



POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Echolocation for the RO-BAT Platform: Biomimetic Autonomous Mobile Robot Navigation Exploiting Ultrasonic Echoes

TESI DI LAUREA MAGISTRALE IN
MUSIC AND ACOUSTIC ENGINEERING
INGEGNERIA ACUSTICA E MUSICALE

Author: **Gabriele Baroli**

Student ID: 221120

Advisor: Prof. Fabio Antonacci

Co-advisors: Dr. Thejasvi Beleyur, Prof. Dr.-Ing. Heiko Hamann,
Dr. Andreagiovanni Reina

Academic Year: 2024-25

Abstract

Large groups of bats can achieve coordinated collective motion relying only on echolocation while experiencing the so-called Cocktail Party Problem: in order to echolocate they need to hear the echoes of their own call, but in groups the echoes overlap with conspecifics' emissions and the echoes thereof, resulting in masking and jamming effects. It is still unclear how they manage to maneuver in such conditions. To shed light on coordination mechanisms in crowded environments, biologists and engineers teamed up to implement the Ro-BAT platform as a robotic model of swarms of active sensing agents. Previous work realised a first prototype of a Ro-BAT capable of passive sound source localisation.

In this thesis I propose a second version of the Ro-BAT capable of active echolocation, equipped with an electrostatic ultrasonic transducer for emission, and an array of Micro-Electro-Mechanical System (MEMS) microphones specifically designed for ultrasound recording. I also implemented a signal processing pipeline for handling the incoming echoes of the emitted call to establish the relative position of nearby obstacles in terms of range and angle. For the angle, I tested three different Direction of Arrival (DOA) estimation algorithms, being them Delay and Sum (DAS), the Capon method, and Multiple Signal Classification (MUSIC), in static conditions. I then chose the most efficient one, i.e. DAS, to be included in the final pipeline for obstacle avoidance. The robot was then tested in lab experiments regarding its ability to autonomously navigate an arena with reflective obstacles only through echolocation.

The tests' results show that my bioinspired robot, mainly assembled with consumer-grade hardware, is capable of locating obstacles with a good level of accuracy thanks to the echoes of the robot's emission and steer its path accordingly to avoid collisions. The proposed solution can therefore be considered a starting point to be scaled to a full swarm of active sensing agents. This robotic model could then be used to perform control studies to deepen the understanding of collective behaviour mediated by acoustics in swarms of biological and robotic agents.

Keywords: ultrasound, echolocation, Direction of Arrival estimation, swarm robotics,

collective behaviour.

Abstract in lingua italiana

Gruppi numerosi di pipistrelli sono in grado di muoversi in modo coordinato sfruttando solamente l'ecolocalizzazione nonostante siano sottoposti al Problema del Cocktail Party: per ecolocalizzare devono necessariamente percepire gli echi del proprio richiamo, ma in gruppo questi sono spesso sovrapposti a richiami di altri pipistrelli, e agli echi derivanti. Questi segnali possono mascherare o disturbare la ricezione dei propri echi. Non è ancora stato spiegato del tutto come i pipistrelli riescano a destreggiarsi in condizioni simili. Per approfondire la ricerca su meccanismi di coordinazione in ambienti affollati, biologi ed esperti di robotica riuniti in un team hanno progettato la piattaforma Ro-BAT come un modello robotico di uno sciame di agenti che percepiscono in maniera attiva. Lavori precedenti si sono concentrati sulla realizzazione di un primo prototipo di Ro-BAT in grado di localizzare passivamente sorgenti sonore.

In questa tesi io presento una seconda versione del Ro-BAT in grado di ecolocalizzare, equipaggiato con un trasduttore elettrostatico per l'emissione di ultrasuoni e con una schiera di microfoni MEMS specificamente ideati per la registrazione di ultrasuoni. Ho anche implementato una catena di processamento dei segnali che permette al robot di stabilire la posizione relativa di ostacoli nei dintorni, in termini di distanza e angolo, tramite gli echi percepiti. Ho testato tre diversi algoritmi per la stima della Direzione di Arrivo in condizioni statiche, i quali sono Delay and Sum (DAS), il metodo di Capon, e Multiple Signal Classification (MUSIC). Il più efficiente dei tre, DAS, è stato integrato nel comportamento di evitamento degli ostacoli impiegato. Il robot è stato poi messo alla prova in esperimenti di laboratorio, testando la sua capacità di navigare autonomamente un'arena con ostacoli riflettenti solamente sfruttando l'ecolocalizzazione.

I risultati ottenuti nei test indicano che il mio robot bioispirato, assemblato in gran parte con materiali di qualità per il consumatore, è capace di localizzare ostacoli con un buon grado di accuratezza tramite gli echi della sua emissione. La soluzione proposta può quindi essere considerata come un punto di partenza per scalare verso un intero sciame di agenti che percepiscono in maniera attiva. Il risultante modello robotico può quindi essere usato per studi di controllo per approfondire la conoscenza del comportamento collettivo mediato tramite l'acustica in sciami di agenti biologici e robotici.

Parole chiave: ultrasuoni, ecolocalizzazione, stima della Direzione d'Arrivo, robotica degli sciame, comportamento collettivo.

Contents

Abstract	i
Abstract in lingua italiana	iii
Contents	v
1 Introduction	1
1.1 Aim	2
1.2 Hosting Institution	2
1.3 Original Contributions	2
1.4 Thesis Outline	3
2 Background and State of the Art	5
2.1 Background	5
2.1.1 Ultrasound Properties and Echolocation	5
2.1.2 Microphone Arrays	6
2.1.3 Signal Model	7
2.1.4 Spatial Aliasing and Effective Array Aperture	9
2.1.5 Direction of Arrival Estimation Problem Formulation	10
2.2 State of the Art	12
2.2.1 Bio-Inspired Acoustic Sensors	13
2.2.2 Autonomous Echolocating Robots	16
2.2.3 Summary and Conclusions	18
3 Signal Processing Methods	19
3.1 Signal Processing Pipeline	19
3.1.1 Call Emission and Array Recording	20
3.1.2 Filtering and Ranging	21
3.1.3 Direction of Arrival Estimation Algorithms	24

3.1.4	DOA Algorithms Performance Comparison	26
3.2	Final Remarks on the Processing Pipeline	31
4	Robot Design and Implementation	33
4.1	Hardware	33
4.1.1	Robot Base Configuration	33
4.1.2	Audio Equipment	35
4.2	Acoustic Characterisation of the System	41
4.2.1	Transducer Frequency Response	42
4.2.2	Transducer Directivity	43
4.2.3	Microphones Frequency Response	45
4.2.4	Microphones Directivity	46
4.2.5	Conclusions	47
4.3	Software	47
4.3.1	Latency Rating of the System	50
4.3.2	Obstacle Avoidance Behaviour	50
5	Tests and Results	53
5.1	Experimental Setup	53
5.2	Data Extraction and Post Processing	56
5.3	Quantitative Results	58
5.3.1	Echolocation Error	58
5.3.2	Robot Trajectory Analysis	65
5.4	Future Directions	67
6	Conclusions and Future Developments	69
	Bibliography	71
	List of Figures	79
	List of Tables	81
	List of Symbols	83
	List of Acronyms	85

1 | Introduction

A fascinating and still disputed topic in biology is how large groups of echolocating bats can coordinate their collective movement while emerging from their roost location for foraging, sometimes, in groups comprising up to a few millions individuals [31]. Bats are active sensing agents, meaning that in order to extract information on their surroundings they probe the environment with short and powerful ultrasonic emissions and listen back to the echoes generated by reflectors around them. Therefore, they are expected to be very susceptible to signals which may mask or jam the echoes they use for echolocation.

