Politecnico di Milano
V Facoltà di Ingegneria
Corso di laurea in Ingegneria delle Telecomunicazioni
Dipartimento di Elettronica e Informazione

A Geometric Approach to Localization of Acoustic Sources and Reflectors

Relatore: Prof. Augusto SARTI
Correlatore: Dr. Fabio ANTONACCI

Tesi di Laurea di:
Paolo BESTAGINI Matr. 733451

Anno Accademico 2009-2010

Abstract

Space-time audio processing has assumed great significance and importance for many applications such as source localization, tracking, beam-shaping, and many more. In this thesis we want to propose a different approach to acoustic source localization on a plane by measuring time differences of arrival (TDOA). We base our approach on a frame which uses three coordinates instead of only two, also for 2D localization. We could say that the third coordinate is to be interpreted as time or, vice-versa, that the three-coordinate system has a similar structure to projective geometry coordinates, where scaling factor now matters. In practice each point of this space identifies a point on the plane where source lies, in a specific time instant. In this framework we represent the signal emitted by the source as a cone the vertex coinciding with the source position, or a circle projected in the third dimension increasing its radius as third coordinate increases. In the 3D space, each pair of TDOA measurement and microphone position represents a point through which the cone is supposed to pass. Localization problem is thus turned into a cone fitting problem. The above cone fitting method can also be applied to multi-source case. In regard to this, a particular case is given when image sources are created by reflections on walls. Therefore, we show how to apply our localization method also to image sources. In doing so we can infer a reflector position based on real and image sources locations. For all the above problems, we provide simulations and experimental results as well as comparisons with other techniques proposed in the literature. These comparisons show that the proposed algorithms perform at least as well as the best ones we selected, but using the projective space and dealing with conics allows us to write constraints in a compact notation, more suitable when handling many constraints. One of our algorithms is also more robust when used with noisy measurements.

Estratto in Lingua Italiana

Lo space-time audio processing ha assunto notevole importanza per molte applicazioni come la localizzazione di sorgente, il tracking, il beam-shaping, e altre. In questa tesi vogliamo mostrare un approccio differente per la localizzazione di una sorgente acustica su un piano da misure di differenze di tempi d'arrivo (TDOA). Basiamo il nostro approccio su un sistema di coordinate che ne usa tre al posto di due, nonostante localizzi nel 2D. Diciamo che la terza coordinata va interpretata come tempo o, viceversa, che il sistema di tre coordinate ha una struttura simile a quelle della geometria proiettiva, in cui il fattore di scala ora conta. In pratica ogni punto di questo spazio identifica un punto del piano su cui giace la sorgente in uno specifico istante di tempo. In questo scenario rappresentiamo il segnale emesso come un cono col vertice nella posizione della sorgente, o un cerchio proiettato nella terza dimensione col raggio che aumenta all'aumentare della terza coordinata. Nello spazio 3D, ogni coppia di misure di TDOA e di posizione di un microfono rappresenta un punto attraverso il quale il cono dovrebbe passare. Il problema della localizzazione si trasforma dunque nel problema di ricerca di un cono passante per dei punti. Questo metodo di ricerca del cono può essere utilizzato anche nel caso di sorgenti multiple. Un caso particolare si ha quando delle sorgenti immagine sono create dalla presenza di riflettori. In questo caso, localizzando le sorgenti reali e immagine possiamo inferire anche la posizione di un riflettore. Per i problemi citati forniamo simulazioni e risultati sperimentali oltre ad un confronto con tecniche trovate in letteratura. Il confronto mostra che gli algoritmi proposti si comportano bene almeno quanto i migliori tra quelli testati, ma l'uso dello spazio proiettivo e l'avere a che fare con coniche permette di scrivere i vincoli in maniera compatta, preferibile quando si utilizzano molti vincoli. Uno dei nostri algoritmi è anche più robusto degli altri in presenza di misure rumorose.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Space-time audio processing has assumed great significance and importance for many applications such as localization, automatic camera tracking for video-conferencing, separation of acoustic sources, beamformer steering for suppressing noise and reverberation, beam-shaping, and many more. Many of these applications are based on the exploitation of different kind of measurements such as direction of arrival (DOA), time of arrival (TOA), or time difference of arrival (TDOA).

In particular, to locate radiative point sources using passive stationary sensor arrays is of considerable interest and has been a repeated theme of research in radar, underwater sonar, and seismology. A common method is to have the estimate of source location based on TDOA measurements between distinct sensor pairs, or on TOA measurements when sources and sensors are synchronized.

The aim of this thesis is to show an approach to 2D acoustic source localization based on the idea of working in a three coordinates space with TDOAs measurements. In this framework we use the two spatial coordinates used for locating objects on the plane, as well as a third coordinate related to the signal propagation time. We also focus on how to extend this method to the multi-source case. This allows us to perform real and image sources localization when a reflector is present, therefore we can also infer the reflector position. Proposed methods make use of TDOAs, even though shown algorithms can be also applied to TOAs measurements.

In the literature many authors have dealt with these problem with different approaches. A comprehensive collection of source localization algorithms based

on TDOAs can be found in [1], but there are also other papers facing the source localization problem in closed-form solution [2], or with an iterative method based on Taylor series [3], or explicitly giving a solution based on hyperbola intersection [4]. The common approach shared by these methods consists in working with only the two spatial coordinates in order to locating the source in 2D.

However, over the last few years, the technological progress allowed to reduce the size and the cost of the components. Microphones and loudspeakers are now more affordable, and we think that in the near future they will be more extensively used in commercial products. For example, we imagine entertainment systems like television sets or hi-fi systems equipped with sets of miniaturized microphones or speakers or other sensors in order to take advantage of audio techniques such as those based on room awareness to improve auditory experience.

In order to jointly make use of all the constraints given by the increasing number of sensors and managing all of them with a compact notation, a novel approach to localization problem is needed. In this direction, some really interesting solutions are those proposed in [5, 6, 7], as they add a third coordinate to the reference frame. In particular [5, 6] show algorithms which make use of three homogeneous coordinates, extensively used in computer vision. These methods enable us to treat constraints given by measurements as conics and quadrics, leading us to a compact notation which is compatible with constraints from TDOA, TOA and DOA. The interesting aspect of [7] is that it works taking into account spatial coordinates together with propagation time, giving thus a physical meaning to the third coordinate.

Our goal is to propose a solution that takes advantage of both such techniques. We work in a 3D framework as we think that increasing the dimensionality of the reference frame allows us to treat some problem easily, for example turning many nonlinear problems into linear ones. However we also give a physical interpretation to the third coordinate as signal propagation time, which makes constraints more understandable. In this 3D scenario, each point represents a location on the 2D surface where source lies, in a specific time instant. The signal emitted by the source is represented by a cone, the vertex coinciding with the source position. As time goes by, the radius of circle increases. In this 3D space, TDOAs measurements and microphones positions are turned into points through which the cone is supposed to pass. Localization problem is thus turned into a cone fitting

problem where finding the vertex of the cone means finding the source location. Working with a cone in a 3D space allows us to easily take degenerate geometries into account. In fact we can study microphones displacement in order to provide a good cone sampling, which ensures better localization performance.

As far as multi-source localization problem is concerned, various approaches are proposed in the literature. For example in [8] and [9] an algorithm based on blind source separation is proposed, in [10] blind channel identification is used, while in [11] authors make use of clustering techniques. A more interesting approach for our purpose, is that based on correctly assigning TDOAs to each source in order to treat the multi-source localization problem one source at a time [12, 13, 14]. These methods practically consist in exploiting some TDOA properties so as to understand which TDOA belongs to which source. This allows us to split the multi-source localization problem into several single-source localization ones. In doing so we can apply our cone-based single-source localization algorithms also to the multi-source case.

A particular multi-source scenario is given when image sources are created by signal reflection on walls. In fact, when a reflector is present in the scene, microphones receive a signal as if it was emitted by the real source and an image source created by the reflective path. In this particular case we can perform real and image source localization assigning TDOAs to each of them using once more the algorithms for TDOAs assignment as [12, 13, 14]. After the real and image sources are located, we can also infer the reflector position with some simple considerations on sources locations. Nonetheless other solutions for reflector localization problem can be found in [15] and [16]. In these papers authors focus on methods using all the TDOAs together, in order to localize reflectors, without addressing source localization problem.

In brief we sum up the three problems treated in this thesis as: single-source localization in 2D; multi-source localization always in 2D; and reflector localization in the same framework. All the methods we propose for solving this problems are based on the cone idea in order to work in the 3D space, and are provided with simulations or experimental results. We have also made a comparison between some of the above cited techniques found in the literature and our algorithms, which confirms good behavior of our algorithms, especially with noisy measurements.

In Chapter 2, we focus our attention on giving the problem formulation and presenting the notation used. Giving the problem formulation we also introduce the state of the art for all the three problems, except for those techniques which are based on homogeneous coordinates.

The methods employed so as to localize sources and reflectors making use of homogeneous coordinate are discussed in Chapter 3. In this Chapter we also present in a comprehensive way our novel 3D coordinate system, and show how TOAs and TDOAs measurements can be turned into constraints in the extended geometry space, using homogeneous coordinates as described in the literature, as well as with our novel approach.

After presenting the current state of the art and our new framework, in Chapter 4 we show how proposed algorithms, making use of two similar cost functions, work for source localization. In this Chapter we focus on how to use previously shown constraints in order to perform source localization, emphasizing also the study of favorable microphones displacements and degenerate geometries.

In Chapter 5 we then explain how to use the above proposed localization methods for the multi-source case. In doing so, we deeply explain how to correctly assign TDOAs to each source before applying a cone-based algorithm.

After the multi-source general case is introduced, in Chapter 6 we extend the algorithm in order to localize an image source generated by a reflective path. We show how to find real and image sources positions and how to infer reflector position from the sources ones, after which (Chapter 7) we gather all simulations and experimental results in order to prove the effectiveness of our algorithms in contrast to the others found in the literature.

Finally, in Chapter 8 we draw conclusions about the study and show possible future works.

Chapter 2

Problem Formulation

In this Chapter we introduce the mathematical formulation of source and reflector localization problems. We also show the current state of the art in order to understand some possible approaches to these problems. The notation used in the rest of the thesis is also given.

Source localization problem is approached when no obstacles are present within the scene, and only information from the direct signal is available. First we discuss single source case, then we introduce also the multi-source case. Finally we handle reflector localization problem when only one reflector is present, and information about direct and indirect signal are available.

2.1 Acoustic Measurements: Time of Arrival, Time Difference of Arrival

In this Section we will refer about the methodology we use to conduct acoustic measurements in a dry room. In particular, we will relate about the measurement of times of arrival (TOAs) and time differences of arrival (TDOAs) from cross-correlation.

In regard to TOAs, the goal is to measure the time of arrival of the acoustic paths that link the source to the i-th microphone. In order to do so, the loudspeaker produces a known sequence $s(t)$ (time-continuous). The time-continuous signal $x_i(t)$ acquired by sensor i synchronized with source is the delayed replica

of the signal $s(t)$:

$$x_i(t) = hs(t - \tau_i) + v_i(t). \tag{2.1}$$

The coefficient h is the attenuation of the direct path from source to receiver; τ_i is the corresponding delay; finally $v_i(t)$ is an additive noise that alters our measurement. After sampling by the A/D converter we write the time-discrete signal as

$$x_i(k) = x_i(kT) = hs(k - i_i) + v_i(k), \tag{2.2}$$

where T is the sampling period; k is the time-discrete index; i_i is the discrete version of τ_i. For the sake of simplicity TOAs are estimated by picking peaks in the cross-correlations of $x_i(k)$ and $s_(k)$. In particular the lag i_i is the location of the first relevant peak in the cross-correlation.

When measuring TDOAs we remove the hypothesis that sources are synchronized with receivers, while synchronism among microphones still holds. This scenario accounts for situations in which an arbitrary source (e.g. a person uttering a sentence, a ring of a mobile phone, etc.) is active in the environment. Obviously this is a more interesting case because it is more realistic to have a source not synchronized with sensors. Even if we cannot measure the time of arrival, many cues about the source position can still be extracted from the joint knowledge of the synchronized signals available at sensors. After time sampling, the signal acquired by sensor i in the array is

$$x_i(k) = hs(k - i_i) + v_i(k). \tag{2.3}$$

The model in the above equation is equivalent to the model in equation 2.2. Under the assumption that the attenuation h does not depend on the microphone index and that additive noises at different microphones are uncorrelated, the cross-correlation of $x_i(k)$ and $x_j(k)$ gives

$$r_{ij}(k) = h^2 s(k - i_i) \otimes s(k - i_j), \tag{2.4}$$

where $\otimes$ denotes the cross-correlation operator. If the source signal is a white noise, we obtain

$$r_{ij}(k) = h^2 \delta[k - (i_i - i_j)], \tag{2.5}$$

which exhibits a global maximum for $k = i_i - i_j$. For this reason we can extract

TDOAs from cross-correlations searching for the global maximum locations.

2.2 Acoustic Source Localization

Two dimensional localization problem consists in finding an acoustic source location on a plane, having information about the array geometry, time of arrival, or time difference of arrival from source to the array.

We consider an array of $N+1$ microphones located on a plane at positions

$$\boldsymbol{r}_i \triangleq [x_i, y_i]^T \quad , \quad i = 0, ..., N \tag{2.6}$$

in Cartesian coordinate, where $[.]^T$ denotes transpose of a vector. The first microphone ($i = 0$) is regarded as the reference and is placed at the origin of the coordinate system ($\boldsymbol{r}_0 = [0, 0]^T$). The acoustic source is located at $\boldsymbol{r}_s \triangleq [x_s, y_s]^T$.

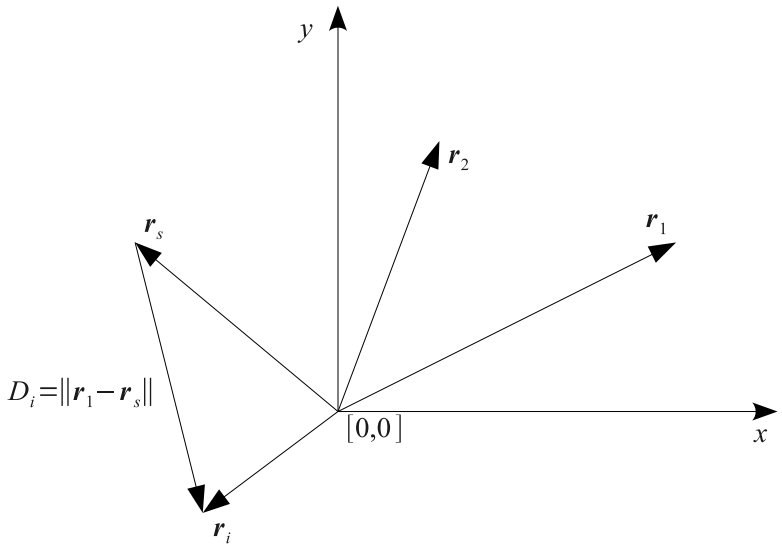


Figure 2.1: Localization Setup: microphones are located at $\boldsymbol{r}_i$, source is located at $\boldsymbol{r}_s$, and D_i represents the distance from the i-th microphone to the source.

The distance from the origin to the i-th microphone and the source are denoted by R_i e R_s, respectively, where

$$R_i \triangleq \|\boldsymbol{r}_i\| = \sqrt{x_i^2 + y_i^2} \quad , \quad i = 1, ..., N, \tag{2.7}$$

$$R_s \triangleq \|\boldsymbol{r}_s\| = \sqrt{x_s^2 + y_s^2}. \tag{2.8}$$

The distance between the source and the i-th microphone is denoted by

$$D_i \triangleq \|\boldsymbol{r}_i - \boldsymbol{r}_s\| = \sqrt{(x_i - x_s)^2 + (y_i - y_s)^2}. \tag{2.9}$$

The difference in the distances of microphones i and j from the source is given by

$$d_{ij} \triangleq D_i - D_j \quad , \quad i, j = 0, ..., N \ . \tag{2.10}$$

The distance D_i and the range difference d_{ij} are proportional to time of arrival (TOA) τ_i and time difference of arrival (TDOA) τ_{ij}, respectively.

If the sound speed is c, then

$$D_i = c \cdot \tau_i, \tag{2.11}$$

$$d_{ij} = c \cdot \tau_{ij}. \tag{2.12}$$

The speed of sound (in m/s) can be estimated from the air temperature t_{air} (in degrees Celsius) according to the following approximate (first-order) formula

$$c \approx 331 + 0.610 \cdot t_{air}. \tag{2.13}$$

The localization problem is then to estimate $\boldsymbol{r}_s$ given the set of $\boldsymbol{r}_i$ and τ_i, or more realistically τ_{ij}, as shown in previous section. Note that having $N+1$ microphones gives us $N+1$ distinct TOA measurements τ_i and $(N+1) \cdot N/2$ distinct TDOA estimates τ_{ij}, which exclude the case $i = j$ and count the $\tau_{ij} = -\tau_{ji}$ pair only once. However, in the absence of noise, the space spanned by these TDOA estimates is N-dimensional. Any N linearly independent TDOAs determine all of the others. In a noisy environment, the TDOA redundancy can be used to improve the accuracy of the source localization algorithms, but this would increase their computational complexity. For simplicity and also without loss of generality, we can choose τ_{i0} , $i = 1, ..., N$ as the basis for this $\mathbb{R}^N$ space.

2.2.1 State of the Art

In this Section we present the state of the art for TDOA and TOA localization problem. However, TOA localization problem is much easier to be solved than TDOA one because TOA knowledge brings more information. For this reason less TOA based localization methods are proposed while it will be given more importance to TDOA localization problem and its state of the art will be better treated. We will show how this problem can be solved with maximum likelihood (ML) or least-squares (LS) criteria, depending on the error probabilistic assumptions, and which are the similarities among the proposed solutions. It will be emphasized how many LS criteria start with the definition of a cost function from a shared equation based on TDOA definition and how these algorithms differ only in the way they treat this equation. In fact some solutions propose different cost functions, while others differ for its minimization method.

2.2.1.1 Localization Based on Time of Arrival

When source localization problem based on TOAs is examined using estimation theory [17], the measurements of distance between the source and the i-th microphone are modeled by

$$D_i = l_i(\boldsymbol{r}_s) + \varepsilon_i \quad , \quad i = 1, ..., N \tag{2.14}$$

where

$$l_i(\boldsymbol{r}_s) = \sqrt{(x_i - x_s)^2 + (y_i - y_s)^2} \tag{2.15}$$

and the ε_i's are measurement errors. The squared distance from the source to i-th microphone is given by

$$\begin{aligned} l_i(\boldsymbol{r}_s)^2 &= (x_i - x_s)^2 + (y_i - y_s)^2 \\ &= K_i - 2x_i x_s - 2y_i y_s + x_s^2 + y_s^2 \end{aligned} \tag{2.16}$$

where $K_i = x_i^2 + y_i^2$ and $x_s^2 + y_s^2 = D_0^2$ if we choose the first microphone as the origin. We can then define the error

$$e_i(\boldsymbol{r}_s) = K_i - 2x_i x_s - 2y_i y_s + x_s^2 + y_s^2 - D_i^2 \tag{2.17}$$

and putting the N error together we obtain

$$\boldsymbol{e} = \boldsymbol{A}\theta - \boldsymbol{b} \tag{2.18}$$

where

$$\boldsymbol{e} = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix} \qquad \boldsymbol{A} = \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_N & y_N \end{bmatrix}$$

$$\boldsymbol{\theta} = \begin{bmatrix} x_s \\ y_s \end{bmatrix} \qquad \boldsymbol{b} = \frac{1}{2} \begin{bmatrix} K_1 - D_1^2 - D_0^2 \\ K_2 - D_2^2 - D_0^2 \\ \vdots \\ K_N - D_N^2 - D_0^2 \end{bmatrix}.$$

If we use a least-squares approach for the error minimization, the LS cost function to be minimized is given by

$$J = \boldsymbol{e}^T \boldsymbol{e} = (\boldsymbol{A}\boldsymbol{\theta} - \boldsymbol{b})^T (\boldsymbol{A}\boldsymbol{\theta} - \boldsymbol{b}) \tag{2.19}$$

and a closed-form solution is given by

$$\hat{\boldsymbol{\theta}} = \left(\boldsymbol{A}^T \boldsymbol{A}\right)^{-1} \boldsymbol{A}^T \boldsymbol{b}. \tag{2.20}$$

2.2.1.2 Localization Based on Time Difference of Arrival

When source localization problem based on TDOAs is examined using estimation theory [1], the measurements of range differences are modeled by

$$d_{i0} = g_i(\boldsymbol{r}_s) + \varepsilon_i \quad , \quad i = 1, ..., N \tag{2.21}$$

where

$$g_i(\boldsymbol{r}_s) = \|\boldsymbol{r}_i - \boldsymbol{r}_s\| - \|\boldsymbol{r}_s\| \tag{2.22}$$

and the ε_i's are measurement errors. In a vector form, the additive measurement error model becomes

$$\boldsymbol{d} = \boldsymbol{g}(\boldsymbol{r}_s) + \boldsymbol{\epsilon} \tag{2.23}$$

where

$$\begin{aligned} \boldsymbol{d} &= [\ d_{10} \quad d_{20} \quad ... \quad d_{N0}\ ]^T, \\ \boldsymbol{g}(\boldsymbol{r}_s) &= [\ g_1(\boldsymbol{r}_s) \quad g_2(\boldsymbol{r}_s) \quad ... \quad g_3(\boldsymbol{r}_s)\ ]^T, \\ \boldsymbol{\epsilon} &= [\ \varepsilon_1 \quad \varepsilon_2 \quad ... \quad \varepsilon_N\ ]^T. \end{aligned}$$

Since the measurement model is highly nonlinear, an efficient estimator that attains the CRLB may not exist or might be impossible to find even if it does exist. For this reason several solutions using different criteria have been proposed.

Maximum Likelihood The maximum likelihood estimator (MLE) [1] is the most popular approach because of the well-proven advantage of asymptotic efficiency for a large sample space. However to apply the maximum likelihood principle the statistical characteristics of the measurements need to be known or properly assumed prior to any processing. From the central limit theorem and also for mathematical simplicity, the measurement error is usually modeled as Gaussian with zero mean and covariance matrix $\boldsymbol{C}_\varepsilon$. Since the exponential function is monotonically increasing, the MLE is equivalent to minimizing a (log-likelihood) cost function defined as

$$\varepsilon_{ML}(\boldsymbol{r}_s) \triangleq [\boldsymbol{d} - \boldsymbol{g}(\boldsymbol{r}_s)]^T \boldsymbol{C}_\varepsilon^{-1} [\boldsymbol{d} - \boldsymbol{g}(\boldsymbol{r}_s)]. \tag{2.24}$$

For this minimization an iterative algorithm such as steepest descent can be used in order to avoid the cost function exhaustive search. However, MLE suffers the limitation given by the probabilistic assumptions to be made about the measured range differences. In order to overcome this problem, many least squares estimators (LSEs) have been proposed in the literature.

Least Squares The LSEs make no probabilistic assumptions about the data and hence can be applied to the source localization problem in which a precise statistical characterization of the data is hard to determine. In the LS approach, we attempt to minimize a squared error function that is zero in the absence of noise and model inaccuracies. Different error functions can be defined for closeness from the assumed (noiseless) signal based on hypothesized parameters to the observed data, and any different error function leads to a different LSE. In the literature two error function definitions are mainly found, named hyperbolic LS error function and spherical LS error function [18], but starting from the definition

of D_i (equation 2.9) other solutions to localization problem are proposed.

Hyperbolic LS Error Function In the source localization problem, an observed range difference d_{i0} defines a hyperbola, which is the locus of points where the difference of the distances to the i-th microphone and the reference one is a constant. All points lying on such a hyperbola are potential source locations and all have the same range difference d_{i0} to the two microphones i and 0. The hyperbolic LS error function is defined as the difference between the observed range difference (d_{i0}) and that generated by a signal model depending upon the unknown parameters ($g_i(\boldsymbol{r}_s)$). Therefore, a sound source that is located by minimizing the hyperbolic LS error criterion has the shortest distance to all hyperbolas associated with different microphone pairs and specified by the estimated range differences.

The definition of this error function is based on the assumption that if the measurements are noiseless, the measured range differences are equal to those generated by a model

$$d_{i0} = \sqrt{(x_i - x_s)^2 + (y_i - y_s)^2} - \sqrt{(x_0 - x_s)^2 + (y_0 - y_s)^2} = g_i(\boldsymbol{r}_s). \tag{2.25}$$

For each microphone we can then define the error

$$e_{h,i}(\boldsymbol{r}_s) \triangleq d_{i0} - g_i(\boldsymbol{r}_s). \tag{2.26}$$

The error function can be expressed in vector form as

$$\boldsymbol{e}_h(\boldsymbol{r}_s) = \boldsymbol{d} - \boldsymbol{g}(\boldsymbol{r}_s) \tag{2.27}$$

and the corresponding LS criterion is given by

$$J_h = \boldsymbol{e}_h^T \boldsymbol{e}_h = [\boldsymbol{d} - \boldsymbol{g}(\boldsymbol{r}_s)]^T [\boldsymbol{d} - \boldsymbol{g}(\boldsymbol{r}_s)]. \tag{2.28}$$

Since J_h is nonlinear, minimizing it leads to a mathematically intractable solution as N gets large. As a result, it is rarely used in practice as is. Methods taking less time in order to lead to a solution are preferred.

Spherical LS Error Function The second LS criterion is based on the errors found in the distances from a hypothesized source location to the microphones. Referring to Figure 2.2, if we draw circles with radius D_i centered at the microphones, they all intersect in one point in the absence of measurement errors. The correct source location is preferably at the intersection of this group

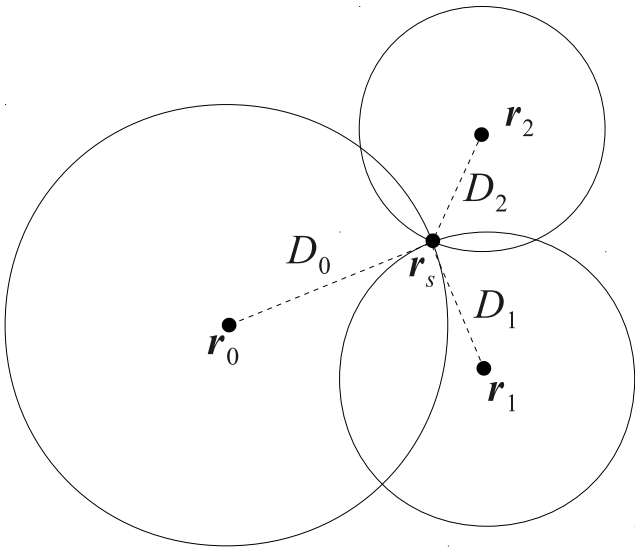


Figure 2.2: Spherical LS Error Function: $\boldsymbol{r}_s$ represents the source position, while $\boldsymbol{r}_0$, $\boldsymbol{r}_1$, and $\boldsymbol{r}_2$ represent microphones. D_0, D_1 and D_2 are the distances from source to microphones. All the circles centered at microphones with radius equal to the distances to the source, intersects in one point. The source is located at this point.

of circles (or spheres in 3D). When measurement errors are present, circles do not intersect in only one point. Therefore the best estimate of the source location would be the point that yields the shortest distance to those circles.

