Jesper R. Jensen's approaches. However, in my approaches a maximum number of simultaneously active speakers must be set. This also applies to Ted Kronvall's and Stefan I. Adalbjörnsson's approaches.

Third, my approaches do not rely on sinusoidal models where the speaker's harmonics has to feature the same DOAs at an instant of time. I showed that the speaker's harmonics usually exhibits different DOAs in reverberant environments. Neither Jesper R. Jensen's approaches nor Ted Kronvall's and Stefan I. Adalbjörnsson's approaches handle such challenging situations.

Fourth, my estimators were successfully tested on signals with unvoiced speech, fricatives, and silence; they featured a robust behavior. In contrast, none of the aforementioned authors tested their algorithms on sentences containing voiced, unvoiced, and silent parts. In fact, they did not consider silence and unvoiced parts in their models. However, my approaches feature an amplitude threshold for all bands. It ensures that there are solely estimates during voiced parts in scenarios described in this thesis.

6.2 Conclusion

In this thesis, I started with two innovative algorithms which characterize and localize harmonic sources. The algorithms based on microphone arrays jointly estimate the sources' directions of arrival, fundamental frequencies, and their respective amplitudes based on a non-parametric signal representation in a sparse joint parameter space. The algorithms are purely deterministic and real-time capable. They neither rely on an explicit statistical estimator, any machine learning algorithms or data-driven methods, nor do they require any training material. The algorithms solve the issue of pitch-period doubling when using cross-correlation functions and cross-spectra. They estimate a source's harmonics even if they feature different directions of arrival at a certain instant of time, which is the case in reverberant environments.

Both algorithms span a sparse joint parameter space (which can be directly fed into a tracker) by applying a framework based on a filter bank and a fast and accurate multidimensional maxima detector. The first algorithm applies variable-scale sampling to cross-correlation functions. Focusing on the joint parameter space, it features invariant period intervals but nonlinearly increasing frequency intervals caused by the sampling-procedure in the lag domain. To overcome the problem of varying frequency intervals, I introduced the second algorithm, which employs a chirp z-transform and relative phase-delay masking. Moreover, a tolerance parameter (related to the direction of arrival) makes the new algorithm robust against small phase mismatches.

I conducted a vast number of Monte Carlo experiments with synthesized harmonic signals in free-field conditions and reverberant conditions as well as experiments with synthetically spatialized speech signals featuring different reverberation times and speech signals recorded in a reverberant environment.

Most of the speech signals were part of the introduced Austrian German multi-sensor corpus (AMISCO). It is a collection of two-room and 43-channel close-talking and distant-talking Austrian German high-quality speech recordings from 24 moving and non-moving single speakers, balanced male and female. It contains around 8.2 hours

of read speech, 53,000 word tokens based on 2,070 unique word types. Furthermore, this corpus features orthographic transcriptions, glottograms, fundamental frequencies, and positions and orientations of speakers located at certain positions or walking along pre-defined trajectories. The synergy of all these different components yields a unique and comprehensive corpus that can be used in several fields of research, e.g., linguistic studies, signal processing, or machine learning.

I even went one step further by applying multiple-target trackers to the estimates of my second and most advanced joint localization algorithm. To find the most appropriate tracker, I conducted experiments with the Gaussian mixture probability hypothesis density filter, the Gaussian mixture cardinalized probability hypothesis density filter, and the Gaussian mixture cardinality-balanced multi-target multi-Bernoulli filter. The Gaussian mixture cardinalized probability hypothesis density filter produce the smoothest spatio-temporal trajectories, whereas the Gaussian mixture cardinality-balanced multi-target multi-Bernoulli filter yields the smallest localization errors.

By using the root-mean-square error, the joint recall measure, and the cumulative distribution function of fundamental frequencies and directions of arrival, I determined the introduced algorithms' performances and compared them with performances of different algorithms. In case of multiple-target tracking, I employed the optimal subpattern assignment distance and its components: the localization error and the cardinality error.

Bringing it all together, the new algorithms are valuable contributions to the field of localizing and characterizing harmonic sources. They can improve the accuracy of a wake-up word or event detector, spatio-temporal filters, blind source separators, or they can decrease the word error rate of voice-controlled, distant-speech interacting systems. I proofed that the joint estimators' outcome can be directly fed into a multiple-target tracker yielding smooth spatio-temporal trajectories. Furthermore, I highlighted a phenomenon in real reverberant environments where the harmonics of a speaker feature different directions of arrival at a certain instant of time.

6.3 Outlook

At the very end of my doctoral program I still have many ideas and open questions in mind that need to be discussed. (Un-)fortunately, time is limited and things have to come to an end. Everyone has to trace new (unknown but exciting) paths, so do I. Tracing a new path means leaving the old one at a crossroads. This implies that there are several other paths that are connected to the crossroads; paths that can be explored by others. Therefore, I would like to share some of my ideas and open questions in the remaining part of this outlook for research of future PhD students in the field of joint parameter estimation. May the following ideas and questions be exciting paths-to-be-explored.

First, both introduced approaches are only applicable to distant-speech scenarios due to the far-field assumption made when determining DOAs. However, in close-talking scenarios I have to assume spherical wave propagation, i.e., near-field conditions. The publications of Ted Kronvall and Stefan I. Adalbjörnsson [24, 27, 42] are a good starting point in order to modify my two proposed algorithms to localize and characterize sources in the near field.

Second, I only focused on localization, characterization, and tracking. However, the next logical step is to feed a subband beamformer with the trajectories of a multiple-

target tracker. The subband beamformer can be, e.g., a cascade of a beamformer and a bandpass filter or a beamformer that considers bandpass filtering. The result should be a steered beam in spatial and frequency domain.

Third, after introducing a proper subband beamformer, a comprehensive evaluation of an overall system, i.e., a system consisting of a joint estimator, a tracker, and a subband beamformer, might yield new insights in that field of research. Someone might figure out how to optimize the overall system and/or how to improve the overall system's accuracy. An automatic speech recognizer can evaluate the system's accuracy.

Last but not least, there are some minor issues-to-be-examined. For instance, is the Kaiser window order-estimated bandpass filter the optimal filter or how can the tracker make use of the amplitudes estimated by the two approaches. In [106] there are hints on how to solve the latter issue.

Appendix A

Other Contributions

Besides working on localizing, characterizing, and tracking one or more harmonic sources, I contributed algorithms and findings in the field of beamforming.

A list of findings and evaluations based on experiments with different beamformers is presented in [140]

> Pessentheiner, H., Petrik, S., and Romsdorfer, H., "Beamforming Using Uniform Circular Arrays for Distant Speech Recognition in Reverberant Environments and Double Talk Scenarios," in *Proc. 13th Annual Conference of the International Speech Communication Association*, Portland, Oregon, USA, Sep. 2012, pp. 1368–1371.

This paper describes an adaptation of the most common state-of-the-art broadband beamformers to uniform circular arrays. The goal was to find the most robust system for distant speech recognition in double talk scenarios by attenuating competing speakers, enhancing the target speaker's signals, and applying a word recognizer and objective speech quality measures. As a result of this work, I presented a new beamformer.

In [141]

> Pessentheiner, H., Kubin, G., and Romsdorfer, H., "Improving Beamforming for Distant Speech Recognition in Reverberant Environments Using a Genetic Algorithm for Planar Array Synthesis," in *Proc. 10. ITG Conference on Speech Communication*, Braunschweig, Germany, Sep. 2012, pp. 1–4,

I highlighted a major disadvantage in beamforming when using a (horizontal) uniform circular array: a high sensitivity to reflections from the ceiling and the floor in a reverberant environment. Furthermore, I presented the so called CVX beamformer for the very first time. It is based on the convex optimization of a three-dimensional cost function considering three-dimensional constraints. To effectively attenuate interfering sources, I introduced a constraint that facilitates placing nulls in any direction.

In the following publications, colleagues and I applied the CVX beamformer to different scenarios, e.g., in reverberant and/or noisy environments with a pentagonal-shaped or star-shaped sensor array:

> Morales-Cordovilla, J. A., Pessentheiner, H., Hagmüller, M., Mowlaee, P., Pernkopf, F., and Kubin, G., "A German Distant Speech Recognizer based

on 3D Beamforming and Harmonic Missing Data Mask," in *Proc. 40th Italian (AIA) Annual Conference on Acoustics and 39th German Annual Conference on Acoustics (DAGA)*, Merano, Italy, Mar. 2013, pp. 2049–2052.

Morales-Cordovilla, J. A., Hagmüller, M., Pessentheiner, H., and Kubin, G., "Distant Speech Recognition in Reverberant Noisy Conditions Employing a Microphone Array," in *Proc. 22nd European Signal Processing Conference (EUSIPCO)*, Lisbon, Portugal, Sep. 2014, pp. 2380–2384.

Morales-Cordovilla, J. A., Pessentheiner, H., Hagmüller, M., González, J. A., and Kubin, G., "CVX-Optimized Beamforming and Vector Taylor Series Compensation with German ASR Employing Star-Shaped Microphone Array," in *Proc. Second International Conference, IberSPEECH 2014*, Las Palmas de Gran Canaria, Spain, Nov. 2014, pp. 148–157.