This condition is a widely studied phenomenon in acoustics, and it is labelled as "the cocktail party problem" [4]. The intensity at which bats are expected to experience this disruptive phenomenon on their echolocation process has led researchers in the field to elevate this condition to the "cocktail party nightmare" [38]. Despite this, they still proceed in coordinated group flights on a daily basis, with few to no collisions reported.

Theoretical modelling has individuated bottlenecks in the sensory abilities of bats and other echolocating agents [5]: their ability to locate other individuals decreases for increasing group sizes. As a matter of fact, a robotic model of a group of active sensing agents can be a helpful tool to validate the theoretical studies, while also giving insights into real world mechanisms of which the full comprehension is still lacking. The Ro-BAT platform has been ideated with this twofold purpose in mind. On the one hand, modelling active sensing agents with robots lets researchers perform control studies on numerous parameters which could influence the efficacy of active sensing in physical swarms, as opposed to simulations. In simulated experiments, simplifying hypothesis are often introduced, with the risk of not fully encapsulating the complexities of the real world in their full extent. The data extracted from such work could then be used to explain biological mechanisms happening in swarms of which we do not have a full understanding yet. On the other hand, synthesising bioinspired behaviours for robots could lead to the development of new technologies, such as a cost-effective on-board localisation system for a coordinated collective motion mediated by acoustics.

1.1. Aim

After the successful implementation of a small robot capable of passive sound source localisation [9], the following step was represented by extending the capabilities of the very same robotic platform by equipping it with the necessary tools to perform active echolocation. Therefore the aim of this thesis is the development of ultrasound emission and reception devoted hardware, as well as the implementation of a signal processing pipeline which emulates the echolocation process performed by bats. The developed technology should then be deployed on the compact and flexible robotic platform already tested previously.

1.2. Hosting Institution

During my thesis work, I was hosted at the Centre for the Advanced Study of Collective Behaviour (CASCBC), an interdisciplinary excellence research cluster based at the University of Konstanz that includes biologists, psychologists and roboticists with the common goal of pushing forward the study of the behaviour of individuals reunited in collectives of various nature, being them animal groups, human crowds, or decentralised robot swarms. I was co-supervised by Dr. Andreagiovanni Reina, leader of the Group Intelligence and Organisation research group, Dr. Thejasvi Beleyur, Principal Investigator of the newly founded Active Sensing Collectives research group and ideator of the Ro-BAT Project, and Prof. Dr.-Ing. Heiko Hamann, leader of the Cyber-Physical Systems research group. Dr. Beleyur provided the necessary competences in bioacoustics and animal behaviour, while both Dr. Reina and Prof. Hamann contributed to the thesis work with their robotics insights.

A significant contribution to the thesis work was also added by the Electronics Service Centre of the University Workshop: thanks to them it was possible to design and fabricate the needed electronics components in a relatively short span of time. Their support was invaluable for the good results of the work presented in these pages.

1.3. Original Contributions

The content of this thesis collocates itself in the greater landscape of the Ro-BAT Project at an initial step, in which the robotic model of active sensing agents is designed and prototyped. Pushing forward what was established in the previous work [9], I designed a microphone array based on Micro-Electro-Mechanical Systems technology, specifically

intended for ultrasound recording. I also selected and tested the necessary hardware components to connect the electrostatic transducer to the audio interface already in use.

On the processing side, I developed the capabilities of the robot in a modular fashion. I coded the ranging mechanism, then the angle finding mechanism, and ultimately I combined the two together in a cohesive script for object localisation based on the echoes of the emitted call. For the angle, I selected and adapted to the system three well established methods for Direction of Arrival (DOA) estimation: Delay and Sum (DAS), Capon, and Multiple Signal Classification (MUSIC). I tested their performance in static conditions, and chose the most efficient one to be included in the final processing pipeline emulating echolocation.

Finally, I coded an obstacle avoidance behaviour for the robot that I tested in lab experiments. I organised an assessment of the performance of the robot with an autonomous navigation of an arena with reflecting obstacles, and extracted and analysed the data gathered in such experiments.

The developed robotic platform was displayed as a representative example of the bio-inspired robotics research carried out at the CASCB in numerous occasions to visiting researchers and institutions. It was also showcased with live demos at a PhD gathering at the University of Konstanz, and also at the annual spring retreat organised by the CASCB, with a general good reception from experts in the field of robotics and behavioural biology.

1.4. Thesis Outline

The rest of thesis is structured in the following chapters.

Chapter 2 introduces the theoretical background needed for the comprehension of the presented work, both in terms of problem statement and necessary tools and technologies which were employed in the development of the robot. A state of the art on the matter of bioinspired sonar sensors for robotics and autonomous echolocating robots is also presented.

Chapter 3 deals with the signal processing pipeline that was ultimately developed for accomplishing the task given at the start of the thesis. The methods employed are described in details, and an evaluation of the algorithms developed is given to the end of justifying the choices taken at the moment of the effective deployment on the robot for autonomous navigation.

Chapter 4 describes how the robot was implemented, giving a detailed description of the

hardware pieces used and the final configuration reached. The Chapter contains also the characterisation procedures and results that were carried out on the final robot, in order to assess the performance of the various components that constitute the prototype built.

Chapter 5 presents the testing modalities and the obtained results after the tests conducted on the robot regarding its capability of navigating autonomously a controlled environment while avoiding obstacles encountered on its route.

Chapter 6, finally, reports the conclusions that can be drawn from this work, and the future developments of the project this thesis is a part of.

2 | Background and State of the Art

This Chapter is structured in two main sections. Section 2.1 will introduce the fundamental concepts which are necessary for the comprehension of the rest of the thesis. Section 2.2 will give instead an overview of works in the field of bioinspired acoustic sensors and robotic models of echolocating bats.

2.1. Background

The implementation of a robotic model of an echolocating bat implies the handling of two distinct phases of the process: the emission of a loud, wide-band ultrasonic call, and the recording and processing of the echoes generated by the presence of objects around the agent. For the emission stage, advancements in transducer technology for robotics has lead to the introduction of audio sources which are able to handle these strict operating conditions. For the reception stage, instead, two different strategies have been explored in literature: microphone arrays, and binaural bat-inspired microphones. A comprehensive analysis of the literature about both systems will be given in section 2.2. Microphone arrays were chosen for the development of this thesis, following the route traced in [9].