From the definition of the range difference and the fact that $D_0 = R_s$, we have

$$\hat{D}_i = R_s + d_{i0} \tag{2.29}$$

where $\hat{D}_i$ denotes the distance from the source to the i-th microphone based on the measured range difference d_{i0}. From the definition of D_i we can also

write

$$D_i^2 = \|\boldsymbol{r}_i - \boldsymbol{r}_s\|^2 = R_i^2 - 2\boldsymbol{r}_i^T\boldsymbol{r}_s + R_s^2. \tag{2.30}$$

The spherical LS error function is then defined as the difference between the measured and hypothesized values

$$\begin{aligned} e_{sp,i}\left(\boldsymbol{r}_s\right) &= \frac{1}{2}\left(\hat{D}_i^2 - D_i^2\right) \\ &= \boldsymbol{r}_i\boldsymbol{r}_s + d_{i0}R_s - \frac{1}{2}\left(R_i^2 - d_{i0}^2\right), i = 1, ..., N \end{aligned} \tag{2.31}$$

Putting the N errors together and writing them in a vector form gives

$$\boldsymbol{e}_{sp}\left(\boldsymbol{r}_s\right) = \boldsymbol{A}\boldsymbol{\theta} - \boldsymbol{b} \tag{2.32}$$

where

$$\boldsymbol{e}_{sp}(\boldsymbol{r}_s) = \begin{bmatrix} e_{sp,1}(\boldsymbol{r}_s) \\ e_{sp,2}(\boldsymbol{r}_s) \\ \vdots \\ e_{sp,N}(\boldsymbol{r}_s) \end{bmatrix} \qquad \boldsymbol{A} = \begin{bmatrix} x_1 & y_1 & d_{1,0} \\ x_2 & y_2 & d_{2,0} \\ \vdots & \vdots & \vdots \\ x_N & y_N & d_{N,0} \end{bmatrix}$$

$$\boldsymbol{\theta} = \begin{bmatrix} x_s \\ y_s \\ R_s \end{bmatrix} \qquad \boldsymbol{b} = \frac{1}{2}\begin{bmatrix} R_1^2 - d_{1,0}^2 \\ R_2^2 - d_{2,0}^2 \\ \vdots \\ R_N^2 - d_{N,0}^2 \end{bmatrix}.$$

The corresponding LS criterion is then given by

$$J_{sp} = \boldsymbol{e}_{sp}(\boldsymbol{r}_s)^T\boldsymbol{e}_{sp}(\boldsymbol{r}_s) = \left(\boldsymbol{A}\boldsymbol{\theta} - \boldsymbol{b}\right)^T\left(\boldsymbol{A}\boldsymbol{\theta} - \boldsymbol{b}\right). \tag{2.33}$$

A closed-form solution is then given by

$$\hat{\boldsymbol{\theta}} = \left(\boldsymbol{A}^T\boldsymbol{A}\right)^{-1}\boldsymbol{A}^T\boldsymbol{b}. \tag{2.34}$$

The spherical error function is linear in $\boldsymbol{r}_s$, therefore the computational complexity to find a solution will not dramatically increase as N gets large. For this reason the spherical LS error function can be used when a fast solution is needed. Unfortunately, this method is based on the assumption that the distance between

the source and the reference microphone (R_s) was correct for the computation of $\hat{D}_i$. As shown in [18], a better solution can be found.

Linear-Correction Least-Squares In the previous solution we assumed that the distance between the source and the reference microphone (R_s) was correct for the computation of $\hat{D}_i$. The whole error of the term d_{i0} was then "loaded" on the i-th microphone location. In order to avoid this problem, the linear-correction least-squares method [18] was proposed. This algorithm searches for the minimum of the Spherical LS Error Function (J_{sp}) taking account of the constraint $x_s^2 + y_s^2 - R_s^2 = 0$ which forces the distance between the source and the reference microphone to be R_s.

Using this method, the solution is given by

$$\min_{\boldsymbol{\theta}} (\boldsymbol{A\theta} - \boldsymbol{b})^T (\boldsymbol{A\theta} - \boldsymbol{b}) \quad s.t. \quad \boldsymbol{\theta}^T \boldsymbol{\Sigma} \boldsymbol{\theta} = 0 \tag{2.35}$$

where

$$\boldsymbol{\Sigma} = \text{diag}(\ 1 \quad 1 \quad -1\ ). \tag{2.36}$$

Using this correction is equal to constrain the distance between the source and the reference microphone to be R_s. In order to solve this constrained minimization problem the technique of Lagrange multipliers is used and the source location is determined by minimizing the Lagrangian

$$\mathcal{L}(\boldsymbol{\theta}, \lambda) = J_{sp} + \lambda \boldsymbol{\theta}^T \boldsymbol{\Sigma} \boldsymbol{\theta} \tag{2.37}$$

where λ is the Lagrange multiplier. In [18] it is shown that in order to find λ, a polynomial of degree six must be solved. Because of its complexity, numerical methods need to be used for root searching. In order to perform this root searching, a multi-step procedure is proposed.

This algorithm is presented as an improvement of the Spherical LS Error Function one. The negative aspect is given by the complexity of root searching used to find the correct value of λ. This aspect makes this algorithm not suitable for real-time problems or when a fast solution is needed, because of time spent in root searching.

Gillette-Silverman Gillette and Silverman propose another closed-form method based on squared range difference between a reference microphone and the others [2]. Their goal is to find a simple solution to the source localization problem, giving it in closed-form. In order to proceed with this algorithm, we need to define a reference microphone, and without loss of generality we can take the microphone placed at the origin as reference one. As an extension of this algorithm, also a method where different reference microphones are simultaneously used is proposed in [2].

The squared distance from the source to the i-th microphone is (from equation 2.9)

$$D_i^2 = (x_i - x_s)^2 + (y_i - y_s)^2 . \tag{2.38}$$

The squared range difference between the reference microphone and the i-th one is then

$$D_i^2 - D_0^2 = (x_i - x_s)^2 + (y_i - y_s)^2 - (x_0 - x_s)^2 - (y_0 - y_s)^2 . \tag{2.39}$$

Starting from this equation, Gillette and Silverman show how to obtain the following linear system

$$\begin{bmatrix} x_0 - x_1 & y_0 - y_1 & d_{1,0} \\ x_0 - x_2 & y_0 - y_2 & d_{2,0} \\ \vdots & \vdots & \vdots \\ x_0 - x_N & y_0 - y_N & d_{N,0} \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ D_0 \end{bmatrix} = \begin{bmatrix} w_{1,0} \\ w_{2,0} \\ \vdots \\ w_{N,0} \end{bmatrix} \tag{2.40}$$

where

$$w_{i,j} \equiv \frac{1}{2}\left(d_{ij}^2 - x_i^2 + x_j^2 - y_i^2 + y_j^2\right) . \tag{2.41}$$

The solution can then be obtained with a least-squares inversion.

As mentioned above the algorithm can then be easily extended to a more general one, using M reference-microphones at the same time. The following example shows how to extend the algorithm for $M = 2$ and $M = 5$

microphones. The system to be solved becomes

$$\begin{bmatrix} x_0 - x_1 & y_0 - y_1 & d_{1,0} & 0 \\ x_0 - x_2 & y_0 - y_2 & d_{2,0} & 0 \\ x_0 - x_3 & y_0 - y_3 & d_{3,0} & 0 \\ x_0 - x_4 & y_0 - y_4 & d_{4,0} & 0 \\ x_1 - x_0 & y_1 - y_0 & 0 & d_{0,1} \\ x_1 - x_2 & y_1 - y_2 & 0 & d_{2,1} \\ x_1 - x_3 & y_1 - y_3 & 0 & d_{3,1} \\ x_1 - x_4 & y_1 - y_4 & 0 & d_{4,1} \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ D_0 \\ D_1 \end{bmatrix} = \begin{bmatrix} w_{1,0} \\ w_{2,0} \\ w_{3,0} \\ w_{4,0} \\ w_{0,1} \\ w_{2,1} \\ w_{3,1} \\ w_{4,1} \end{bmatrix}. \tag{2.42}$$

The solution can be once more obtained with a least-squares inversion.

As mentioned above, this method ensures a simple and closed-form solution. In order to use this technique a greater number of microphones should be used, if compared to other methods. By the way it is a candidate solution when only a few time is given in order to find the solution.

Taylor Series Another interesting solution is given by the Taylor series expansion of equation 2.22 about a reference point and a successive iterative gradient search ([3] ,[19]). This algorithm can also be viewed as a method for iteratively minimize the Hyperbolic LS Error Function.

To determine a reasonably simple estimator, the equation 2.22 can be linearized by retaining only the first two terms of the Taylor series expansion about the reference point $\boldsymbol{r}_{s,0} = [x_{s,0}, y_{s,0}]^T$

$$\boldsymbol{g}(\boldsymbol{r}_s) \backsimeq \boldsymbol{g}(\boldsymbol{r}_{s,0}) + \boldsymbol{G} \cdot (\boldsymbol{r}_s - \boldsymbol{r}_{s,0}). \tag{2.43}$$

Here $\boldsymbol{G}$ is the gradient matrix

$$\boldsymbol{G} = \begin{bmatrix} \frac{\partial g_1}{\partial x_s} & \frac{\partial g_1}{\partial y_s} \\ \vdots & \vdots \\ \frac{\partial g_N}{\partial x_s} & \frac{\partial g_N}{\partial y_s} \end{bmatrix}\Bigg|_{\boldsymbol{r}_s = \boldsymbol{r}_{s,0}} = \begin{bmatrix} \frac{x_0 - x_s}{D_0} - \frac{x_1 - x_s}{D_1} & \frac{y_0 - y_s}{D_0} - \frac{y_1 - y_s}{D_1} \\ \vdots & \vdots \\ \frac{x_0 - x_s}{D_0} - \frac{x_N - x_s}{D_N} & \frac{y_0 - y_s}{D_0} - \frac{y_N - y_s}{D_N} \end{bmatrix}\Bigg|_{\boldsymbol{r}_s = \boldsymbol{r}_{s,0}} \tag{2.44}$$

where

$$\frac{\partial g_i}{\partial x_s} = \frac{x_0 - x_s}{D_0} - \frac{x_i - x_s}{D_i}, \tag{2.45}$$

$$\frac{\partial g_i}{\partial y_s} = \frac{y_0 - y_s}{D_0} - \frac{y_i - y_s}{D_i}, \tag{2.46}$$

and the vector $\boldsymbol{r}_{s,0}$ could be an estimate of $\boldsymbol{r}_s$ determined from a previous iteration or based upon a priori information. The source localization can be performed with a gradient search, as progressive approximation starting from an initial point $\boldsymbol{r}_{s,0}$ and applying the following update at the v-th iteration

$$\boldsymbol{r}_{s,v+1} = \boldsymbol{r}_{s,v} + \left(\boldsymbol{G}_v^T \boldsymbol{G}_v\right)^{-1} \boldsymbol{G}_v^T \boldsymbol{e}_h\left(\boldsymbol{r}_{s,v}\right) \tag{2.47}$$

where $\boldsymbol{e}_h$ is the hyperbolic LS error function taken from equation 2.27. This method is in fact an algorithm to iteratively minimize the Hyperbolic LS Error Function.

This algorithm present a simple way to deal with the Hyperbolic LS Error Function. A negative aspect is given by its iterative way of converging to the solution. In order to use this method, the number of iteration should be controlled in order to give a reasonable time of convergence as well as an accurate solution.

Exact Noniterative Linear Method In [7] a noniterative linear method is proposed. This method is called an exact method because no linearization are made. This algorithm is based on an inverse approach very similar to the idea at the basis of Spherical LS Error Function. In particular receivers are presented as if they act as sources emitting circular wave fronts, so that at a given time all the wave fronts intersect at the source position. This method needs at least four microphones for 2D localization in order to write three equation (one for each independent couple of microphones) linear in the unknowns x_s, y_s, and the propagation time t. Taking account of propagation time as well as the two space coordinate is one of the most interesting aspects of this method.

2.3 Multi-Source Localization

When dealing with multiple sources, we want to be able to localize all of them on the plane where they lie, having information about microphones positions and

TDOAs.

If we have M white and uncorrelated sources $\boldsymbol{r}_{s_a}$, $a = 1, ..., M$ emitting $s_a(t)$ in an anechoic room with $N+1$ sensors at $\boldsymbol{r}_i$, $i = 0, ..., N$, where only the direct paths from source a to sensor i with delay $\tau_{a,i}$ and attenuation $h_{a,i}$ contribute to the sensor signals

$$x_i(t) = \sum_{a=1}^{M} h_{a,i} s_a(t - \tau_{a,i}) \ , \ i = 0, ..., N, \tag{2.48}$$

the cross-correlation $r_{ij}(\tau)$ will show at maximum M local extrema. As previously shown in this Chapter, when only one source is considered, the cross-correlation exhibits only one peak, and TDOA can be extracted from the location of this peak.

In the multi-source case, each of the M peaks location corresponds to a TDOA for a particular source. Unfortunately we can not correctly assign peaks to sources. Resolving this multiple-source ambiguity is important for localization, because we have to assign each TDOA to one source and consider all TDOAs of that particular source together to estimate its geometric position. This problem is known as TDOA disambiguation.

After this problem is solved, and each TDOA extracted from cross-correlation is assigned to the correct source, multi-source localization problem can be treated as separate single source localization problems.

2.3.1 State of the Art

In order to solve TDOA disambiguation problem we can found several different approaches in the literature. Anyway some authors propose methods for simultaneously locate all the sources. Now we sum up some of this methods.

Clustering Technique One possible solution consist in finding source location candidates testing several TDOAs combinations [11], and then perform a check in order to find real sources locations.

In order to use this method array microphones are paired in doublets. Microphones in each doublet are supposed to be physically close to each other, so that they can be assumed to receive a time-shifted replica of each source signal,

filtered by the same acoustic transfer function.

Signals coming from each pair of microphones are pre-processed by a standard LPC algorithm in order to remove the common spectral features present in doublet signals (including the pitch) and minimize spectrum fluctuations.

If the number of sources is not known, it can be estimated, and TDOAs are extracted from the received signals. From TDOAs generated for each pair of doublets, a candidate source position is computed by efficient geometric algorithms available in the literature.

While wrong estimates usually generate disperse clusters containing few points, dense clusters having an approximately elliptical shape are formed around the speakers. Centroids of clusters are finally selected as speaker locations.

DATEMM (Zero-Sum) Disambiguation of TDOA Estimates in Multi-path Multi-Source Environments (DATEMM) algorithm [14] is another solution to the problem. This algorithm is useful for assigning each peak of the cross-correlation to a source. This allow us to separate TDOAs for each source as well as recognizing which TDOA belongs to reflective paths.

The algorithm is composed by two steps. In its first step, DATEMM algorithm establishes which peaks of cross-correlations belong to the direct signal, and which to the reflected signal. If we consider a non reverberant room, we can go further this step.

Once cross-correlation peaks related to reflective paths are removed, a step is performed in order to assign TDOAs to sources. This assignment is based on consideration on TDOAs between microphones couples. A particular check is performed between TDOAs in order to find those which match a condition called Zero-Sum condition. However this step is better explained in Chapter 5, when our algorithm for multi-source localization is explained.

Gaussian Likelihood Criterion Another possible solution consists in using a blind source separation (BSS) method combined with a likelihood criterion. In [8], authors propose a BSS method involving a multiple-input-multiple-output (MIMO) model in order to estimate TDOAs for each microphone pairs. This MIMO system is also shown in [9] as an alternative to the use of cross-correlation for extracting TDOAs from measurements. This method focuses directly on the impulse response between a source and microphones and thus, this approach is

inherently based on the real reverberant propagation model. While performing the separation of sources, the BSS method behaves similarly to a set of adaptive null beamformers, which steer nulls in the directions of the sources.

After this estimation step, a spatial ambiguity for sources location still exists. The TDOAs extracted for each microphones pair can not still be used to localize a single source position. In order to overcome this problem, a likelihood function is proposed, and all the possible source positions are tested. A comparison between the likelihood of all possible source configurations is performed, and the configuration giving the highest likelihood value is considered the correct configuration. This step is shown to sometime fail in simultaneously localize many sources for some configurations. However, it can localize at least one of them successfully most of the time.

This algorithm also takes account of the possibility of tracking moving sources. In this case a particle filter is implemented.

2.4 Reflector Localization

Inference problem considered in this section consists in finding the position of a single reflector present into a scene, dealing with TDOAs measurements.

We consider the same array and source setup used for source localization problem, but we add a reflector. This reflector can be modeled on the plane xy as a line of equation $y = mx+q$. Using the Cartesian coordinate system centered in $\boldsymbol{r}_0 = [0,0]^T$ the reflector can be defined as all the points $\boldsymbol{x} = [x, y, 1]^T$ which satisfy the condition

$$\boldsymbol{x}^T \boldsymbol{l} = 0, \tag{2.49}$$

where $\boldsymbol{l} = [-m, 1, -q]^T$.

Dealing with one reflector, each microphone receives the signal from the direct path as well as from the indirect path created from the reflection of the signal from point $\boldsymbol{r}_p$. A method to treat the multi-path propagation consists in modeling the signal coming from the indirect path as it comes from another source, called the image source. The image source position is found mirroring the real source through the reflector. As we work with real and image sources we define $\boldsymbol{r}_{s'}$ as the image source related to real source $\boldsymbol{r}_s$.

However, in order to estimate real and image source locations we face once

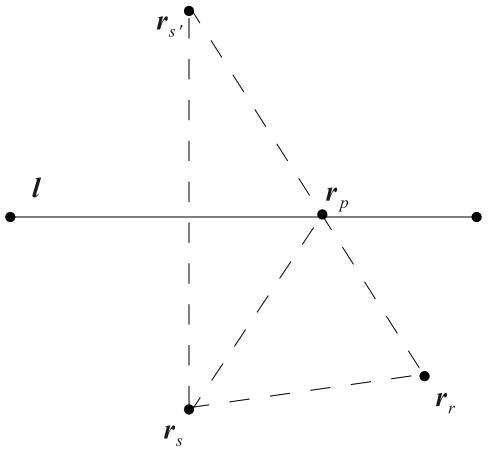


Figure 2.3: Inference Setup: real source is in $\boldsymbol{r}_s$, image source in $\boldsymbol{r}_{s'}$, microphone in $\boldsymbol{r}_r$, and $\boldsymbol{r}_p$ represents the reflection point. $\boldsymbol{l}$ represents a reflector.

more the TDOA disambiguation problem. In the multi-source case we had different sources emitting different signals, which leads to a number of peaks in cross-correlations equal to the number of sources. With a reflector and one source, we can consider having two sources, the real and the image one, synchronized and emitting the same signal. This leads to four peaks in each cross-correlation between microphones instead of only two. For this reason we deal with a situation worse than the multi-source one previously shown.

However, as we will see in Chapter 6, this TDOA ambiguity can be solved with some considerations based on the analysis of the auto-correlations of the signal received by each microphone. Then, once we are able to assign cross-correlation peaks to real and image sources, we can estimate their position $\boldsymbol{r}_{s'}$ and $\boldsymbol{r}_s$ from these TDOAs and derive $\boldsymbol{l}$ as shown in Chapter 6. However other solutions found in the literature are proposed.

2.4.1 State of the Art

The problem of room geometry estimation using microphones and loudspeakers has been addressed by several authors with various approaches. We first show a different approach respect to the cross-correlation in order to extract TDOAs taking account of reverberations. Then we propose two room geometry estima-

tion methods, which differs from our approach, in order to show that problem formulation can be different from ours. Finally we also show a possible solution found in the literature for solving TDOA ambiguity caused by multipath.

Blind Channel Identification Using cross-correlation for estimating TDOAs, we approximate the acoustic room impulse response as a simple delta function, and the TDOA estimation is achieved by maximizing the cross-correlation function. However in [1] and [10] different approaches are proposed based on blind channel identification. Blind channel identification approaches model an acoustic room impulse response as an FIR filter that includes both a direct path and multipath reflections. In these approaches, after the modeling filters have been identified, the TDOA can be easily computed by examining the direct paths in the filters.

In a room with a source and two microphones, the i-th microphone output at time k can be written as:

$$x_i(k) = s(k) * h_i + n_i(k), \tag{2.50}$$

where $*$ denotes linear convolution, $s(k)$ is the source signal, h_i represents the channel impulse response between the source and the i-th microphone, and $n_i(k)$ is a noise signal. The blind channel identification via cross relation is based on a clever observation, $x_2(k) * h_1 = x_1(k) * h_2 = s(k) * h_1 * h_2$, if the microphone signals are noiseless [20]. Then, without requiring any knowledge from the source signal, the channel filters can be identified by minimizing the squared cross-correlation error. In matrix-vector form, the optimization becomes

$$\begin{aligned} \boldsymbol{h}_1^*, \boldsymbol{h}_2^* = \arg\min_{\boldsymbol{h}_1, \boldsymbol{h}_2} \frac{1}{2} \|\boldsymbol{X}_2 \boldsymbol{h}_1 - \boldsymbol{X}_1 \boldsymbol{h}_2\|^2 \\ s.t. \|\boldsymbol{h}_1\|^2 + \|\boldsymbol{h}_2\|^2 = 1, \end{aligned} \tag{2.51}$$

where $\boldsymbol{X}_i$ is the $(N + L - 1) \times L$ convolution Toeplitz matrix whose first row and first column are $[x_i(k-N+1), x_i(k-N), ..., x_i(k-N-L+2)]$ and $[x_i(k-N+1), x_i(k-N+2), ..., x_i(k), 0, ..., 0]^T$ respectively, N is the microphone signal length, L is the filter length, and $\|.\|$ denotes l_2-norm. This

minimization problem can be solved by eigenvalue decomposition. When the filters are estimated, the TDOAs can be computed by examining the direct paths in the filters.

By using a more realistic model, the blind channel identification approaches have been shown to be more effective than cross-correlation approaches to reverberation. However we are working in a simple case where only one reflector is present, and we do not need all the information that the room impulse response brings. For this reason, the computational complexity of this algorithm is not justified for our purpose, and prevents us from using it.

Continuous Signals Method In [15], authors propose a method for estimating reflective surfaces using continuous signals without any prior information on the source signal. For this method, some sources and sensors are placed into a room whose walls positions need to be estimated. TDOAs from microphones pairs are computed by exponential fitting of cross-correlation functions. Then the approach is based on the inverse mapping of the multi-path propagation problem.

Reflective surface can be defined as a plane with points fulfilling perpendicularity to a normal vector $\boldsymbol{n}$:

$$P^2(\boldsymbol{n}, \boldsymbol{a}) = \{\boldsymbol{x} \in R^3 : \boldsymbol{n}^T(\boldsymbol{a} - \boldsymbol{x}) = 0), \tag{2.52}$$

where $\boldsymbol{a}$ defines an arbitrary point within the plane P^2. Replacing the point $\boldsymbol{a}$ with $\boldsymbol{b} + \boldsymbol{n}$ plane parameterization then becomes

$$P^2_{\boldsymbol{b}} = P^2(\boldsymbol{n}, \boldsymbol{b} + \boldsymbol{n}) = \{\boldsymbol{x} \in R^3 : \boldsymbol{n}^T((\boldsymbol{b} + \boldsymbol{n}) - \boldsymbol{x}) = 0\}, \tag{2.53}$$

which has only 3 degrees of freedom but as a trade-off loses ability to represent planes containing the reference point $\boldsymbol{b}$.

To identify a reflective surface P^2, we consider a system of three points $\boldsymbol{r}_i$, $\boldsymbol{r}_s$ and $\boldsymbol{o}_i \in P^2$, namely the positions of i-th receiving sensor, sound source and the point of reflection (Figure 2.4). The former two are assumed to be known and the point of reflection has to be estimated.

Direct path time delay is determined as $\tau_d(\boldsymbol{r}_s, \boldsymbol{r}_i) = c^{-1} \parallel \boldsymbol{r}_i - \boldsymbol{r}_s \parallel$. The

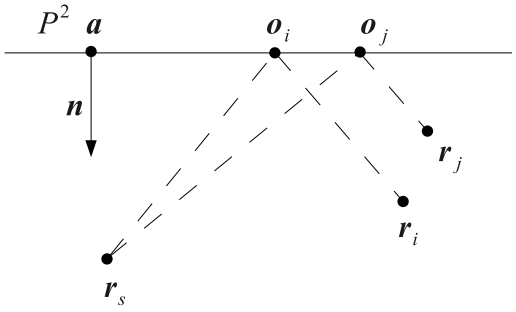


Figure 2.4: Reflector Top View for Continuous Signals Method: P^2 is the reflector, $\boldsymbol{n}$ is the normal vector, $\boldsymbol{a}$ an arbitrary point on the reflector. $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ are two receivers, $\boldsymbol{r}_s$ is the source, and $\boldsymbol{o}_i$, $\boldsymbol{o}_j$ are two point of reflection. The indirect signal path is shown from source to microphones i and j.

time delay for a reflective propagation path is twofold and can be defined using the three points defined earlier with

$$\tau(\boldsymbol{r}_s, \boldsymbol{o}_i, \boldsymbol{r}_i) = \tau(\boldsymbol{r}_s, \boldsymbol{o}_i) + \tau(\boldsymbol{o}_i, \boldsymbol{r}_i). \tag{2.54}$$

Furthermore, the law of reflection states that the piecewise propagation path has equal angles of incidence and reflection. As it happens, ellipsoids $E^2(\boldsymbol{r}_s, \boldsymbol{r}_i)$ with sensor and source acting as its foci fulfill this requirement. This does not, however, gives the explicit point of reflection, rather than a set of points fulfilling (2.54).

Given a plane P^2 the point of reflection can be realized as

$$\boldsymbol{o}_i = \frac{\parallel \text{proj}_{P^2}\boldsymbol{r}_s \parallel \text{proj}_{P^2}\boldsymbol{r}_i + \parallel \text{proj}_{P^2}\boldsymbol{r}_i \parallel \text{proj}_{P^2}\boldsymbol{r}_s}{\parallel \text{proj}_{P^2}\boldsymbol{r}_s \parallel + \parallel \text{proj}_{P^2}\boldsymbol{r}_i \parallel} \tag{2.55}$$

where proj_{P^2} is a point projection on the plane P^2.

