MUSIC for Direction of Arrival

Strebel Sophie
M.S Communication Systems
EPFL
Lausanne, Switzerland
sophie.strebel@epfl.ch

Zhang Michael
M.S Communication Systems
EPFL
Lausanne, Switzerland
michael.zhang@epfl.ch

***Abstract*—Estimating the direction of arrival (DoA) of multiple signals is crucial in sensor array signal processing. Over the past years, many algorithms have been developed to solve this problem but the most popular one by far is the multiple signal classification (MUSIC) algorithm. MUSIC leverages linear properties of the signal and noise subspaces to find directions of arrivals. Despite its popularity, it has been shown that MUSIC's performance drastically decreases when working with coherent and broadband signals. Hence, we propose an overview of MUSIC variants to mitigate those limitations.**

***Index Terms*—MUSIC, DoA, signal processing, Short-time Fourier transform**

I. Introduction

Direction of Arrival (DoA) focuses on finding the direction, i.e. the angle, of arrival of a signal from a source based on the received signals from an array of microphones. It is used for many applications, such as radio, communications, sonar, and rescue missions for example in the case of earthquakes [1] [2]. Multiple algorithms such as MUSIC [1], ESPRIT [2], FRIDA [5], SRP [7], TOPS [8] have been developed for DoA, all of them with their advantages and drawbacks.

The standard multiple signal classification (MUSIC) algorithm is based on the eigenstructure of the covariance matrix and is very robust to the presence of noise. Nevertheless, MUSIC requires strong assumptions on the problem formulation [1] [3]:

- a narrowband signal
- uncorrelated sources
- more microphones than sources

Some alternatives have been proposed to mitigate those assumptions. ESPRIT has shown better results than MUSIC in the case of correlated signals and a lower computational complexity [2]. On the other hand, it offers a lower resolution than the MUSIC algorithm in the case of uncorrelated signals [4].

FRIDA, another algorithm for DoA, has been shown to work better than other methods in the case of lower SNR and for smaller separation angles between sources [5]. The main drawback of another algorithm called SRP is its high complexity as the search space increases [6] [7]. Finally, TOPS provides the best results for in-between mid-SNR ranges, whereas methods for coherent and incoherent signals work best for low and high SNR respectively [8].

Despite its drawbacks, MUSIC, which we will focus on in this work, is very widely used. We propose improvements to the algorithm to mitigate the algorithm's limitations concerning some of its assumptions.

II. Problem setup and MUSIC background

The general setup of the problem and workflow of MUSIC is as follows: first, we assume there are K sources in the far field of M microphones. Source k sends a signal $S_k(t)$ over a white Gaussian noise channel. Thus, microphone m picks up on the signals of all sources as well as a white Gaussian noise which is independent of all other microphones. The received signal of microphone m, located at coordinates (x_m, y_m), is:

$$x^{(m)}(t) = \sum_{k=1}^{K} S_k(t) e^{-j\frac{2\pi}{\lambda}(x_m \cos(\theta_k) + y_m \sin(\theta_k))} + n_m(t) \tag{1}$$

$$= A\boldsymbol{S}(t) + \boldsymbol{n}(t) \tag{2}$$

where $n_m(t)$ is a centered Gaussian noise with variance σ^2, and θ_k is the direction of arrival of source k. A is the matrix of shape $M \times K$ with

$$A_{m,k} = e^{-j\frac{2\pi}{\lambda}(x_m \cos(\theta_k) + y_m \sin(\theta_k))}$$

Note that because we assumed that the signal was narrowband, it is reasonable to assume that it has a single wavelength λ. [1] shows a similar equation but it assumes that microphones are aligned. We derive an arbitrary microphone configuration equation in subsection II-A.

To estimate $\theta_1, ..., \theta_K$, MUSIC calculates the empirical estimation of the covariance matrix R_x of the samples (processes $x^{(m)}$ sampled every $\frac{1}{f_s}$ for a sampling rate f_s) and computes its eigendecomposition. Using properties of independence of the noise and signal subspaces, it can be proven that

$$R_x = AR_sA^H + \sigma^2 I \tag{3}$$

where I is the identity matrix, R_s is the covariance matrix of the signal.

Since $\sigma^2 > 0$, R_x is a full-rank matrix if we assume uncorrelated signals and R_s has rank K. Thus with the eigendecomposition of R_x, the K largest eigenvalues describe the signal subspace whereas the $M - K$ smallest eigenvalues describe the noise subspace and are thus equal to σ^2. Using

this fact, we can write the following equations for those $M-K$ noise eigenvectors $v_{K+1} \dots v_M$:

$$\begin{aligned} R_x v_i &= \sigma^2 v_i \\ (AR_s A^H + \sigma^2 I) v_i &= \sigma^2 v_i \\ AR_s A^H v_i &= 0 \end{aligned}$$

Using the full rank assumption of R_s, we can find that for the eigenvectors $v_{K+1} \dots v_M$ (i.e. the noise eigenvectors), we have $A^H v_i = 0$. So the noise eigenvectors are orthogonal to the conjugate columns of A, that is, orthogonal to the conjugate of the steering vectors of each source:

$$a^H(\theta_k) = \left(e^{-j\frac{2\pi}{\lambda}(x_m \cos(\theta_k) + y_m \sin(\theta_k))} \right)^H_{m=1\dots M}$$

We leverage this orthogonality property by computing the spatial pseudo-spectrum:

$$P(\theta) = \frac{1}{||a^H(\theta) E_n||^2} \tag{4}$$

where the noise matrix is $E_n = (v_{K+1} \dots v_M)$ (so the columns of E_n are the noise eigenvectors of R_x). When θ gets close to one of the angles of the sources, the denominator goes to 0, and $P(\theta)$ shows spikes. Finally, we estimate the direction of arrival by searching for the peaks of $P(\theta)$. This is done using the scipy `find_peaks` implementation.

