A FRAMEWORK FOR THE POSITION CALIBRATION OF ACOUSTIC DEVICES

Braian Daniel Ruiz García, Manuel Iglesias del Campo

Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB), Politecnico di Milano
Piazza Leonardo Da Vinci 32, 20122 Milano, Italy
[braiandaniel.ruiz,manuel.iglesiasdelcampo]@mail.polimi.it

ABSTRACT

This study is the development of a method for acoustic devices position calibration and delay estimation, based on information extracted from the room impulse responses measurements between sources and microphones, in particular the time-of-arrival (TOA). An optimization method for the calibration has been developed and tested in two different scenarios, one with simulated data and one with real data. A graphical user interface (GUI) has been developed as well so it can be used as an application. We show first how with the simulated data the calibration method implemented provides satisfactory results according to the ground-truth locations. Then we show how with real data the calibration method gives good results in two out of three coordinates obtaining different results if the delay estimation is done or not. We conclude the report with a discussion on the results obtained and with possible improvements of the calibration method implemented.

1. INTRODUCTION

When dealing with acoustic measurements, such as source localization, source separation or many other different microphone array techniques, the exact location of the devices is required in order to have a robust algorithm.

In these microphone array techniques and acoustic measurements, there is a high sensitivity to the position of the devices, so accuracy is needed. Moreover, as stated in [1], nowadays arrays can be formed by hundreds or thousands of elements, and they can be of non regular geometry form, which implies that a better device localization algorithm is required. Manual measurement of positions are not good enough when the array of microphones are too big and even if they are not big the human error is always present. Therefore, calibration methods for device localization are needed in these cases. These calibration methods use the acoustic information included in the signal received by the sensors in order to estimate the positions. In particular, and for this research, the information is obtained from the room impulse responses, more in particular from the times-of-arrival (TOA). The next chapter of this report includes a wider explanation of the room impulse responses and the acoustic information that it is possible to find in it.

Normally, in these acoustic measurements and techniques there is a signal processing chain in which different devices are involved. Each of these devices can introduce a delay in the process. Therefore, the acoustic information of the signal, the one that is required to estimate the position of the devices, is affected by these delays. In order to overcome that issue, the calibration algorithm used to estimate the devices location needs to take into account this delay included by the devices and it should be able to estimate it.

This work proposes the development of an acoustic measurement calibration framework, in which the unknown position of the devices is estimated by an optimization problem using the acoustic information given by the room impulse response, taken and not taken into account the delay. A first section is dedicated to give the reader an overview of the theory behind the room impulse response and the information included in it and used for this work. After that the development of the algorithm with its mathematical basis is described followed by the description of the implementation of the algorithm itself. Finally, two sections are dedicated to show the results obtained during this work and conclusion including thoughts regarding these results and possible improvements.

2. ROOM IMPULSE RESPONSE THEORY

As stated in [2], Room Impulse Responses ($RIRs$) play a fundamental role in room acoustics, providing useful parameters that describe sound fields qualitatively. A RIR between two points in a room is typically measured by generating a deterministic signal, e.g., a sine sweep, by using a loudspeaker at one point and recording the sound pressure with a microphone at the other point. Room impulse responses are generally considered to consist of three main parts; the direct sound, a set of sparse early reflections and a stochastic tail. As time progresses the early reflection density grows until individual reflections are no longer distinguishable.[3]

A constraint regarding how to obtain the RIR of our desired devices from a specific room is to repeat the process for every loudspeaker and microphone within the room. Therefore, over the years several methods have been created in order to obtain reliable measurements regarding the RIR from devices, especially when the array of microphones and loudspeakers comes in a bigger volume. It is worth mentioning that these methods won't be introduced in this study since our framework exploits an already developed RIR database.

Within the RIR we introduce the TOA measurement (Time of Arrival) which is the time instant for the microphone to receive the signal emitted by the loudspeaker. TOA always introduces an error when it comes to measure between the sensor and the source, for example arising from the uncertainties in the sound velocity of the medium, when the source is not sufficiently wideband (resulting in some uncertainties in the exact time of the first signal arrival), when microphones are close to obstacles like walls, or even when ambient noise corrupts the measurements.[1] In the following sections we will describe the method used to obtain an adequate TOA measurement out of the RIR databases.