2.1.1. Ultrasound Properties and Echolocation

Ultrasound is defined as a pressure wave with frequencies which exceed the upper human hearing limit. Despite this being different from individual to individual, a commonly accepted definition states the beginning of the ultrasonic spectrum at around 20 kHz [26], which corresponds to a maximum wavelength of about 17 mm. It is known that planar pressure wavefronts undergo specular reflection when their wavelength is much smaller than the effective dimensions of the objects they hit, and so in most practical cases ultrasonic propagation is treated with a geometrical approach [26].

Echolocation in its simplest form is then suited for objects which once hit by an ultrasonic

wavefront cause the immediate return of an echo back to the position of the emitter: this includes flat and smooth surfaces perpendicular to the direction of propagation of the wavefront, objects made out of convex surfaces, and objects which cause the scattering of the incident wave. To this merit, it has been proven that bats have a hard time around smooth and flat artificial surfaces, both horizontal and vertical. In [15] for example is shown that bats often try to drink from sheets of metal on the ground because they mistake them for water bodies. In [16] instead is studied how flat vertical surfaces act as a sensory trap for echolocating bats: such surfaces reflect the emitted echolocation calls away from the listener, causing the illusion of open flyways.

In order to develop a solid groundwork which might be expanded in the future, the robot I built deals with the simple case of echolocating objects which cause an immediate return of the echo to the listener, without considering more complicated scenarios in which the emitted call bounces more than once on nearby objects before getting back to the listener.

2.1.2. Microphone Arrays

Multiple microphones distributed in space which record the same acoustic field constitute an array of sensors. The advantage of having multiple microphones recording the same scene is that spatial information can be extracted from the array recordings, such as the direction from which a source arrives to the microphones. Conversely, it is also possible to focus the array directivity pattern in a certain direction to enhance the reception of the signals coming from that angle, while at the same time attenuating the unwanted interferers from other directions. The first operation is called Direction of Arrival (DOA) Estimation, while the second takes the name Beamforming.

Microphone arrays are usually classified based on their geometry. The sensors can be arranged in a line, which would constitute a linear array, or in more complex shapes which can also be two- or three-dimensional. In the remainder of the Chapter a linear geometry is considered, and in particular a uniform distribution of sensors in a line, which is commonly referred to as a Uniform Linear Array (ULA). Fig. 2.1 reports the situation in which a source located in the far field emits a signal which impinges on a ULA composed of M sensors. The planar wavefronts are perpendicular to the direction of propagation, and hit the microphones in the array each with a specific delay with respect to a microphone taken as a reference. The reference microphone is usually the first one on the left.

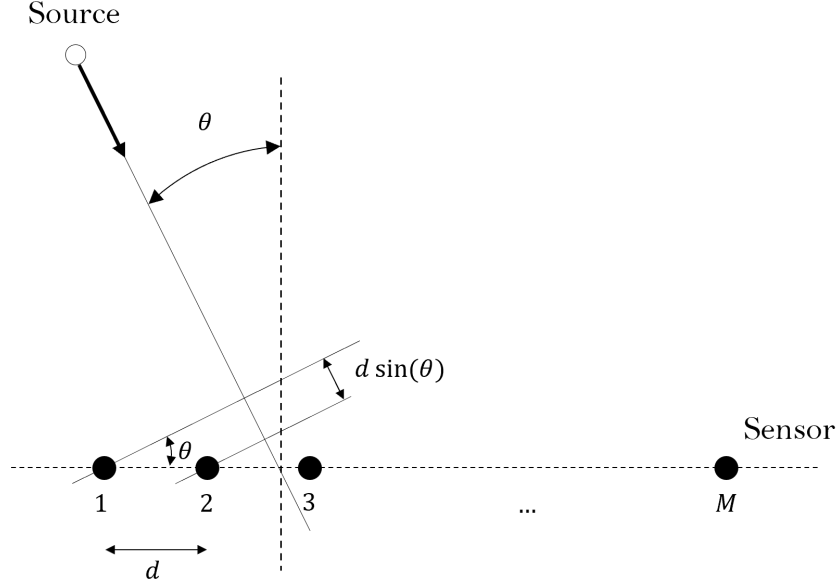


Figure 2.1: Source impinging from the direction θ on a Uniform Linear Array. The angle is measured from the axis of the array, and the microphone taken as a reference is the first one on the left. If the source is in the far-field, the wavefronts are planar, and the time delays between the arrival of the signal to each sensor in the array depend only on the inter-element distance and on the DOA.

2.1.3. Signal Model

Considering an array of M sensors and a single source in the far-field impinging on the array from a certain direction θ , the signal recorded by the k -th microphone in the array can be modelled as:

$$y_k(n) = h_k(n) * s(n - \tau_k) + e_k(n) \quad (2.1)$$

where $s(n)$ is the source signal measured at the source, $h_k(n)$ is the impulse response of the microphone, τ_k is the propagation delay from the position of the source to the sensor, and $e_k(n)$ is the additive noise present at the sensor. For a Uniform Linear Array, the relation between the propagation delay at the k -th channel and the direction of arrival of the source is the following:

$$\tau_k = (k - 1) \frac{d \sin(\theta)}{c} \quad \text{for } \theta \in [-90^\circ, 90^\circ] \quad \text{and } k = 1, \dots, M \quad (2.2)$$

where d is the inter-element distance, θ is the source DOA, and c is the sound speed in air, considered equal to 343 m s^{-1} . Now, the equation that models the array signal can be greatly simplified by assuming a narrow-band source, which is not always the case,

especially for ultrasonic sources. Nonetheless, a wide-band source can be filtered with a filter bank to be separated in multiple narrow-band components, which can be processed individually. This justifies the use of a narrow-band model also for signals which are wide-band in nature.

A narrow-band signal with known carrier frequency ω_c takes the form:

$$s(n) = \alpha(n) \cos[\omega_c n + \phi(n)] \quad (2.3)$$

where $\alpha(n)$ and $\phi(n)$ are respectively the time-varying instantaneous amplitude and phase. The signal is analysed in the time-frequency domain by means of a Short Time Fourier Transform (STFT) employing a rectangular window (e.g. no windowing on the extracted frames) and a unitary hop-size, a method also known as sliding Discrete Time Fourier Transform (DTFT). The extracted frames are indicated as $s_t(n)$, and their DTFT is indicated as $S_t(\omega)$. If the attention is focused on the carrier frequency ω_c , the effect that the STFT has on a narrow-band signal is equivalent to a demodulation of a bandpass signal followed by a low-pass filtering, if $\alpha(n)$ and $\phi(n)$ are slowly-varying in the extracted frame. The result is a baseband signal of the form:

$$S_t(\omega_c) = \alpha(t) e^{j\phi(t)} \quad (2.4)$$

which is only function of time. The notation can be simplified by considering $s(t) \triangleq S_t(\omega_c)$, eliminating the frequency dependency.