For a plane $\boldsymbol{p} \in P^2_{\boldsymbol{b}}$ parameterized as 2.53, the delay becomes independent of the reflection point: $\tau(\boldsymbol{r}_s, \boldsymbol{o}_i, \boldsymbol{r}_i) = \tau(\boldsymbol{r}_s, \boldsymbol{r}_i, \boldsymbol{p})$

Reflective surfaces can be found by performing a search into the plane-

space defined by 2.53 where delay differences from separate source-sensor combinations intersect in the search space.

The surface is estimated as the maximum argument of a pseudo-likelihood function as

$$\hat{\boldsymbol{p}} = \arg\max_{\boldsymbol{p}} \prod_{i,j=1}^{M} R_{x_i,x_j}(\tau_{i,j}(\boldsymbol{r}_s, \boldsymbol{p})), \tag{2.56}$$

where $R_{x_i,x_j}(\tau)$ is the generalized cross correlation between two received signals x_i and x_j, and time delay differences can be calculated as

$$\tau_{ij}(\boldsymbol{r}_s, \boldsymbol{p}) = \tau(\boldsymbol{r}_s, \boldsymbol{r}_i, \boldsymbol{p}) - \tau_d(\boldsymbol{r}_s, \boldsymbol{r}_j). \tag{2.57}$$

This solution to reflector localization is an advanced solution and it makes possible to locate many reflectors. However with the above formulation the source position has to be known.

L1 Regularized Room Modeling An algorithm for obtaining a room model based on studying the room impulse response can be found in [16]. This algorithm uses a constrained room model and l_1-regularized least-squares to achieve good estimation of room geometry. The main idea consists in synthetically or experimentally obtaining a set of impulse responses for a set of hypothesized walls positions, and performing a smart search between this set of impulse responses in order to find walls positions compatibles with the received signal.

Authors define the single wall impulse response (SWIR) $h_m^{(r,\theta,\phi)}(n)$ as the discrete time impulse response from the loudspeaker to the m-th microphone, considering that: the direct path from loudspeaker to the microphone has been removed and the array is mounted on free space, except for the presence of a lossless, infinite wall with normal vector $\boldsymbol{n} = (r, \theta, \phi)$ and which contains the point (r, θ, ϕ). For reasonably small arrays the sound will take approximately the same path from the source to each of the microphones, which implies that it should with high probability reflect off the same walls before reaching each microphone, such that the reflection coefficients will be the same for every microphone. We can then rewrite the SWIR $h_m^{(r,\theta,\phi)}(n)$

related to a generic microphone as a vector $\boldsymbol{h}^{(r,\theta,\phi)}$ of length N just large enough to contain the first order reflections.

The first algorithm step consists in obtaining synthetically or experimentally for the array of interest a set of SWIRs, each measured at fixed range over a grid $\mathcal{A}$ of azimuth angles, and the SWIRs containing only the reflection from a ceiling at the same fixed range. In essence this set of SWIRs carries a time-domain description of the array manifold vector for multiple directions of arrival.

If $\mathcal{A}$ is sufficiently fine, for a set of W walls, there are coefficients c_i , $i = 1, ..., W$ such that we can represent an impulse response $\boldsymbol{h}_{room}$, which has the direct path removed and truncated to only contain early reflections,

$$\boldsymbol{h}_{room} \approx \sum_{i=1}^{W} c_i \boldsymbol{h}^{(r_0,\theta_i,\phi_i)}. \tag{2.58}$$

The problem is then to fit some $\boldsymbol{h}^{(r_0,\theta_i,\phi_i)}$ for some value of (r_0, θ_i, ϕ_i) to the measured impulse response, adjusting for attenuation. In order to accomplish this goal, the SWIRs are ordered into a matrix $\boldsymbol{H}$ and attenuation coefficients c_i are ordered into a vector $\boldsymbol{a}$. The solution to the problem is then given when the correct $\boldsymbol{a}$ is found. In order to find $\boldsymbol{a}$ the following l_1-regularized least-squares problem should be solved

$$\min_{\boldsymbol{a}} \parallel \boldsymbol{h}_{room} - \boldsymbol{H}\boldsymbol{a} \parallel_2^2 + \lambda \parallel \boldsymbol{a} \parallel_1 . \tag{2.59}$$

If we consider only SWIRs with coefficients $[\boldsymbol{a}]_i$ larger than a given threshold, then we have a set of candidate walls. With a post-processing stage, impossible solutions are discarded in order to find the correct one.

This algorithm aim is to find the complete room geometry. For this reason its computational complexity is high. When searching for a single reflector, using a method so complex is not necessary.

DATEMM (Raster Condition) In [13] authors show that by exploiting the peak positions of the auto-correlation function, the extrema positions in the cross-correlation always appear in a certain raster: a set of time marks with known

distances between them. Fig. 2.5 shows a simple example with one source, two sensors, and two paths per sensor. The raster in the cross-correlation consists of

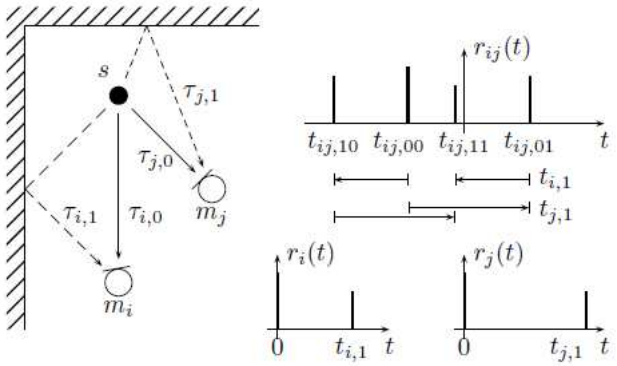


Figure 2.5: TDOA Disambiguation: raster for one source as shown in [13]. Distances between peaks in cross-correlations can be found as peaks positions in auto-correlations.

four time marks whose distances are known from the peak positions of the auto-correlations. By finding this raster in the cross-correlation, its absolute position determines the desired TDOA of direct and indirect paths.

For this purpose, we first extract the relevant peaks from both cross- and auto-correlations. We now consider two TDOAs $\tau_{ij,\mu_1\nu_1}$ and $\tau_{ij,\mu_2\nu_2}$ resulting from the same source where the paths to sensor i are common ($\mu_1 = \mu_2 = \mu$) and one of the paths to sensor j is a direct path ($\nu_1 = 0$ or $\nu_2 = 0$). The distance $|\tau_{ij,\mu\nu_1} - \tau_{ij,\mu\nu_2}|$ can be found as the position of a peak in the auto-correlation of sensor j, see Fig. 2.5. As the direct path is always the shortest, we can also determine the sign of the above difference and hence identify whether ν_1 or ν_2 is the direct path.

Continuing this raster match for all TDOA pairs, $\tau_{ij,00}$ will be most likely identified several times as the direct path (TDOA associated to real source) while all other path combinations $(\mu, \nu) \neq (0, 0)$ will at least once be identified as non-direct. The TDOA $\tau_{ij,\mu\nu}$ identified as non-direct more times than others, is the TDOA associated to image source.

This algorithm allows us to assign TDOAs extracted from cross-correlation to image and real sources. In Chapter 6 we better explain how to use this method in order to solve TDOA disambiguation problem.

2.5 Conclusions

We have introduced the three problems treated in the thesis: single source localization, multi-source localization, and reflector position inference. We have also shown the notation used in next Chapters and we have given an overview of some classical methods found in the literature to deal with the proposed problems.

In the next Chapter we outline a different approach to source localization and inference using geometries considerations and a new coordinate system. We also introduce the new framework for solving these problems.

Chapter 3

From Measurements to Constraints

In the previous Chapter we have given a mathematical formulation of localization and reflector inference problems, showing also some basic algorithms based on TDOAs proposed in the literature to sort out these problems.

In this Chapter we focus on the approach based on the use of homogeneous coordinates. This approach starts describing objects such as microphones and sources in a new reference frame in order to find other possible solutions to the already described problems. We show how in the literature a solution based on working in a 3D frame for 2D localization is already given, and then we present a new framework in order to find new TOAs and TDOAs constraints. Our aim is to present a 3D coordinate system similarly as in [5], but giving a physical meaning to the third coordinate as done in [7], where authors introduce propagation time as a new coordinate.

After this 3D framework is presented, a set of constraints for localization is given.

3.1 State of the Art

In order to solve localization and inference problems, some methods using projective geometry have been proposed. These methods mainly focus on turning measurements like TOAs and TDOAs into constraints acting on geometric primitives, namely points (sources or receivers) and reflectors. An interesting aspect of these methods consists in the compact notation that they use for constraints. In particular, the state of the art shows that each measurement on acoustic paths

turns out to generate a projective constraint that takes the shape of a quadratic or a bilinear form.

3.1.1 Direct Time of Arrival Constraints

In [5] is shown how measurements of the time of arrival can be turned into constraints that act directly on geometric primitives. As far as the direct signal is concerned, the quadratic form represents a constraint on the position of sources or microphones.

The set τ_i , $i = 0, ..., N$ contains the time of arrival measured from each microphone in the array. Let us consider a single measurement τ_i . If the source location $\boldsymbol{r}_s = [x_s, y_s]^T$ is given, the time of arrival constrains the microphone to be placed on a circumference centered on the source location and with radius $\rho_i = \tau_i \cdot c$, where c is the sound speed. The equation of the circumference is

$$(x_i - x_s)^2 + (y_i - y_s)^2 = \rho_i^2. \tag{3.1}$$

We can describe the circumference using a parameter vector $\boldsymbol{c} = [a, b, c, d, e, f]^T$. The computation of the parameter vector $\boldsymbol{c}$ from the knowledge of ρ_i and $\boldsymbol{r}_s$ can be performed in two different ways:

1. by comparing the expansion of equation (3.1) with the general form of a conic given by

   $$ax_i^2 + bx_iy_i + cy_i^2 + dx_i + ey_i + f = 0 \tag{3.2}$$

   we obtain that

   $$\boldsymbol{c} = \left[1, 0, 1, -2x_s, -2y_s, -\rho_i^2 + x_s^2 + y_s^2\right]^T. \tag{3.3}$$

2. the parameter is obtained by determining the null-space of the coefficient

matrix of equation system

$$\begin{bmatrix} x_1^2 & x_1y_1 & y_1^2 & x_1 & y_1 & 1 \\ x_2^2 & x_2y_2 & y_2^2 & x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_5^2 & x_5y_5 & y_5^2 & x_5 & y_5 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} = \mathbf{0} \tag{3.4}$$

3.1.2 Indirect Time of Arrival Constraints

As shown in [6], dealing with reflected signal, the constraint acts on the line on which the reflector generating the acoustic path lies.

As a result of the measurement of the TOA we obtain a set τ_i of TOAs related to reflected signals. Let us consider an acoustic path that links the source located at $\boldsymbol{r}_s = [x_s, y_s]^T$ and the receiver located at $\boldsymbol{r}_i = [x_i, y_i]^T$ through a reflection, as depicted in Figure 3.1. The corresponding time of arrival is τ_i . The reflection point $\boldsymbol{r}_p$ on the reflector honors the Snell's law. Our measurement τ_i is the sum of

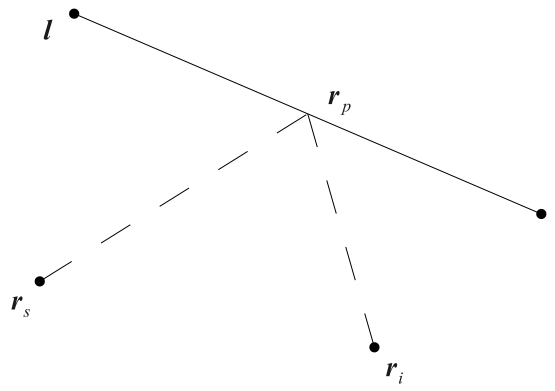


Figure 3.1: TOA, Reflection on a Wall: An acoustic path links $\boldsymbol{r}_s$ and $\boldsymbol{r}_i$ through the reflector $\boldsymbol{l}$. The reflection point $\boldsymbol{r}_p$ honors the Snell's law.

the Time of Flight $\tau(\boldsymbol{r}_s, \boldsymbol{r}_p)$ from $\boldsymbol{r}_s$ to $\boldsymbol{r}_p$ and the Time of Flight $\tau(\boldsymbol{r}_p, \boldsymbol{r}_i)$ from $\boldsymbol{r}_p$ to $\boldsymbol{r}_i$. Without any information except τ_i , as a consequence, we are constraining

the reflection point $\boldsymbol{r}_i$ to lie on an ellipse whose foci are $\boldsymbol{r}_s$ and $\boldsymbol{r}_i$ and whose major axis is $\alpha = \tau_i c/2$, where c is the sound speed. As for the circumference, we can directly compute the parameter vector $\boldsymbol{c}$ from the geometrical information associated to the problem. We first define some auxiliary variables:

- the minor axis of the ellipse is

$$\beta = \frac{1}{2}\sqrt{\tau_i^2 c^2 - [(x_s - x_i)^2 + (y_s - y_i)^2]};$$

- the major axis of the ellipse lies on a line whose slope is

$$\phi = \arctan\left(\frac{y_s - y_i}{x_s - x_i}\right);$$

- the ellipse has its center in

$$\boldsymbol{r}_c = \left[x_c = \frac{1}{2}(x_s + x_i), y_c = \frac{1}{2}(y_s + y_i)\right]^T.$$

With these definitions at hand, the parameter vector is

$$\boldsymbol{c} = \begin{bmatrix} \frac{\cos(\phi)^2}{\alpha^2} + \frac{\sin(\phi)^2}{\beta^2} \\ 2\sin(\phi)\cos(\phi)(\frac{1}{\beta^2} - \frac{1}{\alpha^2}) \\ \frac{\sin(\phi)^2}{\alpha^2} + \frac{\cos(\phi)^2}{\beta^2} \\ \cos(\phi)\sin(\phi)x_c(\frac{2}{\alpha^2} - \frac{2}{\beta^2}) - \frac{2x_c}{\alpha^2}\cos(\phi)^2 - \frac{2x_c}{\beta^2}\sin(\phi)^2 \\ \cos(\phi)\sin(\phi)y_c(\frac{2}{\alpha^2} - \frac{2}{\beta^2}) - \frac{2y_c}{\alpha^2}\cos(\phi)^2 - \frac{2y_c}{\beta^2}\sin(\phi)^2 \\ \cos(\phi)\sin(\phi)x_c y_c(\frac{1}{\beta^2} - \frac{1}{\alpha^2}) + x_c^2(\frac{\cos(\phi)^2}{\alpha^2} + \frac{\sin(\phi)^2}{\beta^2}) + y_c^2(\frac{\sin(\phi)^2}{\alpha^2} + \frac{\cos(\phi)^2}{\beta^2}) - 1 \end{bmatrix}$$

We observe that the constraint that imposes the presence of the reflection point on an ellipse does not involve any geometric primitive. In particular, we are interested in finding the line on which the reflector lies.

In order to do so, we use an alternative method to define conics. In particular, we can rearrange the parameter vector $\boldsymbol{c} = [a, b, c, d, e, f]^T$ in matrix form as

$$\boldsymbol{C} = \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix}. \tag{3.5}$$

If $\boldsymbol{x} = [x_i, y_i, 1]^T$ are the homogeneous coordinates corresponding to the point $[x_i, y_i]^T$, the conic can be written as

$$\boldsymbol{x}^T \boldsymbol{C} \boldsymbol{x} = 0.$$

If the reflector is defined as all the points $\boldsymbol{x} = [x_i, y_i, 1]^T$ which satisfy the condition

$$\boldsymbol{x}^T \boldsymbol{l} = 0,$$

where $\boldsymbol{l} = [l_1, l_2, l_3]^T$, we can define the conic using the set of lines $\boldsymbol{l}$ tangent to the conic as

$$\boldsymbol{l}^T \boldsymbol{C}^* \boldsymbol{l} = 0.$$

This constraint now acts directly on the reflector line $\boldsymbol{l}$.

3.1.3 Direct Time Difference of Arrival Constraints

Also TDOA measurements can be turned into constraints in a projective geometry space [5].

If τ_{ij} is the time difference of arrival between the signals acquired by microphones $\boldsymbol{r}_i = [x_i, y_i]^T$ and $\boldsymbol{r}_j = [x_j, y_j]^T$, the source $\boldsymbol{r}_s = [x_s, y_s]^T$ is constrained to lie on an ellipse whose foci are $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ and whose axis is $\alpha = \tau_{ij} c/2$. The parameter vector $\boldsymbol{c}$ that determines this conic can be determined by expanding the equation of a generic hyperbola with foci $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ and with major axis α. We first define some variables that are useful for the determination of the parameter vector:

- the center of the hyperbola is in the point

$$\boldsymbol{r}_c = \left[x_c = \frac{x_i + x_j}{2}, y_c = \frac{y_i + y_j}{2} \right]^T; \tag{3.6}$$

- the "minor axis" of the hyperbola is

$$\beta = \frac{\alpha^2 - (x_i - x_j)^2 - (y_i - y_j)^2}{\alpha \sqrt{1 - \frac{(x_i - x_j)^2 + (y_i - y_j)^2}{\alpha^2}}}; \tag{3.7}$$

- the major axis of the hyperbola lies on a line whose slope is

$$\phi = \arctan\left(\frac{y_j - y_i}{x_j - x_i}\right). \tag{3.8}$$

With these definitions at hand, the parameter vector of the hyperbola is

$$\boldsymbol{c} = \begin{bmatrix} \frac{\cos(\phi)^2}{\alpha^2} - \frac{\sin(\phi)^2}{\beta^2} \\ -2\sin(\phi)\cos(\phi)(\frac{1}{\beta^2} + \frac{1}{\alpha^2}) \\ \frac{\sin(\phi)^2}{\alpha^2} - \frac{\cos(\phi)^2}{\beta^2} \\ \cos(\phi)\sin(\phi)x_c(\frac{2}{\alpha^2} + \frac{2}{\beta^2}) - \frac{2x_c}{\alpha^2}\cos(\phi)^2 + \frac{2x_c}{\beta^2}\sin(\phi)^2 \\ \cos(\phi)\sin(\phi)y_c(\frac{2}{\alpha^2} + \frac{2}{\beta^2}) - \frac{2y_c}{\alpha^2}\cos(\phi)^2 + \frac{2y_c}{\beta^2}\sin(\phi)^2 \\ -\cos(\phi)\sin(\phi)x_c y_c(\frac{1}{\beta^2} + \frac{1}{\alpha^2}) + x_c^2(\frac{\cos(\phi)^2}{\alpha^2} - \frac{\sin(\phi)^2}{\beta^2}) + y_c^2(\frac{\sin(\phi)^2}{\alpha^2} - \frac{\cos(\phi)^2}{\beta^2}) - 1 \end{bmatrix}$$

3.1.4 Indirect Time Difference of Arrival Constraints

In [5], authors consider a scenario in which a single reflector is present in the environment and causes the presence of a reflective acoustic path that links $\boldsymbol{r}_s$ to $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ through a specular reflection. The time difference of arrival related to this acoustic path is τ_{ij}. $\boldsymbol{r}_s = [x_s, y_s]^T$ is the source location, $\boldsymbol{r}_{s'} = [x_{s'}, y_{s'}]^T$ is the image source location obtained by mirroring $\boldsymbol{r}_s$ over the reflector. The versor normal to the reflector and pointing to the half-space containing $\boldsymbol{r}_{s'}$ is $\boldsymbol{n}$. The distance between $\boldsymbol{r}_s$ and the reflector is d. Finally, $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ are the receiver locations on which the time difference of arrival is computed.

The constraint on the reflector location is then derived as a combination of other two constraints: the first expresses $\boldsymbol{r}_{s'}$ as the reflection of $\boldsymbol{r}_s$ over the reflector, while the second is related to the measurement of the time difference of arrival of the acoustic path from $\boldsymbol{r}_s$ to $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ over the reflector. The projective coordinates of $\boldsymbol{r}_s$ and $\boldsymbol{r}_{s'}$ are, respectively, $\boldsymbol{x}$ and $\boldsymbol{x}'$. The homography that relates $\boldsymbol{x}$ and $\boldsymbol{x}'$ is

$$\boldsymbol{x}' = \boldsymbol{H}_r \boldsymbol{x},$$

where

$$\boldsymbol{H}_r = \begin{bmatrix} \boldsymbol{I} - 2\boldsymbol{n}\boldsymbol{n}^T & 2d\boldsymbol{n} \\ \boldsymbol{0} & 1 \end{bmatrix}. \tag{3.9}$$

From geometric considerations we observe that, if $\boldsymbol{r}_s$ is known, the knowledge of the distance d and the versor $\boldsymbol{n}$ is sufficient to univocally identify the reflector.

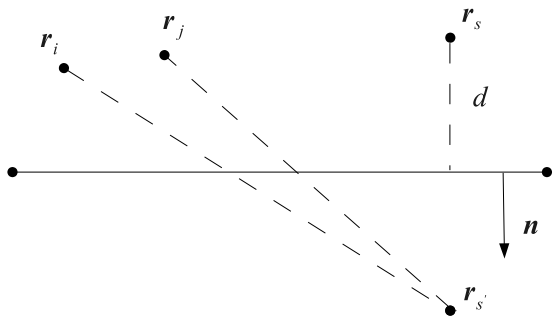


Figure 3.2: TDOA, Reflection on a Wall: an acoustic path links $\boldsymbol{r}_s$ to $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ through the reflector $\boldsymbol{l}$. The reflective path can be thought as generated from the image source $\boldsymbol{r}_{s'}$, obtained by mirroring $\boldsymbol{r}_s$ over the reflector.

The time difference of arrival τ_{ij} between the acoustic paths $\boldsymbol{r}_{s'}\boldsymbol{r}_i$ and $\boldsymbol{r}_{s'}\boldsymbol{r}_j$ constrains the image source $\boldsymbol{r}_{s'}$ to lie on a hyperbola with foci in $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ and with axis $\tau_{ij}c$. As a consequence, the constraint derived from the knowledge τ_{ij} is

$$\boldsymbol{x}'^T\boldsymbol{C}\boldsymbol{x} = 0, \tag{3.10}$$

where the conic-matrix $\boldsymbol{C}$ is obtained as in indirect TOA case from the parameter vector $\boldsymbol{c}$ used in direct TDOA case. From previews equations we obtain

$$\boldsymbol{x}^T\boldsymbol{H}_r^T\boldsymbol{C}\boldsymbol{H}_r\boldsymbol{x} = 0. \tag{3.11}$$

We observe that if the source position $\boldsymbol{r}_s$ is known, the only unknowns in this equation are the distance d and the versor $\boldsymbol{n}$, embedded into the matrix $\boldsymbol{H}_r$.

3.2 Problem Reformulation

Since now, using homogeneous coordinates, objects in a 2D space (sources, microphones, ...) have been turned into objects in a 3D space, but no specific meaning to the third coordinate was given. However it is possible to use a 3D working space giving a physical interpretation to the third coordinate in order to better understand the new obtained constraints.

Let us consider an array composed by $N+1$ microphones on a plane, located at $[x_i, y_i]^T$, $i = 0, ..., N$. The source is located on the same plane at $[x_s, y_s]^T$. If the source sends a signal at time t_s ,and the i-th microphone receives it at t_i, we can represent the microphones in a 3D space at $[x_i, y_i, t_i]^T$, $i = 0, ..., N$ and the source at $[x_s, y_s, t_s]^T$. In this new 3D space, x and y represent the real objects location on the plane where they lie, and the third coordinate represents the signal propagation time, so that points with same t are points receiving the signal simultaneously.

In this scenario the time of arrival (TOA) from the source to the i-th microphone is given by $\tau_i = t_i - t_s$, while time difference of arrival (TDOA) between microphones i and j is given by $\tau_{ij} = t_i - t_j$ for $i \neq j$.

Without loss of generality it is possible to multiply the t coordinate by a scale factor c corresponding to sound speed. In this new coordinate system, all the three coordinates $(x, y, z = t \cdot c)$ represent distances. Microphones are then located at $\boldsymbol{r}_i = [x_i, y_i, z_i]^T$, $i = 0, ..., N$ and the source is located at $\boldsymbol{r}_s = [x_s, y_s, z_s]^T$. In this new frame the difference $z_i - z_s$ represent the distance from the i-th microphone to the source, while $z_i - z_j$ for $i \neq j$ is the difference between distances from i-th microphone to the source and j-th microphone to the source.

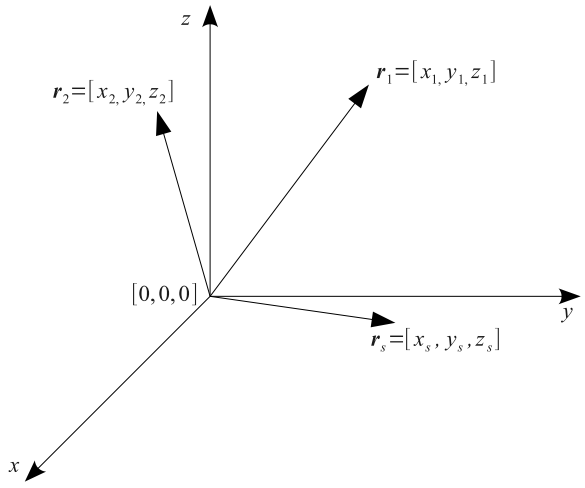


Figure 3.3: New Set of Coordinates: x and y represent the real objects location on the plane where they lie, while z is proportional to signal propagation time.

If the propagation medium is isotropic and homogeneous, the signal propa-

gating in a 2D space is described by a circle centered on the source location with radius increasing over time. In the new 3D space, any time instant corresponds to a z value, so for any radius value the circle lies on a specific plane parallel to the xy plane with z coordinate increasing with radius. These circles gives birth to a cone with apex in $\boldsymbol{r}_s$ and 45° angular aperture, and all the points $\boldsymbol{r}_i$ are constrained to lie on this cone (Figure 3.4) described by

$$(x - x_s)^2 + (y - y_s)^2 = (z - z_s)^2. \tag{3.12}$$

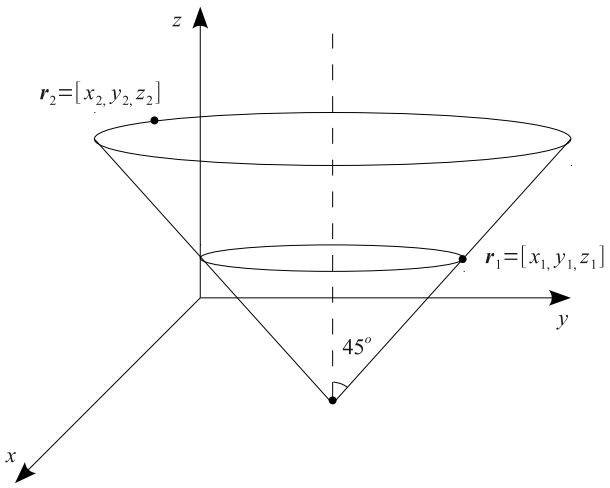


Figure 3.4: TOA Cone: the cone describes the signal propagating from the source located in $\boldsymbol{r}_s$, every microphone is constrained to lie on this cone. $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ represent two microphones.