3. POSITION CALIBRATION METHOD

In this section it is described in detail the calibration method used in this work. This section is divided in two subsections, one without an estimation of the delay introduced in the processing chain, and the second one estimating that delay.

3.1. Position calibration method without delay estimation

The scenario is composed of a number of devices with known positions, a number of devices with unknown positions and a set of $RIR's$ related to these devices. Since this method presents a symmetry in the roles of sources and microphones, from now on, it is distinguished between devices with known positions and devices with unknown position, no matter if they are microphones or speakers, but always taking into account that all the devices with known positions are of the same kind (microphones or sources), the same applies to the devices with unknown positions. It is also needed to be said that it does not matter if the position vector is in 3-dimensions (x, y, z) or in 2-dimensions (x, y), since the method works the same for both cases.

The first thing that it is needed to be done is to obtain the TOA, in order to obtain the distance between devices. This information can be found in the Room Impulse Responses as explained in the previous section. The time-of-arrival is obtained by calculating the time of the direct path of the room impulse response, that is, in most of the cases the first noticeable peak. When working with room impulse responses there are samples and amplitude corresponding to each sample. Therefore, depending on the sample frequency that it is being used the resolution of the RIR will be different. This leads to possible errors finding the peak, in particular the error could be of:

$$t_{res} = \frac{1}{f_s} \tag{1}$$

Being t_{res} the time resolution of the RIR and f_s the sample frequency. Although it is a small error, a quadratic interpolation is applied to the RIR to avoid it. Another aspect to be taken into account is that the direct path could be a first negative peak, i.e, a minimum in the RIR, so the absolute value of it is taken in order to avoid misleading finding this peak. At this point, finding the direct path is as easy as finding the first peak of the RIR, and so the time-of-arrival and the correspondent distance between two devices are calculated as:

$$TOA = \frac{m}{f_s} \tag{2}$$

$$d = TOA * c \tag{3}$$

where m is the sample where the first peak is, f_s the sample frequency and c the sound velocity (343 m/s).

Once the TOA and the correspondent distance is obtained, a geometrical situation like the one shown in Figure 1 would be obtained. As it can be seen knowing the distance between two devices, a circle in a 2-dimension case or a sphere in a 3-dimension case, can be drawn with center in the position of the known device and with radius equal to the distance. The position of the unknown device could be in any of the position correspondent to the edge of the circle(sphere) drawn. Therefore, in order to minimize the possibilities and to find a convergence for the position unknown, all the known position should be used. It is needed to be said that in order to use this method a minimum of 3 devices in a 2-D case and a minimum of 4 in a 3-D case would be needed to find this convergence, because with less devices different positions would satisfy the minimization problem.

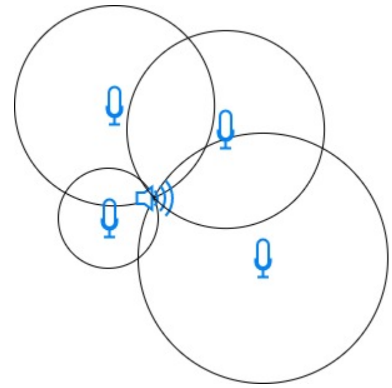

Figure 1: Geometrical set up

The method used to solve this is a optimization problem in which it is required to solve the next least-squares problem:

$$\begin{gathered} argmin \sum_{m=1}^{M} ||||s_l - r_m|| - d_{l,m}||^2 \\ \text{subject to } s_{l,min} \leq s_l \leq s_{l,max} \end{gathered} \tag{4}$$

where s_l is the unknown position of the lth device, r_m is the known position of the mth device, $d_{l,m}$ is the estimated distance from the lth device to the mth device obtain from RIR measurements and $s_{l,min}$ and $s_{l,max}$ are the lower and upper bounds for the unknown position of the lth device, that correspond to the bounds of the room in which the RIR is measured.

In order to minimize this least-squares problem a sequential least squares programming (SLSQP) method is used. This method is a kind of sequential quadratic programming method, in which an optimization subproblem is set up to find a solution to this subproblem that is used to construct a new iteration. This process is repeated for more iterations and then a solution can be found. Although it resembles the Newton methods for solving nonlinear algebraic systems of equations, the fact that there are constraints in this kind of problem makes the method more complicated[4].