These publications [142–144] highlight the newly introduced beamformer's power and abilities by conducting experiments in the field of distant speech recognition.

Besides working on beamformers, colleagues and I also dealt with room localization for distant speech recognition, i.e., determining the speaker's room in a set of rooms connected via an open door [145]:

Morales-Cordovilla, J. A., Pessentheiner, H., Hagmüller, M., and Kubin, G., "Room Localization for Distant Speech Recognition," in *Proc. 15th Annual Conference of the International Speech Communication Association*, Singapore, Sep. 2014, pp. 2450–2453.

In 2013, colleagues and I participated in the 2nd CHiME challenge. As a result, we published a paper about single-channel speech separation and model-driven speech enhancement algorithms [146]:

Mowlaee, P., Morales-Cordovilla, J. A., Pernkopf, F., Pessentheiner, H., Hagmüller, M., and Kubin, G., "The 2nd CHiME Speech Separation and Recognition Challenge: Approaches on Single-Channel Speech Separation and Model-Driven Speech Enhancement," in *Proc. 2nd CHiME Speech Separation and Recognition Challenge, IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Vancouver, Canada, May 2013, pp. 59–64.

The 2nd CHiME challenge addressed the development of machine listening applications for operation in real-world multiple-source reverberant and noisy conditions. The goal was to recover the sound field's target speech signal.

AMISCO's preceding corpus is the GRASS corpus, i.e., the Graz Corpus of read and spontaneous speech presented in

Schuppler, B., Hagmüller, M., Morales-Cordovilla, J. A., and Pessentheiner, H., "GRASS: The Graz Corpus of Read and Spontaneous Speech," in *Proc. Ninth International Conference on Language Resources and Evaluation (LREC)*, Reykjavik, Iceland, May 2014, pp. 1465–1470.

Colleagues and I set up a comprehensive Austrian German corpus for, e.g., speech recognition. As described in [123], the corpus consists of three components. First, the conversation speech component contains free conversations. Second, the commands component contains commands and keywords. Third, the read speech component contains phonetically balanced sentences and digits.

Appendix B

Inter-/National Projects and Research Programs

During my doctoral program, I was working in several projects listed below, where I substantially contributed algorithms and ideas in the field of signal processing.

B.1 Advanced Audio Processing

The program named Advanced Audio Processing (AAP) [147] was a K-Project in the COMET program of the Austrian government. Co-sponsored by the provincial government and industrial partners, its aims were, among others, the cooperation between scientific and industrial partners. The project's developed core competencies in the field of acoustic multiple-input multiple-output systems and in the area of signal improvement and perceptual optimization. One of its goals was combining the development of sophisticated algorithms and the development of real-time solutions in the field of audio signal processing. The algorithms and systems were applied to the area of professional audio and communication technologies, automotive applications, and applications in the entertainment industry yielding new algorithms/systems for in-car-communications, dictation and teleconferencing, professional headphones and loudspeakers, and casino gaming machines.

B.2 Acoustic Sensing and Design

Acoustic Sensing and Design (ASD) [148] is a K-Project in the COMET program of the Austrian government, and it was the successor of the AAP program. It combined aspects of acoustic sensing and designing, e.g., combining different sensors, microphones and video cameras, in terms of sensor arrays and sensor networks for acoustic monitoring. The resulting algorithms and systems can be used in the field of acoustic intelligence for automotive applications and in the area of ambient audio for personal mobility and health.

B.3 Distant-Speech Interaction for Robust Home Applications

A European project named DIRHA [1] addressed the development of voice-enabled automated home environments on distant-speech interaction in different languages. The major goal of this project was the successful acoustic interaction between a human target speaker and an intelligent computer-operated apartment. A target speaker assigned tasks to the computer-operated apartment just by saying commands anywhere (without relying on close-talking or head-mounted microphones). The system monitored the environment by processing acoustic activities in different rooms in parallel by utilizing distributed microphone arrays and sophisticated signal-processing algorithms. Typical challenges were speaker localization, acoustic echo cancellation, speech enhancement, acoustic event segmentation and classification, speech understanding, dialogue management, and speech synthesis. As a result, my colleagues and I set up a prototype at Graz University of Technology.

B.4 Psychological Status Monitoring by Content Analysis and Acoustic-Phonetic Analysis of Crew Talks and Video Diaries

Psychological Status Monitoring by Content Analysis and Acoustic-Phonetic Analysis of Crew Talks and Video Diaries (short: CAPA) [149] is an international project related to a funding program by the European Space Agency (ESA) [150]. Its main goal is the phonological and content analysis of speech samples periodically recorded at Concordia Antarctic Research Station. Two circular microphone arrays installed above a dining table capture social conversations once a week. From a signal processing point of view, the major challenge is to separate and enhance each speaker's signals to provide high quality recordings with negligible acoustic interferences and noise. The main purpose of this international project is to study the psychological behavior of a crew working and living together in an isolated environment (the Antarctic research station). The findings are essential for future manned space flights.

Appendix C

Glossaries

C.1 List of Acronyms

AAP	Advanced Audio Processing
ASD	Advanced Sensing and Design
ADMM	Alternating Direction of Multipliers Optimization
AIA	Associazione Italiana di Acustica
APEBS	Array DOA and Pitch Estimation using Block Sparsity
AMISCO	Austrian German Multi-Sensor Corpus
ATHENA	Greek Multi-Sensory Database for Home Automation Control
AVG	Average
aNLS	Approximated Nonlinear Least-Squares
CAPA	Psychological Status Monitoring by Content Analysis and Acoustic-Phonetic Analysis of Crew Talks and Video Diaries
CBMeMBer	Cardinality-Balanced Multi-Target Multi-Bernoulli
CCF	Cross-Correlation Function
CDF	Cumulative Distribution Function
CHiME	International Workshop on Machine Listening in Multisource Environments
COMET	Compentence Centers for Excellent Technologies
CPHD	Cardinalized Probability Hypothesis Density
CPR	Cocktail Party Room
CVX	Convex (Optimized)
CZT	Chirp z-Transform
DAGA	Deutsche Arbeitsgemeinschaft für Akustik
DIRHA	Distant-speech Interaction for Robust Home Application
DFT	Discrete Fourier Transform

DOA	Direction of Arrival
EAP	Expected a Posteriori
ESA	European Space Agency
EUSIPCO	European Signal Processing Conference
FIR	Finite Impulse Response
FISST	Finite Set Statistics
FN	False Negative
GM-PHD	Gaussian Mixture Probability Hypothesis Density
GM-CPHD	Gaussian Mixture Cardinalized Probability Hypothesis Density
GM-CBMeMBer	Gaussian Mixture Cardinality-Balanced Multi-Target Multi-Bernoulli
GNU	GNU's not Unix!
GRASS	Graz Corpus of Read and Spontaneous Speech
HALO	Harmonic Audio Localization Using Block Sparsity
IEEE	Institute of Electrical and Electronics Engineers
IIR	Infinite Impulse Response
IPA	International Phonetic Alphabet
ITG	Informationstechnische Gesellschaft
JPS	Joint Parameter Space
LE	Little Endian
LREC	Language Resources and Evaluation Conference
MAP	Maximum a Posteriori
MeMBer	Multi-Target Multi-Bernoulli
MEMS	Microelectromechanical Systems
MMSE	Minimum Mean Square Error
MPOPI	Multiband Position-Pitch
MPOPI-FS	Frequency selection-based Multiband Position-Pitch
MPOPI-STF	Spectro-Temporal Fragment-based Multiband Position-Pitch
NLS	Nonlinear Least-Squares
OSPA	Optimal Subpattern Assignment
PCM	Pulse-Code Modulation
PEBS_2TV	Pitch Estimation using l_2-norm Block Sparsity Including Total Variation Penalty
PHD	Probability Hypothesis Density
PTDB-TUG	Pitch Tracking Database from Graz University of Technology
POPI	Position-Pitch
R	Recall
RAPT	Robust Algorithm for Pitch Tracking
RFS	Random Finite Set
RMSE	Root Mean Square Error
RPDM	Relative Phase-Delay Masking
SDK	Software Development Kit

SGL	Sparse Group Least Absolute Shrinkage and Selection Operator
SIR	Signal-to-Interference Ratio
SJPS	Sparse Joint Parameter Space
SNR	Signal-to-Noise Ratio
SPSC	Signal Processing and Speech Communication Laboratory
TCP/IP	Transmission Control Protocol and Internet Protocol
TDOA	Time-Difference of Arrival
TIMIT	Texas Instruments and Massachusetts Institute of Technology
TP	True Positive
UCA	Uniform Circular Array
USB	Universal Serial Bus
USS	United States Ship
VSS	Variable-Scale Sampling
YIN	Oriental Philosophy (Yin)