Working in the STFT domain, the complete array model for a single source can be derived starting from:

$$y_k(t) = H_k(\omega_c) s(t) e^{-j\omega_c \tau_k} + e_k(t) \quad (2.5)$$

where $H_k(\omega_c)$ is the frequency response of the k -th sensor evaluated for $\omega = \omega_c$. This equation is the STFT of Eq. 2.1. Using a vectorial representation, the array signal for a single source can be written as:

$$\mathbf{y}(t) = \mathbf{a}(\theta) s(t) + \mathbf{e}(t) \quad (2.6)$$

where $\mathbf{y}(t) = [y_1(t), \dots, y_M(t)]^T$, $\mathbf{a}(\theta) = [H_1(\omega_c) e^{-j\omega_c \tau_1}, \dots, H_M(\omega_c) e^{-j\omega_c \tau_M}]^T$ and $\mathbf{e}(t) = [e_1(t), \dots, e_M(t)]^T$.

Assuming ideal linear acoustic propagation, Eq. 2.6 can be extended to the case of N

sources impinging on the array with N different DOAs as:

$$\mathbf{y}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{e}(t) \quad (2.7)$$

where $\mathbf{A} = [\mathbf{a}(\theta_1), \dots, \mathbf{a}(\theta_N)]$ is the $M \times N$ propagation matrix, and $\mathbf{s}(t) = [s_1(t), \dots, s_N(t)]^T$ is the source vector.

The propagation vectors can be simplified by introducing the notion of spatial frequency: this quantity, defined as

$$\omega_s \triangleq \omega_c \frac{d \sin(\theta)}{c} \quad (2.8)$$

is directly related to the DOA of the source, and if substituted into the propagation vector formulation leads to

$$\mathbf{a}(\theta) = [1, e^{-j\omega_s}, \dots, e^{-j(M-1)\omega_s}]^T \quad (2.9)$$

where the sensors' frequency responses were also assumed to be $H_1(\omega_c) = H_2(\omega_c) = \dots = H_M(\omega_c) = 1$. From the propagation vector expression in 2.9, it is clear how the array samples a complex sinusoid of frequency ω_s , with a unitary sampling frequency.

2.1.4. Spatial Aliasing and Effective Array Aperture

Since the array is effectively sampling a sinusoid with a given frequency, the Nyquist-Shannon theorem must hold also in this case to avoid the aliasing phenomenon, which introduces ambiguity in the representation of a certain DOA due to the appearance of grating lobes in the array directivity pattern.

The condition to be satisfied is:

$$|\omega_s| \leq \pi \quad (2.10)$$

Substituting to the spatial frequency its definition, and introducing the wavelength $\lambda = \frac{2\pi c}{\omega_c}$ the condition becomes:

$$|2\pi \frac{d \sin(\theta)}{\lambda}| \leq \pi \quad (2.11)$$

The anti-aliasing condition can be conveniently expressed as a function of the inter-element distance d :

$$d \leq \frac{\lambda}{2} \quad (2.12)$$

This condition establishes the minimum wavelength that won't be aliased by an array with a sensor distance d . The maximum frequency which won't cause spatial aliasing can therefore be written as:

$$f_{max} = \frac{c}{2d} \quad (2.13)$$

For larger frequencies it is then necessary to use densely packed microphone arrays, to make sure the spatial aliasing phenomenon is avoided.

Microphone arrays exhibit also an empirical lower frequency bound, given by the relationship between the wavelengths of the recorded signal and their aperture. Their functioning is based on capturing a phase difference between signals impinging on each sensor in the array. If the wavelength of an incoming signal is comparable to the array aperture, given by:

$$L = M \cdot (d - 1) \quad (2.14)$$

the phase difference between each channel becomes negligible since the wavefront vary too slowly. A good rule of thumb for an effective microphone array aperture states that the aperture should be at least twice the maximum wavelength recorded. This can be translated to a condition on the frequency of the recorded sounds as:

$$f_{min} = \frac{c}{2L} \quad (2.15)$$

The array designed for the active sensing Ro-BAT is composed of eight microphones with an inter-element distance of 2.7 mm. The two frequency bounds are then:

$$\begin{cases} f_{min} = 9.1 \text{ kHz} \\ f_{max} = 63.5 \text{ kHz} \end{cases}$$

2.1.5. Direction of Arrival Estimation Problem Formulation

As stated previously, the spatial frequency of a signal is directly related to its Direction of Arrival (DOA). The problem of DOA estimation can then be formulated as a spatial spectral estimation of the impinging signals on the microphone array. Two popular methods for accomplishing this task are spatial filtering, and subspace based methods.

Spatial Filtering

One possible approach to DOA estimation is constituted by spatial filtering. This strategy consists in designing a linear filter which will combine the array signals in such a way that constructive interference will enhance signals arriving from directions of interest, and destructive interference will attenuate signals coming from unwanted directions. Delay and Sum [12] and the Capon method [7] are classical examples of such filters. In general, for an array of M elements the filter will be composed by M complex weights $\mathbf{h} = [h_1, \dots, h_M]^H$, where the superscript H indicates the hermitian transpose. The filtered

signal takes the form:

$$y_F(t) = \sum_{k=1}^M h_k \cdot y_k(t) = \mathbf{h}^H \mathbf{y}(t) \quad (2.16)$$

The power of the filtered signal is then:

$$E \{ |y_F(t)|^2 \} = \mathbf{h}^H \mathbf{R} \mathbf{h} \quad (2.17)$$

where $\mathbf{R}$ is the covariance matrix of the array data, defined as:

$$\mathbf{R} = E \{ \mathbf{y}(t) \mathbf{y}^H(t) \} \quad (2.18)$$

The operator E indicates the statistical expectation. Since the array signal is finite in time, in most practical cases the covariance matrix is swapped for its sample estimate, defined as:

$$\hat{\mathbf{R}} = \frac{1}{K} \sum_{t=1}^K \mathbf{y}(t) \mathbf{y}^H(t) \quad (2.19)$$

The power of the filtered signal constitutes a good indication of the energy coming from a certain direction θ , so by comparing the energy coming from all the directions it is possible to compute a spatial spectrum, whose peaks correspond to directions from which signals impinge on the array.

Subspace Based Methods

An alternative to spatial filtering for DOA estimation is represented by methods which separate the covariance matrix of the array data into a signal subspace, which is related to the signals impinging on the array, and in a noise subspace, related to the noise introduced by the sensors. These methods are based on a series of assumptions, which are:

- Exactly N sources impinge on the array from N distinct directions, thus leading to N distinct spatial frequencies. N must be also lower than M , which is the number of sensors in the array.
- The covariance matrix of the source signals $\mathbf{R}_s = E \{ \mathbf{s}(t) \mathbf{s}^H(t) \}$ is full-rank.
- The sensor noise is spatially white, with independent and identically distributed components having identical variance. The covariance matrix of the error signals is then $E \{ \mathbf{e}(t) \mathbf{e}^H(t) \} = \sigma^2 \mathbf{I}_M$. The noise is also uncorrelated to the source signals.