3.2.1 Direct Time of Arrival Constraints

Let us consider the case in which we have TOAs measurements for each microphone. We can set the coordinate system reference so that $z_s = 0$. The values assumed by z_i are then directly proportional to TOAs

$$z_i = c \cdot \tau_i \quad , \quad i = 0, ..., N \tag{3.13}$$

where τ_i represents the TOA measurement at the i-th microphone. It is then possible to write the following condition for each microphone

$$z_i = \sqrt{x_i^2 + y_i^2} = c \cdot \tau_i \quad , \quad i = 0, ..., N \ . \tag{3.14}$$

In order to estimate the source location $[x_s, y_s]^T$ we should find the equation of the cone passing through the points $[x_i, y_i, z_i]^T$ according to the constraints

$$\begin{aligned} (x_i - x_s)^2 + (y_i - y_s)^2 &= z_i^2 \\ &= (c \cdot \tau_i)^2 \quad , \quad i = 0, ..., N. \end{aligned} \tag{3.15}$$

3.2.2 Indirect Time of Arrival Constraints

If only one reflector is present, every microphone receives both the direct and the reflected signal. The reflected signal can be modeled as the signal sent from an image source obtained mirroring the real source through the reflector. The reflector localization problem can then be solved if both the real and image source locations are found.

A simple solution of the problem is then provided in a few steps. We use the direct TOA constraints on direct signal in order to find the real source location, and the direct TOA constraints on reflected signal in order to find the image source location. Then we derive the reflector location on the plane as the line perpendicular to the segment joining the two sources passing from the middle point of this segment.

3.2.3 Direct Time Difference of Arrival Constraints

Let us consider the case in which we have TDOAs measurements for each pair of microphones. We can set the coordinate system origin to $\boldsymbol{r}_0$, so that z_i values are directly proportional to TDOAs referred to the microphone located at $\boldsymbol{r}_0$

$$\begin{aligned} z_i &= c \cdot (\tau_i + t_s) \\ &= c \cdot (\tau_i + t_0 - \tau_0) \\ &= c \cdot (\tau_i - \tau_0) \\ &= c \cdot \tau_{i0} \quad , \quad i = 1, ..., N \end{aligned} \tag{3.16}$$

where τ_{i0} is the TDOA between the i-th microphone and the reference one.

Compared to the TOA case, now we have one additional unknown parameter, z_s. For this reason we should estimate all the $\boldsymbol{r}_s$ components. In order to find $\boldsymbol{r}_s$ we constrain once more a cone to pass through the point $\boldsymbol{r}_i$, $i = 0, ..., N$ obtaining the following conditions

$$\begin{aligned} (x_i - x_s)^2 + (y_i - y_s)^2 &= (z_i - z_s)^2 \\ &= (c \cdot \tau_{i0} - z_s)^2 \quad , \quad i = 0, ..., N. \end{aligned} \tag{3.17}$$

3.2.4 Indirect Time Difference of Arrival Constraints

As with indirect TOA, when only one reflector is present, each microphone receives both the direct and the reflected signal. Once more the reflected signal can be modeled as a signal received from an image source, and can be distinguished from the direct one.

We can then use the direct TDOA constraints on both signals in order to find the real and image source locations. Once the sources are located, we can find the line representing the reflector on the plane. This line is perpendicular to the segment joining the real source to the image one and passes through the segment middle point.

3.3 Conclusions

In this Chapter we have shown how it is possible to perform source localization or inference working in a different space, in particular a 3D space. We have first described methods found in the literature, then we have proposed a novel approach in order to work in a new framework giving a meaning to all the three coordinates.

With this novel approach we have described a set of new constraints for TDOAs and TOAs measurements. Now we can show algorithms that make use of these new constraints so as to sort out source localization or inference problems.

Chapter 4

From Constraints to Localization

In the previous Chapters we have shown how source localization and reflector inference problems from TOAs or TDOAs measurements can be formulated, how they are treated in the literature and how they can be solved using the homogeneous coordinate space.

In this Chapter we discuss new TDOAs localization algorithms which make use of geometric constraints presented in the previous Chapter. These algorithms can be easily applied also to TOAs scenario. Some pieces of information about favorable microphone arrays displacement are then given.

4.1 Algorithms

As shown in the previous Chapter, source localization problem from TOAs and TDOAs measurements is solved when the apex of a special cone is found. The cone equation is

$$(x - x_s)^2 + (y - y_s)^2 - (z - z_s)^2 = 0. \tag{4.1}$$

but with noisy z_i measurements, the points $\boldsymbol{r}_i$ will not lie on this cone. We can model TOAs and TDOAs measurements as

$$\begin{array}{rcl} \hat{\tau}_i & = & \tau_i + \varepsilon_i \\ \hat{\tau}_{ij} & = & \tau_{ij} + \varepsilon_{ij} \end{array},$$

where τ_i and τ_{ii} represent noiseless measurements, and ε_i and ε_{ij} are additive noise terms.

The following proposed algorithms are mainly methods to build a cost function based on the cone idea. This function should then be minimized in order to find the apex of the cone at minimal distance from the points $\mathbf{r}_i$. Different cost functions are based on different distance definitions.

4.1.1 Cost Function Based on the Cone Equation

This algorithm is based on the idea that every point $\boldsymbol{r}_i$ is supposed to lie on the cone of equation 4.1. The error for each microphone is defined as

$$\varepsilon_{c,i} = (x_i - x_s)^2 + (y_i - y_s)^2 - (z_i - z_s)^2 \quad , \quad i = 0, ..., N. \tag{4.2}$$

The source coordinates are then given by

$$\boldsymbol{r}_s = \underset{\boldsymbol{r}_s}{min} \left(\sum_{i=0}^{N} \varepsilon_{c,i}^2 \right) = \underset{\boldsymbol{r}_s}{min} \left(\boldsymbol{\varepsilon}_c^T \boldsymbol{\varepsilon}_c \right) \tag{4.3}$$

where

$$\boldsymbol{\varepsilon}_c = \begin{bmatrix} \varepsilon_{c,0} \\ \varepsilon_{c,1} \\ \vdots \\ \varepsilon_{c,N} \end{bmatrix}. \tag{4.4}$$

In order to efficiently minimize this function, a method based on Taylor series expansion is proposed, following the same idea of [3] and [19]. The error vector $\boldsymbol{\varepsilon_c}$ can be written, using Taylor series expansion about a reference point $\boldsymbol{r}_{s,0} = [\ x_{s,0} \quad y_{s,0} \quad z_{s,0}\ ]^T$, as

$$\boldsymbol{\varepsilon}_c \simeq \boldsymbol{\varepsilon}_c|_{\boldsymbol{r}_{s,0}} + \nabla \boldsymbol{\varepsilon}_c|_{\boldsymbol{r}_{s,0}} \cdot (\boldsymbol{r}_s - \boldsymbol{r}_{s,0}) \tag{4.5}$$

where

$$\nabla \boldsymbol{\varepsilon}_c = \begin{bmatrix} \frac{\partial \varepsilon_{c,0}}{\partial x_s} & \frac{\partial \varepsilon_{c,0}}{\partial y_s} & \frac{\partial \varepsilon_{c,0}}{\partial z_s} \\ \frac{\partial \varepsilon_{c,1}}{\partial x_s} & \frac{\partial \varepsilon_{c,1}}{\partial y_s} & \frac{\partial \varepsilon_{c,1}}{\partial z_s} \\ \vdots & \vdots & \vdots \\ \frac{\partial \varepsilon_{c,N}}{\partial x_s} & \frac{\partial \varepsilon_{c,N}}{\partial y_s} & \frac{\partial \varepsilon_{c,N}}{\partial z_s} \end{bmatrix}, \quad \boldsymbol{r}_s - \boldsymbol{r}_{s,0} = \begin{bmatrix} x_s - x_{s,0} \\ y_s - y_{s,0} \\ z_s - z_{s,0} \end{bmatrix} \tag{4.6}$$

and

$$\begin{aligned} \frac{\partial \varepsilon_{c,i}}{\partial x_s} &= -2(x_i - x_s) \\ \frac{\partial \varepsilon_{c,i}}{\partial y_s} &= -2(y_i - y_s) \\ \frac{\partial \varepsilon_{c,i}}{\partial z_s} &= 2(z_i - z_s). \end{aligned} \tag{4.7}$$

The 4.3 can then be written as

$$\begin{aligned} \boldsymbol{r}_s &= \min_{\boldsymbol{r}_s} \left\{ \left[\boldsymbol{\varepsilon}_c|_{\boldsymbol{r}_{s,0}} + \nabla \boldsymbol{\varepsilon}_c|_{\boldsymbol{r}_{s,0}} \cdot (\boldsymbol{r}_s - \boldsymbol{r}_{s,0}) \right]^T \left[\boldsymbol{\varepsilon}_c|_{\boldsymbol{r}_{s,0}} + \nabla \boldsymbol{\varepsilon}_c|_{\boldsymbol{r}_{s,0}} \cdot (\boldsymbol{r}_s - \boldsymbol{r}_{s,0}) \right] \right\} \\ &= \boldsymbol{r}_{s,0} - \left(\nabla \boldsymbol{\varepsilon}_c^T|_{\boldsymbol{r}_{s,0}} \cdot \nabla \boldsymbol{\varepsilon}_c|_{\boldsymbol{r}_{s,0}} \right)^{-1} \nabla \boldsymbol{\varepsilon}_c^T|_{\boldsymbol{r}_{s,0}} \cdot \boldsymbol{\varepsilon}_c. \end{aligned} \tag{4.8}$$

The solution can be iterated as

$$\boldsymbol{r}_{s,v+1} = \boldsymbol{r}_{s,v} - \left(\nabla \boldsymbol{\varepsilon}_c^T|_{\boldsymbol{r}_{s,v}} \cdot \nabla \boldsymbol{\varepsilon}_c|_{\boldsymbol{r}_{s,v}} \right)^{-1} \nabla \boldsymbol{\varepsilon}_c^T|_{\boldsymbol{r}_{s,v}} \cdot \boldsymbol{\varepsilon}_c \tag{4.9}$$

where $v+1$ is the step number. We can consider the final source estimate $\boldsymbol{r}_s$ as the point $\boldsymbol{r}_{s,v+1}$, when $|\boldsymbol{r}_{s,v+1} - \boldsymbol{r}_{s,v}|$ it is less than or equal to a given threshold.

4.1.2 Cost Function Based on the Cone Aperture

As shown in the previous Chapter, the cone we are looking for has a 45° aperture angle. In order to have this 45° aperture angle, the distance from point $\boldsymbol{r}_i$ on cone surface to cone axis (l_1) should be equal to the distance from the apex to the point on the axis with third coordinate equal to z_i (l_2) as shown in Figure 4.1. This information can then be used in order to define a new error

$$\varepsilon_{a,i} = l_1 - l_2 = \sqrt{(x_i - x_s)^2 + (y_i - y_s)^2} - (z_i - z_s) \quad , \quad i = 0, ..., N. \tag{4.10}$$

We notice that if l_1 in equation 4.10 is equal to l_2, the error is zero, conversely an error occurs and the cone does not fit the given points $\boldsymbol{r}_i$.

The source coordinates are then given by

$$\boldsymbol{r}_s = \min_{\boldsymbol{r}_s} \left(\sum_{i=0}^{N} \varepsilon_{a,i}^2 \right) = \min_{\boldsymbol{r}_s} \left(\boldsymbol{\varepsilon}_a^T \boldsymbol{\varepsilon}_a \right) \tag{4.11}$$

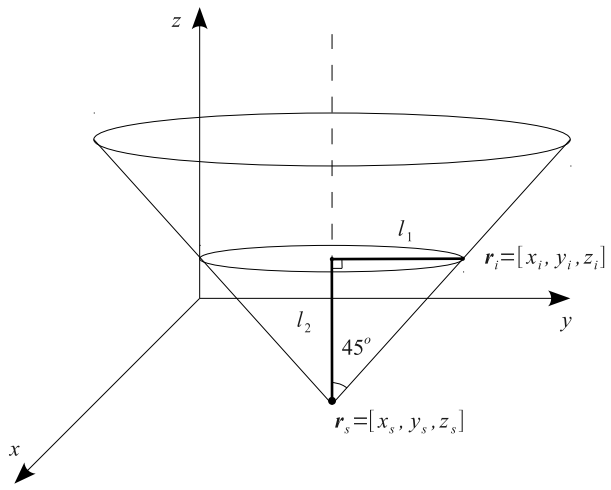


Figure 4.1: Cone Aperture Constraint: if the aperture angle measures 45°, the bold lines in figure $\boldsymbol{l}_1$ and $\boldsymbol{l}_2$ should be equal in length. $\boldsymbol{r}_s$ represents the source position, and $\boldsymbol{r}_i$ is the i-th microphone.

where

$$\boldsymbol{\varepsilon}_a = \begin{bmatrix} \varepsilon_{a,0} \\ \varepsilon_{a,1} \\ \vdots \\ \varepsilon_{a,N} \end{bmatrix}. \tag{4.12}$$

For the minimization of the cost function used in eq. 4.11, the Taylor series based method already seen (eq. 4.8) can be used.

$$\begin{aligned} \boldsymbol{r}_s &= \min_{\boldsymbol{r}_s} \left\{ \left[\boldsymbol{\varepsilon}_a|_{\boldsymbol{r}_{s,0}} + \nabla\boldsymbol{\varepsilon}_a|_{\boldsymbol{r}_{s,0}} \cdot (\boldsymbol{r}_s - \boldsymbol{r}_{s,0}) \right]^T \left[\boldsymbol{\varepsilon}_a|_{\boldsymbol{r}_{s,0}} + \nabla\boldsymbol{\varepsilon}_a|_{\boldsymbol{r}_{s,0}} \cdot (\boldsymbol{r}_s - \boldsymbol{r}_{s,0}) \right] \right\} \\ &= \boldsymbol{r}_{s,0} - \left(\nabla\boldsymbol{\varepsilon}_a^T|_{\boldsymbol{r}_{s,0}} \cdot \nabla\boldsymbol{\varepsilon}_a|_{\boldsymbol{r}_{s,0}} \right)^{-1} \nabla\boldsymbol{\varepsilon}_a^T|_{\boldsymbol{r}_{s,0}} \cdot \boldsymbol{\varepsilon}_a. \end{aligned} \tag{4.13}$$

where

$$\nabla\boldsymbol{\varepsilon}_a = \begin{bmatrix} \frac{\partial\varepsilon_{a,0}}{\partial x_s} & \frac{\partial\varepsilon_{a,0}}{\partial y_s} & \frac{\partial\varepsilon_{a,0}}{\partial z_s} \\ \frac{\partial\varepsilon_{a,1}}{\partial x_s} & \frac{\partial\varepsilon_{a,1}}{\partial y_s} & \frac{\partial\varepsilon_{a,1}}{\partial z_s} \\ \vdots & \vdots & \vdots \\ \frac{\partial\varepsilon_{a,N}}{\partial x_s} & \frac{\partial\varepsilon_{a,N}}{\partial y_s} & \frac{\partial\varepsilon_{a,N}}{\partial z_s} \end{bmatrix}, \quad \boldsymbol{r}_s - \boldsymbol{r}_{s,0} = \begin{bmatrix} x_s - x_{s,0} \\ y_s - y_{s,0} \\ z_s - z_{s,0} \end{bmatrix} \tag{4.14}$$

and

$$\begin{aligned} \frac{\partial \varepsilon_{a,i}}{\partial x_s} &= -\frac{(x_i - x_s)}{\sqrt{(x_i-x_s)^2+\sqrt{(y_i-y_s)^2}}} \\ \frac{\partial \varepsilon_{a,i}}{\partial y_s} &= -\frac{(y_i - y_s)}{\sqrt{(x_i-x_s)^2+\sqrt{(y_i-y_s)^2}}} \\ \frac{\partial \varepsilon_{a,i}}{\partial z_s} &= 1. \end{aligned} \tag{4.15}$$

The solution can be obtained iteratively as in eq. 4.9.

4.1.3 Discussion

The cost functions we deal with are both based on cone equation 4.1. For this reason they are very similar.

The most noticeable difference is given by the use of the square root for the Cone Aperture error definition $\varepsilon_{a,i}$ (equation 4.10). However, $\varepsilon_{c,i}$ is different from the square of $\varepsilon_{a,i}$. In fact, if we compute $\varepsilon_{a,i}^2$ we obtain

$$\varepsilon_{a,i}^2 = (x_i - x_s)^2 + (y_i - y_s)^2 + (z_i - z_s)^2 + 2\sqrt{(x_i - x_s)^2 + (y_i - y_s)^2}(z_i - z_s), \tag{4.16}$$

which has only some terms in common with

$$\varepsilon_{c,i} = (x_i - x_s)^2 + (y_i - y_s)^2 - (z_i - z_s)^2. \tag{4.17}$$

This is why the Taylor series expansions of the two cost functions are different. Particularly, we notice that when computing the terms

$$\begin{aligned} \frac{\partial \varepsilon_{a,i}}{\partial x_s} &= -\frac{(x_i - x_s)}{\sqrt{(x_i-x_s)^2+\sqrt{(y_i-y_s)^2}}}, \\ \frac{\partial \varepsilon_{c,i}}{\partial x_s} &= -2(x_i - x_s), \end{aligned} \tag{4.18}$$

the first one is a normalized version of the second one. This happens also to the terms related to y coordinate.

The other main difference consists in the derivative term related to z coordinate. In this case we notice that when computing

$$\begin{aligned} \frac{\partial \varepsilon_{a,i}}{\partial z_s} &= 1, \\ \frac{\partial \varepsilon_{c,i}}{\partial z_s} &= 2(z_i - z_s), \end{aligned} \tag{4.19}$$

the first term does not take account of z_i or z_s.

Now that we have noticed these differences, it makes sense to continue using both cost functions, in order to see with simulations and experiments how these differences affect results.

4.2 Degenerate Geometry Characterization

With the shown algorithms, the solution is obtained fitting given points ($\boldsymbol{r}_i$) with a cone. Some points displacements are more likely to produce better results than others, because they sample the cone in a better way. It is obvious that if we have many points, but all near in space, they sample only a little portion of the cone, and it is more difficult to find the correct cone passing through these points when noise is present.

Point locations depend only on source and microphones locations. It is then natural that the found solution to the localization problem is strongly dependent on microphones displacements around source. For this reason in order to know if these algorithms can be used with a particular array, it is important to find favorable and degenerate geometries.

We first present the mathematical problem of finding a cone equation given some points, and then we discuss some tested geometries.

4.2.1 Mathematical Approach

In order to find the minimum number of constraints needed to localize a source, we make some considerations related to the mathematical approach to cone fitting.

We need to constrain a cone of equation

$$(x - x_s)^2 + (y - y_s)^2 - (z - z_s)^2 = 0 \tag{4.20}$$

to pass for given points $\boldsymbol{r}_i$'s, in order to find x_s and y_s. Expanding the equation of the cone leads to

$$x^2 + y^2 + z^2 - 2x_s x - 2y_s y + 2z_s z + x_s^2 + y_s^2 - z_s^2 = 0, \tag{4.21}$$

or in matrix form

$$\left[x^2, y^2, z^2, x, y, z, 1\right] \begin{bmatrix} 1 \\ 1 \\ 1 \\ -2x_s \\ -2y_s \\ 2z_s \\ x_s^2 + y_s^2 - z_s^2 \end{bmatrix} = 0. \tag{4.22}$$

We can then solve the following system

$$\begin{bmatrix} x_1^2, y_1^2, z_1^2, x_1, y_1, z_1, 1 \\ x_2^2, y_2^2, z_2^2, x_2, y_2, z_2, 1 \\ x_3^2, y_3^2, z_3^2, x_3, y_3, z_3, 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ -2x_s \\ -2y_s \\ 2z_s \\ x_s^2 + y_s^2 - z_s^2 \end{bmatrix} = \mathbf{0}. \tag{4.23}$$

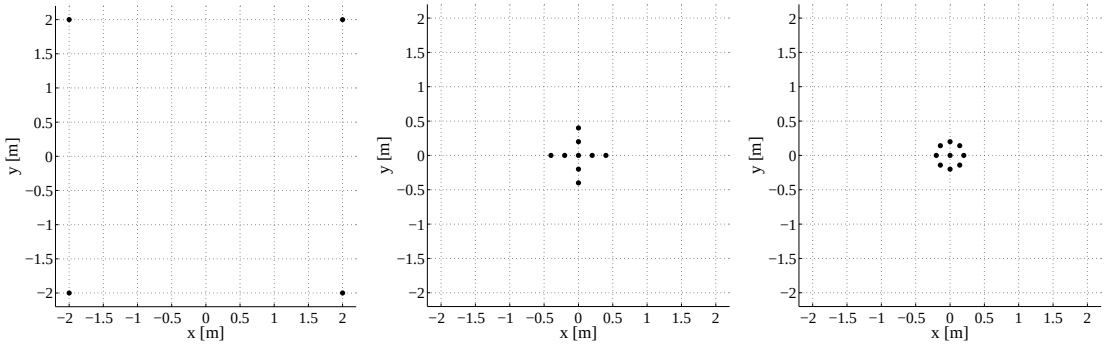
in order to find the unknowns x_s, y_s and z_s.

We notice that only three non collinear points $\boldsymbol{r}_i$ are needed. In fact the cone we are searching for has less degrees of freedom than a generic cone. This is due to the fact that we constrain the axis to lie on a line whose orientation is known (perpendicular to xy plane), and the aperture angle is constrained to be 45°.

4.2.2 Case Studies

We have just shown that at least three non collinear points are needed in order to localize the vertex of the cone, so we can use arrays of three microphones, or more in order to overcome noise. As mentioned before, also array geometry is important in order to sample the cone and use cone algorithms.

For this purpose we show how cost functions used in equation (4.3) and equation (4.11) change using a cross array, a circular array, and a squared array when the source is placed in different positions respect to the array. The three array geometry are shown in Figure 4.2.



(a) Square Array (b) Cross Array (c) Circular Array

Figure 4.2: Tested Arrays: (a) is the square arreay, (b) is the cross array, and (c) is the circular array. Dots represent microphones on xy plane where source is moved. All measures are in meters.

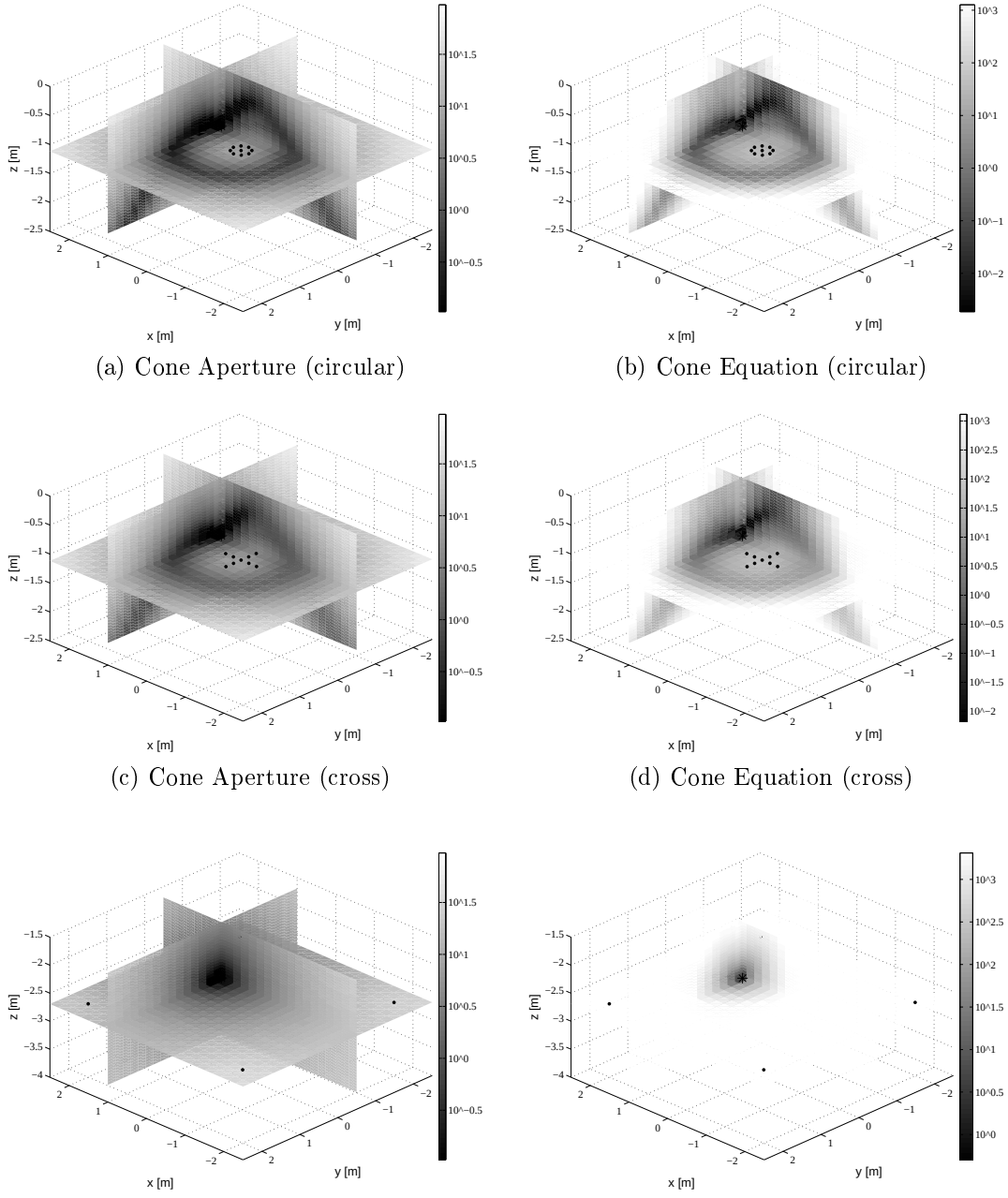
Cost functions related to different geometries for a source at $[x_s, y_s] = [1, -0.5]$ are shown in Figure 4.3.

We notice that with the square array, both cost functions are monotonically decrescent in a big area around the minimum. Using a cross or circular array leads to functions difficult to minimize. This is due to the fact that the square array cover a big area on xy plane and the source lies in this area. In this way microphones sample the cone in a better way, taking cone points all around cone axis. If the cone is sampled only from a side and not all around the axis, finding the cone equation from samples become more difficult.