A. *Proof of distance of microphone to the origin*

In the original MUSIC work [1], the microphones are assumed to be aligned (meaning $x_m = 0$ and $y_m = (m-1)\Delta$ where Δ is the distance between microphones so the distance of the first microphone to the origin). However, microphones are not always arranged in a straight line. In this part, we will derive an equation to handle arbitrary microphone configurations. [1] shows the following equation:

$$x^{(m)}(t) = \sum_{k=1}^{K} S_k(t) e^{-j\frac{2\pi}{\lambda}\Delta(m-1)\sin\theta_k} + n_m(t) \tag{5}$$

which is a specific case of Equation 1. To prove Equation 1, we try to find the additional distance (which is $\Delta(m-1)\sin\theta_k$ in [1]) with respect to the origin in terms of the coordinates of microphone m and the source angle θ_k. The general setting is drawn in Figure 1.

Note that we compare the (possibly negative) "additional distance" traveled by the source to the origin because the distance between microphones is relative (thus we can place the origin and orientation of the coordinate system wherever we want). Let us take a generic microphone at (x, y) and see what happens compared to the origin.

For simplicity, we define the angle φ as in the drawing. We start by noticing that $\cos\varphi = \frac{d}{\sqrt{x^2+y^2}}$ and recalling that $\cos(a-b) = \cos a \cos b + \sin a \sin b$. Also, note that the two parallel lines are the lines from the source to (x, y) and to the origin respectively. We again assume that the lines are parallel because the source is in the far-field. The source signal "reaches" (x, y) first and then has to go through an additional distance d to reach the origin. So our goal is to express d in terms of x, y and θ. We see that $\arctan(\frac{y}{x}) = \theta + \varphi$, so $\varphi = \arctan(\frac{y}{x}) - \theta$. Then, we can use the decomposition formula above.

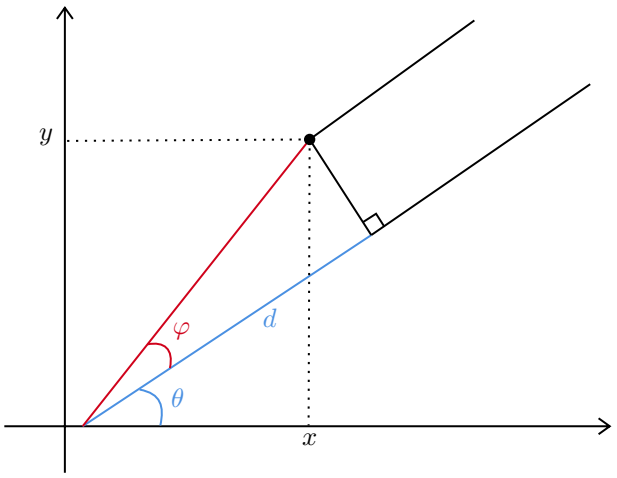

Fig. 1. Distance of a microphone to the origin

First, $\frac{d}{\sqrt{x^2+y^2}} = \cos(\varphi) = \cos(\arctan(\frac{y}{x}) - \theta) = \cos(\arctan(\frac{y}{x}))\cos(\theta) + \sin(\arctan(\frac{y}{x}))\sin(\theta)$.

Since

$$\cos(\alpha) = \frac{1}{\sqrt{1+\tan^2(\alpha)}}$$

and

$$\begin{aligned} \sin(\beta) &= \sqrt{1-\cos^2(\beta)} = \sqrt{1 - \frac{1}{1+\tan^2(\beta)}} \\ &= \sqrt{\frac{1+\tan^2(\beta)-1}{1+\tan^2(\beta)}} = \frac{\tan(\beta)}{\sqrt{1+\tan^2(\beta)}} \end{aligned}$$

this becomes:

$$\begin{aligned} \frac{d}{\sqrt{x^2+y^2}} &= \frac{1}{\sqrt{1+\frac{y^2}{x^2}}}\cos(\theta) + \frac{\frac{y}{x}}{\sqrt{1+\frac{y^2}{x^2}}}\sin(\theta) \\ &= \frac{x}{\sqrt{x^2+y^2}}\cos(\theta) + \frac{y}{\sqrt{x^2+y^2}}\sin(\theta) \end{aligned}$$

and finally, this implies that $d = x\cos(\theta) + y\sin(\theta)$. Essentially, we are projecting the microphone coordinates onto the steering vector for an angle θ.

Note that our θ is not the same as the angle used in [1] (the one we used before was actually $\frac{\pi}{2} - \theta$), but this does not matter as we only calculate the additional distance and as long as we stick to a single convention.

III. Materials and Methods

Despite its remarkable performance, MUSIC exhibits some limitations:

1) MUSIC assumes narrowband signals
2) Our formulation of MUSIC doesn't take into account the frequency symmetry of real signals
3) MUSIC assumes that the sources are non-coherent. This is necessary for the full rank assumption of R_s

In this section, we build upon the presented mathematical background to implement the standard MUSIC and variants. Here we will present three improvements to the standard MUSIC implementation:

- A MUSIC variant based on short-time Fourier transform
- The use of Hilbert transform on real-valued signals
- Spatial smoothing for aligned microphones
- A neural network to estimate DoA

Our methods include an implementation of the MUSIC algorithm as described above. We test this on simulated and real data. The results are discussed below. We use simulated data with various crafted signals, simulations, and real data, built on the project done last year.