C.2 List of Symbols

Variables

A	Amplitude of a sinusoidal component
A_b	CZT's complex-valued starting point of contour in the z-plane
A_0	CZT's Radius of contour's starting point in the z-plane
C_j^l	Binomial coefficient
P_j^n	Permutation coefficient
T	Any period
T_0	Fundamental period
T_2	Length of a sweep in seconds
T_{60}	Reverberation time (60 dB)
$T_{60,c}$	Reverberation time (60 dB) in the cocktail party room
$T_{60,k}$	Reverberation time (60 dB) in the kitchen
T_s	Sampling Period
W_b	CZT's complex-valued parameter defining if contour spirals in or out in z-plane
W_0	CZT's spiral parameter
X	Random variable
Y	Random variable
a	Exponentiated scaling factor of the optimal subpattern assignment's cardinality error
c	Penalty assigned to labeling error
d_a	Microphone array's maximum dimension
$\hat{d}_{\min}$	Minimum distance ensuring plane wave propagation for all DOAs and f_0
$d_{\beta,k-1}$	Difference between spawned target and parent target
f	Frequency

f_0	Fundamental frequency
f_1	Start frequency of a sweep
f_2	End frequency of a sweep
f_k	Frequency
$\dot{f}_k$	Differentiated frequency
f_l	Lowest fundamental frequency of interest
f_u	Highest cut-off frequency
$f_{\min}^{(i_b)}$	Band's lowest fundamental frequency of interest
$f_{\max}^{(i_b)}$	Band's highest fundamental frequency of interest
f_s	Sampling frequency
l_{k-1}	Label
$l_{S,k\|k-1}$	Label of survived targets
$l_{\beta,k\|k-1}$	Label of spawned targets
$l_{P,k\|k-1}$	Label of survived targets
$l_{\gamma,k}$	Label of born targets
l_k	Label
$l_{L,k}$	Label of legacy track
$l_{U,k}$	Label of observation-corrected track
$\hat{l}_k$	Label of observed track
m_k	Means of posterior intensity's target states
$m_{U,k}$	Means of observation-corrected target states
$m_{\gamma,k}$	Means of born target states
m_{k-1}	Means of previous posterior intensity's target states
$m_{k\|k-1}$	Weights of predicted intensity's target states
$m_{\beta,k\|k-1}$	Means of previous target states
$m_{S,k\|k-1}$	Means of survived target states
n_s	Time shift in samples
p	Probability density
p_0	Initial density
$p_{\gamma,k}$	Probability density of born track
$p_{L,k}$	Probability density of legacy track
$p_{U,k}$	Probability density of observation-corrected track
$p_{S,k}$	Survival probability
$p_{D,k}$	Detection probability
p_{k-1}	Probability density of previous track
$p_{k\|k-1}$	Probability density of predicted track
$p_{P,k\|k-1}$	Probability density of survived track
p_β	Spawning probability
p_γ	Birth probability
q_k	Drawn sample of a normal distribution
r	Existence probability

$r_{L,k}$	Existence probability of legacy track
$r_{U,k}$	Existence probability of observation-corrected track
$r_{\gamma,k}$	Existence probability of born track
r_{k-1}	Existence probability of previous track
$r_{k\|k-1}$	Existence probability of predicted track
$r_{P,k\|k-1}$	Existence probability of survived track
$\|s\|$	Distance between array's center and source
t_k	Discrete-time support point
v	Speed of sound
v_t	Target velocity
w_k	Weights of posterior intensity's target states
w_{k-1}	Weights of previous posterior intensity's target states
$w_{k\|k-1}$	Weights of predicted intensity's target states
$w_{U,k}$	Weights of observation-corrected target states
$w_{S,k\|k-1}$	Weights of survived target states
$w_{\beta,k}$	Weights of spawned target states
$w_{\gamma,k}$	Weights of born target states
$w_{\beta,k\|k-1}$	Weights of previous target states weighted with weights of spawned target states
z	Point in z-plane
α	Amplitude of a harmonic component
β_{s,i_b}	CZT's normalized angular starting point on contour in z-plane
$\beta_{a,ib}$	CZT's normalized angular spacing between points on contour in z-plane
γ_f	Smoothing constant for falling signal edges
γ_i	Angle of incidence
γ_r	Smoothing constant for rising signal edges
δ	Kronecker delta ($\delta[\cdot]$) or Dirac delta ($\delta(\cdot)$)
$\delta^{(\text{init})}$	Step size of an adaptive filter
ε	Very small number
ε_φ	Robustness parameter (w.r.t. azimuth) for relative phase-delay masking
ε_ϑ	Robustness parameter (w.r.t. elevation) for relative phase-delay masking
ϵ	Recall-related variable for computing cumulative distribution functions
ϵ_s	Constant controlling maximum speed of increasing the estimated noise level
η_b	Base-distance order
η_o	Optimal subpattern assignment order
θ	Elevation
ϑ	Elevation

ϑ_{l_ϑ}	Elevation of direct-path or reflected component
λ_c	Clutter rate or average number of Poisson-distributed false alarms
λ_ω	Wavelength of interest
μ_s	Reference measure on subset of targets
μ_{SNR}	Threshold for computing noise power
ξ_e	Threshold for selecting states as final estimates
ξ_m	Threshold for merging states with small distance
ξ_p	Threshold for pruning states with low weights
ξ_r	Threshold for robustness parameter
ξ_v	Threshold for pruning states with high velocity
$\varrho_{U,k}$	Weighted and drawn sample of a normal distribution
σ_J	Standard deviation of a cardinality
σ_Q	Standard deviation of process noise
σ_R	Standard deviation of observation noise
$\sigma_{M,k}$	Variance of a cardinality distribution
τ	Time difference of arrival
ϕ	Azimuth
ϕ_n	Azimuth of noise source
ϕ_s	Azimuth of source
ϕ_t	Azimuth of target
φ	Azimuth
φ_k	Azimuth angle
$\dot{\varphi}_k$	Differentiated azimuth angle
φ_{l_φ}	Azimuth of direct-path or reflected component
$\phi_{i,j}$	Phase of a signal
χ	Parameter related to observation-corrected track
ψ	Ground-truth angle
ω	Angular frequency
ω_q	Angular frequency of the q-th harmonic
ω_{l_ω}	Harmonic or inharmonic component
Δf	Bandwidth
ΔN_h	Bandpass filter's group delay
$\Delta\varphi$	Angular step size

Operators

$\mathbb{E}$	Expectation operator
$\mathcal{H}$	System operator denoting a source's spatialization
$\lfloor\cdot\rceil$	Rounding operator (rounding to nearest integer)

Indices

i_b	Band index
i_g	Frame index
i_m	Microphone index

i_p	Index of a harmonic source
i_r	Index of an interfering noise source
i_s	Index of a set of grouped trajectories
i_x	Index of Cartesian coordinate (x)
i_y	Index of Cartesian coordinate (y)
j_x	Index of Cartesian coordinate (x)
j_y	Index of Cartesian coordinate (y)
k	Index of discrete Fourier transform
k	Time index of a state/observation (used as subscript)
l	Lag index of a cross-correlation function
i_φ	Index of azimuth angle of a direct-path or reflected component
i_ϑ	Index of elevation angle of a direct-path or reflected component
i_ω	Index of angular frequency of a harmonic or inharmonic components
m	Sample index of a windowed signal
n	Sample index of a windowed signal
n_t	Absolute time in samples or sample index

Numbers (Cardinalities)

$\widehat{M}$	Total number of target states in a scene
$M_{\mathrm{EAP},k}$	Expected a posteriori cardinality of the multi-Bernoulli posterior multiple-target density
$M_{\mathrm{MAP},k}$	Maximum a posteriori cardinality of the multi-Bernoulli posterior multiple-target density
$M_{\mathrm{MIN},k}$	Minimum number of tracks
M_{k-1}	Number of persistent tracks
$M_{k\|k-1}$	Number of legacy tracks
$M_{\mathbf{x}}$	Number of Bernoulli random finite sets
N	Arbitrary number of samples
N_a	Number of points lying on a z-plane's contour
N_b	Number of bands
N_c	Number of Monte Carlo experiments
N_d	Number of sampling points minus one divided by 2
N_e	Number of maxima
N_f	Number of frames
N_g	Number of pairs of microphones
N_h	Number of samples of a bandpass filter's impulse response
N_k	Number of observed states
N_{k-1}	Number of survived target states
$N_{k\|k-1}$	Number of predicted target states
N_m	Number of microphones
N_{M_k}	Number of ground-truth tracks
N_{N_k}	Number of estimated tracks

N_q	Number of harmonics
N_r	Number of interfering noise sources
N_s	Number of harmonic sources
N_T	Number of sampling periods
N_u	Number of utterances
N_v	Number of extended sampling periods
N_x	Number of samples of a microphone signal
$N_{\mathbf{x}}$	Number of states
$\hat{N}_{\mathbf{x}}$	Total number of targets in a scene
$N_{\mathbf{z}}$	Number of observations
$N_{\beta,k}$	Number of spawned target states
$N_{\gamma,k}$	Number of born target states
N_φ	Number of azimuth angles
N_θ	Number of elevation angles
N_{Φ}	Number of directions of arrival
N_{BPF}	Number of bandpass filters (equal to N_b)
N_{CSP}	Number of cross-spectrums
N_{CZT}	Number of chirp z-transforms (not their length)
N_{DFT}	Number of discrete Fourier transforms (not their length)
N_{IDFT}	Number of inverse discrete Fourier transforms (not their length)
N_{JPS}	Number of joint parameter spaces
N_{MAX}	Number of maxima detections
N_{RPD}	Number of relative phase delays
N_{RPDM}	Number of applied relative phase-delay masks
N_{SJPS}	Number of sparse joint parameter spaces
N_{SUM}	Number of summations
N_{VSS}	Number of variable-scale sampling procedures
N_{WGT}	Number of weightings