3.2. Position calibration method with delay estimation

In this case the scenario for the problem is the same as the previous section but the delay introduced in the processing chain is taken into account. That means that a new variable is needed to be estimated and so, assuming all the process to obtain the TOA done, the method used is the same optimization problem as before but with this new variable of the delay, getting the next problem:

$$\begin{gathered} argmin \sum_{m=1}^{M} ||||s_l - r_m|| - (d_{l,m} - \tau)||^2 \\ \text{subject to } 0 \leq \tau \leq \tau_{max} \text{ and } s_{l,min} \leq s_l \leq s_{l,max} \end{gathered} \tag{5}$$

where all the variables are the same as before, being τ the desired delay to be estimated and τ_{max}, the upper bound of the unknown delay, which it should be infinite but in order to get a result in the optimization problem it will be set as 3 miliseconds.

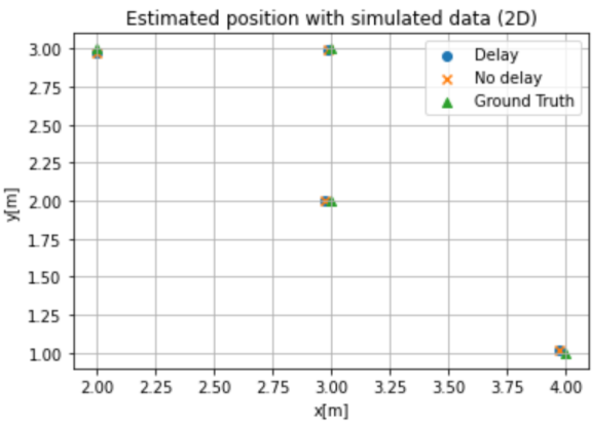

Figure 2: Estimated positions with simulated data (2D)

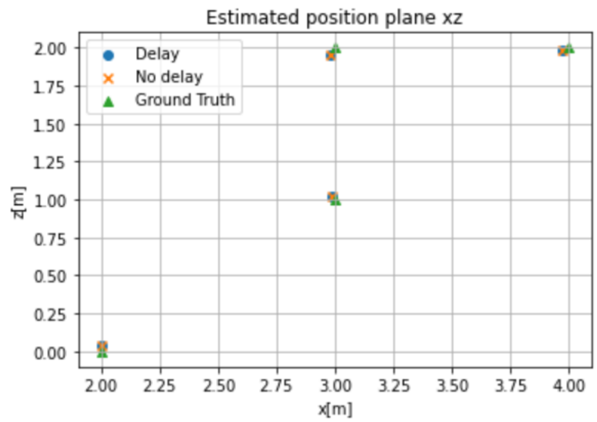

Figure 3: Estimated positions with simulated data (3D)

Since τ does not depend on the number of unknown positions, sometimes it is convenient to solve the following optimization problem:

$$\sum_{l=1}^{L}\sum_{m=1}^{M} ||P_s - (D - \tau 1_{L,M})||_F^2$$
$$\text{subject to } 0 \leq \tau \leq \tau_{max} s_{l,min} \leq s_l \leq s_{l,max} \tag{6}$$

where P_s is a $L \times M$ matrix for wich $[P_s]l, m = ||s_l - r_m||$, D is a $L \times M$ matrix for wich $[D]l, m = d_{l,m}$, $1_{L,M}$ is a $L \times M$ matrix full of ones and $||\cdot||_F$ is the Frobenius norm.

To solve this optimization problem the same method (SQLSP) is applied as in the previous case.

4. IMPLEMENTATION

In this section the implementation of the position calibration method described before is explained. In order to implement the work the programming language used was $Python$. In $Python$ is needed to import libraries in order to use different functions and make the implementation easier. In this project we use libraries like: *'scipy'* for implementing functions as interpolation or the SQLSP method (among others), *'pyroomacoustics''* for designing artificial datasets of room impulse responses, *'numpy'* 'pandas' for an easier treatment of the data used and *'tkinter'* for designing a GUI.

Regarding the functions used for the implementation, four main functions have been implemented. Two of them are dedicated to find the direct path, the TOA and the distance between two devices, these are *'find_directPath'* that finds the direct path of a given room impulse response and *'compute_distance'*, that computes the distance between two devices given the room impulse response. It uses the *'find_directPath'* function.