Another good aspect of the used square array is that microphones are far from each other. In Figure 4.4 we show how cost functions behave when the source is located outside the square array area ($[x_s, y_s] = [3, 4]$).

We can notice that also if source is located outside the area covered by microphones, the minimum is easy to be found. In fact taking microphones far from each other, we ensure a better cone sampling.

In conclusion the best solution seems to be given by the square array with microphones far from each other, because the source is forced to lie within the array area and microphones sample far points.



(a) Cone Aperture (circular)

(b) Cone Equation (circular)

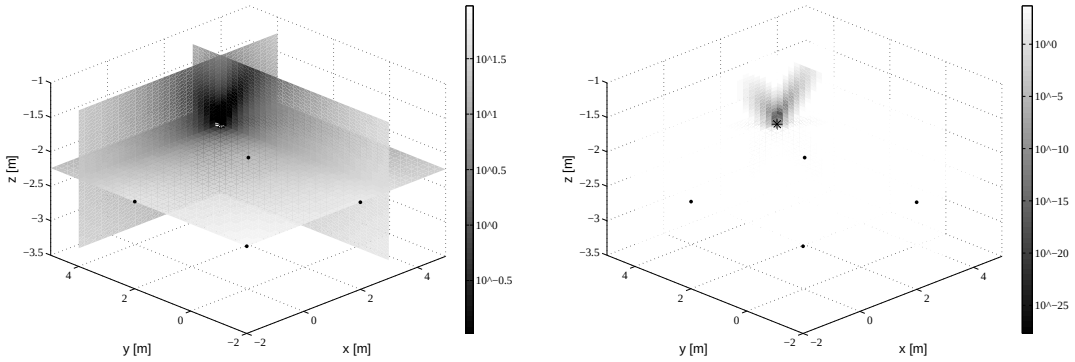
(c) Cone Aperture (cross)

(d) Cone Equation (cross)

(e) Cone Aperture (square)

(f) Cone Equation (square)

Figure 4.3: Cost Functions: Cone Aperture and Cone Equation cost functions for circular (a) (b), cross (c) (d), and square (e) (f) arrays in logarithmic scale. The small dots represent arrays locations on xy plane, the big dot centered in the black area represents cost function minimum. We notice that in (e) and (f) the minimum is well localized. For this reason we prefer the square array when used with cone-based algorithms.



(a) Cone Aperture (square) (b) Cone Equation (square)

Figure 4.4: Source Outside Array Area: also if source does not lie into array covered area, Cone Aperture (a) and Cone Equation (b) cost functions have an absolute minimum well defined. Small dots represent arrays locations on xy plane, the big dot represents cost function minimum.

4.3 Conclusions

In this Chapter we showed how to use cone constraints in order to find a solution to localization problem. We also showed that at least three microphones (points) are needed in order to find the cone and thus the source location, and that some geometries are more favorable than others. Next step consists in extending these algorithms in order to infer information about reflectors, using information of reflected signal.

Chapter 5

Extension to Multi-Source Localization

In previous Chapters we have only considered the presence of one source at a time for the localization problem.

In this Chapter we show how we can deal with the multi-source case with cone-based algorithms, considering the case of two sources emitting different signals. Algorithms can be easily extended to work with more than two sources.

We first give a brief overview of data model in order to understand TDOAs disambiguation problem. Finally we present algorithms to overcome this problem and localize the two sources.

5.1 Data Model

Dealing with more than one source, we have already shown in Chapter 2 that the signal received from i-th microphone is

$$x_i(t) = \sum_{a=1}^{M} h_{a,i} s_a(t - \tau_{a,i}), \tag{5.1}$$

where each of M sources emits the white gaussian signal $s_a(t)$, $a = 1, ..., M$, $h_{a,i}$ is the attenuation term and $\tau_{a,i}$ is the time of arrival from source a to the i-th microphone.

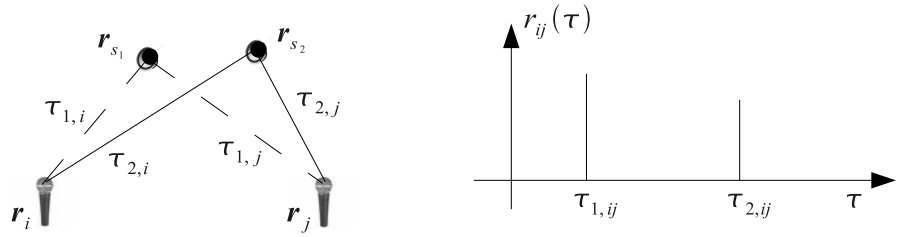


Figure 5.1: Multi-Source Cross-Correlation: a simple setup with two microphones ($\boldsymbol{r}_i$,$\boldsymbol{r}_j$) and two sources ($\boldsymbol{r}_{s_1}$,$\boldsymbol{r}_{s_2}$). Time of arrival for each couple source-microphone is represented by $\tau_{a,i}$. An example of how cross-correlation should be is also given.

The cross-correlation function of two sensors signals $x_i(t)$ and $x_j(t)$

$$r_{ij}(\tau) = E[x_i(\tau + t_0)x_j(t_0)] \tag{5.2}$$

exhibits a peak for each source, as shown in Figure 5.1. In particular the peaks are located at $\tau_{a,ij} = \tau_{a,i} - \tau_{a,j}$, $a = 1, ..., M$.

5.2 TDOA Disambiguation

As shown in Chapter 2, we can measure TDOAs from cross-correlation peaks, but we do not know which source each TDOA belongs to. This problem is referred to as TDOA disambiguation problem.

For cone-based algorithms, the solution with four microphones on the vertices of a rectangle is a good array solution. In this situation, if only a few sources are present in a non reverberant room, the TDOAs combinations are few.

For example, let us consider the case of having two sources and four microphones as in Figure 5.2, and we work with a reference microphone. If we call $\tau_{a,i}$ the TOA from source i to microphone j, and $\tau_{a,ij}$ the TDOA computed as $\tau_{a,i} - \tau_{a,j}$, the possible TDOAs combinations are only four:

1. $[\tau_{1,01}, \tau_{1,02}, \tau_{2,03}]$ for a source, and $[\tau_{2,01}, \tau_{2,02}, \tau_{1,03}]$ for the other;

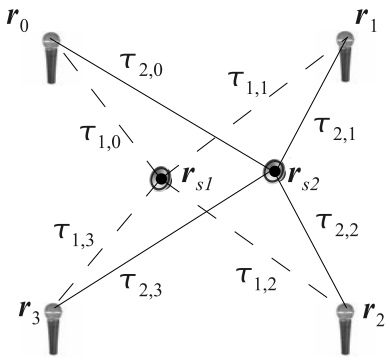


Figure 5.2: Multi-Source TDOAs; $\tau_{a,i}$ represents time of arrival from source a to microphone i, $\boldsymbol{r}_i$ for $i = 0, ..., 3$ represent microphones, and $\boldsymbol{r}_{s_1}$ and $\boldsymbol{r}_{s_2}$ are the two sources. All the direct paths are drawn.

2. $[\tau_{1,01}, \tau_{1,02}, \tau_{1,03}]$ for a source, and $[\tau_{2,01}, \tau_{2,02}, \tau_{2,03}]$ for the other;

3. $[\tau_{1,01}, \tau_{2,02}, \tau_{1,03}]$ for a source, and $[\tau_{2,01}, \tau_{1,02}, \tau_{2,03}]$ for the other;

4. $[\tau_{1,01}, \tau_{2,02}, \tau_{2,03}]$ for a source, and $[\tau_{2,01}, \tau_{1,02}, \tau_{1,03}]$ for the other.

It is obvious that only the second combination leads to the correct source localization, while the others lead to wrong sources locations. Working with so few TDOAs combinations, we propose a method based on the evaluation of cone-based cost functions, in order to select the correct TDOAs combination.

Of course if the number of sources increases, a brute force method can not be used anymore. For this purpose we also show a method based on DATEMM consideration about zero-sum condition, already briefly described in Chapter 2.

5.3 Algorithms

In this Section we first show a brute force algorithm which can be used under ideal conditions and when only a few sources and microphones are present, then we also propose a more realistic and robust algorithm based on DATEMM.

Brute Force Approach The algorithm is based on the idea that both Cone Aperture and Cone Equation cost functions should be minimized in order to find a source location. This means that the more the cost function approaches the 0 value, the more the probability of correctly finding a source increases.

We can then use a cone algorithm to locate a candidate source position for each TDOAs combination. Then we can easily find which combination is the correct one just by comparing the cost function values.

The algorithm is so composed by a few steps. For the case of two sources and four microphones these steps are described in Algorithm 1. Cost functions definitions for the above case are also given in Algorithm 1.

Algorithm 1 Brute Force Approach Algorithm: following these steps we can localize the two sources positions evaluating each TDOAs pair.

1: we compute TDOAs from cross-correlations of signal received by microphones;
2: we compute every TDOAs combination;
3: we apply a cone-based algorithm to every TDOAs combination in order to find possible sources locations $\boldsymbol{r}_{s1}$ and $\boldsymbol{r}_{s2}$;
4: we evaluate the cone cost function for every TDOAs combination and sources locations found as

$$J = \sum_{i=0}^{3} \Big((x_i - x_{s1})^2 + (y_i - y_{s1})^2 - (z_i - z_{s1})^2 + (x_i - x_{s2})^2 + (y_i - y_{s2})^2 - (z_i - z_{s2})^2 \Big) \tag{5.3}$$

if we are using Cone Equation algorithm, or

$$J = \sum_{i=0}^{3} \Big(\sqrt{(x_i - x_{s1})^2 + (y_i - y_{s1})^2} - (z_i - z_{s1}) + \sqrt{(x_i - x_{s2})^2 + (y_i - y_{s2})^2} - (z_i - z_{s2}) \Big); \tag{5.4}$$

if we are using Cone Aperture one;
5: we choose as correct source locations those who give the minimum value of the cost function J.

Zero-Sum Condition A smarter solution is given by working on TDOAs combinations, in order to discard all the wrong ones before applying a localization algorithm.

For this purpose we propose a slightly modified version of DATEMM algorithm, in order to simplify it. In fact we are not considering the case of a reverberant room, and we have fixed the number of used microphones to four. For this reason we do not need to use the full algorithm.

In the ideal case, using two sources, we should find two peaks for each cross-correlation. Working with four microphones, we need to compute six cross-correlations, one for each microphones pair.

The zero-sum condition states that for each microphones triplet, the sum of the three TDOAs associated to a source and these microphones should be zero. Also the sum of TDOAs between all four microphones should sum to zero, if they are associated to the same source.

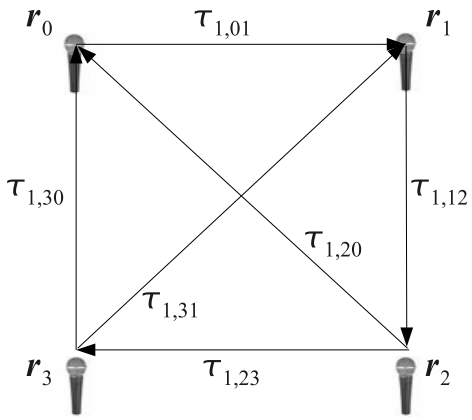


Figure 5.3: Multi-Source DATEMM: TDOAs graph for a four microphones and two sources setup. $\boldsymbol{r}_i$ for $i = 0, ..., 3$ represent microphones, each branch represents a TDOA from a source to the two linked microphones. In this case we have associated TDOAs belonging to source number one to each branch. This way the graph verify the zero-sum condition.

For example, for the configuration of four microphones and two sources, we can build a graph for each source (Figure 5.3) where nodes represent microphones, while branches represent TDOAs to the linked microphones for the considered source.

According to Figure 5.1 we can define all the TDOAs for the first source as

$$\begin{aligned}
\tau_{1,01} &= \tau_{1,0} - \tau_{1,1}, \\
\tau_{1,12} &= \tau_{1,1} - \tau_{1,2}, \\
\tau_{1,23} &= \tau_{1,2} - \tau_{1,3}, \\
\tau_{1,30} &= \tau_{1,3} - \tau_{1,0}, \\
\tau_{1,31} &= \tau_{1,3} - \tau_{1,1}, \\
\tau_{1,20} &= \tau_{1,2} - \tau_{1,0}.
\end{aligned}$$

In this case, if we associate the correct TDOAs to the considered source, the conditions to be satisfied for triplets and for the four microphones are

1. $\tau_{1,01} + \tau_{1,12} + \tau_{1,20} = (\tau_{1,0} - \tau_{1,1}) + (\tau_{1,1} - \tau_{1,2}) + (\tau_{1,2} - \tau_{1,0}) = 0$;
2. $\tau_{1,12} + \tau_{1,23} + \tau_{1,31} = (\tau_{1,1} - \tau_{1,2}) + (\tau_{1,2} - \tau_{1,3}) + (\tau_{1,3} - \tau_{1,1}) = 0$;
3. $\tau_{1,23} + \tau_{1,30} - \tau_{1,20} = (\tau_{1,2} - \tau_{1,3}) + (\tau_{1,3} - \tau_{1,0}) - (\tau_{1,2} - \tau_{1,0}) = 0$;
4. $\tau_{1,30} + \tau_{1,01} - \tau_{1,31} = (\tau_{1,3} - \tau_{1,0}) + (\tau_{1,0} - \tau_{1,1}) - (\tau_{1,3} - \tau_{1,1}) = 0$;
5. $\tau_{1,01} + \tau_{1,12} + \tau_{1,23} + \tau_{1,30} = (\tau_{1,0} - \tau_{1,1}) + (\tau_{1,1} - \tau_{1,2}) + (\tau_{1,2} - \tau_{1,3}) + (\tau_{1,3} - \tau_{1,0}) = 0$.

If we find the set of TDOAs respecting all the five conditions, we have found a set of TDOAs referred to the same source.

If we can not find this combination, we search for a combination of TDOAs respecting only condition five, because if we find this combination, we can derive all the TDOAs from the four external ones.

If we cannot find this combination, it means that we have not correctly extracted peaks from cross-correlations. In this case we need to study microphones triplets in order to restore missing peaks.

Table (5.1) explains when we can use zero-sum condition in order to find missing peaks supposing that only a cross-correlation between the six computed is corrupted. The algorithm steps are summarized in Algorithm 2.

Algorithm 2 Zero-Sum Condition Algorithm: following these steps we can easily find which TDOAs combinations respect the zero-sum condition.

```
1: we compute all the cross-correlations and extract two peaks for each one.
2: we search for a TDOAs combination respecting all the zero-sum conditions
   (conditions from one to five) imposing that TDOAs sums should not exceed
   a given threshold.
3: if this combination does not exist then
4:    we search for a combination which respects at least condition five.
5:    if no combinations respect condition five then
6:       we search for a combination leading to a case shown in Table (5.1).
7:       if this combination is found then
8:          we restore the corrupted TDOA according to Table (5.1).
9:       end if
10:   end if
11: end if
12: if one or more combinations respect the zero-sum condition then
13:   we pick the one whose sum is closer to zero, and we have the TDOAs
      associated to a source.
14: else
15:   the algorithm exits with no success.
16: end if
```

Table 5.1: Zero-Sum Restoration: this table shows the relation between zero-sum conditions. When only one TDOA between $\tau_{1,01}$, $\tau_{1,12}$, $\tau_{1,23}$, $\tau_{1,30}$ is corrupted, it can be still restored.

	Cond 1	Cond 2	Cond 3	Cond 4	Cond 5	Wrong Peak Restored
respected	yes	yes	yes	yes	yes	-
	no	yes	yes	no	no	$\tau_{1,01} = -(\tau_{1,12} + \tau_{1,20})$
	no	no	yes	yes	no	$\tau_{1,12} = -(\tau_{1,23} + \tau_{1,31})$
	yes	no	no	yes	no	$\tau_{1,23} = -(\tau_{1,12} + \tau_{1,31})$
	yes	yes	no	no	no	$\tau_{1,30} = \tau_{1,31} - \tau_{1,01}$

5.4 Conclusions

We have shown in this Chapter that we can use a brute force method, or a more robust algorithm based on DATEMM in order to overcome TDOA disambiguation problem. The shown algorithms work when cross-correlations have a number of peaks equal to the number of sources. In practice this condition is satisfied when sources are independents and emit different signals.

In the next Chapter we show how to solve TDOA disambiguation when two synchronized sources emit the same signal which is the case found studying the inference problem.

Chapter 6

Application to Reflector Localization

In the previous Chapters we have presented some algorithms used for source localization problem when microphones record the signal emitted by one audio source or several ones in a dry room. However, if a reflector is present in the scene each microphone record the direct and indirect signal. As mentioned in Chapter 2, the presence of the reflected signal can be treated with the image source technique. In doing so, we deal with a situation similar to the multi-source one, where the second source in the scene is the image source.

In this Chapter we first show problematics derived from having TDOAs from direct and indirect path, and then we show how algorithms previously presented can be easily adapted to reflector localization.

6.1 Direct and Indirect Paths

If only a reflector is present, each microphone records the direct signal and the reflected one. For a single source signal propagating on two paths to microphone i, we obtain the signal

$$x_i(t) = h_{i,0}s(t - \tau_{i,0}) + h_{i,1}s(t - \tau_{i,1}), \tag{6.1}$$

where $s(t)$ is the source signal, $h_{i,0/1}$ and $\tau_{i,0/1}$ are the amplitude and delay of direct/indirect path between the source and i-th microphone (Figure 6.1). If $s(t)$

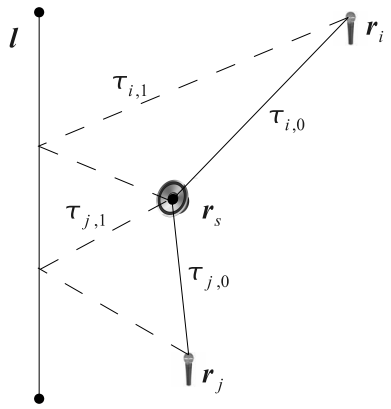


Figure 6.1: Direct and Indirect Path: $\boldsymbol{r}_s$ is the real source, $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ are two microphones, and $\boldsymbol{l}$ is the line where the reflector lies. The direct paths TOAs are $\tau_{i,0}$ and $\tau_{j,0}$, the indirect path TOAs are $\tau_{i,1}$ and $\tau_{j,1}$.

is zero mean and white, the cross-correlation $r_{ij}(\tau) = E[x_i(\tau + t)x_j(t)]$ between the two microphones signals $x_i(t)$ and $x_j(t)$ will exhibit up to four extrema at TDOA positions

$$\tau_{ij,\mu\nu} = \tau_{i,\mu} - \tau_{j,\nu}, \mu, \nu \in \{0, 1\}. \tag{6.2}$$

However there is a difference between this case and multi-source one previously analyzed. Since the two sources emit the same signal, the number of peaks in cross-correlation function increases. We can then work as if two sources were present, the real source and the image one, but keeping in mind that they are synchronized and emit the same signal.

As an example, working with one source, four microphones, and one reflector, leads up to four peaks for each of the six cross-correlations. Computing all the possible combinations of TDOAs in order to use a brute force algorithm as the one shown for multi-source localization is now a long operation.

In order to assign the correct peaks to real and image sources we use a method based on considerations about peaks position in cross- and auto-correlation made in [13].

The cross-correlation extrema appear at well-defined distances, which can be

predicted by the extrema positions in the auto-correlations. Let

$$\begin{aligned} x_i(t) &= h_{i,0}s(t-\tau_{i,0}) + h_{i,1}s(t-\tau_{i,1}), \\ x_j(t) &= h_{j,0}s(t-\tau_{j,0}) + h_{j,1}s(t-\tau_{j,1}), \end{aligned} \tag{6.3}$$

be two microphone signals. The autocorrelations r_{ii} and r_{jj} show, in addition to the zero-lag extrema $r_{ii}(0)$ and $r_{jj}(0)$, four other extrema at the positions

$$\begin{aligned} \tau_{ii,10} &= \tau_{i1} - \tau_{i0}, \\ \tau_{ii,01} &= \tau_{i0} - \tau_{i1}, \\ \tau_{jj,10} &= \tau_{j1} - \tau_{j0}, \\ \tau_{jj,01} &= \tau_{j0} - \tau_{j1}. \end{aligned} \tag{6.4}$$

They coincide with the differences of the cross-correlation extrema

$$\begin{aligned} \tau_{ii,10} &= \tau_{i1} - \tau_{i0} = (\tau_{i1} - \tau_{j0}) - (\tau_{i0} - \tau_{j0}) \\ &= \tau_{ij,10} - \tau_{ij,00} > 0, \\ \tau_{jj,10} &= \tau_{j1} - \tau_{j0} = (\tau_{i1} - \tau_{j0}) - (\tau_{i1} - \tau_{j1}) \\ &= \tau_{ij,10} - \tau_{ij,11} > 0. \end{aligned} \tag{6.5}$$

This condition is referred to as the raster condition.

Since the direct path always has the shortest delay, $\tau_{ii,10}$ and $\tau_{jj,10}$ are positive. This implies for the sensor i that the cross-correlation extremum $\tau_{ij,1x}$ of the echo path is always right to the extremum $\tau_{ij,0x}$ of the direct path 0. In contrast, for the second sensor j, the extremum of the echo path is left to the extremum of the direct path. In Figure 6.2, these relationships are illustrated by the arrows below r_{ij}.

We conclude that each arrow points from the direct path extremum to the echo path extremum for the corresponding sensor. Clearly, the direct path TDOA $\tau_{ij,00}$ is that extremum in r_{ij} which shows only arrow tails, while the indirect path TDOA $\tau_{ij,11}$ shows only arrowheads.

This raster matching approach combines the extremum positions of both auto- and cross-correlations and enables us to identify the desired direct path TDOA

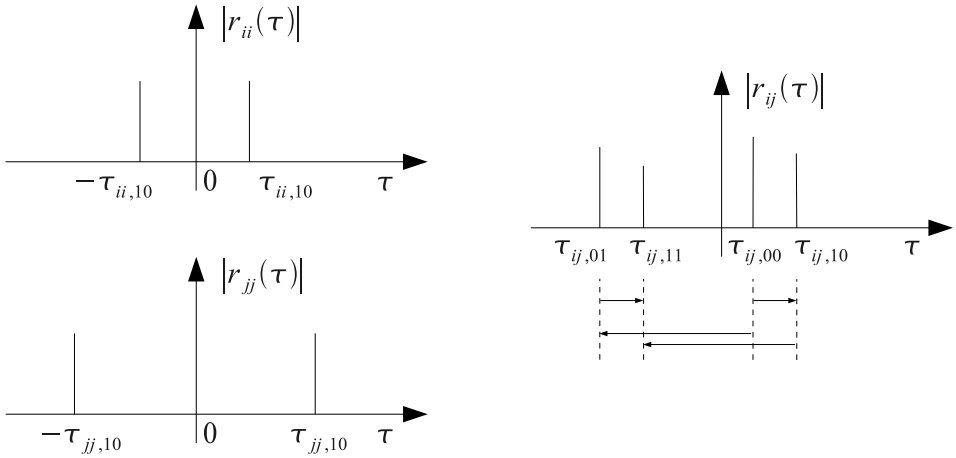


Figure 6.2: Raster Condition: the two auto-correlations of signal received by microphones i and j on the left, and the cross-correlation of this two signals on the right. In finding which differences between cross-correlation peaks coincide with extrema positions of auto-correlations, and taking account of these differences signs, we can understand which cross-correlation peak is the TDOA of direct path, and which is the TDOA of the indirect path.

6.2 Algorithms

When TDOAs related to real and image sources are separated, we decompose reflector localization problem into separate source localization problems. In fact as soon as real and image sources are located, the reflector is constrained to lie on the line perpendicular to the segment joining the two sources and passing for its middle point (Figure 6.3).

For this reason the problem is solved in three steps:

- first TDOAs are extracted from cross-correlation peaks using the raster condition,
- then real and image source are located with either Cone Equation or Cone Aperture algorithms,
- finally the location of the reflector is found.

In order to extract TDOAs we proceed with the following steps:

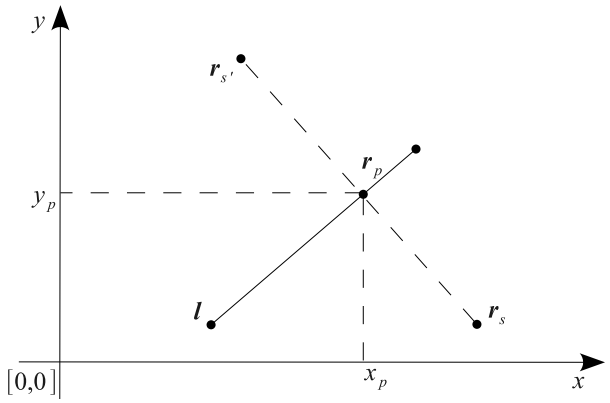


Figure 6.3: Reflector Geometry: $\boldsymbol{r}_s$ is the real source, $\boldsymbol{r}'_s$ is the image source $\boldsymbol{r}_p$ is the reflection point, and $\boldsymbol{l}$ is the line where the reflector lies.

- we compute auto-correlations for each microphone,

- we compute cross-correlations for each microphone pair,

- we apply the raster condition to each cross-correlation in order to find direct and indirect path peaks.

Applying the raster condition simply means computing the difference between every cross-correlation peaks pairs, and search for those peaks whose location differences correspond to an autocorrelation peak location.

After these few steps we have TDOAs assigned to real and image sources, and we can proceed applying a cone-based algorithm for source localization. After this second step, we can finally derive the reflector position.

As we show in Chapter 2, in order to define the line where the reflector lies, only two parameters are needed (m and q). In order to find these two parameters, we refer to the geometry depicted in Figure 6.3.

The reflection point $\boldsymbol{r}_p$, which is known to be the middle point of line joining $\boldsymbol{r}_s$ and $\boldsymbol{r}_{s'}$, can be found from real and image source location as

$$\boldsymbol{r}_p = \left[\frac{x_s + x_{s'}}{2}, \frac{y_s + y_{s'}}{2}\right]^T . \tag{6.6}$$

The angular coefficient of the line joining $\boldsymbol{r}_s$ and $\boldsymbol{r}_{s'}$ is

$$m' = \frac{y_s - y_{s'}}{x_s - x_{s'}}. \tag{6.7}$$

The reflector line is perpendicular to line joining real and image sources, so its angular coefficient is

$$m = -\frac{1}{m'} = -\frac{x_s - x_{s'}}{y_s - y_{s'}}. \tag{6.8}$$

In order to find the parameter q, we impose that the line $y = mx + q$ passes through the point $\boldsymbol{r}_p$, and we obtain

$$q = \frac{y_s + y_{s'}}{2} - m \cdot \frac{x_s + x_{s'}}{2}. \tag{6.9}$$

6.3 Conclusions

In this Chapter we have shown that it is possible to use cone-based algorithms also in presence of one reflector. We have first shown that TDOAs disambiguation can be solved with a simple method found in the literature in this case. Then we have described how to perform reflector inference from these pre-processed measurements.