Distributions

$\mathcal{N}$	Normal distribution
$\mathcal{U}$	Uniform distribution

Vectors

$\mathbf{k}$	Spherical unit vector
$\mathbf{m}$	Microphone coordinates vector
$\mathbf{w}$	Tolerance vector of joint recall and root-mean-square error
$\mathbf{x}$	Target state vector
$\mathbf{y}$	Arbitrary state vector
$\mathbf{z}$	Observation vector

Matrices

0_{N_0}	Matrix containing $N_0 \times N_0$ zeros
F_{k-1}	Transition matrix

$F_{\beta,k-1}$	Transition matrix for spawning targets
H_k	Observation matrix
I_{N_0}	Identity matrix containing N_0 ones in the main diagonal
$P_{U,k}$	Covariance matrix of observation-corrected target states
P_{k-1}	Covariance matrix of previous posterior intensity's target states
$P_{\beta,k\|k-1}$	Covariance matrix of previous target states
$P_{k\|k-1}$	Covariance matrix of predicted intensity's target states
P_k	Covariance matrix of posterior intensity's target states
$P_{S,k\|k-1}$	Covariance matrix of survived target states
$P_{\gamma,k}$	Covariance matrix of born target states
Q_{k-1}	Process noise covariance matrix
$Q_{\beta,k-1}$	Process noise covariance matrix for spawning targets
R_k	Observation noise covariance matrix

Tuples

$L_{\Phi,T}$	Tuple consisting of sampling phase and sampling period
Θ	Tuple of ground-truth items
$\hat{\Theta}$	Tuple of estimated items
$\mathcal{L}_{k-1}$	Track table
$\mathcal{L}_{k\|k-1}$	Track table
$\mathcal{L}_k$	Track table
$\mathcal{T}$	Discrete-time support points
b	Permutation of a set of permutations

Sets

$B_{k\|k-1}(\mathbf{x}_{k-1})$	Random finite set of spawned states
G	Trajectory of a harmonic source
H	Trajectory of a harmonic source
I_c	Set of Cartesian coordinates' indices
I_k	Set of Cartesian coordinates' indices
I_x	Set of Cartesian coordinates' indices
I_y	Set of Cartesian coordinates' indices
I_{xy}	Set of tuples consisting of Cartesian coordinates' indices
J_{xy}	Set of tuples consisting of Cartesian coordinates' indices
K_k	Random finite set of clutter and spurious observations
L_T	Set of sampling periods
L_Φ	Set of sampling phases
$L_{\Phi,T}$	Set of sampling phases and sampling periods
$O_k(\mathbf{z}_k)$	Random finite set of observed states
$S_{k\|k-1}(\mathbf{x}_{k-1})$	Random finite set of survived states
V_G	Set containing the support points of a trajectory
V_H	Set containing the support points of a trajectory
$\mathfrak{W}_k$	Non-empty set of tracks

$\mathfrak{X}_k$	Set of ground-truth states
$\mathfrak{Y}_k$	Set of estimated states
$\mathfrak{Z}_k$	Set of tuples of states
$\mathcal{X}_k$	Set of all possible target states
$\mathcal{Z}_k$	Set of all possible observations
X_k	Finite set of states
Z_k	Finite set of observations
$\mathcal{F}(\mathcal{X}_k)$	Finite subset of states
$\mathcal{F}(\mathcal{Z}_k)$	Finite subset of observations
$\lvert X_k \rvert$	Set's cardinality of target states
$\lvert Z_k \rvert$	Set's cardinality of observations
$\mathcal{S}$	Region of the state space to be integrated
Γ_k	Random finite set of born states
Π	Set of permutations
$\emptyset$	Empty set
$\mathbb{X}_k$	Set of all tracks
$\mathbb{C}$	Set of complex numbers
$\mathbb{N}$	Set of natural numbers
$\mathbb{R}$	Set of real numbers
$\mathbb{Z}$	Set of integer numbers

Functions

$s[n_t]$	Harmonic signal
$\nu[n_t]$	Interfering noise source
$x[n_t]$	Signal captured by a microphone
$h[n]$	Impulse response
$c_{x_{i_1}x_{i_2}}[l]$	Cross-correlation function of two microphone signals
$c_{x_{i_1}x_{i_2},L}[l]$	Sampled cross-correlation function of two microphone signals
$c_{x_{i_1}x_{i_2},L}$	Sampled and summed cross-correlation function of two microphone signals
$C_{x_{i_1}x_{i_2}}[k]$	Cross-spectrum of two microphone signals
$\mathcal{F}\{\cdot\}$	Fourier transform
$\mathrm{CZT}\{\cdot\}$	Chirp z-transform
$X[k]$	Discrete Fourier transform of captured signal
$\tau_{i,j}(\varphi, \vartheta)$	Time-difference of arrival
$L_{\Phi}(\varphi, \vartheta)$	Sampling phase
$L_T(T_0)$	Sampling phase
$\mathrm{III}[l]$	Sampling function
$\mathrm{III}_L[l]$	Sampling function for a certain sampling phase and sampling period
$\mathrm{R}(\varphi, f_0)$	Joint recall
$\overline{\mathrm{R}}(\varphi, f_0)$	Joint recall averaged over all Monte Carlo simulations
$\mathrm{TP}(\varphi, f_0)$	True positive

$\mathrm{FN}(\varphi, f_0)$	False negative
$\mathrm{FP}(\varphi, f_0)$	False positive
$\mathrm{RMSE}(\hat{\Theta})$	Root mean square error of a parameter tuple
$\overline{\mathrm{RMSE}}(\hat{\Theta})$	Root mean square error of a parameter tuple averaged over all Monte Carlo simulations
$F_X(\cdot)$	Cumulative distribution function of random variable X
$F_Y(\cdot)$	Cumulative distribution function of random variable Y
$P(\cdot)$	Probability function
$X(\cdot)$	Transformed signal in z-domain
$\Xi(\phi, \theta, \omega_b[k])$	Binary relative phase-delay mask
$\widehat{\Xi}(\phi, \theta, \omega_b[k])$	Weighted relative phase-delay mask
$f_{k\|k-1}(\mathbf{x}_k\|\mathbf{x}_{k-1})$	Density function of translating a state
$g_k(\mathbf{z}_k\|\mathbf{x}_k)$	Density function of receiving an observation
$p_k(\mathbf{x}_k\|\mathbf{z}_{1:k})$	Posterior density of states
$p_{k\|k-1}(\mathbf{x}_k\|\mathbf{z}_{1:k-1})$	Prior distribution of states
$f_{k\|k-1}(X_k\|X_{k-1})$	Density function of translating sets of states
$g_k(Z_k\|X_k)$	Density function of receiving sets of observations
$p_k(X_k\|Z_{1:k})$	Posterior density of sets of states
$p_{k\|k-1}(X_k\|Z_{1:k-1})$	Prior distribution of sets of states
$p_0(\mathbf{x}_0)$	Initial distribution of states
$D_X(\mathbf{x})$	Nonnegative intensity function
$D_k(\mathbf{x})$	Posterior intensity function
$D_{k\|k-1}(\mathbf{x})$	Intensity function describing a translation
$D_{k-1}(\mathbf{x})$	Posterior intensity function
$D_{S,k\|k-1}(\mathbf{x})$	Intensity function describing surviving targets
$D_{\beta,k\|k-1}(\mathbf{x})$	Intensity function describing spawning targets
$D_{D,k}(\mathbf{x})$	Detection intensity function
$\kappa_k(\mathbf{z})$	Clutter intensity function
$c_k(\mathbf{z}_k)$	Spatial distribution of clutter
$\beta_{k\|k-1}(\mathbf{x})$	Spawning intensity function
$\gamma_k(\mathbf{x})$	Birth intensity function
$e_i(Z)$	Elementary symmetric function of order i
$p_{k-1}[n]$	Previous posterior cardinality distribution
$p_{k\|k-1}[n]$	Predicted cardinality distribution
$p_k[n]$	Posterior cardinality distribution
$p_{\Gamma,k}[n]$	Cardinality distribution of born targets
$\Upsilon_k^u[D_{k\|k-1}\|Z_k](n)$	Weighting function for posterior intensity and posterior cardinality distribution
$\Lambda_k(D, Z)$	Function describing the arguments of an elementary symmetric function
$\psi_{k,\mathbf{z}}(\cdot)$	Weighting function for posterior intensity and posterior cardinality distribution considering clutter

