Distributed Localization and Tracking of Acoustic Sources

Yuval Dorfan

Faculty of Engineering

Ph.D. Thesis

Submitted to the Senate of Bar-Ilan University

Ramat-Gan, Israel March 2018

This work was carried out under the supervision of:

Prof. Sharon Gannot

Faculty of Engineering, Bar-Ilan University, Israel

Acknowledgment

I wish to express my deep gratitude and appreciation to my supervisor Prof. Sharon Gannot for his outstanding dedicated guidance and supervision. Thank you for your professional support, for your encouragement to excellency, and for many valuable suggestions throughout all stages of this research. Your dedication to educate new researchers is unique and impressive.

Another highly experience person that helped me along the way is Dr. Gershon Hazan. Thank you for fruitful discussions and everything else you did for me. It has been my pleasure to work with you.

Thanks to my parents Miya and Mannie Dorfan for active involvement in the research beyond the general support (including lab recordings). Thanks to my supporting parents-in-law Marcelle and Acher Seroussi of blessed memory. From their perspective I should have already finished long ago (This sentence has been written when they were still alive. Now it has an additional deeper meaning). Special thanks to my beloved wife Iris and our four amazing children: Hof, Peleg, Gal and Li-Yam who encouraged and supported me through the whole way.

Contents

List of Figures

List of Tables

List of Algorithms

Notation

x	Scalar
$\boldsymbol{x}$	Column vector
x_i	The ith element of the vector $\boldsymbol{x}$
$\boldsymbol{A}$	Matrix
$\boldsymbol{A}^{-1}$	Matrix inverse
$(\cdot)^T$	Transpose operation
$(\cdot)^*$	Conjugate operation
$(\cdot)^H$	Transpose-conjugate operation
$\mathrm{diag}\{\boldsymbol{x}\}$	Diagonal matrix with the vector $\boldsymbol{x}$ on its diagonal
$(\cdot)^{\frac{1}{2}}$	For diagonal matrices, a diagonal matrix with the square root of the diagonal
$\\|\cdot\\|$	Euclidean norm operation
$\boldsymbol{I}$	Identity matrix
$\mathrm{E}\left(\cdot\right)$	Expectation operation
$x(\ell, k)$	Time-frequency coefficient

Abbreviations

AIR acoustic impulse response

AOA angle of arrival

AR auto regressive

ASA auditory scene analysis

ASfS affine structure from sound

ASN acoustic sensor network

ATF acoustic transfer function

AWGN additive white Gaussian noise

BDEM batch distributed expectation-maximization

BF beamformer

BM block matrix

B-MWF binaural multichannel Wiener filter

BOF bag of features

BSI blind system identification

BSS blind source separation

BW band width

CASA computational auditory scene analysis

CPU central processing unit

CREM Cappé and Moulines recursive expectation-maximization

CRLB Cramer-Rao Lower-Bound

CTF convolution transfer function

DAC digital to analog converter

DALAS distributed algorithm for localization and separation

DANSE distributed adaptive node-specific signal estimation

DAS delay and sum

DCT discrete cosine transform

DEM distributed expectation-maximization

DFT discrete Fourier transform

DNC diffuse noise coherence

DNN deep neural network

DOA direction of arrival

DPD direct positioning determination

DREM distributed recursive expectation-maximization

DRR direct-to-reverberant ratio

DSB delay and sum beamformer

DTF direct transfer function

DWSN distributed wireless sensor network

ECM expectation-conditional maximization

EKF extended Kalman filter

EKPF extended Kalman particle filter

EM expectation-maximization

ERB equivalent rectangular bandwidth

EVD eigenvalue decomposition

FA false alarm

FB-RDEM forward backward recursive distributed expectation maximization

FB-REM Forward-Backward Recursive Expectation-Maximization

FFT fast Fourier transform

FIM Fisher information matrix

FIR finite impulse response

GCC generalized cross correlation

GCC-PHAT generalized cross correlation with phase transform

GFCC gammatone frequency cepstral coefficient

GMM Gaussian mixture model

GRG generalized reduced gradient

GPS global positioning system

GSC generalized side-lobe canceler

GSCT generalized state coherence transform

HATS head and torso simulator

HMM hidden Markov model

ICA independent component analysis

IDEM incremental distributed expectation-maximization

IEM incremental expectation-maximization

i.i.d. independent identically distributed

IID interaural intensity difference

IIR infinite impulse response

ILD interaural level difference

INR interference to noise ratio

IS importance sampling

ISTFT inverse short-time Fourier transform

ITD interaural time difference

KDE kernel density estimate

KF Kalman filter

KLD Kullback-Leibler divergence

LACES localization and calibration EM sequence

LOS line of sight

LP linear prediction

LPC linear prediction coefficient

LPoG local projection of global

LS least squares

LSA log spectral amplitude

LSD log-spectral distance

LSO lateral superior olive

LTI linear time invariant

MAE mean absolute error

MAP maximum *a posteriori*

MC Monte Carlo

MCCC multichannel cross correlation coefficient

MD miss detection

MDS multidimensional scaling

MEMS micro electro-mechanic systems

MESSL model-based expectation-maximization source separation and localization

MFCC mel frequency cepstral coefficient

MIMO multiple input multiple output

MINT multichannel inverse theorem

MIPS mega instruction per second

ML maximum likelihood

MLE maximum likelihood estimator

MLS maximum length sequence

MMSE minimum mean square error

MoG mixture of Gaussians

MOS mean opinion score

MPT multiple pairwise TDOA

MSE mean square error

MSNR maximum signal to noise ratio

MSO medial superior olive

MSS minimal sufficient statistics

MTDOA maximum time difference of arrival

MTF multiplicative transfer function

MTT multi-target tracking

MVDR minimum variance distortion-less response

MUSIC multiple signal classification

MWF multichannel Wiener filter

NC noise canceler

NMF non-negative matrix factorization

NO-GSC non-orthogonal generalized side-lobe canceler

OMSLE optimal multiple source location estimation

PCA principal component analysis

p.d.f. probability density function

PESQ perceptual evaluation of speech quality

PF particle filter

PHAT phase transform

PHD probability hypothesis density

POAP peak over average position

PRA pair-wise relative absolute ratio

PRP pair-wise relative phase ratio

PSD power spectral density

QRD orthogonal triangular decomposition

RADAR radio detection and ranging

RANSAC random sampling consensus

RBF radial basis function

RBPF Rao-Blackwellised particle filtering

RBS reference-broadcast synchronization

RDEM recursive distributed expectation-maximization

RDTF relative direct transfer function

REM recursive expectation-maximization

RETF relative early transfer function

RF radio frequency

RIDEM recursive incremental distributed expectation-maximization

RIR room impulse response

RLS recursive least squares

RMS root mean square

RMSE root mean square error

RSEM random swap expectation-maximization

RTF relative transfer function

RV random variable

SAI stabilized auditory images

SAR synthetic aperture radar

SCT state coherence transform

SDR signal to distortion ratio

SDW speech distortion weighted

SF-GSC straightforward generalized side-lobe canceler (GSC)

SIMO single input multiple output

SIR signal to interference ratio

SLAM simultaneous localization and mapping

SLAT simultaneously localizing and tracking

SNR signal to noise ratio

SOC superior olivary complex

SOI signal of interest

SR success rate

SRD square root matrix decomposition

SRNR signal-to-reverberation plus noise ratio

SRP steered response power

SRP-PHAT steered response power-phase transform

SRR signal-to-reverberation ratio

SS sufficient statistics

STAR source to artifacts ratio

STEM significant tracking expectation-maximization

STFT short-time Fourier transform

SVD singular value decomposition

SVM support vector machine

TDOA time difference of arrival

T-F time-frequency

TF transfer function

TF-GSC transfer function generalized side-lobe canceler

TOA time of arrival

TOF time of flight

TREM Titterington recursive expectation-maximization

UBM universal background model

UKF unscented Kalman filter

VAD voice activity detector

WASN wireless acoustic sensor network

WER word error rate

w.p.1. with probability one

w.r.t. with respect to

WSN wireless sensor network

WSS wide-sense stationary

Abstract

Localization, separation and tracking of acoustic sources are ancient challenges that lots of animals and human beings are doing intuitively and sometimes with an impressive accuracy. Artificial methods have been developed for various applications and conditions. The majority of those methods are centralized, meaning that all signals are processed together to produce the estimation results.

The concept of distributed sensor networks is becoming more realistic as technology advances in the fields of nano-technology, micro electro-mechanic systems (MEMS) and communication. A distributed sensor network comprises scattered *nodes* which are autonomous, self-powered modules consisting of sensors, actuators and communication capabilities. A variety of layout and connectivity graphs are usually used. Distributed sensor networks have a broad range of applications, which can be categorized in ecology, military, environment monitoring, medical, security and surveillance.

In this dissertation we develop algorithms for distributed sensor networks with applications to speech processing, but some of the techniques can be applied also for other applications. Such wireless acoustic sensor networks (WASNs) can be found useful in lots of modern scenarios. The first example that can be dealt is ambient immerse communications. Nowadays, almost everyone carries 'his/her personal microphones' as part of the cellular phone, laptop computer or tablet. These spatially distributed sensors allow exploitation of spatial information in addition to spectro-temporal information. Spatial information in this context relates to location of active speakers and other acoustic sources. These sensors make the establishment of an *ad hoc* (distributed) microphones network feasible and allow the application of sophisticated signal extraction algorithms without the need to install expensive audio systems. A second example is smart homes that became very popular in the recent years. Intelligent networks of microphones are crucial components for control and monitoring systems as well as for communication in emergency cases. The last example to be mentioned

here is law enforcement. Authorities like the police or homeland security use eavesdropping and acoustic surveillance of public spaces as part of their regular procedure. This is usually done under adverse conditions.

The availability of only partial information in the nodes, the dynamics of the network, and the limited communication, connectivity and power capabilities call upon developing novel algorithms that address these challenges. The latter challenges are typical to distributed algorithms and cannot be found in classical array processing algorithms.

The contribution of the dissertation is fivefold. Firstly, distributed localization algorithms are derived using a novel set of hidden variables that are estimated by static or dynamic microphone arrays. It turns out that in addition to distributed computation, the new set of hidden variables improves the convergence speed and accuracy compared to previous approaches, since they enable the usage of incremental expectation-maximization (IEM) principle for the spatial domain. The distributed localization algorithms developed covers the batch EM and the on-line recursive expectation-maximization (REM).

Secondly, we developed a few localization techniques that reduces significantly the effect of reverberation on the performance. Processing that emphasizes the direct path is integrated with our modified localization algorithm and is shown to improve the performance especially when the number of concurrent speakers in the room is increased. In order to strengthen the node of two microphones, we have shown that instead of the pair-wise relative phase ratio (PRP), we can use the raw samples themselves with any known microphone geometry. Those samples can be processed within a new model that takes into account the late tail of the reverberation in addition to the direct path.

Thirdly, we have found out that the localization results of the mentioned algorithms can be utilized also for blind source separation (BSS). A major contribution for the separation algorithms is the hidden variables used for the EM mechanism. They were proven to be very efficient spectral masks, since their physical meaning is association of time-frequency bins to various speakers.

Fourthly, a major challenge with ad hoc networks is that the arrays locations are not known. We suggest a solution for joint calibration of the arrays and localization of the sources. They are all estimated relatively to an anchor array.

Finally, we address dynamic problems. Distributed tracking based on the recursive distributed expectation-maximization (RDEM) algorithm is described first for static arrays. Tracking multiple concurrent speakers is highly challenging, since the signals and the room

impulse responses (RIRs) are varying in a complex way. For example, the speakers do not utter speech continuously, but they might move continuously. It means that there are time gaps that need to be filled or extrapolated. A possible way to deal with those gaps is to utilize future information about the speakers. A short delay enables adding non-causal processing to the classical non-Bayesian tracking mechanism. Another problem (actually more realistic) is localization and tracking speakers using dynamic arrays of microphones. The movement of a microphones pair is utilized to localize and track speakers using Bayesian and non-Bayesian techniques.

Chapter 1

Introduction

Acoustic source localization, separation and tracking techniques, utilizing microphone arrays, have attracted the attention of many researchers for the last thirty years, especially in hands-free communication tasks (remote conference calls, smart lecture rooms, hearing aids, etc.). Typical problems in this field are: noise reduction, speaker separation and de-reverberation. The beamformer (BF) algorithms extend the dimension of the solutions and introduce spatial filtering in addition to the classical temporal-spectral filtering. Usually, the received speech signals are contaminated by interfering sources (such as competing speakers and noise) and also distorted by the reverberating environment. Whereas single microphone algorithms might show satisfactory results in noise reduction, they are not very-well suited in competing speaker mitigation task, as they lack the spatial information, or the statistical diversity used by multi-microphone algorithms. Despite the obvious advantages over single-microphone systems, traditional microphone arrays still suffer from severe performance limitations.

The relatively small aperture of conventional arrays is a limiting factor in the performance of spatial processing algorithms, since they only sample the sound field locally, typically at a relatively large distance from the target source(s). In these scenarios, low SNR and direct-to-reverberant ratio (DRR) are expected, resulting in deteriorated performance of regular small-aperture microphone arrays. Consequently, classical microphone-arrays cannot serve as a complete solution in the following example scenarios:

1. *Communication*: State of the art telecommunication systems attempt not only to faithfully convey the semantics of a conversation, but also to enable natural experiences and interactions among physically separated people, as if they share the same room. Many telephone conversations, especially those held in hands-free mode, are corrupted with

background noise, interfering signals and room reverberation.

2. *Smart houses*: A unified system for controlling all systems of the house (lights, air-condition, electronic devices such as television, sound system etc.) becomes popular. In this aspect, intelligent networks of microphones are crucial components for control and monitor systems as well as for communication in emergency cases.

3. *Smart classroom*: It is very important for the success and for the social lives of many children that have hearing disorder. Simple audio amplification provided by conventional hearing-aids enables better access to the auditory information, hence better detection, identification and recognition. However, amplification and even more sophisticated signal processing using only two arrays on the ears are insufficient. Many children with well-fitted hearing aids continue to have listening difficulties in a noisy reverberant environment such as the classroom. Background sounds, such as shuffling of papers, children talking and noisy ventilation systems around the child may create distractions and hinder the child's ability to distinguish speech from background sounds.

4. *Military and law enforcement*: Authorities, e.g., police and homeland security, use eavesdropping and acoustic surveillance of public spaces as part of their regular procedure. This is usually done under adverse conditions. The microphones should be deployed in a large area to ensure proper reception of the desired speakers.

A straightforward *centralized* approach is to place sensors in various locations and to convey all available data from them to a fusion center, where the processing is performed. Though optimal, this simple method requires transmission of huge amounts of data. Moreover, the aforementioned simple algorithm is sensitive to a failure of the fusion center, rendering the sensor network useless. Another disadvantage caused by the structure of the centralized solution, is the long communication link between sensors and the fusion center, which might be comprised of several hops (when the fusion center and the sensors cannot communicate directly) manifested as slow adaptation to the dynamically changing network or environment [1, 2].

Recent technological advances have made the vision of a distributed sensor network feasible. A wireless sensor network (WSN) comprises several nodes (or WSN modules) interconnected in some manner via a wireless medium. Each node consists of one or more sensors, a processing unit and a wireless communication module allowing them to exchange data.

The concept of the WSN is to split the system resources (sensors, processing units and actuators) among the nodes and to provide a scalable, fully covering the environment, easy to deploy, and robust structure. The wireless interface allows for the extension of the sensing range beyond the limits of the wired fusion center systems. The distribution of the sensors in a larger volume enables better coverage with higher SNR and DRR. For a survey on the topic of WSN please refer to [1, 3, 4, 5].

This work aims at developing distributed signal processing algorithms. The derived algorithms should maintain low complexity and communication band width (BW). However, performance issues should not be sacrificed (although, sub-optimal algorithms can be applied). Another requirement introduced by the nature of the problem at hand is the need for robust algorithms, which are not sensitive to failure of a few nodes, and which can handle a dynamically changing connectivity of the network. Naturally, most applications require algorithms that also adapt to changes in the environment, or to objects under observation.

Communication (especially in wireless networks) is often the most energy consuming operation of a node. In addition, it consumes BW resources, which are common to all electronic devices within a certain region. For ad hoc networks those resources might vary in time and space. An alternative signal processing paradigm to the *centralized* one is the *local* processing in which each node utilizes only its own measurement data, independent of other nodes, rendering wideband communication unnecessary. Although minimizing communication load, this method obviously imposes performance limitations, as only small subset of the data is used in each processing unit. Common systems utilize compression schemes for reducing the required bandwidth for conveying sensor data to the fusion center. Though straightforward, these methods do not consider the signal processing algorithm that takes place. Compression might destroy information that is relevant to a specific algorithm. Distributed algorithms aim at achieving the performance of the fusion center paradigm, while considerably reducing the required communication bandwidth. Each node performs a local calculation/filtering and distributes the results in the network.

Besides communication other issues should be addressed when using ad hoc networks. Firstly, in classical methods the geometry of the array is usually pre-determined to fit the problem at hand. However, in some sensor network scenarios it is not possible to determine the array's layout. Secondly, designing distributed algorithms for WSN necessitates the application of additional considerations. The computational power available at the nodes is limited. Furthermore, as each node uses independent clock, sampling frequency offsets between nodes are inevitable. The latter results in non-equal sampling rates in the network, and eventually performance degradation. These effects will be taken into account in our algorithms.

The main issue of this research is finding and tracking the locations of speakers in a room using ad hoc networks. Speaker localization and tracking algorithms have various civil [6] and military [7] applications, e.g. surveillance systems, camera steering in conference calls, BF, dereverberation, BSS, noise reduction, etc. This survey is organized in the following way. We start this chapter with an overview on localization with emphasis on distributed computations in Section 1.1. Section 1.2 deals with mitigation of reverberation in the context of localization and tracking algorithms. Byproducts of our algorithms enable distributed BSS. Other BSS techniques are mentioned in Section 1.3. A major challenge for ad hoc networks is calibration. In Section 1.4 we give some background about joint calibration and localization. In Section 1.5 we describe the work already done in the field of source tracking (centralized and distributed versions). We focus on tracking of concurrent sources. Section 1.6 introduces the structure of the rest of this dissertation. In Section 1.7 we list all our publications that are relevant for this work.

1.1 Distributed localization of multiple sources

Localization has been dealt with for years in the acoustic field. For example in [8] TDOA values are utilized for intersecting loci. Localization techniques can be divided into two groups: Bayesian [9, 10, 11] or non-Bayesian [12, 13] approaches. With a non-Bayesian technique, the maximum likelihood (ML) is widely used for localization. ML estimation procedures for localization are usually characterized by high computational complexity and by the nonexistence of closed-form solutions. It was therefore decided to apply either iterative EM [14, 15] or REM procedures. The challenge of distributed localization is discussed in Chapter 3.

1.1.1 Static arrays

State-of-the-art methods operate in the time-frequency domain and rely on narrowband approximation of the convolutive mixing process by complex-valued multiplication in each frequency bin. The sparsity of speech signals in the short-time Fourier transform (STFT) domain is widely used in the context of speaker localization [16, 17, 18]. In [16] for example, the localization task was carried out by using spatially distributed (located in various positions around the room) microphone nodes (more specifically, each node was comprised of a pair of microphones). Accurate location estimates can only be obtained if the microphones are spatially distributed.

Recent technological advances have made microphones cheaper and smaller. This results in multiple microphones integrated into many electronic devices (televisions, cell phones, hearing aids, computers, toys, etc.) [19]. Sometimes, due to limited computational resources in each node and the BW constraint on the communication link connecting the nodes, *distributed computation* is needed. Distributed networks can be utilized to jointly estimate parameters [20, 21] or signals by applying BF techniques [22, 23].

The concept of a WSN [24] can be naturally adapted to the speaker localization problem. In principle, algorithms that are tailored to WSNs are characterized by *local* processing of the raw data, while only compact, intermediate results are communicated through the network to perform the *global* task. Many of the drawbacks attributed to centralized processing can be circumvented by distributed computations [25]. In the acoustic community, the term WSN is replaced by WASN [22]. The WASNs were mainly used in speech enhancement and speaker separation problems [22, 23]. In [23], the sidelobe canceler over WASN has been dealt with.

Centralized localization solutions that can be mentioned include [26, 27, 28]. Hierarchical methods compute partial results in the nodes before combining them. In several hierarchal localization methods, the nodes provide DOAs estimates that are later combined by triangulation at a central processing point [29, 30]. In contrast, distributed algorithms carry out most of the calculations (or even all of them) in the nodes of the network.

Distributed localization algorithms for WASN that only use the amplitude or power of the received signals, but ignore phase information, can be found in [31, 32, 33, 34, 35, 36, 37]. It is well-known, from the radio frequency (RF) literature, that localization schemes based on phase differences exhibit higher accuracy, robustness and lower sensitivity than schemes that are solely based on the received signal strength [38].

An EM-based method for angle of arrival (AOA) estimation of multiple sources that uses phase information extracted from stereo recordings was proposed in [39, 40, 41]. Acoustic measurements by a spatially condensed array can only estimate the angle to the source [42, 43, 44, 45, 46]. In order to compute the sources' two- or three-dimensional coordinates, sensors have to be spatially distributed over a wider area.

Extension to a two dimensional localization and tracking problems by an array of microphone pairs can be found in [16]. The authors have defined a set of parameters and a set of hidden variables and developed algorithms based on the EM and two tracking variants of the REM technique. A detailed mathematical description of the application of the EM to mixture of Gaussians (MoG) can be found in [47]. In [16] the Guassians were centered around grid points around the room. The number of sources and their rough locations were assumed to be known a priori.

Both [16] and [39] used *centralized computation* approaches. The distributed expectation-maximization (DEM) usage was presented in [48] for clustering stochastic variables using a MoG model. [49] also presented a DEM algorithm for MoG in a slightly different way. In [50], three distributed strategies were presented for the MoG EM (incremental, consensus and diffusion). The general idea in this kind of algorithms is that the hidden variables of the EM are local and that the majority of the calculations is applied in the nodes (locally) in a way that enables sharing a compact set of results between the nodes. This set is often called sufficient statistics (SS). We adapt those principles to our algorithms for localization, tracking and source separation.

An incremental variant of the EM, denoted IEM, was suggested by Neal and Hinton [51]. They proved that the hidden variables can be estimated incrementally rather than in a single batch step. The classical EM steps are replaced by partial steps. One version is to run partial M-step followed by the regular E-step. The second version, which we adapt here applies a partial E-step and a regular M-step.

The convergence speed and accuracy of IEM was analyzed and demonstrated in [52, 53]. It was shown through many examples that the incremental strategy enables faster convergence. In addition, the IEM has a higher probability to converge to the ML. In other words, it does not tend to converge to a local optimum. There is no mathematical proof for these properties, but they are explained intuitively and demonstrated empirically.

They also dealt with the REM mechanism, which contains almost the same idea in time axis. We adapt here the Titterington recursive expectation-maximization (TREM) algorithm [54], which is based on a Newton search for the maximization of the likelihood. We adapt this idea both for our localization algorithms and for the tracking.

1.1.2 Localization using moving arrays

Localization using static sensors has been dealt with theoretically and practically for various signal processing applications including passive or active radio detection and ranging (RADAR). Passive sonar using moving hydrophones has been dealt for years [55]. In particular, approaches for acoustic sensors deal with the specific challenges of reverberation, see, e.g., [16, 56, 57, 58]. In general, most sound source localization approaches in the literature utilize either time of arrival (TOA) [59], TDOAs or DOAs as measurements of the source in order to reconstruct the Cartesian source position. A common assumption is that the acoustic sensor is static and that its position is known. Nonetheless, spatial diversity of an acoustic sensor installed on a moving platform could be exploited for improved inference of the source position. Moving microphone arrays are particularly suitable for the field of robot audition [60, 61, 62, 63], where microphone arrays can be installed in the limbs and head of an autonomous robot.

The movement of microphone arrays is particularly useful in situations where the sensor moves faster than the sources. In this case, the displacement of the sensors over time can be interpreted as a synthetic widening of the array aperture in space. This interpretation was first implemented for synthetic aperture radar (SAR) [64]. We implement this principle for localizing the coordinates of a source with a single pair of microphones.

At the end of Chapter 3 we address the challenge of sound source localization from a moving platform by considering and comparing two philosophically different approaches.

1.2 Reverberation mitigation for localization

Many localization algorithms are highly sensitive to reverberation. The challenge of localizing a number of concurrent acoustic sources in reverberant enclosures is addressed in Chapter 4.

1.2.1 Localization in highly reverberant environment

A vast number of algorithms for localization of sources employing multiple concurrent sensor measurements have been developed [65, 66]. Important distinguishing aspects between methods are the sensor configuration, assumptions on the signal propagation, measurement type and probabilistic modeling.

Some approaches deal with the localization [67] or tracking [68] of a single source. In [68], Bayesian estimation based on a Kalman filter, applied to tracking, was described. Distributed algorithms for multiple sources were already mentioned in the previous section. Their performance highly depends on the reverberation level.

Acoustic signals often suffer from high level of reverberation that hamper localization accuracy and sometimes also from noise of various types. The main obstacles that hamper localization and DOA estimation accuracy are reverberation level, diffused noise or sensor noise [69, 70, 71]. The next distinguishing criterion for localization algorithms is the assumption regarding strong unimpeded direct path signals. As many algorithms rely on a strong signal from the direct path, reverberations degrade the localization performance, especially in indoor scenarios [72]. This problem becomes more severe when the number of speakers increases, since each speaker produces additional reflections. Several approaches are applicable without relying on the direct path from the target to the sensor [73]. In the method proposed in [74], direct-path dominance is not mandatory and various room shapes are supported. Methods that utilize the multi-path signals are limited by a set of assumptions about the room and might be more sensitive to changes in room characteristics [75, 76, 77, 78, 79]. The set of techniques vary from un-supervised through semi-supervised to fully supervised. Our proposed approach is un-supervised, which uses multiple distributed sensor nodes and assumes that the direct path is dominant in at least part of them.

Next, we may also distinguish between algorithms according to the type of measurements used. A large number of acoustic localization and direction finding methods employ correlation between the microphone signals in order to determine the TDOA. If a node is equipped with more than a single microphone, the DOA can be estimated from TDOA values. Some approaches use full band measurements and some use subbands [80] or different frequencies.

The simplest estimator for finding the TDOA between two observed signals is the cross-correlation method and its variants, mainly the generalized cross correlation (GCC) [81] with phase transform (PHAT) normalization. In [82, 83], a generalization of the GCC-PHAT to an array of microphones in far-field scenarios is proposed. Performance degradation is often demonstrated in multiple speakers and reverberant environments. The challenge of multiple speaker localization using generalized cross correlation with phase transform (GCC-PHAT) is presented in [84]. The separation of concurrent speakers in time domain is limited when there are significant energy differences and when their TDOAs are too close. Another way to apply GCC-PHAT for multiple sources is given in [85] for DOA estimation. A plethora of methods are based on the GCC-PHAT [86] or the steered response power-phase transform

(SRP-PHAT) [27, 87], which are full-band TDOA estimation algorithms. Some algorithms compute subband TDOAs in order to improve the robustness and to facilitate concurrent speaker direction finding [88, 89].

Biologically inspired approaches use multiple frequency bands as well [90, 91]. The benefit of applying modeling of the sound processing in the human cochlea and mid-brain [92, 93] stems from the ability to include a number of mechanisms that render the estimation robust. The approaches summarized in [93] focus on the binaural case including GCC based methods. They present direction finding by subband GCC. This includes modeling of the precedence effect, which focuses on the first wavefronts from the direct path [94].

One of the consequences of the precedence effect is the neural onset dominance [95], which can be implemented in mono channel processing [57]. Methods that use phase locked spikes centered on the maxima in certain frequency bands [96] can be advantageous [57] as compared to the more common zero crossing methods [92, 90].

More generally, the fact that only high SNR segments are used [97] can be exploited in tracking applications [98]. These principles were shown to add robustness in tracking with multiple microphone arrays [99] and were combined with auditory scene analysis (ASA)-oriented processing for application to distributed nodes [30].

The next distinguishing criterion between localization algorithms refers to the probabilistic modeling used. Many algorithms employ the ML criterion, which is often implemented by applying the EM procedure, to localize sources. This procedure is adopted for estimating the DOA [100], or the source position after triangulation [28], by utilizing the measured TDOAs. The EM often faces a convergence problem. The random swap expectation-maximization (RSEM), based on MoG distribution [101], reduces the dependency of the EM algorithm on initial conditions.

In contrast to the methods that use a global MoG [47, 16], distributed localization versions will be suggested in Chapter 3. We define there multiple local MoGs that share a set of weights, referred to as the global parameters. Although exhibiting good performance in a wide range of acoustic conditions, high reverberation still degrades the performance. This may be attributed to the influence of multiple secondary reflections on the phase information embedded in the direct-path of the acoustic propagation [39, 16] or in the respective TDOA measurements.

1.2.2 Single array direction of arrival estimation

An estimate of the speakers' DOAs is required in many applications, including navigation, surveillance, beamforming, source separation, target acquisition and tracking. In reverberant environments, sound reflections may result in erroneous DOA estimates. The existence of multiple speakers in the reverberant environment, may even further degrade the accuracy of the DOA estimates.

In [102], a maximum likelihood estimator (MLE) of multiple DOAs in noisy environments was derived. The speech and the noise components were described as zero-mean Gaussian vectors, with rank-1 power spectral density (PSD) and full-rank PSD matrices, respectively. A closed-form MLE for the speech and noise PSDs was derived. The MLE for the DOAs was obtained by a Newton search. The sparsity assumption of the speech was not utilized. In [103], the de-reverberation task for hearing aids was addressed. The reverberation was modeled as a diffuse field with time-varying level.

1.3 Single and multiple arrays blind source separation

Localization results are often used for source separation algorithms. Within this scope, we are also interested both in a basic array (a pair), arrays with more than two microphones and in multiple arrays approaches. The challenge of blind source separation in different scenarios is discussed in Chapter 5.

1.3.1 Separation based on distributed and centralized arrays of microphones

The BSS is an unsupervised technique for recovering the underlying signals from a set of their combinations. In acoustic applications [104], as a cocktail party problem [105, 106], the sources (speakers) are typically mixed in a convolution manner. The respective source separation is referred to as a convolution BSS or de-convolution.

The convolution BSS problem is much more challenging compared with the instantaneous one, since the separation filter will have thousands of coefficients in a typical room environment. The instantaneous BSS framework still can be used for convolution mixture separation in the frequency domain [107]. However, once this domain is used, the inherent scaling and permutation ambiguity of BSS methods [108] will appear in each frequency band and have to be resolved for obtaining meaningful separation results.

Additional difficulty may arise for audio applications in the under-determined case (the number of sources is greater than that of the sensors). In this case, the linear separation scheme will fail to separate the sources. However, this scenario is still tractable if the signals have a sparse representation [109]. Audio sources (such as speech and music) are often attributed by a sparse representation in the STFT domain [110]. The sparseness of the speech attracted significant attention in the signal processing community in general, and in source separation in particular [111, 112, 113]. Separation in the time-frequency (T-F) domain is achieved by clustering the T-F bins into groups, while each group is associated with a source signal. The clustering usually relies on features such as TDOA [110, 41, 114]. In [115] the BSS problem is solved using GCC-PHAT for a single array of microphones. Distributed source separation techniques can be found in [22, 23]. However, for impulse responses estimation, dedicated portions of the signals are needed. We will refer to this issue in our research about distributed BSS.

1.3.2 Separation based on arrays of more than two microphones

Noisy observations may result in wrong or biased DOA estimates and may deteriorate the separation performance. A scenario with a varying unknown number of active speakers was addressed in [116], where clustering of the STFT bins that are associated with each desired source was obtained via DOA estimation. In [117], an EM-based separation algorithm is presented, which utilizes spectral cues and phase-based spatial cues. Spatial filtering for sound acquisition in the presence of noise and interfering sources is presented in [118], where DOA estimates are used to indicate the activity of a source within a specific location. The frequency-domain association problem for BSS is an important challenge [119].

The estimation of multiple DOAs by EM was proposed in [39, 16]. The noise spatial characteristics was not explicitly modeled. In [102], an ML estimator of multiple DOAs in noisy environment was derived using an array of microphones. Closed-form estimators of the speech and noise power spectral densities were derived, while the ML for the DOAs was obtained by a grid-search.

1.4 Calibration for ad hoc networks

Localization and tracking using multiple sensor arrays are often handled under the assumption that the locations of the microphone arrays are known precisely. The recent usage of ad hoc networks introduces a new challenge of estimating the array locations in parallel to routine tasks, such as speaker localization [120, 121, 122, 16, 123], noise or reverberation reduction [124, 103, 125]and speaker separation. The new algorithm we present in the calibration chapter combines three tasks: DOA estimation, direct positioning determination (DPD), and calibration for ad hoc networks.

The DOAs estimation in the presence of various noise types can be formulated as a ML estimation problem of deterministic parameters [102, 126, 127, 128]. The DOA challenge in the presence of an unknown noise field was dealt with in [126]. The DOA estimation of the speakers can be obtained by applying a Newton search. The W-disjoint assumption [110] is often exploited for such tasks [129].

A straightforward solution to higher-dimension localization problems involves triangulation of the $1D$ problems solved locally by each array of the network [130]. It has the advantage of simplification, especially for distributed network computations. There are many approaches that use triangulation of DOAs to solve the $2D$ or $3D$ localization problem. An example of such an approach for an acoustic ad hoc network was given in [131].

Because only part of the information is used during the first step of the estimation, these solutions are not necessarily optimal. In a small room, a near-field solution is essential using a more general model.

A possible general solution, which directly estimates the location without any intermediate steps, is frequently referred to as DPD [132]. For acoustic localization, this idea was suggested by [16]. However, they use a non-realistic assumption within the context of ad hoc networks relying on perfect knowledge of array positions. This is often referred to as the calibration problem.

In several algorithms, the nodes positions are assumed to be unknown [133]. Even if they are not measured beforehand, methods for automatic calibration of positions can be employed [58, 134, 135]. Recently, self-calibration methods that can be applied online were introduced [136, 137].

Array-shape calibration has been analyzed from a theoretical point of view for both far-field [138] and near-field scenarios [139]. For acoustic arrays, some approaches have already been suggested for calibration, some of which are suitable for scenarios with a dedicated time for calibrations [140]. Other algorithms use ambient sound to find the inter-distances of microphones [141, 142].

Calibration performed jointly with localization or tracking of sources presents a greater challenge. A family of algorithms called simultaneous localization and mapping (SLAM) for robots was described in [143, 144]. They find the trajectory of a single moving array, the positions of static sources, and major reflectors (for example, walls).

Another popular problem is estimation of static array locations jointly with tracking of moving acoustic sources [145, 146]. The problem is sometimes referred to as simultaneously localizing and tracking (SLAT) [147]. A genetic algorithm was recently presented for a scenario where speakers move around a table in the middle of the room [148]. The arrays are located on the table and the algorithm estimates the arrays' locations and tracks the speakers.

It seems that the most effective algorithms that have been demonstrated are for dynamic scenarios, in which multiple locations of each speaker can be used to calibrate the positions of the microphones. However, approaches that are suitable for static scenarios also exist (for example, [149, 150]). They rely on TDOAs between adjacent microphones. The challenge of self-calibration for static scenarios is discussed in Chapter 6.

1.5 Tracking using static and dynamic arrays

Tracking based on acoustic signals is a widely covered field. One of the methods proposed for that purpose is probability hypothesis density (PHD) [151]. A PHD for MoG has been presented in [152]. We also use MoG for our localization and tracking algorithms, as will be explained in Chapter 7.

Another approach dealing with distributed multi-target tracking (MTT) has been described in [153]. Many algorithms for tracking were developed for RF sensors, driven mainly by the military needs of the United States of America during the previous century. We can learn a lot from that field and borrow some ideas. For example, in [154], WSNs were applied for tracking.

Distributed tracking has been proposed in [155] for the case of a ship moving on a two-dimensional surface. In [156], a three-dimensional distributed tracking algorithm based on GCC was calculated for multiple arrays. The whitened version of GCC is compared to adaptive filtering for under water WASN in case of tracking of moving sources [157] (shallow water).

Outdoor approaches for multi-sources tracking are mainly noise limited [158]. Another outdoor approach dealt with the locations and capabilities of the nodes [159]. We are mainly focused on a different environment: the indoor, which contains significant reverberations.

Some works try to jointly solve the detection and tracking tasks, as described in [160]. They used particle filter (PF) for those tasks and a distributed solution. Tracking by PF was also suggested in [161, 162, 163]. In [163], both simulation and real data results can be found. There are works that suggest a solution for three challenges together: detection, tracking and classification. For example, [164] solved those problems for ships (on the water surface). In [165], those challenges were solved for sources inside the water.

Distributed tracking is quite a new field in the acoustic community. However, centralized tracking techniques that are widely covered can be modified. Very popular tracking mechanisms are Kalman filter and its modifications. Tracking with various Kalman methods using the temporal information has been dealt with in [166].

Tracking speakers is a great challenge due to several obstacles. The first obstacle is the non-stationarity of the signals produced. During a certain time frame, a speaker might be active or non-active. Even when its signal's energy is high enough, PSD might significantly vary from one time slot to another. The second obstacle in tracking acoustic sources results in irregular variations of the RIR. Changes in RIR are significant even for small movements of the source or the microphones. It means that knowing its values for one location might not be relevant for estimation in the next location.

The other obstacles that should be mentioned relate to two groups. The first group consists of the regular tracking issues of multiple sources like crossing paths, dominant source masking others, etc. The second group relates to the distribution of the computations, which might result in sub-optimal solutions compared to the centralized solutions. This challenge is discussed in Chapter 7.

1.5.1 Tracking using static arrays

The task of multiple target tracking (or dynamic localization) using static or quasi-static arrays has significant importance in military and surveillance applications [167, 168, 169, 170]. Classical tracking support only a subset of expected trajectories, since it can assume certain things about the velocity, acceleration and jerk of the target.

When the targets emit acoustic radiation, main applications of tracking are speech separation, indoor robotic assistance, automatic steering of cameras, as well as security and military technologies. Various tracking algorithms for distributed microphone arrays were suggested in [73, 77, 160, 171, 172, 173, 174]. Some of them are more focused on outdoor cases, where sensor noise is usually dominant and others deal with indoor environment [175, 176], which contains significant reverberation.

Classical tracking focuses on a subset of expected trajectories, since it can make certain assumptions regarding the velocity, acceleration and jerk of the targets, enabling a Bayesian approach to the parameters' estimation have been proposed. Well-known Bayesian approaches to target tracking are PHD, particle filters and other statistical based methods [11, 177, 178, 179].

For the task of online speakers dynamic localization, a recursive version of the EM is more suitable, since it allows the tracking of time-varying parameters, and does that with relatively low computational and memory loads. General REM algorithms were proposed, where the parameter estimate is updated by using the new observed data. A recursive version of the EM algorithm was first suggested by Titterington [54], based on a Newton search for the maximization of the likelihood, and assuming that the observations are independent. An almost sure convergence of the TREM algorithm was proved by Wang and Zhao in [180], based on the results of Delyon [181]. A stochastic approximation version for the EM algorithm was proposed by Delyon et al. in [182], and its convergence was proven therein. A further study of the REM approach appears in [183] for the problem of DOA estimation, using TREM and another recursive algorithm suggested by the authors. The authors show that both algorithms converge with probability one (w.p.1.) to a stationary point of the likelihood function.

A different recursive-EM approach was proposed by Cappé and Moulines [184] for exponential-family distributions, where the parameter and the sufficient statistic of the problem are updated iteratively. In the E-step, the sufficient statistic is recursively updated using the latest observation, and in the M-step, the parameter is optimized accordingly. This series of parameter estimates is proven to converge to local minima of the Kullback-Leibler divergence (KLD) in the case of independent observations [184].

REM-based algorithms for speaker dynamic localization in noisy and reverberant environments were presented in [16], where both the Titterington's REM [54] and the Cappé-

Moulines' REM [184] were applied to the problem. Spatially distributed microphone nodes were used in [16], but computation was carried out in a central processing unit.

Since the methods in [16] apply only regular recursion, silent periods result in reduced estimation accuracy. Given a set of measurements, exploiting both past and future observations constitutes a preferable estimation of the current state, especially for non-stationary signals. The estimation pertaining to the past data is usually referred to as forward filtering, while the state estimation pertaining to the future data is referred to as backward filtering, which runs backwards. In online applications, the usage of future observations adds latency to the overall system, and should therefore be restricted.

A bi-directional version has been presented in [185] for video signals, where the data is processed offline in order to use the future samples in addition to the past samples. Some of the Bayesian approaches deal with specific problems of speech with techniques developed originally for communication applications. A maximum *a posteriori* (MAP) approach that exploits Viterbi algorithm was presented in [186], where a forward-backward recursion is used to cope with pauses of the speech signal.

1.5.2 Tracking using moving arrays

The ability of robots to engage in verbal dialogs is a fundamental prerequisite for intuitive interaction between humans and machines. To focus on desired sound sources subject to interference and noise, autonomous systems, such as robots, rely on BF [187] in the direction of salient acoustic events. However, in realistic environments, reverberation causes localization errors and spurious detections due to dominant early reflections [188].

For improved robustness of source localization, spatial diversity of microphones installed on moving platforms can be exploited constructively (using the robot movements) in order to infer the source-sensor distance and to disambiguate the direction of impinging sound waves due to the direct path of a source [189, 190, 191].

1.6 Dissertation structure

The structure of the dissertation follows next. We briefly introduce the chapters' topics and their sections. We give references to our publications on each of the subjects.

We start from problem formulation in the next chapter. It deals with a few versions needed throughout the dissertation.

In Chapter 3 we survey our first papers on distributed localization of concurrent speakers. In Section 3.1 we present the DEM concept based on local hidden variables. The next sections presents the articles that we published for DEM localization.

The first article [192] (Y. Dorfan, G. Hazan, and S. Gannot, “Multiple acoustic sources localization using distributed expectation-maximization algorithm” in 4th Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA), May 2014, pp. 72–76) introduced the new hidden variables and a directed ring based algorithm as presented in Section 3.2.

In Section 3.3 an extension to other network topologies and the recursive version are given based on [193] (Y. Dorfan and S. Gannot, “Tree-based expectation-maximization algorithms for localization of acoustic sources,” IEEE Transactions on Audio, Speech and Language Processing, vol. 23, no. 10, pp. 1692-1703, 2015).

The implementation analysis and comparison is given in Section 3.4. Simulation and experimental results of those articles are given in Section 3.5.

In Section 3.6, a modification of the IDEM for a moving pair is derived. It is applied to robot audition utilizing motion for spatial diversity and compared to a Bayesian algorithm [194] (Y. Dorfan, C. Evers, S. Gannot, and P. A. Naylor, “Speaker localization with moving microphone arrays” in Proc. European Signal Processing Conf. (EUSIPCO), Budapest, Hungary, Aug. 2016).

In Chapter 4, we develop localization algorithms for high reverberation levels. In Section 4.1 an algorithm is proposed that enables to estimate location of more than two concurrent speakers in higher levels of reverberation. This section is based on [195] (Y. Dorfan, A. Plinge, Hazan G., and S. Gannot, “Distributed expectation-maximization algorithm for speaker localization in reverberant environments,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 26, no. 3, pp. 682 - 695, Mar. 2018.).

Section 4.2 is dedicated for presenting a new model that takes into account the diffused nature of the late reflections. An arbitrary array of microphones (at least two in any planar geometry) is used for DOA estimation of multiple concurrent speakers [196] (O. Schwartz, Y. Dorfan, E.A.P. Habets, and S. Gannot, “Multiple DOA estimation in reverberant conditions using EM” in International Workshop for Acoustic Echo Cancellation and Noise Control (IWAENC), Xi’an, China, 2016).

In Chapter 5 the topic of BSS is considered for centralized and distributed algorithms. Section 5.1, is dedicated to derivation of a distributed BSS which enables even a single pair of microphones to filter any speaker based on the shared global parameters that were presented for our IDEM localization algorithm [197] (Y. Dorfan, D. Cherkassky, and S. Gannot, “Speaker localization and separation using incremental distributed expectation-maximization” in European Signal Processing Conference (EUSIPCO), 2015, pp. 1256–1260.).

In Section 5.2, it is shown how the hidden variables of our DOA estimation algorithm from the previous chapter can be utilized (as already done in our previous work described in the first section) for construction of an efficient spectral mask [198] (Y. Dorfan, O. Schwartz, B. Schwartz, E. A. P. Habets, and S. Gannot, “Multiple DOA estimation and blind source separation using estimation-maximization” in IEEE Science of Electrical Engineering (ICSEE), Eilat, Israel, 2016). The geometry of the array is general, although the experimental part analyzes a linear array of four microphones.

In Chapter 6 we discuss a major challenge in the field of ad hoc networks: the calibration. The idea is to use one of the arrays as an anchor and to jointly estimate the relative positions of the other arrays and the active speakers. This chapter is based on [199] (Y. Dorfan, O. Schwartz, and S. Gannot, “Joint speaker localization and array calibration using expectation-maximization,” To be submitted to IEEE...).

In Chapter 7 we propose procedures for tracking. In Section 7.1, a systematic scheme for tracking using future samples in addition to past samples is derived. This section is based on [200] (Y. Dorfan, B. Schwartz, and S. Gannot, “Speaker tracking using forward-backward recursive expectation-maximization,” To be submitted to IEEE...).

In Section 7.2 we consider a tracking algorithm utilizing information from a single moving pair. This section is based on [201] (C. Evers, Y. Dorfan, S. Gannot, and P. A. Naylor, “Source tracking using moving microphone arrays for robot audition” in IEEE International Conference on Audio and Acoustic Signal Processing (ICASSP), New Orleans, USA, 2017).

In Chapter 8 we summarize and discuss shortly a few future research directions that might be interesting to explore.

At the end of this dissertation, there is an appendix which describes detailed implementation analysis of three localization algorithms presented in Chapter 3.

1.7 List of publications

The list of publications of the author included in this dissertation are enlisted next in chronology order.

Journal publications

1. Y. Dorfan and S. Gannot, “Tree-based expectation-maximization algorithms for localization of acoustic sources,” IEEE Transactions on Audio, Speech and Language Processing, vol. 23, no. 10, pp. 1692-1703, 2015.

2. Y. Dorfan, B. Schwartz, and S. Gannot, “Speaker tracking using forward-backward recursive expectation-maximization,” To be submitted to IEEE...

3. Y. Dorfan, A. Plinge, Hazan G., and S. Gannot, “Distributed expectation-maximization algorithm for speaker localization in reverberant environments,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 26, no. 3, pp. 682 - 695, Mar. 2018.

4. Y. Dorfan, O. Schwartz, and S. Gannot, “Joint speaker localization and array calibration using expectation-maximization,” To be submitted to IEEE...

Peer reviewed conference publications

1. Y. Dorfan, G. Hazan, and S. Gannot, ”Multiple acoustic sources localization using distributed expectation-maximization algorithm” in 4th Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA), May 2014, pp. 72–76 [**Best student paper award**].

2. Y. Dorfan, D. Cherkassky, and S. Gannot, ”Speaker localization and separation using incremental distributed expectation-maximization” in European Signal Processing Conference (EUSIPCO), 2015, pp. 1256–1260.

3. Y. Dorfan, C. Evers, S. Gannot, and P. A. Naylor, ”Speaker localization with moving microphone arrays” in Proc. European Signal Processing Conf. (EUSIPCO), Budapest, Hungary, Aug. 2016.

4. O. Schwartz, Y. Dorfan, E.A.P. Habets, and S. Gannot, ”Multiple DOA estimation in reverberant conditions using EM” in International Workshop for Acoustic Echo Cancellation and Noise Control (IWAENC), Xi’an, China, 2016.

5. Y. Dorfan, O. Schwartz, B. Schwartz, E. A. P. Habets, and S. Gannot, ”Multiple DOA estimation and blind source separation using estimation-maximization” in IEEE Science of Electrical Engineering (ICSEE), Eilat, Israel, 2016.

6. C. Evers, Y. Dorfan, S. Gannot, and P. A. Naylor, ”Source tracking using moving microphone arrays for robot audition” in IEEE International Conference on Audio and Acoustic Signal Processing (ICASSP), New Orleans, USA, 2017.

Chapter 2

Problem formulation

Problem formulation for our localization, BSS and tracking algorithms varies a bit along the various chapters. This chapter summarizes the common parts of the formulations. The first section deals with the formulation of PRP used for source localization and BSS. The second section is about sub-band time difference of arrivals, which is more common and will be used also by our tracking algorithm. It is interesting to explore the novel usage of the complex STFT measurements themselves without eliminating the amplitude information, as introduced in the third subsection. This kind of measurement can be used for localization, tracking and BSS. The fourth formulation deals with the usage of STFT for DOA estimation, BSS, multi-dimensional localization and sensors calibration.

2.1 The PRP localization

For our first algorithms we adapted the PRP measurements described in [16]. We used it for multiple static pairs and for a single moving pair. The motivation was mainly simplicity and the fact that it contains most of the relevant information. The usage of PRP assumes a specific model of acoustic transfer function (ATF) described in [41].

2.1.1 Multiple static pair-wise relative phase ratios

Multiple sources are located at unknown positions in a reverberant enclosure. The purpose is to find the number of active sources and their locations. We define unknown weights or probabilities for all possible locations on a grid. The algorithms will estimate the weights. Applying a threshold enables us to determine the number of sources and their locations. We define these weights as global parameters that are estimated at each iteration or sub-iteration. In addition to those global parameters, we can define local ones. In order to estimate the parameters with EM, we define local hidden variables that are updated at each node. They are aggregated to global indicators that can be seen as SS used for parameter estimation. It allows distributed computations with minimal communication between the nodes.

The problem is formulated in the STFT domain with $t = 1, \ldots, T$ as the time index and $k = 0, \ldots, K-1$ as the frequency index. There are J acoustic signals captured by M microphone pairs. The signal received by the ith microphone, $i = 1, 2$, of the mth pair, $m = 1, \ldots, M$, is given by:

$$z_m^i(t,k) = \sum_{j=1}^{J} a_{jm}^i(t,k) v_j(t,k) + n_m^i(t,k), \tag{2.1}$$

where $j = 1, \ldots, J$ is the source index. $v_j(t,k)$ denotes the jth source signal, $n_m^i(t,k)$ denotes additive noise as captured by the ith microphone of the mth pair, and $a_{jm}^i(t,k)$ denotes the ATF from the jth source to the ith microphone of the mth node. The ATF in reverberant environments consists of a direct path (which bears the desired information for localization) and reflections (which usually degrade the localization performance).

The first stage of all localization procedures discussed below consists of PRP extraction, given for the mth microphone pair:

$$\phi_m(t,k) \triangleq \frac{z_m^2(t,k)|z_m^1(t,k)|}{z_m^1(t,k)|z_m^2(t,k)|}. \tag{2.2}$$

A schematic block diagram of the pre-processing stage is depicted in Fig. 2.1. The STFT is applied to each microphone signal. The PRPs are then calculated for each time-frequency bin separately. These PRPs are induced by the TDOA between the microphone-pair signals as a response to an acoustic source located in $\mathbf{p} \in \mathcal{P}$:

$$\tau_m(\mathbf{p}) \triangleq \frac{||\mathbf{p} - \mathbf{p}_m^2|| - ||\mathbf{p} - \mathbf{p}_m^1||}{c}, \tag{2.3}$$

where $\mathbf{p}_m^1$ and $\mathbf{p}_m^2$ are the locations of the microphones in pair m, $||\cdot||$ denotes the Euclidean norm and c is the sound velocity. $\mathcal{P}$ is the set of all possible source locations in the enclosure. In this work, we selected a regular grid of possible locations with a desired resolution. Note that any PRP could be associated with multiple source locations. The locus of all these

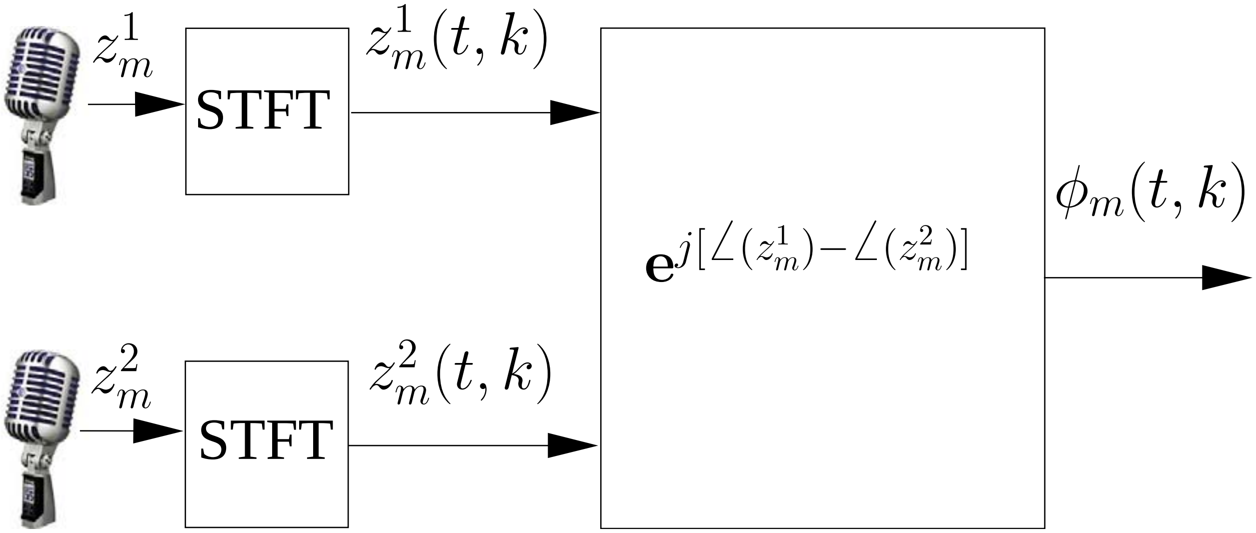


Figure 2.1: Pre-processing at the mth microphone pair to extract the pair-wise relative phase ratio.

locations is a one-sheet hyperboloid.

The various speakers are assumed to exhibit disjoint activity in the STFT domain [16, 17, 18]. This assumption is often referred to as the W-disjoint property of the speech signals [202]. Under this assumption and an upper bound of the number of concurrent speakers, each time-frequency bin can be associated with at most a single active position (and therefore with a single speaker). The case of no active speakers could be treated in various ways. We express the PRP in the following statistical model:

$$\phi_m(t,k) \sim \sum_{\mathbf{p}} \psi_{\mathbf{p}} \mathcal{N}^c\left(\phi_m(t,k); \tilde{\phi}_m^k(\mathbf{p}), \sigma^2\right), \tag{2.4}$$

where $\psi_{\mathbf{p}}$ is the probability of a position $\mathbf{p}$ to be induced by a speaker. $\mathcal{N}^c(\cdot;\cdot,\cdot)$ denotes the complex-Gaussian probability with variance σ^2:

$$\mathcal{N}^c\left(\phi_m(t,k); \tilde{\phi}_m^k(\mathbf{p}), \sigma^2\right) = \frac{1}{\pi\sigma^2} \exp\left(-\frac{|\phi_m(t,k) - \tilde{\phi}_m^k(\mathbf{p})|^2}{\sigma^2}\right). \tag{2.5}$$

The complex-Gaussian distribution is used here as in [16, 39], although it is not accurate. The Von Mises distribution is sometimes used for TDOA estimation [77, 203], since its properties fit phase measurements better.

The mean-value of each Gaussian, $\tilde{\phi}_m^k(\mathbf{p})$ is set to the nominal PRP induced by $\mathbf{p}$. It is given by:

$$\tilde{\phi}_m^k(\mathbf{p}) \triangleq \exp\left(-j\frac{2\pi k\tau_m(\mathbf{p})}{KT_s}\right) \forall \mathbf{p} \in \mathcal{P}, \tag{2.6}$$

where T_s denotes the sampling period and $j = \sqrt{(-1)}$.

Being a probability function, the following set of equations can be written:

$$\sum_{\mathbf{p}\in\mathcal{P}} \psi_{\mathbf{p}} = 1, 0 < \psi_{\mathbf{p}} < 1; \forall \mathbf{p} \in \mathcal{P}. \tag{2.7}$$

As shown in [47] for a vector of independent data set, we can augment the PRP readings in all time-frequency bins, $\boldsymbol{\phi}_m = \text{vec}_{t,k}\left(\phi_m(t,k)\right)$. Using the W-disjoint property above the p.d.f. of the observation set for each node m can be stated as:

$$f(\boldsymbol{\Phi}_m = \boldsymbol{\phi}_m; \boldsymbol{\psi}) = \prod_{t,k}\sum_{\mathbf{p}} \psi_{\mathbf{p}}\mathcal{N}^c\left(\phi_m(t,k); \tilde{\phi}_m^k(\mathbf{p}), \sigma^2\right). \tag{2.8}$$

In this work, we consider the actual location of the sources; i.e., information that cannot be extracted reliably from a single microphone-pair. We therefore concatenate all microphone-pair readings to describe the probabilistic model of the source locations assuming that all microphone-pair readings are independent:

$$f(\boldsymbol{\Phi} = \boldsymbol{\phi}; \boldsymbol{\psi}) = \prod_m f(\boldsymbol{\Phi}_m = \boldsymbol{\phi}_m; \boldsymbol{\psi}) = \prod_{m,t,k}\sum_{\mathbf{p}} \psi_{\mathbf{p}}\mathcal{N}^c\left(\phi_m(t,k); \tilde{\phi}_m^k(\mathbf{p}), \sigma^2\right). \tag{2.9}$$

The independence assumption can be justified by the different reflection patterns the signal undergoes before being captured by the microphone-pair.