A. MUSIC for broadband signals

Because of our assumptions, MUSIC can not handle broadband signals. To overcome this problem, we implement a variant of the MUSIC algorithm for DoA using the short-time Fourier transform (STFT) [9] [11]. Note that this has also been done with FRIDA and TOPS algorithms [5] [8] but we fill focus on the MUSIC implementation. The main idea is to consider a frequency range instead of a single main frequency and to average MUSIC across frequencies sampled in this frequency range. More formally, we denote $\boldsymbol{S_x^{(m)}}(t,f)$ the short-time Fourier transform of $x^{(m)}$. The spatial time-frequency distribution (STFD) is defined as follows:

$$\begin{aligned}\boldsymbol{D_x}(t,f) &= \begin{bmatrix} \boldsymbol{S_x^{(1)}}\boldsymbol{S_x^{(1)}}^H & \dots & \boldsymbol{S_x^{(1)}}\boldsymbol{S_x^{(M)}}^H \\ \vdots & \ddots & \vdots \\ \boldsymbol{S_x^{(M)}}\boldsymbol{S_x^{(1)}}^H & \dots & \boldsymbol{S_x^{(M)}}\boldsymbol{S_x^{(M)}}^H \end{bmatrix}(t,f) \\ &= \boldsymbol{S_x}(t,f)\boldsymbol{S_x}(t,f)^H\end{aligned}$$

where $\boldsymbol{S_x}(t,f) = \begin{bmatrix} \boldsymbol{S_x^{(1)}} \\ \vdots \\ \boldsymbol{S_x^{(M)}} \end{bmatrix}$

With this notation, we apply the linearity of STFT to Equation 2:

$$\begin{aligned}\boldsymbol{S_x}(t,f) &= \mathbf{STFT}\{\boldsymbol{S}(t)\}(t,f) \\ &= A\boldsymbol{S_s}(t,f) + \boldsymbol{S_n}(t,f)\end{aligned}$$

Which leads to:

$$\mathbb{E}\left[\boldsymbol{D_x}(t,f)\right] = A\boldsymbol{D_s}(t,f)A^H + \mathbb{E}\left[\boldsymbol{D_n}(t,f)\right] \tag{6}$$

where we again use the independence between the noise and the signal to eliminate the cross-terms. Notably, $\boldsymbol{S_n}(t,f)$ follows a complex Gaussian distribution with zero-mean. Thus $\mathbb{E}\left[\boldsymbol{D_n}(t,f)\right] = \tilde{\sigma}^2 I$

Using this fact, we notice that Equation 6 has the same structure as Equation 3 which means that we can apply the same MUSIC algorithm on the empirical STFD of $\mathbf{x}(t)$ to get the direction of arrivals. The empirical STFD can be computed as follows:

$$\bar{\boldsymbol{D}}_{\boldsymbol{x}} = \frac{1}{|\Omega|}\sum_{(t,f)\in\Omega} \boldsymbol{D_x}(t,f)$$

where Ω contains every time-frequency pair of the STFT that we are interested in (typically, we choose every f in a certain frequency range). MUSIC is then applied on $\bar{\boldsymbol{D}}_{\boldsymbol{x}}$ instead of the regular covariance matrix.

B. Standard MUSIC for real-valued signals

We also use a modification based on the Hilbert transform. The Fourier transform of real-valued signals is symmetric around the y-axis, thus the negative frequencies contain only redundant information. The Hilbert transform is used to remove these negative frequencies before applying MUSIC. Hence, we apply MUSIC on $\mathbf{H}\{x^{(m)}\}(t)$ for $m = 1 \dots M$ instead of $x^{(m)}(t)$.

This greatly impacts the results, as we discuss below. Note that intuitively, the idea is similar to STFT-based MUSIC, as we only consider positive frequencies in the STFT.

C. Spatial smoothing for correlated signals

Our implementation also includes the spatial smoothing improvement for correlated signals based on [1]. Instead of calculating the eigendecomposition of the covariance matrix of the received signals R_x, we calculate the eigendecomposition of

$$R'_x = R_x + JR_x^*J, \tag{7}$$

where

$$J = \begin{pmatrix} 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & 1 & 0 \\ \dots & & & & \\ 0 & 1 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \end{pmatrix}$$

and R_x^* is obtained by applying complex conjugation to each entry of R_x, that is, $(R_x^*)_{i,j} = ((R_x)_{i,j})^*$. This technique works because R'_x and R_x have the same noise subspace and thus the eigendecomposition gives the same noise eigenvectors, meaning that the estimated spatial pseudo-spectrum obtained from the noise eigenvectors of R'_x is also correct. This modification improves the results because in the case of correlated signals, R_x is rank-deficient. Since MUSIC relies on the eigenstructure of this matrix to find the direction of arrival. This modification keeps the same eigenspace and restores the rank of the matrix as averaging different "view" of the same received signals helps to blur the coherence.