In order to implement the optimization problem a main function has been made, it takes as input the data of the room impulse responses, the known positions, the bounds of the room, the number of unknown positions, the sampling frequency and the sound velocity, and it gives as an output the estimated unknown positions and the estimated delay. There are four different functions of this type since it is needed to have two for the 2-dimension case, one without estimation of the delay and one with the estimation, and other two for the 3-dimension case. In these functions can be found the other two functions mentioned before plus the optimization function given by the *'scipy'* library.

Finally, a GUI has been implemented to give the user an easier experience of use. A longer documentation of the implementation can be found in the github[1].

5. RESULTS

In this section the results obtained are shown. Two types of scenarios have been used; one with simulated data using the *'pyroomacoustics'* framework in python and the second with dataset of $RIR's$ of real devices.

5.1. Simulated data

'Pyroomacoustics' is a framework that allows, among other things, to the user to simulate rooms of different sizes, add to them microphones and sources, and simulate the room impulse responses for each pair of microphone and source. Therefore, this framework was used to create a dataset with the simulated room impulse response. Many configurations can be done with it and in this report two cases are shown: one in two dimensions, having four sources with unknown positions and nine microphones with known ones, and one in three dimensions with the same assumptions for the sources and microphones position but varying their height. In both cases, the delay was of 0 seconds. However, a fake delay was included in a different dataset to test the quality of the delay estimation of the method. The estimated position for both cases are shown in figures 2 (2-dimension) and 3 (3-dimension).

In these figures the results taking into account the delay and not taking into account the delay are shown. In Fig. 2 (2-dimension) the plane *'xy'* is shown. In the case of the figure 3 just the plane *'xz'* is shown instead of the three planes *'xy'*, *'xz'* and *'zy'* due to a matter of space and cleanliness of the report. Moreover, this figure shows how the algorithm works including the *'z'* coordinate in the equation just by showing the plane *'xz'*. The plane *'yz'* has similar results to this one, meaning that the estimated positions taking into account the delay and not taking it into account are as close to the ground-truth positions as the plane *'xz'* shown in figure 3.

[1] https://github.com/MrMidc/Project_Course

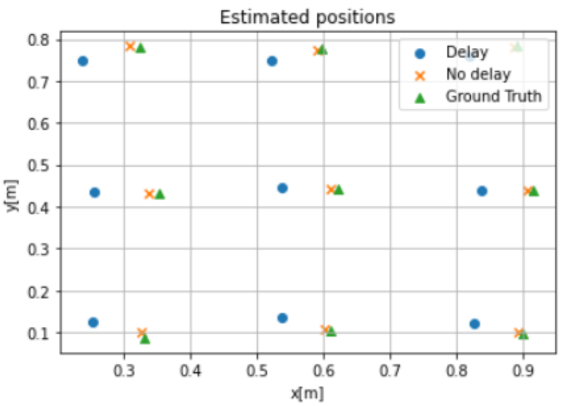

Figure 4: Estimated positions with real data (2D)

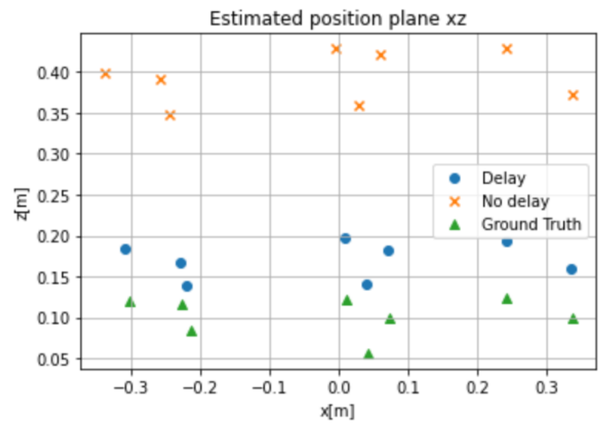

Figure 5: Estimated positions with real data (3D)

As it can be seen the estimated positions given by the calibration method are satisfactory for each of the sources.

5.2. Real data

Two datasets were used to try the algorithm with real data. One of the datasets (let us call it *'dataset 1'*[5][6]) uses just two coordinates (*'x'* and *'y'*) to express the known positions of the microphones, so it is a 2-dimension case. The second dataset (let us call it *'dataset 2'*) uses the three coordinates (*'x'*, *'y'* and *'z'*), so it is a 3-dimension case. In both datasets the delay was of 0 seconds.