A key point in deriving the distributed localization schemes is the inter-relation between the available data in the microphone-pair versus the network. We can now define the set of all probabilities of all possible grid locations in a vectorial notation:

$$\boldsymbol{\psi} = \text{vec}_{\mathbf{p}}\left(\psi_{\mathbf{p}}\right). \tag{2.10}$$

In the same way as described in [47], the MLE of the speakers' location (global parameter) can be obtained by maximizing the expression from equation (2.9) w.r.t. to $\boldsymbol{\psi}$ as dealt in

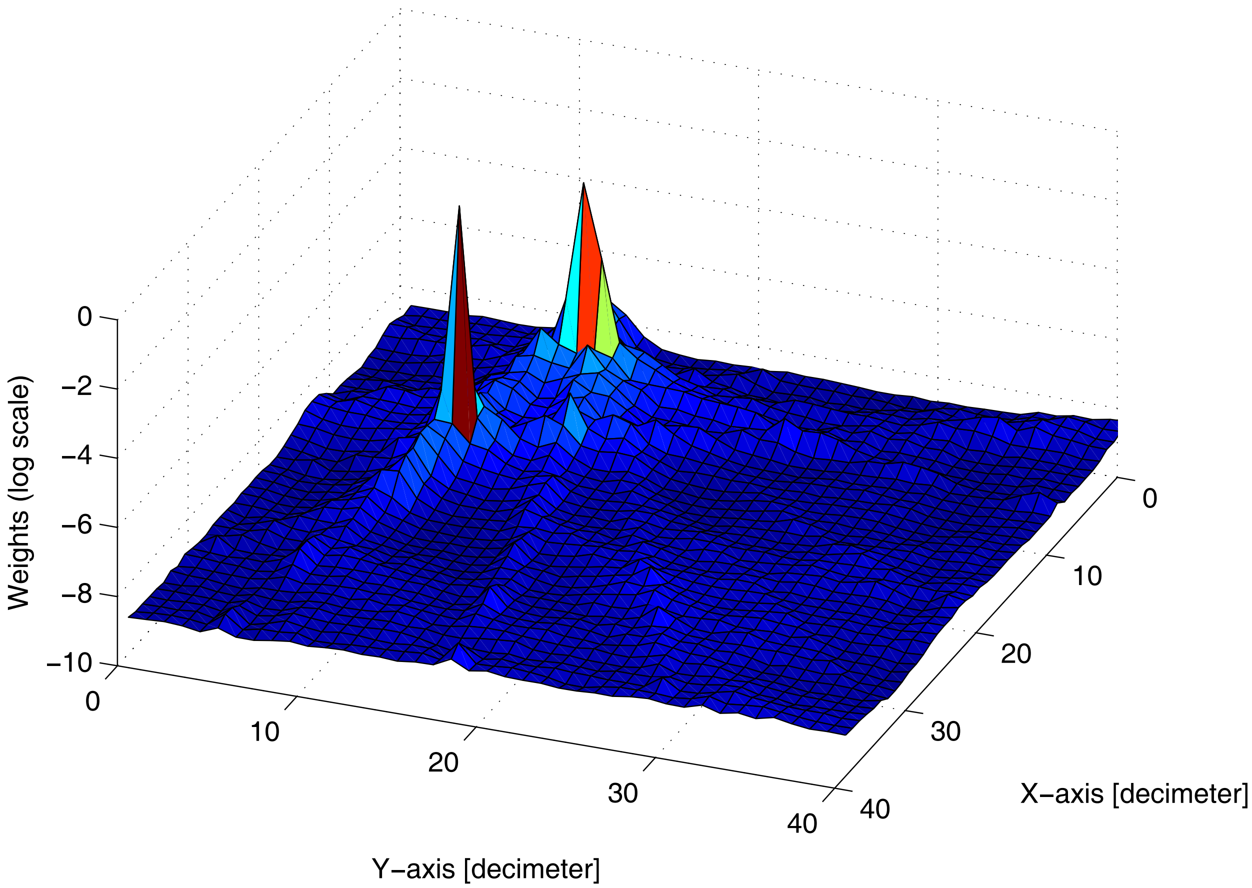


Figure 2.2: Map of $\hat{\psi}_{\mathbf{p}}$ as a function of the room's floor coordinates with resolution of 10 × 10 cm. By applying a threshold to this map it can be deduced that there are two active sources in the room.

depth in the next chapter:

$$\hat{\boldsymbol{\psi}} = \underset{\boldsymbol{\psi}}{\operatorname{argmax}}[\log f(\boldsymbol{\Phi} = \boldsymbol{\phi}; \boldsymbol{\psi})]. \tag{2.11}$$

An example of a typical estimator of $\boldsymbol{\psi}$ as function of positions is depicted in Fig. 2.2 for a two-dimensional case with a resolution of 10 × 10 cm. The methods derived in this work are also applicable to the three-dimensional case. In the figure, the probability of having a source at each location in the room is represented by the z-axis with color (and height) that are proportional to its value (in natural log units). The final estimation of the number of sources and their locations can be deduced from this map by applying a proper threshold.

2.1.2 Single pair-wise relative phase ratio for a dynamic pair

In this subsection models for the source and the single sensor dynamics are defined. Furthermore, the measurement model of the source, which will be used for the sound localization algorithms, is presented.

2.1.2.1 Stationary source dynamics

The source position, $\boldsymbol{p}_t \triangleq \left[x_s(t), y_s(t)\right]^T$, is defined as the absolute two-dimensional Cartesian position within the room. In this section, a static source is assumed, i.e.,

$$\boldsymbol{p}_0 = \cdots = \boldsymbol{p}_t = \boldsymbol{p}. \tag{2.12}$$

2.1.2.2 Measurement model

The microphone array used consists of one pair with two-dimensional Cartesian positions, $\boldsymbol{p}^i(t) \triangleq \left[x^i(t), y^i(t)\right]^T$, $i = 1, 2$ and where the platform moves with speed, $v(t)$. The localization procedure relies on the PRP extraction [16] given above in equation (2.2).

These PRPs are induced by the TDOA, which can be defined as:

$$\tau(\mathbf{p}, \mathbf{p}^1(t), \mathbf{p}^2(t)) \triangleq \frac{||\mathbf{p} - \mathbf{p}^2(t)|| - ||\mathbf{p} - \mathbf{p}^1(t)||}{c}. \tag{2.13}$$

We model the PRPs using a Gaussian mixture model (GMM):

$$\phi(t, k) \sim \sum_{\mathbf{p}} \psi_{\mathbf{p}} \mathcal{N}^c \left(\phi(t, k); \tilde{\phi}^k(\mathbf{p}, t), \sigma^2\right). \tag{2.14}$$

Assuming the sensors position is known at every time, the mean of each Gaussian can be calculated in advance on a grid of all possible locations:

$$\tilde{\phi}^k(\mathbf{p}, t) \triangleq \exp\left(-j\frac{2\pi k\tau(\mathbf{p}, \mathbf{p}^1(t), \mathbf{p}^2(t))}{KT_s}\right). \tag{2.15}$$

2.2 Sub-band TDOA localization

More commonly used measures are TDOAs. In one of our papers we use the sub-band version for multiple speakers localization in a reverberant room. In another paper we use the same kind of measurements for tracking speakers.

We chose a different feature vector that necessitates a new probabilistic description. In the original model, a complex MoG is defined to describe the PRP, which is a complex feature vector. Modeling the PRP as a complex-Gaussian is only an approximation, since magnitude 1 is assumed (see [193]). Here the feature vector is the real-valued TDOAs, hence a real-Gaussian can be used. Moreover, as the values of the TDOAs are confined to a physically plausible range, the truncated Gaussian can be a good model for the estimated TDOAs values. We use subband TDOA estimates, similarly to the model described in [204], and derive a modified version of the IDEM algorithm.

The new statistical model will be described in chapter 4 as a mixture of *truncated* Gaussians [205], which is a real-valued distribution consisting of random variables with a finite support. In [205], two alternative strategies were described for measurements confined to a finite range: *truncation* and *censor*. The truncation strategy classifies each value outside the allowed range as illegal, whereas the censor strategy substitutes these values with the closest value within the physical range. In our case, the direct path can only produce a TDOA within a confined range of values. We adopted the truncation strategy, since a TDOA reading that exceeds the permissible range, most probably does not contain any meaningful localization information. Unlike truncation, adopting the censor strategy may cause artifacts in certain positions in the room.

Consider M microphone pairs receiving signals from J speakers. A noiseless environment is considered. The number of speakers is unknown in advance. Let $y_{m,i}(t)$ be the signals received by the ith microphone of the mth node. The signals in the time domain are given by

$$y_{m,i}(t) = \sum_{j=1}^{J} (g_{j,m,i}(t) * s_j(t) + r_{j,m,i}(t) * s_j(t)), \tag{2.16}$$

where $t = 0, \ldots, T-1$ denotes the time index, j is the speaker index, and $s_j(t)$ denotes the speech signal produced by speaker j. The node index is $m = 1, \ldots, M$. At each node, we use $i = 1, 2$ as the index of each microphone.

The direct part of the RIR, $g_{j,m,i}(t)$, comprises an attenuation and a phase shift. The residual part of the RIR, $r_{j,m,i}(t)$, consists of all the reflections, namely the multi-path components. As the reverberation of each speaker is independent of the other speakers, the total power increases with the number of speakers, J.

In some of the proposed localization algorithms [192, 193], only the direct-path is utilized for the estimation. These algorithms tend to work well for low reverberation levels and small number of concurrent speakers, since they did not take the other reflections into account. In contrast, the method we propose in chapter 4 tries to reduce the late reflections by the construction of the feature vector.

In the presence of additive noise we can add a noise reduction algorithm similarly to [70]. Although the focus is on reverberation, additive noise influence is discussed in the experimental section as well.

2.3 Localization and source separation based on STFT samples

In this section the statistical model for the STFT samples is described. Unlike PRPs, the STFT coefficients contain also the amplitude information about the sources. In this chapter we focus on general definitions. Details about the applications (localization and BSS) will be given in the following chapters.

2.3.1 Signal model

Consider an array of N microphones receiving signals from J speakers. A noiseless environment is considered. While applying the algorithm, the number of speakers is not known. Let $Z_n(t,k)$ be the signals received by the nth microphone, where $n = 1,\dots,N$. The signals in the STFT domain are given by:

$$Z_n(t,k) = \sum_{j=1}^{J} G_{j,n}(k) S_j(t,k) + R_{j,n}, \tag{2.17}$$

where $S_j(t,k)$ denotes the speech signal produced by speaker j, $G_{j,n}(k)$ denotes the relative direct transfer function (RDTF) from source j to microphone n w.r.t. the reference microphone (arbitrary chosen as microphone #1) and $R_{j,n}$ denotes the reverberation associated with speaker j.

In vector notation, the N microphone signals can be written as:

$$\boldsymbol{z}(t,k) = \sum_{j=1}^{J} \boldsymbol{g}_j(k) S_j(t,k) + \boldsymbol{r}_j(t,k),$$

where:

$$\begin{aligned}\boldsymbol{z}(t,k) &= \begin{bmatrix} Z_1(t,k) & \ldots & Z_N(t,k) \end{bmatrix}^{\mathrm{T}} \\ \boldsymbol{g}_j(k) &= \begin{bmatrix} G_{j,1}(k) & \ldots & G_{j,N}(k) \end{bmatrix}^{\mathrm{T}} \\ \boldsymbol{r}_j(t,k) &= \begin{bmatrix} R_{j,1}(t,k) & \ldots & R_{j,N}(t,k) \end{bmatrix}^{\mathrm{T}}.\end{aligned}$$

The RDTF can be expressed by the direct arrival alone (assuming far-field and a linear microphone array):

$$G_{j,n}(k) = \exp\left(-j\frac{2\pi k}{K}\frac{\tau_{j,n}}{T_s}\right), \tag{2.18}$$

where $\tau_{j,n}$ is the TDOA of speaker j between microphone n and the reference microphone given by:

$$\tau_{j,n} = \frac{d_n \cos(\vartheta_j)}{c}. \tag{2.19}$$

d_n is the distance between microphone n and the reference microphone, and ϑ_j is the angle of arrival of speaker j.

Define the vector of the concatenated unknown angles of arrival as: $\boldsymbol{\vartheta} = \begin{bmatrix} \vartheta_1 & \cdots & \vartheta_J \end{bmatrix}^T$. Estimating the DOAs is the goal of this section.

2.3.2 The maximum likelihood problem

The gist of the proposed method, adopted from [39, 16], is to predefine a set of candidate angles and to determine the probability that source signals arrive from these angles, rather than directly estimating the DOA. The statistical model used here, significantly differs from [39, 16], as will be clarified in the sequel.

The various speakers are assumed to exhibit disjoint activity in the STFT domain. Therefore, by means of a clustering process, every T-F bin of $\boldsymbol{z}(t,k)$ can be associated with a single active source. Hence, the observations are given the following probabilistic MoG description:

$$\boldsymbol{z}(t,k) \sim \sum_{\vartheta=1}^{V} \psi_\vartheta \mathcal{N}^c\left(\boldsymbol{z}(t,k); \mathbf{0}, \boldsymbol{\Phi}_\vartheta(t,k)\right). \tag{2.20}$$

The prior ψ_ϑ is the (unknown) probability of a speaker to be located at a candidate angle ϑ. In case no *a priori* knowledge on these candidate DOAs is available, they can be uniformly distributed in the range 1° − 180°. Naturally, $\sum_{\vartheta=1}^{V} \psi_\vartheta{=}1$, where V is the total number of candidate DOAs.

In our case, $\boldsymbol{\Sigma} = \boldsymbol{\Phi}_\vartheta(t,k)$ is the PSD matrix of $\boldsymbol{z}(t,k)$, given that $\boldsymbol{z}(t,k)$ is associated with speaker angle ϑ:

$$\boldsymbol{\Phi}_\vartheta(t,k) = \boldsymbol{g}_\vartheta \boldsymbol{g}_\vartheta^{\mathrm{H}} \phi_{S,\vartheta}(t,k) + \boldsymbol{\Phi}_{\boldsymbol{r},\vartheta}(t,k), \tag{2.21}$$

where the RDTF $\boldsymbol{g}_\vartheta$ depends on the candidate source DOA ϑ through (2.18)-(2.19).

Note, that unlike [16], where the positions estimates were embedded in the means of the Gaussians, here they are instead embedded in the correlation matrices. This major difference between the statistical models enables the improved treatment of reverberation, proposed in the current section.

The time-varying PSD of the speaker $\phi_{S,\vartheta}(t,k)$ and time-varying PSD matrix of the reverberation $\boldsymbol{\Phi}_{\boldsymbol{r},\vartheta}(t,k)$ (both of a candidate source at DOA ϑ) is defined as:

$$\phi_{S,\vartheta}(t,k) = E\left\{|S_\vartheta(t,k)|^2\right\}, \tag{2.22}$$

$$\boldsymbol{\Phi}_{\boldsymbol{r},\vartheta}(t,k) = E\left\{\boldsymbol{r}_\vartheta(t,k)\boldsymbol{r}_\vartheta^{\mathrm{H}}(t,k)\right\}. \tag{2.23}$$

Although the PSD matrix of the reverberation is time-varying, its spatial characteristics can be assumed time-invariant, as long as the speaker and microphones positions are fixed. Therefore, it is reasonable to model the PSD matrix of the reverberation as:

$$\boldsymbol{\Phi}_{\boldsymbol{r},\vartheta}(t,k) = \boldsymbol{\Gamma}(k)\phi_{R,\vartheta}(t,k), \tag{2.24}$$

where $\boldsymbol{\Gamma}(k)$ is the time-invariant spatial coherence matrix of the reverberation and $\phi_{R,\vartheta}(t,k)$ is the time-varying PSD of the reverberant component of a speaker at angle ϑ.

In the current section, we assume that the reverberation can be modeled using a spatially homogeneous and spherically isotropic sound field and set the i-th row and j-th column of $\boldsymbol{\Gamma}(k)$ accordingly [206, 207]:

$$\Gamma_{ij}(k) = \operatorname{sinc}\left(\frac{2\pi k}{K}\frac{d_{i,j}}{T_{\mathrm{s}}c}\right) + \epsilon\delta(i-j), \tag{2.25}$$

where $\operatorname{sinc}(x) = \sin(x)/x$, ϵ is the level of diagonal loading, and $d_{i,j}$ is the inter-distance between microphones i and j.

Finally, by augmenting all observations in $\boldsymbol{z} = \operatorname{vec}_{t,k}(\{\boldsymbol{z}(t,k)\})$, the p.d.f. of the entire observation set can be stated as:

$$f(\boldsymbol{z}) = \prod_{t,k} \sum_{\vartheta=1}^{V} \psi_\vartheta \mathcal{N}^c\left(\boldsymbol{z}(t,k); \mathbf{0}, \boldsymbol{\Phi}_\vartheta(t,k)\right), \tag{2.26}$$

where we assume that the observations for all time segments and frequency bins are independent.

Let the unknown parameter set be: $\boldsymbol{\theta} = \left[\boldsymbol{\psi}^{\mathrm{T}}, \boldsymbol{\phi}_S^{\mathrm{T}}, \boldsymbol{\phi}_R^{\mathrm{T}}\right]^{\mathrm{T}}$, where $\boldsymbol{\psi} = \mathrm{vec}_\vartheta\left(\psi_\vartheta\right)$, $\boldsymbol{\phi}_S = \mathrm{vec}_{\vartheta,t,k}\left(\phi_{S,\vartheta}(t,k)\right)$ and $\boldsymbol{\phi}_R = \mathrm{vec}_{\vartheta,t,k}\left(\phi_{R,\vartheta}(t,k)\right)$. The desired DOAs can be extracted from the prior probabilities $\boldsymbol{\psi}$. The parameter set $\boldsymbol{\phi}_S$ and $\boldsymbol{\phi}_R$ are nuisance parameters of the problem at hand, since they do not contain any information regarding the DOAs, but need to be estimated. The MLE problem can readily be stated as:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\mathrm{argmax}} \log f\left(\boldsymbol{z}; \boldsymbol{\theta}\right). \tag{2.27}$$

2.4 Applying BSS based on DOA estimation

The STFT coefficients described above in the previous section can be also used for BSS. We will focus on the slight differences during this section.

2.4.1 Signal model

The measured array output is a linear combination of the incoming waveforms. Let $Z_n(t,k)$ denote the signals received by the nth microphone, where $n = 1, \ldots, N$. The signals in the STFT domain are given by:

$$Z_n(t,k) = \sum_{j=1}^{J} G_{n,j}(k) S_j(t,k) + V_n(t,k). \tag{2.28}$$

where $V_n(t,k)$ denotes the ambient noise at microphone n.

The N microphone signals can be concatenated in a vector form:

$$\boldsymbol{z}(t,k) = \sum_{j=1}^{J} \boldsymbol{g}_j(k) S_j(t,k) + \boldsymbol{v}(t,k),$$

where:

$$\begin{aligned}\boldsymbol{z}(t,k) &= \begin{bmatrix} Z_1(t,k) & \dots & Z_N(t,k) \end{bmatrix}^{\mathrm{T}} \\ \boldsymbol{g}_j(k) &= \begin{bmatrix} G_{j,1}(k) & \dots & G_{j,N}(k) \end{bmatrix}^{\mathrm{T}} \\ \boldsymbol{v}(t,k) &= \begin{bmatrix} V_1(t,k) & \dots & V_N(t,k) \end{bmatrix}^{\mathrm{T}}.\end{aligned}$$

The relative transfer function (RTF) can be expressed by the direct arrival only, since non-reverberant enclosure is assumed:

$$G_{n,j}(k) = \exp\left(-j\frac{2\pi k}{K}\frac{\tau_{n,j}}{T_s}\right), \tag{2.29}$$

Estimating the speakers $S_j(t,k)$ for $j = 1, \dots, J$ is the goal of BSS. Define the vector of the concatenated unknown AOAs of the speakers: $\boldsymbol{\vartheta} = \begin{bmatrix} \vartheta_1 & \cdots & \vartheta_J \end{bmatrix}^{\mathrm{T}}$. The ML estimator for $\boldsymbol{\vartheta}$ is obtained by applying the EM procedure. Spectral masks and minimum variance distortion-less response (MVDR)-BF, which can be used for the estimation, are obtained as a by-product of the EM algorithm as described in the sequel.

2.4.2 Statistical model

We use MoG to characterize the mixture of all potential speakers' DOA. The various speakers are assumed to exhibit disjoint activity in the STFT domain (W-disjoint orthogonality assumption [110]). Therefore, by means of clustering, every T-F bin of $\boldsymbol{z}(t,k)$ can be associated with a single active source.

Hence, based on the above arguments, the observations may be given the following probabilistic description:

$$\boldsymbol{z}(t,k) \sim \sum_{\vartheta=1}^{V} \psi_\vartheta \mathcal{N}^c\left(\boldsymbol{z}(t,k); \mathbf{0}, \boldsymbol{\Phi}_\vartheta(t,k)\right). \tag{2.30}$$

The matrix $\boldsymbol{\Phi}_\vartheta(t,k)$ is the PSD of $\boldsymbol{z}(t,k)$ given that $\boldsymbol{z}(t,k)$ is associated with the speaker from the ϑ-th angle:

$$\boldsymbol{\Phi}_\vartheta(t,k) = \boldsymbol{g}_\vartheta(k)\boldsymbol{g}_\vartheta^{\mathrm{H}}(k)\phi_{S,\vartheta}(t,k) + \boldsymbol{\Phi}_{\boldsymbol{v}}(k), \tag{2.31}$$

where the RTF $\boldsymbol{g}_\vartheta(k)$ is given by (2.29), $\phi_{S,\vartheta}(t,k)$ is the speech PSD and $\boldsymbol{\Phi}_{\boldsymbol{v}}(t,k)$ is the noise

PSD, i.e.

$$\phi_{S,\vartheta}(t,k) = E\left\{|S_\vartheta(t,k)|^2\right\}, \tag{2.32}$$

$$\boldsymbol{\Phi}_{\boldsymbol{v}}(k) = E\left\{\boldsymbol{v}(t,k)\boldsymbol{v}^{\mathrm{H}}(t,k)\right\}. \tag{2.33}$$

The PSD matrix of the noise is assumed time-invariant and either known in advance or can be estimated during speech absence periods.

Finally, by augmenting all observations in $\boldsymbol{z} = \mathrm{vec}_{t,k}(\{\boldsymbol{z}(t,k)\})$, the p.d.f. of the entire observation set can be stated as

$$f(\boldsymbol{z};\boldsymbol{\theta}) = \prod_{t,k}\sum_{\vartheta=1}^{V}\psi_\vartheta\mathcal{N}^c\left(\boldsymbol{z}(t,k);\mathbf{0},\boldsymbol{\Phi}_\vartheta(t,k)\right), \tag{2.34}$$

where independency is assumed between T-F bins.

Let the unknown parameter set be $\boldsymbol{\theta} = \left[\boldsymbol{\psi}^{\mathrm{T}},\boldsymbol{\phi}_S^{\mathrm{T}}\right]^{\mathrm{T}}$, where, $\boldsymbol{\psi} = \mathrm{vec}_\vartheta\left(\psi_\vartheta\right)$ and $\boldsymbol{\phi}_S = \mathrm{vec}_{\vartheta,t,k}\left(\phi_{S,\vartheta}(t,k)\right)$. The ML problem of the DOA estimation can readily be stated as:

$$\widehat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\mathrm{argmax}}\log f\left(\boldsymbol{z};\boldsymbol{\theta}\right). \tag{2.35}$$

Chapter 3

Localization algorithms

The material presented in this chapter is based on [192], [193] and [194]:

[192] Y. Dorfan, G. Hazan, and S. Gannot, "Multiple acoustic sources localization using distributed expectation-maximization algorithm," in *the 4th Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA)*, May 2014, pp. 72–76

[193] Y. Dorfan and S. Gannot, "Tree-based recursive expectation-maximization algorithm for localization of acoustic sources," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 10, pp. 1692–1703, 2015

[194] Y. Dorfan, C. Evers, S. Gannot, and P. A. Naylor, "Speaker localization with moving microphone arrays," in *Proc. European Signal Processing Conf. (EUSIPCO)*, Budapest, Hungary, Aug. 2016

The challenge of localizing a number of concurrent acoustic sources in reverberant enclosures is addressed. The presence of sources and their number are assumed to be unknown. The location of the sources are estimated by the localization algorithms that we derive. We formulate the task as a ML parameter estimation problem, and develop DEM algorithms to enable usage of ad hoc networks with limited communication and computation resources.

Unlike direction estimation, $2D$ or $3D$ localization of acoustic sources is naturally a *distributed sensing* task, since the sensors must be spatially deployed. In the case of an ad hoc WASN, *distributed computation* is also required, since there is no central point and resources (communication and computation) are limited. We are interested in algorithms that enable localization of the speakers without a central point. Taking advantage of the distributed constellation of the sensors, we propose algorithms that utilize multiple processing nodes and consider communication constraints between them.

The first localization algorithm we propose, denoted here as IDEM, is based on the IEM framework mentioned above. The IDEM uses a new definition of *local* hidden variables. They enable *distributed computation* and have a convergence advantage. The IDEM presented here is a modified version of [192].

We use the concept of local hidden variables and expand their usage. We propose a family of DEM algorithms. In addition to the IDEM implemented over a directed-ring, we develop the batch distributed expectation-maximization (BDEM) and the RDEM algorithms, which are implemented over a bi-directional tree. Unlike the IDEM algorithm, these algorithms enable parallel computations by the nodes and higher robustness for communication failures. The RDEM better fits real-time applications due to its recursive nature. The two new algorithms are compared for the static case, in terms of implementation and performance. Actually those topologies are not the only ones that can be used for the local hidden variables suggested, but they are very useful examples that can be analyzed easily.

3.1 The distributed expectation-maximization concept

Following [16, 39] and based on the definitions in subsection 2.1.1, we formulate the localization problem as an MLE and solve it using EM iterations. As already stated above, unlike [16], which used *distributed sensing* calculated in a central way, we are interested also in *distributed computation*. The hidden variables are defined globally in [16], as the association of each time-frequency bin (t, k) with a particular source s located in position $\mathbf{p}$. This indicator is denoted $x(t, k, \mathbf{p}, s)$. In our research, we are interested in *distributed computations* and therefore propose to replace the global hidden variables with local ones that can be estimated at each node (equipped with a microphone pair). Moreover, we simplify the exposition in [16] and omit the source index s from the indicator. Omitting the index s enables us to estimate the number of active sources and their locations without any a priori knowledge.

The *local* hidden variables, $y_m(t, k, \mathbf{p})$ can then be defined as the *indicator* of the time-frequency bin (t, k) associated with a speaker in location $\mathbf{p}$ from the mth microphone pair perspective:

$$y_m(t,k,\mathbf{p}) = \begin{cases} 1, & \mathbf{p} \text{ active for } (m,t,k) \\ 0, & \text{otherwise} \end{cases}. \tag{3.1}$$

For each time-frequency band (t,k) they equal to zero everywhere except to $\mathbf{p}$, the position of the active speaker, since not more than one speaker is likely to be active at each time-frequency band [110].

Please note that, in contrast to methods where global hidden variables are used [16], local hidden data support a case where some of the nodes measurements are physically unfeasible and hence assigned with zero probability. The expectation of this indicator is given by:

$$E\{y_m(t,k,\mathbf{p})\} = \psi_{\mathbf{p}}. \tag{3.2}$$

A vectorial version of the model proposed in [41] can be defined for the $2D$ (or $3D$) positioning problem, meaning that instead of a single node, local indicator, $y_m(t,k,\mathbf{p})$, a global vector of all local indicators can be used. Let $\mathbf{y}(t,k,\mathbf{p}) = \text{vec}_m\left(y_m(t,k,\mathbf{p})\right)$ be the set of all local indicators that relate to a certain time-frequency bin. The local components of this vector are independent identically distributed (i.i.d.). The expectation is therefore:

$$E\left\{\mathbf{y}(t,k,\mathbf{p})\right\} = \psi_{\mathbf{p}} \cdot \mathbf{1}, \tag{3.3}$$

where $\mathbf{1}$ is a vector of all ones of length M.

We can further define a longer vector $\mathbf{y}(\mathbf{p}) = \text{vec}_{t,k}\left(\mathbf{y}(t,k,\mathbf{p})\right)$ as the set of all indicators. Assuming independence in time, frequency and node, the probability density function of $\mathbf{y}$ is given by:

$$f(\mathbf{Y} = \mathbf{y}; \boldsymbol{\psi}) = \prod_{t,k,m} \sum_{\mathbf{p}\in\mathcal{P}} \psi_{\mathbf{p}} y_m(t,k,\mathbf{p}). \tag{3.4}$$

Evidently, $y_m(t,k,\mathbf{p})$ is an indicator. The number of local indicators is $M \times T \times K$. Their support is $\mathbf{p} \in \mathcal{P}$, the set of all discrete position values on the grid, and $|\mathcal{P}|$ stands for its cardinality.

Following chapter 9 of [47], the p.d.f. of the observations is given by:

$$
\begin{aligned}
f(\boldsymbol{\Phi} = \boldsymbol{\phi} | \mathbf{Y} = \mathbf{y}; \boldsymbol{\psi}) = \prod_{m} f(\boldsymbol{\Phi}_m = \boldsymbol{\phi}_m | \mathbf{Y}_m = \mathbf{y}_m; \boldsymbol{\psi}) = \\
\prod_{m,t,k} \sum_{\mathbf{p} \in \mathcal{P}} y_m(t,k,\mathbf{p}) \mathcal{N}^c \left(\phi_m(t,k); \tilde{\phi}_m^k(\mathbf{p}), \sigma^2 \right).
\end{aligned}
\tag{3.5}
$$

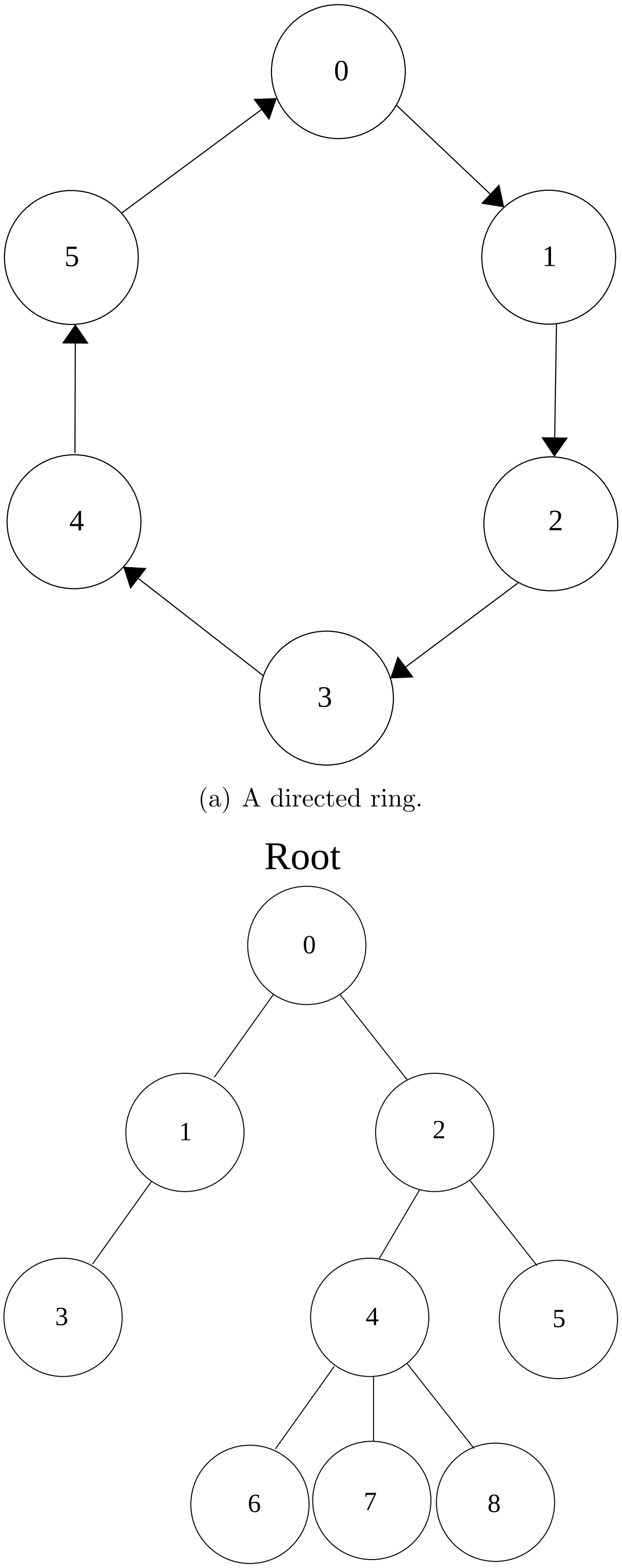
The p.d.f. of the *complete data* can be deduced from (3.4)-(3.5) and some simplification utilizing the indicator properties:

$$
\begin{aligned}
& f(\boldsymbol{\Phi} = \boldsymbol{\phi}, \mathbf{Y} = \mathbf{y}; \boldsymbol{\psi}) = f(\mathbf{Y} = \mathbf{y}; \boldsymbol{\psi}) f(\boldsymbol{\Phi} = \boldsymbol{\phi} | \mathbf{Y} = \mathbf{y}; \boldsymbol{\psi}) \\
&= \left(\prod_{m,t,k} \sum_{\mathbf{p} \in \mathcal{P}} \psi_{\mathbf{p}} y_m(t,k,\mathbf{p}) \right) \left(\prod_{m,t,k} \sum_{\mathbf{p} \in \mathcal{P}} y_m(t,k,\mathbf{p}) \mathcal{N}^c \left(\phi_m(t,k); \tilde{\phi}_m^k(\mathbf{p}), \sigma^2 \right) \right) \\
&= \prod_{m,t,k} \sum_{\mathbf{p} \in \mathcal{P}} \psi_{\mathbf{p}} y_m(t,k,\mathbf{p}) \mathcal{N}^c \left(\phi_m(t,k), \tilde{\phi}_m^k(\mathbf{p}), \sigma^2 \right).
\end{aligned}
\tag{3.6}
$$

3.2 Localization over a directed ring

We develop this DEM procedure based on the IEM framework. The algorithm enables localization of the speakers without a central point. Taking advantage of the distributed constellation of the sensors, we propose a distributed algorithm that enables multiple processing nodes and considers communication constraints between them. To develop a distributed version, we adopt the IEM method presented in [51]. The IEM is updating the hidden variables node-by-node, as new data arrive. This is a different concept from the batch EM solution. The IEM is better than the conventional EM algorithm in two important aspects, namely significantly improved convergence speed [51, 53], and weaker dependency on initialization [53, 52]. Note that one of the major drawbacks of the EM mechanism is that it might be trapped in a local optimum, depending on the initial conditions. Hence, decreasing dependency on initial conditions is highly important.

The proposed distributed implementation is over a directed ring topology, shown in Fig. 3.1(a). In the IDEM algorithm each node contributes to the estimation incrementally, by *locally* updating the hidden variables, rather than the *global* update, as in the conventional EM. The resulting algorithm is capable of detecting the number of active sources (including the detection of no activity) and their locations.



(a) A directed ring.

(b) Bi-directional tree.

Figure 3.1: Two network topologies used for our distributed expectation-maximization algorithms.

The EM algorithm for the problem at hand can now be derived. The *E-step* is given by:

$$\begin{aligned} Q(\boldsymbol{\psi}|\hat{\boldsymbol{\psi}}^{(\ell-1)}) \triangleq E\left\{\log\left(f(\boldsymbol{\Phi}=\boldsymbol{\phi},\boldsymbol{Y}=\mathbf{y};\boldsymbol{\psi})\right)|\boldsymbol{\phi};\hat{\boldsymbol{\psi}}^{(\ell-1)}\right\} = \\ \sum_{m,t,k,\mathbf{p}} E\left\{y_m(t,k,\mathbf{p})|\phi_m(t,k);\hat{\boldsymbol{\psi}}^{(\ell-1)}\right\}\cdot\left[\log\psi_{\mathbf{p}}+\log\mathcal{N}^c(\phi_m(t,k);\tilde{\phi}^k_m(\mathbf{p}),\sigma^2)\right], \end{aligned} \tag{3.7}$$

which in our case, simplifies to the calculation of:

$$\upsilon^{(\ell)}_m(t,k,\mathbf{p}) \triangleq E\left\{y_m(t,k,\mathbf{p})|\phi_m(t,k);\hat{\boldsymbol{\psi}}^{(\ell-1)}\right\} = \frac{\hat{\psi}^{(\ell-1)}_{\mathbf{p}}\mathcal{N}^c\left(\phi_m(t,k);\tilde{\phi}^k_m(\mathbf{p}),\sigma^2\right)}{\sum_{\tilde{\mathbf{p}}}\hat{\psi}^{(\ell-1)}_{\tilde{\mathbf{p}}}\mathcal{N}^c\left(\phi_m(t,k);\tilde{\phi}^k_m(\tilde{\mathbf{p}}),\sigma^2\right)}. \tag{3.8}$$

Note that $\upsilon^{(\ell)}_m(t,k,\mathbf{p})$ is a node-specific entity.

We can define aggregation of the hidden variables estimate:

$$\mu^{(\ell)}(t,k,\mathbf{p}) \triangleq \frac{1}{M}\sum_m \upsilon^{(\ell)}_m(t,k,\mathbf{p}), \tag{3.9}$$

which can be used to estimate $Q(\boldsymbol{\psi}|\hat{\boldsymbol{\psi}}^{(\ell-1)})$. It can be observed that $\mu^{(\ell)}(t,k,\mathbf{p})$ is the estimated probability that a source is present at time t, frequency k and position $\mathbf{p}$. The EM algorithm iterates between estimating the local node probabilities and then updating the global priors $\psi^{(l)}_p$ of the source positions for the next iteration.

Two paradigm can be adopted in calculating the MLE. The paradigm relevant to this section is denoted IDEM. Another paradigm, the batch version, is described in the next section, and will be denoted BDEM.

The IDEM algorithm is based on the partial (also denoted IEM) procedure [51]. It consists of partial *E-step* updates at each node followed by an *M-step*. In this way each component of the hidden vector is estimated incrementally with the most updated values of the parameters. The IDEM algorithm is detailed below.

For BW reduction and for the case where only the parameters estimation is of interest, we define an averaged version of the hidden variables in the following way:

$$\bar{\upsilon}^{(i)}_m(\mathbf{p}) \triangleq \frac{\sum_{t,k}\upsilon^{(i)}_m(t,k,\mathbf{p})}{TK}, \tag{3.10}$$

where i is the partial iteration index.

When the hidden variables are also of interest, like for the BSS application, this BW reduction is possible only for the localization phase of the algorithm. These applications are discussed in the next chapters.

The averages of the hidden variables are aggregated around the ring according to the following relations:

$$\hat{\boldsymbol{\psi}}_{\mathbf{p}}^{(i-1)} \triangleq \bar{\mu}^{(i-1)}(\mathbf{p}) = \frac{\sum_{m=1}^{M} \bar{\upsilon}_m^{(i-m)}(\mathbf{p})}{M}. \tag{3.11}$$

From this equation we derive the following compact version of the *M-step*:

$$\bar{\mu}^{(i)}(\mathbf{p}) = \bar{\mu}^{(i-1)}(\mathbf{p}) - \frac{\bar{\upsilon}_m^{(i-M)}(\mathbf{p})}{M} + \frac{\bar{\upsilon}_m^{(i)}(\mathbf{p})}{M}. \tag{3.12}$$

The IDEM, similarly to [16], starts by extracting local complex phases $\phi_m(t,k)$ from the received signals at each node and calculating $\tilde{\phi}_m^k(\mathbf{p})$ using (2.6). Then, the local hidden variables, $\upsilon_m^{(-M+1)}(t,k,\mathbf{p}) \ldots \upsilon_m^{(0)}(t,k,\mathbf{p})$ are initialized to a uniform distribution. To finalize the initialization stage, the local hidden variables are aggregated to yield $\bar{\mu}^{(0)}(\mathbf{p})$.

The algorithm then starts iterating for $\ell = 1,\ldots,L$ and $m = 1,\ldots,M$, where L is a predefined number of iterations of the entire network. Note, that in the proposed IDEM algorithm we have M node-specific updates before increasing the value of ℓ. The two iteration indices can be combined to one index $i = (\ell-1)M + m$ denoting the partial iteration. At each node the local hidden estimation (partial *E-step*) is followed by an *M-step*. Finally, the number of active sources J is determined as the number of values of $\psi_{\mathbf{p}}^{(i=L\cdot M)}$ that are higher than a predefined threshold. The coordinates of these Gaussians determine the location of the sources. Algorithm 1 summarizes the IDEM algorithm.

The IDEM exhibits two important advantages compared to previously suggested centralized algorithms. Firstly, the choice of new hidden variables and the partial update mechanism have the potential of improving the convergence rate of the algorithm. Secondly, it reduces communication bandwidth requirements between nodes.

3.3 Tree-based distributed expectation-maximization

In this section the proposed tree-based DEM algorithms are derived. We start by listing the drawbacks of a directed-ring topology, and show that they can be circumvented by using a bi-directional tree topology. We then derive two DEM algorithms suitable for that topology. Firstly, a batch algorithm, denoted BDEM, is proposed. Secondly, a recursive version of the BDEM algorithm, denoted RDEM, is derived.

Algorithm 1 Acoustic source localization with the incremental distributed expectation-maximization algorithm.

```
Obtain $z_m^1(t,k)$ and $z_m^2(t,k)$; $\forall m$.
Calculate $\phi_m(t,k)$; $\forall m$ using (2.2).
set $\tilde{\phi}_m^k(\mathbf{p})$ using (2.6).
initialize $\bar{\upsilon}_m^{(-M+1)}(\mathbf{p}), \ldots, \bar{\upsilon}_m^{(0)}(\mathbf{p})$.
Calculate $\bar{\mu}^{(0)}(\mathbf{p}) \triangleq \frac{1}{M}\sum_{m=1}^{M} \bar{\upsilon}_m^{(-m+1)}(\mathbf{p})$.
for $\ell = 1$ to $L$ do
    for $m = 1$ to $M$ do
        $i = (\ell - 1)M + m$ (partial iteration index)
        $\hat{\psi}_{\mathbf{p}}^{(i-1)} \triangleq \bar{\mu}^{(i-1)}(\mathbf{p})$ is received from previous node.
        E-step
        Calculate $\upsilon_m^{(i)}(t,k,\mathbf{p})$ using (3.8) with index $\ell$ replaced by index $i$.
        M-step
        Update: $\bar{\mu}^{(i)}(\mathbf{p})$ using (3.12) and send it to the next node in the ring.
    end
end
By applying a threshold to $\hat{\psi}_{\mathbf{p}}^{(L\cdot M)}$, the final estimate of the number of sources $J$, and their respective locations $\mathbf{p_j}$; $j = 1, \ldots, J$ can be obtained.
```

3.3.1 Comparison of directed ring and bi-directional tree topologies

In this subsection we compare two well-known network topologies [208] that are used by our DEM algorithms. They are schematically depicted in Fig. 3.1.

As mentioned above, the IDEM algorithm is implemented over a directed-ring topology, which suffers from several structural drawbacks that hinder its application to distributed algorithms. The two topologies can be compared as follows:

Latency The message propagation latency of a directed-ring linearly increases with the number of nodes, M. For the bi-directional tree-based topology the respective latency depends on $\log(M)$, where the base of the log depends on the structure of the bi-directional tree (i.e. the number of branches of each node).

Computational resources Computations in the directed-ring cannot be carried out in parallel, since each node has to wait for the result from the previous node. In bi-directional tree topologies all computations can be carried out in parallel.

Number of fatal failure points In a directed-ring topology, a disruption in any node disconnects the entire data flow. For the bi-directional tree topology, the only fatal failure point is the root. Moreover, even in the case of a root failure there are standard measures that can be taken to overcome this failure. This issue is beyond the scope.

Unlike the simple role of the nodes in the IDEM algorithm, implemented over a directed-ring, the role of the nodes in the proposed algorithms, which are based on a bi-directional tree, is more complex. On the one hand, the nodes of the tree are identical with respect to their sensing capabilities. On the other hand, they differ with respect to their role in the communication network according to their location in the tree topology. For example, the root of the tree (marked '0' in Fig. 3.1) plays a unique role in terms of management of the network, as will be described below.

3.3.2 Tree-based batch distributed expectation-maximization algorithm

We now derive the bi-directional tree based BDEM algorithm. The term *batch* refers to the specific processing with respect to the time axis. Unlike the IDEM (described above), which updates the estimation after partial processing of the data and the RDEM (described in the next subsection), which updates the estimation along the time axis (on-line processing), the BDEM processes all samples together after all of them have been acquired (off-line).

Assume that $\bar{\mu}^{(\ell)}(\mathbf{p})$, the recent average estimation of the hidden variables, is available in the root of the bi-directional tree. This is actually the result of the *M-step*, yielding the current parameter estimation:

$$\hat{\psi}_{\mathbf{p}}^{(\ell)} \triangleq \bar{\mu}^{(\ell)}(\mathbf{p}) = \frac{\sum_m \bar{\upsilon}_m^{(\ell)}(\mathbf{p})}{M}. \tag{3.13}$$

In the BDEM algorithm, the sub-iteration index i (see Algorithm 1) is no longer used, and only the global iteration index, ℓ is required. The *M-step* is actually the aggregation of local estimations from the leaves towards the root. The current parameter estimate, $\hat{\psi}_{\mathbf{p}}^{(\ell)}$ is diffused in the network until it reaches all leaves (nodes).

The *E-step* is executed in all leaves simultaneously, based on the current parameter estimation. They transmit their *E-step* results to the root and the new *M-step* estimation is aggregated. The new parameter estimation is then diffused back to the leaves.

Like the IDEM algorithm, each node is responsible for its local measurements and for

Algorithm 2 Acoustic source localization with the batch distributed expectation-maximization algorithm.

```
Obtain $z_m^1(t,k)$ and $z_m^2(t,k)$; $\forall m$.
Calculate $\phi_m(t,k)$; $\forall m$ using (2.2).
set $\tilde{\phi}_m^k(\mathbf{p})$ using (2.6).
initialize $\upsilon_m^{(0)}(t,k,\mathbf{p})$.
Calculate $\bar{\mu}^{(0)}(\mathbf{p}) = \frac{\sum_m \bar{\upsilon}_m^{(0)}(\mathbf{p})}{M}$.
for $\ell = 1$ to $L$ do
    $\hat{\psi}_{\mathbf{p}}^{(\ell-1)} \triangleq \bar{\mu}^{(\ell-1)}(\mathbf{p})$ is received from the root.
    E-step
    $\forall m = 1 : M$ calculate simultaneously and locally $\upsilon_m^{(\ell)}(t,k,\mathbf{p})$ using (3.8).
    M-step
    Calculate by the tree: $\bar{\mu}^{(\ell)}(\mathbf{p})$ using (3.13) and send it from the root to the leaves.
end
By applying a threshold to $\hat{\psi}_{\mathbf{p}}^{(L)}$, the final estimation of the number of sources $J$, and their respective locations $\mathbf{p_j}$; $j = 1, \ldots, J$ can be obtained.
```

estimating its local hidden variables. However, unlike the IDEM algorithm, the *E-step* can be executed simultaneously in all nodes using the *same* value of the parameters. The *E-step* is locally applied at all nodes in a standard manner, as opposed to the partial E-Step in Algorithm 1. The average of all estimations of the *local* hidden variables are fused towards the root. The root is then ready to send the next parameters estimation. The algorithm iterates until convergence or until the number of pre-defined iterations has been reached. Algorithm 2 summarizes the proposed BDEM algorithm.

3.3.3 Tree-based recursive distributed expectation-maximization algorithm

A recursive version of the BDEM, denoted RDEM, is derived below. Even for the static scenario, a recursive on-line algorithm can help reduce latency and computations. As shown experimentally, the RDEM also exhibits fast convergence and high accuracy. Recursive algorithms are often used also for tracking a dynamic scenario as explored later. In this chapter we concentrate on static scenarios.

Recursive EM versions have been derived in [184, 54]. We adopt the TREM version [54].

The basic adaptation scheme for the parameter of interest, $\boldsymbol{\psi}$ is given by:

$$\hat{\boldsymbol{\psi}}_R^{(t)} = \hat{\boldsymbol{\psi}}_R^{(t-1)} + \gamma_t \boldsymbol{I}^{-1}_{\boldsymbol{y}_t,\boldsymbol{\phi}_t;\hat{\boldsymbol{\psi}}_R^{(t-1)}} \nabla_{\boldsymbol{\psi}} \log f(\boldsymbol{\Phi}_t = \boldsymbol{\phi}_t; \boldsymbol{\psi})|_{\hat{\boldsymbol{\psi}}_R^{(t-1)}}, \tag{3.14}$$

where subscript t in the notation V_t stands for all components of a certain vector, V of the current time frame. The Fisher information matrix (FIM) is defined as:

$$\boldsymbol{I}_{\boldsymbol{y}_t,\boldsymbol{\phi}_t;\hat{\boldsymbol{\psi}}_R^{(t-1)}} \triangleq -E\left\{\nabla^2_{\boldsymbol{\psi}} \log f(\boldsymbol{Y}_t = \boldsymbol{y}_t, \boldsymbol{\Phi}_t = \boldsymbol{\phi}_t; \boldsymbol{\psi})|_{\hat{\boldsymbol{\psi}}_R^{(t-1)}}\right\}. \tag{3.15}$$

We stress that the operator, $E\{\cdot\}$ is a conditional expectation using the current parameter estimate $\hat{\boldsymbol{\psi}}_R^{(t)}$. We therefore use the constrained version of TREM proposed in [16]:

$$\begin{aligned}\hat{\boldsymbol{\psi}}_R^{(t)} = \hat{\boldsymbol{\psi}}_R^{(t-1)} &+ \gamma_t \boldsymbol{I}^{-1}_{\boldsymbol{y}_t,\boldsymbol{\phi}_t;\hat{\boldsymbol{\psi}}_R^{(t-1)}} \nabla_{\boldsymbol{\psi}} \log f(\boldsymbol{\Phi}_t = \boldsymbol{\phi}_t; \boldsymbol{\psi})|_{\hat{\boldsymbol{\psi}}_R^{(t-1)}} \\ &- \gamma_t \frac{\boldsymbol{I}^{-1}_{\boldsymbol{y}_t,\boldsymbol{\phi}_t;\hat{\boldsymbol{\psi}}_R^{(t-1)}} \boldsymbol{a}}{\boldsymbol{a}^T \boldsymbol{I}^{-1}_{\boldsymbol{y}_t,\boldsymbol{\phi}_t;\hat{\boldsymbol{\psi}}_R^{(t-1)}} \boldsymbol{a}} \left(\boldsymbol{a}^T \hat{\boldsymbol{\psi}}_R^{(t-1)} + \boldsymbol{a}^T \boldsymbol{I}^{-1}_{\boldsymbol{y}_t,\boldsymbol{\phi}_t;\hat{\boldsymbol{\psi}}_R^{(t-1)}} \nabla_{\boldsymbol{\psi}} \log f(\boldsymbol{\Phi}_t = \boldsymbol{\phi}_t; \boldsymbol{\psi})|_{\hat{\boldsymbol{\psi}}_R^{(t-1)}} - b\right),\end{aligned} \tag{3.16}$$

where in this case, due to the constraints in (2.7) we have:

$$\boldsymbol{a} = \mathbf{1}^T, b = 1. \tag{3.17}$$

To calculate the FIM, expectation of the second derivative of the log likelihood is required. The joint p.d.f. of the *instantaneous* measurements and the hidden variables (compare to (3.6)) is given by:

$$f(\boldsymbol{Y}_t = \boldsymbol{y}_t, \boldsymbol{\Phi}_t = \boldsymbol{\phi}_t; \boldsymbol{\psi}) = \prod_{m,k} \sum_{\mathbf{p}\in\mathcal{P}} \psi_{\boldsymbol{p}} y_m(t,k,\boldsymbol{p}) \mathcal{N}^C\left(\phi_m(t,k), \tilde{\phi}_m^k(\boldsymbol{p}), \sigma^2\right). \tag{3.18}$$

Applying the log operation and the indicator properties yields:

$$\begin{aligned}&\log f(\boldsymbol{Y}_t = \boldsymbol{y}_t, \boldsymbol{\Phi}_t = \boldsymbol{\phi}_t; \boldsymbol{\psi}) = \\ &\quad \sum_{m,k,\mathbf{p}\in\mathcal{P}} y_m(t,k,\boldsymbol{p}) \left(\log(\psi_{\boldsymbol{p}}) + \log\left(\mathcal{N}^C\left(\phi_m(t,k), \tilde{\phi}_m^k(\boldsymbol{p}), \sigma^2\right)\right)\right).\end{aligned} \tag{3.19}$$

Evaluating the second derivative in the previous parameter estimate and using definition (2.10) yield:

$$-\frac{\partial^2}{\partial \psi_{\boldsymbol{p}}^2} \log f(\boldsymbol{Y}_t = \boldsymbol{y}_t, \boldsymbol{\Phi}_t = \boldsymbol{\phi}_t; \boldsymbol{\psi})|_{\hat{\psi}_R^{(t-1)}} = \frac{\sum_{k,m} y_m(t,k,\boldsymbol{p})}{\left(\hat{\psi}_{\boldsymbol{p},R}^{(t-1)}\right)^2}. \tag{3.20}$$

Taking expectation and utilizing the local indicator property result in:

$$E\left\{-\frac{\partial^2}{\partial \psi_{\boldsymbol{p}}^2} \log f(\boldsymbol{Y}_t = \boldsymbol{y}_t, \boldsymbol{\Phi}_t = \boldsymbol{\phi}_t; \boldsymbol{\psi})|_{\hat{\psi}_R^{(t-1)}}\right\} = \frac{K \cdot M}{\left(\hat{\psi}_{\boldsymbol{p},R}^{(t-1)}\right)}. \tag{3.21}$$

The p.d.f. of the current observation is given by (comparing to (2.8) and (2.9)):

$$f(\boldsymbol{\Phi}_t = \boldsymbol{\phi}_t; \boldsymbol{\psi}) = \prod_{m,k} \sum_{\mathbf{p} \in \mathcal{P}} \psi_{\mathbf{p}} \mathcal{N}^c \left(\phi_m(t,k); \tilde{\phi}_m^k(\mathbf{p}), \sigma^2\right). \tag{3.22}$$

Taking the logarithm of (3.22) and calculating the derivative yield:

$$\frac{\partial}{\partial \psi_{\boldsymbol{p}}} \log f(\boldsymbol{\Phi_t} = \boldsymbol{\phi_t}; \boldsymbol{\psi}) = \sum_{m,k} \frac{\mathcal{N}^C \left(\phi_m(t,k), \tilde{\phi}_m^k(\boldsymbol{p}), \sigma^2\right)}{\sum_{\tilde{\mathbf{p}} \in \mathcal{P}} \psi_{\tilde{\mathbf{p}}} \mathcal{N}^C \left(\phi_m(t,k), \tilde{\phi}_m^k(\tilde{\mathbf{p}}), \sigma^2\right)}. \tag{3.23}$$

Evaluating (3.23) at the previous parameter estimate and expressing the result in terms of local expressions yield:

$$\frac{\partial}{\partial \psi_{\boldsymbol{p}}} \log f(\boldsymbol{\Phi_t} = \boldsymbol{\phi_t}; \boldsymbol{\psi})|_{\hat{\psi}_R^{(t-1)}} = \sum_{m,k} \frac{\upsilon_m^{(t)}(k,\boldsymbol{p})}{\hat{\psi}_{\boldsymbol{p},R}^{(t-1)}}, \tag{3.24}$$

where $\upsilon_m^{(t)}(k,\mathbf{p})$, the local hidden variables, are calculated using the approximation of the parameters:

$$\upsilon_m^{(t)}(k,\mathbf{p}) \triangleq \frac{\hat{\psi}_{\mathbf{p},R}^{(t-1)} \mathcal{N}^c \left(\phi_m(t,k); \tilde{\phi}_m^k(\mathbf{p}), \sigma^2\right)}{\sum_{\tilde{\mathbf{p}} \in \mathcal{P}} \hat{\psi}_{\tilde{\mathbf{p}},R}^{(t-1)} \mathcal{N}^c \left(\phi_m(t,k); \tilde{\phi}_m^k(\tilde{\mathbf{p}}), \sigma^2\right)}. \tag{3.25}$$

We define the frequency mean of the local hidden variables as:

$$\bar{\upsilon}_m^{(t)}(\boldsymbol{p}) \triangleq \frac{1}{K} \sum_{k=1}^{K} \upsilon_m^{(t)}(k,\boldsymbol{p}). \tag{3.26}$$

Averaging over all frequencies at each node enables a significant reduction of the communication BW.

We define the current global parameter estimation (before recursion) as the multiplication of (3.21) and (3.24):

$$\hat{\psi}_{\boldsymbol{p}}^{(t)} \triangleq \frac{1}{M}\sum_{m} \bar{\upsilon}_{m}^{(t)}(\boldsymbol{p}) \triangleq \bar{\mu}^{(t)}(\mathbf{p}) = \left(E\left\{-\frac{\partial^2}{\partial\psi_{\boldsymbol{p}}^2}\log f(\boldsymbol{Y}_t=\boldsymbol{y}_t,\boldsymbol{\Phi}_t=\boldsymbol{\phi}_t;\boldsymbol{\psi})|_{\hat{\psi}_{\boldsymbol{p},R}^{(t-1)}}\right\}\right)^{-1}\frac{\partial}{\partial\psi_{\boldsymbol{p}}}\log f(\boldsymbol{\phi}_{\boldsymbol{t}};\boldsymbol{\psi})|_{\hat{\psi}_{\boldsymbol{p},R}^{(t-1)}}. \tag{3.27}$$

Using the term from equation (3.27), the recursive distributed estimation procedure from equation (3.16), for the case of (3.17) results in the following simple form:

$$\hat{\boldsymbol{\psi}}_R^{(t)} = \hat{\boldsymbol{\psi}}_R^{(t-1)} + \gamma_t\left(\hat{\boldsymbol{\psi}}^{(t)} - \hat{\boldsymbol{\psi}}_R^{(t-1)}\right). \tag{3.28}$$

This simple recursive calculation is applied by the root.

The global parameters of the algorithm $\hat{\boldsymbol{\psi}}_R^{(t)}$ are initialized by a uniform p.d.f.. Based on the recent parameter estimate (using the initialization at $t=0$) a local *E-step* is executed at every node of the bi-directional tree at the current time frame, yielding $\bar{\upsilon}_m^{(t)}(\mathbf{p})$.

This operation may, at the first glance, appear identical to the *E-step* of the BDEM algorithm. However, as an off-line algorithm, the BDEM calculates the *E-step* for all time frames together. Its recursive version, the RDEM, processes each time frame separately immediately after being acquired.

The local hidden variables are averaged along the frequency axis locally and $\bar{\upsilon}_m^{(t)}(\boldsymbol{p})$ is aggregated through the bi-directional tree towards the root. The results are used by the root for the calculation of its current estimate (3.27) and the recursion equation (3.28). The root then transmits (diffuses) the updated parameters to all nodes through the tree before the next time frame is processed. This process is summarized in Algorithm 3.

The root of the bi-directional tree has two roles in this process. Its first role is to broadcast the latest global parameter estimation $\hat{\boldsymbol{\psi}}_R^{(t)}$ to all nodes efficiently. The second role of the root (in the opposite direction) is to aggregate local results that will enable calculation of the next recursion step.

The advantages of a bi-directional tree structure compared to a directed-ring based structure (used by the IDEM algorithm) were discussed above. To summarize this section, some of the RDEM advantages over the non-recursive algorithms (IDEM and BDEM) are described. First, the RDEM is an on-line algorithm that does not need to store previous estimates of

Algorithm 3 Acoustic source localization with the recursive distributed expectation-maximization algorithm.

```
set $\tilde{\phi}_m^k(\boldsymbol{p})$ using (2.6).
initialize $\hat{\boldsymbol{\psi}}_R^{(0)}$.
for $t = 1$ to $T$ do
    Obtain $z_m^1(t,k)$ and $z_m^2(t,k)$; $\forall m$.
    Calculate $\phi_m(t,k)$; $\forall m$ using (2.2).
    Calculate simultaneously and locally $\bar{\upsilon}_m^{(t)}(\boldsymbol{p})$ using (3.26) $\forall m = 1, \dots, M$.
    Use the bi-directional tree to aggregate local results from leaves (fusion).
    Calculate by the root processor: $\hat{\boldsymbol{\psi}}_R^{(t)}$ using (3.28).
    Transmit $\hat{\boldsymbol{\psi}}_R^{(t)}$ from the root to all nodes for the next time frame (diffusion).
end
Find $J$, the number of sources, and their respective locations $\mathbf{p_j}$; $j = 1, \dots, J$ by applying a threshold to $\hat{\boldsymbol{\psi}}_R^{(T)}$, which is the final result of the algorithm.
```

the hidden variables. Second, transmitting only the averaged hidden variables (along the frequency axis) makes it BW efficient even for large values of K (the number of frequencies). Third, replacing the iteration index with the time index decreases the computational complexity and the communication BW. In the following sections we compare these algorithms in terms of implementation and performance.

3.4 Localization implementation analysis

This section deals with various implementation features of the DEM algorithms. The factors that influence these implementation features are summarized in Table 3.1.