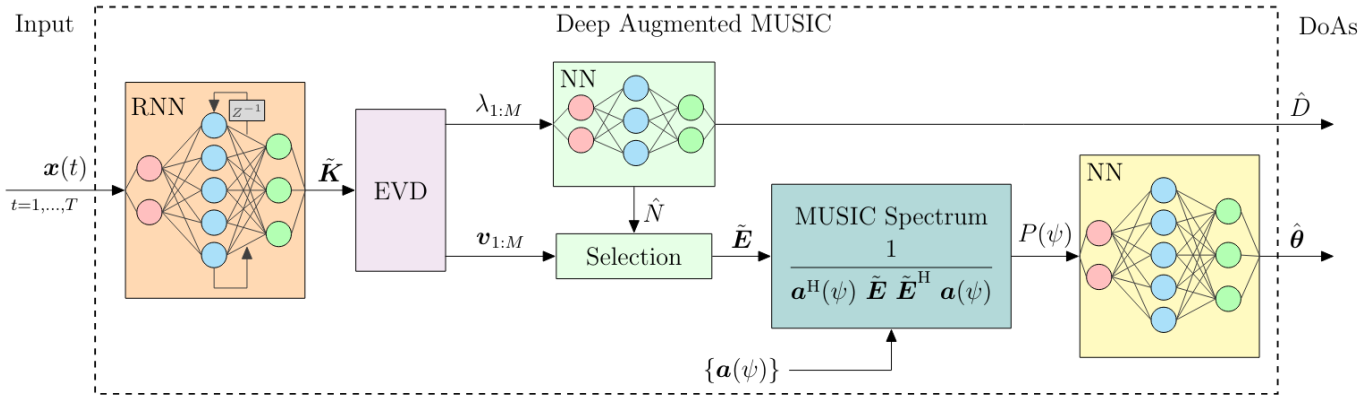

Fig. 2. Diagram of Deep Augmented MUSIC approach presented in [12] (the figure is taken from [12])

D. Neural Network

One of the greatest challenges in the direction of arrival estimations is dealing with correlated signals. One of the most common techniques to handle correlated signals is based on spatial smoothing that we presented. In practice, however, the number of coherent sources is unknown, and applying spatial smoothing is not so obvious [12]. Furthermore, spatial smoothing becomes more challenging with arbitrary microphone array setups. With the rise of machine learning, ongoing research has drifted towards data-driven signal analysis. In particular, a hybrid approach that combines the interpretability of MUSIC and the expressiveness of neural networks has been proposed in this work [12].

Figure 2 shows the general architecture of the Deep Augmented MUSIC algorithm (note that this figure is taken from [12]). The input is first handled by a Recurrent Neural Network (a Gated Recurrent Unit in the final version) that processes the samples at each timestamp. At each of those timestamps, the hidden state of the RNN keeps a representation of the signal up to this time. We add a dense linear layer that transforms the final hidden state to the pseudo-covariance matrix $\tilde{\boldsymbol{K}}$. This addition allows more flexibility in the choice of hidden state dimension for the RNN.

Then, we compute the eigendecomposition of $\tilde{\boldsymbol{K}}$ to separate the noise subspace and the signal subspace. The MUSIC spatial spectrum $P(\psi)$ is then computed similarly as Equation 4 from the noise subspace $\tilde{\boldsymbol{E}}$. A neural network is appended to the pseudo-spectrum to extract the direction of arrivals from the pseudo-spectrum estimation. The output of the neural network has the same size as the number of microphones. Because of the linear properties of MUSIC, the maximum number of sources that M microphones can predict is M.

Figure 2 also shows how Deep Augmented MUSIC can predict the number of sources $\hat{D}$ (and thus the dimension of the noise subspace $\hat{N}$) using the eigenvalues $\lambda_{1:M}$. Nevertheless, this step is unnecessary if the number of sources is known.

To train the neural network, we generate a dataset composed of noisy Gaussian processes that are altered by the steering vector. The objective of the neural network is to minimize the Root Mean-Square Permutation Error defined as follows:

$$\mathrm{RMSPE}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}) = \min_{\boldsymbol{P} \in \mathcal{P}_D} \left(\frac{1}{D} \left\| \boldsymbol{\theta} - [\hat{\boldsymbol{\theta}}]_{\boldsymbol{P}} \mod 2\pi \right\|^2 \right)^{\frac{1}{2}} \quad (8)$$

where $\boldsymbol{\theta}$ contains the true direction of arrivals. This loss is minimized over all permutations of $\hat{\boldsymbol{\theta}}$. Note that the loss should take into account the 2π periodicity of angles. $\mathcal{P}_D$ represents the set of all permutations of size D where D is the number of sources.

IV. Experiments and Discussion

A. MUSIC for different SNR

To understand how MUSIC behaves in the presence of noise, we look at different signal-to-noise ratios (SNR). We generate multiple narrowband signals from two sources based on some fixed signal-to-noise ratios and apply the standard MUSIC algorithm to them. Their spatial spectrums can be seen in Figure 3. Note that the SNR we use here is in db, that is: $SNR = 10 \log_{10} \frac{\sigma_s^2}{\sigma_n^2}$, where σ_s^2 is the variance of the signal and σ_n^2 is the variance of the noise and that the spectrum (y-axis) is shown in log-scale.

Figure 4 shows the corresponding width of the peaks, when the width is smaller, the peak is sharper. The red-dotted vertical lines show the actual direction of arrivals (note that this is the case in all figures with red-dotted vertical lines).

Figure 3 shows that MUSIC for DoA works much better for higher SNR as expected. That is, the peaks of the spectrum are much higher and sharper, even more than they seem since the scale is logarithmic. To compare the peaks more precisely, Figure 4 shows the peak width in radians. For both angles, the width is much lower for higher SNR values. We compute the MSE of the source angle estimation. It is lower for higher SNR (though they are all low, in the range of 10^{-3} and 10^{-4}).