Under these assumptions, the covariance matrix of the array data can be written as the sum of two components, one related to the source signals and the other to the noise, as:

$$\mathbf{R} = \mathbf{A}\mathbf{R}_s\mathbf{A}^H + \sigma^2\mathbf{I}_M \quad (2.20)$$

By performing an eigenvalue decomposition and sorting the eigenvalues in ascending order, it is possible to express the covariance matrix of the array data as:

$$\mathbf{R} = [\mathbf{U}, \mathbf{V}] \mathbf{\Lambda}_y [\mathbf{U}^H, \mathbf{V}^H]^T \quad (2.21)$$

where the eigenvectors in $\mathbf{U}$ are associated to the eigenvalues $\{\lambda_1, \dots, \lambda_N\}$ and the eigenvectors in $\mathbf{V}$ to the eigenvalues $\{\lambda_{N+1}, \dots, \lambda_M\}$. The diagonal matrix $\mathbf{\Lambda}_y$ contains the sorted eigenvalues of $\mathbf{R}$. It is possible to prove that the eigenvectors in $\mathbf{U}$ are associated to the source signals, therefore spanning the signal subspace, and the eigenvectors in $\mathbf{V}$ are associated to the sensors' noise, giving origin to the noise subspace of the covariance matrix. Each of the two subspaces found possesses properties which can be leveraged to estimate the DOAs of the sources looked after. MUSIC [43] exploits the orthogonality of the noise subspace to the propagation vectors corresponding to DOAs of actual sources, while an example of a method that exploits properties of the signal subspace is constituted by ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques) [41].

The matrices $\mathbf{U}$ and $\mathbf{V}$ are obtainable if and only if the conditions reported above apply, and the sources signal power is higher than the noise power.

2.2. State of the Art

In recent years the rapid development of mobile robotics has lead to a vast exploration of sensing strategies with which a robot can acquire information on its surroundings to then plan its movement strategy. Among these solutions a vast majority is focused on cameras or light-based sensors, but also sound and in particular ultrasound emission and echoes detection is currently being investigated. In this latter case, though, almost all of the research made is dedicated to advanced sonar sensors for robotics and algorithms for Simultaneous Localisation and Mapping (SLAM) exploiting acoustic signals inspired by the echolocation capabilities of bats to various degrees. These works are mainly focused on strategies that individual agents can exploit, and it is not clear if the proposed solutions can scale to groups of agents. This is why a cheap and scalable platform such as the Ro-BAT was designed, with the goal of tackling the cocktail party problem in swarms of active sensing agents.

Section 2.2.1 will give an overview of bioinspired sensors for localisation and navigation based on acoustics. Most of the works presented in this Section come from the Cosys-Lab at the University of Antwerp [39], and constitutes a great bulk of work on bat-inspired sensory solutions for robotics.

In Section 2.2.2, instead, other works which present a full-fledged autonomous echolocating robot are reviewed, presenting the novelty and advancements that they introduced, and also underlying their strengths and weaknesses with respect to scalability.

2.2.1. Bio-Inspired Acoustic Sensors

Sonar sensing is surely not a novel area of research in engineering, but recent advancements in behavioural studies on bats and a refinement of the sonar technology involved have sparked new interest for what a bioinspired sensor could bring to the table. In mobile robotics, on the one side, new sensing strategies are explored for autonomous navigation in complicated environments. On the other side, robotic models could help in improving the knowledge on biological mechanisms which are not fully understood yet.

In [47] two different types of audio receivers for a bioinspired sonar sensor are evaluated: binaural microphones with 3D-printed replicas of bats' external ears are compared against microphone arrays in combination with digital beamforming. The comparison was based on an objective information theoretic performance measure that quantified their localisation capabilities in the presence of realistic noise. The conclusions drawn in the paper are that the two systems are more or less equivalent on the localisation performance side, but a possible advantage in the array solution, despite the increased complexity in managing more than two microphones, consists in the possibility to digitally steer the directivity pattern of the sensor in desired directions without resorting to mechanical modifications of the sensors, which would be needed in the case of two microphones encased in pinnae replicas.

Both binaural microphones and microphone arrays were used to extract information then fed to a SLAM model based on the mammalian hippocampus in subsequent works. In [45], a biomimetic sonar sensor composed of an electrostatic ultrasound transducer and two omnidirectional microphones encased in plastic replicas of a bat's pinnae is used as an exteroceptive sensor, while interoceptive sensory input is given by the robot's odometry retrieved by the motor commands. The echoes, which are filtered by an Echolocation Related Transfer Function (ERTF) given by the superimposed effect of the emitter's directivity and the spatial filtering introduced by the artificial outer ears, are then processed to extract a "cochleogram". This time-frequency representation of the recorded echoes

is obtained by modelling the functioning of a bat's cochlea with a bank of gammatone filters. The obtained cochleograms are used as local view templates of the environment, which combined with the robot's odometry constitute the input to the SLAM algorithm. The result is a topological map of the environment. The paper's authors drove a robot equipped with such sensor in an unmodified office environment, and a geometrically consistent topological map was the result of the SLAM algorithm. The authors conclude that cochleograms are in general sufficiently descriptive of a particular local view to be used as a base for localisation and mapping of a large environment, despite still giving rise to ambiguity in some cases. Such a system would work in conditions of scarce visibility which are problematic for cameras or LIDAR sensors.

The other approach to SLAM [46] employs an array-based biomimetic sensor [48] consisting of the same emitter as in the previous work, but on the receiving end a planar random array of 32 condenser microphones is used in conjunction with a Delay and Sum beamformer. The output of this sensor is a so-called "energy scape" representation of the environment, where each point identified by three coordinates (range, azimuth and elevation) represents the energy reflected back to the sensor from that particular location. This sensor was exploited for obtaining both information on the nearby environment, and also odometry data based on the acoustic flow of consecutive snapshots acquired by the device. The sensor was mounted on an electric wheelchair and driven in a typical office environment. A geometrically consistent topological representation of the environment was obtained also for this case, by feeding to the SLAM algorithm the 3D energy scapes as local view templates, and the odometry data extracted from the acoustic flow model as the interoceptive sensory input.

Both sensors architectures were refined in subsequent works: in [27] a biomimetic sonar sensor based on the common big-eared bat (*Micronycteris microtis*) outer ears replicas is used to guide the autonomous navigation of a two-wheeled robot, in conjunction with a subsumption architecture whose layers are:

1. Collision avoidance: the robot rotates on itself until its pathway is free of reflectors. This behaviour is triggered by the presence of obstacles under a certain range threshold.
2. Obstacle avoidance: the robot proceeds on a curved trajectory distancing itself from nearby obstacles, moving in the opposite direction with respect to the highest amount of energy received between the two ears.
3. Straight drive: the robot proceeds forward at constant velocity along a straight line when its pathway is free of obstacles.