$\pi(\{\mathbf{x}_1, \ldots, \mathbf{x}_{N_k}\})$	Multi-Bernoulli random finite set's probability density
$\pi_{k-1}(\{\mathbf{x}_1, \ldots, \mathbf{x}_{N_{k-1}}\})$	Multi-Bernoulli random finite set's probability density
$\pi_{k\|k-1}(\{\mathbf{x}_1, \ldots, \mathbf{x}_{N_{k\|k-1}}\})$	Predicted multi-Bernoulli random finite set's probability density
$\pi_k(\{\mathbf{x}_1, \ldots, \mathbf{x}_{N_k}\}$	Posterior multi-Bernoulli random finite set's probability density
$\overline{\tau}(\varphi, \vartheta)$	Frequency-independent time difference of arrival
$\mathbf{1}_k$	Indicator function returning zero or one
$\mathcal{D}$	Optimal subpattern assignment distance
$\overline{\mathcal{D}}$	Averaged optimal subpattern assignment distance
$\mathcal{Q}$	Localization error
$\mathcal{R}$	Cardinality error
$\overline{\mathcal{Q}}$	Averaged localization error
$\overline{\mathcal{R}}$	Averaged cardinality error
P_b	Noise power
P_s	Signal power
γ_s	Function selecting smoothing constant for rising and falling signal edges
$w[l]$	Function describing the CCF's window
$d(G^{(i_g)}, H^{(i_g)})$	Function describing the distance in frequency domain between two trajectories
$d_{\min}(\gamma_i, \gamma_w)$	Function describing the minimum distance ensuring plane wave propagation
$\overline{h}_{i_b}[n]$	Band-dependent delay
$\phi_{i_1,i_2}^{(i_b)}[k]$	Cross-spectrum's phase
$p_{S,k}(\mathbf{x_{k-1}})$	Surviving probability of a state
$p_{D,k}(\mathbf{x_k})$	Detection probability of a state

Bibliography

[1] "DIRHA: Distant-speech Interaction for Robust Home Applications," Accessed on: Aug. 25, 2015, Fondazione Bruno Kessler (FBK), Povo, Trento, Italy. [Online]. Available: http://dirha.fbk.eu

[2] W. Täger and Y. Mahieux, "Reverberant Sound Field Analysis using a Microphone Array," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 1, Munich, Germany, Apr. 1997, pp. 383–386.

[3] M. Guillaume and Y. Grenier, "Sound Field Analysis with a Two-Dimensional Microphone Array," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 5, Toulouse, France, May 2006, pp. 321–324.

[4] D. de Vries and M. M. Boone, "Wave Field Synthesis and Analysis using Array Technology," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, Oct. 1999, pp. 15–18.

[5] F. Pinto and M. Vetterli, "Wave Field Coding in the Spacetime Frequency Domain," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Las Vegas, Nevada, USA, Mar. 2008, pp. 365–368.

[6] A. S. Bregman, *Auditory Scene Analysis: The Perceptual Organization of Sound.* Cambridge, MA, USA: MIT Press, May 1990, reprint: 09/29/1994.

[7] D. Wang and G. J. Brown, Eds., *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications.* Hoboken, NJ, USA: Wiley, Sep. 2006.

[8] H. Teutsch, *Modal Array Signal Processing: Principles and Applications of Acoustic Wavefield Decomposition.* Berlin, Germany: Springer, Jan. 2007.

[9] K. Imoto, Y. Ohishi, H. Uematsu, and H. Ohmuro, "Acoustic scene analysis based on latent acoustic topic and event allocation," in *Proc. IEEE International Workshop on Machine Learning for Signal Processing*, Southampton, U.K., Sep. 2013, pp. 1–6.

[10] H. Kwon, H. Krishnamoorthi, V. Berisha, and A. Spanias, "A sensor network for real-time acoustic scene analysis," in *Proc. IEEE International Symposium on Circuits and Systems*, Taipei, Taiwan, May 2009, pp. 169–172.

[11] A. de Cheveigné and M. Slama, "Acoustic Scene Analysis Based on Power Decomposition," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 5, Toulouse, France, May 2006, pp. 49–52.

[12] "SHINE: Speech-Acoustic Scene Analysis and Interpretation," Fondazione Bruno Kessler (FBK), Povo, Trento, Italy, Accessed on: Aug. 25, 2015. [Online]. Available: http://shine.fbk.eu

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

[14] B. Yegnanarayana, S. Prasanna, R. Duraiswami, and D. Zotkin, "Processing of reverberant speech for time-delay estimation," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 6, pp. 1110–1118, Nov. 2005.

[15] H. Christensen, N. Ma, S. N. Wrigley, and J. Barker, "Integrating pitch and localisation cues at a speech fragment level," in *Proc. 8th Annual Conference of the International Speech Communication Association*, Antwerp, Belgium, Aug. 2007, pp. 2769–2772.

[16] G. Liao, H. C. So, and P. C. Ching, "Joint time delay and frequency estimation of multiple sinusoids," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, Salt Lake City, UT, USA, May 2001, pp. 3121–3124.

[17] L. Y. Ngan, Y. Wu, H. C. So, and P. C. Ching, "Joint time delay and pitch estimation for speaker localization," in *Proc. IEEE International Symposium on Circuits and Systems*, vol. 3. Bangkok, Thailand: IEEE, May 2003, pp. 722–725.

[18] J. R. Jensen, M. G. Christensen, J. Benesty, and S. H. Jensen, "Joint Spatio-Temporal Filtering Methods for DOA and Fundamental Frequency Estimation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 1, p. 174, Jan. 2015.

[19] J. R. Jensen, J. K. Nielsen, M. G. Christensen, and S. H. Jensen, "On Frequency Domain Models for TDOA Estimation," in *Proc. 40th IEEE International Conference on Acoustics, Speech and Signal Processing*, Brisbane, Australia, Apr. 2015, pp. 11–15.

[20] Y. Wu, A. Leshem, J. R. Jensen, and G. Liao, "Joint Pitch and DOA Estimation Using the ESPRIT Method," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 1, pp. 32–45, Jan. 2015.

[21] J. R. Jensen, M. G. Christensen, and S. H. Jensen, "Joint DOA and Fundamental Frequency Estimation Methods based on 2-D Filtering," in *Proc. 18th European Signal Processing Conference*, Aalborg, Denmark, Aug. 2010, pp. 2091–2095.

[22] S. Karimian-Azari, J. R. Jensen, and M. G. Christensen, "Fast joint DOA and pitch estimation using a broadband MVDR beamformer," in *Proc. 21st European Signal Processing Conference*, Marrakech, Morocco, Sep. 2013, pp. 1–5.

[23] J. R. Jensen, M. G. Christensen, and S. H. Jensen, "Nonlinear Least Squares Methods for Joint DOA and Pitch Estimation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 923–933, Jan. 2013.

[24] T. Kronvall, S. I. Adalbjörnsson, and A. Jakobsson, "Joint DOA and multi-pitch estimation using block sparsity," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Florence, Italy, May 2014, pp. 3958–3962.

[25] S. N. Wrigley and G. J. Brown, "Recurrent timing neural networks for joint f0-localisation based speech separation ," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Honolulu, HI, USA, Apr. 2007, pp. 157–160.

[26] M. W. Hansen, J. R. Jensen, and M. G. Christensen, "Pitch and TDOA-based Localization of Acoustic Sources With Distributed Arrays," in *Proc. 40th IEEE International Conference on Acoustics, Speech and Signal Processing*, Brisbane, Australia, Apr. 2015, pp. 2664–2668.

[27] S. I. Adalbjörnsson, T. Kronvall, S. Burgess, K. Åström, and A. Jakobsson, "Sparse Localization of Harmonic Audio Sources," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 1, pp. 117–129, Jan. 2016.

[28] M. Képesi, L. Ottowitz, and T. Habib, "Joint Position-Pitch Estimation for Multiple Speaker Scenarios," in *Proc. Joint Workshop on Hands-Free Speech Communication and Microphone Arrays*, Trento, Italy, May 2008, pp. 85–88.

[29] T. Habib, M. Képesi, and L. Ottowitz, "Experimental evaluation of the joint position-pitch estimation (POPI) algorithm in noisy environments," in *Proc. 5th IEEE Sensor Array and Multichannel Signal Processing Workshop*, Darmstadt, Germany, Jul. 2008, pp. 369–372.

[30] T. Habib, L. Ottowitz, and M. Képesi, "Experimental evaluation of multi-band position-pitch estimation (m-popi) algorithm for multi-speaker localization," in *Proc. 9th Annual Conference of the International Speech Communication Association*, Brisbane, Australia, Sep. 2008, pp. 1317–1320.

[31] T. Habib and H. Romsdorfer, "Comparison of SRP-PHAT and Multiband-PoPi Algorithms for Speaker Localization Using Particle Filters," in *Proc. 13th International Conference on Digital Audio Effects*, Graz, Austria, Sep. 2010, pp. 1–6.

[32] ——, "Combining multiband joint position-pitch algorithm and particle filters for speaker localization," in *Proc. IEEE Workshop on Sensor Array and Multichannel Signal Processing*, Tel-Aviv, Israel, Oct. 2010, pp. 149–152.