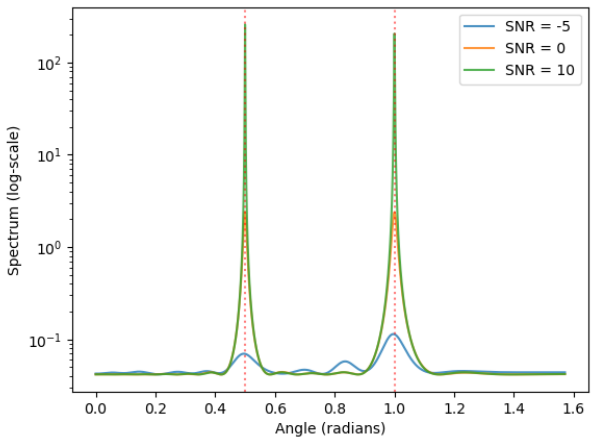

Fig. 3. Spatial spectrum as a function of the angle for different SNR values

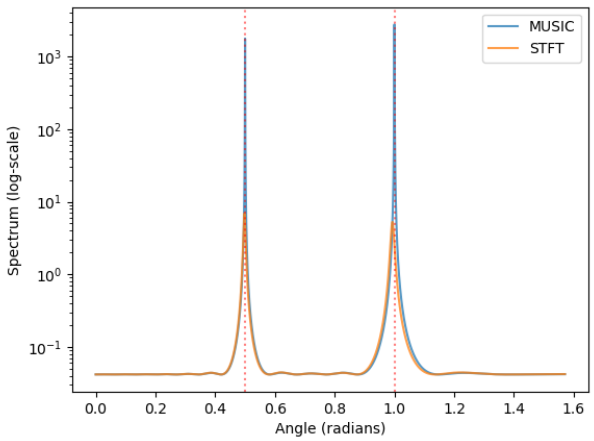

Fig. 5. Spatial spectrum of a narrowband signal for both MUSIC and STFT-based MUSIC with low-frequency range

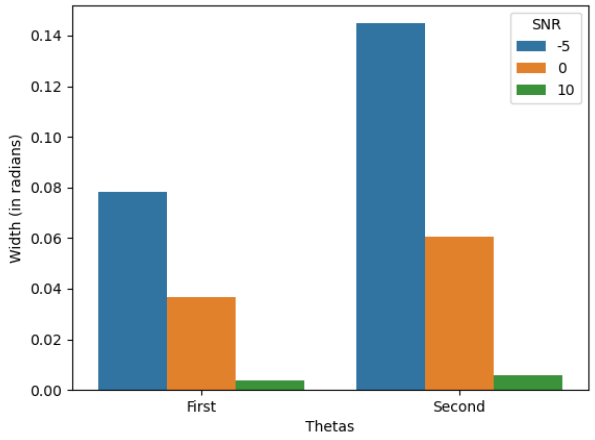

Fig. 4. Width of peak for different SNR values

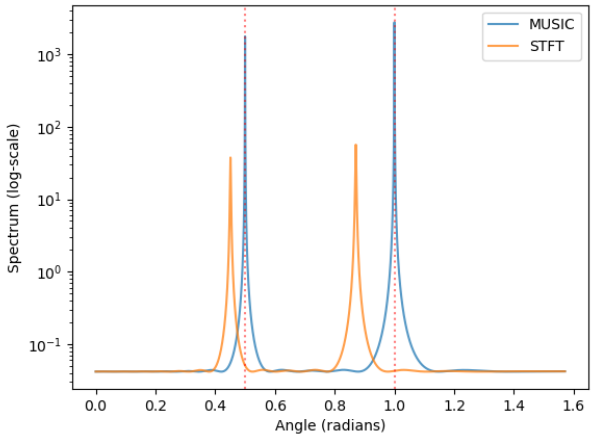

Fig. 6. Spectrum of a narrowband signal for both MUSIC and STFT-based MUSIC with higher frequency range

B. Narrowband signals

As a benchmark for our advanced tool based on the STFT, we generate narrowband signals and use the classical MUSIC algorithm and our STFT-based improvement to find the spatial spectrum. The result is shown in Figure 5 (again, the y-axis is in log-scale). Note that the spectrum on Figure 5 for STFT-based MUSIC is generated using a narrow frequency range centered around the main frequency. Figure 6 shows the resulting DOA when increasing the frequency range of STFT-based MUSIC. We see a shift in the spikes, which makes sense because by changing the frequency range, we change the wavelength λ and thus, using Equation 5, since the coordinates, distance and microphones indices are constant, the only thing that can change are the θ_k. So the direction of arrival is incorrectly estimated in the spatial spectrum.

The spectrum peaks on Figure 5 are higher and clearer for MUSIC compared to our STFT-based MUSIC implementation. Nevertheless, both algorithms show great precision when predicting the direction of arrivals for our experiment. However, on the other hand, when we increase the frequency range of STFT-based MUSIC, Figure 6 shows that STFT-based MUSIC's prediction gets worse. By increasing the frequency range, STFT-based MUSIC takes more noise into account leading to worse predictions.

It is important to note that the STFT-based MUSIC modification was built to surpass the limitations of MUSIC in the case of wideband signals. On the other hand, if we stick to narrowband signals, standard MUSIC seems to show better results and is more robust to the choice of hyperparameter than STFT-based MUSIC.

C. Wideband signals and runtime

As presented before, one of the assumptions made by MUSIC is the narrowband characteristic of the sources. This

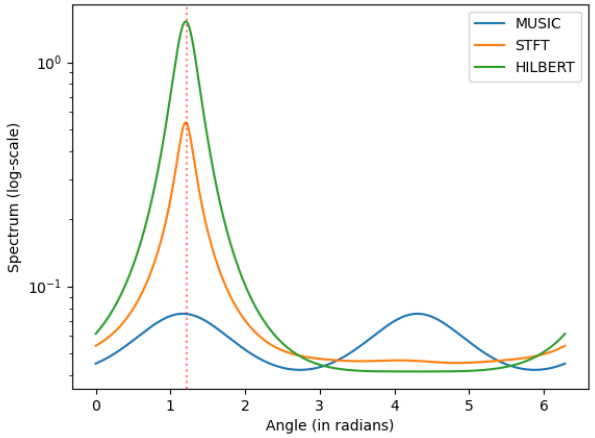

Fig. 7. Spatial spectrum of a wideband signal for standard MUSIC, STFT-based MUSIC, and MUSIC applied to the Hilbert transform of the signal.