The input to the control system is constituted exclusively by data extracted from the echoes received by the two microphones inside the bat ears replicas. The robot was successful in performing an autonomous navigation of a cluttered environment. An interesting modification introduced to the sensor architecture was a baffle around the ultrasonic transducer, which enlarged its radiation pattern. The effect of baffles on ultrasonic transducers was studied in [20], concluding that good-quality 3D-printed baffles can enlarge the narrow directivity pattern of electrostatic transducers to near-omnidirectionality without significant loss in the output power.

In [22] instead, the array-based sensor architecture was refined by swapping the condenser microphones for digital Micro-Electro-Mechanical Systems (MEMS) microphones, reducing costs and need for extra Analog-to-Digital conversion (ADC) equipment, and the hardware implementation is completely embedded, constituting a plug-and-play sensor for robotic applications. As a beamforming technique, the Delay and Sum method was still used. The resulting sensor was then deployed on a mobile robot which was driven in a cluttered corridor to realise a 3D map of the traversed environment. The data obtained from the sensor was confronted with LIDAR measurements for validation. A reduced form factor version of the sensor explained above was realised in [53]. Contrary to the previous sensor model, this device works with narrow-band emissions and the Multiple Signal Classification (MUSIC) algorithm for retrieving the Direction of Arrival (DOA) of the echoes. This device architecture reduces production costs with respect to the previous sensor, but it suffers from the multipath propagation of the echoes, generating "ghost" reflectors. A more in-depth simulation study on such sensor is found in [54], where different DOA estimation algorithms were employed with the cited sensor architecture varying a number of parameters such as the Signal-to-Noise Ratio (SNR) of the received echoes, the number of snapshots taken for the same scenario, the spacing of two closely placed objects and the amount of objects which could be resolved by the sensor at the same time. The authors conclude that the most limiting aspect of sonar sensors is the need for a high dynamic range, since objects placed at sharper angles suffer from harsh attenuations due to both the directivity of the transducer employed, and the reduced aperture of the array. The SNR and the number of snapshots acquired do not present any significant effect on the localisation process, which is desirable since mobile robots need fast updating sensor readings to operate, often in noisy conditions.

One last frontier of research in the sensor architecture field is represented by [28], in which the aperture of an array based sensor is reduced by means of 3D-printed waveguides which effectively realise an array geometry that could otherwise be impossible to reach due to the physical dimensions of the MEMS microphones employed. This approach extends the

frequency range at which ultrasonic microphone arrays can operate before incurring in the spatial aliasing phenomenon, but introduces a significant amount of attenuation in the signals recorded by the microphones, and requires a calibration step before operation for taking into account the phase shifts introduced by the baffles. A notable result of the integration of such baffles to microphone arrays is a directivity pattern resemblant of a bat Head-Related Transfer Function (HRTF), which is an interesting aspect for biological modelling researches.

2.2.2. Autonomous Echolocating Robots

In addition to bioinspired sensors, another branch of research is focused on the realisation of full-fledged robots which can navigate autonomously relying on the information extracted from the environment by probing it with short ultrasonic emissions.

A first example of an autonomous robot whose navigation strategy is based on the echolocation capabilities of bats is represented by [57]. In this work, an autonomous robot was equipped with a minimal biomimetic sonar sensor, consisting of an ultrasonic transducer and two ultrasound sensors. The narrow-band signal emitted in combination with the reduced dimensions of the transducer led to a radiation pattern comparable to the one of a real bat, having a full width of 80-100° at -6 dB. Objects could be located in range by the time of arrival of the echoes, and in direction by the Time Difference of Arrival (TDOA) at the two receivers. The sensor unit was mounted on a servomotor which could steer the unit in the direction of the echoes received, to keep the localised obstacle in the sensor's field of view (FOV). This strategy was borrowed by bats, which were shown to turn their heads to focus specific reflectors previously identified. An Inter-Pulse Interval (IPI) control mechanism based on the nearest obstacle distance was also implemented. When navigating autonomously a rectangular arena with cylindrical obstacles, the head steering mechanism was proven to perform better than a fixed sensor unit aligned with the direction of motion, but the localisation of only the nearest obstacle did not perform satisfyingly, leading to numerous collisions. It was for this reason that the authors also implemented a Multi-Object-Detecting Navigation System (Multi-ODNS). This obstacle localisation method took into account multiple echoes, corresponding to more than one obstacle, and after locating each one a repulsion force vector was computed for each obstacle as well. The direction in which the robot steered was then a consequence of choosing to avoid the obstacle with the highest repulsion force vector. This new approach to obstacle avoidance demonstrated itself to be successful where the Mono-ODNS previously failed. This robot configuration presents a bio-inspired obstacle avoidance behaviour introducing also the possibility of mechanically steering the sonar sensor, which is a natural

behaviour for bats. The emitter circuit is too simple, though, being only capable of emitting 40 kHz short tones. The flexibility to model species of Frequency Modulating (FM) bats is therefore excluded.

Another example of autonomous bio-inspired robot is constituted by [10]. The output of this work is a fully autonomous terrestrial robot which can map the 2D structure of a novel outdoor environment, label the obstacles as plants or non-plants, and then plan a path accordingly to navigate while avoiding obstacles. This all-terrain, rugged robot is equipped with an ultrasonic electrodynamic speaker, and two condenser microphones specialised for ultrasound recording. The sensor head is mounted on a gimbal which can rotate while maintaining stability. The A/D and D/A conversions are handled by a high-end Data Acquisition (DAQ) board. The robot moves forward in steps of 0.5 m, then stops for 30 s to acquire echoes from the environment by emitting three 10 ms sweeps from 100 kHz to 20 kHz in the directions -60° , 0° and 60° with respect to the driving orientation to emulate the directivity of a bat's emission. The resulting echoes are processed to individuate the distance and the angle from nearby obstacles, and the extension of their contour. Audio features are also extracted from the received echoes to classify the reflectors encountered as plants or non-plants thanks to an artificial neural network. The robot was capable of mapping the position of obstacles in a novel environment and their contour with a satisfying degree of accuracy and planning its route accordingly without any recorded collision. The classification accuracy was also above the chance level of 50% of right decisions. This work implements an acoustic based autonomous navigation method which is biologically plausible, even though with evident limitations related to the movement speed of the robot and its emission strategy. Furthermore, the audio feature based classification of the type of obstacles encountered is an interesting introduction, which could be further exploited in other settings, also in a swarm context. The dimensions, slow reactivity and the realisation cost of a single agent are not suited for swarm robotics, though.