[33] ——, "Concurrent speaker localization using multi-band position-pitch (M-POPI) algorithm with spectro-temporal pre-processing," in *Proc. 11th Annual Conference of the International Speech Communication Association*, Makuhari, Chiba, Japan, Sep. 2010, pp. 2774–2777.

[34] ——, "Improving Multiband Position-Pitch Algorithm for Localization and Tracking of Multiple Concurrent Speakers by Using a Frequency Selective Criterion," in *Proc. 12th Annual Conference of the International Speech Communication Association*, Florence, Italy, Aug. 2011, pp. 2897–2900.

[35] ——, "Auditory inspired methods for localization of multiple concurrent speakers," *Computer Speech & Language*, vol. 27, no. 3, pp. 634–659, May 2013.

[36] M. Wohlmayr and M. Képesi, "Joint Position-Pitch Extraction from Multichannel Audio," in *Proc. 8th Annual Conference of the International Speech Communication Association*, Antwerp, Belgium, Aug. 2007, pp. 1629–1632.

[37] L. Thurman and G. Welch, *Bodymind & Voice: Foundations of Voice Education, Revised Edition.* Chicago, IL, USA: The Voicecare Network, 2000.

[38] X. Alameda-Pineda and R. Horaud, "A Geometric Approach to Sound Source Localization from Time-Delay Estimates," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 6, pp. 1082–1095, Jun. 2014.

[39] J. Benesty, M. M. Sondhi, and Y. Huang, Eds., *Springer Handbook of Speech Processing*, 1st ed. Berlin, Germany: Springer, 2008.

[40] N. H. Fletcher and T. Rossing, *The Physics of Musical Instruments*, 2nd ed. New York: Springer, Jun. 1998.

[41] J. R. Jensen, "Enhancement of Periodic Signals: with Application to Speech Signals," Ph.D. dissertation, Aalborg University, Niels Jernes Vej 12, 9220 Aalborg st, Denmark, Aug. 2012.

[42] T. Kronvall, "Sparse Modeling of Grouped Line Spectra," Ph.D. dissertation, Lund University, Box 118, SE-221 00 Lund, Sweden, Jun. 2015.

[43] S. I. Adalbjörnsson, "Sparse Modeling Heuristics for Parameter Estimation: Applications in Statistical Signal Processing," Ph.D. dissertation, Lund University, Box 118, SE-221 00 Lund, Sweden, Jun. 2016.

[44] S. I. Adalbjörnsson, A. Jakobsson, and M. G. Christensen, "Multi-pitch estimation exploiting block sparsity," *Signal Processing*, vol. 109, pp. 236–247, Apr. 2015.

[45] M. Képesi, F. Pernkopf, and M. Wohlmayr, "Joint Position-Pitch Tracking for 2-Channel Audio ," in *Proc. International Workshop on Content-Based Multimedia Indexing*, Bordeaux, France, Jun. 2007, pp. 303–306.

[46] M. Képesi, M. Wohlmayr, and G. Kubin, "Joint position-pitch estimation of acoustic sources for their tracking and separation," Dec. 2008 (Accessed on: May 27, 2016), pub.: WO2008144784 A1, app.: PCT/AT2007/000265. [Online]. Available: http://www.google.com/patents/WO2008144784A1?cl=pt

[47] T. Habib, "Auditory Inspired Methods for Multiple Speaker Localization and Tracking Using a Circular Microphone Array," Ph.D. dissertation, Graz University of Technology, Rechbauerstrae 12, 8010 Graz, Austria, Jul. 2011.

[48] H. Christensen, N. Ma, S. N. Wrigley, and J. Barker, "A speech fragment approach to localising multiple speakers in reverberant environments," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Taipei, Taiwan, Apr. 2009, pp. 4593–4596.

[49] H. Pessentheiner, M. Hagmüller, and G. Kubin, "Localization and Characterization of Multiple Harmonic Sources," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 8, pp. 1348–1363, 2016.

[50] H. Pessentheiner, "AMISCO: The Austrian German Multi-Sensor Corpus," Graz University of Technology, Austria, 2015. [Online]. Available: https://www.spsc.tugraz.at/tools/amisco

[51] H. Pessentheiner, T. Pichler, and M. Hagmüller, "AMISCO: The Austrian German Multi-Sensor Corpus," in *Proc. 10th International Conference on Language Resources and Evaluation*. Portorož, Slovenia: European Language Resources Association, May 2016, pp. 760–766.

[52] J. Benesty, J. Chen, and Y. Huan, *Microphone Array Signal Processing*. Berlin, Germany: Springer, Mar. 2008.

[53] I. J. Tashev, *Sound Capture and Processing: Practical Approaches*. Chichester, U.K.: Wiley, Jul. 2009.

[54] A. Klapuri and M. Davy, *Signal Processing Methods for Music Transcription*. New York, NY, USA: Springer, Feb. 2007.

[55] W. M. Hartmann, "Pitch, periodicity, and auditory organization," *The Journal of the Acoustical Society of America*, vol. 100, no. 6, pp. 3491–3502, Dec. 1996.

[56] M. Stark, M. Wohlmayr, and F. Pernkopf, "Source–Filter-Based Single-Channel Speech Separation Using Pitch Information," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 2, pp. 242–255, Feb. 2011.

[57] W. Zhang and B. D. Rao, "A Two Microphone-Based Approach for Source Localization of Multiple Speech Sources," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 8, pp. 1913–1928, Nov. 2010.

[58] H. L. V. Trees, *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*. New York, NY, USA: Wiley, May 2002.

[59] J. G. Ryan, "Criterion for the minimum source distance at which plane-wave beamforming can be applied," *The Journal of the Acoustical Society of America*, vol. 104, no. 1, pp. 595–598, Jul. 1998.

[60] L. Rabiner, "On the Use of Autocorrelation Analysis for Pitch Detection," *IEEE Transactions on Acoustic, Speech, and Signal Processing*, vol. 25, no. 1, pp. 24–33, Feb. 1977.

[61] W. Hess, *Pitch Determination of Speech Signals: Algorithms and Devices*, 1st ed. Berlin, Germany: Springer, Jun. 1983.

[62] C. Roads, *The Computer Music Tutorial*. Cambridge, MA, USA: MIT Press, Feb. 1996.

[63] A. Jourjine, S. Rickard, and O. Yilmaz, "Blind Separation of Disjoint Orthogonal Signals: Demixing N Sources from 2 Mixtures," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 5, Istanbul, Turkey, Jun. 2000, pp. 2985–2988.

[64] A. Nadas, D. Nahamoo, and M. A. Picheny, "Speech Recognition using Noise-Adaptive Prototypes," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 10, pp. 1495–1503, Oct. 1989.

[65] S. T. Roweis, "One Microphone Source Separation," in *Advances in Neural Information Processing Systems.* Cambridge, MA, USA: MIT Press, 2000, vol. 13, pp. 793–799.

[66] D. Wang, "Time-Frequency Masking for Speech Separation and Its Potential for Hearing Aid Design," *Tends in Amplification*, vol. 12, no. 4, pp. 332–353, Dec. 2008.

[67] G. Weinreich, "Coupled piano strings," *Journal of the Acoustical Society of America*, vol. 62, no. 6, pp. 1474–1484, 1977.

[68] J. F. Kaiser, "Nonrecursive digital filter design using I_0-sinh window function," in *Proc. IEEE International Symposium on Circuits and Systems*, San Francisco, CA, USA, Apr. 1974, pp. 20–23.

[69] ——, "On the use of the I_0-sinh window for spectrum analysis," *IEEE Transactions on Acoustic, Speech, and Signal Processing*, vol. ASSP-28, no. 1, pp. 105–107, 1980.

[70] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, 3rd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, Aug. 2009.

[71] "MATLAB and Signal Processing Toolbox Release 2015a," Mathworks, Natick, MA, USA, Accessed on: May 31, 2016. [Online]. Available: https://se.mathworks.com/products/signal.html

[72] M. Slaney, "An Efficient Implementation of the Patterson-Holdsworth Auditory Filter Bank," Apple Computer, Inc., Cupertino, CA, USA., Tech. Rep. 35, 1993.

[73] M. D. Lutovac, D. V. Tošić, and B. L. Evans, *Filter Design for Signal Processing Using MATLAB and MATHEMATICA*, 1st ed. Englewood Cliffs, NJ, USA: Prentice-Hall, Sep. 2000.

[74] D. W. Ricker, *Echo Signal Processing.* Boston, MA, USA: Kluwer Academic Publishers, Feb. 2003.

[75] N. Wiener, "Generalized Harmonic Analysis," *Acta Mathematica*, vol. 55, pp. 117–258, 1930.

[76] J. O. Smith, *Mathematics of the Discrete Fourier Transform (DFT): with Audio Applications*, 2nd ed. North Charleston, SC, USA: BookSurge Publishing, Apr. 2007.

[77] J. A. Morales-Cordovilla, A. M. Peinado, V. Sanchez, and J. A. Gonzalez, "Feature Extraction Based on Pitch-Synchronous Averaging for Robust Speech Recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 3, pp. 640–651, mar 2011.

[78] R. Bracewell, *Fourier Analysis and Imaging*, 1st ed. New York, NY, USA: Springer, 2003.