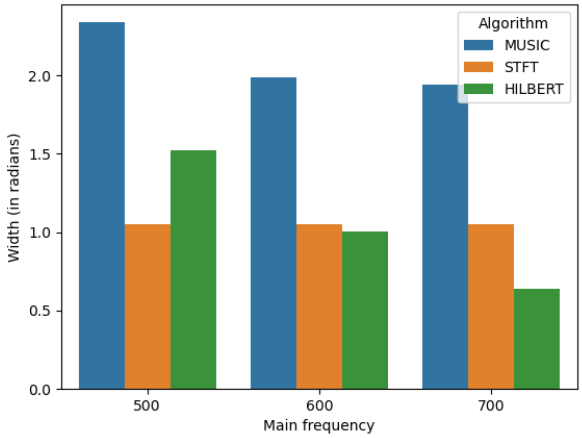

Fig. 8. Width of peak for different algorithms

is the main motivation for an STFT-based MUSIC. Our data is a real voice recording, which is wideband from a source at 70 degrees or roughly 1.22 radians (the data was taken from last year's submission). We choose the main frequency for standard MUSIC and Hilbert-based MUSIC by observing its Fourier spectrum. Similarly, for STFT-based MUSIC, we take a look at the frequency range and choose a frequency range of interest based on this Fourier spectrum.

The results are shown in Figure 7. The frequency of STFT-based MUSIC ranges approximately in $[0, 1800]$ Hz while the main frequency of standard and Hilbert MUSIC is roughly 600 Hz. Figure 8 shows the width of the peaks for each of the three algorithms for different main frequencies: 600Hz, as used for Figure 7 as well as 500 and 700 Hz.

Figure 9 shows the running time of our three algorithms when processing the wideband signal as a function of their spatial pseudo-spectrum resolution (i.e. how many points we choose to plot the pseudo-spectrum). We plot the average running time from 100 to 1000 points in our spectrum resolution over 10 different trials.

Figure 7 shows that MUSIC's performance worsens when working with real-valued wideband sources. The peak corresponding to the source is less visible than in Figure 5 and, because the data is real-valued and is thus symmetric in the frequency domain, there are two peaks even though there is only one source. The Hilbert transform and the STFT-based modifications remove the negative frequencies (frequencies between π and 2π) since they carry no additional information. Both advanced algorithms show clearer spatial spectrums than MUSIC. Figure 8 shows that Hilbert-based MUSIC has the sharpest peaks among the three algorithms studied.

Nevertheless, because of the large frequency range of the source, the Hilbert transform-based algorithm is sensitive to the choice of the main frequency. As mentioned in the results, we choose 600Hz as the main frequency, but Hilbert-based

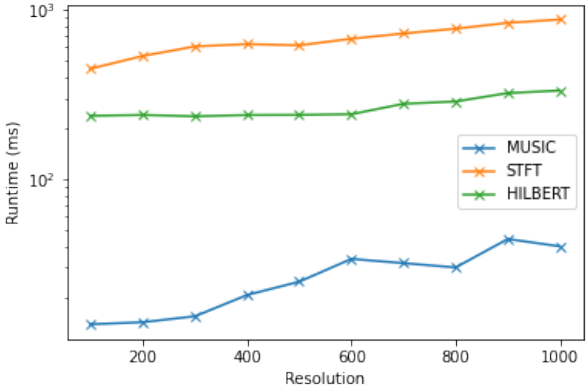

Fig. 9. Running times of the algorithms on the wideband signal as a function of the spatial pseudo-spectrum resolution

MUSIC's performance changes drastically if we choose the main frequency to be 500Hz or 700Hz instead. This is a key limitation of the Hilbert-based MUSIC approach when dealing with wideband signals as we select this main frequency by visually evaluating the FFT of the speech data. In real applications, we would like to set a range of frequency for speech (typically 0 to 1800Hz for speech for instance) and reliably run an algorithm with this choice of hyperparameter on any signal of the same type. The width of peaks depending on the main frequency as shown in Figure 8 indeed shows that the sharpness of peaks of the spatial spectrum of the Hilbert-based approach varies greatly depending on the main frequency, and can be much narrower or much wider than the STFT-based approach. There is thus a tradeoff between the potential performance of the Hilbert-based approach and the consistency of STFT-MUSIC.

We also compare the runtime of the three algorithms. Figure 9 shows a trade-off between accuracy and computational time. Both STFT-based MUSIC and Hilbert MUSIC have a

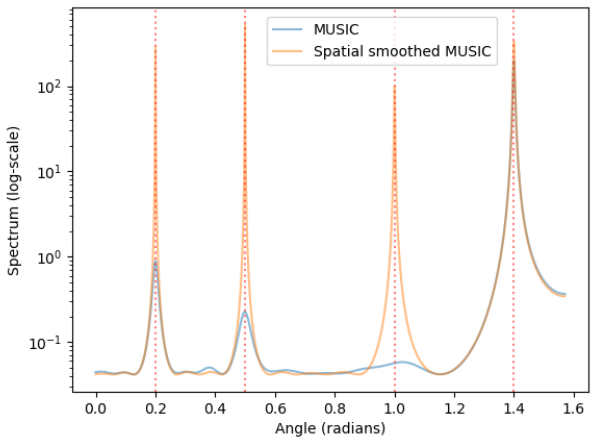

Fig. 10. Spatial spectrum of MUSIC and spatial smoothed MUSIC for four correlated sources

significantly higher runtime than standard MUSIC (more than 10 times slower than standard MUSIC), even though the only difference for Hilbert-based MUSIC is the computation of the Hilbert transform before applying MUSIC. Note that STFT-based MUSIC also depends on the size of the frequency range, if the frequency range is larger, STFT-based MUSIC becomes slower. Therefore, the precision of our predictions comes at a larger computational cost.