Notice that we selected a regular grid of possible locations with a desired resolution, but other schemes are applicable as well. The grid and its resolution have an influence on the computational complexity.

Table 3.2 summarizes the computational complexity, latency, communication BW and memory requirements of the proposed algorithms.

The comparison between IDEM and BDEM is quite complicated. On the one hand, BDEM requires higher communication BW than IDEM and it has higher computational complexity (since $L_{\text{BDEM}} > L_{\text{IDEM}}$). On the other hand, its memory requirements and its latency (assuming $M > \frac{L_{\text{BDEM}}}{L_{\text{IDEM}}}$) are smaller.

When comparing the RDEM to the first two DEM variants, a significant improvements

Notation	Meaning
L_{BDEM} or L_{IDEM}	number of iterations for each algorithm (typically larger for BDEM)
M	number of nodes
T	number of time frames
K	number of frequency bins
$\|P\|$	size of the positions grid
T_{B}	block length in seconds required for reliable localization of the IDEM and BDEM
T_{Global}	latency caused by the global calculations of IDEM and BDEM
$T^{R}_{\mathrm{Global}} < T_{\mathrm{Global}}$	latency caused by the global calculations of RDEM
T_{Local}	latency resulting from the local calculations of IDEM and BDEM
$T^{R}_{\mathrm{Local}} < T_{\mathrm{Local}}$	latency resulting from the local calculations of RDEM

Table 3.1: Implementation factors definition.

Criteria	IDEM	BDEM	RDEM
Computation	$\mathcal{O}(L_{\mathrm{IDEM}} \cdot M \cdot T \cdot K \cdot \|P\|)$	$\mathcal{O}(L_{\mathrm{BDEM}} \cdot M \cdot T \cdot K \cdot \|P\|)$	$\mathcal{O}(M \cdot T \cdot K \cdot \|P\|)$
Latency	$\mathcal{O}(T_{\mathrm{B}} + L_{\mathrm{IDEM}} \cdot M \cdot T_{\mathrm{Global}} + L_{\mathrm{IDEM}} \cdot M \cdot T_{\mathrm{Local}})$	$\mathcal{O}(T_{\mathrm{B}} + L_{\mathrm{BDEM}} \cdot T_{\mathrm{Global}} + L_{\mathrm{BDEM}} \cdot T_{\mathrm{Local}})$	$\mathcal{O}(T^{R}_{\mathrm{Global}} + T^{R}_{\mathrm{Local}})$
BW	$\mathcal{O}(L_{\mathrm{IDEM}} \cdot M \cdot \|P\|)$	$\mathcal{O}(L_{\mathrm{BDEM}} \cdot M \cdot \|P\|)$	$\mathcal{O}(M \cdot T \cdot \|P\|)$
Memory	$\mathcal{O}(M \cdot T \cdot K \cdot \|P\|)$	0	$\mathcal{O}(\|P\|)$

Table 3.2: Implementation feature table. The incremental distributed expectation-maximization, batch distributed expectation-maximization and recursive distributed expectation-maximization are compared with respect to computation, delay, communication band width and memory.

in computation complexity and latency is observed. The RDEM requires more BW. Its memory requirements are small, in fact much smaller than those of IDEM, but slightly larger than those of BDEM (which does not impose any storage requirements).

3.5 Experimental study

This section reports an experimental study of the two last DEM localization algorithms. As a reference algorithm we used a modified version of the SRP-PHAT [87]. It should be emphasized that localization in acoustic signal processing is usually developed for $1-D$ only. This is the reason we found it difficult to compare our results to more state-of-the-art algorithms.

In order to evaluate performance, we use both simulation and real-life recordings of concurrent sources. For simplicity, we limited the localization problem to the two-dimensional case. It should be noted that the algorithms can be applied to the three-dimensional case as well.

3.5.1 Practical considerations

There are a few practical considerations to be addressed regarding the proposed algorithms.

3.5.1.1 Sensor positions

As in most of the localization approaches, we also assume perfect knowledge of the sensors' positions in the room. As mentioned above, we assume a $2D$ set-up purely for simplicity reasons. Therefore, the elevation value is ignored. However, the algorithms tested can be easily applied to $3D$ cases as well. In the general case a $3D$ location can be estimated.

3.5.1.2 Node synchronization

We assume perfect synchronization between the nodes. In practice, it is most likely to have synchronization since cell phones and other equipment are synchronized through the network. In cases where there are clock differences, synchronization methods such as [209] can be adapted. Actually, later we discovered that our distributed algorithms are not sensitive to node synchronization errors.

3.5.1.3 Microphone inter-distance in each node

We used a microphone inter-distance of 50 cm, which is a good compromise between resolution and ambiguity.

3.5.1.4 Calculation precision

For all calculations we used natural log operations, since they convert multiplications and divisions into additions and subtractions, while maintaining high precision. In other words, the probability $\hat{\psi}_{\mathbf{p}}^{(i)}$ is replaced by $\log\left(\hat{\psi}_{\mathbf{p}}^{(i)}\right)$. In a similar way, $\log\left(\mu^{(i)}(t,k,\mathbf{p})\right)$ is used rather than its original counterpart.

3.5.1.5 Number of iterations for batch distributed expectation-maximization

We used 20 iterations for the BDEM algorithm. If resources are highly limited, the number of iterations can be reduced to around 10.

3.5.2 Simulation results

To evaluate the localization performance of the algorithms, we simulated the following scenario. Twelve pairs of omni-directional microphones were located around a room. The dimensions of the simulated room were $6 \times 6 \times 4$ m, with a low reverberation level, $T_{60} = 200$ msec. Two sources, randomly located in the room, were simulated using short speech files and an efficient implementation [210] of the image method [211]. An example of the speaker-microphone constellation is depicted in Fig. 3.2(a).

To compare the performance of the algorithms we followed the procedure described in [192]. We executed 100 Monte-Carlo trials and calculated three statistical measures: 1) The MD rate, defined as the percentage of sources that were miss-detected out of the total number of sources; 2) The FA rate, defined as the percent of falsely-detected sources normalized by the total number of sources; and 3) The mean square error (MSE), defined as the accuracy of localization for all successfully detected sources. Note that the accuracy of the location estimation was limited by the grid resolution which was 10×10 cm. Table 7.1 summarizes the measures for all algorithms. The reference algorithm SRP-PHAT [87] had a higher MD rate and a higher FA rate than the others. The MSE of all algorithms was low with respect to the grid resolution.

Algorithm	MD[%]	FA[%]	MSE[cm]
SRP-PHAT	7.5	11.5	4
BDEM	6.5	9.0	4
RDEM	6.5	8.5	4

Table 3.3: Localization statistics for 100 Monte-Carlo trials with two randomly located acoustic sources.

3.5.3 Analysis of actual recordings

The algorithms were also tested using real recordings of two simultaneous sources and nine synchronized microphone pairs. Real-life recordings are important to validate localization algorithms, since not everything can be accurately simulated. For example, the spatial volume of the sources is complicated to simulate.

The recordings were carried out in the speech and acoustic lab of Bar-Ilan University. This is a $6 \times 6 \times 2.4m$ m room that has a reverberation time controlled by 60 interchangeable panels covering the room facets.

To simulate real human sources, we used a mouth simulator (B&K, type 4227) and a head and torso simulator (HATS) mannequin (B&K, type 4128C-002) to emulate head and torso shadowing effects. The measurement equipment also included a RME Hammerfall DSP Digiface sound-card and a RME Octamic (for Microphone Pre Amp and digitization (A/D)). AKG type CK-32 omnidirectional microphones were used. All measurements were carried out with a sampling frequency of 48 KHz and a resolution of 24-bits. The multi-microphone signals were acquired using Matlab$^{\copyright}$. An example of the room layout is depicted in Fig. 3.2(b). Two different reverberation levels were tested by changing the panel configuration; namely $T_{60} = 150$ msec (low) and $T_{60} = 450$ msec (medium). For the higher reverberation level, the localization results were not accurate enough. A picture of the room setup, with two sources facing each other 61.5 cm apart and a low reverberation level, is depicted in Fig. 3.3.

The results of the localization algorithms are presented. We refer only to the center part of the room encircled by the microphones.

In the following figures we depict the results of the various algorithms. In all figures, only the area encircled by the microphones is shown. Figure 3.4 depicts the localization probability maps ($\hat{\psi}_{\mathbf{p}}$) of the proposed algorithms and the output of the SRP-PHAT algorithm for the low reverberation level and with a source inter-distance of 61.5 cm. The SRP-PHAT had poor resolution and exhibited a wide peak combining both sources. The BDEM and the RDEM algorithms detected the two sources. In addition, the localization of the RDEM was slightly more significant than the localization of the BDEM.

Figure 3.5 depicts a two source localization (inter-distance of 71.5 cm) for the medium reverberation level. The SRP-PHAT only detected one of the sources. Again, the BDEM and the RDEM algorithms detected both sources.

3.6 Speaker localization with moving microphone arrays

In practice, applications such as robot audition often require near real-time processing, such that sound sources must be localized from short frames of audio data. We use moving microphone arrays with known trajectories. We decided to adapt a Bayesian technique to sequentially estimate the source positions from on-line data. Bayesian estimation [212, 56] considers not only the likelihood of the desired random variables, but also incorporates prior information by modeling belief about the dynamics of the sources. The posterior p.d.f. is therefore maximized instead of the likelihood. In this case the prior imposes the static location of the source. The resulting MAP estimator can therefore be considered as a penalized ML approach. A particle filter is proposed for sequential sound source localization.

For both approaches, we assume that only a single, static source is localized and that the trajectory and positions of the microphones are known *a priori*.

3.6.1 The expectation-maximization localization algorithm

Based on [41] an EM localization algorithm has been suggested in [16] with a vector of PRP measurements. In [197] IDEM has been presented for the same localization problem. Following [197] and [16], we present an algorithm for the moving microphones case. The basic formalization was given in subsection 2.1.2.

3.6.1.1 Maximum likelihood for dynamic localization

The joint p.d.f. of the PRPs in (2.2), assuming independence along time and frequency indexes, is given by:

$$f(\boldsymbol{\Phi} = \boldsymbol{\phi}; \boldsymbol{\psi}) = \prod_{t,k} \sum_{\mathbf{p}} \psi_{\mathbf{p}} \mathcal{N}^c \left(\phi(t,k); \tilde{\phi}^k(\mathbf{p}, t), \sigma^2 \right), \tag{3.29}$$

where $\boldsymbol{\psi} = \text{vec}_{\mathbf{p}}(\psi_{\mathbf{p}})$ and $\boldsymbol{\phi} = \text{vec}_{t,k}(\phi(t,k))$.

The ML estimate of the source locations is given by:

$$\hat{\boldsymbol{\psi}} = \underset{\boldsymbol{\psi}}{\text{argmax}} [\log f(\boldsymbol{\Phi} = \boldsymbol{\phi}; \boldsymbol{\psi}) \text{ s.t.} \sum_{\mathbf{p} \in \mathcal{P}} \psi_{\mathbf{p}} = 1 \text{ and } 0 < \psi_{\mathbf{p}} < 1]. \tag{3.30}$$

3.6.1.2 Hidden variables

The hidden variables, $y(t,k,\mathbf{p})$ are defined as the association of each measurement with a source at position $\mathbf{p}$. Let $\mathbf{y} = \mathrm{vec}_{t,k,\mathbf{p}}\left(y(t,k,\mathbf{p})\right)$ be the vector concatenation of the hidden variables. The p.d.f. of $\mathbf{y}$ is given by:

$$f(\mathbf{Y}=\mathbf{y};\boldsymbol{\psi}) = \prod_{t,k}\sum_{\mathbf{p}} \psi_{\mathbf{p}} y\left(t,k,\mathbf{p}\right). \tag{3.31}$$

Given the hidden variables, the p.d.f. of the observations is:

$$f(\boldsymbol{\Phi}=\boldsymbol{\phi}|\mathbf{y};\boldsymbol{\psi}) = \prod_{t,k}\sum_{\mathbf{p}} y\left(t,k,\mathbf{p}\right)\mathcal{N}^c\left(\phi(t,k);\tilde{\phi}^k(\mathbf{p},t),\sigma^2\right). \tag{3.32}$$

The p.d.f. of the *complete data* can be deduced from (3.31)-(3.32):

$$f(\boldsymbol{\Phi}=\boldsymbol{\phi},\mathbf{Y}=\mathbf{y};\boldsymbol{\psi}) = \prod_{t,k}\sum_{\mathbf{p}} \psi_{\mathbf{p}} y\left(t,k,\mathbf{p}\right)\mathcal{N}^c\left(\phi(t,k),\tilde{\phi}^k\left(\mathbf{p},t\right),\sigma^2\right). \tag{3.33}$$

3.6.1.3 The expectation-maximization localization algorithm

The original IDEM algorithm is capable of detecting the number of active sources (including the detection of no activity) and their locations for static scenarios. The IDEM is applied here for moving sensors. Thanks to the dynamics of the sensors, we can use it even with a single node. It means that this version is not necessarily a distributed algorithm, although it can be easily expanded to more than a single node.

The *E-step* can be stated as:

$$\begin{aligned} Q(\boldsymbol{\psi}|\hat{\boldsymbol{\psi}}^{(\ell-1)}) &\triangleq E\left\{\log\left(f(\boldsymbol{\Phi}=\boldsymbol{\phi},\boldsymbol{Y}=\mathbf{y};\boldsymbol{\psi})\right)|\boldsymbol{\phi};\hat{\boldsymbol{\psi}}^{(\ell-1)}\right\} \\ &= \sum_{t,k,\mathbf{p}} E\left\{y(t,k,\mathbf{p})|\phi(t,k);\hat{\boldsymbol{\psi}}^{(\ell-1)}\right\}\left[\log\psi_{\mathbf{p}} + \log\mathcal{N}^c(\phi(t,k);\tilde{\phi}^k(\mathbf{p},t),\sigma^2)\right], \end{aligned} \tag{3.34}$$

which, in our case, simplifies to:

$$v^{(\ell)}\left(t,k,\mathbf{p}\right) \triangleq E\left\{y\left(t,k,\mathbf{p}\right)|\phi(t,k);\hat{\boldsymbol{\psi}}^{(\ell-1)}\right\} = \frac{\hat{\psi}_{\mathbf{p}}^{(\ell-1)}\mathcal{N}^c\left(\phi(t,k);\tilde{\phi}^k\left(\mathbf{p},t\right),\sigma^2\right)}{\sum_{\mathbf{p}'}\hat{\psi}_{\mathbf{p}'}^{(\ell-1)}\mathcal{N}^c\left(\phi(t,k);\tilde{\phi}^k\left(\mathbf{p}',t\right),\sigma^2\right)}. \tag{3.35}$$

The EM applies the E-step, followed by the M-step, as summarized in Algorithm 4. This summary is for a single node, although a generalized version can also be used.

Algorithm 4 Dynamic EM localization.

```
input $z^1(t,k)$, $z^2(t,k)$;.
Calculate $\phi(t,k)$ using (2.2).
set $\tilde{\phi}^k(\mathbf{p},t)$ using (2.15).
init $\hat{\psi}_{\mathbf{p}}^{(-1)}$ to uniform p.d.f..
Calculate $\upsilon^{(0)}(t,k,\mathbf{p})$ using (3.35).
Calculate their mean: $\hat{\psi}_{\mathbf{p}}^{(0)} = \frac{\sum_{t,k} \upsilon^{(0)}(t,k,\mathbf{p})}{T \cdot K}$ .
for $\ell = 1$ to $L$ do
    E-step
    Calculate $\upsilon^{(\ell)}(t,k,\mathbf{p})$ using (3.35).
    M-step
    Calculate $\hat{\psi}_{\mathbf{p}}^{(\ell)} = \frac{\sum_{t,k} \upsilon^{(\ell)}(t,k,\mathbf{p})}{T \cdot K}$.
end
output $\hat{\psi}_{\mathbf{p}}^{(L)}, \upsilon^{(L)}(t,k,\mathbf{p})$.
```

3.6.2 Bayesian filter

As discussed in the previous section, ML estimation infers knowledge about the source position from the observations only (see (3.30)). As only knowledge about the measured data is taken into account, ML estimators are based on purely objective observations. Prior belief about the source position can also be utilized when considering a Bayesian framework.

Under the Bayesian paradigm the desired source position, $\boldsymbol{p}$, is considered as a static state. Estimates of $\boldsymbol{p}$ can hence be obtained by construction of the posterior p.d.f. of the states, $f_t(\boldsymbol{p}|\boldsymbol{\phi}_{1:t})$, given the PRPs, $\boldsymbol{\phi}_{1:t} \triangleq \left[\boldsymbol{\phi}_1^T, \ldots, \boldsymbol{\phi}_t^T\right]^T$, where $\boldsymbol{\phi}_t \triangleq \left[\phi(t,1), \ldots, \phi(t,K)\right]^T$. This estimation is related to the likelihood, $f(\boldsymbol{\phi}_t|\boldsymbol{p})$, via Bayes's theorem:

$$f_t(\boldsymbol{p}|\boldsymbol{\phi}_{1:t}) = \frac{f(\boldsymbol{\phi}_t|\boldsymbol{p})\, f_{t|t-1}(\boldsymbol{p}|\boldsymbol{\phi}_{1:t-1})}{f(\boldsymbol{\phi}_t)}. \tag{3.36}$$

The instantaneous likelihood, $p\left(\boldsymbol{\phi}_t \mid \boldsymbol{p}\right)$ is modeled similar to (3.29) by assuming independence of PRPs in time and frequency:

$$p\left(\boldsymbol{\phi}_t \mid \boldsymbol{p}\right) = \prod_{k=1}^{K} \mathcal{N}^c\left(\phi(t,k); \tilde{\phi}^k(\boldsymbol{p},t), \sigma^2\right), \tag{3.37}$$

where σ^2 is the measurement noise variance.

Furthermore, $f_{t|t-1}(\boldsymbol{p}|\boldsymbol{\phi}_{1:t-1})$ in (3.36) is the predicted p.d.f. given by:

$$f_{t|t-1}(\boldsymbol{p}|\boldsymbol{\phi}_{1:t-1}) = \int\limits_{\mathbb{R}^2} f(\boldsymbol{p})\, f_{t-1}(\boldsymbol{p}|\boldsymbol{\phi}_{1:t-1}) d\boldsymbol{p}, \tag{3.38}$$

where $f(\boldsymbol{p})$ is the prior p.d.f. capturing the static nature of the source and $f_{t-1}(\boldsymbol{p}|\boldsymbol{\phi}_{1:t-1})$ is the posterior p.d.f. at time $t-1$.

To sequentially obtain the optimal value of $\boldsymbol{p}$ at each time, t, MAP estimates can be evaluated by maximization with respect to the variables of interest, i.e.,

$$\hat{\boldsymbol{p}} \triangleq \operatorname*{argmax}_{\boldsymbol{p}} f(\boldsymbol{p}|\boldsymbol{\phi}_{1:t}). \tag{3.39}$$

3.6.2.1 Sequential importance sampling

To impose real-valued source states despite the complex observations, sequential importance sampling [213] is used. The posterior at $t-1$ is approximated by:

$$f_{t-1}(\boldsymbol{p}|\boldsymbol{\phi}_{1:t-1}) = \sum_{j=1}^{J_{t-1}} w_{t-1}^{(j)}\, \delta_{\hat{\boldsymbol{p}}^{(j)}}(\boldsymbol{p}), \tag{3.40}$$

where $\delta_{\hat{\boldsymbol{p}}^{(j)}}(\boldsymbol{p})$ denotes the Dirac measure of random variable, $\boldsymbol{p}$, centered on particle $\hat{\boldsymbol{p}}^{(j)}$. $w_{t-1}^{(j)}$ are the weights updated along the sampling iterations and J_{t-1} is their number for previous iteration $t-1$. Inserting (3.40) into (3.38) yields:

$$f_{t|t-1}(\boldsymbol{p}|\boldsymbol{\phi}_{1:t-1}) = \int\limits_{\mathbb{R}^2} f(\boldsymbol{p}) \sum_{j=1}^{J_{t-1}} w_{t-1}^{(j)}\, \delta_{\hat{\boldsymbol{p}}^{(j)}}(\boldsymbol{p})\, d\boldsymbol{p}. \tag{3.41}$$

In order to capture the static nature of the source whilst modeling uncertainty in the particles, $\hat{\boldsymbol{p}}^{(j)}$, the prior, $f(\boldsymbol{p})$ is approximated by drawing P importance samples for each particles, $\hat{\boldsymbol{p}}^{(j)}$ from the proposal distribution:

$$\pi\left(\boldsymbol{p}|\hat{\boldsymbol{p}}^{(j)}\right) = \mathcal{N}\left(\boldsymbol{p}\,|\,\hat{\boldsymbol{p}}^{(j)},\, \mathbf{Q}\right) \tag{3.42}$$

where the covariance, $\mathbf{Q}$, allows for deviations of the new particles via (3.42), $\hat{\boldsymbol{p}}^{(j,p)}$ from the old particles, $\hat{\boldsymbol{p}}^{(j)}$.

Algorithm 5 Particle filter source tracker

```
Input PRPs, $\left\{\tilde{\phi}^k(\mathbf{p}, t)\right\}_{k=1}^{K}$
for $j = 1$ to $J$ do
    for $p = 1$ to $P$ do
        Sample $\hat{\boldsymbol{p}}^{(j,p)}$ from (3.42)
        Evaluate $\tilde{w}_t^{(j,p)}$ from (3.45)
    end
end
Normalize weights, $w_t^{(j,p)}$, from (3.44)
Re-sample $\hat{\boldsymbol{p}}^{(j,p)}$ [214]
Extract point estimate, $\tilde{\boldsymbol{p}}$ (3.46)
Output Cartesian source position, $\tilde{\boldsymbol{p}}$.
```

Using (3.41) and (3.36), the posterior p.d.f. of the states, $\boldsymbol{p}$, can hence be expressed as:

$$f_t(\boldsymbol{p}|\boldsymbol{\phi}_{1:t}) = \sum_{j=1}^{J_{t-1}} \sum_{p=1}^{P} w_t^{(j,p)} \delta_{\hat{\boldsymbol{p}}^{(j,p)}}(\boldsymbol{p}), \tag{3.43}$$

with weights:

$$w_t^{(j,p)} = \frac{\tilde{w}_t^{(j,p)}}{\sum\limits_{j=1}^{J_{t-1}} \sum\limits_{p=1}^{P} \tilde{w}_t^{(j,p)}}, \tag{3.44}$$

where the unnormalized weights, $\tilde{w}_t^{(j,p)}$, are defined as:

$$\tilde{w}_t^{(j,p)} \triangleq w_{t-1}^{(j)}\, f(\boldsymbol{\phi}_t|\hat{\boldsymbol{p}}^{(j,p)}). \tag{3.45}$$

The point estimate of the source position at each t is extracted as the weighted average of the particles:

$$\tilde{\boldsymbol{p}} = \sum_{j=1}^{J} \sum_{p=1}^{P} w_t^{(j,p)}\, \hat{\boldsymbol{p}}^{(j,p)} \Bigg/ \sum_{j=1}^{J} \sum_{p=1}^{P} w_t^{(j,p)}. \tag{3.46}$$

In order to avoid an explosion of the number of particles, systematic re-sampling to $J_{\max}$ particles is applied to the particle cloud, $\left\{\hat{\boldsymbol{p}}^{(j,p)} : j = 1, \dots, J\,;\, p = 1, \dots, P\right\}$ after each recursion [214]. The Bayesian algorithm is summarized in Algorithm 5.

3.6.3 Simulation study and performance measures

3.6.3.1 Simulation setup

To evaluate the performance of the algorithms, audio data was generated using the following simulation. The origin of the microphone array, $\boldsymbol{p}^0(t) = \left[x^0(t), y^0(t)\right]^T$, is generated using the following constant velocity model:

$$\boldsymbol{p}^0(t) = \mathbf{F}(t)\,\boldsymbol{p}^0(t-1) + \boldsymbol{n_p}(t), \tag{3.47}$$

where $\boldsymbol{n_p}(t) \sim \mathcal{N}\left(\mathbf{0}_{2\times 1},\, \boldsymbol{\Sigma}^0(t)\right)$ is the process noise with covariance, $\boldsymbol{\Sigma}^0(t)$.

The matrix $\mathbf{F}(t)$ captures the dynamic model, defined here as a constant velocity model given by:

$$\mathbf{F}(t) = \begin{bmatrix} 1 & 0 & \Delta_T\, v(t)\, \sin\gamma(t) \\ 0 & 1 & \Delta_T\, v(t)\, \cos\gamma(t) \end{bmatrix}, \tag{3.48}$$

where Δ_T is the time delay between $t-1$ and t, $v(t)$ is the velocity of the moving platform, and where the array orientation, $\gamma(t)$, is given by the random walk:

$$\gamma(t) = \gamma(t-1) + v_\gamma(t), \qquad v_\gamma(t) \sim \mathcal{N}\left(0,\, \sigma^2_{v_\gamma}(t)\right). \tag{3.49}$$

The microphone array elements are placed at $\boldsymbol{r}^{(\{1,2\})} = \left[\pm 0.25, 0, 0\right]^T$ relative to the array center, such that the positions of microphone, $m \in \{1, 2\}$, is given by:

$$\boldsymbol{p}^m(t) = \mathbf{R}^{-1}(\gamma_t)\,\boldsymbol{r}^m + \boldsymbol{p}^0(t), \tag{3.50}$$

where $\mathbf{R}(\gamma_t)$ is the rotation matrix, defined as:

$$\mathbf{R}(\gamma_t) \triangleq \begin{bmatrix} \cos(\gamma_t) & -\sin(\gamma_t) \\ \sin(\gamma_t) & \cos(\gamma_t) \end{bmatrix}. \tag{3.51}$$

Using (3.47) and (3.50), the trajectory of the microphone center in (3.47) within a room of size $6 \times 6 \times 2.5$ m was simulated with the initial position at $(2, 2, 1.5)$ m at a speed of 1 m/sec with orientation variance of $\sigma^2_{v_\gamma}(t) = 0.1$ rad^2 and process noise covariance, $\boldsymbol{\Sigma}^c = 10^{-9} \times \mathbf{I}_4$. A single static source was placed at $(4, 4, 1.5)$ m. The scenario is shown in Fig. 3.6.

Using the RIR generator in [210] the RIRs of 100 time steps at time delays of 0.2 sec along the trajectory of the microphone pair were simulated for a reverberation time of 0.3 sec. The resulting RIRs were convolved with a 20 sec speech signal from a female speaker constructed from the TIMIT database. For localization, the height of the microphones and sources is assumed constant and known, such that the $2D$ model can be used.

The input of both algorithms is constructed by the STFT with a rectangular window for each microphone. The results are used to produce the PRPs as described in [16].

3.6.3.2 Results

We present here the results of the two proposed algorithms: IDEM and Particle filter. It should be emphasized that localization with dynamic arrays for acoustic signal processing has not been dealt much before. This is the reason we found it difficult to compare our results to more algorithms.

The IDEM in Alg. 4 is evaluated for $\sigma^2 = 0.1$. The estimated posterior p.d.f. after 4 iterations is plotted in Fig. 3.7.

It can be seen that the position error is zero, when the source is located on the grid. When it is not on the grid the error is dictated mainly by the grid resolution.

This algorithm is very accurate, but it assumes all samples are used together. As an on-line approach, we have decided to use the particle filter.

The particle filter in Alg. 5 is evaluated for $P = 100$ particles for $\mathbf{Q} = 0.01\mathbf{I}$ and with $\sigma^2 = 0.04$ and $J_{\max} = 100$. The filter is initialized by $J_0 = 500$ particles that are uniformly spread within the room region, with at least 1 m distance from each of the four walls. The importance weights are initialized to $w_0^{(i)} = 1/J_0$, $\forall i \in 1, \ldots, J_0$.

The Euclidean distance between the true source position and the point estimates of the source over time are plotted in Fig. 3.8.

The filter achieves its optimal estimation performance of 5.8 cm at $t = 38$ when the sensor pair is steering towards at a source-sensor distance of 1.5 m (see Fig. 3.6). The performance degrades to up a Euclidean distance between $30 - 40$ cm between $t \in [60, 98]$ when the sensor is steering away from the source.

3.7 Conclusions

In this chapter we addressed the challenge of distributed source localization in a reverberant room using two kinds of sensor configurations. We presented algorithms using pairs of static microphones and algorithms that use a single moving pair of microphones.

A new set of hidden variables for EM and REM enabled development of new algorithms that can be implemented without any central point and surprisingly have better convergence properties: they are less dependent on initial conditions and they converge much faster.

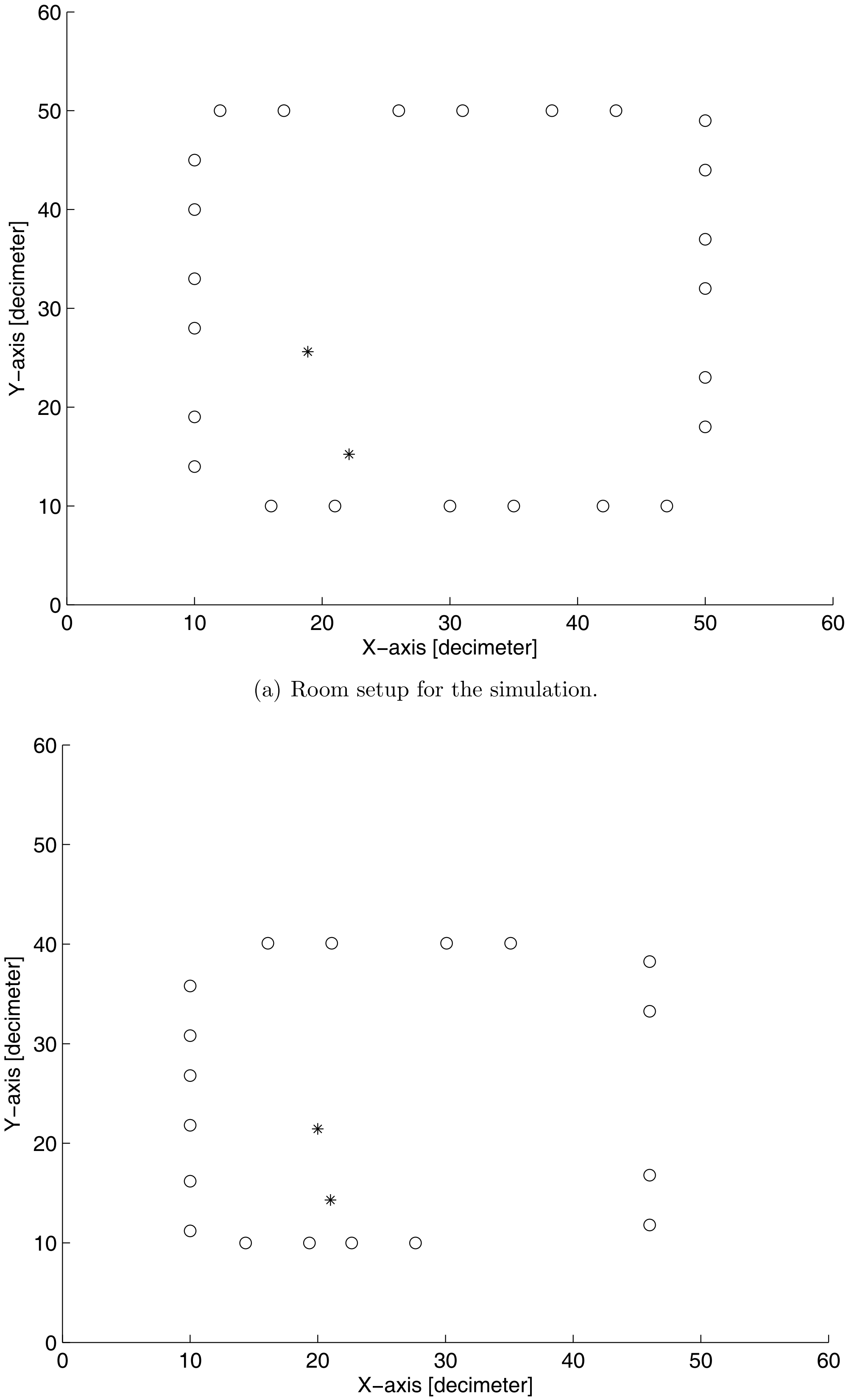
The incremental version of one of those algorithms was also applied for a robot audition (dynamic) case. A moving pair of microphones was used for localization utilizing the movement for spatial diversity.

We adapt the SAR concept from the RF field. Reverberant audio data was simulated for a microphone array with two sensors and the complex-valued PRPs were extracted as measurements. Two approaches for sound source localization using the PRPs were proposed.

The first approach is a ML implemented by an EM algorithm, which processes all data as a batch. Localization accuracy is dictated by the grid resolution. The static nature of the source and the dynamics of the microphones enable accurate results.

In order to facilitate sequential sound source localization from on-line measurements, a particle filter was also proposed. Particle filters are typically aimed at source tracking in highly dynamic scenarios. In this section, the approach was chosen in order to ensure real-valued source position estimates from the complex PRP measurements. Due to the sequential nature of the algorithm, this performance was shown to be dependent on the path of the robot.

In this approach we have decided to compare two modified algorithms from different paradigms, as a first step of addressing the challenge of source localization with a single pair of microphones. Possible extensions include using REM for on-line processing, solving uncertainty in robot trajectory and tracking multiple moving sources.



(a) Room setup for the simulation.

(b) Room setup for the recording analysis.

Figure 3.2: Room setups for the experimental section. Microphone pairs are marked by pairs of circles and sources are marked by $*$.

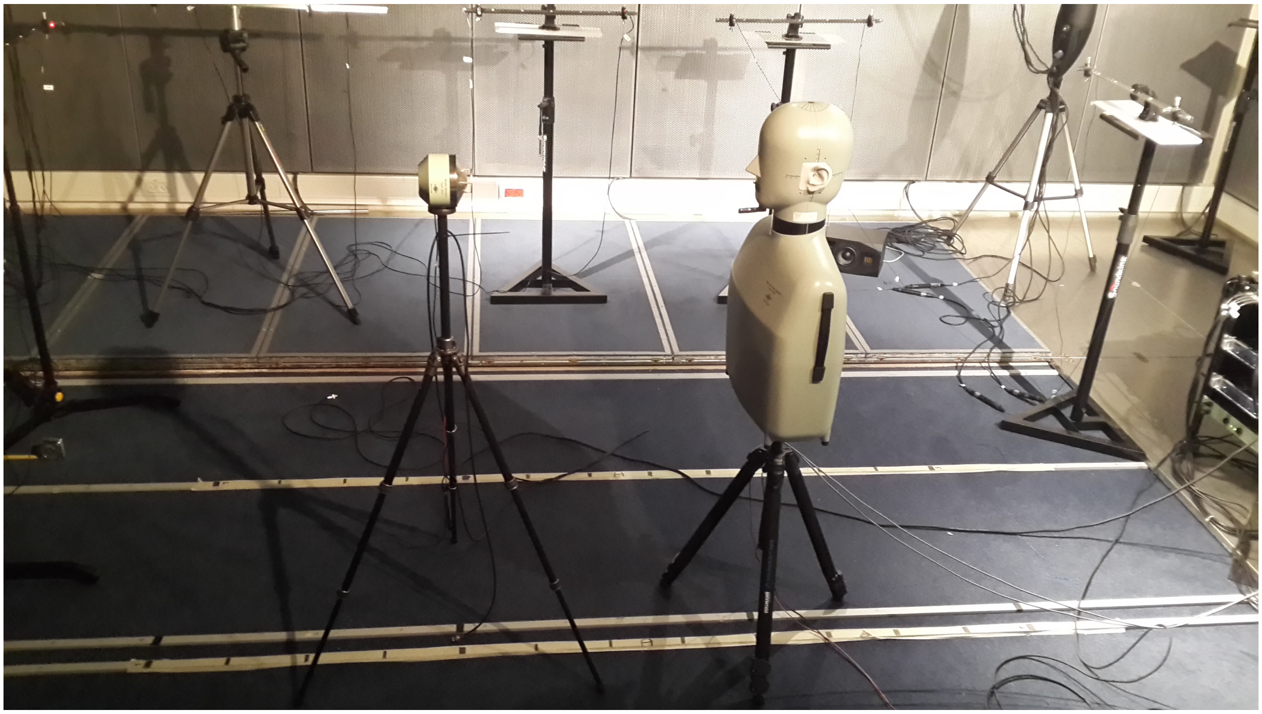
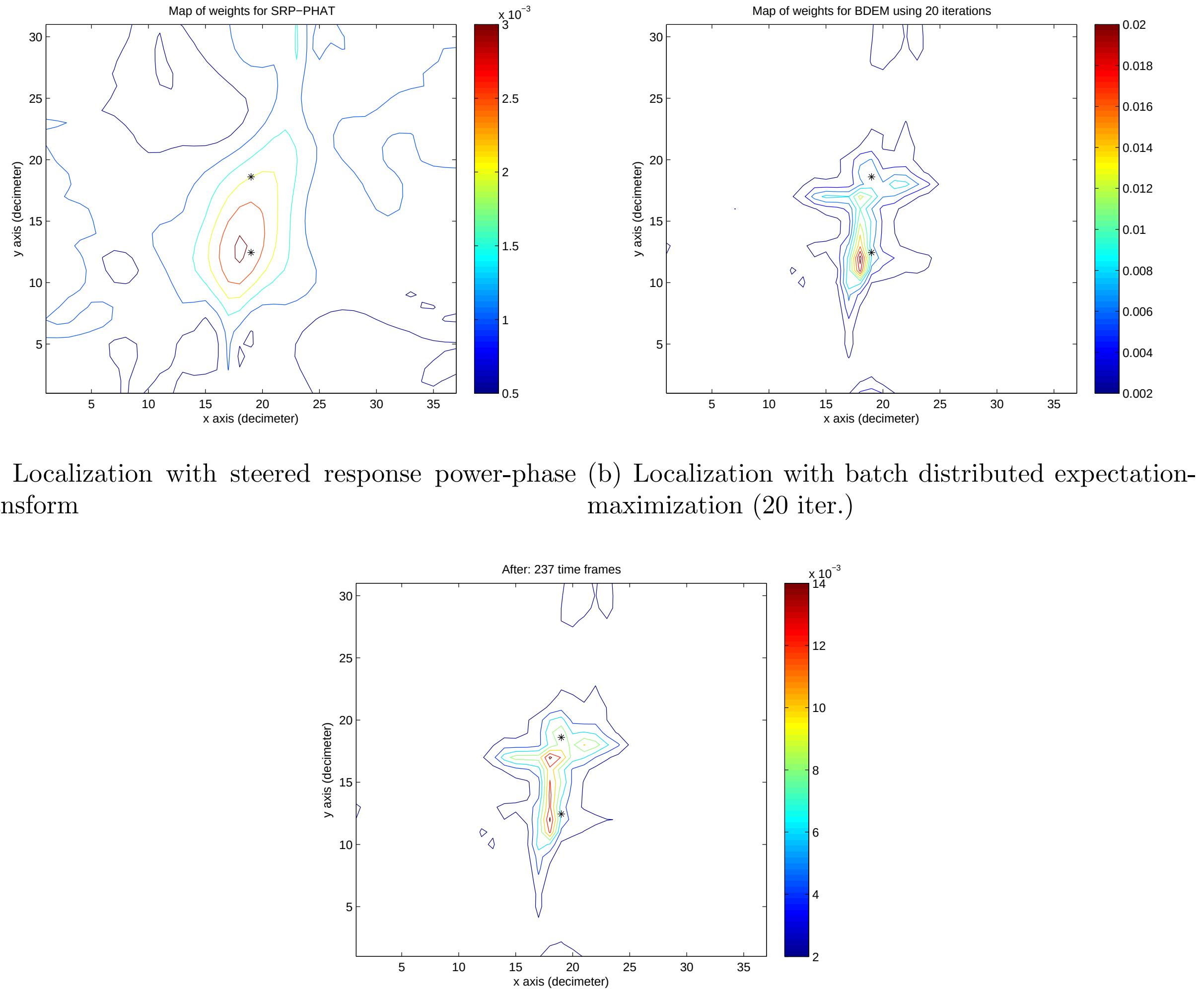


Figure 3.3: Experimental setup in the speech and acoustic lab of the engineering faculty at Bar-Ilan University. Two sources 61.5 cm from each other and $T_{60} = 150$ msec.

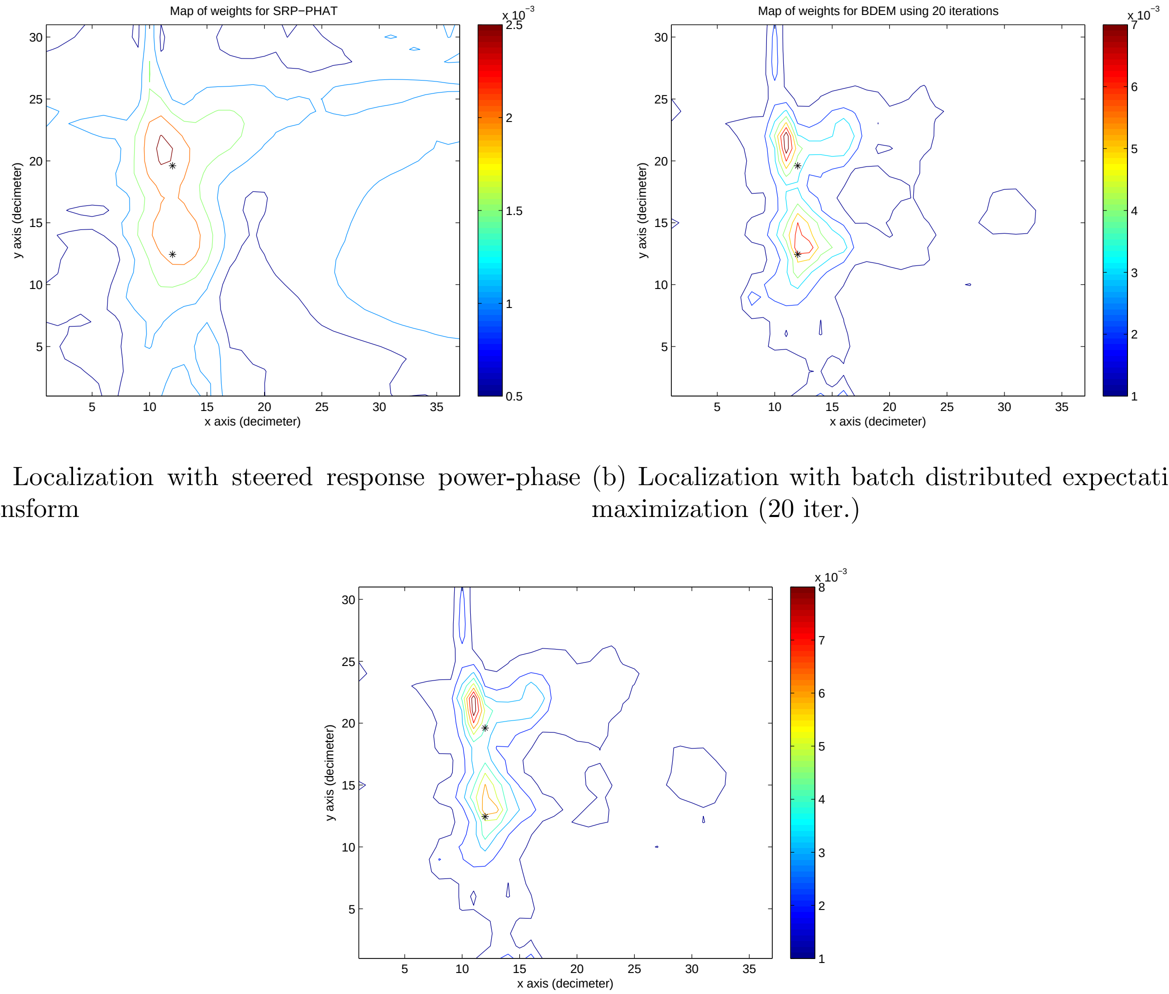


(a) Localization with steered response power-phase transform

(b) Localization with batch distributed expectation-maximization (20 iter.)

(c) Localization with recursive distributed expectation-maximization

Figure 3.4: Map of one experimental trial for $T_{60} = 150$ msec and a source inter-distance of 61.5 cm. The poor resolution of steered response power-phase transform is depicted in (a). The batch distributed expectation-maximization detected both sources, as depicted in (b). The result achieved by recursive distributed expectation-maximization, the clear detection of both sources, is shown in (c).



(a) Localization with steered response power-phase transform

(b) Localization with batch distributed expectation-maximization (20 iter.)

(c) Localization with recursive distributed expectation-maximization

Figure 3.5: Map of experimental trial with $T_{60} = 450$ msec and a source inter-distance of 71.5 cm. The steered response power-phase transform algorithm only detected one of the sources, as depicted in (a). The batch distributed expectation-maximization and recursive distributed expectation-maximization were capable of detecting both sources as shown in (b)-(c).

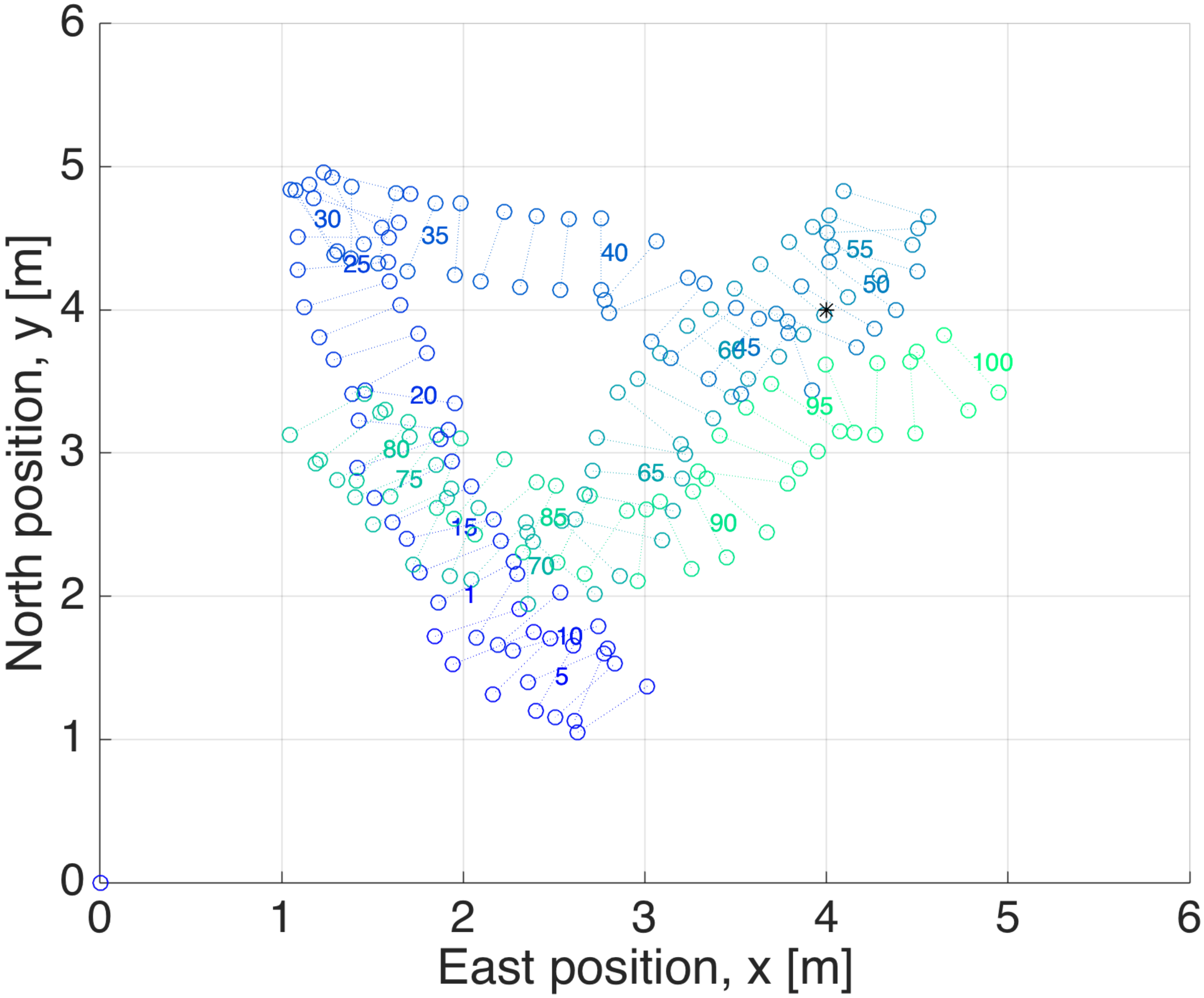


Figure 3.6: Scenario of $6 \times 6 \times 2.5$ m room, with source (black asterisk) at $(4, 4, 1.5)$ m and moving sensor with initial position at $(2, 2, 1.5)$ m. The color code represents a continuum of colors from blue at $t = 1$ to green at $t = 100$.

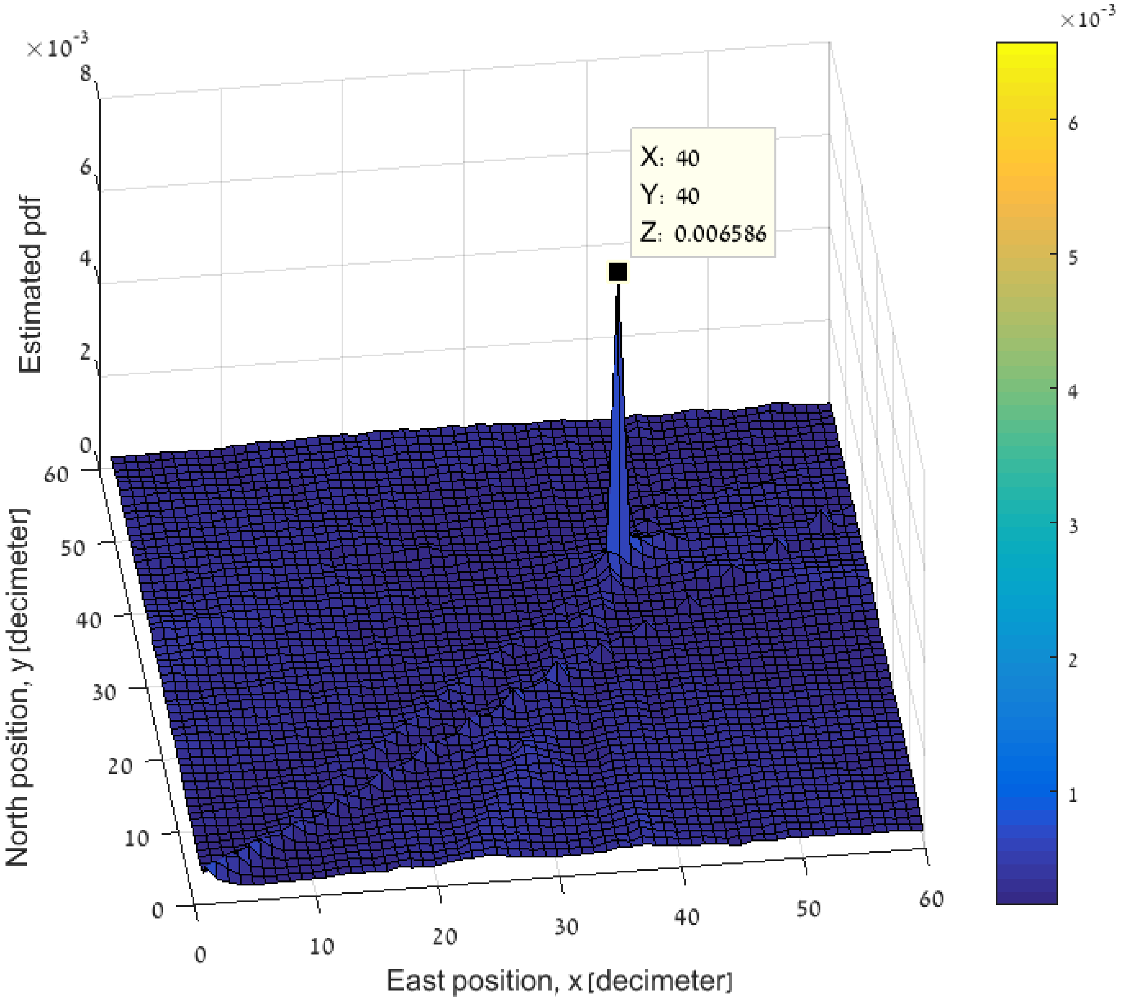


Figure 3.7: The estimated posterior p.d.f., where a black circle marks the true source position.

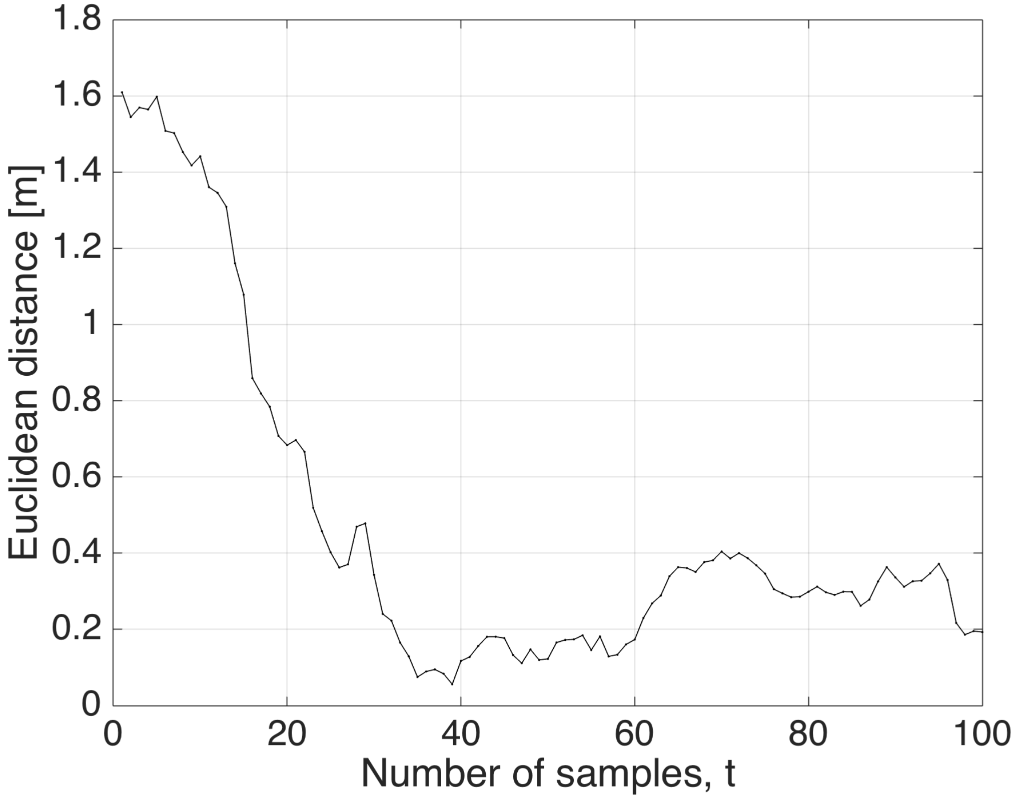


Figure 3.8: Euclidean distance between the state estimates, $\tilde{\boldsymbol{p}}$, and true source positions, $\boldsymbol{p}$ for the particle filter.

Chapter 4

Localization for high reverberation levels

The material presented in this chapter is based on [195] and [196]:

[195] Y. Dorfan, A. Plinge, Hazan G., and S. Gannot, “Distributed expectation-maximization algorithm for speaker localization in reverberant environments,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 3, pp. 682–695, 2018

[196] O. Schwartz, Y. Dorfan, E.A.P. Habets, and S. Gannot, “Multiple DOA estimation in reverberant conditions using EM,” in *International Workshop for Acoustic Echo Cancellation and Noise Control (IWAENC), Xi’an, China*, 2016

As already discussed above, localization often tightly rely on the direct path. When dealing with indoor scenarios, high reverberation levels is a key challenge. The nature of the reflection is non-stationary and it is increased with the number of acoustic sources in the room. In this chapter we present a few ways for reducing the influence of the reverberation and enhance the direct path to enable $1D$, $2D$ or $3D$ localization of multiple concurrent speakers.

4.1 Distributed localization: high reverberation levels

A major obstacle to high localization accuracy is the presence of reverberation, the influence of which obviously increases with the number of active speakers in the room. The problem formulation was defined above in section 2.2.

The challenge of distributed localization in reverberant environment for concurrent speakers is addressed. We adopt the IDEM methodology presented in [192], which was later adapted for BSS [197]. In the proposed method, we combine the power of machine hearing processing and that of the distributed MoG estimation. A bio-inspired pre-filtering is presented to eliminate a large percentage of the reflections. Consequently, the effect of reverberation on the localization accuracy is significantly reduced.

An additional modification applied to the original IDEM algorithm [192] is the ordering of the nodes. The ordering refers to the sequence of processing along the nodes of the network. Instead of a constant order, as used by the regular directed ring, we adopt here the concept of pseudo-random order, which is known to improve the convergence of the EM [101] and the robustness of the network to failures. In order to enable the new ordering we should replace the directed-ring with a topology that contains a much reacher connectivity. For simplicity we use the fully connected topology.

The contribution of this algorithm is a combination of the bio-inspired pre-filtering, filter-bank approach, truncated model presented below and the IDEM algorithm. It is shown that it produces good results for multiple speakers in reverberant rooms. The idea is that we are able to filter out few clean signals that represent the direct path. Unlike other applications, localization can be achieved even with a very small number of reliable measurements.

The resulting algorithm is also attractive from the network point of view, since, similarly to the method proposed in [192], it can be implemented without any central node. In addition, exact, sample-level, synchronization of the nodes is not required. In contrast to the model presented in [192], by virtue of its randomized order, it is capable of overcoming link failures.

4.1.1 The proposed method

The proposed distributed method comprises three major steps. In the first step, a bio-inspired model is applied to the microphone signals for reducing the effect of reverberation. In the second step, cross-correlation is applied to estimate the TDOAs. The final step, which is based on the IDEM, is derived from the ML model in order to find the number of speakers and their locations. These steps and the statistical model of the measurements are described in this subsection.

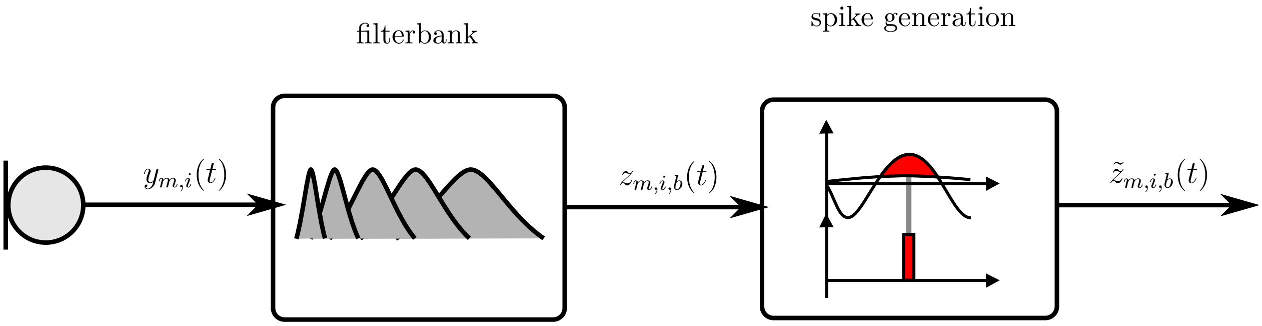


Figure 4.1: Signal preconditioning stage. The microphones' signals are each split into multiple bands by a gammatone filterbank and then transformed into spike trains.