[79] D. Lemire, "Streaming maximum-minimum filter using no more than three comparisons per element," *Nordic Journal of Computing*, vol. 13, no. 4, pp. 328–339, Dec. 2006.

[80] B. Luong, "The Min/Max Filter," Mathworks, Natick, MA, USA, Dec. Accessed on: Aug. 25, 2015. [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/24705-min-max-filter

[81] D. G. Zill and W. S. Wright, *Calculus: Early Transcendentals*, 4th ed. Sudbury, MA, USA: Jones & Bartlett, Apr. 2011.

[82] B. Ristic, B.-N. Vo, and D. Clark, "Performance Evaluation of Multi-Target Tracking Using the OSPA Metric," in *Proc. 13th Conference on Information Fusion*, Edinburgh, UK, Jul. 2010, pp. 1–7.

[83] B. Ristic, B.-N. Vo, D. Clark, and B.-T. Vo, "A Metric for Performance Evaluation of Multi-Target Tracking Algorithms," *IEEE Transactions on Signal Processing*, vol. 59, no. 7, pp. 3452–3457, Jul. 2011.

[84] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed., ser. Springer Series in Operations Research and Financial Engineering. New York, NY, USA: Springer, 2006.

[85] E. Hänsler and G. Schmidt, *Acoustic Echo and Noise Control: A Practical Approach*. Hoboken, NJ, USA: Wiley, Jun. 2004.

[86] H. Pessentheiner and G. Kubin, "Robust Source Localization and Characterization Based on Relative Phase-Delay Masking (submitted in June 2016)," *IEEE Transactions on Audio, Speech, and Language Processing*, 2017.

[87] L. I. Bluestein, "A Linear Filtering Approach to the Computation of Discrete Fourier Transform," *IEEE Transactions on Audio and Electroacoustics*, vol. 18, no. 4, pp. 451–455, Dec. 1970.

[88] L. R. Rabiner, R. W. Schafer, and C. M. Rader, "The Chirp z-Transform Algorithm and Its Application," *The Bell System Technical Journal*, vol. 48, no. 5, pp. 1249–1292, May-Jun 1969.

[89] S. A. Shilling, "A study on the Chirp Z-Transform and its Applications," Kansas State University, Manhattan, KS, USA, Tech. Rep., 1972.

[90] J. Kickliter, "Radio.jl: A Digital Communications Package for the Julia Language." 720 West Monument Street, Suite 100 Colorado Springs, CO, USA, Accessed on: May 31, 2016.

[91] M. Matassoni and P. Svaizer, "Efficient Time Delay Estimation based on Cross-Power Spectrum Phase," in *Proc. 14th European Signal Processing Conference*, Florence, Italy, Sep. 2006, pp. 1–5.

[92] E. A. Lehmann, "Image-Source Method for Room Impulse Response Simulation (Room Acoustics)," Mathworks, Natick, MA, USA, Mar. Accessed on: Dec. 28, 2015. [Online]. Available: http://www.mathworks.com/matlabcentral/profile/authors/512607-eric-a-lehmann

[93] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustic, Speech, and Signal Processing*, vol. 24, no. 4, pp. 320–327, Aug. 1976.

[94] M. Brandstein and D. Ward, Eds., *Microphone Arrays: Signal Processing Techniques and Applications*, 1st ed. Berlin, Heidelberg: Springer, 2001.

[95] R. P. S. Mahler, *Statistical Multisource-Multitarget Information Fusion.* Norwood, MA, USA: Artech House, Inc., Feb. 2007.

[96] ——, *Advances in Statistical Multisource-Multitarget Information Fusion.* Norwood, MA, USA: Artech House, Inc., Oct. 2014.

[97] B.-N. Vo and W.-K. Ma, "The Gaussian Mixture Probability Hypothesis Density Filter," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091–4104, Nov. 2006.

[98] B.-T. Vo, B.-N. Vo, and A. Cantatoni, "Analytic Implementations of the Cardinalized Probability Hypothesis Density Filter," *IEEE Transactions on Signal Processing*, vol. 55, no. 7, pp. 3553–3567, Jul. 2007.

[99] ——, "The Cardinality Balanced Multi-Target Multi-Bernoulli Filter and Its Implementations," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 409–423, Feb. 2009.

[100] D. Clark and B.-N. Vo, "The Random Set Filtering Website," Heriot-Watt University, Edinburgh, UK, University of Melbourne, AUS, Accessed on: Aug. 18, 2016. [Online]. Available: http://randomsets.eps.hw.ac.uk/disclaimers.html

[101] R. P. S. Mahler, "Multitarget Bayes filtering via first-order multitarget moments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, Oct. 2003.

[102] ——, "Random-set approach to data fusion," in *Proc. SPIE 2234 – Automatic Object Recognition IV, 287*, vol. 2234, Orlando, Fl, USA, Apr. 1994, pp. 287–295.

[103] I. R. Goodman, R. P. S. Mahler, and H. T. Nguyen, *Mathematics of Data Fusion.* Norwell, MA, USA: Kluwer Academic Publishers, Aug. 1997.

[104] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo Methods for Multi-Target Filtering with Random Finite Sets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1224–1245, Oct. 2005.

[105] B.-N. Vo and W.-K. Ma, "A Closed-Form Solution for the Probability Hypothesis Density Filter," in *Proc. 8th International Conference on Information Fusion*, Philadelphia, PA, USA, Jul. 2005, pp. 856–863.

[106] A. Masnadi-Shirazi and B. D. Rao, "An ICA-SCT-PHD Filter Approach for Tracking and Separation of Unknown Time-Varying Number of Sources," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 4, pp. 828–841, Apr. 2013.

[107] R. P. S. Mahler, "PHD filters of higher order in target number," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 4, pp. 1523–1543, Oct. 2007.

[108] Á. F. García-Fernández and B.-N. Vo, "Derivation of the PHD and CPHD Filters Based on Direct Kullback-Leibler Divergence Minimization," *IEEE Transactions on Signal Processing*, vol. 63, no. 21, pp. 5812–5820, Nov. 2015.

[109] B.-T. Vo, B.-N. Vo, and A. Cantoni, "The Cardinalized Probability Hypothesis Density Filter for Linear Gaussian Multi-Target Models," in *Proc. 40th Annual Conference on Information Sciences and Systems*, Princeton, NJ, USA, Mar. 2006, pp. 681–686.

[110] P. Borwein and T. Erdélyi, *Polynomials and Polynomial Inequalities*. New York, NY, USA: Springer-Verlag New York, Inc., Sep. 1995.

[111] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A Consistent Metric for Performance Evaluation of Multi-Object Filters," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, Aug. 2008.

[112] J. R. Hoffman and R. P. S. Mahler, "Multitarget miss distance via optimal assignment," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 34, no. 3, pp. 327–336, May 2008.

[113] E. Fridling, B., "Performance Evaluation Methods for Multiple Target Tracking Algorithms," in *Proc. SPIE 1481 – Signal and Data Processing of Small Targets*, vol. 1481, Orlando, FL, USA, Aug. 1991, pp. 371–383.

[114] S. Ziesemer, "Entwurf und Bau einer Variablen Akustik," Bachelor's Thesis, Graz University of Technology, Signal Processing and Speech Communication Laboratory, Inffeldgasse 16c/EG, 8010 Graz, Austria, 2016. [Online]. Available: https://www.spsc.tugraz.at

[115] P. Vary and R. Martin, *Digital Speech Transmission: Enhancement, Coding and Error Concealment*. Chichester: Wiley, Jan. 2006.

[116] M. Woelfel and J. McDonough, *Distant Speech Recognition*. Chichester: Wiley, Apr. 2009.

[117] A. de Cheveigné and H. Kawahara, "YIN, a Fundamental Frequency Estimator for Speech and Music," *The Journal of the Acoustical Society of America*, vol. 111, no. 4, pp. 1917–1930, Apr. 2002.

[118] D. Talkin, *A Robust Algorithm for Pitch Tracking (RAPT).* Amsterdam: Elsevier Science, Dec. 1995, pp. 495–518.

[119] M. Wohlmayr, M. Stark, and F. Pernkopf, "A Probabilistic Interaction Model for Multipitch Tracking with Factorial Hidden Markov Models," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 799–810, May 2011.

[120] A. Wrench, "A Multi-Channel/Multi-Speaker Articulatory Database for Continuous Speech Recognition Research," in *Proc. Workshop on Phonetics and Phonology in ASR*, Saarbrücken, Germany, Mar. 2000, pp. 1–14.

[121] F. Plante, G. F. Meyer, and W. A. Ainsworth, "A Pitch Extraction Reference Database," in *Proc. 4th European Conference on Speech Communication and Technology*, Madrid, Spain, Sep. 1995, pp. 837–840.

[122] G. Pirker, M. Wohlmayr, S. Petrik, and F. Pernkopf, "A Pitch Tracking Corpus with Evaluation on Multipitch Tracking Scenario," in *Proc. 12th Annual Conference of the International Speech Communication Association*, Florence, Italy, Aug. 2011, pp. 1509–1512.