D. Correlated signals

Figure 10 shows the estimated spatial spectrum of the standard MUSIC algorithm and the spatially smoothed variation based on Equation 7. We use four sources situated at the same frequencies as the vertical lines on the plot: the source angles are $0.2, 0.5, 1$ and 1.4 in radians. The correlated sources are created as follows: the samples of the first, third and last sources $S_1(t), S_3(t)$ and $S_4(t)$ are represented by i.i.d. centered Gaussian samples. The samples of the second source are $S_2(t) = \frac{1}{2}S_1(t) + 2S_3(t)$, so a linear combination of two other source samples with different weights. The microphones here are aligned, that is, the coordinates of microphone m are $(0, m \cdot d)$ where d is the distance between microphones.

As we can see on Figure 10, the MUSIC spatial spectrum peak for the last source is almost the same as the for the spatial smoothed MUSIC. As mentioned before, S_1, S_3 and S_4 are iid, and $S_2 = \frac{1}{2}S_1 + 2S_3$. The peak is very sharp and the direction of arrival can be well recovered. The results are very different for S_1, S_2 and S_3. The spectrum is clearest for S_1, which is logical because S_1 is only correlated to one other signal, S_2, and only weakly since (using the fact that all samples are centered):

$$\begin{aligned}\mathbb{E}[S_1 S_2] &= \mathbb{E}\left[S_1\left(\frac{1}{2}S_1 + 2S_3\right)\right] = \frac{1}{2}\mathbb{E}[S_1^2] + 2\mathbb{E}[S_1 S_3] \\ &= \frac{1}{2}\mathbb{E}[S_1^2] + 2\mathbb{E}[S_1]\mathbb{E}[S_3] = \frac{1}{2}\mathbb{E}[S_1^2] \\ &= \frac{1}{2}\mathrm{Var}(S_1)\end{aligned}$$

So the spatial spectrum of MUSIC still looks more or less right for the first direction of arrival. The spectrum still has a peak for the second source, even though that source is correlated with two others, whereas for the third source, there is no visible peak at all. This is somewhat counter-intuitive: a source that is correlated to two of the others has a better estimation than a source that is highly correlated to only one of the others. One possible hypothesis is data normalization. Indeed, S_2 has a higher power than S_3 (because of the $\frac{1}{2}S_1$ we added to S_2), so that is the peak that "survives" whereas the peak due to S_3 "dies" completely as it is overshadowed by S_2. When we normalize the signals, the peaks for S_2 and S_3 are much more similar and both are still present, even though they were very low and wide.

On the contrary, we can see that the spatial smoothed MUSIC algorithm is a huge improvement as the peaks of the spatial spectrum are very sharp and very close to the actual directions of arrival. A few drawbacks of the spatial smoothed MUSIC algorithm is that the microphones need to be aligned (our tests with non-aligned microphones gave very bad spectrums and the directions of arrival were not estimated well at all).

E. Neural Network

To handle correlated signals, we leverage the expressiveness power of neural networks. [12] shows an implementation of such algorithm. We generate $10,000$ samples of setup with 2 coherent sources, $SNR = 10$ and we train the end-to-end Deep Augmented MUSIC model on one epoch. The training set consists of 200 random snapshots i.i.d. and each snapshot follows a normal distribution. Each of those snapshots is artificially altered using a steering transformation. In the end, we get 200 samples for each of our $M = 24$ microphones. Table I shows our training hyperparameters. Note that our implementation uses GPU computing power to accelerate the training process. Table II shows the results of our model along with the performance of other standard models with our generating testing set. Note that our testing set is generated similarly to the testing set.

End-to-end DA-MUSIC corresponds to our trained Deep Augmented MUSIC model. Standard MUSIC corresponds to the standard MUSIC algorithm presented in section II. The random predictor is a random model that predicts the direction of arrivals uniformly at random in $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$. The zero predictor always predicts 0 for the direction of arrival.

Figure 11 shows a prediction for a sample of our testing set. The blue lines represent the predicted direction of arrival. The red-dotted lines are the actual direction of arrivals. The

Parameter	Value
optimizer	ADAM
learning rate	1e-3
epochs	14
samples	10,000
batch size	100

TABLE I
TRAINING HYPERPARAMETERS

Algorithm	Test error
End-to-end DA-MUSIC	0.054
standard MUSIC	0.24
Random	0.69
Always zero	0.85

TABLE II
TEST ERROR ON THE DIFFERENT DIRECTION OF ARRIVAL PREDICTORS FOR TWO COHERENT SOURCES

plot also shows the spatial pseudo-spectrum generated by our Deep Augmented architecture.

According to our results presented in Table II, Deep Augmented MUSIC shows better performances on our generated testing set. Standard MUSIC has a test error almost 5 times higher than End-to-end DA-MUSIC. However, it must be important to note that our testing set is generated using the same distribution as our training set. It is still unclear if deep-augmented MUSIC performs well outside our setup with different source distributions and real data.