4.1.1.1 Cochlear model

The speech peak detection method applied as a preprocessing procedure for each microphone independently is described in this subsubsection. The efficient on-line cochlear model introduced in [57] is used to generate a sparse spike representation of the microphone signals. It consists of a filter bank modeled on the basilar membrane and a model of the cochlear nucleus generating spikes from the band filtered signals. The preconditioning is illustrated in Fig. 4.1. It is tailored to facilitate localization in reverberant environment from three aspects: first, only highly modulated parts of the signal are used; second, phase-locked spikes are generated; and third, echo suppression is achieved by modeling the neural saturation (cf. [95]).

A common approach in computational auditory scene analysis (CASA) is to use an infinite impulse response (IIR) filter bank to model the basilar membrane [90]. In our approach, a fast Fourier transform (FFT) filter bank is used. The filters are defined in the spectral domain using a gammatone [43] approximation [215]:

$$\hat{G}_b(f) = \left(1 + \frac{\imath(f - f_b)}{w \cdot w_b}\right)^{-4}, \tag{4.1}$$

where $\imath$ is the imaginary unit and $b = 1, \dots, B$ denotes the frequency band index. The total number of bands is $B = 16$. The bands are defined to imitate the critical bands found in human hearing [90, 91].

The nonlinear spacing of the bands with center frequencies ($f_b = f_1, \dots, f_B$) between 0.3 kHz and 3 kHz is chosen as equidistant on the quasi-logarithmic equivalent rectangular bandwidth (ERB) scale [216, 91]. The Glasberg-Moore bandwidth [216], w_b, is widened by a factor w in order to increase frequency content.

For each microphone signal $y_{m,i}(t)$, B signals $z_{m,i,b}(t)$ are obtained at the output of the filterbank. In each band, the signal is transformed into a sparse spike representation. The process of computing the spikes is illustrated in Fig. 4.2. One of the key concepts is to model the neural saturation in order to detect the modulation and imitate monaural echo suppression. This is achieved by a halfway rectification and a peak over average comparison. Thus, for each band signal $z_{m,i,b}(t)$ is halfway rectified $\max(0, z_{m,i,b}(t))$ before a moving average $\overline{z}_{m,i,b}(t)$ is computed. The averages are calculated over an intermediate interval of, e.g. 30 msec to encompass the pitch modulation. This delay is a non-issue for this off-line static case. If applied to dynamic on-line applications, the latency might be still reasonable, since indoor movements of speakers is not expected to be too fast.

Modulated intervals $U_n = [t_u, t_d]$ are found as time segments, where the signal exceeds the shifted version of the calculated average $\overline{z}_{m,i,b}(t - t_{\text{shift}})$, where t_{shift} is a time shift used to enhance the first wavefronts as illustrated in Fig. 4.2. Its maximum position in time, p_n is a candidate position for a spike. If the area above the average is 6 dB higher than the signal, an impulse spike is generated at p_n in the output signal $\tilde{z}_{m,i,b}$.

The pulse height h_n of the spike corresponds to the amplitude computed by square root compression:

$$h_n = \sum_{t=t_u}^{t_d} \sqrt{z_{m,i,b}(t) - \overline{z}_{m,i,b}(t - t_{\text{shift}})}. \tag{4.2}$$

The pulse width is set to 20μsec to imitate the temporal resolution of human neural processing.

This nonlinear processing is summarized in Algorithm 6. The resulting signal of that processing is sparse. Although only few bands are used, it still complies with the model of sparse representation of the signal, which will be required for the EM algorithm.

The signals $\tilde{z}_{m,i,b}(t)$ of the microphones of all nodes are used for calculating the features, the multiple pairwise TDOAs (MPTs). The MPTs, denoted by $\tau_{m,b}(\mathbf{p})$, are used as feature vectors in the ML. The model of these features is discussed in the next subsubsection.

4.1.1.2 Multiple pairwise time difference of arrival

In order to calculate the MPT, $\tau_{m,b}(\mathbf{p})$ we apply the following steps at each node to $\tilde{z}_{m,i,b}(t)$: 1) Calculate the cross-correlation of the two microphones. 2) Find the peak and compare it to a threshold. 3) If it passes the threshold, calculate $\tau_{m,b}(\mathbf{p})$, the exact (interpolated) time

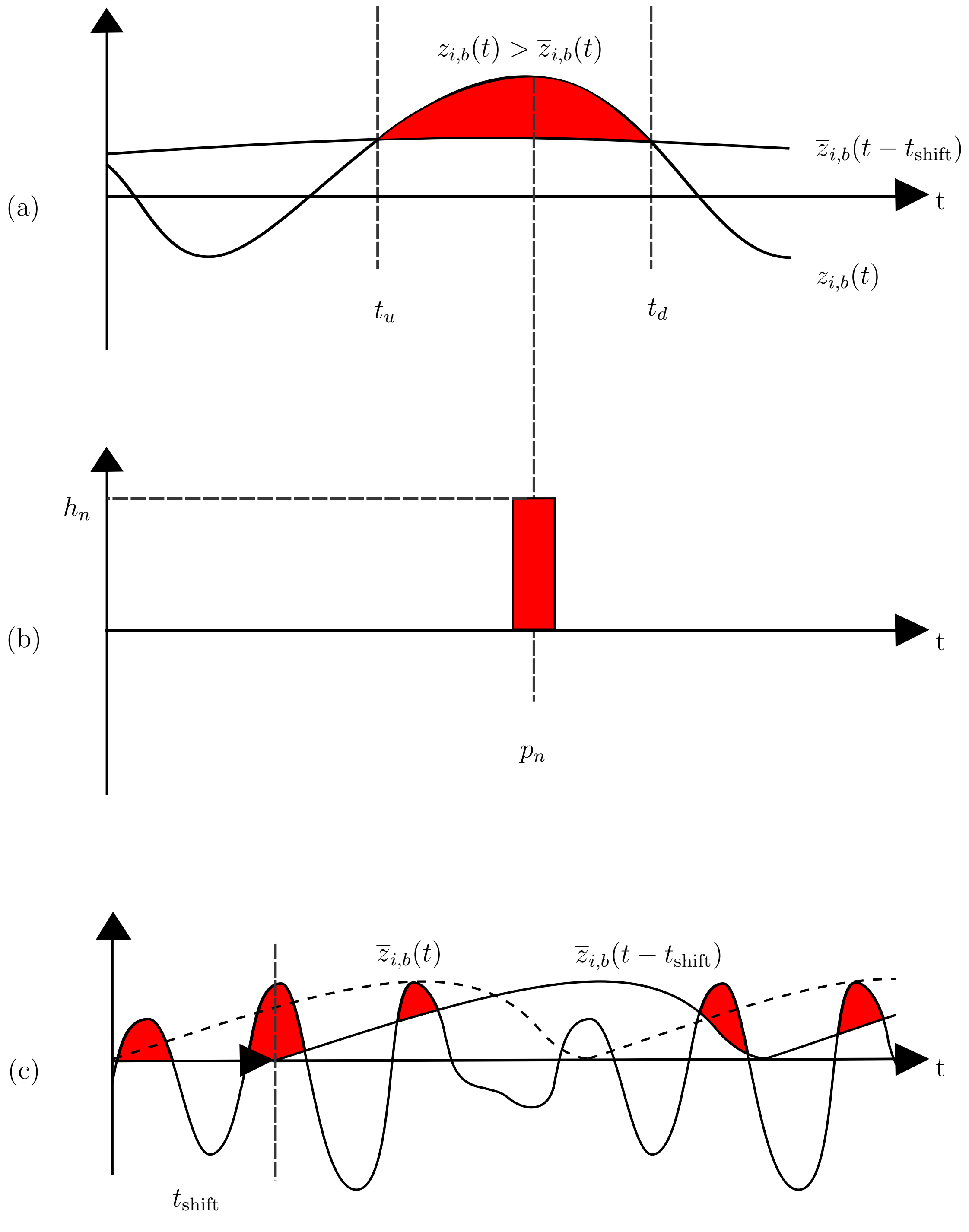

(a)
$z_{i,b}(t) > \overline{z}_{i,b}(t)$
$\overline{z}_{i,b}(t - t_{\text{shift}})$
t
$z_{i,b}(t)$
t_u
t_d
(b)
h_n
t
p_n
(c)
$\overline{z}_{i,b}(t)$
$\overline{z}_{i,b}(t - t_{\text{shift}})$
t
t_{shift}

Figure 4.2: Spike generation procedure. (a) The band signal is compared to its short time average. (b) When the area above the average exceeds the signal by 6 dB, a spike of height h_n is generated at the maximum position p_n. (c) By shifting the average in time, the first wavefronts are enhanced.

Algorithm 6 Cochlear processing model

```
initialize $\tilde{z}_{m,i,b}(t) = 0$
calculate moving average $\overline{z}_{m,i,b}(t)$
find modulated intervals $U_n = [t_u, t_d]$
where $z_{m,i,b}(t) > \overline{z}_{m,i,b}(t - t_{\text{shift}}) \quad \forall t_u < t < t_d$
foreach $U_n$ do
    $p_n = \arg\max_{t_u<t<t_d} z_{m,i,b}(t) - \overline{z}_{m,i,b}(t - t_{\text{shift}})$
    if $z_{m,i,b}(p_n) > 2\overline{z}_{m,i,b}(p_n - t_{\text{shift}})$ (6 dB) then
        Compute $h_n = \sum_{t=t_u}^{t_d} \sqrt{z_{m,i,b}(t) - \overline{z}_{m,i,b}(t - t_{\text{shift}})}$
        Insert $h_n$ with width 20 $\mu$sec at $p_n$ in $\tilde{z}_{m,i,b}$
    end
end
```

difference of the peak.

In contrast to the PRP used in [192], the MPTs are real-valued quantities. The speakers are assumed to exhibit disjoint activity in the time-frequency domain [110, 16, 17, 18]. Therefore, every time-frequency band can be associated with at most a single active position (or speaker). Here we claim that the sparsity assumption is still valid although our frequency resolution is poorer.

It is assumed that the speakers can be located in a final set of positions on a grid with a desired resolution. The time differences can be calculated in advance for every candidate location $\mathbf{p}$ on that grid:

$$\tilde{\tau}_{m,b}(\mathbf{p}) \triangleq \frac{||\mathbf{p} - \mathbf{p}_m^1|| - ||\mathbf{p} - \mathbf{p}_m^2||}{c}, \tag{4.3}$$

where $\mathbf{p}_m^1$ and $\mathbf{p}_m^2$ are the locations of the microphones, assumed to be known, $||\cdot||$ denotes the Euclidean norm, and c is the sound velocity. The set $\mathbf{p} \in \mathcal{P}$ contains the grid of points.

In the following subsubsection, we describe the statistical model of the MPTs. Since they have different physical properties than the PRPs, a new statistical model is presented.

4.1.1.3 Statistical model

We assume that the MPTs are random observations drawn from a mixture of truncated normal distributions:

$$\tau_{m,b}(t) \sim \sum_{\mathbf{p}} \psi_{\mathbf{p}} g\left(\tau_{m,b}(t); \tilde{\tau}_{m,b}(\mathbf{p}), \sigma^2, F, \tau_{\max}\right), \tag{4.4}$$

where $\psi_{\mathbf{p}}$ is the probability of a speaker being at position $\mathbf{p}$.

The p.d.f., $g(\cdot;\cdot,\cdot,\cdot,\cdot)$ is the truncated Gaussian probability [205]:

$$g\left(x;\mu,\sigma^2,F,\tau_{\max}\right) = \begin{cases} \frac{\mathcal{N}(x;\mu,\sigma^2)}{\int\limits_{x_L}^{x_H}\mathcal{N}(x;\mu,\sigma^2)}, & x \in [x_L, x_H] \\ 0, & \text{otherwise} \end{cases} \tag{4.5}$$

$$x_L = \max(\mu - F\sigma, -\tau_{\max}); x_H = \min(\mu + F\sigma, \tau_{\max})$$

where

$$\mathcal{N}\left(x;\mu,\sigma^2\right) = \frac{1}{\sqrt{2\pi\cdot\sigma^2}}\exp\left(-\frac{(x-\mu)^2}{2\cdot\sigma^2}\right) \tag{4.6}$$

is the normal distribution.

The maximal time difference, $\tau_{\max}$ is used to limit the truncated supports. It is the maximal time difference that can be produced by sound traveling between the microphones at each node. We denote τ_{max} the maximal physical delay.

The factor of the truncation F is empirically determined in the experimental subsection. The variance σ^2 is used to control errors in estimating the MPTs due to noise and reverberation. It also supports off-grid source positions.

The truncation is applied according to the lower and higher support limits, x_L and x_H, respectively. The truncated Gaussian distribution is used here rather than the Gaussian used in [39], since it allows irrelevant measurements to get zero probability, hence better fitting the physical model. The truncation for grid positions near the end-fire is asymmetric due to the maximal physical delay. Please notice two different restrictions. One of them is derived from the distance between the microphones. The second one is applied locally around the grid point. The second one means that the distribution is finite and the reason is that we filter out noisy measures.

We state now the ML estimation procedure of the localization parameters from a given set of local measurements. As in [192], the parameters to be estimated are the weights, $\psi_{\mathbf{p}}$, which are the probabilities of finding acoustic sources in each position, $\mathbf{p}$.

The other parameters are assumed to be known. The variance σ^2 was manually tuned to 6 [samples^2]. The variance could be assumed unknown, as often done, and its estimation can be easily incorporated in the EM iteration [47]. It is kept fixed, since we have not observed any significant advantage of adding it to the parameter estimation in this particular algorithm.

Further variance analysis is presented in the experimental subsection.

The vector of the unknown parameters is defined as

$$\boldsymbol{\theta} \triangleq \mathrm{vec}_{\mathbf{p}}(\psi_{\mathbf{p}}). \tag{4.7}$$

Note that $\psi_{\mathbf{p}}$ has a probability interpretation, namely $\sum_{\mathbf{p}} \psi_{\mathbf{p}} = 1$. The goal of the algorithm is to estimate these parameters. Then we will apply a threshold in order to estimate the number of active speakers (also referred to as speaker detection) and their locations. These parameters are global, meaning common to all nodes.

Augmenting all MPT readings in $\boldsymbol{\tau} = \mathrm{vec}_{m,t,b}\left(\tau_{m,b}(t)\right)$ and following [193], the p.d.f. of the observation set can be written as

$$f(\mathbf{T} = \boldsymbol{\tau}; \boldsymbol{\theta}) = \prod_{m,t,b} \sum_{\mathbf{p}} \psi_{\mathbf{p}} g\left(\tau_{m,b}(t); \tilde{\tau}_{m,b}(\mathbf{p}), \sigma^2, F, \tau_{\max}\right). \tag{4.8}$$

The MLE problem can be stated as

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\mathrm{argmax}}[\log f(\mathbf{T} = \boldsymbol{\tau}; \boldsymbol{\theta})]. \tag{4.9}$$

A straightforward solution of the ML is centralized and of high complexity. This type of problems is often solved by the EM in order to reduce its complexity.

As already mentioned above, we are interested in a distributed solution of this estimation problem. The global parameters will be estimated jointly by the nodes of the network. Several algorithms can be suggested for this problem. In this contribution, we are interested mainly in the application of the IDEM algorithm [192] to the MPTs.

4.1.1.4 An incremental distributed expectation-maximization with pseudo-random topology

This subsubsection is divided into two parts. The first deals with the choice of the hidden variables of the DEM algorithms that can be used for the problem at hand. Here, we focus on one of these algorithms, the IDEM [192], which is presented in the second part. The IDEM processes the measurements and the hidden variables incrementally through the network.

It is shown that for updating the minimal sufficient statistics (MSS), only the recent parameter estimation is needed. This estimation is very compact and hence requires less communication BW.

Local hidden variables

The global parameters can be estimated by applying EM techniques using global hidden variables [16] or local hidden variables [192].

Synchronization for acoustic network has been dealt in [217, 218]. Rough synchronization of nodes is maintained in our case by the communication link. We also assume that the signals acquired by the microphones of the same node are fully synchronized, since the accuracy of the MPTs depends on it. A much looser assumption is made regarding the signals of different nodes [204].

For DEM algorithms, it was suggested that local hidden variables be defined [192]. The hidden variables are defined to be the indicators

$$y_m(t,b,\mathbf{p}) = \begin{cases} 1, & \mathbf{p} \text{ active for } (m,t,b) \\ 0, & \text{otherwise} \end{cases}. \tag{4.10}$$

For each time-frequency band (t,b) they equal to zero everywhere except to $\mathbf{p}$, the position of the active speaker, since not more than one speaker is likely to be active at each time-frequency band [110].

The total number of indicators in the problem is $T \times B \times M$. Each of these is defined over $|\mathcal{P}|$ possible values, with $|\mathcal{P}|$ the cardinality of the set of all possible grid positions. Please note that, in contrast to methods where global hidden variables are used [16], local hidden data support a case where some of the nodes measurements are physically unfeasible and hence assigned with zero probability.

Let $\mathbf{y} = \text{vec}_{t,b,m,\mathbf{p}}\left(y_m(t,b,\mathbf{p})\right)$ be the set of all local indicators. The expectation of the indicator is given by

$$E\left\{y_m(t,b,\mathbf{p})\right\} = \psi_{\mathbf{p}}. \tag{4.11}$$

Under the W-disjoint assumption [110], namely that each observation can be associated with only a single speaker (and hence a single position) and under the static model assumption, the probability density function of $\mathbf{y}$ is given by

$$f(\mathbf{Y}=\mathbf{y};\boldsymbol{\theta})=\prod_{m,t,b}\sum_{\mathbf{p}}\psi_{\mathbf{p}}y_m(t,b,\mathbf{p}). \tag{4.12}$$

Given the local hidden variables, the p.d.f. of the observations is

$$f(\mathbf{T}=\boldsymbol{\tau}|\mathbf{y};\boldsymbol{\theta})=\prod_{m,t,b}\sum_{\mathbf{p}}y_m\left(t,b,\mathbf{p}\right)g\left(\tau_{m,b}(t);\tilde{\tau}_{m,b}(\mathbf{p}),\sigma^2,F,\tau_{\max}\right). \tag{4.13}$$

The p.d.f. of the complete data can be deduced from (4.12)-(4.13) and from some simplifications due to the properties of the indicator $y_m\left(t,b,\mathbf{p}\right)$:

$$\begin{aligned}&f(\mathbf{T}=\boldsymbol{\tau},\mathbf{Y}=\mathbf{y};\boldsymbol{\theta})=f(\mathbf{Y}=\mathbf{y};\boldsymbol{\theta})f(\mathbf{T}=\boldsymbol{\tau}|\mathbf{y};\boldsymbol{\theta})=\\&\prod_{m,t,b}\sum_{\mathbf{p}}\psi_{\mathbf{p}}y_m\left(t,b,\mathbf{p}\right)\cdot g\left(\tau_{m,b}(t);\tilde{\tau}_{m,b}(\mathbf{p}),\sigma^2,F,\tau_{\max}\right).\end{aligned} \tag{4.14}$$

A family of distributed algorithms can be derived using these local hidden variables. The algorithm simulated in [192, 197] was implemented over a simple fixed directed ring. In [193], we suggested a BDEM over a bi-directional tree. In the subsequent part of this subsection, we derive a version of the IDEM for the case of MPTs measurements, which is implemented over a pseudo randomly-ordered topology.

Random order IDEM

The gist of this algorithm is to run the EM with local computations in a pseudo-random order of executing the local *E-step* and updating the localization parameters. Each such local *E-step* is referred to as a sub-iteration.

In addition, the *M-step* is applied after each such sub-iteration locally according to the incremental equation, which updates only the local contribution to the parameters estimation. The incremental principle was first introduced in [51].

The idea is that after the initialization round, the algorithm has two types of iterations. The global iterations are defined as those introduced for every batch EM algorithm. However, for the IDEM we also define a sub-iteration index, which is

$$i=M*(\ell-1)+n(m,\ell). \tag{4.15}$$

The order index function, $n(m,\ell)$ assigns a unique integer (between 1 and M) to every node m, which determines the order in which the M nodes are updated in the global iteration ℓ.

This sub-iteration is local.

As in the study reported in [192], each node in its turn obtains the recent parameters' estimation from the previous node and calculates its local (partial) *E-step* in order to update its local hidden variables. Then, it can apply a new *M-step* and transmit the result to the next node.

In contrast to the method proposed in [192], which uses directed-ring topology, the current scheme determines a random update order in each iteration. Reliable updates in each iteration can be guaranteed if an identical pseudo-random generator is applied in all nodes.

This random order has two major advantages. The first arises from the optimization aspect. The IEM is known to produce faster convergence and more accurate estimation in the case of a pseudo random order of measurements processing. The second advantage arises from the network considerations. For example, when the update order is pseudo random, easy ways to circumvent a link failure exist. The idea is that if a single (but permanent) link failure occurs, a directed-ring topology renders useless. Changing the order after each iteration enables recovery from such a link failure. It should be noted that such a strategy of changing the order depends on the network topology, which in our case assumed to be fully-connected.

As previously mentioned, each node is responsible for its local measurements and local hidden variables, but in contrast to the BDEM method that executes the *E-step* simultaneously at all nodes for the same parameter estimation, our proposed method executes the *E-step* and the *M-step* incrementally.

The *E-step* is carried out in a distributed manner for each node on its turn (sub-index i):

$$\begin{aligned}\upsilon_m^{(i)}\left(t,b,\mathbf{p}\right) &\triangleq E\left\{y_m\left(t,b,\mathbf{p}\right)|\tau_{m,b}(t);\hat{\boldsymbol{\theta}}^{(i-1)}\right\} \\ &= \frac{\hat{\psi}_{\mathbf{p}}^{(i-1)} g\left(\tau_{m,b}(t);\tilde{\tau}_{m,b}(\mathbf{p}),\sigma^2,F,\tau_{\max}\right)}{\sum_{\tilde{\mathbf{p}}}\hat{\psi}_{\tilde{\mathbf{p}}}^{(i-1)} g\left(\tau_{m,b}(t);\tilde{\tau}_{m,b}(\tilde{\mathbf{p}}),\sigma^2,F,\tau_{\max}\right)}.\end{aligned} \tag{4.16}$$

The algorithm is based on the IEM [51]. The authors proved that, not only the classical (batch) EM converges, but also other modifications of the EM. For example partial *E-step* can be followed by the regular *M-step*. In many cases the partial version is even better in terms of convergence speed and accuracy. The IEM algorithm can be applied in the distributed case to local hidden data.

The network topology can be used for incremental updates of the hidden variables and the parameter estimation by transmitting only the recent parameters' values.

Define the local summation of the hidden variables as

$$\bar{\upsilon}_m^{(i)}(\mathbf{p}) \triangleq \sum_{t,b} \upsilon_m^{(i)}(t,b,\mathbf{p}). \tag{4.17}$$

The *M-step* of the BDEM was shown to be the average of all the hidden variables [193]. The *M-step* of the IDEM in this case, as for many other IEM algorithms, simplifies to an update of the local contribution of node m to the global parameters:

$$\hat{\psi}_{\mathbf{p}}^{(i)} = \hat{\psi}_{\mathbf{p}}^{(i-1)} + \frac{\bar{\upsilon}_m^{(i)}(\mathbf{p}) - \bar{\upsilon}_m^{(M*(\ell-2)+n(m,\ell-1))}(\mathbf{p})}{T \cdot B \cdot M}. \tag{4.18}$$

The new estimation, $\bar{\upsilon}_m^{(i)}(\mathbf{p})$ is added while the previous local contribution, $\bar{\upsilon}_m^{(M*(\ell-2)+n(m,\ell-1))}(\mathbf{p})$ is subtracted. Each node of the network uses its local memory to store this local contribution in order to use it in the next iteration. The denominator contains a normalization factor.

The initialization of the global parameters is obtained by a preliminary global iteration:

$$\hat{\psi}_{\mathbf{p}}^{(0)} = \frac{\sum_m \bar{\upsilon}_m^{(0)}(\mathbf{p})}{T \cdot B \cdot M}, \tag{4.19}$$

where the initial hidden data estimates, $\bar{\upsilon}_m^{(0)}(\mathbf{p})$ are calculated at each node and aggregated through the network.

After the last iteration a threshold is applied to the latest estimation of the parameters $\psi_{\mathbf{p}}^{(LM)}$. Each position above the threshold is declared as a speaker location. This way we estimate the number of speakers and their locations.

This concludes the description of the proposed algorithm. Algorithm 7 summarizes the proposed IDEM approach.

4.1.2 Experimental study

This subsection is dedicated to an experimental study of the localization algorithms. We use the proposed algorithm in various versions and two other algorithms for comparison. The first, denoted IDEM2014 is described in [192]. The second, is the well-known SRP-PHAT algorithm [87]. This is an approach that searches the position that maximizes the output of a steered delay and sum BF. As already mentioned above, localization in acoustic signal

Algorithm 7 IDEM Speaker localization.

```
Obtain the valid local measurements $\tau_{m,b}(t)$.
initialize $\bar{v}_m^{(0)}(\mathbf{p})$ and $\psi^{(0)}(\mathbf{p})$.
for $\ell = 1$ to $L$ do
    Randomly choose the order $n(m, \ell)$.
    for $n(m, \ell) = 1$ to $M$ do
        Calculate the current sub-iteration index: $i = M * (\ell - 1) + n(m, \ell)$.
        Calculate the current node index: $m = n^{-1}(m, \ell)$.
        E-step
        Calculate locally $v_m^{(i)}(t, b, \mathbf{p})$ (4.16).
        Aggregate locally $\bar{v}_m^{(i)}$ (4.17).
        M-step
        Calculate the global parameters, $\hat{\psi}_{\mathbf{p}}^{(i)}$ (4.18).
        Transmit the global parameters to the next node.
    end
end
Find $\hat{J}$, the number of speakers, and their respective locations $\mathbf{p}_j$; $j = 1, \ldots, \hat{J}$ by applying
a threshold to $\psi_{\mathbf{p}}^{(LM)}$, which is the result of the last M-step.
```

processing is usually developed for $1 - D$ only. This is the reason we found it difficult to compare our results to more state-of-the-art algorithms.

The subsection is organized as follows. The first subsubsection presents the setup of both the simulated room and the acoustic laboratory. The parameter settings of the algorithms and complexity analysis are described in the second subsubsection. The third subsubsection presents the results of the simulation analysis. The forth subsubsection examines different components of the proposed algorithm. The last subsubsection discusses the real recording results.

4.1.2.1 Simulation and acoustic room setup

This subsubsection contains two parts: simulated room and real room descriptions.

Simulated room

We simulated 100 random positions of 1, 2, or 3 concurrent speakers in a reverberant room ($6 \times 6 \times 2.4$ m) with T_{60} in the range of 150 msec to 600 msec, using the image method [211]. The time duration of each localization experiment was 12sec. Note that the grid resolution, which was 10×10 cm, is sufficiently good taking into account the volume of a real speaker, which is not a point source.

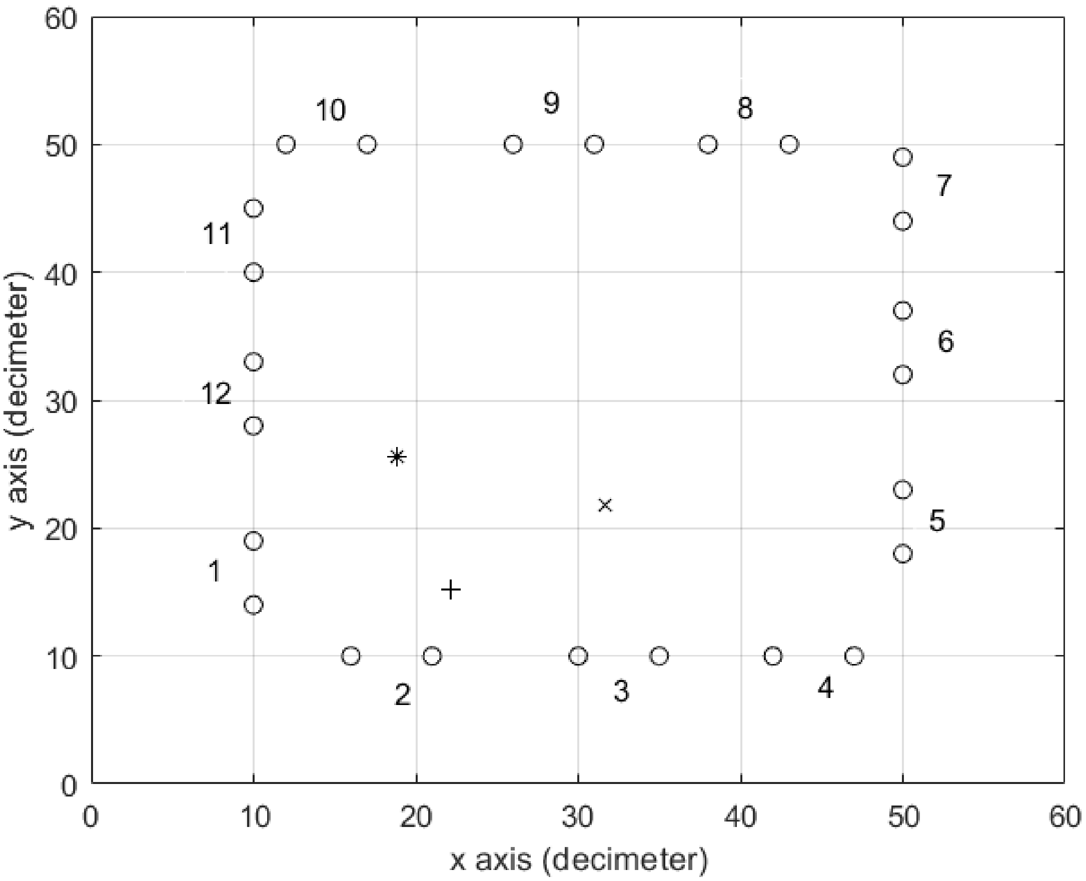


Figure 4.3: Room setup with twelve microphone pairs (circles) with numbers and three speakers ($*$,x,$+$). Grid of 60 × 60 positions.

The microphone pairs are placed at an inter-distance of 50 cm, which was determined to offer a good compromise between resolution and ambiguity [192]. A few simulations were also performed for 10 cm as explained below. The speakers and microphones are assumed to be located in a 2D plane at a height of 135 cm in order to examine the performance when the speakers has a typical mouth height (accuracy of height is not critical).

An example for such a room setup is shown in Fig. 4.3. The microphone pairs are numbered and marked with circles. The three speakers are marked with a $*$, a x and a $+$.

Real room

The room measures 6 × 6 × 2.4 m and has a controllable reverberation time (T_{60}) in the range of 100 msec to 1100 msec. Also here the microphone pairs are placed at an inter-distance of 50 cm. An example of the room setup for one of the room recordings is shown

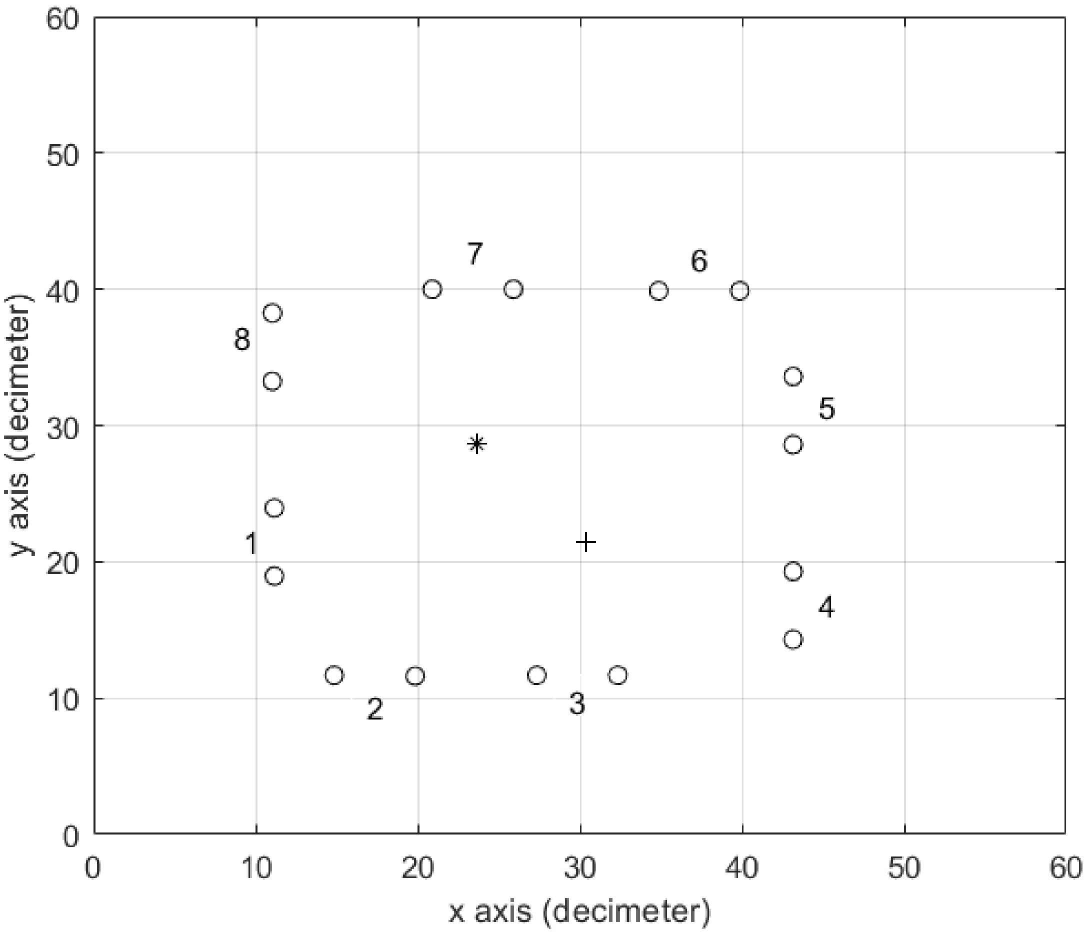


Figure 4.4: Room setup with eight microphone pairs (circles) with numbers and two speakers (∗,+). Grid of 60 × 60 positions.

in Fig. 4.4. The microphone pairs are numbered and marked with circles. The two speakers are marked with a ∗ and a +. For the real recordings, two reverberation setups were tested. We recorded two concurrent speakers using eight microphone pairs. The speakers uttered English sentences while standing in the room. The sampling frequency was set to 48 KHz.

The first speakers were recorded with a low reverberation level ($T_{60} = 200$ msec) and a relatively large distance between them. A snapshot from this recording is depicted in Fig. 4.5.

In order to test a considerably more challenging scenario, the other speakers were recorded in the same room with a much higher reverberation time, $T_{60} = 930$ msec, and with a smaller distance between them. A snapshot from that recording is depicted in Fig. 4.6. The changes in the reflection nature of the floor can be seen in the pictures. Other facets of the room

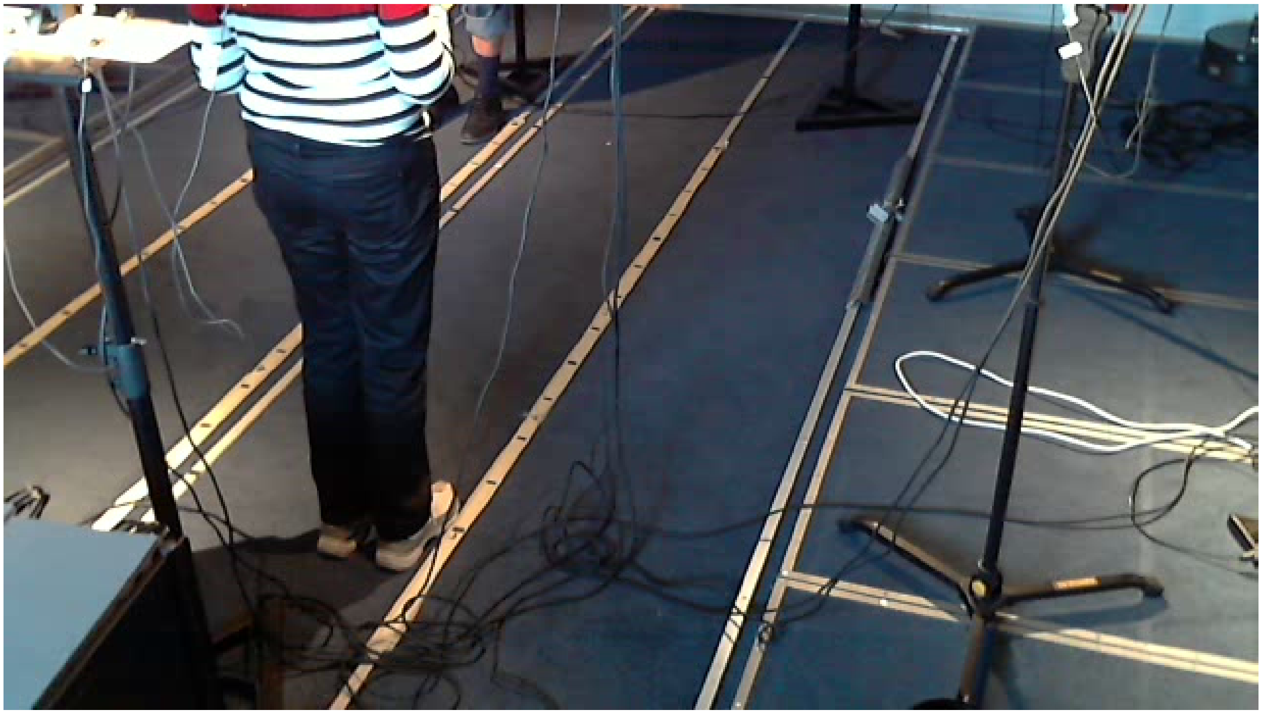


Figure 4.5: First scenario: low reverberation level($T_{60} = 200$ msec); two speakers with a large distance between them.

(ceiling and walls) were changed from absorbing to reflective as well.

4.1.2.2 Parameter setting and complexity of the proposed algorithm

A few practical issues must be addressed regarding the proposed algorithm. They are listed below and refer to all experimental results.

Tuning onset algorithm

The cochlear model provided robust features in previous applications [57]. Some of its parameters were determined experimentally; for example, $w = 6.0$, which is used to scale the bandwidth of the filters. As mentioned above, the number of bands used was $B = 16$. We experimentally chose it, observing significant degradation for lower values in the case of multiple concurrent speakers. This may be attributed to the invalidity of the sparseness (w-disjoint orthogonality) for the lower resolution.

Beam width of a microphone pair

A typical problem, which arises using microphone pairs, is the reduced resolution in the planar domain in the end-fire directions. The case dealt here is 2D with both pairs and speakers on the same plane. It means that broadside measurements have a better resolution.

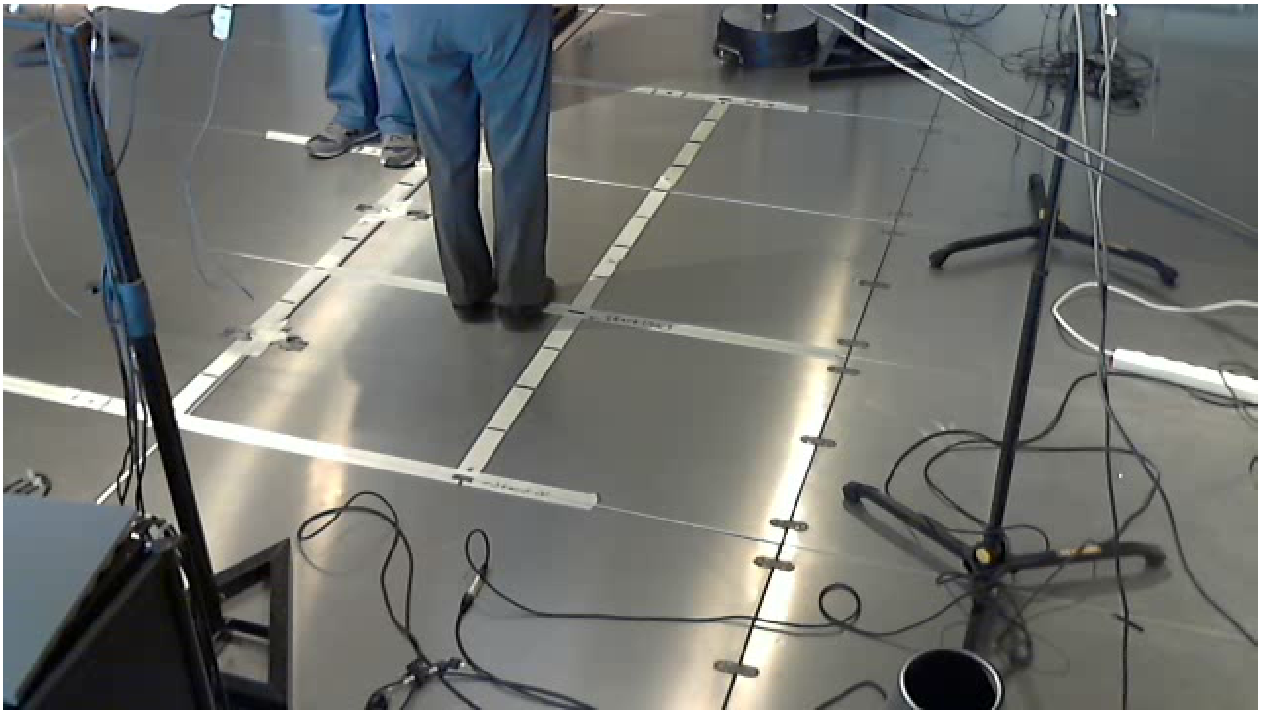


Figure 4.6: Second scenario: high reverberation level ($T_{60} = 930$ msec); two speakers standing close to each other.

As a large number of distributed nodes were used, many nodes would provide accurate MPTs measurements corresponding to near broadside positions.

One of the advantages of using local hidden variables rather than global hidden variables is that the consensus might be achieved even if only part of the nodes reliably measure a specific speaker.

Calculation precision

When summing small numbers and numbers close to 1, precision problems should be addressed. For this reason, a natural log is applied to some of the equations to alleviate dynamic range problems that may arise even for double precision calculations [219, 220]. It is first applied to the Gaussian equation (4.5):

$$\log\left(\mathcal{N}\left(x;\mu,\sigma^2\right)\right) = -\frac{1}{2}\log(2\pi\cdot\sigma^2) - \frac{(x-\mu)^2}{2\cdot\sigma^2} \tag{4.20}$$

The same operation has been applied to the EM equations (4.16)-(4.19). For example, the log version of equation (4.16) is:

$$\log \upsilon_m^{(i)}(t,b,\mathbf{p}) = \log(\hat{\psi}_{\mathbf{p}}^{(i-1)}) + \log\left(g\left(\tau_{m,b}(t);\tilde{\tau}_{m,b}(\tilde{\mathbf{p}}),\sigma^2,F,\tau_{\max}\right)\right) \tag{4.21}$$
$$-\log\left(\sum_{\tilde{\mathbf{p}}}\hat{\psi}_{\tilde{\mathbf{p}}}^{(i-1)} g\left(\tau_{m,b}(t);\tilde{\tau}_{m,b}(\tilde{\tilde{\mathbf{p}}}),\sigma^2,F,\tau_{\max}\right)\right),$$

where log of sum of numbers is calculated by applying the following equation:

$$\log(\sum_{i=1}^{I}(e^{v_i})) = v_{max} + \log\sum_{i=1}^{I} e^{v_i - v_{max}}. \tag{4.22}$$

For probabilities calculations e_i^v are between 0 and 1. It means that v_i are negative real numbers, with v_{max} their maximum.

In addition, multiplication and division operations are substituted by summation and subtraction operations and small numbers can be stored together with much larger ones (within the same vector).

Truncated Gaussian

Another computational issue is the Gaussians truncation, which is applied according to a $\pm F\sigma$ and the physical valid range, where $F = 3$ was empirically tuned. It is notable again that this truncation is applied around each grid point. In addition to what mentioned above, the truncation has a computational advantage, since extremely low probability values are neglected.

Complexity analysis

Complexity analysis is briefly summarized. When we deal with DEM algorithms, we have to examine several aspects. Some of them can be taken from the complexity analysis in [193]:

- We are interested in the number of calculations. Some of them are done once, before the iterations starts. For example the MPTs calculation. Others are applied every iteration. Since all calculations are applied in the nodes and they are all identical, we will analyze the complexity per node. The most significant operations are multiplications and additions, when assuming usage of lookup table implementation of operations like log.

- Communication BW is addressed from a single node perspective, since the process is serial. At each sub-iteration the map of probabilities should be transmitted from one

Table 4.1: The proposed algorithm is analyzed with respect to computation, communication band width and memory.

Criteria	Rough estimation
Computation	$\mathcal{O}(L \cdot T \cdot B \cdot \lvert P \rvert)$
BW	$\mathcal{O}(L \cdot \lvert P \rvert)$
Memory	$\mathcal{O}(T \cdot B + \lvert P \rvert)$

node to the next. As the algorithm converges, this map becomes sparser, since the majority of locations are not active.

- Memory usage per node relates to the local measurements and the latest contribution to the global parameters that should be subtracted in the next iteration.

Table 4.1 summarizes the computational complexity, communication BW and memory requirements of the proposed algorithm.

4.1.2.3 Simulation results

We carried out 100 Monte Carlo trials and calculated three statistical measures: 1) the MD rate, defined as the percentage of speakers that were miss-detected out of the total number of actual speakers; 2) the FA rate, defined as the percentage of falsely-detected speakers normalized by the total number of real speakers; and 3) the RMSE, defined as the accuracy of localization for all successfully detected speakers.

Detection, false- and miss-detections are defined with respect to the threshold applied. The statistical analysis matches each ground truth speaker position to the closest candidate (if any). The RMSE measure is calculated from all the matched positions. If a ground truth speaker does not have a matching detection, it is counted as a miss-detection. We count extra detections as false alarms.

We present here results of various reverberation levels for different speakers locations after Monte Carlo (MC) averaging. In addition, we identified for each number of speakers the level of reverberation for which the proposed algorithm performance is still good and marked them in **bold**. Table 4.2 summarizes the measures for all simulated cases.

Inspecting the results in Table 4.2, it can be deduced that as the reverberation level increases from $T_{60} = 200$ msec to $T_{60} = 600$ msec, the number of speakers allowing reliable localization by the proposed algorithm decreases from three to one, respectively. This can be attributed to the increased density of reflections due to the co-existence of multiple concurrent speakers in the same environment. In any case, the proposed algorithm significantly outperforms the reference algorithms [192, 87] in terms of detection, as well as RMSE. The obtained RMSE values are better than the designated grid resolution. Note that when either MD or FA rates are very high (above 50 percents), the exact statistics is less relevant, since it indicates that controlling the detection threshold did not yield any acceptable FA-MD combination.

4.1.2.4 Experimental insights

We carried out a few additional Monte Carlo simulations to gain insights about the proposed algorithm. This subsubsection contains five different simulation tables.

With or without truncation

The first comparison examines the contribution of the truncation. Working with the regular Gaussian distribution causes artifacts and hence we can produce reasonable results only for low reverberation levels. The comparison is presented for $T_{60} = 150$ msec. We kept the same setup, including inter distance of 50 cm. The results are presented in Table 4.3.

Inspecting the results, it can be deduced that truncation is an essential part of the algorithm. Even for low reverberation level the non-truncated version produces lots of artifacts causing high FA rate. The proposed algorithm detects the same locations with a much lower FA rate.

The influence of setting the variance

The second analysis explores the influence of setting the variance on the performance of the proposed algorithm.

We use inter distance of 50 cm and the case of two speakers mentioned above with $T_{60} = 400$ msec. The results are presented in Table 4.4.

Inspecting the results, it can be deduced that setting the variance to 6 samples^2 provides a good compromise between FAs and MDs.

Sensor noise influence

The third analysis explores the sensor noise influence on the performance. The same setup mentioned above is used, but this time sensor noise is added. The results are presented in Table 4.5.

Inspecting the results, it can be deduced that for this reverberation level SNR of 20 dB degrades the performance significantly.

Number of nodes influence

The fourth analysis explores the number of nodes influence on the performance. The same setup mentioned above is used, but this time only part of the nodes is used. The results are presented in Table 4.6.

As expected, adding nodes improves the localization results. Significant degradation is encountered when the number of nodes is 8 or less. This is a major drawback of the current algorithm and might be a subject for a future research.

Subband GCC-PHAT for TDOAs estimation

The last comparison is of the proposed algorithm and a classical TDOAs estimation, GCC-PHAT adapted to the multi-dimensional localization challenge. The preprocessing of the proposed algorithm applied at each node of the network is replaced with the subband GCC-PHAT estimator. The MPTs are now calculated by the GCC-PHAT. The rest of the algorithm is kept similar.

We tuned the setup according to the limitations of this estimator. For example the inter-distance of 50 cm is too large for this estimator, yielding aliasing effects. Therefore, we tuned it to 10 cm. In addition, the GCC-PHAT cannot deal with reverberation so well, hence a much lower level is simulated ($T_{60} = 150$ msec). After those modifications we compared the two versions using the same setup. The results are presented in Table 4.7.

Inspecting the results, it can be deduced that even in those conditions the GCC-PHAT has inaccurate localization results and high FA rates. The proposed algorithm has very good results even though the inter-distance is only 10 cm. We can observe very small degradation in localization accuracy in the case of three speakers, but the detection rate and FA rate are kept low.

4.1.2.5 Acoustic room results

To further evaluate the performance of the algorithm, we analyzed real recordings as well. The sensor SNR in those cases is around 40 dB. The first examined scenario had a large distance between the speakers and a low reverberation level. The second case is considerably more challenging, since the speakers are closer to each other and the reverberation level is much higher.

In order to examine the results we plot a map, which is the set of $\psi_{\mathbf{p}}^{(LM)}$ values for the grid of positions in the room. The lines are contours of equivalent levels.

The results of the localization algorithms for the first case with a low reverberation level (200 msec) and large distance between the speakers (225 cm) are shown in Fig. 4.7.

It can be observed that the proposed algorithm (shown in Fig. 4.7(a)) produces a very accurate map of the speakers' positions with very low levels of artifacts. It can be easily estimated that there are two speakers. Their locations are identified with high accuracy (taking into account the volume of a speaker's body). The first reference algorithm, IDEM2014 (shown in Fig. 4.7(b)) produces a map with a few spurious peaks and large uncertainty ellipsoid even for this simpler scenario. This means that, in addition to the two real speakers, we might get quite a few FAs. The second reference algorithm, SRP-PHAT (shown in Fig. 4.7(c)) is also not good even for this simple scenario.

The result of the localization algorithms for the second case with a high reverberation level (930 msec) and a small distance between the speakers (100 cm) is depicted in Fig. 4.8.

Even for this challenging case, the proposed algorithm (shown in Fig. 4.8(a)) is able to produce meaningful results. The detection of the speakers can still be obtained and location estimation is very accurate. The results for IDEM2014 (shown in Fig. 4.8(b)) are poor, as in addition to the two real speakers, we get many FAs. The SRP-PHAT (shown in Fig. 4.8(c)) demonstrates unacceptable performance in this challenging case.

4.1.3 Conclusions

In this section we have introduced a new localization algorithm using a bio-inspired cochlear model to achieve robustness against reverberation and handle concurrent speakers. The new algorithm uses a mixture of truncated Gaussian probabilistic model instead of the regular MoG usually used in the EM and the DEM algorithms.

The proposed algorithm is a distributed solution, which uses the IEM principle. Each node locally applies most of the calculations (such as the spikes detection) and has its own hidden variables, sharing only static localization parameters.

As the reverberation effect increases with the number of concurrent speakers, it becomes more challenging to detect the number of speakers and estimate their locations.

Simulations with up to three speakers were carried out in several acoustic scenarios. Real recordings analysis was carried out with two speakers in low and high reverberation levels. The proposed algorithm outperforms IDEM2014 [192] and SRP-PHAT [87] for both the simulated signals and the real-life recordings.

4.2 DOA estimation in presence of late reverberation

A novel DOA estimator for concurrent speakers in reverberant environment is presented. Reverberation, if not properly addressed, is known to degrade the performance of DOA estimators. Following [102], a closed-form solution for the MLE of the time-varying reverberation PSD and the anechoic speech PSD is derived. The reverberation model, proposed in [103], is explicitly incorporated into the MoG-based clustering for DOA estimation proposed in [39].

While in [16] vectors of relative phase between microphone pairs are used for clustering, here we resort to the raw STFT samples. The means of these Gaussians are set to zero, and their covariance matrices include an explicit modeling of both the reverberation and the required DOA information. Consequently, the nuisance parameters, namely the speech and the reverberation PSDs, should also be estimated by the M-step.

The reverberant components are modeled by a time-invariant coherence matrix multiplied by time-varying PSD. The PSDs of the anechoic speech and reverberant components are estimated as part of the EM procedure. It is shown that the DOA estimates, obtained by the proposed algorithm, are less affected by reverberation than competing algorithms that ignore the reverberation. Experimental study demonstrates the benefit of the presented algorithm in reverberant environment using measured RIRs.

4.2.1 Proposed direction of arrival estimator

In this subsection, an EM procedure for estimating the sources' DOAs in the maximum likelihood sense, is derived. The EM algorithm requires the definition of three datasets and their probability model: 1) the observations; 2) the target parameters; and 3) the hidden data. The first two datasets were already defined in Section. 2.3.

Here we define the hidden data to be the *association* of each T-F bin with a single source transmitting from a particular angle. Define the hidden data, $x(t,k,\vartheta)$, to be the indicator that the T-F bin (t,k) belongs to a speaker from angle ϑ. The total number of indicators in the problem is $T \times K \times V$. Note that at each T-F bin only a single indicator equals 1. In the sequel, the EM steps are derived.

The same way applied in [16, Eq. (17)], the p.d.f. of the complete dataset, $\boldsymbol{z}$ and $\mathbf{x}$, is:

$$f(\boldsymbol{z},\mathbf{x};\boldsymbol{\theta}) = \prod_{t,k}\sum_{\vartheta=1}^{V}\psi_{\vartheta}x(t,k,\vartheta)\mathcal{N}^c\left(\boldsymbol{z}(t,k);\mathbf{0},\mathbf{\Phi}_{\vartheta}(t,k)\right), \tag{4.23}$$

where $\mathbf{x} = \mathrm{vec}_{t,k,\vartheta}\left(\{x(t,k,\vartheta)\}\right)$.

The EM algorithm for the problem at hand can now be derived. The E-step is given by:

$$\begin{aligned} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(\ell-1)}) &\triangleq E\left\{\log\left(f(\boldsymbol{z},\mathbf{x};\boldsymbol{\theta})\right)|\boldsymbol{z};\boldsymbol{\theta}^{(\ell-1)}\right\} \\ &= \sum_{t,k,\vartheta} E\left\{x(t,k,\vartheta)|\boldsymbol{z}(t,k);\boldsymbol{\theta}^{(\ell-1)}\right\} \times \left[\log\psi_\vartheta + \log\mathcal{N}^c\left(\boldsymbol{z}(t,k);\mathbf{0},\boldsymbol{\Phi}_\vartheta(t,k)\right)\right], \end{aligned} \tag{4.24}$$

where ℓ is the iteration index.

As already shown in the previous chapter, in order to implement the E-step, the hidden variables are evaluated by:

$$\begin{aligned} \mu^{(\ell-1)}(t,k,\vartheta) &\triangleq E\left\{x(t,k,\vartheta)|\boldsymbol{z}(t,k);\boldsymbol{\theta}^{(\ell-1)}\right\} \\ &= \frac{\psi_\vartheta^{(\ell-1)}\mathcal{N}^c\left(\boldsymbol{z}(t,k);\mathbf{0},\boldsymbol{\Phi}_\vartheta^{(\ell-1)}(t,k)\right)}{\sum_{\vartheta'}\psi_{\vartheta'}^{(\ell-1)}\mathcal{N}^c\left(\boldsymbol{z}(t,k);\mathbf{0},\boldsymbol{\Phi}_{\vartheta'}^{(\ell-1)}(t,k)\right)}, \end{aligned} \tag{4.25}$$

where $\boldsymbol{\Phi}_\vartheta^{(\ell)}(t,k) = \boldsymbol{g}_\vartheta\boldsymbol{g}_\vartheta^{\mathrm{H}}\phi_{S,\vartheta}^{(\ell)}(t,k) + \boldsymbol{\Gamma}(k)\phi_{R,\vartheta}^{(\ell)}(t,k)$. $\boldsymbol{g}_\vartheta$ and $\boldsymbol{\Gamma}(k)$ were defined in Section 2.3.

Maximizing (4.24) with respect to the parameters $\boldsymbol{\theta}$ constitutes the M-step. Similarly to [16, Eq. (20a)], ψ_ϑ is obtained by the constrained maximization of (4.24):

$$\psi_\vartheta^{(\ell)} = \frac{\sum_{t,k}\mu^{(\ell-1)}(t,k,\vartheta)}{T\,K}. \tag{4.26}$$

Note that in (4.26) the TDOA probabilities are estimated using all time and frequency bins.

From (4.24), the M-step expressions for $\phi_{R,\vartheta}(t,k)$ and $\phi_{S,\vartheta}(t,k)$ are obtained by maximizing only $\log\mathcal{N}^c\left(\boldsymbol{z}(t,k);\mathbf{0},\boldsymbol{\Phi}_\vartheta^{(\ell)}(t,k)\right)$. Following [102, Eqs. (16) and (17)], the M-step for $\phi_{R,\vartheta}(t,k)$ and $\phi_{S,\vartheta}(t,k)$ are given by:

$$\phi_{R,\vartheta}^{(\ell)}(t,k) = \frac{1}{N-1}\boldsymbol{z}^{\mathrm{H}}(t,k)\boldsymbol{\Gamma}^{-1}(k)\left[\boldsymbol{I} - \boldsymbol{g}_\vartheta(k)\boldsymbol{w}_\vartheta^{\mathrm{H}}(k)\right]\boldsymbol{z}(t,k), \tag{4.27a}$$

$$\phi_{S,\vartheta}^{(\ell)}(t,k) = \boldsymbol{w}_\vartheta^{\mathrm{H}}(k)\left[\boldsymbol{z}(t,k)\boldsymbol{z}^{\mathrm{H}}(t,k) - \phi_{R,\vartheta}^{(\ell)}(t,k)\,\boldsymbol{\Gamma}(k)\right]\boldsymbol{w}_\vartheta(k), \tag{4.27b}$$

where $\boldsymbol{w}_\vartheta(k) \equiv \frac{\boldsymbol{\Gamma}^{-1}(k)\boldsymbol{g}_\vartheta(k)}{\boldsymbol{g}_\vartheta^{\mathrm{H}}(k)\boldsymbol{\Gamma}^{-1}(k)\boldsymbol{g}_\vartheta(k)}$.

The DOA estimation is summarized in Algorithm 8.

Note that, if $\epsilon \to 0$ (defined in Section 2.3 as diagonal loading), $\boldsymbol{w}_\vartheta(k)$ identifies with the *maximum directivity* BF that attenuates diffuse noise field (i.e. reverberation in our case), while steering a beam towards the direction of the candidate DOA ϑ.