[123] B. Schuppler, M. Hagmüller, J. A. Morales-Cordovilla, and H. Pessentheiner, "GRASS: The Graz Corpus of Read and Spontaneous Speech," in *Proc. 9th International Conference on Language Resources and Evaluation*, Reykjavik, Iceland, May 2014, pp. 1465–1470.

[124] A. Tsiami, I. Rodomagoulakis, P. Giannoulis, A. Katsamanis, G. Potamianos, and P. Maragos, "ATHENA: A Greek Multi-Sensory Database for Home Automation Control," in *Proc. 15th Annual Conference of the International Speech Communication Association*, Singapore, Sep. 2014, pp. 1608–1612.

[125] M. Matassoni, R. F. Astudillo, A. Katsamanis, and M. Ravanelli, "The DIRHA-GRID corpus: baseline and tools for multi-room distant speech recognition using distributed microphones," in *Proc. 15th Annual Conference of the International Speech Communication Association*, Singapore, Sep. 2014, pp. 1613–1617.

[126] J. Le Roux, E. Vincent, J. R. Hershey, and D. P. W. Ellis, "Micbots: Collecting large realistic datasets for speech and audio research using mobile robots," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Brisbane, Australia, Apr. 2015, pp. 5635–5639.

[127] M. D. Crawford, G. J. Brown, M. P. Cooke, and P. D. Green, "Design, Collection and Analysis of a Multi-Simultaneous-Speaker Corpus," in *Proc. The Institute of Acoustics 1994 Autumn Conference, Speech & Hearing, Windermere*, Windermere, England, UK, 1994, pp. 183–190.

[128] G. Lathoud, J.-M. Odobez, and D. Gatica-Perez, "AV16.3: An Audio-Visual Corpus for Speaker Localization and Tracking," in *Proc. First International Workshop on Machine Learning for Multimodal Interaction*, Martigny, Switzerland, Jun. 2004, pp. 182–195.

[129] A. Janin, D. Baron, J. Edwards, D. P. W. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters, "The ICSI Meeting Corpus," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, Hong Kong, Apr. 2003, pp. 364–367.

[130] J. S. Garofolo, C. D. Laprun, M. Michel, V. M. Stanford, and E. Tabassi, "The NIST Meeting Room Pilot Corpus," in *Proc. 4th International Conference on Language Resources and Evaluation*, Lisbon, Portugal, May 2004, pp. 1411–1414.

[131] D. Mostefa, N. Moreau, K. Choukri, G. Potamianos, S. M. Chu, A. Tyagi, J. R. Casas, J. Turmo, L. Cristoforetti, F. Tobia, A. Pnevmatikakis, V. Mylonakis, F. Talantzis, S. Burger, R. Stiefelhagen, K. Bernardin, and C. Rochet, "The chil audiovisual corpus for lecture and meeting analysis inside smart rooms," *Language Resources and Evaluation*, vol. 41, pp. 389–407, 2007.

[132] S. Renals, T. Hain, and H. Bourlard, "Interpretation of Multiparty Meetings: The AMI and AMIDA Projects," in *Proc. 2008 Hands-Free Speech Communication and Microphone Arrays*, Trento, Italy, May 2008, pp. 115–118.

[133] G. Gravier, G. Adda, N. Paulsson, M. Carré, A. Giraudel, and O. Galibert, "The ETAPE Corpus for the Evaluation of Speech-based TV Content Processing in the French Language," in *Proc. 8th International Conference on Language Resources and Evaluation*, Istanbul, Turkey, May 2012, pp. 114–118.

[134] S. Moosmüller, C. Schmid, and J. Brandstätter, "Standard Austrian German," *Journal of the International Phonetic Association*, vol. 45, no. 3, pp. 339–348, Dec. 2015.

[135] B. Schuppler, M. Adda-Decker, and J. A. Morales-Cordovilla, "Pronunciation variation in read and conversational Austrian German," in *Proc. 15th Annual Conference of the International Speech Communication Association*, Singapore, Sep. 2014, pp. 1453–1457.

[136] M. Hagmüller, L. Cristoforetti, and M. Omologo, "DIRHA-Deliverable 2.5-2.6: Design, Collection and Transcription of Real Acoustic Corpora and Text Data (DIRHA Corpora II)," Apr. 2015, Grant Agreement No. FP7-ICT-2011-7-288121. [Online]. Available: http://dirha.fbk.eu

[137] J. Ziegerhofer, "Excitation Signal Analysis: Gender Aspects," Master's Thesis, Graz University of Technology, Signal Processing and Speech Communication Laboratory, Inffeldgasse 16c/EG, 8010 Graz, Austria, 2016. [Online]. Available: https://www.spsc.tugraz.at

[138] T. C. Pichler, "Speaker Localization and Separation with Differential Microphone Arrays (working title)," Master's Thesis, Graz University of Technology, Signal Processing and Speech Communication Laboratory, Inffeldgasse 16c/EG, 8010 Graz, Styria, Austria, 2016. [Online]. Available: https://www.spsc.tugraz.at

[139] SPSC.tugraz.at, "AMISCO: The Austrian German Multi-Sensor Corpus," Accessed on: Oct. 13, 2015, Inffeldgasse 16c/EG, 8010 Graz, Styria, Austria, 2015. [Online]. Available: https://www.spsc.tugraz.at/tools/amisco

[140] H. Pessentheiner, S. Petrik, and H. Romsdorfer, "Beamforming Using Uniform Circular Arrays for Distant Speech Recognition in Reverberant Environments and Double Talk Scenarios," in *Proc. 13th Annual Conference of the International Speech Communication Association*, Portland, Oregon, USA, Sep. 2012, pp. 1368–1371.

[141] H. Pessentheiner, G. Kubin, and H. Romsdorfer, "Improving Beamforming for Distant Speech Recognition in Reverberant Environments Using a Genetic Algorithm for Planar Array Synthesis," in *Proc. 10. ITG Conference on Speech Communication*, Braunschweig, Germany, Sep. 2012, pp. 1–4.

[142] Morales-Cordovilla, J. A. and Pessentheiner, H. and Hagmüller, M. and Mowlaee, P. and Pernkopf, F. and Kubin, G., "A German Distant Speech Recognizer based on 3D Beamforming and Harmonic Missing Data Mask," in *Proc. 40th Italian (AIA) Annual Conference on Acoustics and 39th German Annual Conference on Acoustics (DAGA)*, Mar. 2013, pp. 2049–2052.

[143] Morales-Cordovilla, J. A. and Hagmüller, M. and Pessentheiner, H. and Kubin, G., "Distant Speech Recognition in Reverberant Noisy Conditions Employing a Microphone Array," in *Proc. 22nd European Signal Processing Conference*, Lisbon, Portugal, Sep. 2014, pp. 2380–2384.

[144] Morales-Cordovilla, J. A. and Pessentheiner, H. and Hagmüller, M. and González, J. A. and Kubin, G., "CVX-Optimized Beamforming and Vector Taylor Series Compensation with German ASR Employing Star-Shaped Microphone Array," in *Proc. Second International Conference, IberSPEECH 2014*, Las Palmas de Gran Canaria, Spain, Nov. 2014, pp. 148–157.

[145] Morales-Cordovilla, J. A. and Pessentheiner, H. and Hagmüller, M. and Kubin, G., "Room Localization for Distant Speech Recognition," in *Proc. 15th Annual Conference of the International Speech Communication Association*, Singapore, Sep. 2014, pp. 2450–2453.

[146] Mowlaee, P. and Morales-Cordovilla, J. A. and Pernkopf, F. and Pessentheiner, H. and Hagmüller, M. and Kubin, G., "The 2nd CHiME Speech Separation and Recognition Challenge: Approaches on Single-Channel Speech Separation and Model-Driven Speech Enhancement," in *Proc. 2nd CHiME Speech Separation and Recognition Challenge, IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Vancouver, Canada, May 2013, pp. 59–64.

[147] "AAP: Advanced Audio Processing," Accessed on: Jan. 03, 2017, Joanneum Research, Leonhardstraße 59, 8010 Graz, Austria. [Online]. Available: http://www.comet-aap.at/

[148] "ASD: Acoustic Sensing and Design," Accessed on: Jan. 03, 2017, Joanneum Research, Leonhardstraße 59, 8010 Graz, Austria. [Online]. Available: http://comet-asd.at/

[149] "CAPA: Psychological Status Monitoring by Content Analysis and Acoustic-Phonetic Analysis of Crew Talks and Video Diaries," Accessed on:

Jan. 03, 2017, Signal Processing and Speech Communication Laboratory, Graz University of Technology, Inffeldgasse 16c/EG, 8010 Graz, Austria. [Online]. Available: https://www.spsc.tugraz.at/research/projects/psychological-status-monitoring-content-analysis-and-acoustic-phonetic-analysis

[150] "CAPA: Psychological Status Monitoring by Content Analysis and Acoustic-Phonetic Analysis of Crew Talks and Video Diaries," Accessed on: Jan. 03, 2017, European Space Agency, Paris, Île-de-France, France. [Online]. Available: http://www.esa.int