Now that we have shown all the algorithms, we can focus on simulations and experimental results for validation purposes.

Chapter 7

Experimental Results

In this Chapter we show simulation and experimental results obtained using the shown algorithms. These results are compared to those obtained by using localization algorithms found in the literature in order to justify the proposed approach.

First we present the hardware used in laboratory for experiments. Then we show results for simulations and experiments on single and multi-source localization problem. Finally we show simulations for reflector localization algorithms.

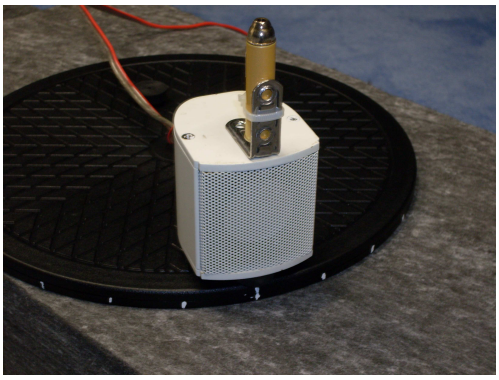
7.1 Experimental Hardware

The hardware we use in our experiments includes one small loudspeaker (Figure 7.1a and 7.1b) (for single source cases), an octagonal speaker array (for multi-source purpose) and several microphones (Figure 7.1c and 7.1d), obviously together with a sound card connected to a computer. The single loudspeaker needs to be powered by an amplifier. When we use only one source, we move it by hand in order to perform more tests and record multiple acquisitions. For multi-source experiments, we light on and off different pairs of sources from the array.

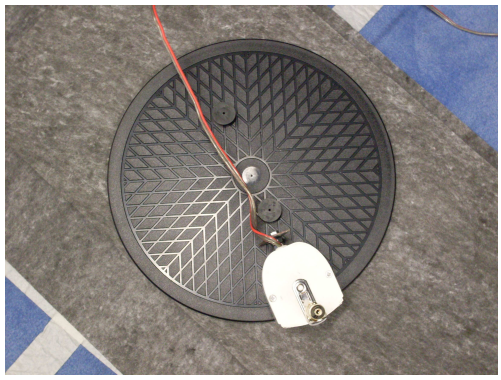
Receivers are *"Beyerdynamic MM1"* microphones whose specifications are described in Table 7.1. We use 4 of them for each experiment fixed on different bars. Main features of these microphone are:

- Linear Frequency response

- Omnidirectional Polar Pattern



(a) *Loudspeaker(zooming).*



(b) *Loudspeaker(top view).*

(c) *Microphones(zooming).*

(d) *Microphones(configuration).*

Figure 7.1: Used Hardware: some examples of hardware.

- Calibrated Open Circuit Voltage

- Narrow Tubular Construction

The analog-to-digital/digital-to-analog converter, whose specification are widely described in Tables 7.2, 7.3, 7.4, 7.5, 7.6, 7.7, is an *"Aurora Lynx 16"* and it is used together with the *"Focusrite Octopre LE"* soundcard.

This soundcard allows several sampling frequencies: 44.1, 48, 88.2, 96, 176.4, 192 kHz. We work to 44.1 kHz, to check the effectiveness of the algorithm in the worst case.

Table 7.1: Experimental setup: microphone specifications.

Model	*Beyerdynamic MM1*
Operating principle	Pressure
Transducer type	Condenser (back electret)
Operating principle	Pressure
Polar pattern	Omnidirectional, diffuse field calibrated
Open circuit voltage at 1 kHz (0dB = 1V/Pa)	15 mV/Pa (= -36.5 dBV) ±1 dB
Nominal impedance	330 ohm
Load impedance	2.2 kohm
Connector	3-pin XLR
Length	133 mm
Shaft diameter	19 mm
Head diameter	9 mm
Weight without cable	88 g
Frequency response	20 - 20.000 Hz (50 - 16.000 Hz ±1.5 dB)
Max. SPL at 1 kHz	128 dB
S/N ratio rel. to 1 Pa	$>$ 57 dB
A-weighted equivalent SPL	approx. 28 dB(A)
Power supply	12 - 48 V phantom power
Current consumption	approx. 3.4 mA

Table 7.2: Experimental setup: *Aurora Lynx 16* specifications (Analog I/O).

Analog I/O	
Aurora 16	Sixteen inputs and sixteen outputs
Type	Electronically balanced or unbalanced
Level	+4 dBu nominal / +20 dBu max. or -10 dBV nominal / +6 dBV max
Input Impedance	Balanced mode: 24 kohm Unbalanced mode: 12 kohm
Output Impedance	Balanced mode: 100 ohm Unbalanced mode: 50 ohm
Output Drive	600 ohm impedance, $0.2\mu F$ capacitance
A/D and D/A Type	24-bit multi-level, delta-sigma

Table 7.3: Experimental setup: *Aurora Lynx 16* specifications (Analog In Performance).

Analog In Performance	
Frequency response	20 Hz - 20 kHz, +0/-0.1 dB
Dynamic range	117 dB, A-weighted
Channel crosstalk	-120 dB maximum, 1 kHz signal, -1 dBFS
THD + N	-108 dB (0.0004%) @ -1 DBFS -104 dB (0.0006%) @ -6 DBFS 1 kHz signal, 22 Hz - 22 kHz BW

Table 7.4: Experimental setup: *Aurora Lynx 16* specifications (Analog Out Performance).

Analog Out Performance	
Frequency response	20 Hz - 20 kHz, +0/-0.1 dB
Dynamic range	117 dB, A-weighted
Channel crosstalk	-120 dB maximum, 1 kHz signal, -1 dBFS
THD + N	-107 dB (0.0004%) @ -1 DBFS -106 dB (0.0006%) @ -6 DBFS 1 kHz signal, 22 Hz - 22 kHz BW

Table 7.5: Experimental setup: *Aurora Lynx 16* specifications (Digital I/O).

Digital I/O	
Number / Type	16 inputs and 16 outputs 24 bit AES/EBU format, transformer coupled
Channels	16 in/out in single-wire mode 8 in/out in dual-wire mode
Samples Rates	All standard rates and variable rates up to 192 kHz in both single-wire and dual-wire modes

Table 7.6: Experimental setup: *Aurora Lynx 16* specifications (On-board Digital Mixer).

On-board Digital Mixer	
Type	Hardware-based, low latency
Routing	Ability to route any input to any or multiple outputs
Mixing	Up to four input or playback signals mixed to any output, 40-bit precision
Status	Peak levels to -114 dB on all inputs and outputs

Table 7.7: Experimental setup: *Aurora Lynx 16* specifications (Connections).

CONNECTIONS	
Digital I/O Ports	25-pin female D-sub connectors. Port A: channels 1-8 I/O, Port B: channels 9-16 I/O. Yamaha pinout standard
Analog I/O Ports	25-pin female D-sub connectors. Analog In 1-8, Analog In 9-16, Analog Out, 1-8 Analog Out 9-16. Tascam pinout standard
External Clock	75-ohm BNC word clock input and output
MIDI	One input and one output. Standard opto-isolated, 5-pin female DIN connectors

7.2 Evaluation Methodology

This section shows the main aspects of methodologies used for simulations and experiments. In particular we explain how simulations and experiments are conducted, as well as the metrics used for results analysis.

For simulations and experiments, the estimation of a source location is performed many times in order to propose statistical results. With regard to simulations, this is done building a set of noisy TDOAs for each source position, and testing each algorithm with all the corrupted TDOAs. With regard to experiments, the same result is obtained recording a continuous signal and windowing it in many frames. TDOAs are then extracted from each frame, and this allows us to build the noisy TDOAs set.

The accuracy of source localization algorithms is measured with the following metrics:

- *mean bias on x*:

$$b_x = \left| \frac{1}{n} \sum_{i=1}^{n} (x_s - \hat{x}_{s,i}) \right|, \tag{7.1}$$

  where n is the number of noisy TDOAs tested for a source location, x_s is the source's x coordinate, and $\hat{x}_{s,i}$ is the estimated source one based on the i-th TDOAs realization. This is the measure of the absolute value of the difference between x coordinate of the real source and the estimated one.

- *RMSE on x*:

$$R_x = \left(\frac{1}{n-1} \sum_{i=1}^{n} \left((\varepsilon_i) - (\overline{\varepsilon}) \right)^2 \right)^{\frac{1}{2}}, \tag{7.2}$$

  where n is the number of noisy TDOAs tested for a source location, ε_i is the difference between the source location x coordinate x_s and the estimated one with the i-th TDOAs realization $\hat{x}_{s,i}$, and $\overline{\varepsilon}$ is the mean error computed as

$$\overline{\varepsilon} = \frac{1}{n} \sum_{i=1}^{n} (x_s - \hat{x}_{s,i}). \tag{7.3}$$

  This is the root mean squared error obtained on estimated value of source's x coordinate on many simulations.

For the reflector localization problem, we have defined in Chapter 2 the reflector as all points $[x, y]$ satisfying the condition $y = mx + q$. For this reason

two additional metrics are used in order to evaluate parameters m and q:

- *mean angle error*:

$$\Delta\alpha = \frac{180}{\pi}\left(\frac{1}{n}\sum_{i=1}^{n}\arctan(m-\hat{m}_i)\right), \tag{7.4}$$

  where n is the number of noisy TDOAs tested for a source location, m is the angular coefficient of the line representing the reflector, and $\hat{m}_i$ is its estimation based on the i-th TDOAs realization. This is the difference in degrees between real and estimated reflector azimuth angle.

- *mean distance error*:

$$\Delta q = \frac{1}{n}\sum_{i=1}^{n}(q-\hat{q}_i), \tag{7.5}$$

  where n is the number of noisy TDOAs tested for a source location, q is the intercept between the y axis and the line representing the reflector, and $\hat{q}_i$ is its estimation based on the i-th TDOAs realization. This is the mean difference between the intercept between the y axis and the line representing the reflector and its estimation.

7.3 Localization

This Section shows a comparison between results obtained with the cone proposed algorithms and other algorithms found in the literature for the problem of source localization. First we present results obtained with simulations, then we present results from a real experiment.

7.3.1 Simulations

In this Paragraph we show some simulations to illustrate how cone algorithm performs with respect to others found in the literature.

The simulation setup is described in Figure 7.2:

- the array is made up of 4 microphones placed on the vertices of a square measuring $(4m \times 4m)$,

- the source is moved on 81 points on a square grid centered into the array.

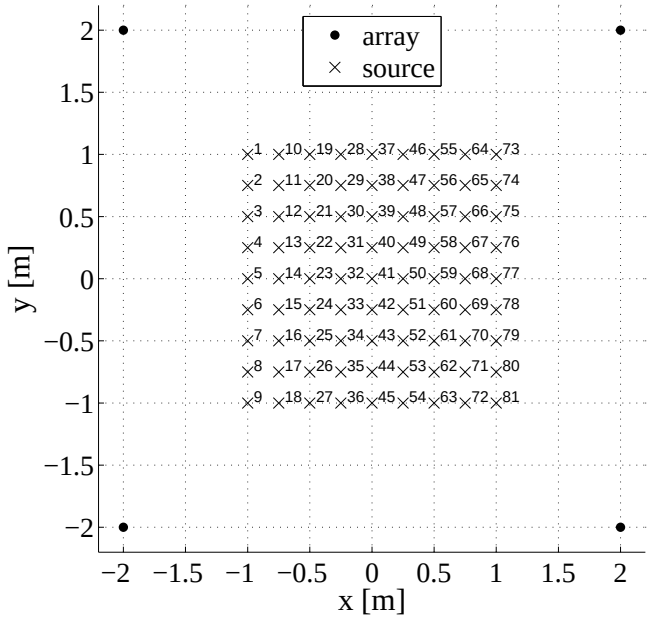


Figure 7.2: Simulation Setup: asterisks represent microphones, crosses represent the grid where sources were located. Each source location is identified by a number. All distances are in meters.

For each source location, 50 different set of noisy TDOAs were tested in order to find a reasonable statistic. Each TDOA is computed from range difference of arrival corrupted with zero mean and 2 cm standard deviation gaussian noise.

Figure 7.6 shows the b_x and R_x comparison between different algorithms. In particular we tested Gillette-Silverman (GS) [2], Linear Correction Least Square (LCLS) [18], Taylor series decomposition of hyperbola (Taylor) [3], and the two cone algorithms (Cone Eq. and Cone Ap.). We have decided to perform a test in order to check if a source was located inside or outside the simulated room. Sources localized outside the room (outliers) have not been taken into account for b_x and R_x computation.

This particular microphones setup is very effective with cone algorithm. In fact, we are searching for a cone whose vertex lies into an area surrounded by microphones. This allows us to sample the cone in points far from each other, and the fitting cone is found with a great precision.

In particular these data show that GS and LCLS methods are the worst ones between those tested, while both Cone Ap. and Cone Eq. algorithms can be compared to Taylor series decomposition of hyperbola equation, which performs

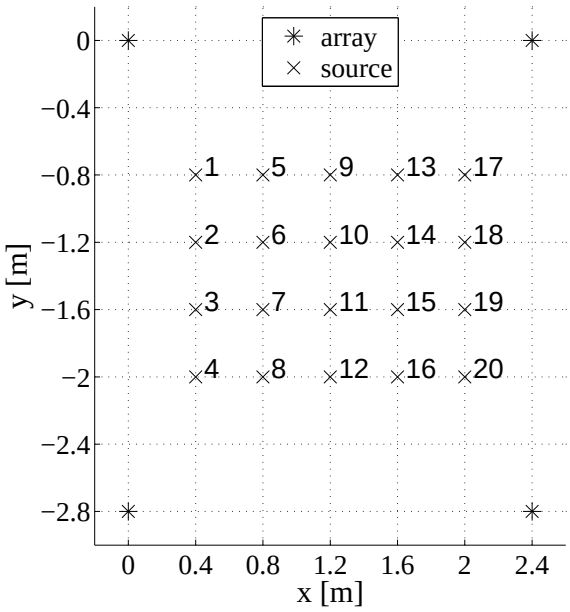


Figure 7.3: Experimental Setup: asterisks represent microphones, crosses represent the grid where sources were located. Each source position is associated to a number. This way we can refer to sources positions easily. All distances are in meters.

very well. In favor of Cone Eq. algorithm we have to say that it is the one with less outliers using the same TDOAs used by others algorithms. For this reason Cone Eq. algorithm is preferable over the others for its performance.

7.3.2 Experiment

In this paragraph we show some experimental results in order to confirm simulations results in the real world. The metric used is the same one used for simulations.

The experimental setup is described in Figure 7.3.

- the array is composed by 4 microphones placed on the vertices of a rectangle measuring $(2.4m \times 2.8m)$,
- the source is moved on 20 points on a square grid centered into the array.

The measurements for TDOAs computation were taken from the emission of 2 different signals into a dry room. First a $10s$ white noise signal, then a $10s$ recorded speech were tested. Each recorded signal was then windowed into parts

of $10ms$ each, in order to dispose of different frames. TDOAs were computed extracting the maxima of cross-correlations.

Figure 7.7 and 7.8 show the same comparison graphics as for simulations, respectively for gaussian noise and speech signals. Also experimental data were post-processed in order to discard outliers. In fact data about source localized outside the room were not taken into account.

Graphics from experimental data confirm the same results obtained from simulations on algorithms comparison and show that using gaussian noise or speech leads to the same results.

We also made a test on time of convergence of used algorithms. Figure 7.4 show the time used for each localization algorithm for estimating a source for each source position. This time is measured having already TDOAs measurements, so it takes account only of time of convergence of the algorithms. However it is also shown the cross-correlation time for TDOAs computation as a comparison. From this data we can notice that Gillette-Silverman method is faster than the others. However this result does not surprise us, because GS gives the solution in closed form, while other methods are iterative methods. Time spent for cross-correlation is really high compared to those taken by algorithms for their convergence. The only exception is represented from LCLS method, which has probably some problem in estimating source position with this particular setup.

Figure 7.5 shows better the comparison between time used by GS and time used by cone algorithms, as well as time used for cross-correlation for each source location. We notice that GS is about 6 times faster than Cone Aperture and 3 times faster than Cone Equation algorithms. However Cone Equation algorithm spent from 12 to 18 times less time than cross-correlation. For this reason we consider time of algorithms convergence negligible. This allows us to use a cone-based algorithm also if they are iterative methods.

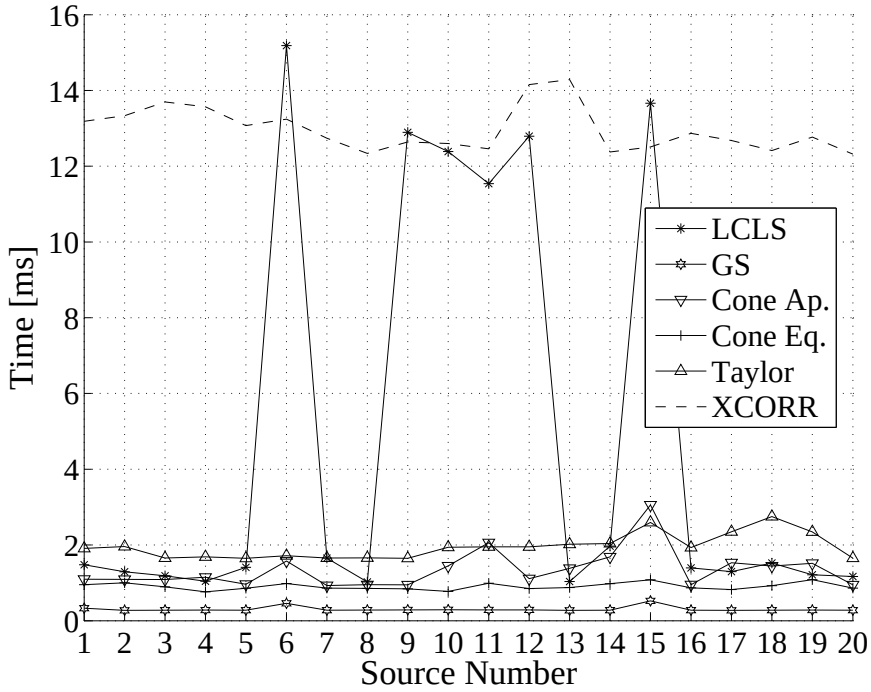
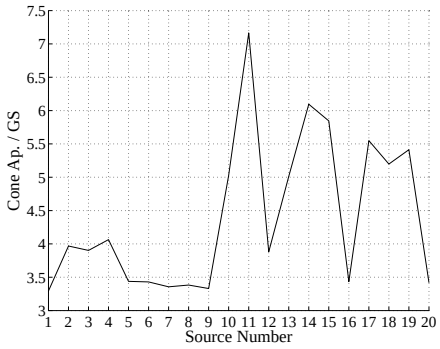
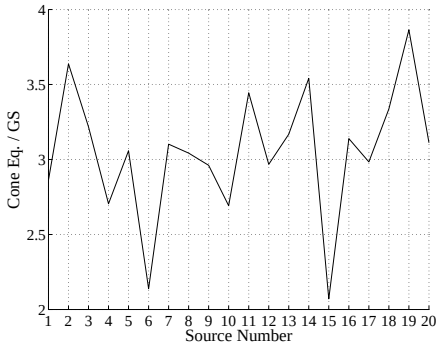


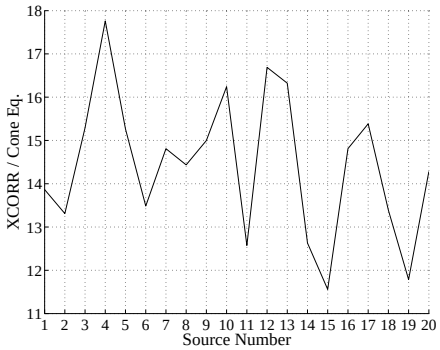
Figure 7.4: Time of Convergence: time taken for one source to be localized once in each position (from 1 to 20 according to sources numbers in Figure 7.3). Cross-correlation time (XCORR) is time spent for TDOAs computation. The tested algorithms are: Linear Correction Least Square (LCLS), Gillette-Silverman (GS), Cone Aperture (Cone Ap.), Cone Equation (Cone Eq.), and Taylor expansion of hyperbola (Taylor). We notice that LCLS has some convergence problem with the proposed setup, but all the other algorithms converge in less time than that spent for computing cross-correlations.



(a)

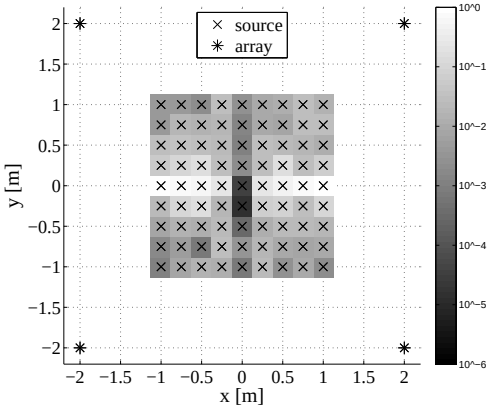


(b)

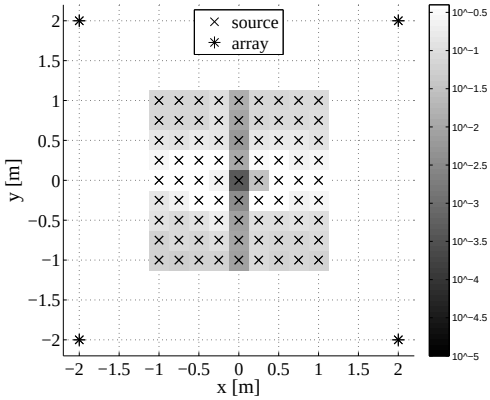


(c)

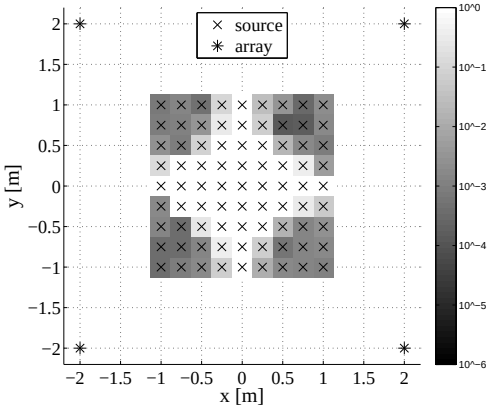
Figure 7.5: Time Comparison: (a) represents time used by Cone Aperture over time used by GS, (a) time used by Cone Equation over time spent by GS, and (c) time used for cross-correlation over time used by Cone Equation for each source position (according to sources numbers given in in Figure 7.3). We notice that both cone-based algorithms converge in a time comparable to GS one. From (c) we notice that time of convergence of Cone Eq. algorithm is negligible in contrast to cross-correlation time.



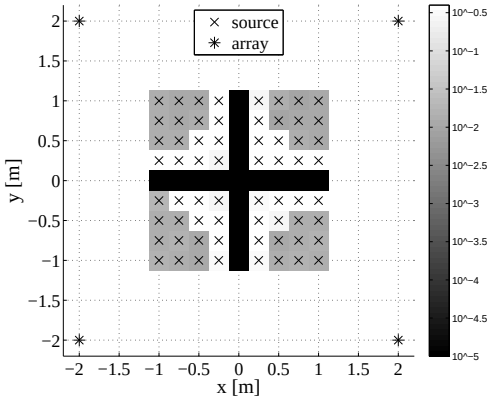
(a) b_x (GS)



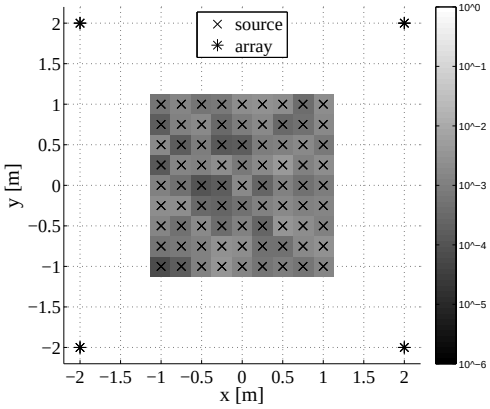
(b) R_x (GS)



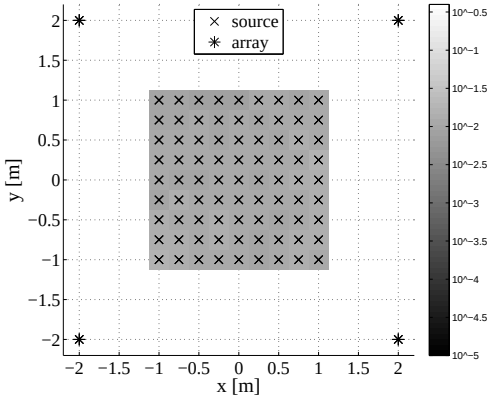
(c) b_x (LCLS)



(d) R_x (LCLS)

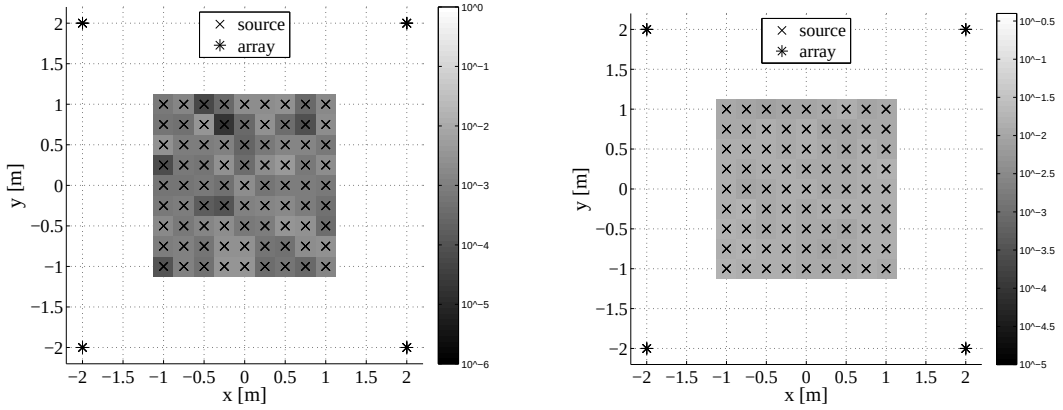


(e) b_x (Taylor)

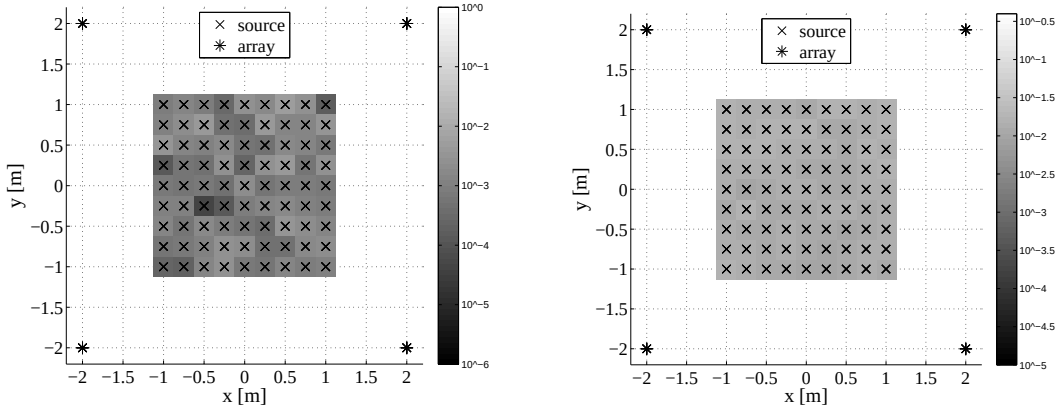


(f) R_x (Taylor)

Figure 7.6

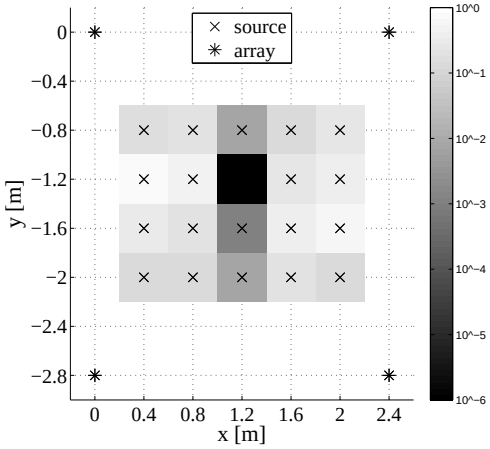


(g) b_x (Cone Ap.) (h) R_x (Cone Ap.)

