POLITECNICO
MILANO 1863

Sound Analysis, Synthesis And Processing

Module 1: Digital Audio Analysis and Processing

# Source Localization Assignment

Academic Year 2023/2024

Enrico Dalla Mora, Gabriele Baroli

May 10, 2024

## Contents

# 1 - Introduction

The aim of this assignment is to deal with the acoustic source localization problem by employing a *delay-and-sum beamformer* approach for spatial filtering, starting from data acquired through a *uniform linear microphone array (ULA)*. This is achievable, under a set of assumptions and approximations which are later specified, by evaluating the propagation time delay with which the various microphones receive the source signal. The physical problem is schematized in figure (1) in which M identical omnidirectional sensors are uniformly spaced on a line. $\theta$ is defined as the planar angle referring to the source position, while $d$ measures the distance between two consecutive sensors.

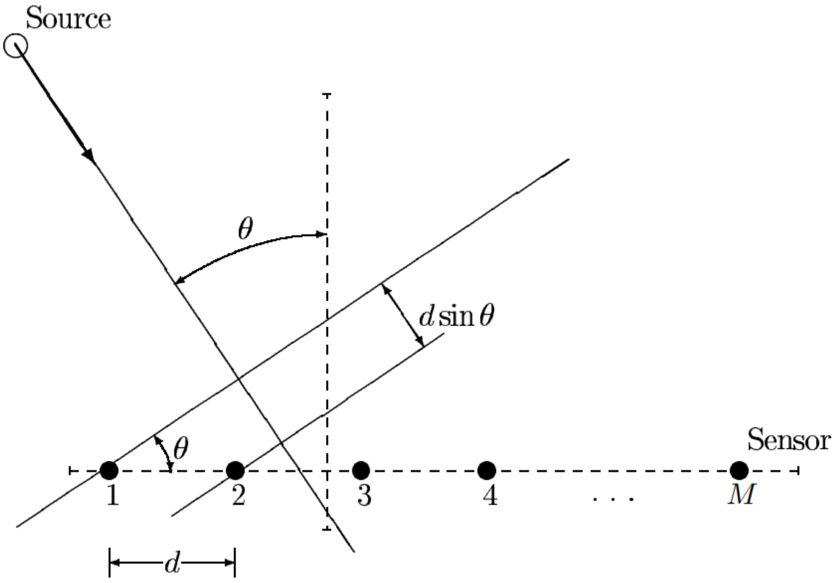

Figure 1: Uniform linear array and source propagation model.

## - Assumptions

- A single wide-band source is present. Given that the spatial filtering technique employed is suitable only for narrow-banded sources, the results will be obtained by processing a number of bands separately and then averaging the estimates.
- The propagation is homogeneous, reverberation is neglected and no dispersion is taken into account.
- Sensors and sources lie on the same plane → 2D geometry allows to characterize the *direction of arrival (DOA)* by means of the angle $\theta$ as previously seen in figure (1).
- Sources are located in the far-field and therefore wavefronts are planar. This assumption highly simplifies the problem, allowing sources to be characterized entirely by the direction of arrival.
- The sensors are ominidirectional and identical therefore it can be assumed that their impulse response is constant: $H_1(\omega) = H_2(\omega) = ... = H_M(\omega) = 1$.

Given these assumptions, the time delays $\tau_k$ can be expressed as a function of the angle $\theta$:

$$\tau_k = (k-1)\frac{d\sin(\theta)}{c}, \quad for \quad \theta \in [-90°, -90°] \quad and \quad k = 1, 2, ..., M \tag{1}$$

These time delays can be employed in defining a propagation vector $\mathbf{a}(\theta)$ as:

$$\mathbf{a}(\theta) = \begin{bmatrix} 1 & e^{j\omega_s} & \dots & e^{j(M-1)\omega_s} \end{bmatrix}^T , \tag{2}$$

where

$$\omega_s = \omega_c \frac{d \sin\theta}{c} \tag{3}$$

is the so-called *spatial frequency.*

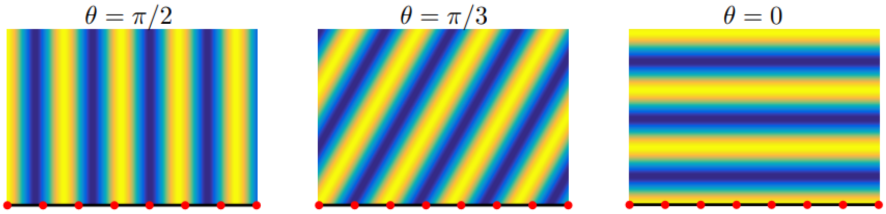

Figure 2: Wavefronts impinging the ULA for different directions of arrival $\theta$.