Algorithm 8 DOA estimation with late reverberation.

```
**initialize** $\psi_\vartheta^{(0)} = \frac{1}{TK}$.
**for** $\ell = 1$ **to** $L$ **do**
    **E-step**
    Calculate $\mu^{(\ell-1)}(t,k,\vartheta)$.
    **M-step**
    Calculate the parameters, $\psi_\vartheta^{(\ell)}$.
**end**
```

Note that as $\phi_{R,\vartheta}^{(\ell)}(t,k)$ and $\phi_{S,\vartheta}^{(\ell)}(t,k)$ are independent of $\mu^{(\ell-1)}(t,k,\vartheta)$. It means that the outcome of the E-step, $\phi_{R,\vartheta}^{(\ell)}(t,k)$ and $\phi_{S,\vartheta}^{(\ell)}(t,k)$ can be calculated in advance. Denoting L as the number of iterations, the DOAs and the number of sources can be determined by analyzing $\psi_\vartheta^{(L)}$, as described in Subsec. 4.2.2.

4.2.2 Performance evaluation

The performance of the proposed algorithm is evaluated by estimating the DOA of two concurrent speakers. The simulation setup is elaborated in Subsubsec. 4.2.2.1. In Subsubsec. 4.2.2.2, the accuracy of the proposed DOA estimator in comparison with two competing estimators is evaluated.

4.2.2.1 Simulation setup

Anechoic speech signals were convolved by RIRs, downloaded from an open-source database recorded in our lab. The reverberation time was adjusted by flipping the room panels, and was measured to be approximately $T_{60} = \{0.16, 0.36, 0.61\}$ sec. Details about the database can be found in [221].

The loudspeakers were positioned at various angles on a half a circle with radius 2 m facing a four microphone linear array, with inter-distances [3, 8, 3] cm. The sampling frequency was 16 kHz, the frame length of the STFT was 64 ms with no overlap. The number of frequency bins was 1024. Two utterances, $\sim$ 4.5 sec long, of male and female speakers, were used.

While applying the algorithm, only the frequencies below 4 kHz were used. In addition, we used only time-frequency bins for which $\mathrm{En}(t,k) > 10^{-5}\,\mathrm{Max}\,\{\mathrm{En}(t,k)\}$, where $\mathrm{En}(t,k) \equiv \frac{1}{N}\boldsymbol{z}^{\mathrm{H}}(t,k)\boldsymbol{z}(t,k)$ denotes the average energy of the (t,k)-th bin. The number of iterations L was set to 20. The Gaussians were uniformly distributed in the range $0° - 179°$ with

1° resolution, namely $\vartheta = \vartheta_o^\circ$ with $\vartheta_o = 0, 1, \ldots, V = 179$. The weights of the MoG were uniformly initialized to $\psi_\vartheta^{(0)} = \frac{1}{180}$ and ϵ was set to 0.5.

The performance of the proposed algorithm was compared with two other competing algorithms: 1) the steered response power (SRP)-PHAT [82], 2) our previous localization algorithm [16] denoted henceforward 'Schwartz2014' (with necessary modifications applied to obtain DOAs rather than positions). The outputs of the SRP-PHAT were normalized to sum to 1, to allow for clear comparison with the probability curves of the proposed algorithm and 'Schwartz2014'.

4.2.2.2 The direction of arrival estimation performance

In the first set of experiments we compared the performance of the three algorithm for $T_{60} = 0.61$ sec, by averaging all possible two-speakers combinations in the range $15^\circ - 165^\circ$. Since, the resolution of the database is 15°, we have 11 angles (altogether $11 \times 10 = 110$ different two-speakers combinations). For each scenario, female and male anechoic signals were filtered with the corresponding RIR for $T_{60} = 0.61$ sec.

Similarly to search-based algorithms, e.g. [82, 222], the exact number of sources cannot be determined from the localization results. Here, the two DOAs with the highest probabilities are selected as estimates. The MAE was calculated as the average of all errors between the estimated DOA and the true angle. In Table 4.8, the MAEs for the various algorithms are presented. It can be seen that the proposed algorithm outperforms the competing algorithms.

We further examine a specific scenario with one combination of speakers' locations for various levels of reverberation. The results are depicted in Fig. 4.9. The speakers were positioned at 45° and 105°, respectively. The probabilities of each DOA are shown for the three levels of reverberation. The SRP-PHAT fails to separate between the speakers even in non-reverberant scenario. 'Schwartz2014' locates well the speakers in low reverberation conditions but tends to deviate from the true DOAs in high reverberation. The proposed algorithm demonstrates low deviations from the true DOAs even in the highest reverberation level.

4.2.3 Conclusions

A DOA estimator for concurrent speakers in reverberant environment was presented. The proposed algorithm uses the EM procedure for clustering T-F bins under a MoG probabilistic model. The presented algorithm differs from a previously proposed algorithm [16], by the explicitly modeling the reverberation.

The anechoic speech and reverberation PSDs were also estimated as nuisance parameters. An experimental study demonstrated the advantage of the proposed DOA estimator in actual reverberant environments, compared with the methods proposed in [16] and [82].

Table 4.2: Localization statistics for 100 Monte Carlo trials (Inter distance 50 cm). The first column indicates the reverberation level. The second one contains the number of simulated speakers that were randomly located.

T_{60}	Sim. speakers	Algorithm	MD[%]	FA[%]	RMSE[cm]
600	**1**	**Proposed**	0.0	6.0	4
600	**1**	**IDEM2014**	10.0	63.0	99
600	**1**	**SRP-PHAT**	95.0	1.0	206
400	1	Proposed	0.0	4.0	4
400	1	IDEM2014	6.0	73.0	99
400	1	SRP-PHAT	95.0	1.0	157
200	1	Proposed	0.0	3.0	4
200	1	IDEM2014	41.0	14.0	175
200	1	SRP-PHAT	91.0	1.0	153
600	2	Proposed	31.5	16.5	12
600	2	IDEM2014	88.5	3.0	116
600	2	SRP-PHAT	99.0	1.0	101
400	**2**	**Proposed**	8.0	9.0	4
400	**2**	**IDEM2014**	35.5	54.0	73
400	**2**	**SRP-PHAT**	97.0	0.5	138
200	2	Proposed	0.0	8.5	4
200	2	IDEM2014	16.0	67.0	82
200	2	SRP-PHAT	75.0	8.0	149
400	3	Proposed	21.3	34.3	10
400	3	IDEM2014	56.7	22.7	82
400	3	SRP-PHAT	96.3	1.3	103
200	**3**	**Proposed**	2.3	4.0	4
200	**3**	**IDEM2014**	41.0	35.3	97
200	**3**	**SRP-PHAT**	97.0	0.3	126

Table 4.3: Comparison with and without truncation (Inter distance 50 cm): 100 Monte Carlo trials of two randomly located simulated speakers.

T_{60}	Sim. speakers	Algorithm	MD[%]	FA[%]	RMSE[cm]
150	2	Proposed	0.0	5.0	4
150	2	No-Trunc	0.0	100.0	4

Table 4.4: Variance influence(Inter distance 50 cm): 100 Monte Carlo trials with two randomly located simulated speakers

T_{60}	Sim. speakers	σ^2[samp^2]	MD[%]	FA[%]	RMSE[cm]
400	2	16	27.0	3.0	4
400	**2**	**6**	**8.0**	**9.0**	**4**
400	2	1	1.0	50.0	4
400	2	0.01	0.0	100.0	5

Table 4.5: Sensor noise influence(Inter distance 50 cm, T_{60} = 400 msec): 100 Monte Carlo trials with two randomly located simulated speakers

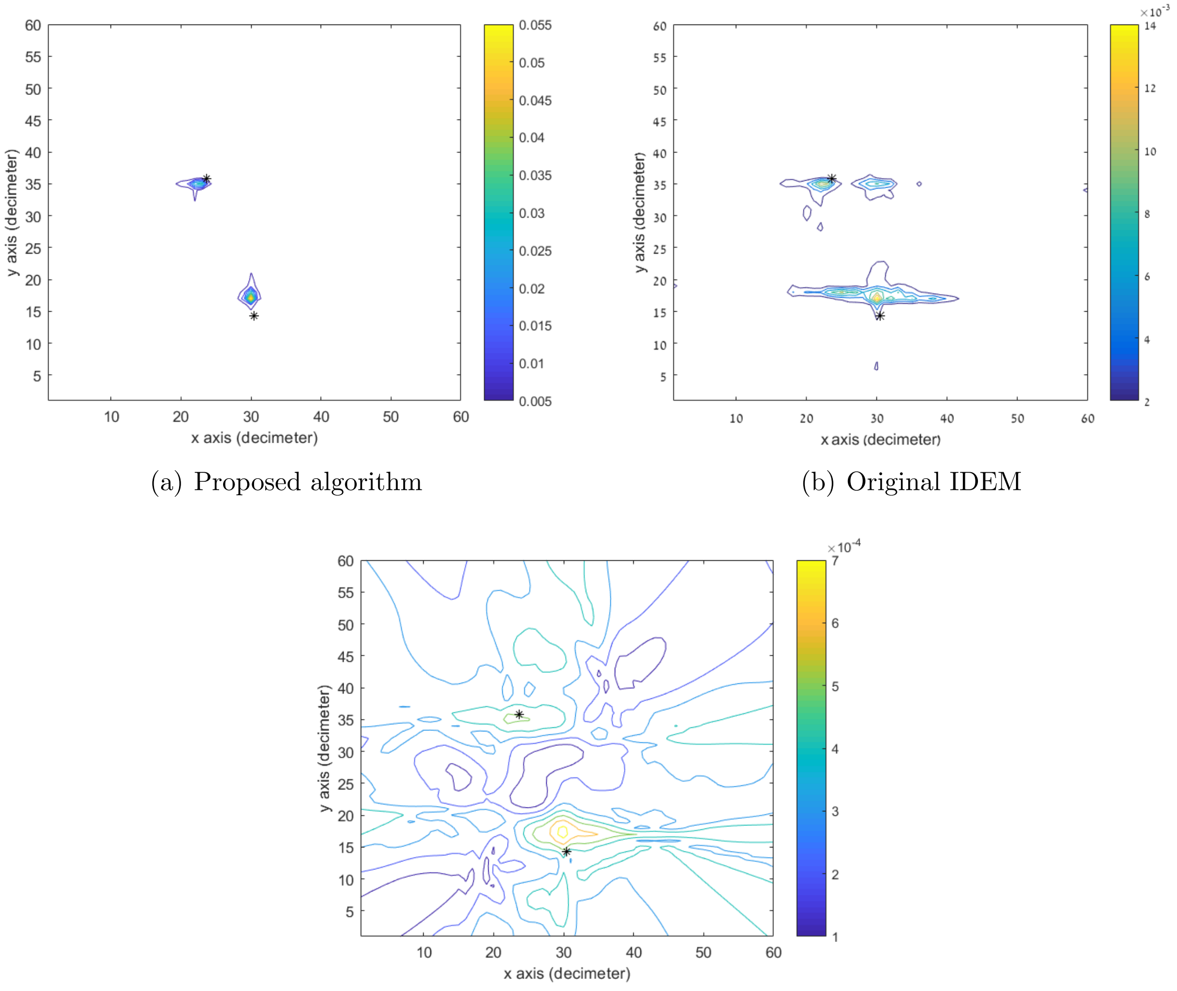
SNR[dB]	Sim. speakers	MD[%]	FA[%]	RMSE[cm]
0	2	100.0	0.0	–
20	2	41.0	1.5	5
30	2	13.0	9.0	4
40	2	11.0	11.0	4
Noiseless	2	8.0	9.0	4

Table 4.6: Number of nodes influence(Inter distance 50 cm, T_{60} = 400 msec): 100 Monte Carlo trials with two randomly located simulated speakers

Nodes	Sim. speakers	MD[%]	FA[%]	RMSE[cm]
6	2	6.5	72.5	6
8	2	12.4	28.1	6
10	2	12.0	16.0	4
11	2	9.5	12.5	4
12	2	8.0	9.0	4

Table 4.7: Subband GCC-PHAT for TDOA estimation (Inter distance 10 cm): 100 Monte Carlo trials with two or three randomly located simulated speakers

T_{60}	Sim. speakers	Algorithm	MD[%]	FA[%]	RMSE[cm]
150	2	Proposed	0.0	5.0	4
150	2	GCC-PHAT	0.5	88.5	80
150	3	Proposed	2.3	0.0	6
150	3	GCC-PHAT	8.3	69.7	78



(a) Proposed algorithm

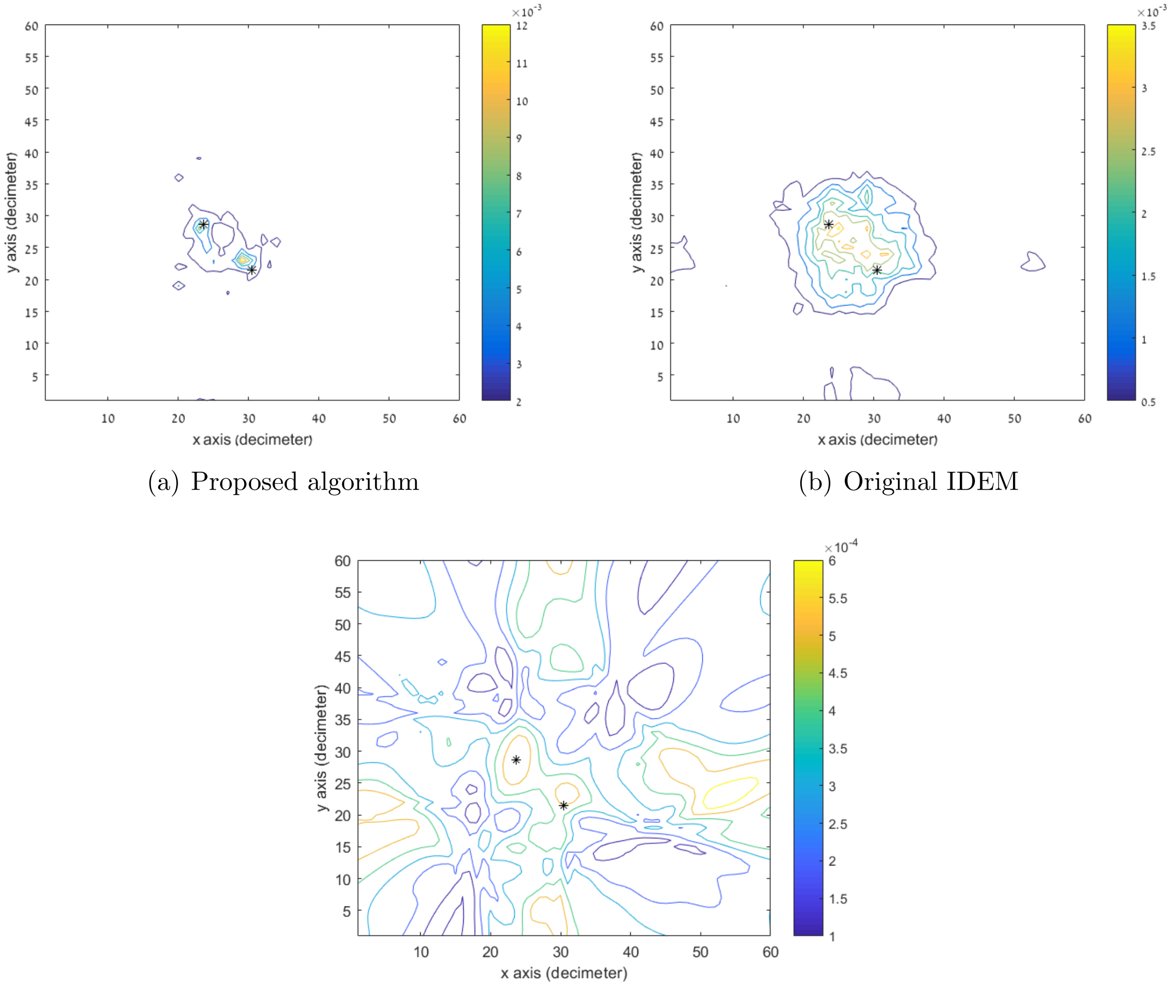
(b) Original IDEM

(c) SRP-PHAT

Figure 4.7: Low reverberation level ($T_{60} = 200\,\text{msec}$), two speakers, recording length 12 sec. Real positions are marked by asterisks.

Speaker \ Alg.	SRP-PHAT	Schwartz2014	Proposed
Female	16.58°	11.31°	6.8°
Male	19.03°	15.66°	7.46°

Table 4.8: MAEs for the three alternative algorithms, averaged on 110 possible pairs of signals in the range $15° - 165°$ for $T_{60} = 0.61$ sec.

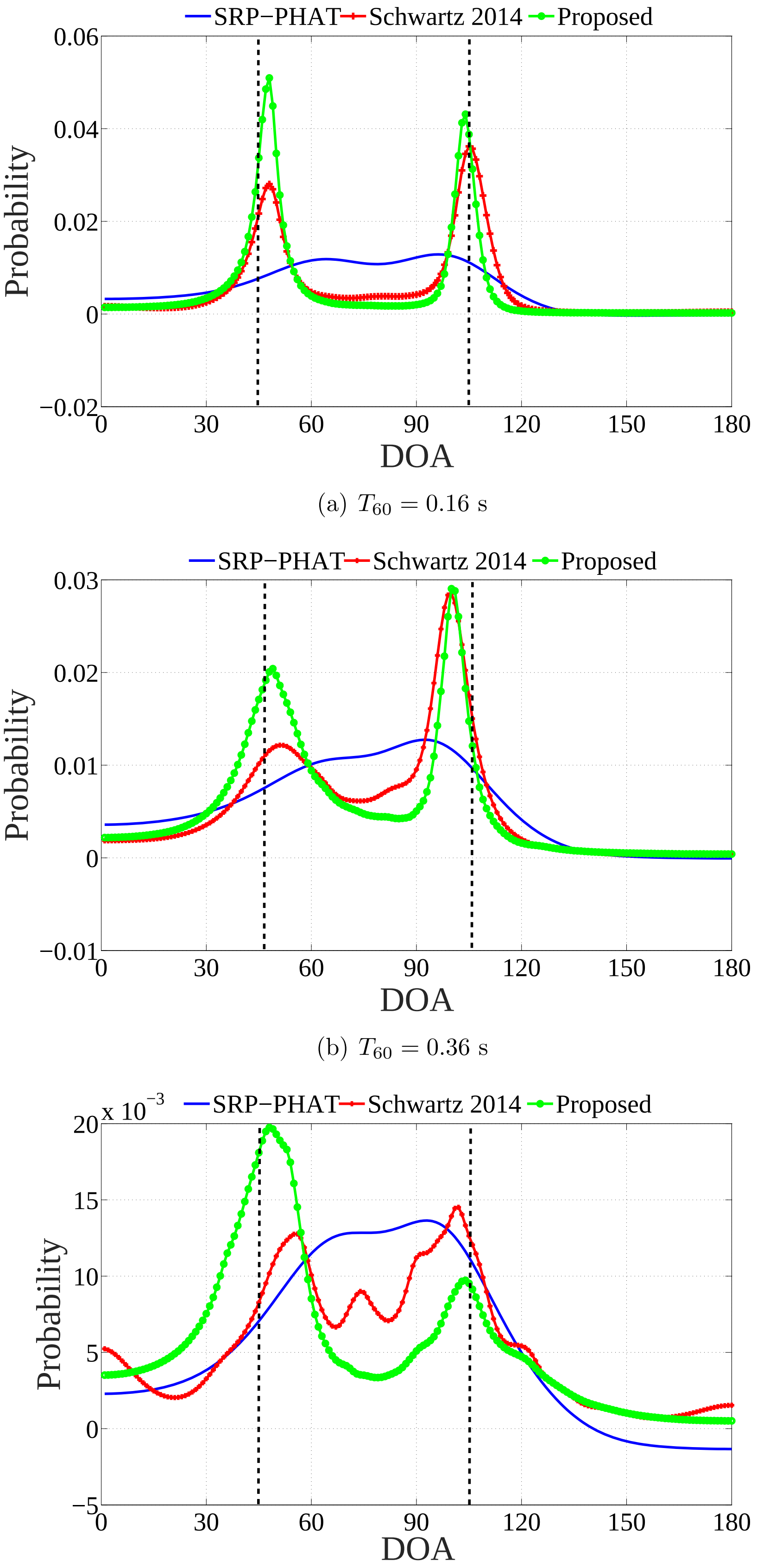


(a) Proposed algorithm

(b) Original IDEM

(c) SRP-PHAT

Figure 4.8: Maximal reverberations level ($T_{60} = 930\,\text{msec}$), two speakers, recording length 12 sec. Real positions are marked by asterisks.



(a) $T_{60} = 0.16$ s

(b) $T_{60} = 0.36$ s

(c) $T_{60} = 0.61$ s

Figure 4.9: Probabilities (y-axis) vs. DOA (x-axis) for two speakers at 45° and 105°.

Chapter 5

Blind source separation

The material presented in this chapter is based on [197] and [198]:

[197] Y. Dorfan, D. Cherkassky, and S. Gannot, "Speaker localization and separation using incremental distributed expectation-maximization," in *European Signal Processing Conference (EUSIPCO)*, 2015, pp. 1256–1260

[198] Y. Dorfan, O. Schwartz, B. Schwartz, E. A. P. Habets, and S. Gannot, "Multiple DOA estimation and blind source separation using estimation-maximization," in *IEEE Science of Electrical Engineering (ICSEE)*, Eilat, Israel, 2016

After estimating the information about the number (detections) and the positions (localization), we try to achieve BSS. We extend the localization algorithm to address the second task, namely blindly separating the speech sources.

The basic idea is common to both approaches. The algorithms are comprised of two steps. The first step estimates localization parameters regarding each active source. By-products of the localization step (such as activity masks) are utilized by the second step, the BSS.

The first approach is a distributed algorithm. The localization is solved by all nodes together, while the separation can then be applied locally.

The second approach enables usage of any array geometry. The localization information in this case refers only to far field DOAs.

5.1 Distributed blind source separation

The proposed algorithm, denoted distributed algorithm for localization and separation (DALAS), is capable of speaker separation in a reverberant enclosure without a priori information on the number of sources and their locations. In the first stage of the proposed algorithm, the localization EM is applied for blindly detecting the active sources and to estimate their locations. Examples for that first stage are the IDEM from Chapter 3 or DOA estimation algorithm from Chapter 4. In the second stage, the localization estimates are utilized for selecting the most useful node of microphones for the subsequent separation stage. Separation is finally obtained by utilizing the hidden variables of the EM algorithm to construct masks for each source in the relevant node.

We utilize the location information to choose the best node for extracting each source. This is done simply by choosing the nearest pair. This usage is nice, but a much bigger effect is achieved when using a by-product of the localization step. For each source we construct its spectral mask. This mask and the global localization information are utilized for node level filtering.

The hidden variables used for localization can be utilized as masks for source separation [197], since it associates time-frequency bins and room positions.

Source separation might be obtained by utilizing spectral masking. Hard masking is the simplest method, which might however result in the *musical noise* phenomenon. A technique that reduces this noise is called soft masking. We will apply the following combination of soft and hard masking using a threshold on the estimated local indicators. The result will be used as a mask for the STFT representation of the input signal. The proposed mask calculation is given (after L iterations of M nodes) by:

$$C_m^n(t,k) = \begin{cases} 1, & \upsilon_m^{(L\cdot M)}(t,k,\hat{\mathbf{p}}^n) > T_H \\ 0, & \upsilon_m^{(L\cdot M)}(t,k,\hat{\mathbf{p}}^n) < T_L \\ \upsilon_m^{(L\cdot M)}(t,k,\hat{\mathbf{p}}^n), & \text{otherwise} \end{cases}, \tag{5.1}$$

where T_H and T_L are the high and the low thresholds, respectively. Their selection is a trade-off between decreasing interference power and maintaining desired spectral contents.

The masking is applied at the best node of each source. The node signals $z_m^r(t,k)$ are masked by multiplying their STFT by the mask in (5.1), a value in the range $[0,1]$. Although applied locally, the location estimates are using the global information (through the control mechanism), hence potentially improving the credibility of the mask.

The masked signals, $\hat{s}_m^{r,n}(t,k)$, for speaker n at microphones $r = 1,2$ of node m are transformed back to the time domain using inverse short-time Fourier transform (ISTFT)

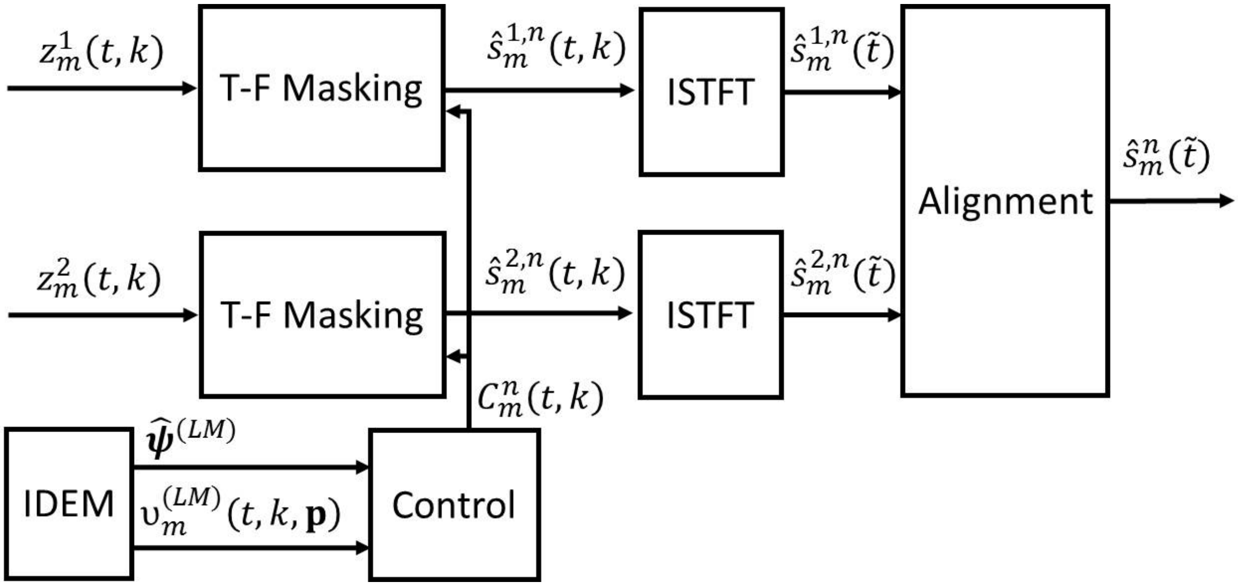


Figure 5.1: Filtering at the mth microphone pair.

and then aligned (sub-sample delay applied) and averaged:

$$\hat{s}_m^n(\tilde{t}) = \frac{1}{2}\sum_{r=1}^{2} \hat{s}_m^{r,n}(\tilde{t} - \hat{t}_m^{r,n}), \tag{5.2}$$

where $\hat{t}_m^{r,n} = \frac{||\hat{\mathbf{p}}^n - \mathbf{p}_m^r||}{c}$ is the delay between the location of source n and the microphones of node m, and $\tilde{t}$ is the time index. This alignment can be either implemented in the time-domain or in the frequency-domain. The result of (5.2) is a signal with an enhanced direct path. The BSS algorithm is summarized in a block-diagram depicted in Fig. 5.1. The localization parameters are utilized to detect the number of active speakers and their locations. The hidden variables are used to build spectral masks.

The filtering part of the DALAS is examined by a signal to interference ratio (SIR) measurement for the input and the output signals, calculated in the following way:

$$\mathrm{SIR}^n = 10\log\left(\frac{E^n}{\sum_{\tilde{n}\neq n} E^{\tilde{n}}}\right) \text{ dB}, \tag{5.3}$$

where E^n is the nth source power.

Two examples of two concurrent sources, a man and a woman, are presented. In the first scenario the distance between the sources is rather large. In this case, based on the localization results, the algorithm selected the 1st node as the best node for extracting the

	Distant		Close	
	Man	Woman	Man	Woman
$\mathrm{SIR_i}$	9	14	-1	4
$\mathrm{SIR_o}$	19	17	19	20

Table 5.1: Separation measures for two sources in two cases

woman and the 11th node as the best node for extracting the man. This selection results in high input SIR and hence potentially improves the output separation quality. The measures are summarized in Table 5.1.

The second example is more challenging. The sources are very close to each other and located in the bottom left corner of the room. The localization results are depicted in Fig. 5.2.

In this case, the algorithm selected the 4th node as the best node for separating the man and the 1st node as the best node for separating the woman. In this case the input SIR is low, since the sources are very close to each other. However, the algorithm is still capable of separating the sources and improving the SIRs significantly as evident from Table 5.1.

This algorithm should be modified in order to be more robust for high reverberation levels. The support of multiple sources should be expanded from two to three or more concurrent sources. The masking of low frequencies is sometimes poor. It might be improved if we use more than 2 microphones per node. This might be a subject of future research study.

5.2 Centralized blind source separation

A blind source separation technique in noisy environment is proposed based on spectral masking and MVDR-BF. Formulating the ML of the DOAs and solving it using the EM, enables the extraction of the masks and the associated MVDR as byproducts. The adopted estimator uses an explicit model of the ambient noise, which results in more accurate DOAs estimation and good source separation.

In this work, we adopt the DOA estimation procedure presented in [196] (subsection 4.2.1) for deriving an estimator for multiple DOA in noisy environments (and low reverberation) based on the EM algorithm. The speech PSD level is also estimated by the M-step, while the noise PSD matrix is assumed to be known. To achieve source separation, we propose to utilize two by-products of the EM, namely the MVDR-BF and the associated T-F mask. An experimental study, using real RIRs and artificial diffuse noise, shows improved DOA estimates compared to [16, 82]. The MVDR-BFs and the spectral masks associated with the DOA estimates are then used to separate the speakers.

The experimental study demonstrates both DOA estimation results and the separation capabilities of the proposed method using real room impulse responses and diffuse noise.

5.2.1 The expectation-maximization for direction of arrivals estimation

In this subsection, the ML estimate of the sources' DOAs are obtained using the EM algorithm. The EM algorithm uses three datasets and their probability model: the observations, the target parameters (these datasets were already defined in section. 2.4) and the hidden data. We use $x(t,k,\vartheta)$ defined in the previous chapter as the hidden data. This is the *association* of each T-F bin with a single source emitting from angle ϑ.

In the same way derived in [16, Eq. (17)], the p.d.f. of the complete dataset, $\boldsymbol{z}$ and $\mathbf{x}$, is:

$$f(\boldsymbol{z},\mathbf{x};\boldsymbol{\theta}) = \prod_{t,k}\sum_{\vartheta=1}^{V} \psi_\vartheta x(t,k,\vartheta)\mathcal{N}^c\left(\boldsymbol{z}(t,k);\mathbf{0},\boldsymbol{\Phi}_\vartheta(t,k)\right), \tag{5.4}$$

where $\mathbf{x} = \text{vec}_{t,k,\vartheta}\left(\{x(t,k,\vartheta)\}\right)$.

The EM algorithm for the problem at hand is now derived. The E-step is given by:

$$\begin{aligned} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(\ell-1)}) &\triangleq E\left\{\log\left(f(\boldsymbol{z},\mathbf{x};\boldsymbol{\theta})\right)|\boldsymbol{z};\boldsymbol{\theta}^{(\ell-1)}\right\} \\ &= \sum_{t,k,\vartheta} E\left\{x(t,k,\vartheta)|\boldsymbol{z}(t,k);\boldsymbol{\theta}^{(\ell-1)}\right\}\left[\log\psi_\vartheta + \log\mathcal{N}^c\left(\boldsymbol{z}(t,k);\mathbf{0},\boldsymbol{\Phi}_\vartheta(t,k)\right)\right]. \end{aligned} \tag{5.5}$$

For implementing the E-step, as already shown in the previous chapters, the hidden variables are evaluated by:

$$\begin{aligned} \mu^{(\ell-1)}(t,k,\vartheta) &\triangleq E\left\{x(t,k,\vartheta)|\boldsymbol{z}(t,k);\boldsymbol{\theta}^{(\ell-1)}\right\} \\ &= \frac{\psi_\vartheta^{(\ell-1)}\mathcal{N}^c\left(\boldsymbol{z}(t,k);\mathbf{0},\boldsymbol{\Phi}_\vartheta^{(\ell-1)}(t,k)\right)}{\sum_s \psi_\vartheta^{(\ell-1)}\mathcal{N}^c\left(\boldsymbol{z}(t,k);\mathbf{0},\boldsymbol{\Phi}_\vartheta^{(\ell-1)}(t,k)\right)}, \end{aligned} \tag{5.6}$$

where $\boldsymbol{\Phi}_\vartheta^{(\ell)}(t,k) = \boldsymbol{g}_\vartheta\boldsymbol{g}_\vartheta^{\mathrm{H}}\phi_{S,\vartheta}^{(\ell)}(t,k) + \boldsymbol{\Phi}_{\boldsymbol{v}}(t,k)$. Note that $\mu^{(\ell-1)}(t,k,\vartheta)$ is actually a soft T-F mask following the activity of the speaker from the ϑ-th angle in the (t,k)-th T-F bin.

Maximizing (5.5) with respect to the parameters $\boldsymbol{\theta}$ constitutes the M-step. Similarly to [16, Eq. (20a)], ψ_ϑ is obtained by a constrained[1] maximization of (5.5):

$$\psi_\vartheta^{(\ell)} = \frac{\sum_{t,k} \mu^{(\ell-1)}(t,k,\vartheta)}{T \cdot K}. \tag{5.7}$$

From (5.5), the M-step expressions for $\phi_{S,\vartheta}(t,k)$ are obtained by maximizing $\log \mathcal{N}^c \left(\boldsymbol{z}(t,k); \mathbf{0}, \boldsymbol{\Phi}_\vartheta^{(\ell)}(t,k) \right)$. Following [102, Eq.(17)], the M-step for $\phi_{S,\vartheta}(t,k)$ can be expressed as:

$$\phi_{S,\vartheta}^{(\ell)}(t,k) = \boldsymbol{w}_\vartheta^{\mathrm{H}}(k) \left[\boldsymbol{z}(t,k)\boldsymbol{z}^{\mathrm{H}}(t,k) - \boldsymbol{\Phi}_{\boldsymbol{v}}(k) \right] \boldsymbol{w}_\vartheta(k), \tag{5.8}$$

where $\boldsymbol{w}_\vartheta(k) \equiv \frac{\boldsymbol{\Phi}_{\boldsymbol{v}}^{-1}(k)\boldsymbol{g}_\vartheta(k)}{\boldsymbol{g}_\vartheta^{\mathrm{H}}(k)\boldsymbol{\Phi}_{\boldsymbol{v}}^{-1}(k)\boldsymbol{g}_\vartheta(k)}$. Note that $\boldsymbol{w}_\vartheta(k)$ is the MVDR-BF steered towards ϑ that reduces the ambient noise.

Note also that $\phi_{S,\vartheta}^{(\ell)}(t,k)$ is independent on the output of the E-step $\mu^{(\ell-1)}(t,k,\vartheta)$ and can therefore be calculated in advance.

5.2.2 Source separation using minimum variance distortion-less response beamformer and soft mask

The DOA estimates can be taken as the DOA values associated with the Gaussians with the largest probability. Let $\bar{\vartheta}$ be an index of one of the DOA estimates. Using the mask $\mu^{(L-1)}(t,k,\bar{\vartheta})$ obtained by the E-step and the MVDR-BF $\boldsymbol{w}_{\bar{\vartheta}}(k)$, the individual speech signal from angle $\bar{\vartheta}$ can be estimated by:

$$\widehat{S}_{\bar{\vartheta}}(t,k) = \mu^{(L-1)}(t,k,\bar{\vartheta})\, \boldsymbol{w}_{\bar{\vartheta}}^{\mathrm{H}}(k)\, \boldsymbol{z}(t,k), \tag{5.9}$$

where $\mu^{(L-1)}(t,k,\bar{\vartheta})$ is responsible for enhancing the $\bar{\vartheta}$-th speaker and attenuating the other speakers and $\boldsymbol{w}_{\bar{\vartheta}}(k)$ is responsible for reducing the ambient noise.

5.2.3 Performance evaluation

The performance of the proposed algorithm is evaluated with a scenario of two concurrent speakers. The simulation setup is elaborated in Subsubsec. 5.2.3.1. In Subsubsec. 5.2.3.2, the DOA estimation and BSS in noisy environment is performed.

[1] The sum of ψ_ϑ equals 1.

5.2.3.1 Simulation setup

Anechoic speech signals were convolved by RIRs, downloaded from a freely-available database recorded in our lab [221]. Reverberation time was set by adjusting the room panels and was measured as $T_{60} \cong 0.16$ sec. Note that since the reverberation is low, a free-field propagation can be assumed. The loudspeakers were positioned in front of a four microphone linear array. The inter-distances between the microphones were [3, 8, 3] cm. The sampling frequency was 16 kHz, the frame length of the STFT was 64 msec with overlap of 75%. The number of frequency bins was 1024. Two utterances (~ 4.5 sec long) of male and female speakers were used.

The speakers were positioned at 60° and 105° degrees. An artificial diffuse noise[2] was added to the speech signals with various SNR levels. Accordingly, the noise PSD matrix was modeled as $\mathbf{\Phi}_{\boldsymbol{v}}(k) = \phi_V(k)\mathbf{\Gamma}(k)$, where:

$$\Gamma_{ij}(k) = \operatorname{sinc}\left(\frac{2\pi k}{K}\frac{d_{i,j}}{T_{\mathrm{s}}c}\right) + \epsilon\delta(i-j) \tag{5.10}$$

and $\phi_V(k)$ was estimated using speech absent segments.

The diagonal loading ϵ was set to 0.1. The frequency band $300 - 3400$ Hz was used. We use for localization and BSS only T-F bins for which the a posteriori SNR was higher than a predefined threshold:

$$\mathrm{aSNR}(t,k) = 10\log_{10}\frac{\boldsymbol{z}^{\mathrm{H}}(t,k)\boldsymbol{z}(t,k)}{\mathrm{Tr}\left[\mathbf{\Phi}_{\boldsymbol{v}}(k)\right]} > \lambda, \tag{5.11}$$

where $\mathrm{Tr}\left[\cdot\right]$ is the trace operation. The number of iterations L was set to 40.

To emphasize the effectiveness of the proposed DOA estimator, two other competing DOA estimators were applied: 1) the SRP-PHAT, 2) our previous localization algorithm [16] denoted henceforward Schwartz2014[3]. The outputs of the SRP-PHAT were normalized to sum of 1.

[2] Details on the diffuse noise generator can be found in [207] and the software can be downloaded from `https://www.audiolabs-erlangen.de/fau/professor/habets/software/noise-generators`

[3] Modifications applied to Schwartz2014: 1) The original algorithm estimates the coordinates rather than the angle. 2) Instead of using a distributed structure of microphone pairs, we use a single array. Thus, we refer to the array as a set of pairs, where each possible pair is used.

SNR_i	SIR_i	$\text{SNR}_{o,1}$	$\text{SNR}_{o,2}$	$\text{SIR}_{o,1}$	$\text{SIR}_{o,2}$
4.0	0.0	12.5	6.5	12.1	6.8
20.0	0.0	25.0	22.1	11.5	11.1

Table 5.2: The input and output objective measures (in dB). SNR_i and SIR_i are the SNR and SIR at the input, respectively. $\text{SNR}_{o,j}$ and $\text{SIR}_{o,j}$ relate to the j-th estimated signal, where $j = 1, 2$.

5.2.3.2 The performance of direction of arrival estimation and blind source separation in noisy environment

The DOA results for the three algorithms are compared in Fig. 5.3 for various levels of SNR.

The SRP-PHAT can barely separate between the speakers even in 15 dB SNR. Schwartz2014 accurately locates the speakers in high SNR conditions, but tends to exhibits many erroneous peaks in low SNR. The proposed DOA estimator presents noticeable peaks at the true DOAs even for low SNR levels.

We further apply the signal separation procedure using (5.9). The SNR and the SIR values at the output of the algorithm were calculated using the following procedure derived in [223]. The filtering coefficients in (5.9) were calculated for every T-F bin, and for every speaker using the noisy mixture signals $\boldsymbol{z}(t,k)$, and afterwards applied separately to each of the steered source signals $\boldsymbol{g}_j(k)S_j(t,k)$, and to the noise signal $\boldsymbol{v}(t,k)$. This procedure enables quantitative analysis of the proposed method with respect to each of the mixture components. The input/output SNR and SIR values are shown in Table. 5.2.

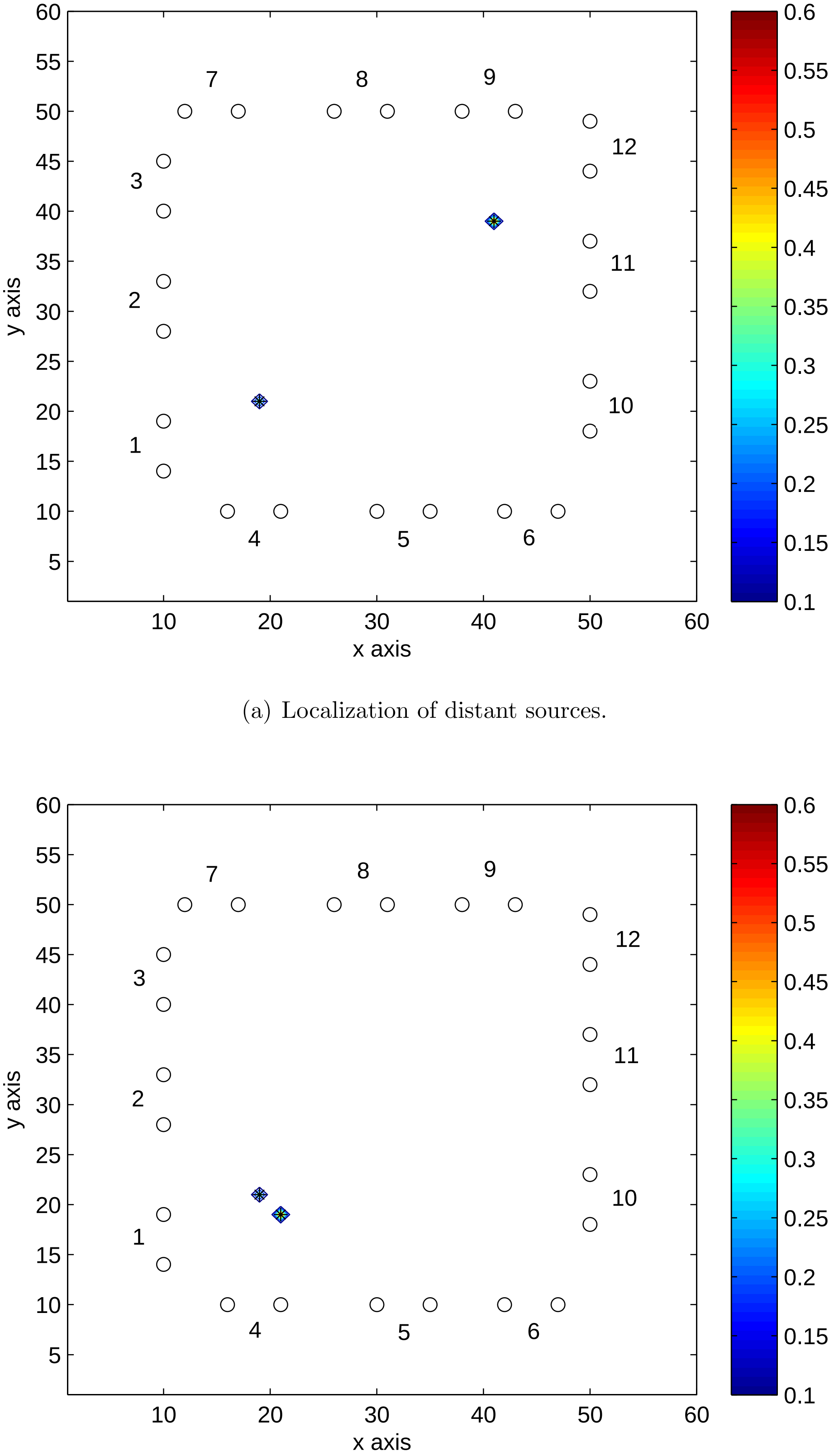
The computation method for SIRs was already described in the previous section. All measurements are in dB. The SNR_i and SIR_i are defined at the input. $\text{SNR}_{o,j}$ and $\text{SIR}_{o,j}$ are the SNR and SIR for the j-th estimated signal, respectively, where $j = 1, 2$.

It can be observed that the SNR and the SIR are improved for both speakers even in high noise conditions.

5.2.4 Conclusions

Localization algorithms within the EM framework were used for BSS: a three dimension localization algorithm and a DOA estimator. A spectral mask obtained in the E-step and the MVDR-BF in the M-step were utilized for jointly separating the sources and reducing the noise.

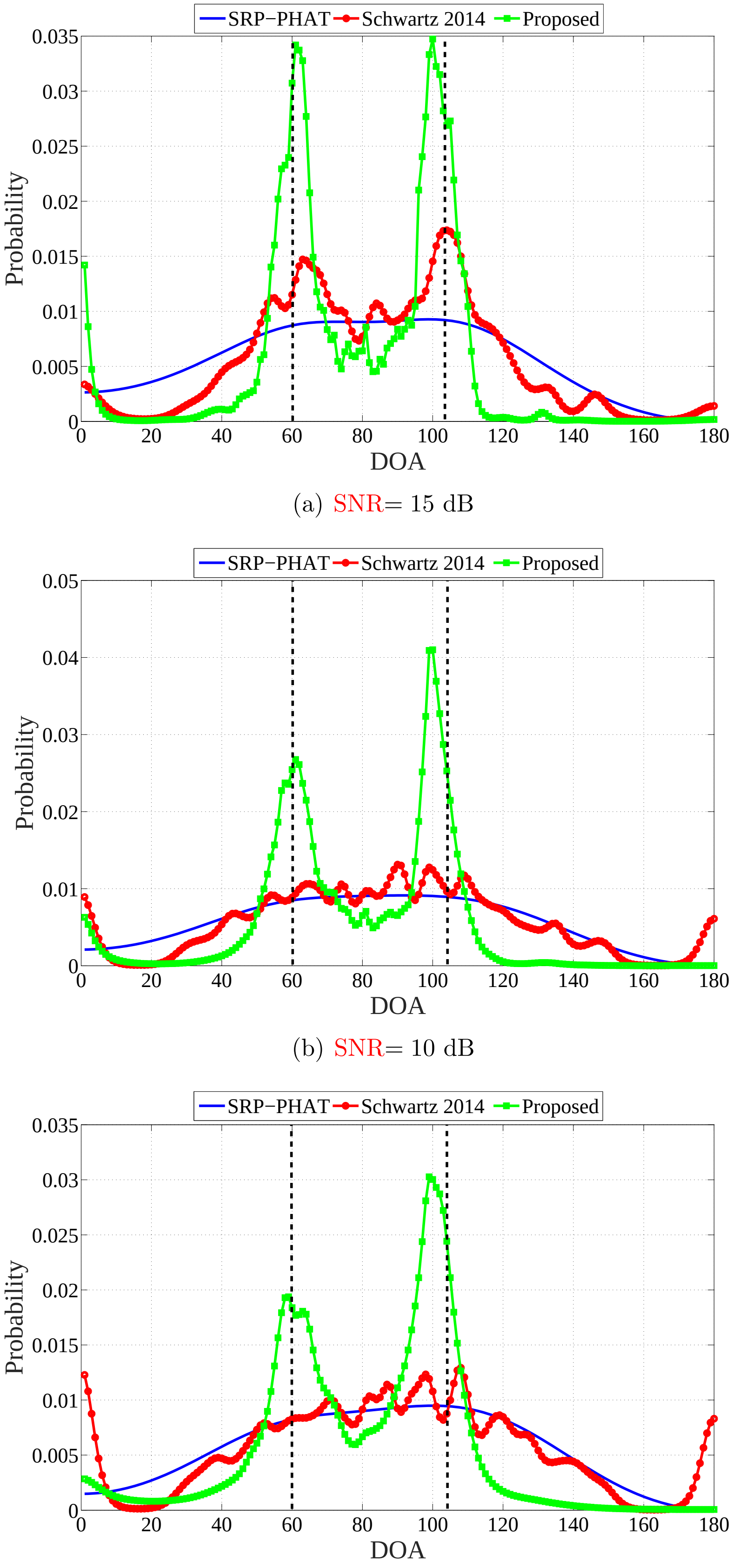
An experimental study demonstrated the advantage of the proposed DOA estimation in the presence of artificial diffuse noise over competing methods. The effectiveness of the proposed BSS technique is demonstrated for real recordings with two different levels of diffuse noise.



(a) Localization of distant sources.

(b) Localization of close sources.

Figure 5.2: Network constellation and Localization results.



(a) SNR= 15 dB

(b) SNR= 10 dB

(c) SNR= 5 dB

Figure 5.3: DOA Probabilities $\psi_{\vartheta}^{(L)}$ for two speakers, positioned at 60° and 105°, for various SNRs.

Chapter 6

Joint Speaker Localization and Array Calibration using Expectation-Maximization

The material presented in this chapter is based on a planned publication [199]:

[199] Y. Dorfan, O. Schwartz, and S. Gannot, "Joint speaker localization and array calibration using expectation-maximization," *To be submitted to IEEE...*

During the previous chapters the positions of the sensors were given, however, ad hoc acoustic networks naturally lack the microphone positions that are often required for e.g. localization, tracking and beamforming. The challenge of joint multiple speaker localization and array calibration is addressed using a novel variant of the EM algorithm. The microphone inter-distances in each array, as well as the orientation of each array, are assumed known in advance. However, the center points of the arrays and the locations of the sources are unknown in advance and are jointly estimated by the algorithm.

The challenge is to solve the localization problem of multiple (more than two) concurrent speakers jointly with the calibration problem of multiple arrays, without any other information or any additional calibration signals. Following [16], we use the EM and the MoG models to cluster the observed data to centroids located on a grid on the room surface. An explicit model of the speech and noise is defined within the MoG model, as used in [198].

To perform also the calibration task, we add the locations of the array centers to the estimation task. As a result, the locations of the array centers are estimated in the M-step. Maximization of the auxiliary function of the EM with respect to (w.r.t.) the array centers does not produce a closed form expression. We assume that the noise signals of different arrays are uncorrelated, which enables us to avoid a multidimensional search of the array centers (that is, a separate search for each array is obtained).

The initialization stage was found to be a complex task because of the large parameter set. We found a novel way for self-initialization that uses the collected data in an incremental fashion. One of the arrays is designated as the anchor array, and all of the other elements (arrays and sources) are localized w.r.t. the anchor. The algorithm is first activated only with the anchor array while the other arrays are disabled, and then the other arrays in the network are added one after the other with the iterations of the EM algorithm.

The chapter is organized as follows. Section 6.1 presents the problem formulation. The EM algorithm is derived in Section 6.2. The performance evaluation is given in the third section, followed by a conclusion section.

6.1 Problem Formulation

We derive a batch EM solution to a static joint estimation. The problem formulation is divided into two subsections. The first describes the ad hoc network signals in the presence of multiple concurrent speakers and sensor noise, and the second subsection deals with their statistical model.

6.1.1 Signal model

Consider Q arrays, each equipped with N microphones receiving signals from J speakers. The number of speakers is not necessarily known in advance. The measured signals are linear combinations of the incoming waveforms. Let $Z_{q,n}(t,k)$ be the STFT coefficients of the signals received by the (q,n)-th microphone, where $q = 1, \ldots, Q$ is the array index and $n = 1, \ldots, N$ is the microphone index within each array. Overall, there are $Q \times N$ microphones. The signals in the STFT domain are given by:

$$Z_{q,n}(t,k) = \sum_{j=1}^{J} G_{q,n,j}(k) \cdot S_j(t,k) + V_{q,n}(t,k), \tag{6.1}$$

where $t = 0, \ldots, T-1$ and $k = 0, \ldots, K-1$ denote the time and frequency indexes, respectively. $G_{q,n,j}(k)$ is the direct transfer function (DTF) associating source j to microphone

(q,n). $S_j(t,k)$ is the speech signal uttered by speaker j and $V_{q,n}(t,k)$ is the ambient noise.

The N microphone signals in the q-th array can be concatenated in a vector form:

$$\boldsymbol{z}_q(t,k) = \sum_{j=1}^{J} \boldsymbol{g}_{q,j}(k) S_j(t,k) + \boldsymbol{v}_q(t,k), \tag{6.2}$$

where:

$$\boldsymbol{z}_q(t,k) = \begin{bmatrix} Z_{q,1}(t,k) & \dots & Z_{q,N}(t,k) \end{bmatrix}^{\mathrm{T}} \tag{6.3}$$

$$\boldsymbol{g}_{q,j}(k) = \begin{bmatrix} G_{q,1,j}(k) & \dots & G_{q,N,j}(k) \end{bmatrix}^{\mathrm{T}} \tag{6.4}$$

$$\boldsymbol{v}_q(t,k) = \begin{bmatrix} V_{q,1}(t,k) & \dots & V_{q,N}(t,k) \end{bmatrix}^{\mathrm{T}}. \tag{6.5}$$

The overall observation set, DTFs, and noise components can be concatenated by:

$$\boldsymbol{z}(t,k) = \begin{bmatrix} \boldsymbol{z}_1^T(t,k) & \dots & \boldsymbol{z}_Q^T(t,k) \end{bmatrix}^{\mathrm{T}}, \tag{6.6}$$

$$\boldsymbol{g}_j(k) = \begin{bmatrix} \boldsymbol{g}_{1,j}^T(k) & \dots & \boldsymbol{g}_{Q,j}^T(k) \end{bmatrix}^{\mathrm{T}}, \tag{6.7}$$

$$\boldsymbol{v}(t,k) = \begin{bmatrix} \boldsymbol{v}_1^T(t,k) & \dots & \boldsymbol{v}_Q^T(t,k) \end{bmatrix}^{\mathrm{T}}, \tag{6.8}$$

such that:

$$\boldsymbol{z}(t,k) = \sum_{j=1}^{J} \boldsymbol{g}_j(k) S_j(t,k) + \boldsymbol{v}(t,k). \tag{6.9}$$

Note that the DTF model includes the attenuation of the direct speech wave and the respective phase. The DTF is therefore:

$$G_{q,n,j}(k) = \frac{1}{d_{q,n,j}^2} \exp\left(-\iota \frac{2\pi k}{K} \frac{d_{q,n,j}}{c \cdot T_s} \right), \tag{6.10}$$

where c is the sound velocity and T_s denotes the sampling period. The distance from speaker j to microphone (q,n), $d_{q,n,j}$ is simply given by:

$$d_{q,n,j} = ||\boldsymbol{p}_j - \boldsymbol{p}_{q,n}||, \tag{6.11}$$

where $\boldsymbol{p}_j$ is the location of the jth speaker and $\boldsymbol{p}_{q,n}$ is the location of the (q,n)-th microphone given by:

$$\boldsymbol{p}_{q,n} = \boldsymbol{p}_q + \boldsymbol{p}_n(q), \tag{6.12}$$

where $\boldsymbol{p}_q$ is the position of the center of the qth array and $\boldsymbol{p}_n(q)$ is the relative position of the n-th microphone with respect to the array center. The inter-structure of the arrays and their orientation (namely, $\boldsymbol{p}_n(q)$) are assumed to be known in advance. Note that the *orientation* of the arrays can be supplied using various means, for example, compass-based technology.

The goal of this study is to jointly estimate the speakers' locations $\boldsymbol{p}_j$ and the arrays' center positions $\boldsymbol{p}_q$.

6.1.2 Statistical model

We use a MoG probability function to characterize the speech signal of all potential speakers. Each speaker can be assumed to be a complex-Gaussian source emitting acoustic waveform from location $\boldsymbol{p}_m$, where m is the index of the Gaussian component. Because the number of speakers and their locations are unknown in advance, we use a predefined grid as candidate source positions.

The various speakers are assumed to exhibit disjoint activity in the STFT domain (W-disjoint assumption by [110]). This means that, by means of clustering, every T-F bin of $\boldsymbol{z}(t,k)$ can be associated with a single active source.

Based on the disjoint activity of the sources, the observations can be given the probabilistic description:

$$\boldsymbol{z}(t,k) \sim \sum_{m=1}^{M} \psi_m \cdot \mathcal{N}^c\left(\boldsymbol{z}(t,k); \boldsymbol{0}, \boldsymbol{\Phi}_m(t,k)\right), \tag{6.13}$$

where ψ_m is the (unknown) probability of a speaker present at $\boldsymbol{p}_m$ and M is the number of Gaussians.

The matrix $\boldsymbol{\Phi}_m(t,k)$ is the PSD of $\boldsymbol{z}(t,k)$, given that $\boldsymbol{z}(t,k)$ is associated with the speaker located at $\boldsymbol{p}_m$:

$$\boldsymbol{\Phi}_m(t,k) = \boldsymbol{g}_m(k)\boldsymbol{g}_m^{\mathrm{H}}(k)\phi_{S,m}(t,k) + \boldsymbol{\Phi}_{\boldsymbol{v}}(k), \tag{6.14}$$

where the DTF $\boldsymbol{g}_m(k)$ is defined in (6.7).

The direct-path temporal PSD $\phi_{S,m}(t,k)$ and the noise PSD matrix $\boldsymbol{\Phi}_{\boldsymbol{v}}(t,k)$ are given by:

$$\phi_{S,m}(t,k) = E\left\{|S_m(t,k)|^2\right\}, \tag{6.15}$$

$$\boldsymbol{\Phi}_{\boldsymbol{v}}(k) = E\left\{\boldsymbol{v}(t,k)\boldsymbol{v}^{\mathrm{H}}(t,k)\right\}. \tag{6.16}$$

The noise components from different arrays are often assumed to be uncorrelated [126], and thus:

$$\boldsymbol{\Phi}_{\boldsymbol{v}}(k) = \text{Blockdiag}\left[\begin{array}{ccc} \boldsymbol{\Phi}_{\boldsymbol{v}_1}(k) & \dots & \boldsymbol{\Phi}_{\boldsymbol{v}_Q}(k) \end{array}\right], \tag{6.17}$$

where $\boldsymbol{\Phi}_{\boldsymbol{v}_q}(k) = E\left\{\boldsymbol{v}_q(t,k)\boldsymbol{v}_q^{\mathrm{H}}(t,k)\right\}$.

The PSD matrices of the noise are assumed to be time-invariant and known in advance, or they can be estimated during speech absence segments.

Finally, by augmenting all observations for $t = 0, \dots, T-1$ and $k = 0, \dots, K-1$ in $\boldsymbol{z} = \text{vec}_{t,k}(\{\boldsymbol{z}(t,k)\})$, the p.d.f. of the entire observation set can be stated as:

$$f(\boldsymbol{z}) = \prod_{t,k}\sum_{m=1}^{M} \psi_m \cdot \mathcal{N}^c\left(\boldsymbol{z}(t,k); \boldsymbol{0}, \boldsymbol{\Phi}_m(t,k)\right), \tag{6.18}$$

where the readings for all T-F bins are assumed to be independent.

Let the unknown parameter set be $\boldsymbol{\theta} = \left[\boldsymbol{p}^{\mathrm{T}}, \boldsymbol{\psi}^{\mathrm{T}}, \boldsymbol{\phi}_S^{\mathrm{T}}\right]^{\mathrm{T}}$, where $\boldsymbol{p} = \text{vec}_q\left(\boldsymbol{p}_q\right)$, $\boldsymbol{\psi} = \text{vec}_m\left(\psi_m\right)$ and $\boldsymbol{\phi}_S = \text{vec}_{m,t,k}\left(\phi_{S,m}(t,k)\right)$. It should be emphasized that, unlike the arrays' locations, the parameters for localization $\boldsymbol{\psi}^{\mathrm{T}}$ are soft decisions in nature. After solving the joint estimation problem, the number and locations of speakers is estimated from those soft values.