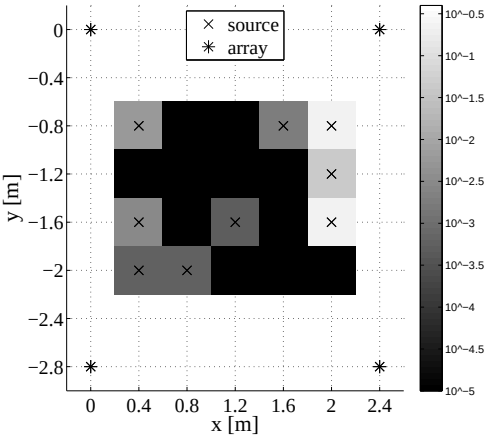


(i) b_x (Cone Eq.) (j) R_x (Cone Eq.)

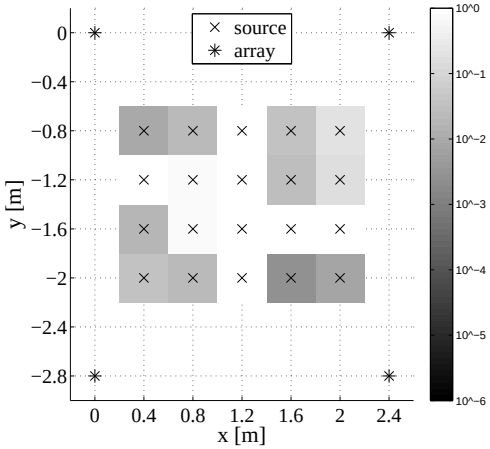
Figure 7.6: Simulations Results: b_x and R_x for different algorithms: (a) (b) GS, (c) (d) LCLS, (e) (f) Taylor, (g) (h) Cone Ap., (i) (j) Cone Eq. . Asterisks represent microphones, crosses represent real sources positions. All distances and errors are in meters. We notice that GS and LCLS performs worse than the other methods. The two cone algorithms and Taylor give almost the same results. However Cone Eq. is the algorithm presenting fewer outliers.



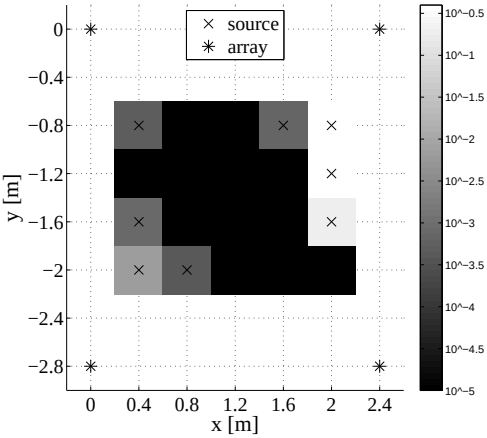
(a) b_x (GS)



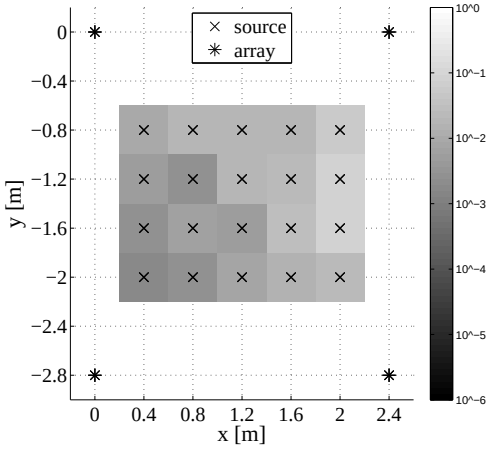
(b) R_x (GS)



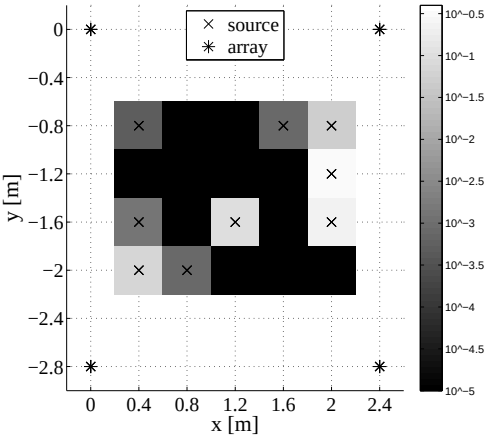
(c) b_x (LCLS)



(d) R_x (LCLS)

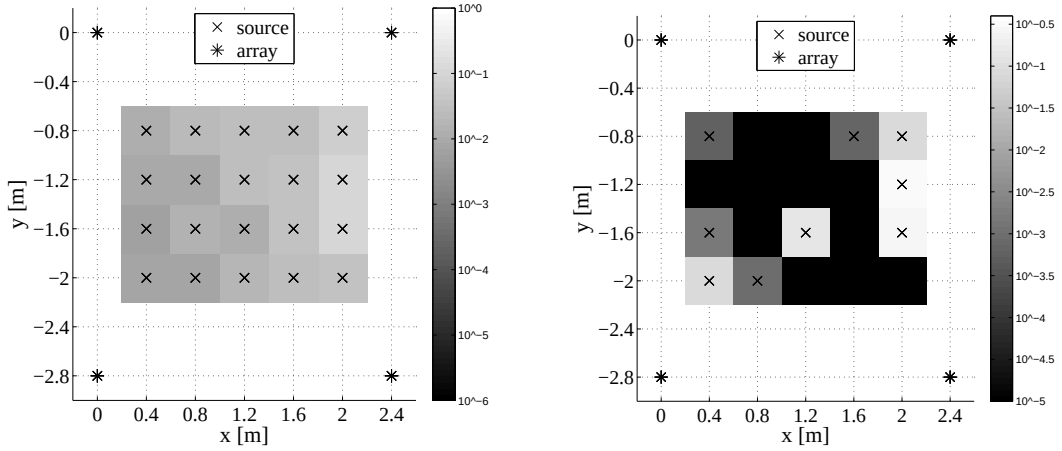


(e) b_x (Taylor)



(f) R_x (Taylor)

Figure 7.7



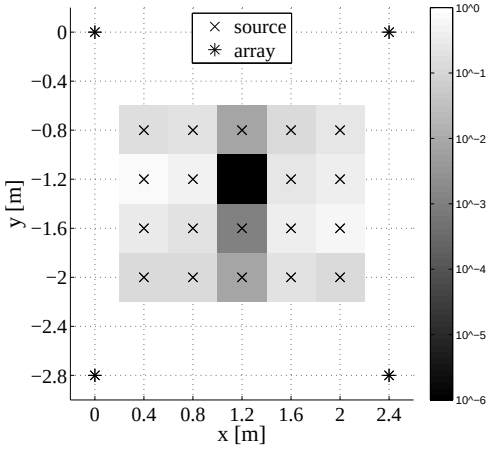
(g) b_x (Cone Aperture)

(h) R_x (Cone Aperture)

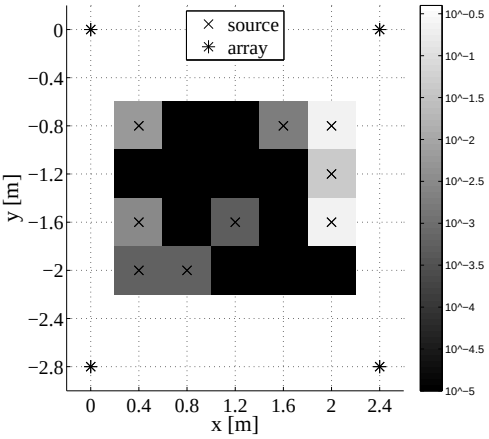
(i) b_x (Cone Equation)

(j) R_x (Cone Equation)

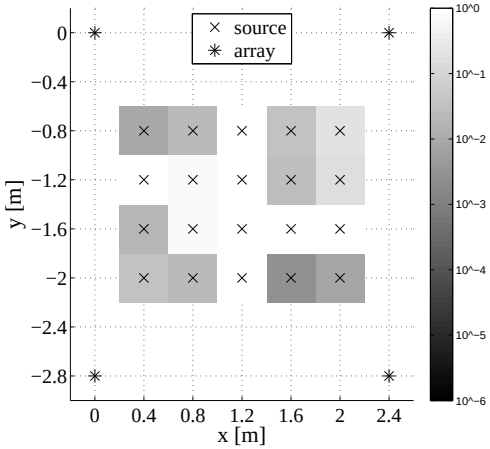
Figure 7.7: Experimental Results (gaussian noise): b_x and R_x for different algorithms when using gaussian signal: (a) (b) GS, (c) (d) LCLS, (e) (f) Taylor series, (g) (h) Cone Ap., (i) (j) Cone Eq.. Asterisks represent microphones, crosses represent real sources positions. All distances and errors are in meters. These results confirm the simulations ones. Taylor and the two cone-based algorithms give the best results.



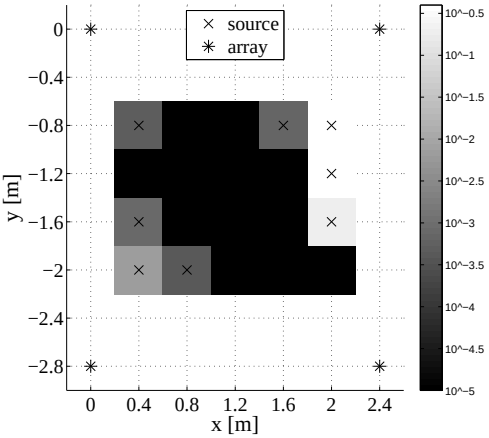
(a) b_x (GS)



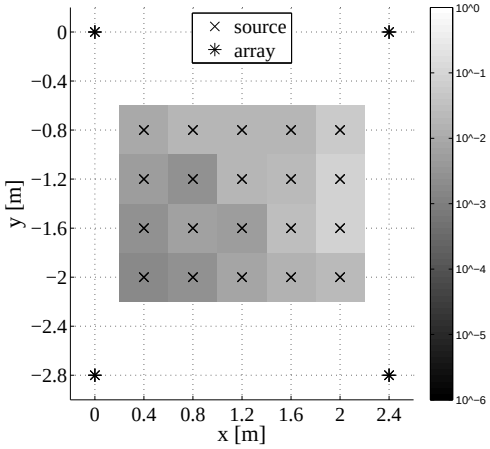
(b) R_x (GS)



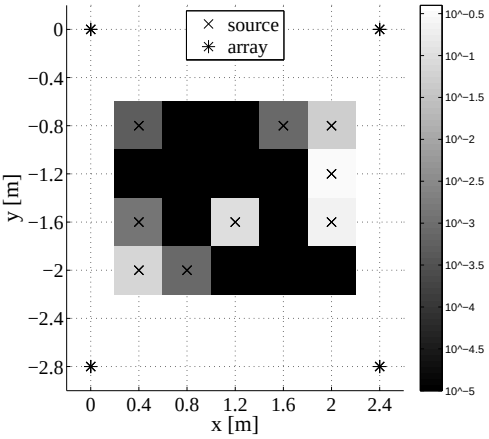
(c) b_x (LCLS)



(d) R_x (LCLS)

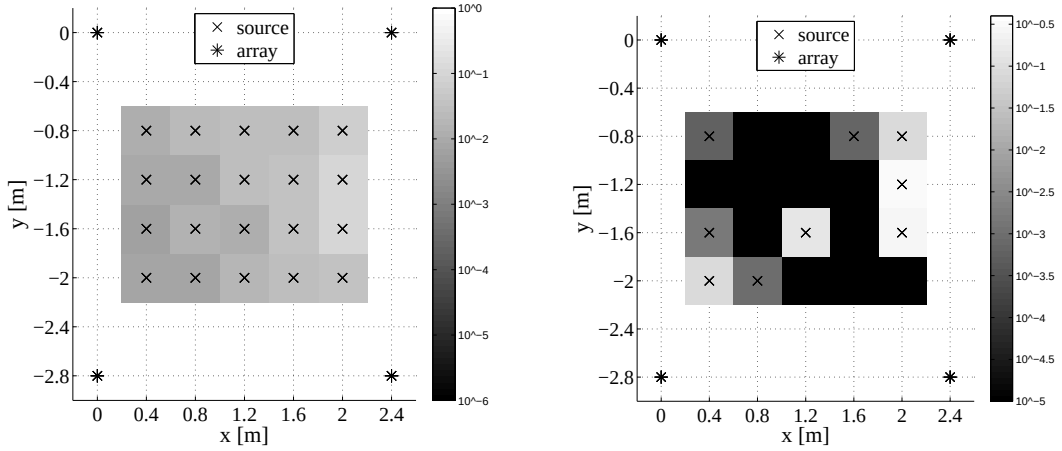


(e) b_x (Taylor)

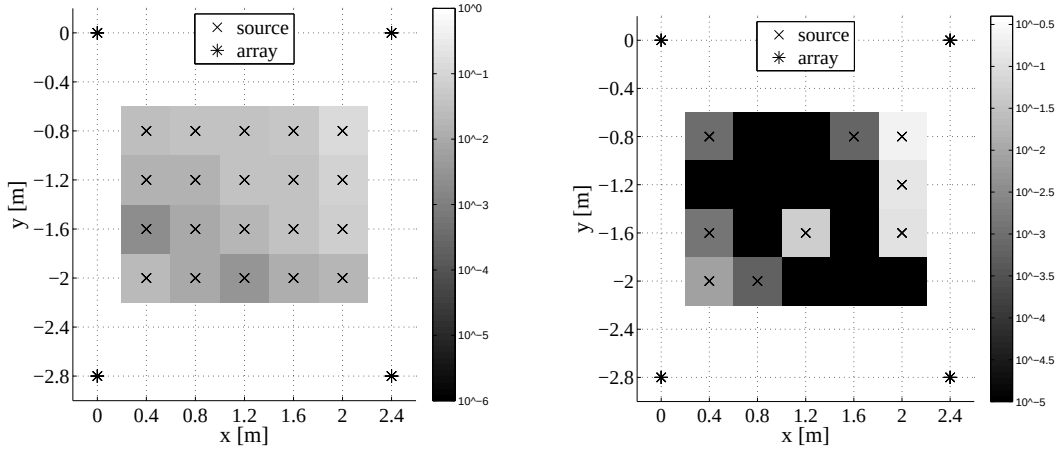


(f) R_x (Taylor)

Figure 7.8



(g) b_x (Cone Aperture) (h) R_x (Cone Aperture)



(i) b_x (Cone Equation) (j) R_x (Cone Equation)

Figure 7.8: Experimental Results (speech): b_x and R_x for different algorithms when using speech as signal: (a) (b) GS, (c) (d) LCLS, (e) (f) Taylor, (g) (h) Cone Ap., (i) (j) Cone Eq.. Asterisks represent microphones, crosses represent real sources positions. All distances and errors are in meters. We notice that using speech or gaussian noise gives the same results. So also in this case the best results are given by Cone Eq., Cone Ap., and Taylor.

7.4 Multi-Source Localization

This Section shows how the proposed algorithms perform in a two sources localization scenario. As for single source, we first show simulations and then experimental results.

7.4.1 Simulations

We tested the brute force algorithm for both Cone Aperture and Cone Equation methods with the setup depicted in Figure 7.10:

- the array is made up of 4 microphones placed on the vertices of a rectangle measuring $(2.4m \times 2.8m)$,
- sources are placed in couples of points inside the array area.

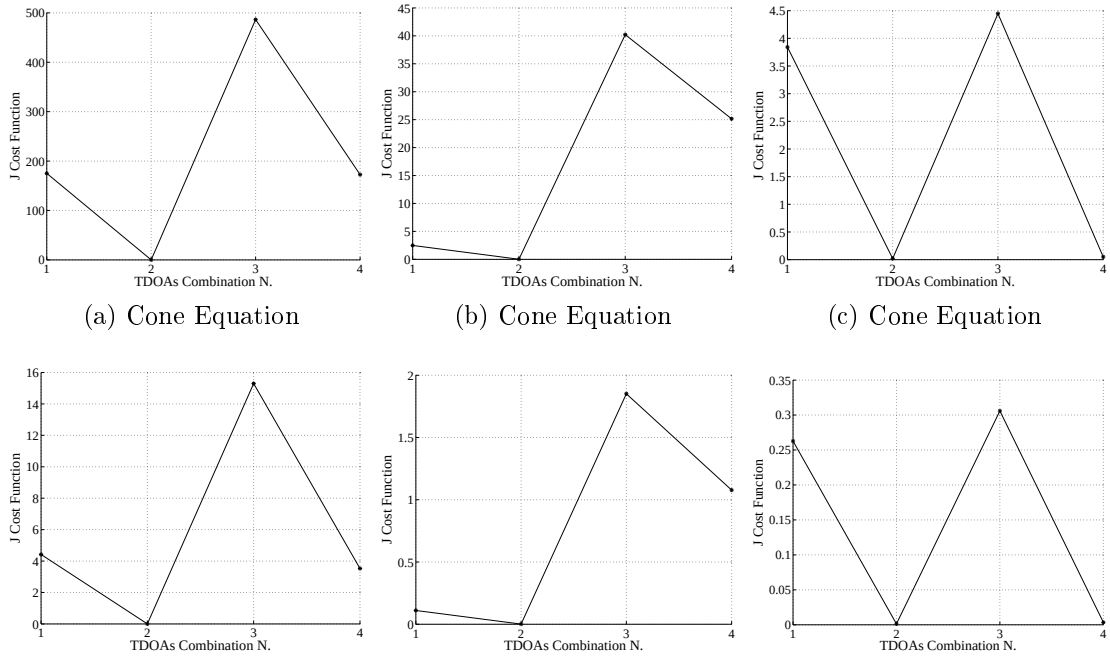
Next Figures show results for three different pairs of sources displaced in different positions:

1. $[x_{s1}, y_{s1}] = [0.4, -0.8]$ and $[x_{s2}, y_{s2}] = [2, -2]$,
2. $[x_{s1}, y_{s1}] = [0.4, -0.8]$ and $[x_{s2}, y_{s2}] = [0.4, -1.6]$,
3. $[x_{s1}, y_{s1}] = [0.8, -1.2]$ and $[x_{s2}, y_{s2}] = [1.2, -1.2]$.

Results are obtained from 50 simulations with TDOAs obtained from ranges differences corrupted with zero mean, 2 cm standard deviation gaussian noise.

Figure 7.9 shows how J cost function behaves when TDOAs combinations are tested with Cone Equation and Cone Aperture algorithms. We can notice that the minimum is clearly found for TDOAs combination number two (which is actually the correct combination) except for case 3. In this case a symmetry of sources and microphones makes two combinations almost equal, so there are two close solutions.

Figure 7.10 shows clusters of sources location estimated with the brute force approach on 50 simulations. For each TDOAs combination there are two clusters of points representing the two sources locations on all the simulations. The dots nearest to real sources locations are always those found with the correct TDOAs combination and they also are the points which minimize the cost function J. In case 3, many clusters are near to sources positions and become source candidates

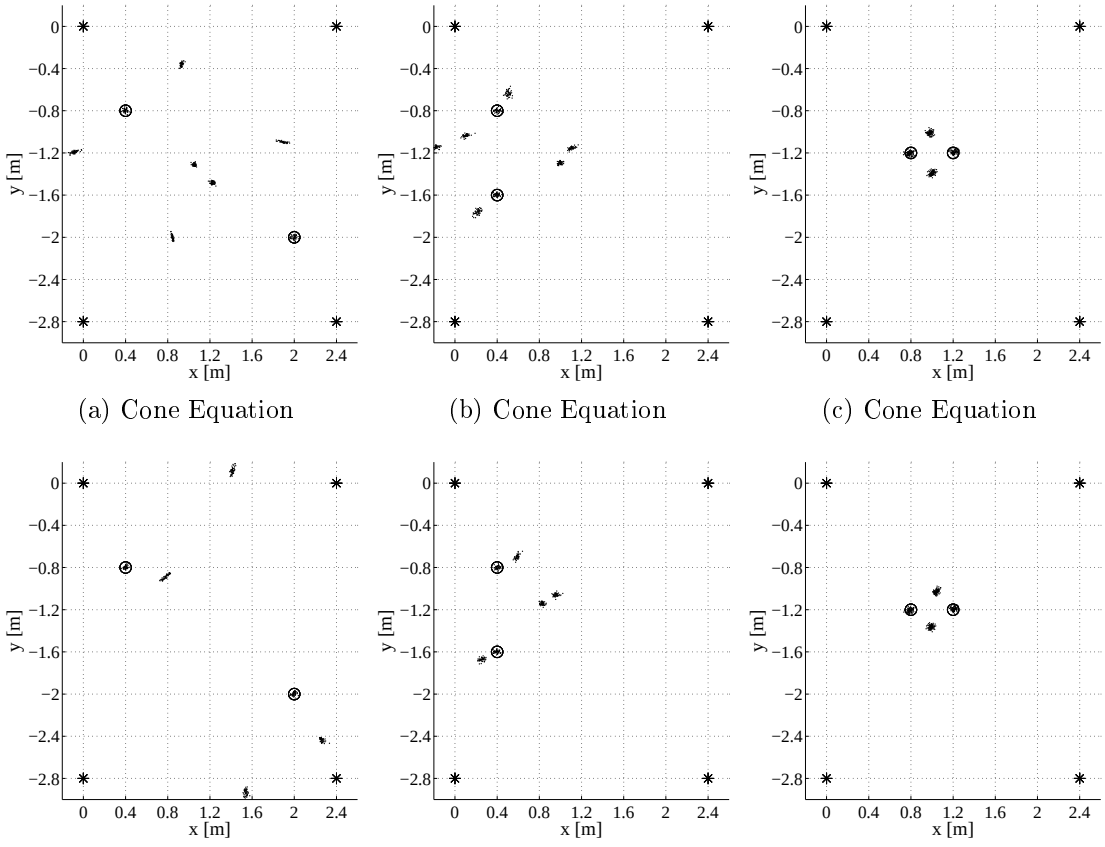


(a) Cone Equation (b) Cone Equation (c) Cone Equation

(d) Cone Aperture (e) Cone Aperture (f) Cone Aperture

Figure 7.9: Multi-Source Cost Function: Cone Equation J cost function for sources location number 1 (a), 2 (b), and 3 (c). Cone Aperture J cost function for sources location number 1 (d), 2 (e), and 3 (f). Four TDOAs combination are tested as shown in Chapter 5. TDOAs combination leading to the minimum of the cost function is the correct combination.

because of the symmetry above shown. These simulations confirm that with brute force methods, both sources can be well localized. If the cost function has several global minima, we expect that more TDOAs combinations lead to similar solutions, so it does not matter which one is taken as the correct combination.



(a) Cone Equation (b) Cone Equation (c) Cone Equation

(d) Cone Aperture (e) Cone Aperture (f) Cone Aperture

Figure 7.10: Multi-Source Simulation Setup: circles are actual sources positions, asterisks are microphones. Dots grouped in clusters are sources located with different TDOAs combinations with Cone Equation algorithm for source location number 1 (a), 2 (b), and 3 (a), and with Cone Aperture algorithm for source location number 1 (d), 2 (e), and 3 (f). Clusters localized outside the array area are not shown.

7.4.2 Experiment

In the laboratory we tested the DATEMM based method for TDOAs disambiguation with Cone Equation, Cone Aperture, and Taylor series decomposition of hyperbola localization algorithms. Figure 7.11 shows the setup:

- 4 microphones are located on the vertices of a square measuring $(1m \times 1m)$,
- 32 sources are placed on an octagonal array as shown in Figure 7.11.

We chose to use the octagonal array in order to have a well-measured ground-truth. In this way we had all sources already fixed in their position, and we chose to switch on and off the needed sources. This prevent us from making errors in source displacement.

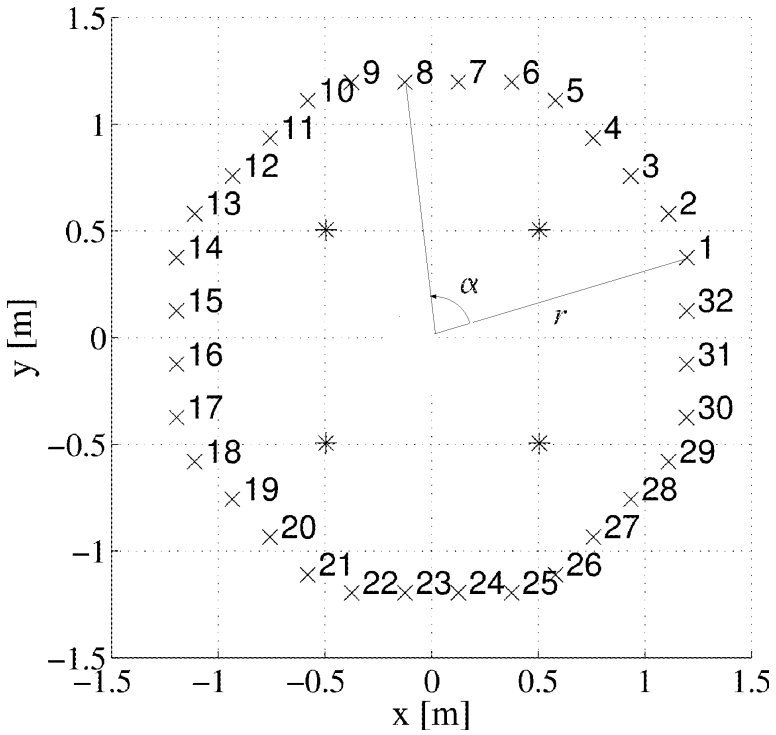


Figure 7.11: Multi-Source Experimental Setup: crosses represent sources, asterisks represent microphones. Angle α are measured anticlockwise starting from source 1. Each source location is associated to a number.