Dataset 1 was made using 32 microphones with known positions and 9 sources with unknown positions. In the case of *dataset 2*, 21 microphones with unknown positions and 10 sources with known position were used.

In figure 4 are shown the results for the *dataset 1*. The estimated position of the sources using the method explained in this report can be seen, both taking into account the delay and not taking it into account. It can be seen how the results are better if the delay is not taken into account than if it is taken into account. When the delay is considered in the method all the estimated x coordinates for each source show an error in average of 7 cm.

In figure 5 the estimated position of the microphones for the *dataset 2* are shown. In this case, as for the simulated data, just the xz plane is shown, again due to a matter of space and cleaning in the report. Moreover, this plane xz is shown since it makes the noticeable difference in the results. In the xy plane the results are as satisfactory as a 2D case. However, as it can be seen the estimated z coordinate is not that accurate with the ground-truth positions, having an bigger error if the delay is not taken into account than if it is not. In the figure are shown just 8 estimated positions instead of the 21 unknown as a matter of clarity since the other results are similar

6. CONCLUSIONS

A calibration method for acoustic devices (sources and microphones) localization has been developed. We have shown how the algorithm works in a controlled scenario with simulated data both in a 2-dimension case and a 3-dimension case, and taking and not taking the possible delay into account in the method. In all the possible cases the method finds a satisfactory solution close to the exact locations expected. Therefore, we can say that the algorithm works as expected. However, we also tested the method with real data measured in a real situation with real acoustic devices. In the case of a 2-dimension dataset the method works good when the delay variable is not taken into account, obtaining results close to the ground-truth positions. If the delay is taken into account the precision is not as good as the previous case but still acceptable. When we talk about the 3-dimension case we find that the estimated position for two out of three coordinates (x and y) works very good, but the third coordinate (z) is not estimated correctly varying the error from taking into account the delay (smaller error) and taking it into account (bigger error).

Regarding the delay estimation, for both cases in 2-dimensions and 3-dimensions the delay was estimated correctly by the method, being 0 in both (simulated and real data) cases. As a matter of testing the method, we included a fake delay in the datasets to test the method and also in this case the estimated delay was correct.

With this results we can say that the calibration method proposed in this report works since with simulated data we obtain almost perfect results. Some reasons could justify this results with real data, among them, the lack of knowledge of the measurements of this data since it was given to us and the human error could be introduced at some point. Moreover, in this cases of optimization problem with a large number of variables there are a lot of local minimums that could introduce this errors found in this report. Some methods to avoid this local minimums and to find the real minimum value are developed in other research studies [1] improving the results of the acoustic device localization, a possible improvement for this research could be to include this kind of methods.

7. REFERENCES

[1] Le, T.-K. and Ono, N. and Nowakowski, T and Daudet, L. and De Rosny, J. "Experimental validation of TOA-based methods for microphones array positions calibration". *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

[2] Antonello, Niccolo and De Sena, Enzo and Moonen, Marc and Naylor, Patrick A. and van Waterschoot, Toon and Antonello, Niccolo and De Sena, Enzo and Moonen, Marc and Naylor, Patrick A. and van Waterschoot, Toon. "Room Impulse Re-

sponse Interpolation Using a Sparse Spatio-Temporal Representation of the Sound Field”, in *IEEE Press*, 2017.

[3] Kelly, Ian J. and Boland, Francis M. “Detecting Arrivals in Room Impulse Responses with Dynamic Time Warping”, in *IEEE Press*, 2014.

[4] Zhengqing Fu, Goulin Liu, Lanlan Guo, ”Sequential Quadratic Programming Method for Nonlinear Least Squares Estimation and Its Application”, *Mathematical Problems in Engineering*, vol. 2019, Article ID 3087949, 8 pages, 2019.

[5] M. Pezzoli, J. J. Carabias-Orti, M. Cobos, F. Antonacci,and A. Sarti. ”Ray-space-based multichannel nonnegative matrix factorization for audio source separation”. *IEEE Signal Processing Letters.*, 2021.

[6] M.Pezzoli, L.Comanducci, J.Waltz,A.Agnello, L.Bondi, A.Canclini, and A. Sarti. “A dante powered modular microphone array system”, in *Audio Engineering Society Convention 145. Audio Engineering Society*, 2018.