In [30], instead, a robotic model for the avoidance of non-localisable obstacles by bats is presented. This work aimed at validating a minimal set of acoustic cues for reacting to a series of non-localisable echoes, meaning a group of highly overlapping echoes caused by complex reflectors such as the vegetation among which bats fly in nature. The cues taken into consideration are the Interaural Intensity Difference (IID), and the distance from the closest obstacle. Moreover, also two different navigation modalities are investigated together with the primary question, being those the Fixed Head strategy, and gaze steering according to a Delayed Linear Adaptive Law (DLAL). The robotic platform used consists of a differential drive robot mounting a narrow-band ultrasonic transducer and

two microphones encased in stylised plastic replicas of a bat's pinnae. The sensor unit was mounted on a pan-tilt system which could emulate head movements observed in bats. The obstacle avoidance behaviour was tested both in a rectangular arena with complex reflectors which produced highly overlapping echoes, and in a cluttered corridor whose walls lined with egg cartons also reflected the robot's calls in a non localisable fashion. By comparing the two navigation strategies in conjunction with the set of cues chosen to extract spatial information from the environment, the authors conclude that the IID and the distance from the nearest obstacle is enough for achieving autonomous navigation, despite collisions were reported, even though only in the 10% of obstacle interactions. The robot was also able to complete the navigation of the corridor successfully. Contrary to [57], they found that acoustic gaze scanning leads to a worse performance than locking the sensing direction to the motion direction. This is due to the amount of angular information that can be extracted from overlapping echoes: in the case of highly cluttered environments steering the sensor unit does not improve the efficiency of the motion. The results reported in this work offer a solid strategy for a simple obstacle avoidance behaviour in heavily cluttered environments which relies on a minimal but robust set of cues extracted from the echoes received. This strategy is mainly applicable to robots and sensors whose receiving unit is constituted by a left and a right "ear". An application to microphone arrays would require further elaboration.

2.2.3. Summary and Conclusions

In this Section a series of bioinspired sensor architectures and robotic platforms based on acoustics were analysed and reviewed. The solutions presented are enormous sources of inspiration for biomimetic acoustic sensor architectures, but are not thought for swarms of echolocating agents: I instead argue that the presence of similar audio sources to the ones employed would jam the sensors to various degrees, rendering them ineffective in a collective of active sensing robots. They can however constitute a first approximation of the echolocation process that most bats exploit continuously during flight.

Another point of crucial importance is the construction complexity of such solutions: they all require the development of highly specialised hardware pieces and are generally expensive and cumbersome if compared to small robotic platforms which are often employed for swarm robotics, making their deployment not suitable for the context of this thesis.

A cheap and scalable robotic platform that emulates echolocating bats has not yet been realised, and the Ro-BAT project aims at filling this gap.

3 | Signal Processing Methods

This Chapter will describe the full signal processing pipeline devoted to the echolocation capabilities of the Ro-BAT and the conditions in which the robot is intended to operate, together with some results regarding the implementation of Direction of Arrival estimation algorithms. The hardware details and their characterisation is the topic for Chapter 4, here only the necessary information for understanding the rationale behind the methodologies employed is presented.

3.1. Signal Processing Pipeline

The signal processing pipeline that was developed is largely inspired by the work of Steckel *et al.* [48] and further developments of the Cosys-Lab at the University of Antwerp [39]. Their idea of a sonar system based on a single emitter and multiple receivers is at the basis of the content of this thesis. There are a few differences regarding the objective with which the two systems are built, though: while their sensor is thought as a plug-and-play reliable alternative for autonomous robotics navigation, the robot proposed in this thesis is a first attempt to deploy active echolocation on a small mobile platform with the greater objective of studying how a swarm of agents which base their movement strategy on acoustic mechanisms can use active echolocation effectively to achieve a coordinated collective behaviour.

The geometry of the problem posed is assumed to be two-dimensional: the robot moves on a plane, so only two coordinates are necessary to establish the relative position of objects around it. These two coordinates are the distance from the robot to the objects, and the angle at which they are positioned with respect to the robot's longitudinal axis. Assuming, as explained above, that the objects around the robot reflect the emitted calls back to the robot's position, the distance from the robot to the objects can be computed by knowing the time difference between the instant in which the call is emitted, and the instant in which the echoes are received. Regarding the incident angle of the echo, Direction of Arrival estimation algorithms can be employed, after recording the received echoes by means of a microphone array. Among the many different array geometries, a

Uniform Linear Array (ULA) configuration has been selected for its simplicity, since it is fully capable of handling a two-dimensional localisation problem.

The field of view (FOV) of the robot is by necessity its front half-plane, since the audio source is pointed towards its front and the microphone array is linear, which introduces a front-to-back ambiguity solved by only considering one of the two half-planes with respect to the array orientation. The angular axis θ is defined as $\theta \in [-90^\circ, 90^\circ]$, where -90° corresponds to the robot's right hand side, 90° to its left hand side, and 0° is in correspondence of its longitudinal axis.

Another simplification introduced in the system is that the robot will focus only on the first echo received, meaning that at every instant only the objects placed on a semi-circle with the lowest radius with respect to the robot's position are located.

3.1.1. Call Emission and Array Recording

Since the Ro-BAT aims at emulating the echolocation strategies of Frequency Modulating (FM) bats, the type of signal that constitutes the emitted call will be a wide-band chirp, in particular of the linear type. This assures that all the frequencies of interest have the same weight in the emitted signal. An example of a call signal used by the robot can be seen in Fig. 3.1. After an emission, echoes will be generated by obstacles posed around the robot. Both the call and the echoes are recorded by the microphone array, which enables a spatial processing of the recorded audio.

The signal captured by the k -th microphone in the array after the emission of a call and the reception of the subsequent echoes can be modelled as:

$$y_k(n) = h_k(n) * \left(s_c(n) + \sum_{i=1}^L \alpha_i s_c(n - \tau_i - \tau_k) \right) + e_k(n) \quad \text{for } k = 1, \dots, M \quad (3.1)$$

where $h_k(n)$ is the impulse response of the single sensor, $s_c(n)$ is the emitted call, α_i and τ_i are respectively the attenuation coefficients and the propagation delays due to the reflection process, L is the number of echoes recorded, τ_k is the propagation delay due to the position of the sensor in the array, M is the number of sensors in the array and e_k is the noise present in each sensor.