As the setup is closer to a circular configuration, we use the polar coordinates α and r. Angle α is the azimuth angle defined as zero for source number 1, and increasing anticlockwise as shown in Figure 7.11. The range r is the distance

from the source to the center of the array. Notice that the range is not the same for each source because they are not placed on a circle but disposed with an octagonal shape.

The metric used for accuracy measuring is similar to the one presented for single source localization, but works on the polar coordinates:

- *bias on* α:

$$b_\alpha = \left| \frac{1}{n} \sum_{i=1}^{n} (\alpha_s - \hat{\alpha}_{s,i}) \right|, \tag{7.6}$$

where n is the number of noisy TDOAs tested for a source location, α_s is the α angle of the real source, and $\hat{\alpha}_{s,i}$ is the estimated one based on the i-th TDOAs realization. This is the measure of the absolute value of the mean difference between the α angle of the real source and the estimated one.

- *bias on* r:

$$b_r = \left| \frac{1}{n} \sum_{i=1}^{n} (r_s - \hat{r}_{s,i}) \right|, \tag{7.7}$$

where n is the number of noisy TDOAs tested for a source location, r_s is the range r of the real source, and $\hat{r}_{s,i}$ is the estimated one based on the i-th TDOAs realization. This is the measure of the absolute value of the mean difference between the range r of the real source and the estimated one.

- *RMSE on* α:

$$R_\alpha = \left(\frac{1}{n-1} \sum_{i=1}^{n} \left((\varepsilon_{\alpha,i}) - (\overline{\varepsilon_\alpha}) \right)^2 \right)^{\frac{1}{2}}, \tag{7.8}$$

where n is the number of noisy TDOAs tested for a source location, $\varepsilon_{\alpha,i}$ is the difference between the source α angle α_s and the estimated one with the i-th TDOAs realization $\hat{\alpha}_{s,i}$, and $\overline{\varepsilon_\alpha}$ is the mean error computed as

$$\overline{\varepsilon_\alpha} = \frac{1}{n} \sum_{i=1}^{n} (\alpha_s - \hat{\alpha}_{s,i}). \tag{7.9}$$

- *RMSE on r*:

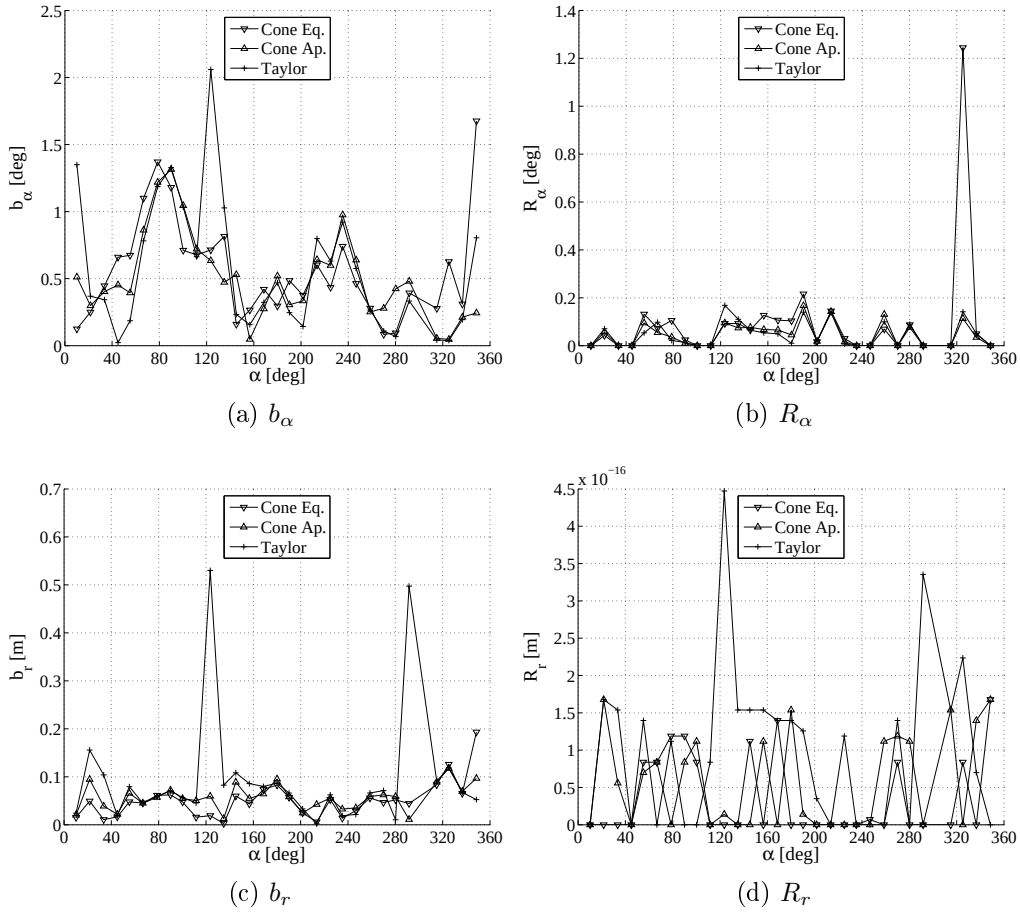
$$R_r = \left(\frac{1}{n-1} \sum_{i=1}^{n} \left((\varepsilon_{r,i}) - (\overline{\varepsilon_r}) \right)^2 \right)^{\frac{1}{2}}, \tag{7.10}$$

where n is the number of noisy TDOAs tested for a source location, $\varepsilon_{r,i}$ is the difference between the source range r_s and the estimated one with the i-th TDOAs realization $\hat{r}_{s,i}$, and $\overline{\varepsilon_r}$ is the mean error computed as

$$\overline{\varepsilon_r} = \frac{1}{n} \sum_{i=1}^{n} (r_s - \hat{r}_{s,i}). \tag{7.11}$$

For the experiment, source number 1 was used in couple with every other source of the array. The two sources emitted two different white gaussian noises for $10s$. Every received signal has been windowed into frame of $0.1s$ each. With 70 of these frames, TDOA disambiguation was then performed. After disambiguation, localization algorithms were applied to pairs of sources.

Figure 7.12 shows results according to explained metrics. No results for source 1 are displayed. Outliers are discarded from shown results. All the above tested algorithms perform well in this situation. Also in this situation Taylor, Cone Ap., and Cone Eq. give results really close. Furthermore also TDOA disambiguation works correctly. Outliers due to TDOA disambiguation are more likely to occur when sources are closer to each other.



(a) b_α (b) R_α

(c) b_r (d) R_r

Figure 7.12: Multi-Source Results: b_α (a), R_α (b), b_r (c), and R_r (d). The two Cone Algorithms are compared to Taylor series one which is the best performing one between those tested from literature.

7.5 Reflector Localization

This Section shows a comparison between results obtained with the cone proposed algorithms and Taylor series one for reflector localization.

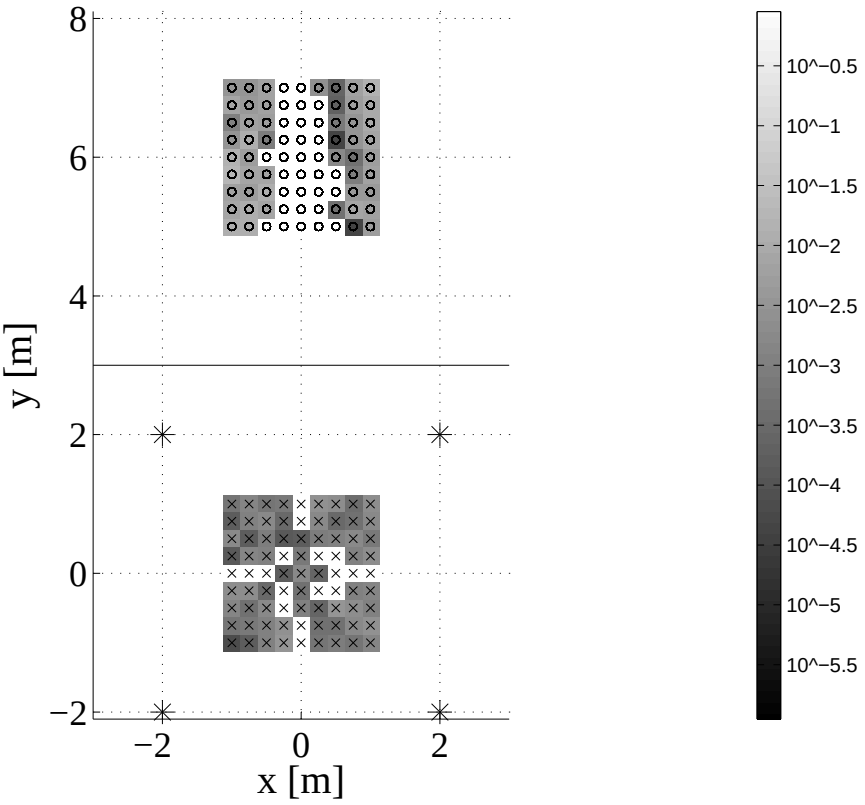
The comparison is made on simulation data with the hypothesis of having already solved the TDOAs ambiguity. The simulation setup was similar to that used for source localization simulations, considering that a reflector is present. The reflector gives birth to an image source for each real source tested. For each source location, 50 sets of corrupted TDOAs were tested for real and image source. TDOAs were computed from range differences corrupted with zero mean and 2 *cm* standard deviation gaussian noise.

The accuracy of inference algorithms is measured with the metrics shown at the beginning of this Chapter.

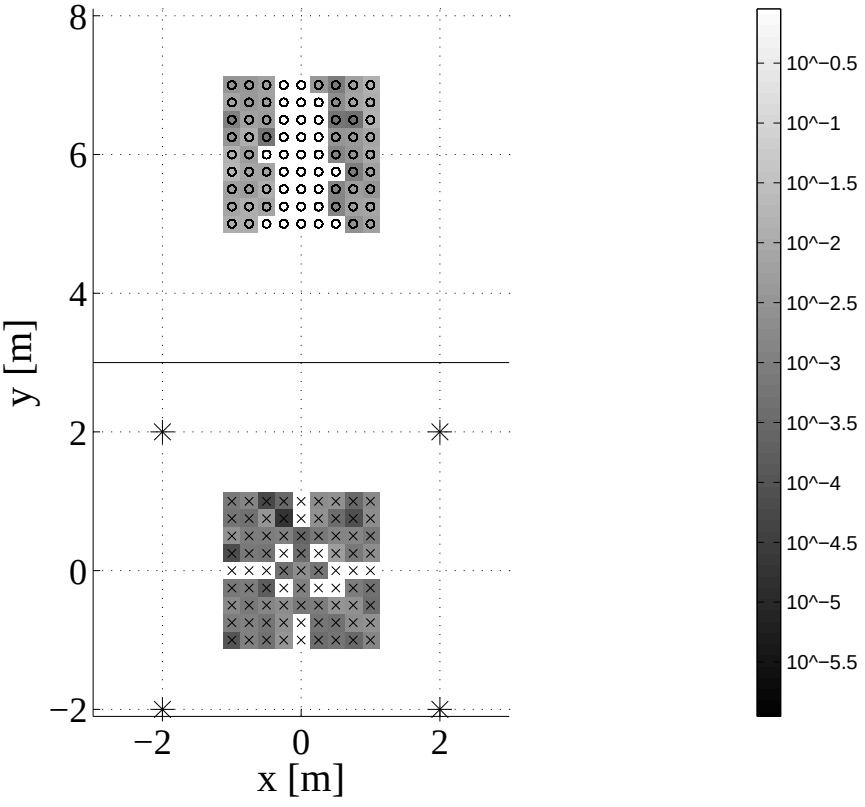
The comparison is made in particular between the cones algorithms and the Taylor series one. The Taylor series algorithm was chosen between those tested in source localization because it is the one which performs best in the literature.

In Figure 7.13 bias for the three algorithms is shown. We notice that when estimating image sources Cone Equation algorithm performs much better than the others. Cone Ap. and Taylor fail in localizing many image sources, and this makes impossible the reflector localization. This is due to the fact that we did not impose a room dimension. This way we could not recognize outliers, so they have not been discarded. From these results it is evident that Cone Eq. is more robust than the other algorithms.

This is why we show reflector localization results only for Cone Eq. in Figure 7.14. From these results we notice that with Cone Eq. the angle error is almost always under one degree, while the distance error can be also of 30 *cm*.

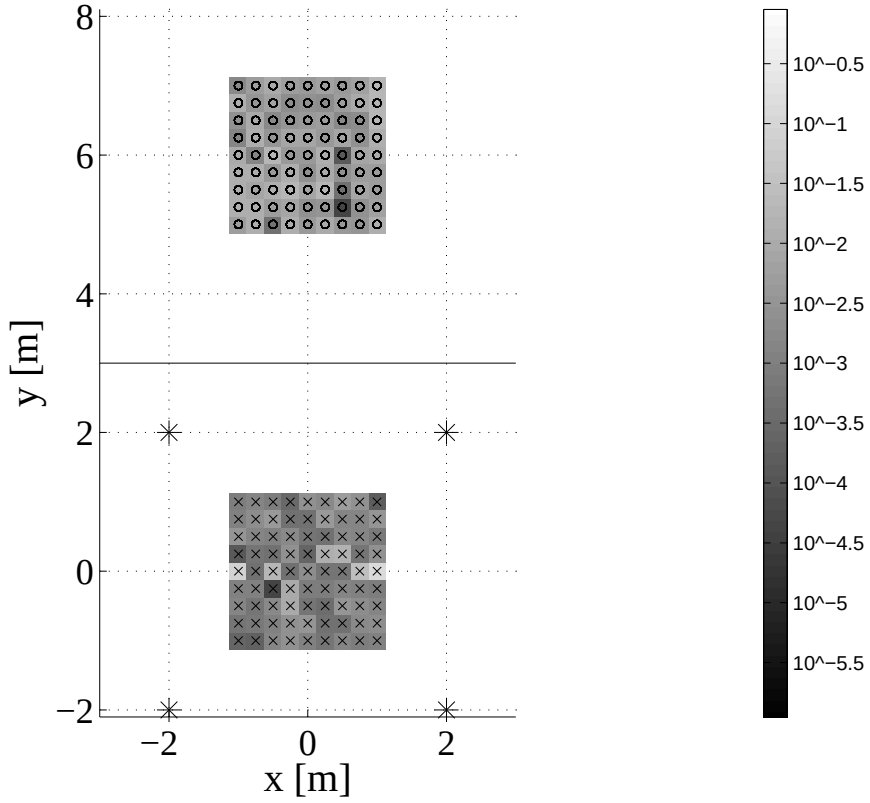


(a) b_x (Taylor)



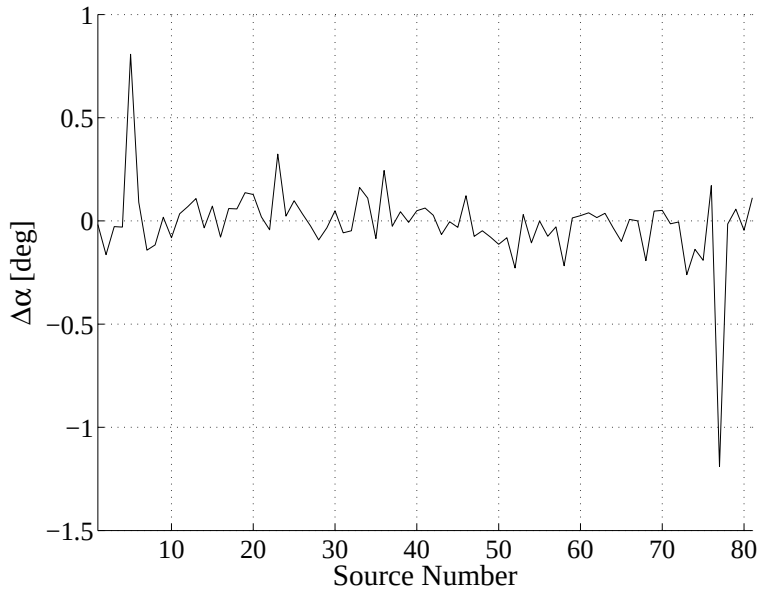
(b) b_x (Cone Ap.)

Figure 7.13

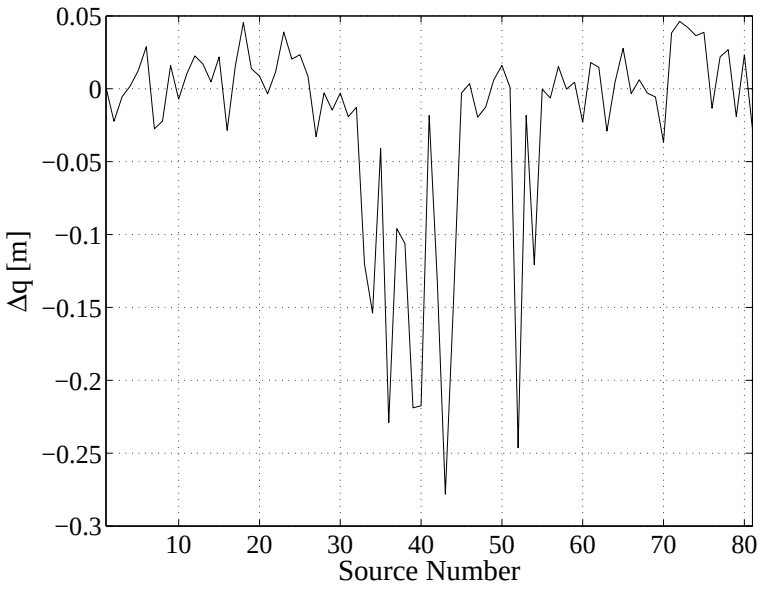


(c) b_x (Cone Eq.)

Figure 7.13: Reflector Bias; b_x for real and image sources with Taylor series (a), Cone Aperture (b) and Cone Equation (c) algorithms. Crosses are real sources, asterisks are microphones, the black line represents the reflector, and circles are image sources. All measurements are in meters. We notice that only Cone Eq. enable us to correctly localize image sources. The other two algorithms make many error in doing this. This is due to the fact that we did not impose a room dimension. This way we could not recognize outliers, so they have not been discarded. The Cone Eq. robustness is so evident.



(a) Angle Error



(b) Distance Error

Figure 7.14: Reflector Localization: $\Delta\alpha$ (a) and (b) Δq for Cone Equation algorithm. Sources numbers are the same used in Figure 7.2. The angle is well estimated, while the parameter q shows more errors. However with Cone Eq. it is possible to estimate a reflector position, while with the others algorithms it is not possible.

Chapter 8

Conclusions and Future Works

In this work we have analyzed the problem of acoustic source localization, especially with TDOAs measurements.

We have also provided details about reflector localization algorithms. In this area we have shown that it is possible to adapt our and others localization methods thanks to DATEMM algorithm, which can be modified for a less general case of study.

For the considered research themes we have first made a summary of existing algorithms in order to help us finding solutions to some problems (in particular those about TDOA disambiguation) and to perform a comparison with our results. Then we have explained what is the idea at the basis of cone algorithms and we have shown how they work. We have also made a study about possible array geometries that can be used with these algorithms so as to obtain the best results.

After the theoretical study, we have implemented our and other algorithms in order to give simulations and experimental results. In general cone-based methods here proposed perform, at least, as well as the best between others localization methods, but making use of conics we can write our constraints with a compact notation. In particular when using the proposed microphones configuration, Cone Equation algorithm is to be considered more robust than the others, notably when sources are outside the array area. This is most clear when we look at simulation results on reflector localization. In fact reflector localization can not be performed with the other algorithms, which make too many errors. Again, Cone Equation algorithm is likely to be more robust to noisy TDOAs, presenting fewer outliers.

This is why this algorithm is probably a better choice when we are not allowed to discard measurements, and we need to estimate source location with every frame of our recorded signal. For instance, in source tracking, it is more suitable to use an algorithm which locates the source with some error instead of an algorithm performing in a more precise way but presenting outliers with some signal frames. In fact, if we do not have information about source trajectory, we can not realize when an estimate is an outlier.

As far as future work is concerned, a research direction is the extension of single and multi-source localization methods to a reverberant room case. In this scenario we will need to study how reverberations affect measurements, in a similar way as we have done for reflector localization. In this configuration it could also be interesting to jointly sort out the problem of multi-source localization and reflector inference. Also the multi-source and multi-reflector case can be studied, obtaining algorithms suitable for more general cases.

The final scope of this research could be the study of how these algorithms evolve into 3D localization world. We could find a 4D constraints space for cone cost functions in which cones evolve into other geometric objects.

If this goal proves successful, the algorithms could serve as a solution to localize all the reflectors and sources in an actual room, enabling us to be in position to cope with the most general case.

Bibliography

[1] Y. Huang and J. Benesty, *Audio Signal Processing for Next Generation Multimedia Communication Systems*. Kluwer Academic Publishers, 2004.

[2] M. Gillette and H. Silverman, "A linear closed-form algorithm for source localization from time-differences of arrival," *Signal Processing Letters, IEEE*, vol. 15, pp. 1 –4, 2008.

[3] W. Foy, "Position-location solutions by taylor-series estimation," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. AES-12, no. 2, pp. 187 –194, 1976.

[4] Y. Chan and K. Ho, "A simple and efficient estimator for hyperbolic location," *Signal Processing, IEEE Transactions on*, vol. 42, pp. 1905 –1915, Aug. 1994.

[5] Politecnico di Milano, University of Erlangen-Nuremberg, Fondazione Bruno Kessler, and Imperial College, "Deliverable D2.1, first report on analysis methodologies (SCENIC project: Self-configuring environment-aware intelligent acoustic sensing)." Last Update: December 16,2009.

[6] F. Antonacci, A. Sarti, and S. Tubaro, "Geometric reconstruction of the environment from its response to multiple acoustic emissions," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pp. 2822 –2825, 2010.

[7] C. Militello and S. R. Buenafuente, "An exact noniterative linear method for locating sources based on measuring receiver arrival times," *The Journal of the Acoustical Society of America*, vol. 121, no. 6, pp. 3595–3601, 2007.

[8] P. Teng, A. Lombard, and W. Kellermann, "Disambiguation in multidimensional tracking of multiple acoustic sources using a gaussian likelihood criterion," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pp. 145 –148, 2010.

[9] H. Buchner, R. Aichner, J. Stenglein, H. Teutsch, and W. Kellennann, "Simultaneous localization of multiple sound sources using blind adaptive MIMO filtering," in *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, vol. 3, pp. iii/97 – iii100 Vol. 3, 2005.

[10] Y. Lin, J. Chen, Y. Kim, and D. D. Lee, "Blind sparse-nonnegative (BSN) channel identification for acoustic time-difference-of-arrival estimation," in *Applications of Signal Processing to Audio and Acoustics, 2007 IEEE Workshop on*, pp. 106 –109, 2007.

[11] E. Di Claudio, R. Parisi, and G. Orlandi, "Multi-source localization in reverberant environments by ROOT-MUSIC and clustering," in *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on*, vol. 2, pp. II921 –II924 vol.2, 2000.

[12] C. Zannini, A. Cirillo, R. Parisi, and A. Uncini, "Improved TDOA disambiguation techniques for sound source localization in reverberant environments," in *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, pp. 2666 –2669, 302010-june2 2010.

[13] J. Scheuing and B. Yang, "Disambiguation of TDOA estimates in multipath multi-source environments (DATEMM)," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 4, pp. IV –IV, May 2006.

[14] J. Scheuing and B. Yang, "Disambiguation of TDOA estimation for multiple sources in reverberant environments," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 16, no. 8, pp. 1479 –1489, 2008.

[15] S. Tervo and T. Korhonen, "Estimation of reflective surfaces from continuous signals," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pp. 153 –156, 2010.

[16] D. Ba, F. Ribeiro, C. Zhang, and D. Florêncio, "L1 regularized room modeling with compact microphone arrays," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pp. 157 –160, 2010.

[17] Y. Zheng, H. Wang, L. Wan, and X. Zhong, "A placement strategy for accurate TOA localization algorithm," in *Communication Networks and Services Research Conference, 2009. CNSR '09. Seventh Annual*, pp. 166 –170, May 2009.

[18] Y. Huang, J. Benesty, G. Elko, and R. Mersereati, "Real-time passive source localization: a practical linear-correction least-squares approach," *Speech and Audio Processing, IEEE Transactions on*, vol. 9, pp. 943 –956, Nov. 2001.

[19] P. Svaizer, M. Matassoni, and M. Omologo, "Acoustic source location in a three-dimensional space using crosspower spectrum phase," in *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on*, vol. 1, pp. 231 –234 vol.1, Apr. 1997.

[20] L. Tong, G. Xu, and T. Kailath, "Blind identification and equalization based on second-order statistics: a time domain approach," *Information Theory, IEEE Transactions on*, vol. 40, pp. 340 –349, Mar. 1994.

[21] J.-Y. Do, M. Rabinowitz, and P. Enge, "Robustness of TOA and TDOA positioning under suboptimal weighting conditions," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 43, no. 3, pp. 1177 –1180, 2007.

[22] B. Fang, "Simple solutions for hyperbolic and related position fixes," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 26, pp. 748 –753, Sept. 1990.

[23] F. Fletcher, B. Ristic, and D. Musicki, "Recursive estimation of emitter location using TDOA measurements from two UAVs," in *Information Fusion, 2007 10th International Conference on*, pp. 1 –8, 2007.

[24] R. I. Reza, R. I. Reza, C. chair Dr, and A. E. Bell, "Data fusion for improved TOA/TDOA position determination in wireless systems," in *Master4.pdf Song, H.L*, pp. 902–908, 2000.

[25] D. Torrieri, "Statistical theory of passive location systems," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. AES-20, no. 2, pp. 183 –198, 1984.

[26] Y. Yu and H. Silverman, "An improved TDOA-based location estimation algorithm for large aperture microphone arrays," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on*, vol. 4, pp. iv–77 – iv–80 vol.4, May 2004.