Despite its accuracy, Figure 11 shows that the spatial pseudo-spectrum generated by the deep-augmented MUSIC does not produce peaks like the other spectrums studied in this report. Because of this, it is almost impossible to interpret our model's results. This raises questions about further possible work. For instance, it would be interesting to discard the MUSIC setup and take a full data-driven approach. The hope is to trade potential interpretability for better performance.

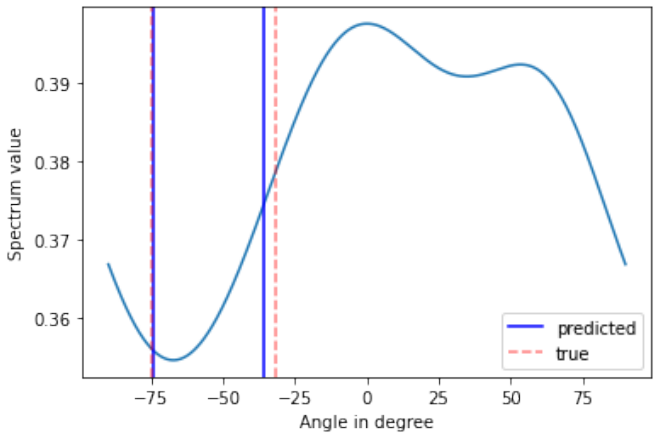

Fig. 11. Example of a Deep Augmented MUSIC prediction for two coherent sources coming from our generated testing set

V. CONCLUSION

We discussed the Direction of Arrivals with five algorithms based on the MUSIC method: standard MUSIC, spatial-smoothed MUSIC, Hilbert-based MUSIC, STFT-based MUSIC, and Deep Augmented MUSIC, each with its advantages and drawbacks. We studied three use cases: uncorrelated narrowband signals, correlated narrowband signals and uncorrelated real wideband signals.

In the case of narrowband uncorrelated signals, standard MUSIC seems to be the most well-suited algorithm. STFT-based MUSIC shows good performances but is very sensitive to the choice of frequency range hyperparameter.

For narrowband correlated signals, spatial-smoothed MUSIC appears to have the best performance. However, spatial smoothing can be difficult to implement, depending on the microphone array configuration. In our work, we only consider aligned microphones but other microphone configurations need further consideration.

With uncorrelated real wideband signals, standard MUSIC's performance drops because of the symmetry of real-valued signals in the frequency domain. To handle real-valued signals, we present the Hilbert-based MUSIC and the STFT-based MUSIC. The Hilbert-based version shows better performance than standard MUSIC but is once again dependent on the choice of main frequency hyperparameter. The STFT-based MUSIC on the other hand shows less sensitivity to the choice of hyperparameter while keeping similar performances as Hilbert-based MUSIC. However, those performances come at a significant computational cost compared to standard MUSIC

VI. SOURCE CODE

The code used for this report can be found in this repository.

REFERENCES

[1] H. Tang, and S. Nordebo, "DOA estimation based on MUSIC algorithm," 2014.
[2] P. Gupta, V. K. Verma and V. Senapati, "Angle of arrival detection by ESPRIT method," 2017 International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 2017, pp. 1143-1147.
[3] Z. Dai, and Y. Du, "DOA Estimation Based on Improved MUSIC Algorithm," 2009.
[4] T. B. Lavate, V. K. Kokate and A. M. Sapkal, "Performance Analysis of MUSIC and ESPRIT DOA Estimation Algorithms for Adaptive Array Smart Antenna in Mobile Communication," 2010 Second International Conference on Computer and Network Technology, Bangkok, Thailand, 2010, pp. 308-311.
[5] H. Pan, R. Scheibler, E. Bezzam, I. Dokmanic and M. Vetterli, "FRIDA: FRI-BASED DOA ESTIMATION FOR ARBITRARY ARRAY LAYOUTS," IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017.
[6] D. Diaz-Guerra and J. R. Beltran, "Direction of Arrival Estimation with Microphone Arrays Using SRP-PHAT and Neural Networks," 2018 IEEE 10th Sensor Array and Multichannel Signal Processing Workshop (SAM), Sheffield, UK, 2018, pp. 617-621.
[7] M. B. Çöteli, O. Olgun and H. Hacıhabiboğlu, "Multiple Sound Source Localisation with Steered Response Power Density and Hierarchical Grid Refinement," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 26, no. 11, pp. 2215-2229, Nov. 2018.

[8] Yeo-Sun Yoon, L. M. Kaplan and J. H. McClellan, "TOPS: new DOA estimator for wideband signals," in IEEE Transactions on Signal Processing, vol. 54, no. 6, pp. 1977-1989, June 2006.
[9] H. Zhang, G. Bi, Y. Cai, S. Gulam Razul and C. Meng Samson See, "DOA estimation of closely-spaced and spectrally-overlapped sources using a STFT-based MUSIC algorithm," Digital Signal Processing 52 (2016):25-34.
[10] J. P. Merkofer, G. Revach, N. Shlezinger, R. Routtenberg and R. J. G. van Sloun, "DA-MUSIC: Data-Driven DoA Estimation via Deep Augmented MUSIC Algorithm," 2023.
[11] Yimin Zhang and Weifeng Ma and Amin, M.G, "Subspace analysis of spatial time-frequency distribution matrices." in IEEE Transactions on Signal Processing, vol. 49, no. 4, pp. 747-759, April 2001.
[12] Julian P. Merkofer, Guy Revach, Nir Shlezinger, Tirza Routtenberg, Ruud J. G. van Sloun, DA-MUSIC: Data-Driven DoA Estimation via Deep Augmented MUSIC Algorithm , 2023