This vector will be crucial in the retrieval of the *Direction of Arrival.* The procedure is treated in the following sections.

## 2 - Spatial Filtering Technique

As stated, the technique involved in the assignment is the *delay-and-sum beamformer.* This approach consists of computing a spatial bandpass filter which attenuates all the signals coming towards the array except the ones in the desired directions. This is done by properly delaying the signals caught by each receiver, then summing them together, hence the name. Since the goal of this assignment is to localize a moving source, the strategy is to scan the entire space at different time instants, and evaluate the so-called *"pseudo-spectrum"* $p(\theta)$ of the source. The index of the peak of this function should provide a good estimate of the source position at that time.
In doing so, the spatial aliasing phenomenon should be taken into account: it can be conveniently expressed in function of the microphone spacing $d$ in the array as:

$$f_{max} \leq \frac{c_{air}}{2d} \tag{4}$$

where $c_{air}$ is the sound velocity in air, which is considered equal to 343 m/s. Given that the array has a length of $L = 45\,\text{cm}$ and it is constituted by $M = 16$ elements, the spacing results to be $d = 3\,\text{cm}$. The maximum frequency allowing to avoid aliasing therefore results as $f_{max} = 5717\,\text{Hz}$. Since the sampling rate at which the *ULA* works is $F_s = 8000\,\text{Hz}$, the spatial aliasing condition is less constraining than the time aliasing one, so we can operate up to the Nyquist frequency $f_{max} = \frac{Fs}{2} = 4000\,\text{Hz}$.

# 3 - MATLAB Implementation

The code is divided into three source files: `main.m`, which takes care of loading the data, launching the processing and displaying the results, a `das_filter.m` function which is responsible for the DOA estimation, and finally a `my_stft.m` function which is the innermost part of the code, and as the name suggests it is a Short Time Fourier Transform implementation which works on two-dimensional arrays.

Since a short time processing approach is required to localize the source in time, the first step is to extract frames from the original signals. This is done by employing a Hanning window of length $K = 1024$ samples and a hop size of 1/2 the window length, for optimal overlap of the different frames. Considering the window length and the sampling frequency of the array, the chosen window determines a spectral resolution of

$$B_w = \frac{4}{K} F_s = 31.25\,\mathrm{Hz}$$

and a temporal resolution of

$$\Delta t = K F_s = 128\,\mathrm{ms}$$

which can be regarded as a good trade-off between time accuracy and peak separation. An angular variable `theta` is defined from $-90°$ to $90°$, with $1°$ spacing, and for each time frame the `das_filter` function is called.
The input arguments are the windowed audio signals, the sampling frequency, the number of channels, the angular variable, the sound velocity and the array spacing. Inside `das_filter.m`, the function `my_stft` is invoked, with input arguments being the signals, the sampling frequency, the number of channels and the hop size. Here, frames of length `w_len` = 256, are extracted with a unitary hop size, as detailed in *DAAP-10_microphone_arrays_spatial_methods.pdf.* This action is equivalent to windowing the signals with a rectangular window of length `w_len` = 256, ensuring a spectral resolution of approximately

$$B_w = \frac{2}{\texttt{w\_len}} F_s = 62.5\,\mathrm{Hz},$$

Inside a `for` loop, the FFT of the windowed signals are computed and placed in the correct spot in the 3-dimensional array `y`, which stacks the STFTs for each channel. The returned values are then a unilateral version of `y`, plus `t` and `f`, which are respectively the coordinated time axis and frequency axis.

Note that, the first windowing, responsible of extracting time frames to be processed, sets the temporal accuracy of the algorithm, while the second windowing, employed in calculating the stft of the different time frames, sets the frequency resolution and consequently the number of frequency bands for which the evaluation is conducted.
The next step is to compute the propagation vector $\mathbf{a}(\theta)$, and the sample estimate of the covariance matrix $\hat{\mathbf{R}}$. In doing so, the number of frequencies at which the narrow-band processing takes place is set: given the spectral resolution of the STFT, a band each 62.5 Hz, starting from 31.25 Hz up to the Nyquist frequency is chosen. This ensures the least degree of redundancy possible.