The MLE problem can readily be stated as:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\text{argmax}} \log f\left(\boldsymbol{z}; \boldsymbol{\theta}\right). \tag{6.19}$$

In the next section, an algorithm is derived for finding $\boldsymbol{\theta}$. The first two components are the required parameters (array centers and source positions). The last component is a set of nuisance parameters.

Because the MLE in this case is of high complexity, it is necessary to use an iterative search algorithm. A widely used algorithm for this kind of problem is the EM algorithm. In the next section, we first derive the basic (batch) version of the algorithm. For performance improvement, and to mitigate the dependency on the algorithm initialization, we also introduce a novel modification of the basic EM.

6.2 The LACES algorithm

In this section, the MLE of $\boldsymbol{\theta}$ is solved using the EM algorithm. It uses three datasets and their probability models: the observations, the target parameters (these datasets were already defined in Sec. 2.4), and the hidden data sets. In our case, we define the hidden data to be 1) the speech signals $S_m(t,k)$, which are potentially emitted from each location m in the room and 2) the *association* of each T-F bin with a single source emitting from a particular location, as in [16].

The *association* of each T-F bin is expressed by $x(t,k,m)$, an indicator that the bin (t,k) is associated with a speaker located at $\boldsymbol{p}_m$. The total number of indicators in the problem is $T \times K$. Note that, under the W-disjoint assumption [110], each T-F bin is dominated by a single speaker.

This section is split into four subsections. In the first, the basic EM equations are derived. The second is dedicated to the E-step and the third to the M-step. The last subsection summarizes the algorithm and its initialization process.

6.2.1 Basic expectation maximization steps derivation

Denote the hidden data as:

$$\mathbf{x} = \text{vec}_{t,k,m}\left(\{x(t,k,m)\}\right) \tag{6.20}$$

$$\mathbf{s} = \text{vec}_{t,k,m}\left(\{S_m(t,k)\}\right). \tag{6.21}$$

Following Bayes' rule, the p.d.f. of the complete dataset, $\boldsymbol{z}$, $\mathbf{x}$ and $\boldsymbol{s}$, is obtained by:

$$f(\boldsymbol{z},\mathbf{x},\boldsymbol{s};\boldsymbol{\theta}) = f(\boldsymbol{z}|\mathbf{x},\boldsymbol{s};\boldsymbol{\theta})f(\mathbf{x}|\boldsymbol{s};\boldsymbol{\theta})f(\boldsymbol{s};\boldsymbol{\theta}). \tag{6.22}$$

The conditional distribution of the observed data given the hidden data can be expressed as:

$$f(\boldsymbol{z}|\mathbf{x},\boldsymbol{s};\boldsymbol{\theta}) = \prod_{t,k}\sum_{m=1}^{M} x(t,k,m) f(\boldsymbol{z}(t,k)|x(t,k,m)=1,\boldsymbol{s};\boldsymbol{\theta}). \tag{6.23}$$

Using the assumption that noise signals from different arrays are uncorrelated (6.17), the p.d.f. of the noise signals from all arrays is expressed as a multiplication of the p.d.f.s of the noise signals from each array:

$$f(\boldsymbol{z}(t,k)|x(t,k,m)=1,\boldsymbol{s};\boldsymbol{\theta}) = \mathcal{N}^c\left(\boldsymbol{z}(t,k)-\boldsymbol{g}_m(k)S_m(t,k);\mathbf{0},\boldsymbol{\Phi}_{\boldsymbol{v}}(k)\right)$$
$$= \prod_q \mathcal{N}^c\left(\boldsymbol{z}_q(t,k)-\boldsymbol{g}_{q,m}(k)S_m(t,k);\mathbf{0},\boldsymbol{\Phi}_{\boldsymbol{v}_q}(k)\right). \quad (6.24)$$

Because the indicator x is independent of speech signals, $\boldsymbol{s}$, its conditional p.d.f. is given by:

$$f(\mathbf{x}|\boldsymbol{s};\boldsymbol{\theta}) = f(\mathbf{x};\boldsymbol{\theta}) = \prod_{t,k}\sum_{m=1}^{M} x(t,k,m)\psi_m. \quad (6.25)$$

The speech p.d.f. is frequently assumed to follow a complex Gaussian distribution:

$$f(\boldsymbol{s};\boldsymbol{\theta}) = \prod_{t,k,m}\mathcal{N}^c\left(S_m(t,k);0,\phi_{S,m}(t,k)\right). \quad (6.26)$$

The p.d.f. of the complete dataset is then given by collecting the terms in (6.22)-(6.26):

$$f(\mathbf{x},\boldsymbol{z},\boldsymbol{s};\boldsymbol{\theta}) = \left(\prod_{t,k}\sum_{m=1}^{M} x(t,k,m)\psi_m \times \prod_q \mathcal{N}^c\left(\boldsymbol{z}_q(t,k)-\boldsymbol{g}_{q,m}(k)S_m(t,k);\mathbf{0},\boldsymbol{\Phi}_{\boldsymbol{v}_q}(k)\right)\right)$$
$$\times\left(\prod_{t,k,m}\mathcal{N}^c\left(S_m(t,k);0,\phi_{S,m}(t,k)\right)\right). \quad (6.27)$$

6.2.2 E-step

The first step of the iterative algorithm is approximation of statistics of the hidden data. The auxiliary function is then given by (6.28) where for any variable $\boldsymbol{a}$ the denotation $\widehat{\boldsymbol{a}}$ means $E\left\{\boldsymbol{a}|\boldsymbol{z};\boldsymbol{\theta}^{(\ell-1)}\right\}$. Note that, because of the indicator properties of $x(t,k,m)$, the summation over m is carried out outside the logarithm operation.

$$
\begin{aligned}
Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(\ell-1)}) \triangleq & \widehat{\log f(\boldsymbol{z}, \mathbf{x}, \boldsymbol{s}; \boldsymbol{\theta})} = \\
& \sum_{t,k,m} \widehat{x}(t,k,m) \Bigg(\log \psi_m + \sum_q \widehat{\log \mathcal{N}^c \left(\boldsymbol{z}_q(t,k) - \boldsymbol{g}_{q,m}(k) S_m(t,k); \mathbf{0}, \boldsymbol{\Phi}_{\boldsymbol{v}_q}(k) \right)} \Bigg) \\
& + \sum_{t,k,m} \widehat{\log \mathcal{N}^c \left(S_m(t,k); 0, \phi_{S,m}(t,k) \right)} \qquad (6.28)
\end{aligned}
$$

For implementing the E-step, the sufficient statistics of the hidden variables are evaluated by the following expressions:

1. The expected associations:

$$
\begin{aligned}
& \widehat{x}^{(\ell)}(t,k,m) \triangleq E\left\{ x(t,k,m) | \boldsymbol{z}(t,k); \boldsymbol{\theta}^{(\ell-1)} \right\} = \\
& \frac{\psi_m^{(\ell-1)} \mathcal{N}^c \left(\boldsymbol{z}(t,k); \mathbf{0}, \boldsymbol{\Phi}_m^{(\ell-1)}(t,k) \right)}{\sum_{m'} \psi_{m'}^{(\ell-1)} \mathcal{N}^c \left(\boldsymbol{z}(t,k); \mathbf{0}, \boldsymbol{\Phi}_{m'}^{(\ell-1)}(t,k) \right)},
\end{aligned}
\qquad (6.29)
$$

where:

$$
\boldsymbol{\Phi}_m^{(\ell-1)}(t,k) = \boldsymbol{g}_m^{(\ell-1)}(k) \cdot \left(\boldsymbol{g}_m^{(\ell-1)}(k) \right)^{\mathrm{H}} \phi_{S,m}^{(\ell-1)}(t,k) + \boldsymbol{\Phi}_{\boldsymbol{v}}(k). \qquad (6.30)
$$

Note that the direct path $\boldsymbol{g}_m^{(\ell-1)}(k)$ is calculated before each E-step according to the estimated array locations for all grid points of the sources. The expression for $\boldsymbol{g}_m^{(\ell-1)}(k)$ is given by (6.7) and (6.10), while exchanging the source index j with the candidate location index m and using the estimated array positions $\boldsymbol{p}_q^{(}l-1)$.

2. The next term for the E-step is the first-order statistics of the speech, given the measurements and the parameters. Using the law of total expectation, the first-order statistics of the speech is then given by (6.31). Note that the expectation of the mth speaker (when the (t,k) bin is associated with the mth speaker) is the multichannel Wiener filter (MWF) (see [224, Eq.(28)]); otherwise, the expectation is zero (because it is a-priori defined in (6.26)).

3. The last term for the E-step is the expected speech second-order statistic. Using the

$$\begin{aligned}\widehat{S}_m^{(\ell)}(t,k) \triangleq E\left\{S_m(t,k)|\boldsymbol{z}(t,k);\boldsymbol{\theta}^{(\ell-1)}\right\} = \\ E\left[E\left\{S_m(t,k)|x(t,k,m),\boldsymbol{z}(t,k);\boldsymbol{\theta}^{(\ell-1)}\right\}|\boldsymbol{z}(t,k);\boldsymbol{\theta}^{(\ell-1)}\right] \\ = E\Big\{x(t,k,m)E\left\{S_m(t,k)|x(t,k,m)=1,\boldsymbol{z}(t,k);\boldsymbol{\theta}^{(\ell-1)}\right\}+ \\ (1-x(t,k,m))E\left\{S_m(t,k)|x(t,k,m)=0,\boldsymbol{z}(t,k);\boldsymbol{\theta}^{(\ell-1)}\right\}|\boldsymbol{z}(t,k);\boldsymbol{\theta}^{(\ell-1)}\Big\} \\ = \widehat{x}^{(\ell)}(t,k,m)\cdot\phi_{S,m}^{(\ell-1)}(t,k)\left(\boldsymbol{g}_m^{(\ell-1)}(k)\right)^{\mathrm{H}}\left(\boldsymbol{\Phi}_m^{(\ell-1)}(t,k)\right)^{-1}\boldsymbol{z}(t,k). \quad (6.31)\end{aligned}$$

law of total expectation, the expected speech second-order statistic is given by (6.32). Note that, when the (t,k) bin is associated with the mth speaker, the expected speech second-order statistic is the squared MWF plus the MWF covariance error term (see [224, Eq.(32)]); otherwise, the expected speech second-order statistic is simply the a-priori variance $\phi_{S,m}^{(\ell-1)}(t,k)$ (the a-priori first-order statistic of the speech is zero).

$$\begin{aligned}\widehat{V}_m^{(\ell)}(t,k) \triangleq E\left\{|S_m(t,k)|^2|\boldsymbol{z}(t,k);\boldsymbol{\theta}^{(\ell-1)}\right\} = \widehat{x}^{(\ell)}(t,k,m) \\ \left[\left|\widehat{S}_m^{(\ell)}(t,k)\right|^2+\phi_{S,m}^{(\ell-1)}(t,k)-\left(\phi_{S,m}^{(\ell-1)}(t,k)\right)^2\left(\boldsymbol{g}_m^{(\ell-1)}(k)\right)^{\mathrm{H}}\left(\boldsymbol{\Phi}_m^{(\ell-1)}(t,k)\right)^{-1}\cdot\boldsymbol{g}_m^{(\ell-1)}(k)\right] \\ +\left(1-\widehat{x}^{(\ell)}(t,k,m)\right)\left[\phi_{S,m}^{(\ell-1)}(t,k)\right]. \quad (6.32)\end{aligned}$$

6.2.3 M-step

The second step of the iterative algorithm is maximization of (6.28) with respect to the unknown deterministic parameters $\boldsymbol{\theta}$ (namely, the M-step):

1. Similar to [16, Eq. (20a)], ψ_m is obtained by a constrained[1] maximization of (6.28):

$$\psi_m^{(\ell)} = \frac{\sum_{t,k}\widehat{x}^{(\ell)}(t,k,m)}{T\cdot K}. \quad (6.33)$$

The number of speakers and their locations are estimated from the soft values ψ_m only after the last iteration of the EM algorithm. The centers of the nodes are given by $\boldsymbol{p}_q^{(L)}$,

[1] The sum of ψ_m equals 1.

and the speakers' locations can be estimated by applying a threshold to $\psi_m^{(L)}$, where L is the number of iterations.

2. The variance of the speech is obtained by:

$$\phi_{S,m}^{(\ell)}(t,k) = \widehat{V}_m^{(\ell)}(t,k). \tag{6.34}$$

3. The array locations are obtained by the maximization:

$$\boldsymbol{p}_1^{(\ell)}...\boldsymbol{p}_Q^{(\ell)} = \text{argmax}_{\boldsymbol{p}_1...\boldsymbol{p}_Q} \sum_{t,k,m} \widehat{x}^{(\ell)}(t,k,m) \cdot$$
$$\sum_q \widehat{\log \mathcal{N}^c \left(\boldsymbol{z}_q(t,k) - \boldsymbol{g}_{q,m}(k) S_m(t,k); \boldsymbol{0}, \boldsymbol{\Phi}_{\boldsymbol{v}_q}(k) \right)}, \tag{6.35}$$

Because there is no closed-form solution for the array centers, a straightforward solution will require a tedious evaluation of the expression (6.35) in $|P|^Q$ points. Such a search is extremely complex; however, because of the summation in (6.35) (which was obtained by the assumption that the noise signals from different arrays are uncorrelated (6.17)), the search for each $\boldsymbol{p}_q^{(\ell)}$ can be carried out separately for each array:

$$\boldsymbol{p}_q^{(\ell)} = \text{argmax}_{\boldsymbol{p}_q} \sum_{t,k,m} \widehat{x}^{(\ell)}(t,k,m) \Big[2\text{Re} \left\{ \boldsymbol{z}_q^H(t,k) \boldsymbol{\Phi}_{\boldsymbol{v}_q}^{-1}(k) \boldsymbol{g}_{q,m}(k) \widehat{S}_m^{(\ell)}(t,k) \right\} -$$
$$\left(\boldsymbol{g}_{q,m}(k) \right)^H \boldsymbol{\Phi}_{\boldsymbol{v}_q}^{-1}(k) \boldsymbol{g}_{q,m}(k) \widehat{V}_m^{(\ell)}(t,k) \Big]. \tag{6.36}$$

Because the search is carried out for each array separately, it requires $|P| \cdot Q$ calculations of the likelihood term in (6.35) for $|P|$ grid points in the room, resulting in a significant save in calculations. Note that $\boldsymbol{p}_q$ controls $\boldsymbol{g}_{q,m}(k)$, as evident from (6.10)- (6.12).

6.2.4 The LACES algorithm: summary

It is well known that the classical batch EM algorithm is sensitive to initialization and might converge to a local maximum instead of the global maximum likelihood [47]. Several solutions have been suggested [225] to circumvent the mis-convergence phenomenon; these include incremental [226], sparse [225], recursive [227], and other variants of the batch EM algorithm. Experimentally, it was shown that the proposed algorithm might suffer from this mis-convergence if a conventional initialization is applied.

In addition, because all locations of the microphones and the speakers in our model are unknown, a spatial origin of the system should be predefined. We decided to use one of the arrays as the origin (referred to as the **anchor** node), and all of the microphones/speakers are measured w.r.t. that node. This means that the EM has to search $Q-1$ array locations. We found experimentally that an incremental estimation of the nodes' locations, rather than all of them together, enables convergence to the appropriate ML parameters.

In the beginning, only the anchor node is used by the algorithm for an initial estimation of ψ_m and $\phi_{S,m}(t,k)$. The nodes are added one at a time until all Q nodes used by the ad hoc network are included. We denote the current number of nodes used by $\tilde{Q}$. After each node addition, a higher dimension EM problem is solved with L iterations. The measurements captured by each node are used to estimate the position of the nodes and the probabilities of speakers occupying all locations.

For better convergence, we execute the M-step before the E-step. This means that, after adding a new node, the M-step is applied based on the previous E-step, which was based on $\tilde{Q}-1$ nodes. The last operation of the M-step is to find the location of all current nodes, including the newly added node (6.36). Only then we run the E-step. This algorithm is summarized in 9.

6.3 Performance Evaluation

The algorithm was evaluated using both simulations and real recordings. The performance of the proposed algorithm was evaluated in terms of both node calibration and concurrent speakers' localization. It should be emphasized that this challenge has not been dealt within the same kind of setup, which makes it difficult to compare the results to state-of-the-art algorithms.

The simulation and recording setups are described in the first subsection. The second subsection summarizes the measures used to evaluate the performance. The simulation results are given in the third subsection, and the last subsection addresses the analysis of real recordings.

Algorithm 9 LACES algorithm for noisy environment

```
Initialize
$\psi_m^{(0)} = \frac{1}{M}$
$\phi_{S,m}^{(0)}(t,k) = |\boldsymbol{z}(t,k)|^2$
$\boldsymbol{p} = \boldsymbol{p}_1$
for $\ell = 1$ to $L$ do
    E-step ($\tilde{Q} = 1$)
    Estimate $\widehat{x}^{(\ell)}(t,k,m)$ using (6.29), $\widehat{S}_m^{(\ell)}(t,k)$ using (6.31) and $\widehat{V}_m^{(\ell)}(t,k)$ using (6.32).
    M-step ($\tilde{Q} = 1$)
    Calculate $\psi_m^{(\ell)}$ using (6.33) and $\phi_{S,m}^{(\ell)}(t,k)$ using (6.34) $\forall m$.
end
for $\tilde{Q} = 2$ to $Q$ do
    Add node center to $\boldsymbol{p}$: $\boldsymbol{p} \leftarrow \left[\boldsymbol{p}^T \ \boldsymbol{p}_{\tilde{Q}}^T\right]^T$.
    for $\ell = 1$ to $L$ do
        M-step
        Calculate $\psi_m^{(\ell)}$ using (6.33), $\phi_{S,m}^{(\ell)}(t,k)$ using (6.34) and $\boldsymbol{p}_q^{(\ell)}$ using (6.36) $\forall q = 2 \ldots \tilde{Q}$.
        E-step
        Calculate $\widehat{x}^{(\ell)}(t,k,m)$ using (6.29), $\widehat{S}_m^{(\ell)}(t,k)$ using (6.31) and $\widehat{V}_m^{(\ell)}(t,k)$ using (6.32).
    end
end
Find $\mathbf{p}_s$ $\forall s$ using a threshold for $\psi_m^{(L)}$.
```

6.3.1 Experimental setup

The setups used for the simulations and recordings were approximately similar: three to five microphone arrays were located randomly in the experiment room. Together, all of the nodes composed an ad hoc network of acoustic sensors. Most of the time, the nodes were rectangular with four microphones simulating smart-phones with known dimensions and orientations. An example of such an array is shown in Fig. 6.1.

Some of the simulations were run with six microphones per node to enhance the localization capability.

The sampling frequency was 8 kHz and the frame length of the STFT was 64 msec with an overlap of 75 %. The number of frequency bins was 512. Utterances approximately 1 sec long of males and females were used. The speakers were located randomly around the table. The number of speakers was six for the simulations, and six for the real recordings.

The frequency band that was proven sufficient for our array sizes was $500 - 2000$ Hz. In the simulations, the speech signals were convolved by a simple RIRs of an anechoic chamber.

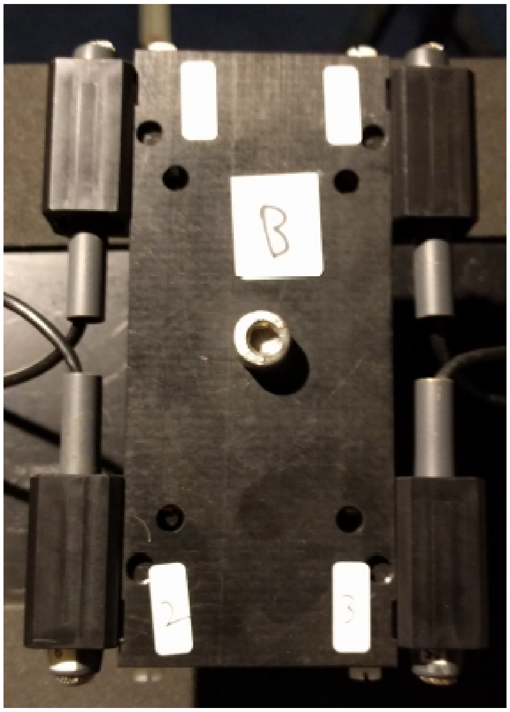


Figure 6.1: Rectangular array with four omni-directional AGG CK microphones at the corners

In the recordings, we recorded the signals in our acoustic room with a very low reverberation level ($T_{60} = 120$ msec). In both cases, a synthetic additive white Gaussian noise (AWGN) was added with various SNR levels.

A picture of the recordings setup can be found in Fig. 6.2. The rectangular arrays mentioned above were used together with Fostex model 6301B3X loudspeakers in the acoustic room. A high-quality recording system was used to measure the T_{60} and to generate the input signals for the new algorithm.

Although the full size of the room was $6 \times 6 \times 2.4$ m, here we focus on a smaller search area of 5×5 m with a constant height of 135 cm.

6.3.2 Performance measures

Calibration SR was calculated for Monte Carlo simulations according to the number of times the estimation of the node center was sufficiently accurate (up to 20 cm):

$$SR(\%) = 100 * S_c/A_e, \tag{6.37}$$

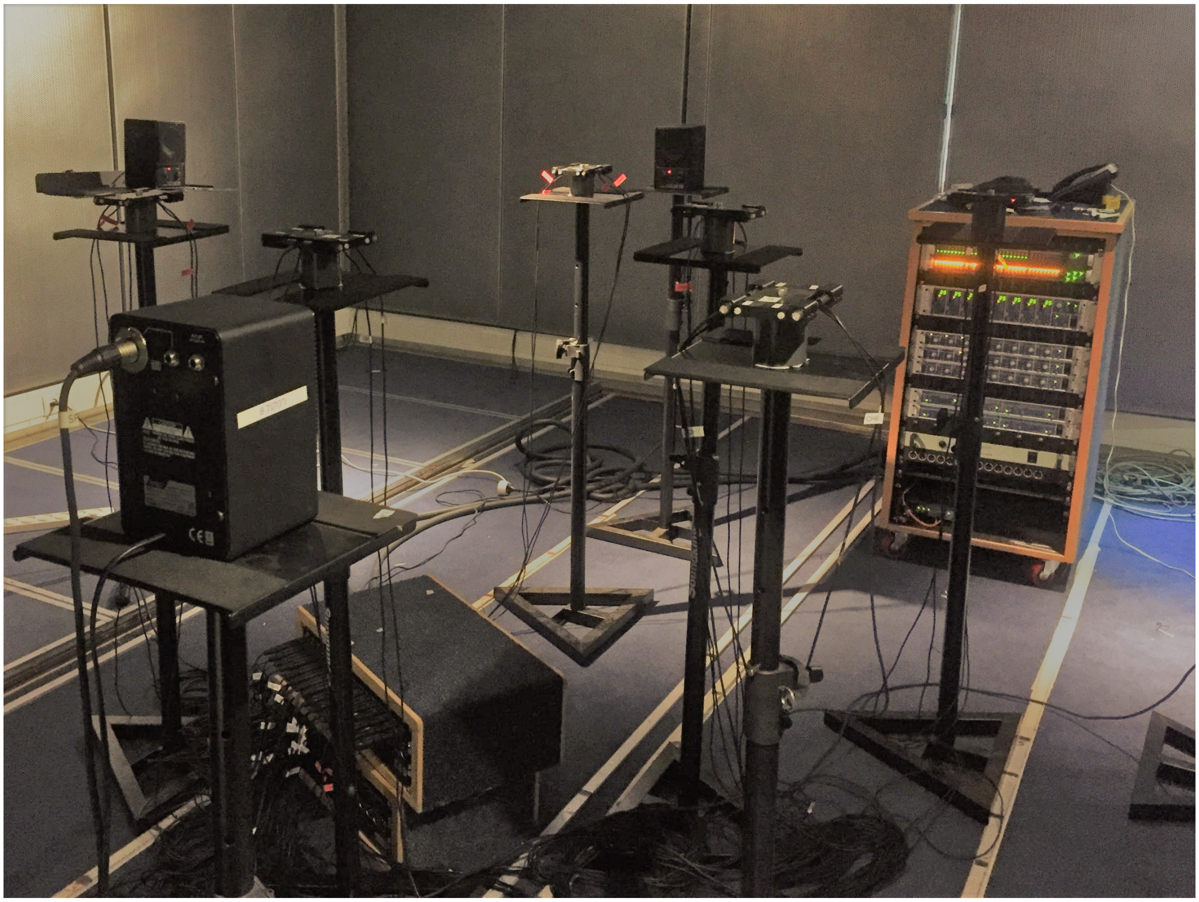


Figure 6.2: Room setup example: loudspeakers, microphone arrays, and recording equipment

where S_c is the number of successful calibrations and A_e is the total number of nodes to be calibrated. This is the only measure used for the calibration part, because this is the nature of this part of the algorithm. If the calibration is good, the accuracy is very good; if it fails, the results are useless and it also ruins the localization part.

For the localization part, we adapted three statistical measures used in [192, 193]. They are all calculated only for the cases of successful calibration.

The MD are counted according to the percentage of mis-detected speakers:

$$MD(\%) = 100 * M_s/R_s, \tag{6.38}$$

where M_s is the number of mis-detected sources and R_s is the total number of real sources.

The FA is the percentage of wrongly detected speakers:

$$FA(\%) = 100 * F_s/R_s, \tag{6.39}$$

Sensor SNR (dB)	Calib. SR (%)	MD (%)	FA (%)	Loc. RMSE(m)
0	45.5	54	-	-
10	70.5	16	1	0.16
20	71.5	6	1	0.16
40	74	6	1	0.16
60	74.5	6	1	0.16

Table 6.1: Statistical measures for various SNR levels. The node calibration SR is measured in percentage(%). The source localization performance measures are MD percentage, FAs percentage, and RMSE in meters.

where F_s is the number of falsely detected sources.

Localization RMSE is a measure of the estimation accuracy of all detected speakers:

$$\mathrm{RMSE} = \sqrt{\frac{1}{R_s - M_s} \sum_{s=1}^{R_s - M_s} e^2(s)}, \tag{6.40}$$

where s is the source index and $e(s)$ is its localization error, in meters.

6.3.3 Simulations of random geometrical setups

The geometric setup for the simulations is shown in Fig. 6.3. Three nodes with a square shape (10 × 10 cm) were randomly located with a random orientation in the middle of the room (each microphone is denoted by 'o'). Six speakers (denoted by the '+' sign) were located away from the room center, to mimic a scenario with nodes in the center (on a table) and speakers around that table. The main purpose of the simulation was to explore the performance for random geometrical setups.

The performance of the algorithm was tested for various levels of SNR and various sensor and source locations. The number of different setups generated was 100.

We noticed that a single EM iteration per new node ($L = 1$) yields satisfactory results. The statistical measures for the simulation study are summarized in Table 6.1. In presence of white sensor noise, as also demonstrated for the real recordings, the algorithm performance transits fast from good results for SNRs of 20 dB to very bad ones around SNRs of 0 dB. Note that the localization search grid is 0.2 m ×0.2 m, which means that the localization error is within the grid resolution.

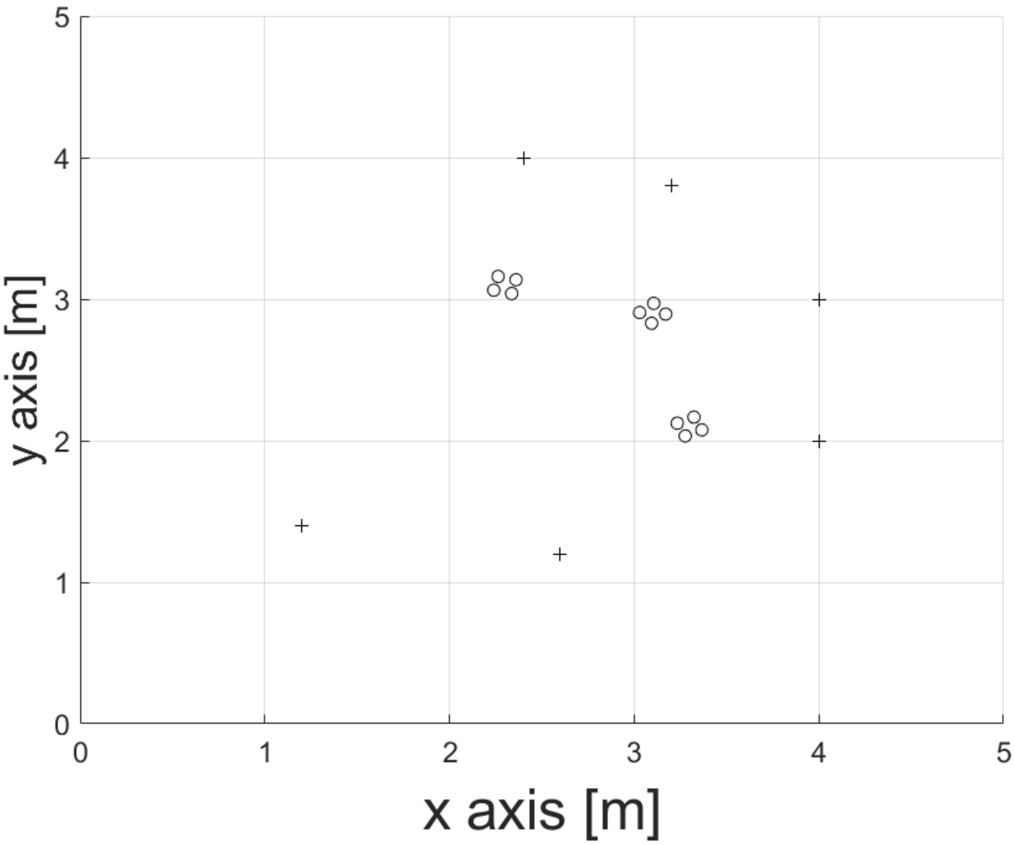


Figure 6.3: Simulation room random setup example. Each microphone is denoted by the sign o. Six speakers are denoted by the sign $+$.

To experimentally examine the localization and calibration EM sequence (LACES) convergence when arrays are added to the estimation, we plotted the intermediate results for the localization parameters, $\boldsymbol{\psi}$ in Fig. 6.4 for $L = 1$. The real locations of the six speakers are marked by '+'.

The improvement of the localization maps can be observed when additional arrays are used. For a single array, only some of the speakers are detected and a lot of noise is present. As arrays are added, the estimation improves for all speakers. The final map can be used to infer the number of speakers and their locations.

6.3.4 Recording results in noisy environment

The geometric setup for the real recordings is shown in Fig. 6.5. Three arrays with a rectangular shape (8.2 × 14.7 cm) were located in the middle of the room (each microphone is denoted by the symbol 'o'). Six speakers (denoted by the symbol '+') were located around

SNR(dB)	Calib. SR	MDs	FAs	Loc. RMSE(m)
>=14	2/2	0/6	0	0.1
10	2/2	1/6	0	0.1
<=3	0/2	N/A	N/A	N/A

Table 6.2: Measures for room recordings in various SNR conditions. The node calibration SR is given as a ratio. The source localization performance measures are MD ratio, FA ratio and RMSE in meters.

the center of the room in a meeting room setup.

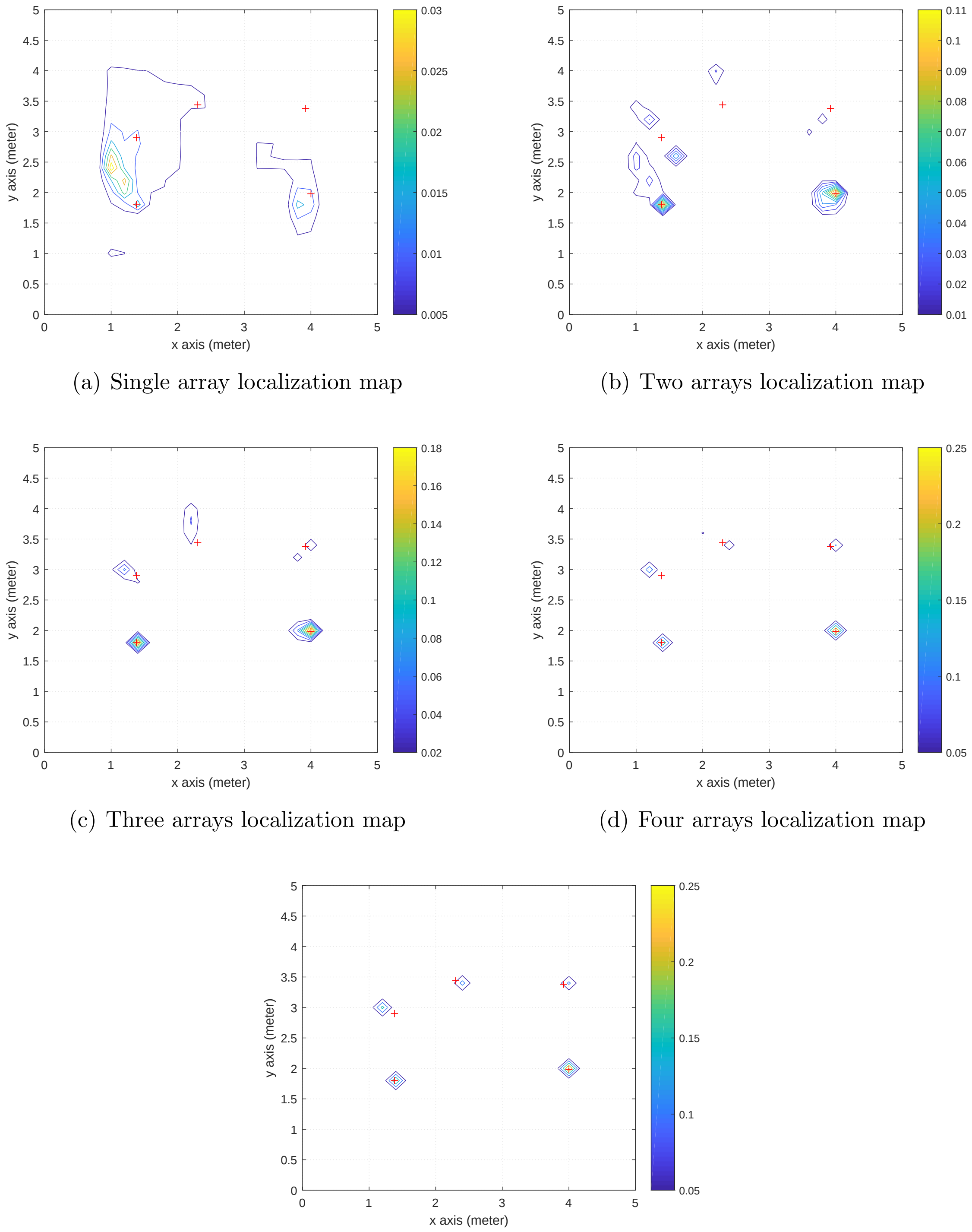
The real recordings were tested for low reverberation level ($T_{60} = 120$ msec) and various levels of sensor AWGN. The analysis of the real recordings is focused on the influence of the SNR level on the calibration and localization results. Table 6.2 summarizes the results for various SNR conditions.

It can be seen that, for any SNR higher than 14 dB, the performance is very good: the calibration was good for the nodes, the number of MDs was zero, there were no FAs, and the localization RMSE was 0.1 m. For an SNR of 10 dB, there is some degradation in the localization results, but the calibration is still good. The algorithm fails for all SNR levels equal to or below 3 dB.

6.4 Conclusions

A major challenge for ad hoc networks is to jointly localize sources and calibrate the positions of the arrays (or nodes) of the network. A novel joint calibration and localization algorithm, suitable for noisy environment was derived using the EM algorithm. One of the nodes is used as an anchor node, and the calibration (that is, the nodes' positions) is applied relative to the position of this anchor node.

To circumvent the initialization challenge of the batch EM, an incremental process was suggested that adds the nodes one after the other, instead of trying to solve the full dimension problem from the beginning of the EM algorithm. The new process, called the LACES algorithm, has been studied using simulations and real recordings.



(a) Single array localization map

(b) Two arrays localization map

(c) Three arrays localization map

(d) Four arrays localization map

(e) Five arrays localization map

Figure 6.4: Localization soft maps intermediate results (a), (b), (c), (d) and (e). The real locations of the simulated speakers is marked by '+'. The estimation is given by colored contours. The grid resolution is 20 cm. We excluded strips of 100 cm near the walls from the search area.

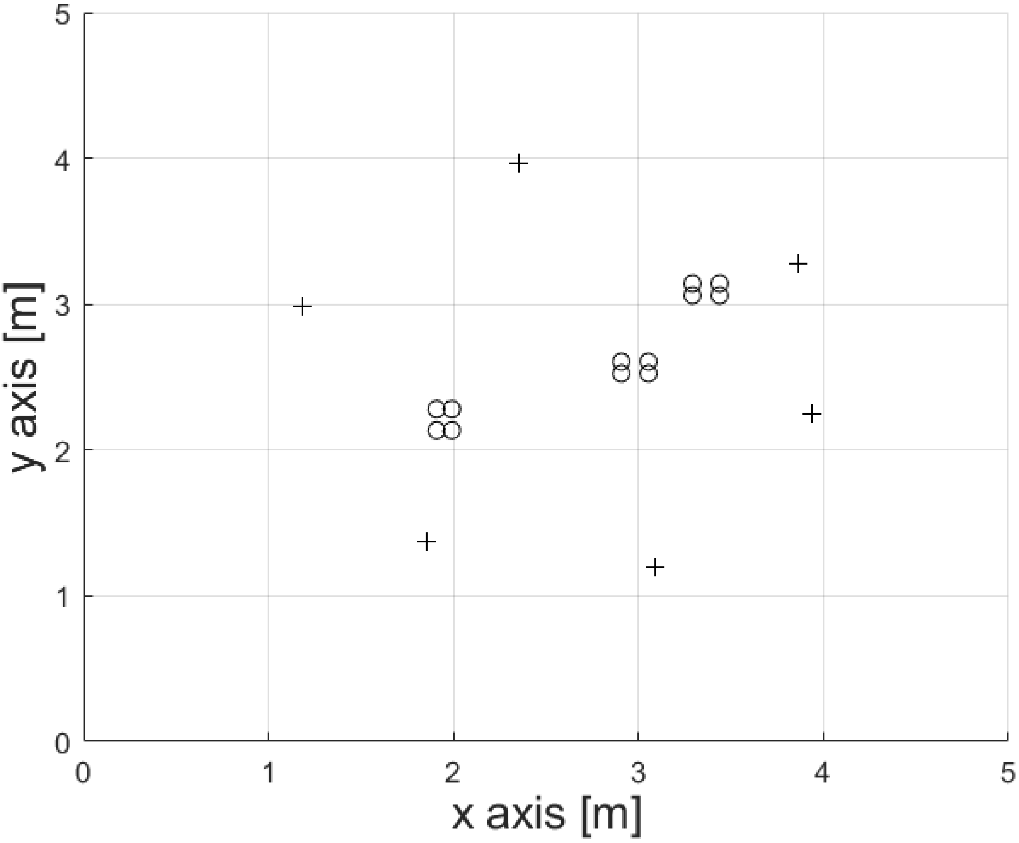


Figure 6.5: Recording room setup. Each microphone is denoted by the sign *o*. Six speakers are denoted by the sign +.

Chapter 7

Tracking using static and dynamic arrays

The material presented in this chapter is based on [200] and [201]:

[200] Y. Dorfan, B. Schwartz, and S. Gannot, "Speaker tracking using forward-backward recursive expectation-maximization," *To be submitted to IEEE...*

[201] C. Evers, Y. Dorfan, S. Gannot, and P. A. Naylor, "Source tracking using moving microphone arrays for robot audition," *IEEE International Conference on Audio and Acoustic Signal Processing (ICASSP), New-Orleans, LA, USA*, 2017

During the previous chapters we were focused on static speakers. The static assumption enables to apply signal processing tools that assume a constant RIR. In reality even minor movements might change the RIR quite significantly and hence other techniques should be applied. This chapter describes two types of tracking algorithms.

The first one is based on the RDEM presented in Chapter 3. In order to enhance the tracking capabilities a non-causal RDEM is applied. The results of the causal and the anti-causal are utilized together for the new tracking algorithm.

The second type addresses a different dynamic scenario. This time not only the sources are moving, but also the arrays of microphones. The algorithm applied is based on particle filtering.

7.1 Forward backward

In this section we address the task of localizing multiple concurrent moving speakers in reverberant conditions with static microphone arrays. Since both past and future observations contribute to the current location estimate, we propose a forward-backward approach that improves localization accuracy by allowing an additional short latency. We apply a non-Bayesian approach, which does not make assumptions regarding the target trajectories, except for assuming a relatively slow change in the parameters. The proposed method is based on the recursive distributed expectation maximization RDEM approach presented previously. It should be emphasized that this approach is different from the majority of tracking (Bayesian) algorithms. Those algorithms assume something about the dynamics of each source and track it separately. Our approach creates a map of the all area and the and the trajectories are drawn later. We assume nothing about each individual source dynamics.

The proposed algorithm is called forward backward recursive distributed expectation maximization FB-RDEM, and its performance is demonstrated using an extensive experimental study. The tested scenarios involve both simulated and recorded signals, with typical reverberation levels and multiple moving sources. It is shown that the proposed algorithm outperforms the regular (causal) RDEM, and that it is tolerant to short speech pauses. The new algorithm has improved accuracy, while adding a configurable latency.

7.1.1 Forward and backward recursive scheme

In this section, we first survey the criterion and then the derivation of the TREM algorithm [54]. The last part presents the main topic of this section, the forward-backward REM scheme.

7.1.1.1 Criterion discussion

Consider the problem of parameter estimation in the general case, where we observe the i.i.d. random time dependent variables $\boldsymbol{\tau}(t)$. The p.d.f. of the observations is denoted by $h(\boldsymbol{\tau}(t)) = h(\boldsymbol{\tau})$, which may be unknown.

A parametric model is attributed to the problem, inducing the parametric p.d.f. denoted by $f(\boldsymbol{\tau};\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is the vector of unknown parameters. We say that a parametric model is *specified*, if the real p.d.f. equals the parametric p.d.f. for a specific value of the parameter, i.e., there exists $\bar{\boldsymbol{\theta}}^*$ such that

$$h(\boldsymbol{\tau}) = f(\boldsymbol{\tau}; \bar{\boldsymbol{\theta}}^*). \tag{7.1}$$

Here, however, it is not required that the model will be *specified*, and we seek the value of $\boldsymbol{\theta}$ by which the parametric p.d.f. best fits the real p.d.f.. A common criterion for this fitness is the minimization of the KLD defined by

$$k(\boldsymbol{\theta}) = \mathbb{E}\{\log h(\boldsymbol{\tau}) - \log f(\boldsymbol{\tau}; \boldsymbol{\theta})\} , \tag{7.2}$$

where $\mathbb{E}\{\cdot\}$ denotes the expectation taken w.r.t. the true p.d.f., $h(\boldsymbol{\tau})$.

In this case, the minimization criterion is written as

$$\boldsymbol{\theta}^* = \operatorname*{argmin}_{\boldsymbol{\theta}} k(\boldsymbol{\theta}) = \operatorname*{argmax}_{\boldsymbol{\theta}}\{\mathbb{E}\{\log f(\boldsymbol{\tau}; \boldsymbol{\theta})\}\} . \tag{7.3}$$

It is noteworthy that $\boldsymbol{\theta}^*$ and the ML (batch) estimation of $\boldsymbol{\theta}$ are asymptotically equivalent; i.e. they converge for growing number of observations. The KLD criterion is common in online procedures, since it optimizes the p.d.f. of a single observation by updating the parameter with every new observed sample, rather than optimizing the p.d.f. of a specified set of observations using multiple iterations. This approach also fits the dynamic case, where the parameters are time-varying.

7.1.1.2 Recursive EM

In many problems, hidden data may be used, for either computational reasons, e.g., when the likelihood is intractable, or physical reasons, e.g., when the hidden data represents a realistic and desired physical value. In Subsection 7.1.3, the hidden data is an indicator that associates a time-frequency band bin of an acoustic signal to a specific speaker; a physical value of interest. We henceforth denote the hidden data as $\mathbf{y}(t)$ and further assume that there exists a joint p.d.f. of $\boldsymbol{\tau}$ and $\mathbf{y}$, namely, the *complete* p.d.f., denoted by $f(\boldsymbol{\tau}, \mathbf{y}; \boldsymbol{\theta})$. As discussed in the first chapter of this thesis, the EM is a common approach for maximizing the likelihood in hidden-data problems, to which a recursive version was presented by Titterington [54].

The TREM algorithm [54] maximizes the KLD using the recursion

$$\widehat{\boldsymbol{\theta}}_{t+1}^{\mathrm{Ti}} = \widehat{\boldsymbol{\theta}}_{t}^{\mathrm{Ti}} + \tfrac{1}{t} \cdot \mathbf{I}_C^{-1}(\widehat{\boldsymbol{\theta}}_{t}^{\mathrm{Ti}}) \cdot \boldsymbol{s}(\boldsymbol{\tau}_t; \widehat{\boldsymbol{\theta}}_{t}^{\mathrm{Ti}}) , \tag{7.4}$$

where $\widehat{\boldsymbol{\theta}}_t^{\mathrm{Ti}}$ is the previous estimate, $\boldsymbol{s}(\boldsymbol{\tau}_t;\widehat{\boldsymbol{\theta}}_t^{\mathrm{Ti}})$ denotes the *scoring* function defined by the gradient,

$$\boldsymbol{s}(\boldsymbol{\tau}_t;\widehat{\boldsymbol{\theta}}_t^{\mathrm{Ti}}) = \nabla_{\boldsymbol{\theta}} \log f(\boldsymbol{\tau};\boldsymbol{\theta})\big|_{\boldsymbol{\tau}=\boldsymbol{\tau}_t,\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_t^{\mathrm{Ti}}} \ , \tag{7.5}$$

and $\mathbf{I}_C(\boldsymbol{\theta}_t)$ denotes the complete-data FIM, i.e.,

$$\mathbf{I}_C(\widehat{\boldsymbol{\theta}}_t^{\mathrm{Ti}}) = -\mathbb{E}_C\{\nabla^2_{\boldsymbol{\theta}} \log f(\boldsymbol{\tau},\mathbf{y};\boldsymbol{\theta})\}\big|_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_t^{\mathrm{Ti}}} \ , \tag{7.6}$$

where $\mathbb{E}_C$ denotes the expectation taken w.r.t. the complete-data p.d.f $f(\boldsymbol{\tau},\mathbf{y};\boldsymbol{\theta})$. Note that by the *Fisher identity* (see Sec.15 of [228]), $\boldsymbol{s}(\boldsymbol{\tau};\boldsymbol{\theta})$ can be calculated via the complete p.d.f. $f(\boldsymbol{\tau},\mathbf{y};\boldsymbol{\theta})$,

$$\boldsymbol{s}(\boldsymbol{\tau};\boldsymbol{\theta}) = \mathbb{E}_C\left\{\nabla_{\boldsymbol{\theta}} \log f(\boldsymbol{\tau},\mathbf{y};\boldsymbol{\theta})|\boldsymbol{\tau};\boldsymbol{\theta}\right\} \ , \tag{7.7}$$

thus recursion (7.4) does not require an explicit form of $f(\boldsymbol{\tau};\boldsymbol{\theta})$, which is the very reason for choosing the hidden data approach. It was shown in [180] that under certain regularity conditions, the TREM converges w.p.1. to a stationary point of the KLD.

Re-examining (7.4), it can be seen that the update term in time t is multiplied by t^{-1}, and hence the parameter update is less significant with time, which is a crucial requirement for the strong (w.p.1.) convergence. In practice, it is common to substitute t^{-1} by a constant smoothing coefficient, allowing tracking abilities for dynamic cases. Define a variant of the TREM algorithm that uses a constant smoothing factor,

$$\widehat{\boldsymbol{\theta}}_{t+1}^{\mathrm{F}} = \gamma_F \cdot \widehat{\boldsymbol{\theta}}_t^{\mathrm{F}} + (1-\gamma_F)\cdot \mathbf{I}_C^{-1}(\widehat{\boldsymbol{\theta}}_t^{\mathrm{F}}) \cdot \boldsymbol{s}(\boldsymbol{\tau}_t;\widehat{\boldsymbol{\theta}}_t^{\mathrm{F}}) \ , \tag{7.8}$$

where $0 \leq \gamma_F \leq 1$ is predefined to fit the dynamics of the problem. This procedure also converges to a stationary point of the KLD, but convergence in this case is weak (see Chapter 8 in [229]).

Using the update scheme in (7.8) is common in tracking and localization problems, when the parameter $\boldsymbol{\theta}$ is time-varying and strong convergence is not desired. A mathematical analysis of algorithms with structure similar to (7.8) is given in [230], using the definition of hyper-model; i.e., a model for the true parameter and for the algorithm dynamics.[1]

[1] Do not confuse the hyper model analysis with the Bayesian approach - this model is for the analysis only, and not used by or known to the algorithm.

The influence of γ_F on the algorithm's performance can be intuitively explained as follows. Tuning the value of γ_F enables a trade-off between the bias and variance of the algorithm. For higher values of γ_F, the update rate of the algorithm is slower, meaning that the estimation is biased by the past values of the parameter, but the estimation variance is lower, due to the longer averaging window. For lower γ_F values, the convergence speed is higher, which means a lower bias, but this comes at the expense of higher variance due to the shorter averaging window. The reader is referred to Chapter 4 in [230], which elaborates on the analysis of time-varying parameter estimation.

We now propose a method that improves the performance of (7.8) in the expense of higher latency. This is done by introducing future observations to the estimation procedure, and may reduce both the bias and the variance of the estimator.

7.1.1.3 The FB-RDEM Algorithm

The recursion (7.8) makes use of the past observations in order to estimate the parameter. In most cases, the near future is informative as well, and can improve the accuracy in the price of higher latency. We propose the Forward-Backward Recursive Expectation-Maximization (FB-REM) algorithm, defined by

$$\widehat{\boldsymbol{\theta}}^{\mathrm{FB}}_{t+1} = \alpha_{FB} \cdot \widehat{\boldsymbol{\theta}}^{\mathrm{F}}_{t+1} + (1 - \alpha_{FB}) \cdot \widehat{\boldsymbol{\theta}}^{\mathrm{B}}_{t+1} \ , \tag{7.9}$$

where $\widehat{\boldsymbol{\theta}}^{\mathrm{F}}_{t}$ was defined in (7.8), and $0 \leq \alpha_{FB} \leq 1$ is a weighting factor of the past and the future terms. The backward estimator $\widehat{\boldsymbol{\theta}}^{\mathrm{B}}_{t}$ is calculated by the backward recursion

$$\begin{aligned} \widehat{\boldsymbol{\theta}}^{\mathrm{B}}_{k} = \gamma_B \cdot \widehat{\boldsymbol{\theta}}^{\mathrm{B}}_{k+1} + (1 - \gamma_B) \cdot \mathbf{I}_C^{-1}(\widehat{\boldsymbol{\theta}}^{\mathrm{B}}_{k}) \cdot \boldsymbol{s}(\boldsymbol{\tau}_k; \widehat{\boldsymbol{\theta}}^{\mathrm{B}}_{k}) \ , \\ k = t + D, ..., t + 1 \ , \end{aligned} \tag{7.10}$$

where $0 \leq \gamma_B \leq 1$ is a smoothing factor, and the number of future samples D will be discussed in the following. Note that $\widehat{\boldsymbol{\theta}}^{\mathrm{F}}_{t+1}$ is calculated using the past observations $\boldsymbol{\tau}_1, ..., \boldsymbol{\tau}_t$, while $\widehat{\boldsymbol{\theta}}^{\mathrm{B}}_{t+1}$ is calculated by the future observations $\boldsymbol{\tau}_{t+D}, ..., \boldsymbol{\tau}_{t+2}, \boldsymbol{\tau}_{t+1}$, thus each observation is considered only once in (7.9). In the following, we will restrict the range of future summation in (7.9), leaving only a constant latency.

In a Bayesian framework, there would have been an optimal choice of the smoothing factors γ_F, γ_B, and α_{FB}, obtained by a Bayesian statistical model. This way the Kalman smoother [231] gain and the coefficient used in Viterbi algorithm [232] are determined. However, since we intentionally adopted a non-Bayesian approach, the values of γ_F, γ_B, and α_{FB} are determined due to the required dynamics of the algorithm, and the nature of the stochastic processes. We previously discussed how γ_F trades-off the update speed versus the accuracy of the algorithm. Similarly, high γ_B reduces the variance of the algorithm in the expense of higher bias. Finally, α_{FB} determines the weight of either the past and the future observations on the current estimation.

In practice, α_{FB} is mainly determined by the number of the future observations that are actually used to update the estimation, which in turn is determined by the latency constraints of the application. Rewriting (7.10) as

$$\widehat{\boldsymbol{\theta}}^{\mathrm{B}}_{t+1} = (1-\gamma_B) \cdot \sum_{k=t+1}^{t+D} \gamma_B^{k-t} \cdot \mathbf{I}_C^{-1}(\widehat{\boldsymbol{\theta}}^{\mathrm{B}}_k) \cdot \boldsymbol{s}(\boldsymbol{\tau}_k; \widehat{\boldsymbol{\theta}}^{\mathrm{B}}_k) \ , \tag{7.11}$$

it can be seen that for every value of γ_B and every computing precision requirement, there exists an integer D_{max} such that $\gamma_B^{D_{max}} \approx 0$. If $D = D_{max}$ is chosen, the backward recursion is equivalent to an infinite backward recursion, similarly to the forward recursion for large enough t values. However, in practical applications, using D_{max} future observations might introduce unacceptable latency, and a lower value of D should be chosen instead. This choice deteriorate the accuracy of the backward recursion (7.10), which should be compensated by increasing the value of α_{FB}.

In the next subsection, the algorithm (7.9) will be realized for the problem of multiple dynamic speakers localization.

7.1.2 Speakers tracking problem formulation

One of the major applications of the recursive schemes is tracking. We focus here on dynamic localization of concurrent speakers, which is very challenging due to non-stationary and non-continuous nature of the speech signals.

The problem is formulated in time-frequency band and in the spatial domain. Let $b = 1, \ldots, B$ be the frequency band index, S be the number of acoustic signals captured by M microphones, organized in $M/2$ independent pairs.

The signal received by the ith microphone, $i = 1, 2$ of the mth pair, $m = 1, \ldots, M/2$, is given by

$$z^i_m(t,b) = \sum_{s=1}^{S} a^i_{sm}(t,b) v_s(t,b) + n^i_m(t,b), \tag{7.12}$$

where $s = 1, \dots, S$ is the source index, $v_s(t,b)$ denotes the sth source signal, $n^i_m(t,b)$ denotes an additive noise, and $a^i_{sm}(t,b)$ denotes the time-variant RIR.

For each band the pair-wise TDOAs $\tau_{m,b}(t)$ are calculated using cross-correlation. Under the assumption of speech sparsity [110], each time-frequency band vector is dominated by the direct path of a single source (W-disjoint). This implies that the summation in (7.12) is dominated by a single source.

Let $\mathbf{p} = [x, y, z]$ be a position in the room described by the 3D Cartesian coordinates and let $\mathcal{P}$ be a set of all possible positions contributed by all speakers without assuming any a priori knowledge about their number or dynamics. Multiple locations may receive high values according to the number of active speakers.

The grid of possible positions is used, as in [193]. The noiseless TDOAs for each grid point can be calculated in advance:

$$\tilde{\tau}_{m,b}(\mathbf{p}) \triangleq \frac{||\mathbf{p} - \mathbf{p}^1_m|| - ||\mathbf{p} - \mathbf{p}^2_m||}{c} : \forall \mathbf{p} \in \mathcal{P}, \tag{7.13}$$

where $\mathbf{p}^1_m$ and $\mathbf{p}^2_m$ are the locations of the microphones assumed to be perfectly known, $||\cdot||$ denotes the Euclidean norm, and c is the sound velocity.

The statistical model applied to the TDOAs is MoG:

$$\tau_{m,b}(t) \sim \sum_{\mathbf{p}} \psi_{\mathbf{p}}(t) \mathcal{N}\left(\tau_{m,b}(t); \tilde{\tau}_{m,b}(\mathbf{p}), \sigma^2\right), \tag{7.14}$$

where σ^2 is the variance, which is assumed to be a known constant parameter.

The weights $\psi_{\mathbf{p}}(t)$ are the probability of a speaker to be in position $\mathbf{p}$ at time t will be estimated by the algorithm. The constraints of the optimization problem for any time stamp t are:

$$\sum_{\mathbf{p} \in \mathcal{P}} \psi_{\mathbf{p}}(t) = 1 \tag{7.15}$$

$$0 \leq \psi_{\mathbf{p}}(t) \leq 1.$$

The set of unknown parameters, $\boldsymbol{\theta}$ to be found by the EM in this MoG case contains only the Gaussian weights, $\psi_{\mathbf{p}}(t)$, since the mean are calculated in advance over the grid of positions and the variance is known.

The p.d.f. of all augmented measurements at each node is

$$f(\boldsymbol{T}_m(t) = \boldsymbol{\tau}_m(t); \boldsymbol{\psi}(t)) = \prod_b \sum_{\mathbf{p}\in\mathcal{P}} \psi_{\mathbf{p}}(t)\mathcal{N}\left(\tau_{m,b}(t); \tilde{\tau}_{m,b}(\mathbf{p}), \sigma^2\right). \tag{7.16}$$

As for every other EM and REM algorithms, a set of hidden variables is defined. The hidden used here is a set of indicators [193] that relates each time-frequency band bin to an active position from that grid of points. The physical meaning of those variables comes from the dis-joint property of speech. However, since those variables are not measured they are referred to as 'hidden'.

Following [193], we derive an RDEM algorithm with the hidden data, $y_m(t, b, \mathbf{p})$, defined as the indicator associated with node m that time-frequency band bin (t, b) belongs to a speaker in a certain position, $\mathbf{p} \in \mathcal{P}$. Intuitively, the dynamic localization challenge becomes much simpler to solve given this additional information.

The conditional observations p.d.f. is

$$\begin{aligned} f(\boldsymbol{T}(t) = \boldsymbol{\tau}(t)|\mathbf{Y}(t) = \mathbf{y}(t); \boldsymbol{\psi}(t)) &= \prod_m f(\boldsymbol{T}_m(t) = \boldsymbol{\tau}_m(t)|\mathbf{Y}_m(t) = \mathbf{y}_m(t); \boldsymbol{\psi}(t)) \\ &= \prod_{m,b} \sum_{\mathbf{p}\in\mathcal{P}} y_m\left(t, b, \mathbf{p}\right)\mathcal{N}\left(\tau_{m,b}(t); \tilde{\tau}_{m,b}(\mathbf{p}), \sigma^2\right), \end{aligned} \tag{7.17}$$

where

$$\mathbf{y}(t) = \mathrm{vec}_{m,b,\mathbf{p}}\left(y_m(t, b, \mathbf{p})\right). \tag{7.18}$$

The *complete data* p.d.f. is

$$f(\boldsymbol{T}(t) = \boldsymbol{\tau}(t), \mathbf{Y}(t) = \mathbf{y}(t); \boldsymbol{\psi}(t)) = \prod_{m,b} \sum_{\mathbf{p}\in\mathcal{P}} \psi_{\mathbf{p}}(t) y_m(t, b, \mathbf{p})\mathcal{N}\left(\tau_{m,b}(t); \tilde{\tau}_{m,b}(\mathbf{p}), \sigma^2\right). \tag{7.19}$$

In the next subsection a new algorithm is derived based on the forward-backward concept introduced in the previous section. For practical reasons we develop a distributed algorithm.