The propagation vector is computed as in Eq. 2 while the sample estimate of the covariance matrix is given by:

$$\hat{\mathbf{R}} = \frac{1}{K} \sum_{t=1}^{K} \mathbf{y}(t)\mathbf{y}^H(t). \tag{5}$$

Inside das_filter.m, these two quantities are computed for each band, and then stored respectively in the variables a and cov_est. Finally, the pseudo-spectra can be computed according to the following equation:

$$p(\theta) = \frac{\mathbf{a}(\theta)^H \hat{\mathbf{R}} \mathbf{a}(\theta)}{M^2}, \tag{6}$$

one for each band. The average pseudo-spectrum, obtained via an arithmetic mean, is then returned to main.m inside the variable avg_pseudo_spec. The last block of instructions inside the loop in main.m takes care of localizing the maximum of the pseudo-spectrum and its index, so to localize the source DOA within the time frame.

# 4 - Results

The first thing to be noticed is highlighted by the signals spectra in figure (4). The signals are actually generated by a wide-band source, as specified by the assignment. Moreover the spectrograms allow to identify the noisy nature of the input since no frequency stands out from others in terms of magnitude. This means that averaging the pseudo-spectra across the whole frequency range can be done regardless of the specific band magnitude, allowing for consistent results.

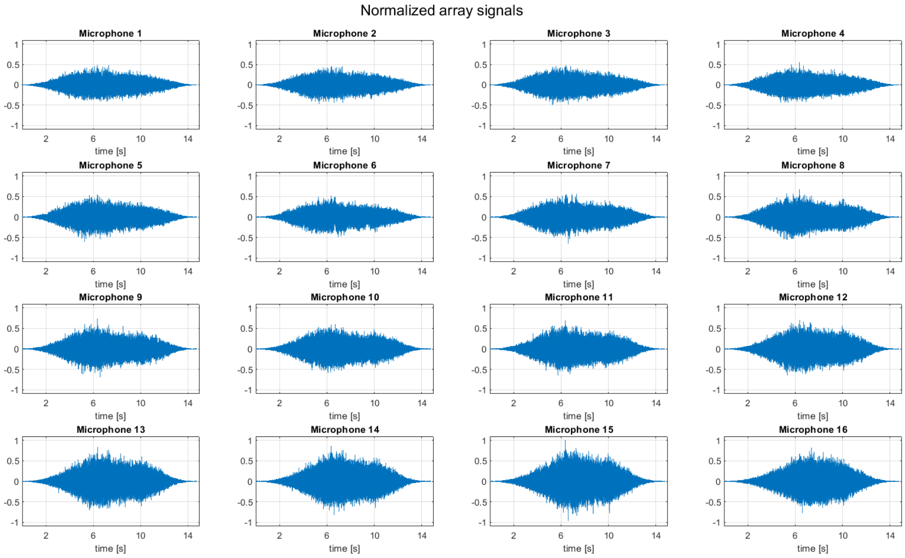

Figure 3: Normalized input signals for each of the 16 channels.

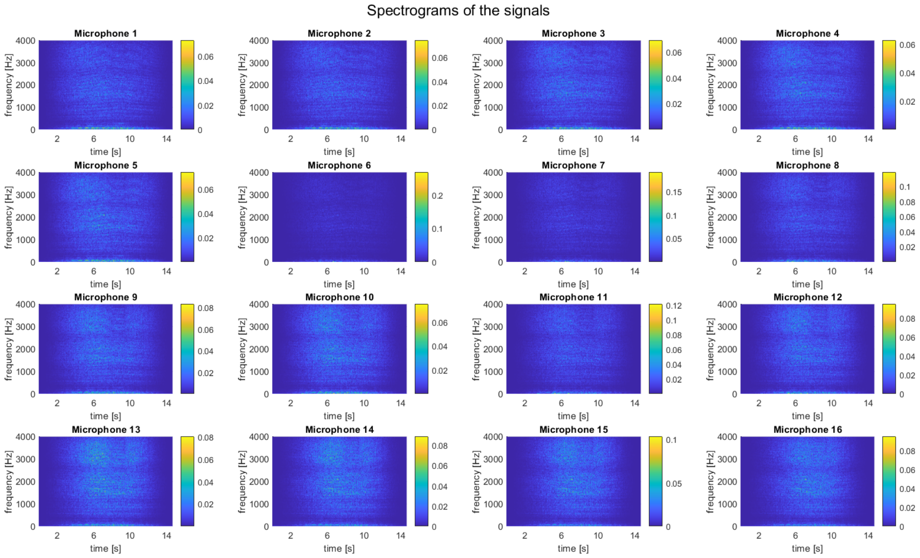

Figure 4: Spectrograms of the input signals for each of the 16 channels.