7.1.3 FB-RDEM for concurrent speakers tracking

Although the forward-backward concept can be applied to various centralized or distributed algorithms, we focus on a specific example: diffusion through a bi-directional tree. The advantages of this topology for this kind of problems has been described in details [193].

Following the bi-directional tree-based RDEM algorithm, we derive a forward-backward version. Two RDEM are applied in opposite directions (forward and backward). The forward runs from the previous reported time stamp till the current. The time step length is denoted as δt. The backward RDEM starts from the future samples in the anti-causal direction utilizing the processing delay. Both recursions run till the current time stamp t_c, and their linear combination (7.9) is calculated:

$$\psi_{\mathrm{FB},R}(t) = \alpha_{FB} \cdot \psi_{\mathrm{F},R}(t) + (1 - \alpha_{FB}) \cdot \psi_{\mathrm{B},R}(t), \tag{7.20}$$

where $\psi_{\mathrm{F},R}(t)$ and $\psi_{\mathrm{B},R}(t)$ are the results of the forward and backward RDEM, respectively. Those processes are derived in this section. Notice that using a trivial case $\alpha_{FB} = 1$, which means we have no relevant information from the future samples, results with a regular forward RDEM.

We start with the forward part of the algorithm, since the backward is similar in our case besides the processing direction. As mentioned above in subsection 7.1.1, recursive EM versions have been derived in [184, 54]. We adopted in (7.8) the TREM version [54].

According to (7.8)-(7.10) and[16, 193], we can derive the following recursive relation based on TREM for our tracking application:

$$\hat{\boldsymbol{\psi}}_{\mathrm{F},R}(t) = \hat{\boldsymbol{\psi}}_{\mathrm{F},R}(t-1) + \gamma_{\mathrm{F}} \left(\hat{\boldsymbol{\psi}}_{\mathrm{F}}(t) - \hat{\boldsymbol{\psi}}_{\mathrm{F},R}(t-1) \right), \tag{7.21}$$

where the current estimation of the parameters is written in a vectorial format:

$$\hat{\boldsymbol{\psi}}_{\mathrm{F}}(t) = \mathrm{vec}_{\mathbf{p}}(\hat{\psi}_{\mathrm{F},\mathbf{p}}(t)). \tag{7.22}$$

In [193] the first and second order derivatives of the log-likelihood were calculated for the case of forward recursion. Following those steps, the RDEM can be easily derived. The instantaneous global parameter estimation (before recursion) is the mean of all the hidden variables, calculated by

$$\hat{\psi}_{\mathrm{F},\mathbf{p}}(t) \triangleq \frac{1}{M/2}\sum_{m=1}^{M/2} \bar{\upsilon}_m^{(\mathrm{F})}(t,\mathbf{p}), \tag{7.23}$$

where the estimation of the local hidden average with respect to frequency bands is defined as:

$$\bar{\upsilon}_m^{(\mathrm{F})}(t,\mathbf{p}) \triangleq \frac{1}{B}\sum_{b=1}^{B} \upsilon_m^{(\mathrm{F})}(t,b,\mathbf{p}). \tag{7.24}$$

The estimation of the hidden data is given by

$$\upsilon_m^{(\mathrm{F})}(t,b,\mathbf{p}) \triangleq \frac{\hat{\psi}_{\mathrm{F},\mathbf{p}}(t-1)\mathcal{N}\left(\tau_{m,b}(t);\tilde{\tau}_{m,b}(\mathbf{p}),\sigma^2\right)}{\sum_{\tilde{\mathbf{p}}\in\mathcal{P}}\hat{\psi}_{\mathrm{F},\tilde{\mathbf{p}}}(t-1)\mathcal{N}\left(\tau_{m,b}(t);\tilde{\tau}_{m,b}(\tilde{\mathbf{p}}),\sigma^2\right))}. \tag{7.25}$$

This form is very common in the EM for MoG. It is the Gaussian for the specific grid position multiplied by the latest weight estimate and normalized by all other positions of the grid.

In the same way, the backward RDEM can be written replacing the notation F with B and the time index $(t-1)$ with $(t+1)$ in equations (7.21)- (7.25) above. In the backward version the time samples are processed in the anti-causal direction. As a result we get the following backward recursion equation:

$$\hat{\boldsymbol{\psi}}_{\mathrm{B},R}(t) = \hat{\boldsymbol{\psi}}_{\mathrm{B},R}(t+1) + \gamma_{\mathrm{B}}\left(\hat{\boldsymbol{\psi}}_{\mathrm{B}}(t) - \hat{\boldsymbol{\psi}}_{\mathrm{B},R}(t+1)\right), \tag{7.26}$$

where the current estimation of the parameters for the backward process is also written in a vectorial format:

$$\hat{\boldsymbol{\psi}}_{\mathrm{B}}(t) = \mathrm{vec}_{\mathbf{p}}(\hat{\psi}_{\mathrm{B},\mathbf{p}}(t)). \tag{7.27}$$

The instantaneous global parameter estimation is again the mean of all the hidden variables, calculated by

$$\hat{\psi}_{\mathrm{B},\mathbf{p}}(t) \triangleq \frac{1}{M/2}\sum_{m=1}^{M/2} \bar{\upsilon}_m^{(\mathrm{B})}(t,\mathbf{p}), \tag{7.28}$$

where the estimation of the local hidden average with respect to frequency bands is defined

Algorithm 10 Acoustic source localization with FB-RDEM.

```
set $\tilde{\tau}_{m,b}(\mathbf{p})$
initialize $\hat{\boldsymbol{\psi}}_{\mathrm{F},R}(0) = 1/|\mathcal{P}|$
Let $t_c = \delta t$ denote the current time for location report
while $t_c < t_{stop}$; $tc = tc + \delta t$ do
    for $t = t_c - \delta t$ to $t_c$ do
        Calculate simultaneously and locally $\bar{\upsilon}_m^{(\mathrm{F})}(t, \mathbf{p})$ using (7.24) $\forall m = 1, \ldots, M/2$
        Aggregate results and calculate $\hat{\boldsymbol{\psi}}_{\mathrm{F},R}(t)$
    end
    initialize $\hat{\boldsymbol{\psi}}_{\mathrm{B},R}(t_c + P_d + 1) = 1/|\mathcal{P}|$
    for $t = t_c + P_d$ to $t_c$ do
        Calculate simultaneously and locally $\bar{\upsilon}_m^{(\mathrm{B})}(t, \mathbf{p})$ using (7.29) $\forall m = 1, \ldots, M/2$
        Aggregate results and calculate $\hat{\boldsymbol{\psi}}_{\mathrm{B},R}(t)$
    end
    Calculate $\hat{\boldsymbol{\psi}}_{\mathrm{FB},R}(t_c)$ according to (7.20)
    Find $\mathbf{p_s}(\mathbf{t_c})$ by applying a threshold to $\hat{\boldsymbol{\psi}}_{\mathrm{FB},R}(t_c)$
    Set $\hat{\boldsymbol{\psi}}_{\mathrm{F},R}(t_c) = \hat{\boldsymbol{\psi}}_{\mathrm{FB},R}(t_c)$
end
```

as:

$$\bar{\upsilon}_m^{(\mathrm{B})}(t, \mathbf{p}) \triangleq \frac{1}{B} \sum_{b=1}^{B} \upsilon_m^{(\mathrm{B})}(t, b, \mathbf{p}). \tag{7.29}$$

The estimation of the hidden data is given by

$$\upsilon_m^{(\mathrm{B})}(t, b, \mathbf{p}) \triangleq \frac{\hat{\psi}_{\mathrm{B},\mathbf{p}}(t-1)\mathcal{N}\left(\tau_{m,b}(t); \tilde{\tau}_{m,b}(\mathbf{p}), \sigma^2\right)}{\sum_{\tilde{\mathbf{p}} \in \mathcal{P}} \hat{\psi}_{\mathrm{B},\tilde{\mathbf{p}}}(t-1)\mathcal{N}\left(\tau_{m,b}(t); \tilde{\tau}_{m,b}(\tilde{\mathbf{p}}), \sigma^2\right)}. \tag{7.30}$$

Due to this anti-causal process, we have to choose a processing delay, denoted as P_d. The algorithm estimates locations along time using relevant past and future samples according to the processing delay P_d and the speakers dynamics.

As for many EM algorithms, initialization is a crucial part. We adopt here quite a simple method. The forward RDEM is initialized according to the previous estimations. The backward RDEM is uniformly initialized in space. We execute the algorithm until the stop time, denoted by t_{stop}. After estimation of $\hat{\boldsymbol{\psi}}_{\mathrm{FB},R}(t_c)$ at each time step, we apply a threshold in order to find the number of speakers and their current locations. As stated above, this is a dynamic localization algorithm and not a classical tracker. The entire FB-RDEM procedure is summarized in Algorithm 10.

7.1.4 Experimental Study

This subsection compares the proposed FB-RDEM algorithm to the RDEM [193] algorithm. It should be emphasized that tracking in acoustic signal processing is usually developed for $1-D$ only. In addition, we deal with a case for which unlike classical tracking algorithms, no assumptions is made regarding the dynamics of the speakers. For those reasons we found it difficult to compare our results to other state-of-the-art algorithms.

The study contains four parts. We start with a parameters discussion that should be tuned for both algorithms. The second part describes the room setup for simulation and recordings. The third part is about the random Monte-Carlo simulations. It enables a rich comparison between the two algorithms. The last part compares them for real recordings that we generated in our acoustic room in order to ensure both algorithms can track real speakers in a real dynamic environment.

7.1.4.1 Parameters choice

There are a few important parameters mentioned above that should be chosen to enable sufficient performance of the localization algorithm.

The first one is the TREM smoothing factors γ_F and γ_B. The γ_F parameter has been already tuned experimentally for the regular RDEM [16]. We tuned both values empirically for our model.

The next parameter to be discussed is the past weighting coefficient, α_{FB}. For offline applications we might expect the past and the future samples to be equally weighted, meaning $\alpha_{FB} = 0.5$, since the correlation is symmetric and decades with time in both directions. Since we deal with online applications, our algorithm is not symmetric. In addition, the number of future samples depends on the latency chosen. For those reasons we had to tune this parameter manually.

Important parameters that have close relations with the smoothing factors are the latency time samples, D and the processing delay P_d. Assuming slow dynamics and the acoustic volume of a real speaker, it is reasonable to work with around 1 sec for both in many applications that require speaker localization within the room. It is obvious that for online applications, like we deal with, this trade-off might be dealt differently. For example, fast dynamics tuning is easy, since the window length of relevant data is short. The most important samples are the closest in time in both directions of the time axis. It means that the idea of non-causal localization can be useful even for much lower latency values.

The number of frequency bands, $B = 16$ was also examined experimentally within the range that is reported in other sub band approaches. For example, algorithms that compute sub band TDOAs in order to improve the robustness and to facilitate concurrent speaker localization [80, 88, 89, 195].

The MoG variance, σ^2 was tuned manually to $2[Samp^2]$. Sampling frequency is 16 KHz. We assume that its value is uniform over time, frequency band and all other relevant variables. Future research can be done about this parameter, which sometimes appear to have a significant influence on the performance of the localization algorithm.

Another important parameter is the resolution of the positions grid. Here we adapted a resolution of 0.10×0.10 m^2, which has been shown [16] to be fine enough for localization of real speakers that have a significant acoustic volume and therefore are not emitting signals like point sources used in the simulations. We observed that for both simulated and real signals this resolution is fine enough. Increasing the resolution further increases the complexity significantly.

Frame length of the FFT, 64 msec was adapted from previous localization algorithms [16, 192, 197, 193, 196, 194, 198].

7.1.4.2 Room Setup

To evaluate the localization performance of the algorithms, we simulated the following scenario. Twelve pairs of microphones were positioned in the room (encompassing the acoustic scene). A map of the microphones locations is shown in Fig. 7.1. The positions of the microphones are marked by 'o' on a two dimensional plane. Each pair belongs to a different node. The nodes are numbered from 1 to 12 in an arbitrary order. The scenario examined is a 2D with all speakers and microphones in the same height for simplicity, but the algorithm can be easily expanded to 3D.

The number of microphones pairs, $M/2$ was already examined in [16]. Here 12 pairs are used for the simulations as in [16]. The number of microphone pairs (or nodes) is very important in the presence of high noise and/or high reverberation levels.

The dimensions of the simulated room were $6 \times 6 \times 2.4$ m. One or two sources randomly moved along a trajectory in the room. Theses dynamics cases were simulated using short speech files and an efficient implementation [210] of the image method [211].

The laboratory recording setup is based on the description of [16, 193]. An example of a

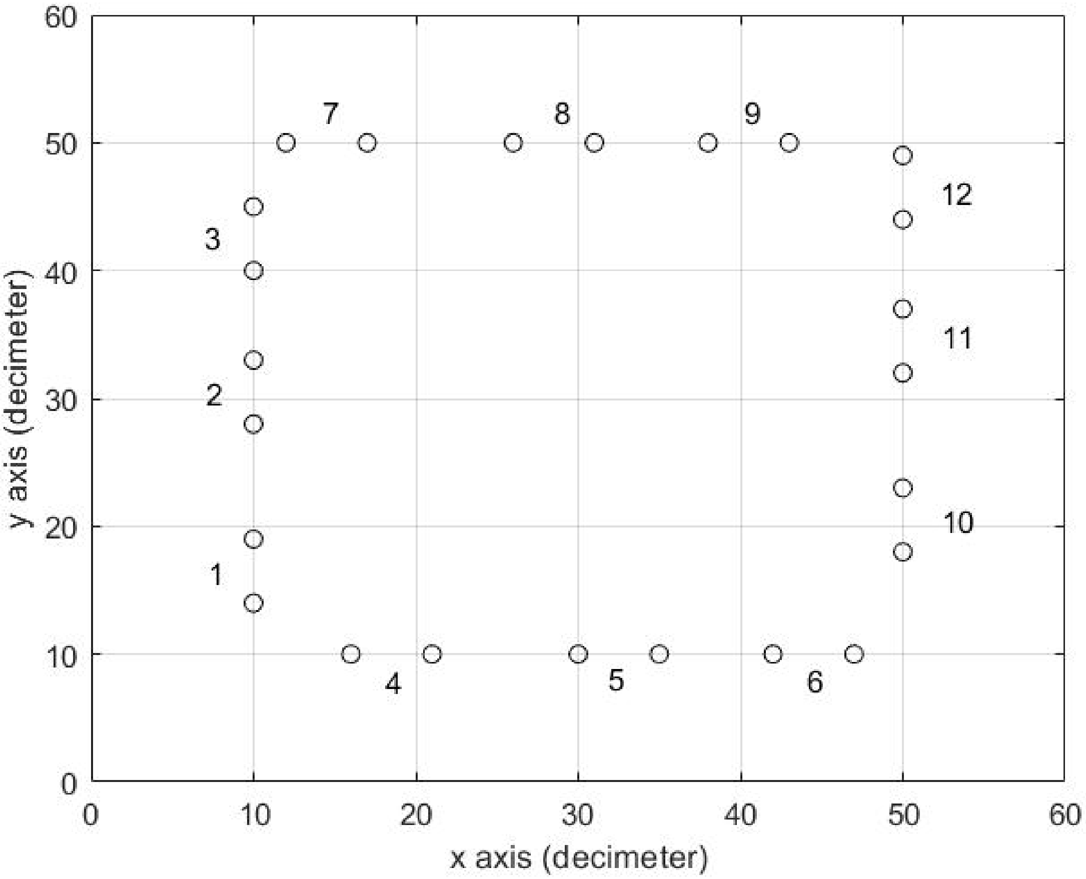


Figure 7.1: Microphone pairs map for the simulated room. The 'o' stands for a microphone positions in the two dimensional plane. The nodes are numbered 1 : 12 containing two microphones each.

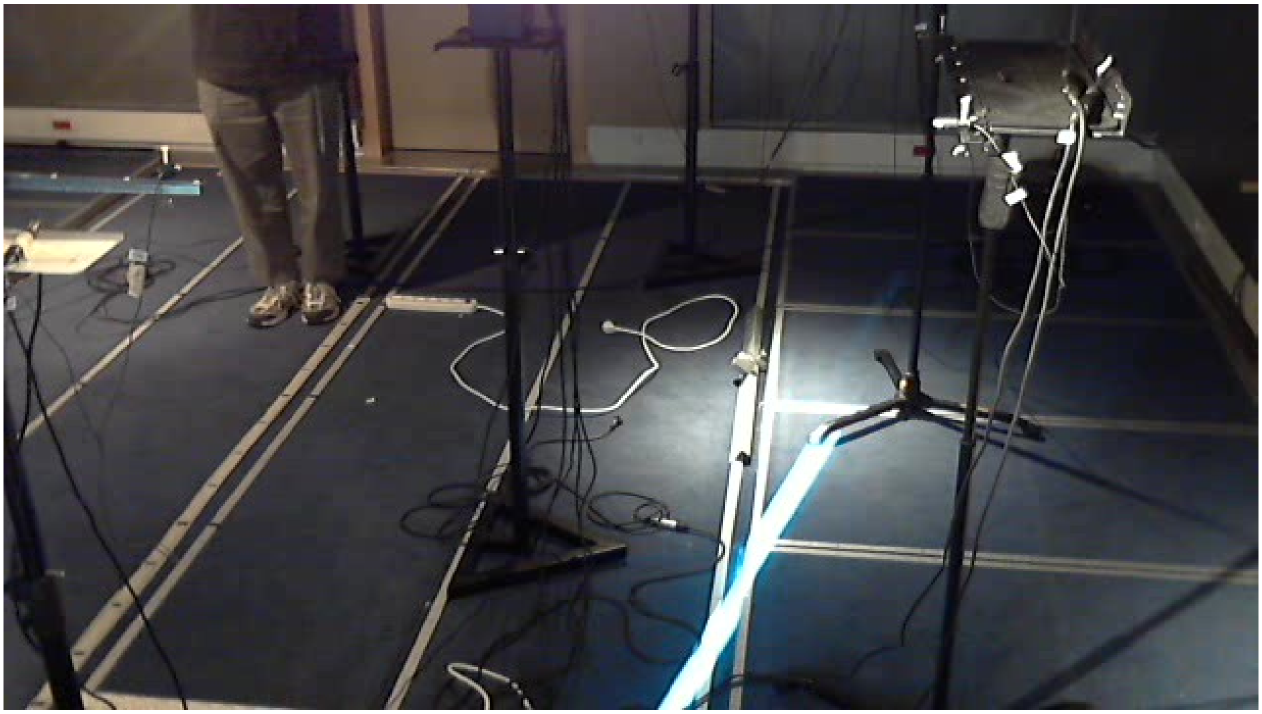


Figure 7.2: Dynamic recording for a single speaker with $T60 = 250$ msec.

single speaker recording with low reverberation level ($T60 = 250$ msec) is shown in Fig. 7.2.

In this case only seven pairs were used, as can be seen in Fig. 7.3. The positions of the microphones around the room are marked by a circle ('o') on a two dimensional plane.

The recordings setup is fixed and the only dynamic is the trajectory of the speakers during the experiment. In all cases we tried to stick to the simple 2D case, placing the microphones in the height of a typical human mouth (135 cm).

7.1.4.3 Simulation Results

The first step of evaluation used simulations. It enables comprehensive statistical tests of a variety of trajectories in the room.

To compare the performance of the algorithms, we followed the procedure described in [193, 192], but this time for a dynamic scenario. We executed 100 Monte Carlo trials and calculated the RMSE.

Note that the accuracy of the location estimation was limited by the grid resolution of 0.10×0.10 m^2. These errors were averaged across the ensemble and along the trajectory. The results are summarized in Table 7.1 for various trajectories.

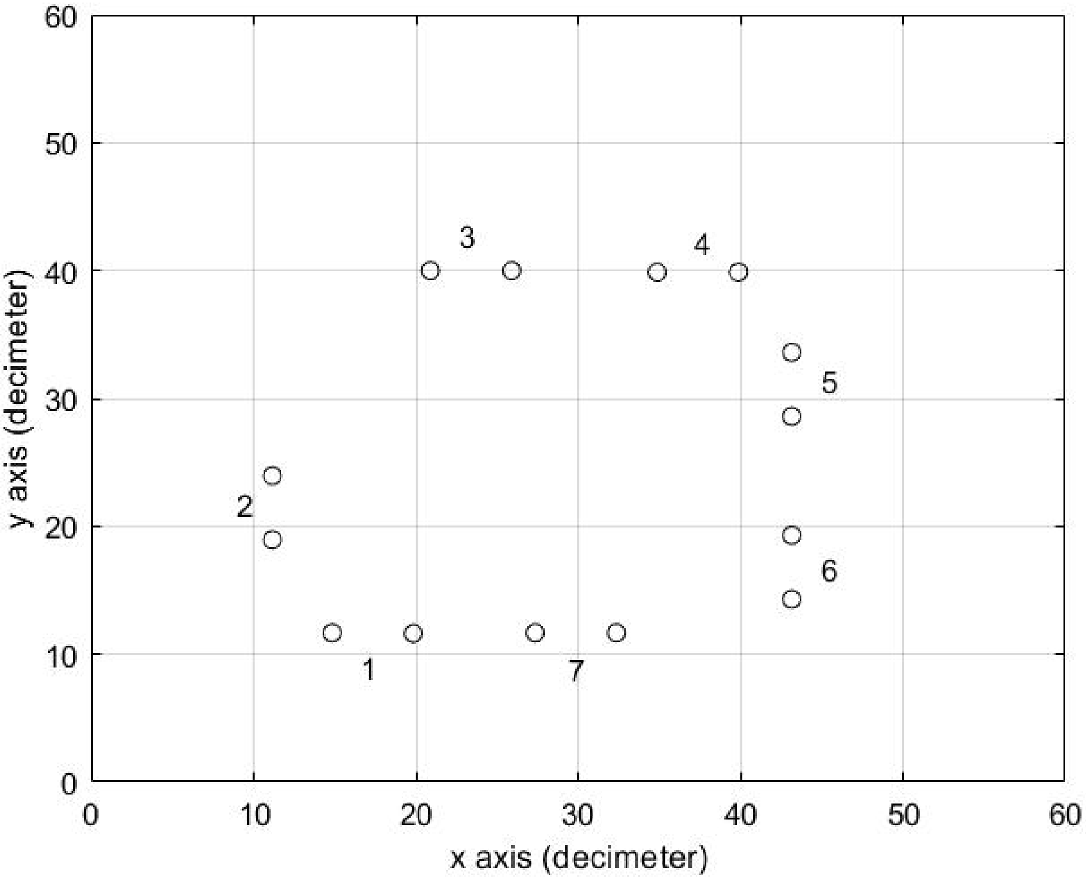


Figure 7.3: Microphone pairs setup during the recordings. The ‘o’ stands for a microphone positions in the two dimensional plane.

Algorithm	2 speakers [m]	1 speaker [m]
RDEM	0.30	0.32
FB-RDEM	0.23	0.30

Table 7.1: RMSE for localization scenarios (100 Monte-Carlo trials) for one or two speakers. The error is calculated in meters.

The RMSE of the first scenario, which consisted of two speakers and where $T_{60} = 120$ msec, is depicted in the first column. The second scenario, which consisted of a single speaker and where $T_{60} = 400$ msec, is depicted in its second column.

The reference algorithm RDEM has a higher RMSE than the FB-RDEM. The improvement is more evident for two speakers, as can be observed also in actual recordings.

7.1.4.4 Analysis of Actual Recordings

The algorithms were also tested using real recordings of sources and eight microphone pairs.

The first recording was in a room with $T60 = 250$ msec and a single speaker standing for 9 sec and then walking on a straight line, 2.10 m long for 33 sec. The trajectory ground truth (Thin line) and the estimated one (Thicker line) are shown in Fig. 7.4. The colors are changing with time from cold to warm (blue to red) for both lines. It can be observed that the estimated trajectory is relatively near the reference.

The localization errors for both algorithms are shown in Fig. 7.5. It can be observed that the FB-RDEM slightly outperforms the RDEM in this case. The RMSE averaged over the trajectory for FB-RDEM is 0.20 m and for RDEM 0.23 m.

The second recording involved two concurrent speakers standing for 2 sec and then walking on parallel straight lines, 1.75 m long for 21 sec. The averaged localization errors for both algorithms are shown in Fig. 7.6. It can be observed that the FB-RDEM significantly outperforms the RDEM. The RMSE for FB-RDEM is 0.48 m and for RDEM 0.83 m. The additional speaker degrades the localization results of both algorithms for two main reasons. First, a speaker may become more dominant than the other. Second, each speaker produces reverberation, which may reduce the localization accuracy of the other source as well.

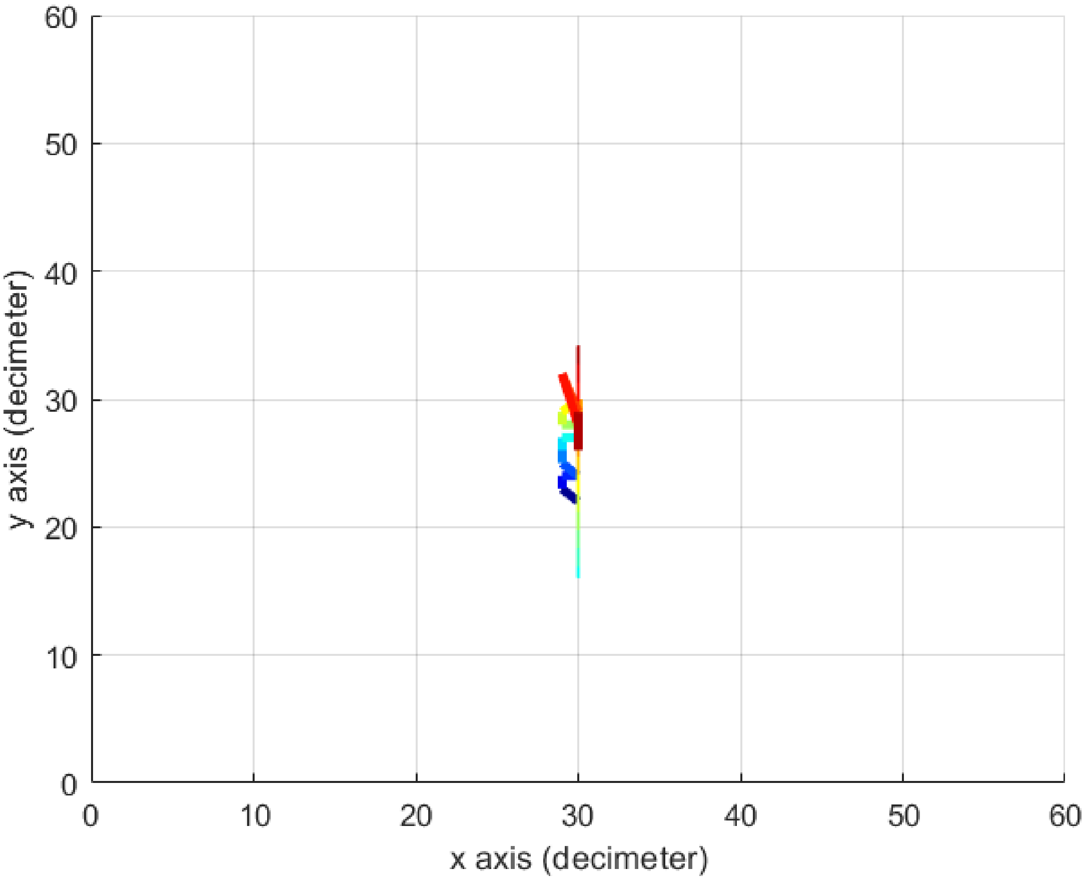


Figure 7.4: Tracking trajectory for one speaker: Ground truth and estimation result.

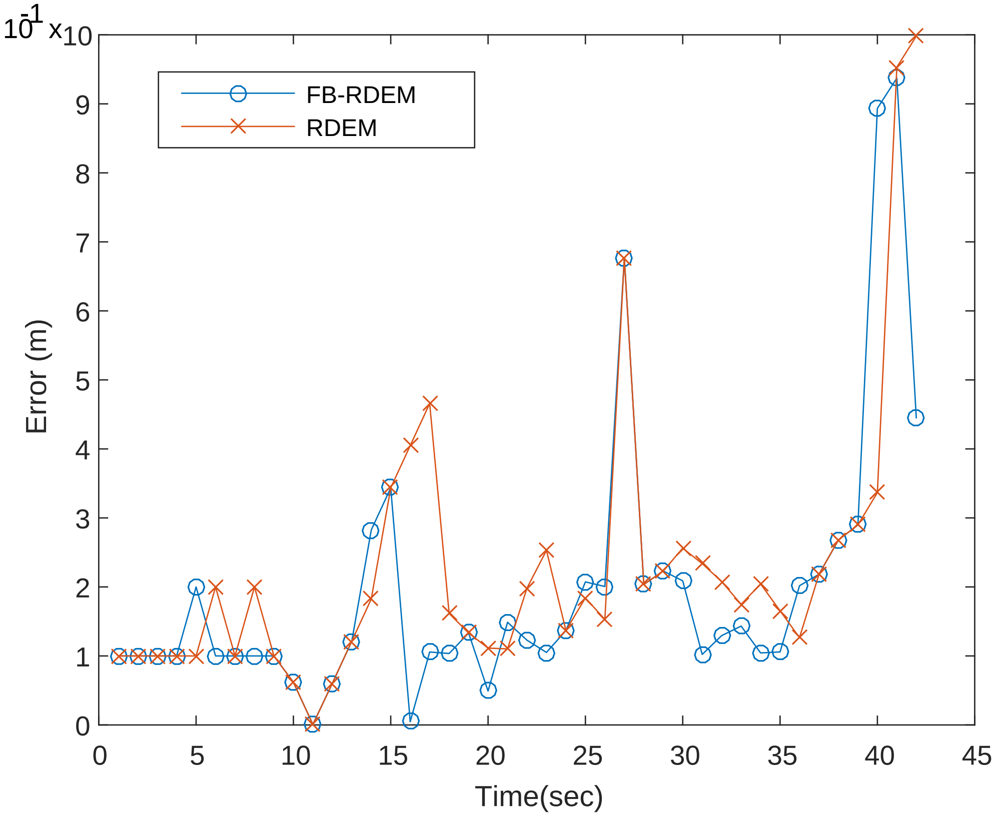


Figure 7.5: Tracking error for one speaker.

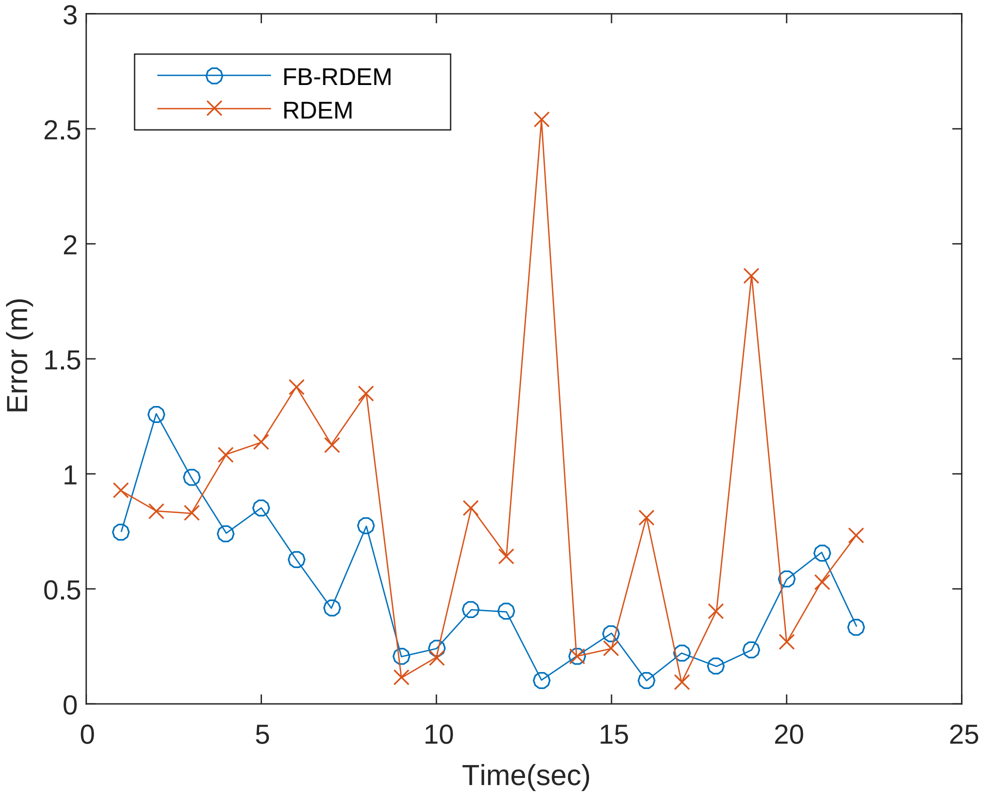


Figure 7.6: Tracking error for two speakers.

7.1.5 Conclusions

The FB-RDEM algorithm was developed for localization of multiple concurrent speakers in reverberant environments under the non-Bayesian assumptions for online applications. A backward RDEM is suggested with the same structure as the forward version. The new algorithm is a combination of the forward RDEM and the backward RDEM.

The network topology used here is a bi-directional tree, which enables efficient implementation of the distributed algorithm.

A short latency has been utilized to enable the new concept and improve the localization results of the RDEM. We demonstrated the advantage of the proposed algorithm by both simulations and real recordings.

7.2 Tracking using moving microphone arrays

We propose a novel approach that combines the EM algorithm within the Bayesian framework, for mutually improved performance. We show that the particle filter can be used to estimate and propagate an adaptive grid of source positions. The particle positions are used within the EM algorithm to estimate and maximize the likelihood of reverberant data, which is subsequently used in the particle filter to assign weights to the particles. Room simulations for realistic conditions demonstrate high accuracy in source tracking using a single moving pair of microphones.

7.2.1 System model

7.2.1.1 Source motion model

The state, $\boldsymbol{s}(t) \triangleq \left[x(t), y(t), \dot{x}(t), \dot{y}(t)\right]^T$, of a source at time step t, and located at position $(x(t), y(t))$ with velocity $(\dot{x}(t), \dot{y}(t))$, can be modeled over time using a Langevin model [233], i.e.,

$$\boldsymbol{s}(t) = \mathbf{F}(t)\boldsymbol{s}(t-1) + \boldsymbol{u}(t), \qquad \boldsymbol{u}(t) \sim \mathcal{N}\left(\mathbf{0}_{4\times 1}, \, \mathbf{Q}(t)\right). \tag{7.31}$$

The dynamics, $\mathbf{F}(t)$, and process noise covariance, $\mathbf{Q}(t)$, are defined as:

$$\mathbf{F}(t) \triangleq \begin{bmatrix} \mathbf{I}_2 & a\Delta_t \mathbf{I}_2 \\ \mathbf{0}_{2\times 2} & a\mathbf{I}_2 \end{bmatrix} \quad \text{and} \quad \mathbf{Q}(t) \triangleq \begin{bmatrix} b^2\,\Delta_t^2 \mathbf{I}_2 & \mathbf{0}_{2\times 2} \\ \mathbf{0}_{2\times 2} & b^2\mathbf{I}_2 \end{bmatrix}, \tag{7.32}$$

where Δ_t is the time step, $a \triangleq e^{-\beta \Delta_t}$ and $b \triangleq \bar{v}\sqrt{1-a^2}$. The parameters β and $\bar{v}$ are the rate constant and the steady-state velocity, respectively.

Therefore, the transition density of the source states, $p\left(\boldsymbol{s}(t) \mid \boldsymbol{s}(t-1)\right)$, is given by probability transformation of (7.31) as:

$$p\left(\boldsymbol{s}(t) \mid \boldsymbol{s}(t-1)\right) = \mathcal{N}\left(\boldsymbol{s}(t) \,\middle|\, \mathbf{F}(t)\boldsymbol{s}(t-1),\, \mathbf{Q}(t)\right). \tag{7.33}$$

7.2.1.2 Robot motion model

The microphone array used here consists of one pair with Cartesian positions, $\boldsymbol{p}_m(t) \triangleq \left[x_m(t), y_m(t)\right]^T$ for $m = 1, 2$ and with an inter-sensor distance of 0.5 m. The microphone pair moves at constant speed along a straight line within the room. In this section, the positions of the microphones are assumed known. For unknown robot trajectory, the source localization approach proposed here can be integrated in the acoustic SLAM from [189, 190, 191].

7.2.1.3 Signal model

The STFT of the clean speech signal emitted by a single source at time t and frequency k is given by $y(t,k)$. The source signal is convolved with the acoustic impulse response (AIR) of the reverberant room, such that the STFT, $z_m(t,k)$, at each of the two microphones is expressed as:

$$z_m(t,k) = h_m(t,k)\, y(t,k), \tag{7.34}$$

where $h_m(t,k)$ is the ATF between the source and microphone $m \in 1, 2$ at time t and frequency k.

The ATF can be separated into the RDTF, $h_m^{(d)}(t,k)$, and the transfer function due to early reflections and late reverberation, $h_m^{(r)}$, such that (7.34) is equivalent to,

$$z_m(t,k) = h_m^{(d)}(t,k)\, y(t,k) + n_m(t,k), \tag{7.35}$$

where the non-direct component, $n_m(t,k)$, captures the effects of early reflections and late reverberation, i.e.,

$$n_m(t,k) \triangleq h_m^{(r)}\, y(t,k). \tag{7.36}$$

Furthermore, $h_m^{(d)}(t,k)$ in (7.35) denotes the RDTF between sensor m and the source, modeled as a function of the source angle, $\vartheta_m(t)$, relative to the array of microphones via the far-field plane-wave approximation [234]:

$$h_m^{(d)}(t,k) = \exp\left\{\frac{2\pi\,\jmath\,k\,d_m\,\cos\vartheta_m(t)}{K\,T_s\,c}\right\}, \tag{7.37}$$

where c is the speed of sound, K is the number of frequency bins, T_s is the sampling period, and d_m is the distance between microphone m and the reference microphone.

The microphone signals, $\boldsymbol{z}(t,k) \triangleq \left[z_1(t,k), z_2(t,k)\right]^T$, can be synonymously expressed in vector form as:

$$\boldsymbol{z}(t,k) = \boldsymbol{h}^{(d)}(t,k)\,y(t,k) + \boldsymbol{n}(t,k), \tag{7.38}$$

where $\boldsymbol{h}^{(d)}(t,k) \triangleq \left[h_1^{(d)}(t,k), h_2^{(d)}(t,k)\right]^T$ with noise term $\boldsymbol{n}(t,k) \triangleq \left[n_1(t,k), n_2(t,k)\right]^T$.

Assuming $\boldsymbol{n}(t,k)$ is white Gaussian, the likelihood of the reverberant signals in (7.38) is given by:

$$p\left(\boldsymbol{z}(t,k)\,|\,\boldsymbol{s}(t)\right) = \mathcal{N}^c\left(\boldsymbol{z}(t,k)\,\middle|\,\mathbf{0}_{M\times 1},\,\boldsymbol{\Phi}(t,k)\right), \tag{7.39}$$

where $\mathcal{N}^c$ denotes the complex Gaussian density, and $\mathbf{0}_{M\times 1}$ is the $M \times 1$ zero vector.

The covariance in (7.39) is given by the PSD, $\boldsymbol{\Phi}(t,k)$:

$$\boldsymbol{\Phi}(t,k) = \boldsymbol{h}^{(d)}(t,k)\,[\boldsymbol{h}^{(d)}(t,k)]^H\,\phi_y(t,k) + \boldsymbol{\Phi}_r(t,k), \tag{7.40}$$

where the direct-path PSD is denoted as $\phi_y(t,k) \triangleq \mathbb{E}\left[|y(t,k)|^2\right]$, and $\boldsymbol{\Phi}_r(t,k) \triangleq \mathbb{E}\left[\boldsymbol{n}(t,k)\,\boldsymbol{n}(t,k)^H\right]$ is the reverberation PSD matrix.

This reverberation PSD can be modeled in terms of the spatial incoherence matrix, $\boldsymbol{\Gamma}(t,k)$, and reverberation level, $\phi_R(t,k)$, as:

$$\boldsymbol{\Phi}_r(t,k) = \boldsymbol{\Gamma}(t,k)\,\phi_R(t,k). \tag{7.41}$$

Assuming reverberation can be modeled as a spatially homogeneous, spherically isotropic sound field [207], element (i,j) for each $\{i,j\} \in \{1,2\}$ of the matrix $\boldsymbol{\Gamma}(t,k)$ can be modeled as:

$$\Gamma_{i,j}(t,k) = \operatorname{sinc}\left(\frac{2\pi k\, d_{i,j}}{K\, T_s\, c}\right) + \epsilon\, \delta(i-j), \qquad \{i,j\} = 1,2 \tag{7.42}$$

where ϵ is the diagonal loading factor, and $d_{i,j}$ is the Euclidean distance between microphone i and j.

7.2.2 Methodology

7.2.2.1 Sequential importance sampling

The MAP estimate of the source position is obtained from the posterior p.d.f., $p\,(\boldsymbol{s}(t) \mid \mathbf{Z}_t)$, via:

$$\hat{\boldsymbol{s}}^{\text{MAP}}(t) = \underset{\boldsymbol{s}(t)}{\operatorname{argmax}}\, p\,(\boldsymbol{s}(t) \mid \mathbf{Z}_{1:t}, \phi_y(t,k), \phi_R(t,k))\,, \tag{7.43}$$

where $\mathbf{Z}_{1:t} = \left[\mathbf{Z}_1^T, \dots, \mathbf{Z}_t^T\right]^T$ with $\mathbf{Z}_t \triangleq \left[\boldsymbol{z}(t,1)^T, \dots, \boldsymbol{z}(t,K)^T\right]^T$.

The posterior density is given via Bayes's theorem as:

$$p\,(\boldsymbol{s}(t) \mid \mathbf{Z}_t, \phi_y(t,k), \phi_R(t,k)) \quad = \frac{p\,(\mathbf{Z}_t \mid \boldsymbol{\theta}_t)\, p\,(\boldsymbol{s}(t) \mid \boldsymbol{s}(t-1))}{\int p\,(\mathbf{Z}_t \mid \boldsymbol{\theta}_t)\, p\,(\boldsymbol{s}(t) \mid \boldsymbol{s}(t-1))\, d\boldsymbol{s}(t)}, \tag{7.44}$$

where $\boldsymbol{\theta}_t \triangleq \left[\boldsymbol{s}^T(t), \phi_y(t,k), \phi_R(t,k)\right]^T$, the likelihood is denoted as $p\,(\mathbf{Z}_t \mid \boldsymbol{\theta}_t)$ and $p\,(\boldsymbol{s}(t) \mid \boldsymbol{s}(t-1))$ is the prior.

However, due to the denominator in (7.44), the posterior p.d.f. is analytically intractable. Nevertheless, the posterior p.d.f. can be approximated by sampling from a proposal density, $\pi\left(\boldsymbol{s}(t) \mid \hat{\boldsymbol{s}}^{(j)}(t-1), \mathbf{Z}_t\right)$, that includes the support of $p\,(\boldsymbol{s}(t) \mid \mathbf{Z}_{1:t})$.

Hence:

$$p\,(\boldsymbol{s}(t) \mid \mathbf{Z}_{1:t}, \phi_y(t,k), \phi_R(t,k)) \approx \sum_{j=1}^{J} \tilde{w}^{(j)}(t)\, \delta_{\hat{\boldsymbol{s}}^{(j)}(t)}(\boldsymbol{s}(t)), \tag{7.45}$$

where $\tilde{w}^{(j)}(t) \triangleq w^{(j)}(t) / \sum_{j=1}^{J} w^{(j)}(t)$ are the normalized importance weights, and the particles, $\hat{\boldsymbol{s}}^{(j)}(t)$, are drawn from:

$$\hat{\boldsymbol{s}}^{(j)}(t) \sim \pi\left(\boldsymbol{s}(t) \mid \hat{\boldsymbol{s}}^{(j)}(t-1), \mathbf{Z}_t\right). \tag{7.46}$$

The unnormalized importance weights, $w^{(j)}(t)$, are given by [235]:

$$w^{(j)}(t) = w^{(j)}(t-1)\frac{p\left(\mathbf{Z}_t \mid \boldsymbol{\theta}_t^{(j)}\right) p\left(\hat{\boldsymbol{s}}^{(j)}(t) \,\middle|\, \hat{\boldsymbol{s}}^{(j)}(t-1)\right)}{\pi\left(\hat{\boldsymbol{s}}^{(j)}(t) \,\middle|\, \hat{\boldsymbol{s}}^{((j))}(t-1), \mathbf{Z}_t\right)}, \tag{7.47}$$

where $\boldsymbol{\theta}_t^{(j)} \triangleq \left[\hat{\boldsymbol{s}}^{(j)}(t), \mathcal{M}\right]^T$.

Using prior importance sampling from (7.33), i.e.,

$$\pi\left(\boldsymbol{s}(t) \mid \hat{\boldsymbol{s}}^{(j)}(t-1), \mathbf{Z}_t\right) = p\left(\boldsymbol{s}(t) \mid \hat{\boldsymbol{s}}^{(j)}(t-1)\right), \tag{7.48}$$

the unnormalized importance weights, $w^{(j)}(t)$ are given by [235]:

$$w^{(j)}(t) = w^{(j)}(t-1)\, p\left(\mathbf{Z}_t \mid \boldsymbol{\theta}_t^{(j)}\right). \tag{7.49}$$

Assuming i.i.d. frequency bins, the likelihood of the reverberant measurements corresponding to particle, $\hat{\boldsymbol{s}}^{(j)}(t)$, can be obtained from (7.39) as:

$$p\left(\mathbf{Z}_t \mid \boldsymbol{\theta}_t^{(j)}\right) = \prod_{k=1}^{K} p\left(\boldsymbol{z}(t,k) \mid \boldsymbol{\theta}_t^{(j)}\right). \tag{7.50}$$

However, the direct-path and reverberant PSDs, $\phi_y(t,k)$ and $\phi_R(t,k)$ are required in order to evaluate the PSD in (7.40). As $\phi_y(t,k)$ and $\phi_R(t,k)$ are unknown in practice, the likelihood in (7.50) and hence the particle weights in (7.49) cannot be evaluated directly from the particles, $\{\hat{\boldsymbol{s}}^{(j)}(t)\}_{j=1}^{J}$.

Nevertheless, considering the cloud of particles as an adaptive grid of source position hypotheses, the ML framework can be used to estimate $p\left(\mathbf{Z}_t \mid \boldsymbol{\theta}_t\right)$ for the importance weights in (7.49).

7.2.2.2 The expectation-maximization algorithm

As the likelihood, $p\left(\mathbf{Z}_t \mid \boldsymbol{\theta}_t\right)$, is generally high-dimensional and multi-modal, it is often difficult to maximize in practice. Rather than maximizing the likelihood directly, the EM algorithm in [196] maximizes the joint density of the observed data and a set of latent, unobserved and discrete variables, $\mathbf{X}_t$:

$$p\left(\mathbf{Z}_t \mid \boldsymbol{\theta}_t\right) = \frac{p\left(\mathbf{Z}_t, \mathbf{X}_t \mid \boldsymbol{\theta}_t\right)}{p\left(\mathbf{X}_t \mid \boldsymbol{\theta}_t\right)}, \tag{7.51}$$

with $\mathbf{X}_t \triangleq \left[x(t,1,\vartheta), \ldots, x(t,K,\vartheta)\right]^T$. The hidden, $x(t,k,\vartheta)$ is an i.i.d. indicator that (t,k) is solely associated with a source in the direction of $\vartheta = \gamma(t) - \tan^{-1}\left(x_r(t)/y_r(t)\right)$, where $(x_r(t), y_r(t))$ is the source position relative to the sensor with orientation $\gamma(t)$.

Moreover, the joint posterior, $p\left(\mathbf{Z}_t, \mathbf{X}_t \mid \boldsymbol{\theta}_t\right)$, can be expressed using (7.39) as [196]:

$$p\left(\mathbf{Z}_t, \mathbf{X}_t \mid \boldsymbol{\theta}_t\right) = \prod_k \sum_{j=1}^{J} x(t,k,\vartheta_j)\, \mathcal{N}^c\left(\boldsymbol{z}(t,k) \,\middle|\, \mathbf{0}_{2\times 1},\, \boldsymbol{\Phi}^{(j)}(t,k)\right). \tag{7.52}$$

In [196] the indicator is evaluated over a predetermined, discrete grid of source directions in $[0, 2\pi]$. This section proposes to use instead the directions of the particles, denoted by $\{\vartheta_j\}_{j=1}^{J}$, as an adaptive grid of $P = J$ source directions.

The log-likelihood corresponding to (7.51) is given by:

$$\ln p\left(\mathbf{Z}_t \mid \boldsymbol{\theta}_t\right) = \ln p\left(\mathbf{Z}_t, \mathbf{X}_t \mid \boldsymbol{\theta}_t\right) - \ln p\left(\mathbf{X}_t \mid \boldsymbol{\theta}_t\right), \tag{7.53}$$

where $p\left(\mathbf{Z}_t, \mathbf{X}_t \mid \boldsymbol{\theta}_t\right)$ is the joint p.d.f. of the complete data and $p\left(\mathbf{X}_t \mid \boldsymbol{\theta}_t\right)$ is marginal density of the indicator.

The p.d.f. of the complete data can be written as [47]:

$$\begin{aligned}\ln p\left(\mathbf{Z}_t, \mathbf{X}_t \mid \boldsymbol{\theta}_t\right) &= Q(\boldsymbol{\theta}_t, \boldsymbol{\theta}_t^{(\ell-1)}) \\ &- \sum_{k,j} p\left(x(t,k,\vartheta_j) \mid \mathbf{Z}_t, \boldsymbol{\theta}_t^{(\ell-1)}\right) \ln p\left(x(t,k,\vartheta_j) \mid \mathbf{Z}_t, \boldsymbol{\theta}_t^{(\ell-1)}\right).\end{aligned} \tag{7.54}$$

The Q function is defined:

$$Q(\boldsymbol{\theta}_t, \boldsymbol{\theta}_t^{(\ell-1)}) \triangleq \sum_{k,j} p\left(x(t,k,\vartheta_j) \mid \mathbf{Z}_t, \boldsymbol{\theta}_t^{(\ell-1)}\right) \ln p\left(\mathbf{Z}_t, x(t,k,\vartheta_j) \mid \boldsymbol{\theta}_t\right). \tag{7.55}$$

The EM algorithm therefore iteratively estimates the maximum likelihood in a two-stage process. Using (7.52), the E-step [196] evaluates:

$$\begin{aligned}\mu^{(\ell-1)}(t,k,j) &\triangleq \mathbb{E}\left[x(t,k,\vartheta_j) \mid \boldsymbol{z}(t,k), \boldsymbol{\theta}^{(\ell-1)}\right] \\ &= \frac{\psi_j^{(\ell-1)} \mathcal{N}^c\left(\boldsymbol{z}(t,k) \,\middle|\, \mathbf{0}_{2\times 1},\ \boldsymbol{\Phi}_j(t,k)\right)}{\sum\limits_{j'=1}^{J} \psi_{j'}^{(\ell-1)} \mathcal{N}^c\left(\boldsymbol{z}(t,k) \,\middle|\, \mathbf{0}_{2\times 1},\ \boldsymbol{\Phi}_{j'}(t,k)\right)} \mu(t,k,\tau(\mathbf{p})) \\ &\triangleq E\left\{x(t,k,\tau(\mathbf{p}))|\phi(t,k);\boldsymbol{\theta}^{(t-1)}\right\} = \frac{\psi_{\tau(\mathbf{p})}^{(t-1)} \mathcal{N}^c\left(\phi(t,k); \tilde{\phi}^k(\tau(\mathbf{p})), \sigma^2\right)}{\sum_{\tau(\mathbf{p})} \psi_{\tau(\mathbf{p})}^{(t-1)} \mathcal{N}^c\left(\phi(t,k); \tilde{\phi}^k(\tau(\mathbf{p})), \sigma^2\right)},\end{aligned} \tag{7.56}$$

where: $\boldsymbol{\Phi}_j(t,k) = \boldsymbol{h}^{(r)}(t,k)\,[\boldsymbol{h}^{(r)}(t,k)]^H\,\phi_{S,j}(t,k) + \boldsymbol{\Gamma}(t,k)\,\phi_{R,j}(t,k)$. $\psi^{(j)}$ is the probability to have a source in the j^{th} direction.

In the M-step, $p\left(\mathbf{Z}_t, x(t,k,\vartheta_j) \mid \boldsymbol{\theta}_t\right)$ is maximized. According to (7.55), this maximization is equivalent to separately maximizing $\ln \psi_j$ and the log-likelihood. By constrained maximization:

$$\psi_j^{(\ell)} = \frac{1}{K}\sum_{k=1}^{K} \mu^{(\ell-1)}(t,k,j). \tag{7.57}$$

Furthermore, maximization of the log-likelihood reduces to [102],

$$\phi_{R,j}(t,k) = \boldsymbol{z}^H(t,k)\left[\mathbf{I}_2 - \boldsymbol{b}_j\,\boldsymbol{h}_j^H(t,k)\,\right]\boldsymbol{\Gamma}^{-1}(t,k)\,\boldsymbol{z}(t,k) \tag{7.58a}$$

$$\phi_{S,j}(t,k) = \boldsymbol{b}_j^H\left[\boldsymbol{z}(t,k)\,\boldsymbol{z}^H(t,k) - \phi_{R,j}(t,k)\,\boldsymbol{\Gamma}(t,k)\right]\boldsymbol{b}_j, \tag{7.58b}$$

where $\boldsymbol{b}_j$ is the minimum variance distortion-less response-BF in the direction ϑ_j, i.e.:

$$\boldsymbol{b}_j = \frac{\boldsymbol{\Gamma}^{-1}(t,k)\,\boldsymbol{h}_j(t,k)}{\boldsymbol{h}_j^H(t,k)\,\boldsymbol{\Gamma}^{-1}(t,k)\,\boldsymbol{h}_j(t,k)}. \tag{7.59}$$

Therefore, instead of using a deterministic grid of source positions, the sampled positions within the particle filter are used to maximize the source directions in (7.56)-(7.58). Simultaneously, the probabilities, $\psi_j^{(L)}$, are used to weight the particles in lieu of (7.49):

$$w^{(j)}(t) = w^{(j)}(t-1)\,\psi_j^{(L)}\,\mathcal{N}^c\left(\boldsymbol{z}(t,k)\,\middle|\,\mathbf{0}_{2\times 1},\ \boldsymbol{\Phi}_j(t,k)\right). \tag{7.60}$$

7.2.3 Results

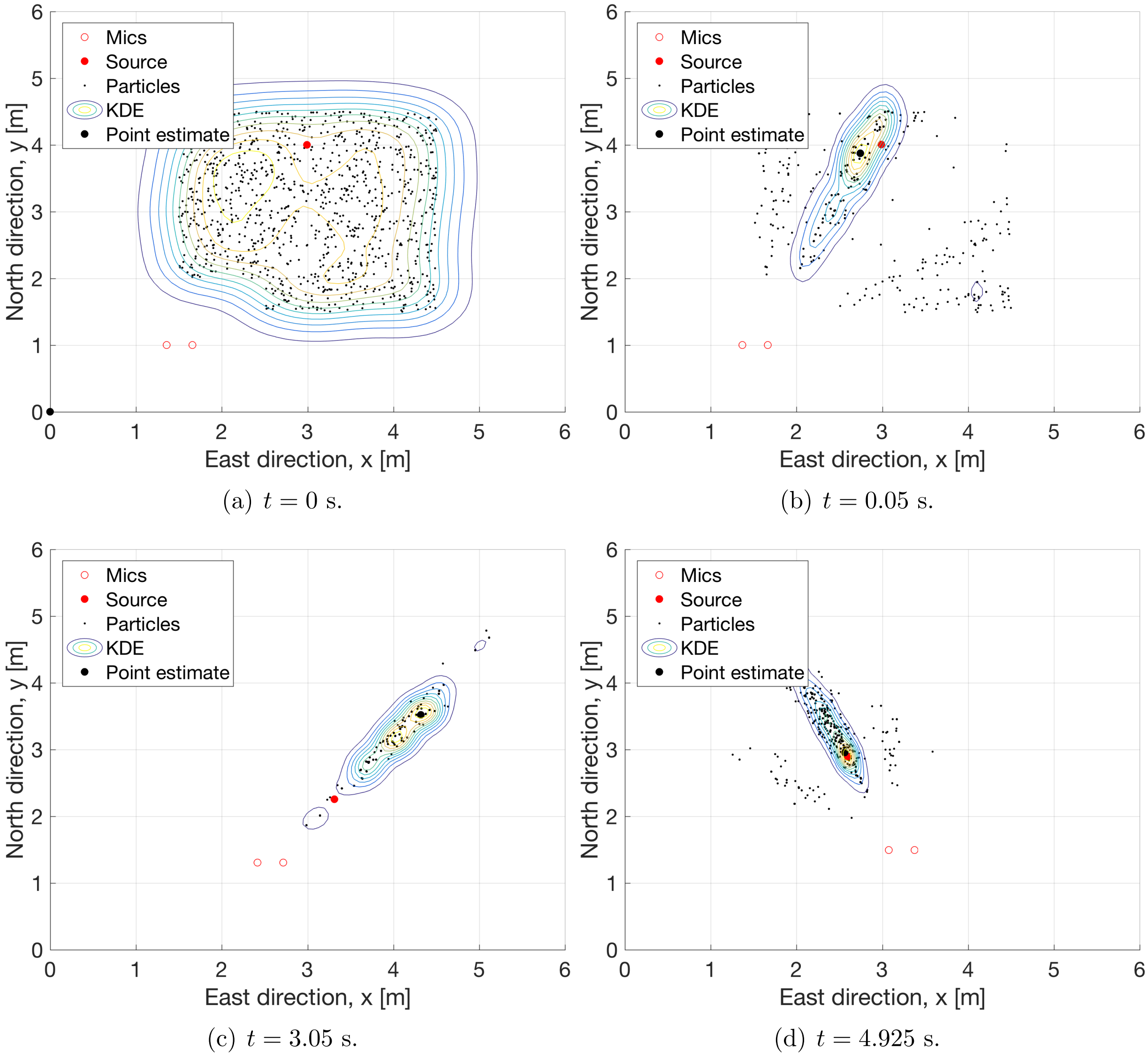
The trajectory of a moving pair of microphones is simulated over 10 sec along a straight line in a $6 \times 6 \times 2.5$ m^3 room with the initial and final position of the origin of the pair at $(1.5, 1, 1.5)$ m and $(5, 2, 1.5)$ m respectively. The two microphones are offset by 0.15 m in x-direction to the left and right of the origin respectively. The trajectory of a moving source is simulated using (7.31) with $\beta = 2$, $\bar{v} = 1$ m/sec. Using the RIR generator in [210] the RIR for each source-sensor geometry (that changes with time) is simulated for $T_{60} = 500$ msec at sampling frequency $f_s = 8$ kHz. The RIR is updated on a frame basis assuming the movement within each frame is small enough. The resulting RIRs are convolved with a 10 sec anechoic speech signal from a female speaker constructed from the TIMIT database. The STFT of the signal is evaluated using a rectangular window for each microphone for a frame length of 50 msec. The proposed approach is evaluated for $\Delta_t = 0.375$ sec using 1000 particles. The initial particles are drawn from a uniform distribution with a minimum distance of 1.5 m to each of the walls and at least 1 m from the microphone origin.

Figure 7.7 shows the distribution of particles at four time steps.

For each time step, the point estimate of the source position is extracted from the particles as the peak of the weighted kernel density estimate (KDE). The KDE is shown in Figure 7.7 as the contour plot, highlighting concentrations of particles with high weight. It can be seen that the peak of the KDE converges to the true source angle. Triangulation of the source position is highly dependent on the source-sensor geometry, resulting in estimation errors of under 0.4 m at $t = 3.05, 4.925$ sec, and an average position error across all time steps of 0.805 m. The main reason for the estimation error is the unmeasured source-sensor distance. The scenario corresponds to source-sensor distances between $[0.688, 4.464]$ m. However, the source position is triangulated from a pair of microphones with only 0.3 m inter-microphone distance and relatively small displacement of approximately 0.15 m between time steps. Nevertheless, high accuracy is achieved in the estimated source angle, with an average of 0.319 deg accuracy.

7.2.4 Conclusion

We proposed a novel approach to sound source tracking in reverberant environments using a single moving pair of microphones. A particle filter is used to propagate hypotheses of source positions across time. At each time step, the EM algorithm uses the particles to estimate



(a) $t = 0$ s.

(b) $t = 0.05$ s.

(c) $t = 3.05$ s.

(d) $t = 4.925$ s.

Figure 7.7: Distribution of particles across time.

and maximize the likelihood of reverberant measurements. The resulting probabilities are used in the particle filters as importance weights. Results for 500 msec reverberation time using two microphones separated by 0.3 m demonstrated estimation accuracy of 0.805 m in position and 0.319 deg in the source direction of arrival.

Chapter 8

Summary and future research directions

This chapter summarizes shortly the main achievements till now and then lists all the future research directions that might be taken from now on.

8.1 Summary

Technology advances in hardware and communication have made the vision of WASN feasible and within grasp. Sensor networks spread over vast environments hold potential for superior performance to classical condensed microphone arrays. WASNs raise several new challenges. This dissertation addressed some of the challenges and proposed possible solutions. The contributions of this dissertation are listed below:

1. Distributed localization: new set of hidden variables for EM and REM enabled development of new algorithms that can be implemented without any central point and surprisingly have better convergence properties: they are less dependent on initial conditions and they converge much faster. The recursive version of one of those algorithms can be applied for a robot audition (dynamic) case. A moving pair of microphones was used for localization utilizing the movement for spatial diversity.

2. Reverberation mitigation: Localization based on direct path in a room is a great challenge, especially for multiple concurrent speakers. New techniques were developed to reduce the influence of the early and late reflections from the walls.

3. Separation: As sometimes happen with the EM algorithm, we discovered that the hidden variables we chose for localization have their own physical meaning. They define the spectral activities of each speaker. We have found out that after localization source separation can be achieved if the hidden variables are applied as spectral masks to any of the microphones in the room. The global localization process enabled local estimation of spectral masks and hence local source separation.

4. Joint calibration and localization: A major challenge for ad hoc networks that localize sources is to calibrate the positions of the arrays (or nodes) of the network. A novel joint calibration and localization algorithm suitable for noisy environment has been derived using the EM algorithm. One of the nodes is used as an anchor node. The calibration, i.e. the nodes positions as well as the speakers localization are applied relatively to the position of this anchor node. A new method has been presented. To overcome the initialization challenge of the batch EM, an incremental process has been suggested that incrementally adds the nodes, instead of trying to solve the entire problem from the beginning of the EM algorithm. The new process, called the LACES algorithm, was studied using both a simulated environment and real recordings.

5. Dynamic scenarios: Tracking moving speakers has been developed based on our localization algorithms. For static arrays we suggested a new REM algorithm, which uses future samples to fill gaps of the trajectories that are caused by short silent time slots. A special case we dealt with is tracking speakers with moving arrays. For this challenge an algorithm that combines Bayesian and non-Bayesian methodologies has been developed.

8.2 Future research directions

Future research directions that can be done are described briefly in this section. They are closely related to the algorithms presented above.

8.2.1 Variance estimation

An important parameter in every MoG is the associated variance. In our case, variance might vary due to many parameters like the directivity of the sources, movements of the sensors or certain channel effects.

Until now, we used a fixed value for the variance. The variance might depend on the node index, on frequency, or on another feature of the measurements.

In case of noisy or remote nodes with respect to a certain source, it might be better to discard some of the nodes. This can be done by utilizing the variance estimation.

Besides its usage for localization or tracking improvement, the variance might be useful for several applications. An interesting feature of the acoustic sources is directivity. Small variance in a node indicates that the source is either close or pointing towards this node. It is interesting to explore whether we can utilize distributed sensing for directivity estimation of the sources. Examples of acoustic sources with a significant directivity are ships in the sea, some types of airplanes, etc. In those cases the directivity of the sources might be important for a few reasons. The first reason is the knowledge of the directivity itself. Given the directivity we can estimate not only the location of the object, but also its orientation. The second reason is its influence on features like location estimation and BSS. Another usage of directivity might be classification of sources.

8.2.2 Acoustic signatures

Besides location of the sources we are interested in some other features. By acoustic signature we mean the spectral contents of a certain source. One application of the acoustic signatures mentioned above is BSS. The hidden variables that allow acoustic signature estimation were used for masking of the BSS algorithm. Another application might be source identification or classification. The acoustic signature of sources has a distinguishable nature. A classical military example is underwater acoustic signatures of ships that can be used for classification. From the local hidden variables we can derive local signatures. Comparing those signatures between different nodes could be also very informative in case of Doppler effect or any other spatial measure.

8.2.3 Prior vs. posterior MoG

An important theoretical discussion that could be developed relates to our choice of prior parameters for the MoG. We could have chosen the posterior instead. Such a comparison could be discussed both from theoretical point of view and from the practical point of view.

8.2.4 Combining supervised learning to the localization and BSS algorithms

There are a few supervised algorithm that became very popular lately. For example deep neural network (DNN). Applying those techniques to our models is an interesting research direction that has not been dealt yet.

8.2.5 Improved BSS based on localization information

The BSS was given in chapter 5 as a proof of concept. In order to take it a few steps forward a few activities are relevant. Firstly, the model should be more comprehensive and support reverberation as mentioned in chapter 4 above. Secondly, additional scores that are often used for BSS including signal to distortion ratio (SDR) and source to artifacts ratio (STAR). Finally, the influence of inaccurate localization results on the separation quality can be explored.

Bibliography

[1] D. Estrin, G. Pottie, and M. Srivastava, “Instrumenting the world with wireless sensor networks,” *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 2033–2036, May 2001.

[2] J. B. Predd, S. R. Kulkarni, and H. V. Poor, “Distributed learning in wireless sensor networks,” *IEEE Signal Processing Magazine*, vol. 23, no. 4, pp. 56–69, Jul. 2006.

[3] D. Culler, D. Estrin, and M. Srivastava, “Overview of sensor networks,” *Computer*, vol. 37, no. 8, pp. 41–49, Aug. 2004.

[4] H. Ochiai, P. Mitran, H.V. Poor, and V. Tarokh, “Collaborative beamforming for distributed wireless ad hoc sensor networks,” *IEEE Transactions on Signal Processing*, vol. 53, no. 11, pp. 4110–4124, Nov. 2005.

[5] M.F.A Ahmed and S.A Vorobyov, “Collaborative beamforming for wireless sensor networks with Gaussian distributed sensor nodes,” *IEEE Transactions on Wireless Communications*, vol. 8, no. 2, pp. 638–643, Feb. 2009.

[6] J. Kotus, K. Lopatka, and A. Czyzewski, “Detection and localization of selected acoustic events in acoustic field for smart surveillance applications,” *Multimedia Tools and Applications*, vol. 68, no. 1, pp. 5–21, Jan. 2014.

[7] J. Sallai, W. Hedgecock, P. Volgyesi, A. Nadas, G. Balogh, and A. Ledeczi, “Weapon classification and shooter localization using distributed multichannel acoustic sensors,” *Journal of Systems Architecture*, vol. 57, no. 10, pp. 869–885, Nov. 2011.

[8] M. S. Brandstein, J. E. Adcock, and H. F. Silverman, “A closed-form location estimator for use with room environment microphone arrays,” *IEEE Transactions on Speech and Audio Processing*, vol. 5, no. 1, pp. 45–50, 1997.

[9] S. E. Dosso and J. Dettmer, "Efficient bayesian multi-source localization using a graphics processing unit," *Proceedings of Meetings on Acoustics*, vol. 19, no. 1, 2013.

[10] S. Prasad, "Asymptotics of bayesian error probability and source super-localization in three dimensions," *Optics Express*, vol. 22, no. 13, pp. 16008–16028, June 2014.

[11] A. Levy, S. Gannot, and E.A.P. Habets, "Multiple-hypothesis extended particle filter for acoustic source localization in reverberant environments," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 6, pp. 1540 –1555, Aug. 2011.

[12] M. Angjelichinoski, D. Denkovski, V. Atanasovski, and L. Gavrilovska, "SPEAR: Source position estimation for anchor position uncertainty reduction," *IEEE Communications Letters*, vol. 18, no. 4, pp. 560–563, Apr. 2014.

[13] T. Routtenberg and J. Tabrikian, "Non-Bayesian periodic cramer-rao bound," *IEEE Transactions on Signal Processing*, vol. 61, no. 4, pp. 1019–1032, Feb. 2013.

[14] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, Jan. 1977, ArticleType: research-article / Full publication date: 1977 / Copyright 1977 Royal Statistical Society.

[15] S. S. Iyengar, K. G. Boroojeni, and N. Balakrishnan, "Expectation maximization for acoustic source localization," in *Mathematical Theories of Distributed Sensor Networks*, pp. 37–54. Springer New York, Jan. 2014.

[16] O. Schwartz and S. Gannot, "Speaker tracking using recursive EM algorithms," *Transactions on Audio, Speech, and Language Processing,*, vol. 22, no. 2, pp. 392–402, 2014.

[17] N. Madhu and J. Wouters, "Localisation-based, situation-adaptive mask generation for source separation," in *4th International Symposium on Communications, Control and Signal Processing (ISCCSP)*, Mar. 2010, pp. 1–6.

[18] F. Nesta and M. Omologo, "Enhanced multidimensional spatial functions for unambiguous localization of multiple sparse acoustic sources," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar. 2012, pp. 213–216.

[19] R. J. Jesper and M. G. Christensen, "Near-field localization of audio: A maximum likelihood approach," *Proceedings of the European Signal Processing Conference (EUSIPCO)*, pp. 1–5, 2014.

[20] S. Haykin and K. J. R. Liu, *Handbook on Array Processing and Sensor Networks*, John Wiley & Sons, Feb. 2010.

[21] W. Yang, G. Chen, X. Wang, and L. Shi, "Stochastic sensor activation for distributed state estimation over a sensor network," *Automatica*, vol. 50, no. 8, pp. 2070–2076, Aug. 2014.

[22] A. Bertrand and M. Moonen, "Distributed signal estimation in sensor networks where nodes have different interests," vol. 92, July 2012.

[23] S. Markovich-Golan and S. Gannot, "Distributed multiple constraints generalized sidelobe canceler for fully connected wireless acoustic sensor networks," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 2, pp. 343–356, 2013.

[24] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, Aug. 2008.

[25] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," in *Proceedings of the 3rd international symposium on Information processing in sensor networks*, New York, NY, USA, 2004, IPSN '04, pp. 20–27, ACM.

[26] P. Pertil, T. Korhonen, and A. Visa, "Measurement combination for acoustic source localization in a room environment," vol. 2008, pp. 1–14, 2008.

[27] Y. Oualil, F. Faubel, and D. Klakow, "A fast cumulative steered response power for multiple speaker detection and localization," in *European Signal Processing Conference (EUSIPCO)*, Marrakech, Morocco, 2013.

[28] M. Taseska, G. Lamani, and E. A. P. Habets, "Online clustering of narrowband position estimates with application to multi-speaker detection and tracking," in *Advances in Machine Learning and Signal Processing*, pp. 59–69. Springer, 2016.

[29] A. Griffin and A. Mouchtaris, "Localizing Multiple Audio Sources from DOA Estimates in a Wireless Acoustic Sensor Network," *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2013.

[30] A. Plinge and G. A. Fink, "Multi-speaker tracking using multiple distributed microphone arrays," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, Florence, Italy, May 2014.

[31] D. Li and Y. H. Hu, "Energy-based collaborative source localization using acoustic microsensor array," *EURASIP Journal on Advances in Signal Processing*, Mar. 2003.

[32] D. Ampeliotis and K. Berberidis, "Low complexity multiple acoustic source localization in sensor networks based on energy measurements," *Signal Processing*, vol. 90, no. 4, pp. 1300–1312, Apr. 2010.

[33] Z. Liu, Z. Zhang, L.-W. He, and P. Chou, "Energy-based sound source localization and gain normalization for ad hoc microphone arrays," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2007, vol. 2, pp. 761–764.

[34] D. Blatt and A.O. Hero, "Energy-based sensor network source localization via projection onto convex sets," *IEEE Transactions on Signal Processing*, vol. 54, no. 9, pp. 3614–3619, 2006.

[35] M. Chen, Z. Liu, L.-w. He, P. Chou, and Z. Zhang, "Energy-based position estimation of microphones and speakers for ad hoc microphone arrays," in *2007 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, Oct. 2007, pp. 22 –25.

[36] C. Meesookho, U. Mitra, and S. Narayanan, "On energy-based acoustic source localization for sensor networks," *IEEE Transactions on Signal Processing*, vol. 56, no. 1, pp. 365 –377, Jan. 2008.

[37] W. Meng, W. Xiao, and L. Xie, "An efficient EM algorithm for energy-based multisource localization in wireless sensor networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 3, pp. 1017–1027, 2011.

[38] C. Hekimian-Williams, B. Grant, X. Liu, Z. Zhang, and P. Kumar, "Accurate localization of RFID tags using phase difference," in *IEEE International Conference on RFID*, Apr. 2010, pp. 89–96.

[39] M. I. Mandel, D. P. W. Ellis, and T. Jebara, "An EM algorithm for localizing multiple sound: Sources in reverberant environments," *Advances in neural information processing systems*, pp. 953–960, 2007.

[40] M.I. Mandel and D.P.W. Ellis, “The ideal interaural parameter mask: A bound on binaural separation systems,” in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2009, pp. 85–88.

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

[42] Q. Shen, W. Liu, W. Cui, S. Wu, Y. D. Zhang, and M. G. Amin, “Low-complexity direction-of-arrival estimation based on wideband co-prime arrays,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 9, pp. 1445–1456, 2015.

[43] W. Xue and W. Liu, “Direction of arrival estimation based on subband weighting for noisy conditions.,” in *Thirteenth Annual Conference of the International Speech Communication Association*, 2012, pp. 142–145.

[44] D. Pavlidi, A. Griffin, M. Puigt, and A. Mouchtaris, “Real-time multiple sound source localization and counting using a circular microphone array,” vol. 21, no. 10, pp. 2193–2206, 2013.

[45] S. Tervo and A. Politis, “Direction of arrival estimation of reflections from room impulse responses using a spherical microphone array,” *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 10, pp. 1539–1551, 2015.

[46] S. Chakrabarty and E. A. P. Habets, “Broadband DOA estimation using convolutional neural networks trained with noise signals,” in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, USA, Oct. 2017.

[47] C. M Bishop et al., *Pattern recognition and machine learning*, vol. 4, springer New York, 2006.

[48] R.D. Nowak, “Distributed EM algorithms for density estimation and clustering in sensor networks,” *IEEE Transactions on Signal Processing*, vol. 51, no. 8, pp. 2245–2253, 2003.

[49] D. Gu, “Distributed EM algorithm for gaussian mixtures in sensor networks,” *Neural Networks, IEEE Transactions on*, vol. 19, no. 7, pp. 1154–1166, 2008.

[50] Y. Weng, W. Xiao, and L. Xie, "Diffusion-based EM algorithm for distributed estimation of Gaussian mixtures in wireless sensor networks," *Sensors*, vol. 11, no. 6, pp. 6297–6316, 2011.

[51] R. Neal and G.E. Hinton, "A view of the EM algorithm that justifies incremental, sparse, and other variants," in *Learning in Graphical Models*. 1998, pp. 355–368, Kluwer Academic Publishers.

[52] P. Liang and D. Klein, "Online EM for unsupervised models," in *Proceedings of Human Language Technologies*, Stroudsburg, PA, USA, 2009, NAACL '09, pp. 611–619, Association for Computational Linguistics.

[53] M. A. Sato, "Fast learning of on-line EM algorithm," *Rapport Technique, ATR Human Information Processing Research Laboratories*, 1999.

[54] D. M. Titterington, "Recursive parameter estimation using incomplete data," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 46, no. 2, pp. 257–267, Jan. 1984.

[55] G. C. Carter, "Time delay estimation for passive sonar signal processing," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 29, no. 3, pp. 463–470, 1981.

[56] C. Evers, A. H. Moore, P. A. Naylor, J. Sheaffer, and B. Rafaely, "Bearing-only acoustic tracking of moving speakers for robot audition," in *IEEE International Conference on Digital Signal Processing (DSP), 2015*, pp. 1206–1210.

[57] A. Plinge, M. H. Hennecke, and G. A. Fink, "Robust neuro-fuzzy speaker localization using a circular microphone array," in *Proc. Int. Workshop on Acoustic Echo and Noise Control (IWAENC)*, Tel Aviv, Israel, 2010.

[58] M. H. Hennecke and G. A. Fink, "Towards acoustic self-localization of ad hoc smartphone arrays," in *Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA), 2011*. IEEE, pp. 127–132.

[59] T.-K. Le and N. Ono, "Closed-form and near closed-form solutions for toa-based joint source and sensor localization.," *IEEE Trans. Signal Processing*, vol. 64, no. 18, pp. 4751–4766, 2016.

[60] K. Nakadai, G. Ince, K. Nakamura, and H. Nakajima, "Robot audition for dynamic environments," in *IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC), 2012*, Aug 2012, pp. 125–130.

[61] S Argentieri, A Portello, M Bernard, P Danes, and B Gas, "Binaural systems in robotics," in *The technology of binaural listening*, pp. 225–253. Springer, 2013.

[62] Gabriel Bustamante, Patrick Danés, Thomas Forgue, and Ariel Podlubne, "Towards information-based feedback control for binaural active localization," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on.* IEEE, 2016, pp. 6325–6329.

[63] Quan V Nguyen, Francis Colas, Emmanuel Vincent, and François Charpillet, "Localizing an intermittent and moving sound source using a mobile robot," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on.* IEEE, 2016, pp. 1986–1991.

[64] G. J. Heard and I. Schumacher, "Synthetic aperture matched field approach to acoustic source localization in a shallow-water environment," *Canadian Acoustics*, vol. 26, no. 2, pp. 3–10, 1998.

[65] M. Brandstein and D. Ward, Eds., *Microphone Arrays*, vol. 8 of *Digital Signal Processing*, Springer Berlin Heidelberg, Jan. 2001.

[66] R. Martin, U. Heute, and C. Antweiler, *Advances in Digital Speech Transmission*, Wiley, 1 edition, 2008.

[67] A. Canclini, P. Bestagini, F. Antonacci, M. Compagnoni, A. Sarti, and S. Tubaro, "A robust and low-complexity source localization algorithm for asynchronous distributed microphone networks," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 10, pp. 1563–1575, 2015.

[68] Y. Tian, Z. Chen, and F. Yin, "Distributed IMM-unscented Kalman filter for speaker tracking in microphone array networks," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 10, pp. 1637–1647, 2015.

[69] A. Hassani, A. Bertrand, and M. Moonen, "Distributed node-specific direction-of-arrival estimation in wireless acoustic sensor networks," in *Proceedings of the 21st European Signal Processing Conference (EUSIPCO)*, 2013, pp. 1–5.

[70] A. Hassani, A. Bertrand, and M. Moonen, "Cooperative integrated noise reduction and node-specific direction-of-arrival estimation in a fully connected wireless acoustic sensor network," *Signal Processing*, vol. 107, pp. 68–81, 2015.

[71] O. Schwartz, Y. Dorfan, M. Taseska, E. AP Habets, and S. Gannot, "DOA estimation in noisy environment with unknown noise power using the EM algorithm," in *Hands-free Speech Communications and Microphone Arrays (HSCMA)*, 2017, pp. 86–90.

[72] F. Antonacci, D. Lonoce, M. Motta, A. Sarti, and S. Tubaro, "Efficient source localization and tracking in reverberant environments using microphone arrays," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2005, vol. 4, pp. 1061–1064.

[73] W. Li, Y. Jia, J. Du, and J. Zhang, "Distributed multiple-model estimation for simultaneous localization and tracking with nlos mitigation," *IEEE transactions on vehicular technology*, vol. 62, no. 6, pp. 2824–2830, 2013.

[74] O. Oçal, I. Dokmanic, and M. Vetterli, "Source localization and tracking in non-convex rooms," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 1429–1433.

[75] F. Ribeiro, D. Ba, C. Zhang, and D. Florêncio, "Turning enemies into friends: Using reflections to improve sound source localization," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2010, pp. 731–736.

[76] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, "Semi-supervised source localization on multiple-manifolds with distributed microphones," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 7, 2017.

[77] I. Marković and I. Petrović, "Speaker localization and tracking with a microphone array on a mobile robot using von mises distribution and particle filtering," *Robotics and Autonomous Systems*, vol. 58, no. 11, pp. 1185–1196, 2010.

[78] C. Busso, S. Hernanz, C.-W. Chu, S. Kwon, S. Lee, P. G Georgiou, I. Cohen, and S. Narayanan, "Smart room: participant and speaker localization and identification," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2005.

[79] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, "Speaker tracking on multiple-manifolds with distributed microphones," in *International Conference on Latent Variable Analysis and Signal Separation*. Springer, 2017, pp. 59–67.

[80] T. Wolff, M. Buck, and G. Schmidt, "A subband based acoustic source localization system for reverberant environments," in *ITG Conference on Voice Communication (SprachKommunikation)*. VDE, 2008.

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

[82] J. H. DiBiase, H. F. Silverman, and M. S. Brandstein, "Robust localization in reverberant rooms," in *Microphone Arrays*, Michael Brandstein and Darren Ward, Eds., Digital Signal Processing, pp. 157–180. Springer Berlin Heidelberg, 2001.

[83] J.H. DiBiase, H.F. Silverman, and M.S. Brandstein, *Microphone arrays : signal processing techniques and applications*, chapter 8: Robust Localization in Reverberant Rooms, pp. 157–180, Springer Verlag, 2001.

[84] B. Kwon, Y. Park, and Y. Park, "Analysis of the GCC-PHAT technique for multiple sources," in *International Conference on Control Automation and Systems (ICCAS), 2010*, 2010, pp. 2070–2073.

[85] M. Swartling, N. Grbic, and I. Claesson, "Direction of arrival estimation for multiple speakers using time-frequency orthogonal signal separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2006.

[86] A. Brutti, M. Omologo, and P. Svaizer, "Localization of multiple speakers based on a two step acoustic map analysis," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2008, pp. 4349–4352.

[87] H. Do, H.F. Silverman, and Y. Yu, “A real-time SRP-PHAT source location implementation using stochastic region contraction(SRC) on a large-aperture microphone array,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2007, vol. 1, pp. 121–124.

[88] A. D. Firoozabadi and H. R. Abutalebi, “Combination of nested microphone array and subband processing for multiple simultaneous speaker localization,” in *Sixth IEEE International Symposium on Telecommunications (IST)*, 2012, pp. 907–912.

[89] A. D. Firoozabadi and H. R. Abutalebi, “Localization of multiple simultaneous speakers by combining the information from different subbands,” in *21st Iranian Conference on Electrical Engineering (ICEE)*, 2013, pp. 1–6.

[90] D. Wang and G. J. Brown, Eds., *Computational auditory scene analysis: principles, algorithms, and applications*, Wiley, 2006.

[91] R. F. Lyon, *Human and Machine Hearing: Extracting Meaning from Sound*, Cambridge University Press, June 2017.

[92] R. F. Lyon, “A computational model of binaural localization and separation,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1983, vol. 8, pp. 1148–1151.

[93] T. May, S. Par, and A. Kohlrausch, “Binaural localization and detection of speakers in complex acoustic scenes,” in *The Technology of Binaural Listening*, Jens Blauert, Ed., pp. 397–425. Springer, Berlin, Heidelberg, 2013.

[94] K. J. Palomäki, G. J. Brown, and D. Wang, “A binaural processor for missing data speech recognition in the presence of noise and small-room reverberation,” vol. 43, no. 4, pp. 361–378, 2004.

[95] M. Brck and J. L. van Hemmen, “Modeling the cochlear nucleus: A site for monaural echo suppression?,” *Journal Acoustic Society of America*, vol. 122, pp. 2226–2235, 2007.

[96] B. Grothe, “New roles for synaptic inhibtion in sound localisation,” *Nature*, vol. 4, no. 7, pp. 540–550, 2003.

[97] M. Cooke, "A glimpsing model of speech perception in noise," *The Journal of the Acoustical Society of America*, vol. 119, no. 3, pp. 1562–1573, 2006.

[98] A. Plinge, D. Hauschildt, M. H. Hennecke, and G. A. Fink, "Multiple speaker tracking using a microphone array by combining auditory processing and a gaussian mixture cardinalized probability hypothesis density filter," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Prague, Czech Republic, 2011, pp. 2476–2479.

[99] A. Plinge and G. A. Fink, "Online multi-speaker tracking using multiple microphone arrays informed by auditory scene analysis," in *European Signal Processing Conference (EUSIPCO)*, 2013, pp. 1–5.

[100] A. Plinge, M. H. Hennecke, and G. A. Fink, "Reverberation-robust online multi-speaker tracking by using a microphone array and casa processing," in *Acoustic Signal Enhancement; Proceedings of IWAENC 2012; International Workshop on.* VDE, 2012, pp. 1–4.

[101] Q. Zhao, V. Hautamki, I. Krkkinen, and P. Frnti, "Random swap EM algorithm for Gaussian mixture models," *Pattern Recognition Letters*, vol. 33, no. 16, pp. 2120–2126.

[102] H. Ye and R. D. DeGroat, "Maximum likelihood DOA estimation and asymptotic Cramér-Rao bounds for additive unknown colored noise," *IEEE Transactions on Signal Processing*, vol. 43, no. 4, pp. 938–949, 1995.

[103] A. Kuklasinski, S. Doclo, S. H. Jensen, and J. Jensen, "Maximum likelihood based multi-channel isotropic reverberation reduction for hearing aids," in *Proceedings of the 22nd European Signal Processing Conference (EUSIPCO)*, 2014, pp. 61–65.

[104] E. Vincent, C. Févotte, R. Gribonval, L. Benaroya, X. Rodet, A. Röbel, E. Le Carpentier, and F. Bimbot, "A tentative typology of audio source separation tasks," in *4th Int. Symp. on Independent Component Analysis and Blind Signal Separation (ICA)*, 2003, pp. 715–720.

[105] E. C. Cherry, "Some experiments on the recognition of speech, with one and with two ears," *The Journal of the Acoustical Society of America*, vol. 25, no. 5, pp. 975–979, 1953.

[106] S. Haykin and Z. Chen, “The cocktail party problem,” *Neural Computation*, vol. 17, no. 9, pp. 1875–1902, 2005.

[107] H. Sawada, S. Araki, and S. Makino, “A two-stage frequency-domain blind source separation method for underdetermined convolutive mixtures,” in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2007, pp. 139–142.

[108] J-F Cardoso, “Blind signal separation: statistical principles,” *Proceedings of the IEEE*, vol. 86, no. 10, pp. 2009–2025, 1998.

[109] P. Bofill and M. Zibulevsky, “Underdetermined blind source separation using sparse representations,” *Signal Processing*, vol. 81, no. 11, pp. 2353–2362, 2001.

[110] O. Yilmaz and S. Rickard, “Blind separation of speech mixtures via time-frequency masking,” *IEEE Transactions on Signal Processing,*, vol. 52, no. 7, pp. 1830–1847, 2004.

[111] P.D. O’Grady and B.A. Pearlmutter, “The LOST algorithm: finding lines and separating speech mixtures,” *EURASIP Journal on Advances in Signal Processing*, , no. 1, 2008.

[112] H. Sawada, S. Araki, and S. Makino, “Underdetermined convolutive blind source separation via frequency bin-wise clustering and permutation alignment,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 3, pp. 516–527, 2011.

[113] N. Roman, D. Wang, and G.J. Brown, “Speech segregation based on sound localization,” *The Journal of the Acoustical Society of America*, vol. 114, no. 4, pp. 2236–2252, 2003.

[114] N. Madhu and R. Martin, “A versatile framework for speaker separation using a model-based speaker localization approach,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 1900–1912, 2011.

[115] H. Sawada, S. Araki, R. Mukai, and S. Makino, “Grouping separated frequency components by estimating propagation model parameters in frequency-domain blind source separation,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1592–1604, 2007.

[116] S. Araki, H. Sawada, and S. Makino, "Blind speech separation in a meeting situation with maximum SNR beamformers," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2007, vol. 1, pp. I–41.

[117] M. Souden, S. Araki, K. Kinoshita, T. Nakatani, and H. Sawada, "A multichannel MMSE-based framework for speech source separation and noise reduction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 9, pp. 1913–1928, 2013.

[118] M. Taseska and E. A. P. Habets, "Spotforming: Spatial filtering with distributed arrays for position-selective sound acquisition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 7, pp. 1291–1304, 2016.

[119] Z. Koldovsky, F. Nesta, P. Tichavsky, and N. Ono, "Frequency-domain blind speech separation using incomplete de-mixing transform," in *European Signal Processing Conference (EUSIPCO)*, 2016.

[120] G. Lathoud, J.-M. Odobez, and D. Gatica-Perez, "Av16. 3: An audio-visual corpus for speaker localization and tracking.," in *MLMI*. Springer, 2004, pp. 182–195.

[121] T. Yamada, S. Nakamura, and K. Shikano, "Robust speech recognition with speaker localization by a microphone array," in *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*. IEEE, 1996, vol. 3, pp. 1317–1320.

[122] S. Doclo and M. Moonen, "Robust adaptive time delay estimation for speaker localization in noisy and reverberant acoustic environments," *EURASIP Journal on Applied Signal Processing*, vol. 2003, pp. 1110–1124, 2003.

[123] N. Madhu and R. Martin, "A scalable framework for multiple speaker localization and tracking," in *in Proceedings of the International Workshop for Acoustic Echo Cancellation and Noise Control (IWAENC 2008*. Citeseer, 2008.

[124] E. A. P. Habets, "Multi-channel speech dereverberation based on a statistical model of late reverberation," in *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*. IEEE, 2005, vol. 4, pp. iv–173.

[125] O. Schwartz, S. Gannot, and E. A. P. Habets, "Multi-microphone speech dereverberation and noise reduction using relative early transfer functions," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 2, pp. 240–251, 2015.

[126] S. A. Vorobyov, A. B. Gershman, and K. M. Wong, "Maximum likelihood direction-of-arrival estimation in unknown noise fields using sparse sensor arrays," *IEEE Transactions on Signal Processing*, vol. 53, no. 1, pp. 34–43, 2005.

[127] K. Yao, J.C. Chen, and R.E. Hudson, "Maximum-likelihood acoustic source localization: experimental results," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2002, vol. 3, pp. III–2949.

[128] H. Wang, C.-E. Chen, A. Ali, S. Asgari, R. E. Hudson, K. Yao, D. Estrin, and C. Taylor, "Acoustic sensor networks for woodpecker localization," in *Advanced Signal Processing Algorithms, Architectures, and Implementations XV*. International Society for Optics and Photonics, 2005, vol. 5910, p. 591009.

[129] S. Araki, H. Sawada, R. Mukai, and S. Makino, "Doa estimation for multiple sparse sources with arbitrarily arranged multiple sensors," *Journal of Signal Processing Systems*, vol. 63, no. 3, pp. 265–275, 2011.

[130] J. C. Chen, K. Yao, and R. E. Hudson, "Source localization and beamforming," *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 30–39, 2002.

[131] A. Griffin, A. Alexandridis, D. Pavlidi, Y. Mastorakis, and A. Mouchtaris, "Localizing multiple audio sources in a wireless acoustic sensor network," *Signal Processing*, vol. 107, pp. 54–67, 2015.

[132] A. J. Weiss and A. Amar, "Direct position determination of multiple radio signals," *EURASIP Journal on Advances in Signal Processing*, vol. 2005, no. 1, pp. 37–49.

[133] J. Teng, H. Snoussi, C. Richard, and R. Zhou, "Distributed variational filtering for simultaneous sensor localization and target tracking in wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 5, pp. 2305–2318, June 2012.

[134] P. Pertilä, M. Mieskolainen, and M. S. Hämäläinen, "Closed-form self-localization of asynchronous microphone arrays," in *Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA)*, 2011, pp. 139–144.

[135] A. Plinge, F. Jacob, R. Haeb-Umbach, and G. A. Fink, "Acoustic microphone geometry calibration: An overview and experimental evaluation of state-of-the-art algorithms," vol. 33, no. 4, pp. 14–29, July 2016.

[136] P. Pertilä, M. Mieskolainen, and M. S. Hämäläinen, "Passive Self-Localization of Microphones using Ambient Sounds," in *Proc. European Signal Processing Conf. (EUSIPCO)*, Bucharest, Romania, Aug. 2012, pp. 1314–1318.

[137] A. Plinge, G. A. Fink, and S. Gannot, "Passive online geometry calibration of acoustic sensor networks," *IEEE Signal Processing Letters*, vol. 24, no. 3, pp. 324–328, 2017.

[138] Y. Rockah and P. Schultheiss, "Array shape calibration using sources in unknown locations–part I: Far-field sources," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 35, no. 3, pp. 286–299, 1987.

[139] Y. Rockah and P. Schultheiss, "Array shape calibration using sources in unknown locations–part II: Near-field sources and estimator implementation," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 35, no. 6, pp. 724–735, 1987.

[140] R. L. Moses, D. Krishnamurthy, and R. M. Patterson, "A self-localization method for wireless sensor networks," *EURASIP Journal on Advances in Signal Processing*, vol. 2003, no. 4, pp. 348–358.

[141] S. Zhayida, F. Andersson, Y. Kuang, and K. rAström, "An automatic system for microphone self-localization using ambient sound," in *Proceedings of the 22nd European Signal Processing Conference (EUSIPCO)*, 2014, pp. 954–958.

[142] P. Pertilä, M. Mieskolainen, and M. S. Hämäläinen, "Passive self-localization of microphones using ambient sounds," in *Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, 2012, pp. 1314–1318.

[143] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[144] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): Part II," *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 108–117, 2006.

[145] N. Kantas, S. S. Singh, and A. Doucet, "Distributed maximum likelihood for simultaneous self-localization and tracking in sensor networks," *IEEE Transactions on Signal Processing*, vol. 60, no. 10, pp. 5038–5047, 2012.

[146] M. Syldatk and F. Gustafsson, "Expectation maximization algorithm for simultaneous tracking and sparse calibration of sensor networks," 2013.

[147] C. Taylor, A. Rahimi, J. Bachrach, H. Shrobe, and A. Grue, "Simultaneous localization, calibration, and tracking in an ad hoc sensor network," in *Proceedings of the 5th International Conference on Information Processing in Sensor Networks*. ACM, 2006, pp. 27–33.

[148] A. Plinge, G. A. Fink, and S. Gannot, "Passive online geometry calibration of acoustic sensor networks," *IEEE Signal Processing Letters*, vol. 24, no. 3, pp. 324–328, 2017.

[149] L. Wang, T.-K. Hon, J. D. Reiss, and A. Cavallaro, "Self-localization of ad-hoc arrays using time difference of arrivals," *IEEE Transactions on Signal Processing*, vol. 64, no. 4, pp. 1018–1033, 2016.

[150] M. Pollefeys and D. Nister, "Direct computation of sound and microphone locations from time-difference-of-arrival data," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2008, pp. 2445–2448.

[151] W. Li, Y. Jia, J. Du, and J. Zhang, "PHD filter for multi-target tracking with glint noise," *Signal Processing*, vol. 94, pp. 48–56, Jan. 2014.

[152] Y. Wang, H. Meng, Y. Liu, and X. Wang, "Collaborative penalized Gaussian mixture PHD tracker for close target tracking," *Signal Processing*, vol. 102, pp. 1–15, Sept. 2014.

[153] Y. Fu, Q. Ling, and Z. Tian, "Distributed sensor allocation for multi-target tracking in wireless sensor networks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 3538–3553, Oct. 2012.

[154] L. Shang, K. Zhao, Z. Cai, D. Gao, and M. Hu, "An energy-efficient collaborative target tracking framework in distributed wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2014, July 2014.

[155] D.P. Eickstedt and M.R. Benjamin, "Cooperative target tracking in a distributed autonomous sensor network," in *OCEANS 2006*, pp. 1–6.

[156] V. Zetterberg, M.I Pettersson, L. Tegborg, and I. Claesson, "Passive scattered array positioning method for underwater acoustic source," in *OCEANS 2006*, pp. 1–6.

[157] V. Zetterberg, M.I Pettersson, and I Claesson, "Comparison between whitened generalized cross correlation and adaptive filter for time delay estimation with scattered arrays for passive positioning of moving targets in baltic sea shallow waters," in *Proceedings of MTS/IEEE OCEANS, 2005*, pp. 2356–2361 Vol. 3.

[158] Y. Li, Z. Wang, S. Zhuo, J. Shen, S. Cai, M. Bao, and D. Feng, "The design and implement of acoustic array sensor network platform for online multi-target tracking," in *IEEE 8th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, May 2012, pp. 323–328.

[159] AM. Ali, R.E. Hudson, and K. Yao, "Tracking of acoustic sources using random set theory," *IEEE Systems Journal*, vol. 8, no. 1, pp. 151–159, Mar. 2014.

[160] X. Zhong and A.B. Premkumar, "A random finite set approach for joint detection and tracking of multiple wideband sources using a distributed acoustic vector sensor array," in *15th International Conference on Information Fusion (FUSION)*, 2012, pp. 519–526.

[161] X. Zhong and A. B. Premkumar, "Multiple wideband source detection and tracking using a distributed acoustic vector sensor array: A random finite set approach," *Signal Processing*, vol. 94, pp. 583–594, Jan. 2014.

[162] S.P. Ebenezer and A Papandreou-Suppappola, "Multiple transition mode multiple target track-before-detect with partitioned sampling," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 8008–8012.

[163] C. Yardim, P. Gerstoft, and W. S. Hodgkiss, "Geoacoustic and source tracking using particle filtering: Experimental results," *The Journal of the Acoustical Society of America*, vol. 128, no. 1, pp. 75–87, 2010.

[164] K. W. Chung, A. Sutin, A. Sedunov, and M. Bruno, "Demon acoustic ship signature measurements in an urban harbor," *Advances in Acoustics and Vibration*, 2011.

[165] A. Sutin, B. Bunin, A. Sedunov, N. Sedunov, L. Fillinger, M. Tsionskiy, and M. Bruno, "Stevens passive acoustic system for underwater surveillance," in *International Waterside Security Conference (WSS)*, 2010, pp. 1–6.

[166] S. Gannot and T. G. Dvorkind, "Microphone array speaker localizers using spatial-temporal information," *EURASIP Journal on Advances in Signal Processing*, vol. 2006, no. 1, pp. 1–17, May 2006.

[167] S. S. Blackman, "Multiple-target tracking with radar applications," *Dedham, MA, Artech House, Inc., 1986, 463 p.*, 1986.

[168] Y. Bar-Shalom, "Multitarget-multisensor tracking: advanced applications," *Norwood, MA, Artech House, 1990, 391 p.*, 1990.

[169] C. Gui and P. Mohapatra, "Power conservation and quality of surveillance in target tracking sensor networks," in *Proceedings of the 10th annual international conference on Mobile computing and networking.* ACM, 2004, pp. 129–143.

[170] M. E. Liggins, C.-Y. Chong, I. Kadar, M. G. Alford, V. Vannicola, and S. Thomopoulos, "Distributed fusion architectures and algorithms for target tracking," *Proceedings of the IEEE*, vol. 85, no. 1, pp. 95–107, 1997.

[171] M. R. Azimi-Sadjadi, Y. Jiang, and G. Wichern, "Properties of randomly distributed sparse acoustic sensors for ground vehicle tracking and localization," May 2006.

[172] A. Brutti and F. Nesta, "Tracking of multidimensional TDOA for multiple sources with distributed microphone pairs," *Computer Speech & Language*, vol. 27, no. 3, pp. 660–682, May 2013.

[173] W.-P. Chen, J. C. Hou, and L. Sha, "Dynamic clustering for acoustic target tracking in wireless sensor networks," *IEEE transactions on mobile computing*, vol. 3, no. 3, pp. 258–271, 2004.

[174] X. Sheng, Y.-H. Hu, and P. Ramanathan, "Distributed particle filter with gmm approximation for multiple targets localization and tracking in wireless sensor network," in *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on.* IEEE, 2005, pp. 181–188.

[175] L. E. Parker, B. Birch, and C. Reardon, "Indoor target intercept using an acoustic sensor network and dual wavefront path planning," in *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on.* IEEE, 2003, vol. 1, pp. 278–283.

[176] F. Talantzis, A. Pnevmatikakis, and A. G. Constantinides, "Audio–visual active speaker tracking in cluttered indoors environments," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 38, no. 3, pp. 799–807, 2008.

[177] C. Evers, A. H. Moore, P. A. Naylor, J. Sheaffer, and B. Rafaely, "Bearing-only acoustic tracking of moving speakers for robot audition," in *Digital Signal Processing (DSP), 2015 IEEE International Conference on.* IEEE, 2015, pp. 1206–1210.

[178] X. Zhong, A.B. Premkumar, and A.S. Madhukumar, "Particle filtering for acoustic source tracking in impulsive noise with alpha-stable process," *IEEE Sensors Journal*, vol. 13, no. 2, pp. 589 –600, Feb. 2013.

[179] L. D. Stone, R. L. Streit, T. L. Corwin, and K. L. Bell, *Bayesian multiple target tracking*, Artech House, 2013.

[180] S. Wang and Y. Zhao, "Almost sure convergence of Titterington's recursive estimator for mixture models," *Statistics & Probability Letters*, vol. 76, no. 18, pp. 2001–2006, Dec. 2006.

[181] B. Delyon, "General results on the convergence of stochastic algorithms," *IEEE Transactions on Automatic Control*, vol. 41, no. 9, pp. 1245–1255, 1996.

[182] B. Delyon, M. Lavielle, and E. Moulines, "Convergence of a Stochastic Approximation Version of the EM Algorithm," *The Annals of Statistics*, vol. 27, no. 1, pp. 94–128, Feb. 1999.

[183] P.-J. Chung and J.F. Bohme, "Recursive EM and SAGE-inspired algorithms with application to DOA estimation," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2664–2677, Aug. 2005.

[184] O. Cappé and E. Moulines, "On-line expectation maximization algorithm for latent data models," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 71, no. 3, pp. 593–613, 2009.

[185] T. Caljon, V. Enescu, P. Schelkens, and H. Sahli, "An offline bidirectional tracking scheme," in *ACIVS*. Springer, 2005, pp. 587–594.

[186] A. Brutti, M. Omologo, and P. Svaizer, "Maximum a posteriori trajectory estimation for acoustic source tracking," in *Acoustic Signal Enhancement; Proceedings of IWAENC 2012; International Workshop on*. VDE, 2012, pp. 1–4.

[187] B. D. van Veen and K. M. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE Acoustics, Speech and Signal Magazine*, vol. 5, no. 2, pp. 4–24, Apr. 1988.

[188] P. A. Naylor and N. D. Gaubitch, Eds., *Speech Dereverberation*, Springer, 2010.

[189] C. Evers, A. H. Moore, and P. A. Naylor, "Acoustic simultaneous localization and mapping (a-SLAM) of a moving microphone array and its surrounding speakers," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, Shanghai, China, Mar. 2016.

[190] C. Evers, A. H. Moore, and P. A. Naylor, "Localization of moving microphone arrays from moving sound sources for robot audition," in *Proc. European Signal Processing Conf. (EUSIPCO)*, Budapest, Hungary, 2016.

[191] C. Evers, A. H. Moore, and P. A. Naylor, "Towards informative path planning for acoustic simultaneous localization of microphone arrays and mapping of surrounding sound sources (a-SLAM)," in *DAGA*, Aachen, Germany, mar 2016.

[192] Y. Dorfan, G. Hazan, and S. Gannot, "Multiple acoustic sources localization using distributed expectation-maximization algorithm," in *the 4th Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA)*, May 2014, pp. 72–76.

[193] Y. Dorfan and S. Gannot, "Tree-based recursive expectation-maximization algorithm for localization of acoustic sources," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 10, pp. 1692–1703, 2015.

[194] Y. Dorfan, C. Evers, S. Gannot, and P. A. Naylor, "Speaker localization with moving microphone arrays," in *Proc. European Signal Processing Conf. (EUSIPCO)*, Budapest, Hungary, Aug. 2016.

[195] Y. Dorfan, A. Plinge, Hazan G., and S. Gannot, "Distributed expectation-maximization algorithm for speaker localization in reverberant environments," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 3, pp. 682–695, 2018.

[196] O. Schwartz, Y. Dorfan, E.A.P. Habets, and S. Gannot, "Multiple DOA estimation in reverberant conditions using EM," in *International Workshop for Acoustic Echo Cancellation and Noise Control (IWAENC), Xi'an, China*, 2016.

[197] Y. Dorfan, D. Cherkassky, and S. Gannot, "Speaker localization and separation using incremental distributed expectation-maximization," in *European Signal Processing Conference (EUSIPCO)*, 2015, pp. 1256–1260.

[198] Y. Dorfan, O. Schwartz, B. Schwartz, E. A. P. Habets, and S. Gannot, "Multiple DOA estimation and blind source separation using estimation-maximization," in *IEEE Science of Electrical Engineering (ICSEE)*, Eilat, Israel, 2016.

[199] Y. Dorfan, O. Schwartz, and S. Gannot, "Joint speaker localization and array calibration using expectation-maximization," *To be submitted to IEEE...*

[200] Y. Dorfan, B. Schwartz, and S. Gannot, "Speaker tracking using forward-backward recursive expectation-maximization," *To be submitted to IEEE...*

[201] C. Evers, Y. Dorfan, S. Gannot, and P. A. Naylor, "Source tracking using moving microphone arrays for robot audition," *IEEE International Conference on Audio and Acoustic Signal Processing (ICASSP), New-Orleans, LA, USA*, 2017.

[202] S. Rickard and O. Yilmaz, "On the approximate w-disjoint orthogonality of speech," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2002, vol. 1, pp. 529–532.

[203] F. Jacob, J. Schmalenstroeer, and R. Haeb-Umbach, "DOA-based microphone array postion self-calibration using circular statistics," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 116–120.

[204] A. Canclini, F. Antonacci, A. Sarti, and S. Tubaro, "Acoustic source localization with distributed asynchronous microphone networks," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 2, pp. 439–443, 2013.

[205] G. Lee and C. Scott, "Em algorithms for multivariate gaussian mixture models with truncated and censored data," *Computational Statistics & Data Analysis*, vol. 56, no. 9, pp. 2816–2829, 2012.

[206] N. D. Degan and C. Prati, "Acoustic noise analysis and speech enhancement techniques for mobile radio applications," *Signal Processing*, vol. 15, no. 1, pp. 43–56, 1988.

[207] E. A. P. Habets and S. Gannot, "Generating sensor signals in isotropic noise fields," *The Journal of the Acoustical Society of America*, vol. 122, pp. 3464–3470, Dec. 2007.

[208] N. A. Lynch, *Distributed Algorithms*, Morgan Kaufmann, Apr. 1996.

[209] D. Cherkassky and S. Gannot, "Blind synchronization in wireless sensor networks with application to speech enhancement," in *Proceedings of the Acoustic Signal Enhancement (IWAENC)*, 2014, pp. 183–187.

[210] E. A. P. Habets, "Room impulse response (RIR) generator," http://www.audiolabs-erlangen.de/fau/professor/habets/software/rir-generator, Sept. 2010.

[211] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.

[212] E. A. Lehmann and R. C. Williamson, "Importance sampling particle filter for robust acoustic source localisation and tracking in reverberant environments," in *Proc. Joint Workshop on Hands-Free Speech Communication and Microphone Arrays*, New Jersey, USA, Mar. 2005, vol. C, pp. 7–8.

[213] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*, Statistics for Engineering and Information Science. Springer, New York, 2001.

[214] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 174–188, 2002.

[215] M. Unoki and M. Akagi, "A method of signal extraction from noisy signal based on auditory scene analysis," vol. 27, no. 3, pp. 261–279, 1999.

[216] B.R. Glasberg and B.C.J. Moore, "Derivation of auditory filter shapes from notched-noise data," *Hearing Research*, vol. 47, no. 1, pp. 103–138, August 1990.

[217] D. Cherkassky and S. Gannot, "Blind synchronization in wireless acoustic sensor networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 3, pp. 651–661, 2017.

[218] A. Bertrand, "Applications and trends in wireless acoustic sensor networks: A signal processing perspective," in *18th IEEE Symposium on Communications and Vehicular Technology in the Benelux (SCVT)*. IEEE, 2011, pp. 1–6.

[219] A. Nádas, 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, 1989.

[220] D. Burshtein and S. Gannot, "Speech enhancement using a mixture-maximum model," *IEEE transactions on speech and audio processing*, vol. 10, no. 6, pp. 341–351, 2002.

[221] E. Hadad, F. Heese, P. Vary, and S. Gannot, "Multichannel audio database in various acoustic environments," in *International Workshop on Acoustic Signal Enhancement (IWAENC), Aachen, Germany, Sep.*, 2014, pp. 313–317.

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

[223] S. Markovich, S. Gannot, and I. Cohen, "Multichannel eigenspace beamforming in a reverberant noisy environment with multiple interfering speech signals," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 6, pp. 1071–1086, 2009.

[224] O. Schwartz, S. Gannot, and E. A. P. Habets, "An expectation-maximization algorithm for multimicrophone speech dereverberation and noise reduction with coherence matrix estimation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 9, pp. 1495–1510, 2016.

[225] R.M. Neal and G.E. Hinton, "A view of the EM algorithm that justifies incremental, sparse, and other variants," *Learning in graphical models*, vol. 89, pp. 355–368, 1998.

[226] S.-K. Ng and G. J. McLachlan, "On the choice of the number of blocks with the incremental em algorithm for the fitting of normal mixtures," *Statistics and Computing*, vol. 13, no. 1, pp. 45–55, 2003.

[227] L. Frenkel and M. Feder, "Recursive expectation-maximization (EM) algorithms for time-varying parameters with applications to multiple target tracking," *IEEE Transactions on Signal Processing*, vol. 47, no. 2, pp. 306–320, 1999.

[228] R. A. Fisher, "Theory of statistical estimation," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 22, pp. 700–725, 1925.

[229] H. J. Kushner and G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*, Springer, second edition, July 2003.

[230] A. Benveniste, M. Metivier, and P. Priouret, *Adaptive Algorithms and Stochastic Approximations*, vol. 22, Springer Science & Business Media, 2012.

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

[232] G.D. Forney Jr, "The Viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, 1973.

[233] E. A. Lehmann and R. C. Williamson, "Particle filter design using importance sampling for acoustic source localisation and tracking in reverberant environments," *EURASIP Journal on Advances in Signal Processing*, vol. 2006, no. 1, pp. 1–9, 2006.

[234] H. Sawada, R. Mukai, and S. Makino, "Direction of arrival estimation for multiple source signals using independent component analysis," in *Proc. Intl. Symposium on Signal Processing and Its Applications*, July 2003, vol. 2, pp. 411–414.

[235] A. Doucet, S. Godsill, and C. Andrieu, "On sequential monte carlo sampling methods for bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.

תקציר

איכון, הפרדה ומעקב של מקורות אקוסטיים הם אתגרים שחיות ובני אדם רבים עושים אינטואיטיבית ולעיתים בדיוק מרשים. שיטות מלאכותיות פותחו למגוון יישומים ותנאים. רובן ריכוזיות, כלומר: כל האותות מעובדים יחד להפיק את תוצאות השערוך.

הקונספט של רשתות חיישנים מבוזרים הופך יותר ממשי עם התקדמויות טכנולוגיות בשדות של ננו-טכנולוגיה, MEMS ותקשורת. רשת חיישנים מבוזרת מורכבת מצמתים עצמאיים, בעלי מקור הספק, חיישנים, מעבד ותקשורת. מגוון טופולוגיות קישוריות משמשים רשתות כאלו. ישנם יישומים בתחומים רבים כמו אקולוגיה, צבא, ניתור איכות סביבה, רפואה, בטחון ועוד.

בתיזה זו אנחנו מפתחים אלגוריתמים עבור רשתות חיישנים מבוזרים ליישומים שקשורים לעיבוד דיבור, אבל חלק מהשיטות יכולות להיות מופעלות גם ליישומים אחרים. הרשתות המבוזרות האקוסטיות יכולות לשמש בהרבה תרחישים מודרניים. הדוגמה הראשונה היא סביבה רוויות תקשורת. כל אחד נושא עמו מיקרופונים אישיים כחלק מטלפון חכם, מחשב נייד או מכשיר אחר. החיישנים המבוזרים הללו מאפשרים חיפוש מידע מרחבי בנוסף למידע זמן-תדר. החיישנים הללו מאפשרים הקמת רשת מיקרופונים מבוזרת אד הוק ויישום של אלגוריתמי עיבוד אותות מתוחכמים בלי הצורך להתקין מערכות שמע יקרות. דוגמה שניה היא בתים חכמים שהפכו מאד מקובלים בשנים האחרונות. רשתות חכמות של מיקרופונים הם מרכיב הכרחי עבור מערכות בקרה, ניתור ותקשורת חירום. הדוגמה האחרונה שנציין כאן הוא אכיפת החוק. רשויות אכיפה כמו משטרה או צבא נעזרים בהאזנות סתר וניתור שמע של מקומות ציבוריים אזרחיים כחלק מהעבודה השוטפת. זה בדר כלל נעשה בתנאים מגוונים.

הזמינות של מידע חלקי בצמתים, הדינמיות של הרשת והמגבלות על ההספק ועל ההתקשורת מצריכים פיתוח אלגוריתמים חדשניים כדי לתת מענה לאתגרים הללו. האתגרים שמנינו הם טיפוסיים לעיבוד מבוזר ולא ניתן למצוא להם מענה באלגוריתמי עיבוד מרחבי קלאסי של מערכים.

התרומה של תיזה זו היא מחומשת. ראשית, אלגוריתמי איכון מבוזר שמפותחים על ידי סט חדשני של משתנים חבויים שמשוערכים על ידי מערכי מיקרופונים נייחים ונעים. מסתבר שבנוסף לביזור החישוב, המשתנים החדשים משפרים את מהירות ההתכנסות והדיוק בהשוואה לאלגוריתמים קודמים, מכיוון שהם מאפשרים שימוש בעיקרון של EM מדורג במרחב המיקום. אלגוריתמי האיכון המבוזר שפותחו מכסים את ה EM הבלוקי והרקורסיבי, שמתאים לזמן אמת.

שנית, פתחנו מספר שיטות איכון להפחתה משמעותית של השפעת ההדהוד על הביצועים. אנחנו מראים שעיבוד שמדגיש את המסלול הישיר משולב עם שיטות האיכון שלנו שעברו שדרוג משפר את התוצאות במיוחד כשמספר הדוברים בו-זמנית בחדר גדל. על מנת לחזק את הצומת עם זוג מיקרופונים, הראנו שניתן להשתמש במקום בהפרשי פאזות בדגימות עצמן של אות הדיבור עם סידור גיאומטרי כלשהו של

המיקרופונים. דגימות אלו ניתנות לעיבוד במסגרת מודל חדש שלוקח בחשבון בנוסף למסלול הישיר גם את הזנב של ההדהוד המאוחר.

שלישית, גילינו שתוצאות האלגורתמי איכון המוזכרים ניתנות לניצול גם לצורך הפרדה עיוורת של מקורות. תרומה משמעותית לאלכוריתמי הפרדה היא המשתנים הנסתרים שמשמשים את ה EM. הם הוכחו כמסכות תדר מאד יעילות, בגלל שהמשמעות הפיזיקלית היא שיוך פסי זמן-תדר לדוברים שונים.

רביעית, אתגר מרכזי עם רשתות אד הוק הוא שהמיקומים היחסיים של המערכים לא ידועים. אנחנו מציעים פתרון עבור כיול המערכים ואיכון המקורות במשולב. כולם משוערכים יחסית למערך אחד שמשמש עוגן.

לבסוף, אנחנו מטפלים בבעיות דינמיות. מעקב מבוזר המבוסס על אלגוריתם ה RDEM מתואר תחילה עבור מערכים נייחים. מעקב אחרי מספר דוברים בו זמנית מאד מאתגר, בגלל שהאותות ותגובת החדר משתנים באופן מורכב. למשל, הדוברים לא מפיקים דיבור באופן רציף, אבל יכולים לנוע ברציפות. זה אומר שיש פערי זמן שצריך למלא. דרך אפשרית לטפל בפערים הללו הוא ניצול מידע עתידי לגבי הדוברים. השהייה קצרה מאפשרת הוספת עיבוד אנטי-סיבתי לאלגוריתם המעקב הלא-ביסיאני הקלאסי. בעיה נוספת (יותר מעשית בעצם) היא איכון ומעקב דוברים על ידי מערכים נעים. התנועה של זוג מיקרופונים מנוצלת לאיכון ומעקב דוברים על ידי שיטות ביסיאניות ולא-ביסיאניות.

תוכן עניינים

תודות

ברצוני להביע את תודתי והערכתי העמוקה למנחה שלי, פרופ׳ שרון גנות על הדרכה והנחייה עם מסירות יוצאת דופן. תודה על התמיכה המקצועית שלך, על העידוד למצוינות ועל הרבה עצות מועילות לכל אורך שלבי המחקר. המחויבות שלך לחנך חוקרים חדשים היא ייחודית ומרשימה.

עוד אדם מאד מנוסה שעזר לי לאורך כל הדרך הוא ד״ר גרשון חזן. תודה לך על דיונים פוריים ועל כל שאר הדברים שעשית למעני. היה לי לעונג לעבוד איתך.

תודות לאמי ואבי, מיה ומני דורפן עבור מעורבות פעילה במחקר מעבר לתמיכה הרגילה (כולל הקלטות במעבדה). תודות לחמותי וחמי התומכים, מרסל ואשר סרוסי זיכרונם לברכה. מבחינתם הייתי אמור לסיים כבר מזמן (משפט זה נכתב עוד בהיותם בחיים. עכשיו יש לו משמעות נוספת, עמוקה יותר). תודה מיוחדת לאשתי האהובה, איריס ולארבעת ילדינו המדהימים : חוף, פלג, גל ולי-ים אשר עודדו ותמכו לאורך כל הדרך.

עבודה זו נעשתה בהדרכתו של:

פרופ' שרון גנות, הפקולטה להנדסה, אוניברסיטת בר-אילן

איכון ומעקב מבוזרים
של מקורות אקוסטים

חיבור לשם קבלת התואר ״דוקטור לפילוסופיה״

מאת:

יובל דורפן

הפקולטה להנדסה

הוגש לסנט של אוניברסיטת בר-אילן

רמת גן אדר, תשע״ח