The following figure depicts the evolution of the frequency-averaged pseudo-spectrum over the duration of the signal. Each time frame has been normalized for visualization purposes. The figure allows to easily identify the time resolution as well as the estimated DOA which is better represented in the lower graph where the maximum value from the pseudo-spectrum has been extracted and associated to the corresponding direction of arrival.

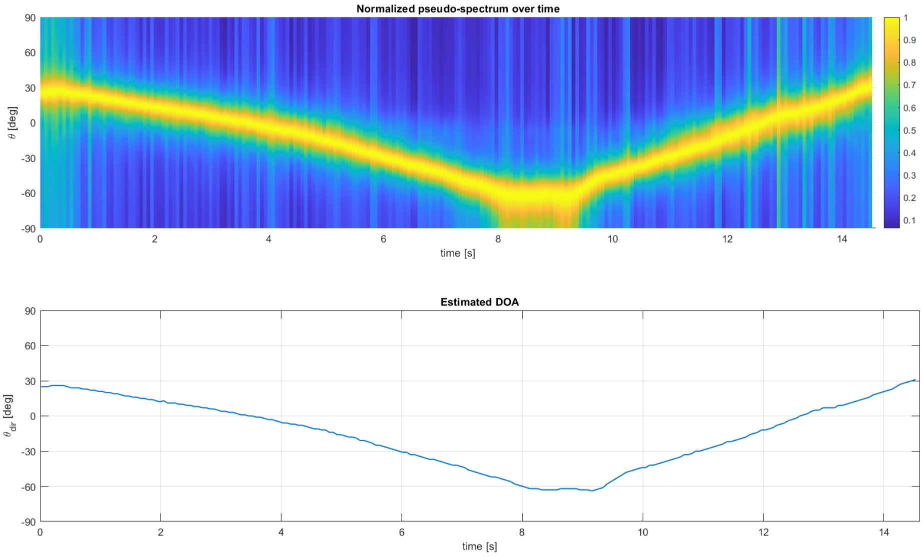

Figure 5: Frequency averaged pseudo-spectrum as function of time and angle of incidence and relative estimated DOA.

The same results are also reported in figure (6). Here the DOAs are represented as arrows departing from the centre point of the array, as a reference, and spanning the angular coverage of the source movement. An animation representing the arrow moving in time is found in the file `doas.mp4`. The algorithm provides reasonable results but it is not suitable for real-time audio processing due to the computational effort required. Lowering the frequency resolution would surely improve performances but with low chances of making the algorithm capable of real-time operation.

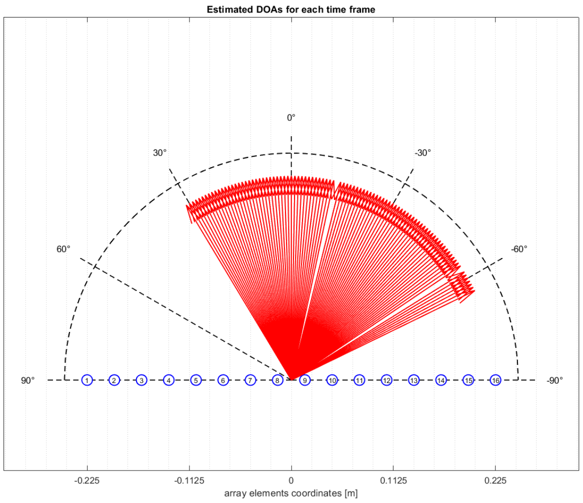

Figure 6: Estimated DOA for different time instants with respect to the ULA axis. The microphones composing the array are represented as blue circles, and are numbered to associate to each receiver the corresponding recording.

Lastly, a brief consideration on the *ULA* geometry can be conducted. The width of the main lobe of the filter, which determines the angular resolution of the array is approximately given by:

$$\frac{\lambda}{l|cos(\theta)|} = \frac{\lambda}{(M-1)d|cos(\theta)|} \tag{7}$$

In the same way, the *ULA* will be capable of distinguishing two sources with a minimum separation angle of $\Delta\theta = \frac{\lambda}{l}$. This means that, generally, increasing the number of microphones improves the resolution and accuracy of DOA estimation. More microphones provide additional spatial samples, allowing for finer spatial sampling, better discrimination between different directions of arrival and a higher angular resolution. The same effect is achieved by increasing the distance $d$ between microphones but with the undesired effect of lowering the upper limit of the spatial aliasing condition. This means that an appropriate *ULA* configuration will depend on the application and the spectral content of the source, requiring a proper trade-off between angular resolution and maximum frequency of analysis.