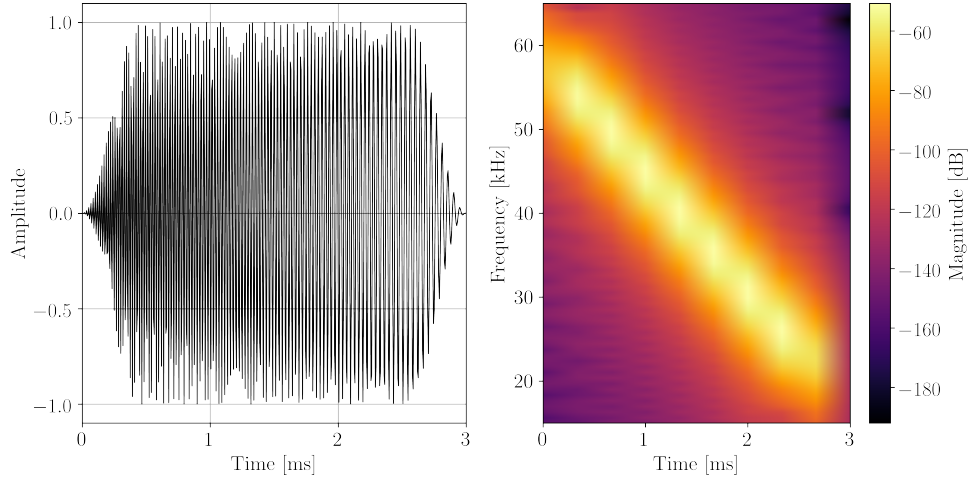


Figure 3.1: Time trend and spectrogram of the emitted signal.

3.1.2. Filtering and Ranging

The first step in the audio data processing is constituted by a matched filter, described by:

$$y_k^{MF}(n) = \mathcal{F}^{-1} \{Y_k(j\omega) \cdot S_c^*(j\omega)\} \quad (3.2)$$

where $Y_k(j\omega)$ is the Discrete Fourier Transform (DFT) of the signal recorded by the i -th microphone, $S_c^*(j\omega)$ is the complex conjugate of the DFT of the call signal, and the $\mathcal{F}^{-1}$ operator denotes the Inverse Discrete Fourier Transform. The output of the filter will be a series of scaled and time-delayed autocorrelation functions of the emitted call, if we consider as negligible the spectral distortions introduced by the emission and reflection process. As a result, the call and the echoes onsets are easily individuated in the cross-correlations, since they are in correspondence of the local maxima of the autocorrelation functions. Other signals that do not match the template of the matched filter will instead be annihilated by the filter.

To better perform a peak-finding operation to reveal the position of the echoes, the envelopes of the cross-correlations are extracted by taking the absolute value of their Hilbert transform. Then, a different strategy is applied to isolate the call and the first echo, respectively.

- **Call detection:** a mean envelope is computed by averaging all the envelopes extracted from the cross-correlations. The call onset is then retrieved by a peak-finding operation executed on the mean envelope, to have a more reliable estimate. The

first index returned by the algorithm represents the time instant wanted, since it is assumed that the call is the first relevant signal appearing in the analysed sequence.

- **Echo detection:** for each envelope a peak-finding operation is computed, starting the search from a tolerance window of samples after the peak representing the call, to avoid considering further oscillations belonging to the call autocorrelation function as reflections caused by very close objects. This introduces a "blind spot" in the robot's field of view. The tolerance window which worked best in terms of avoiding "ghost obstacles" versus locating obstacles as close as possible to the robot corresponds to a radius of 12.5 cm. For each envelope the first index corresponding to a peak is returned, and the smallest among all of them is used as a reference to compute the distance from the nearest object.

Having retrieved the two onsets needed, the range from the nearest object is then computed according to:

$$\Delta d = \frac{1}{2}c\Delta t \quad (3.3)$$

where Δt is the time interval between the two peaks, c is the sound speed in air, considered to be 343 m s^{-1} , and Δd is the estimated distance from the robot to the object. An example of the full filtering and ranging process is shown in Fig. 3.3, while the spectrogram of the same audio recording is reported in Fig. 3.2.

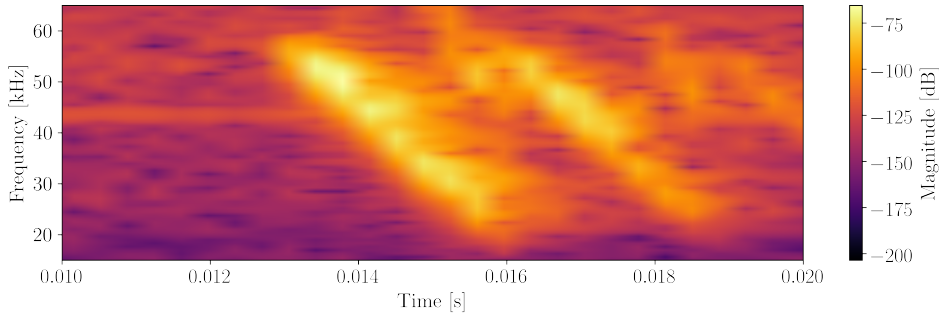
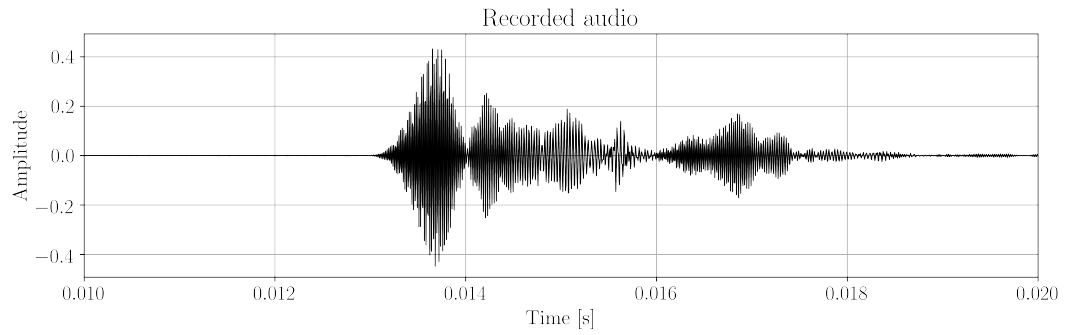
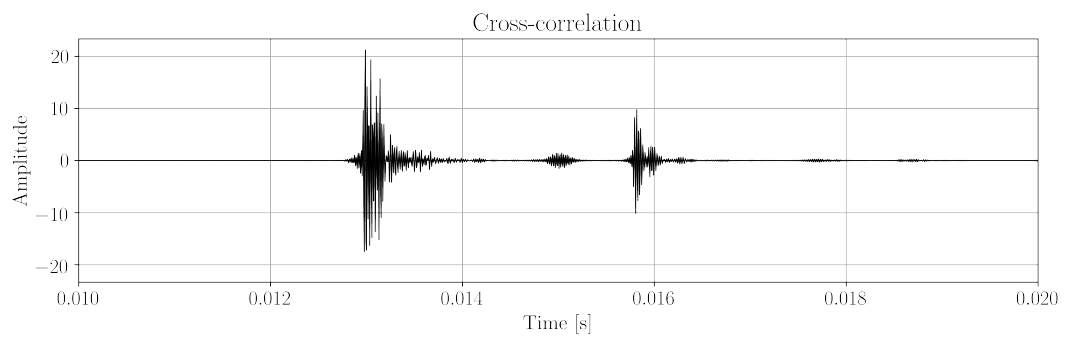


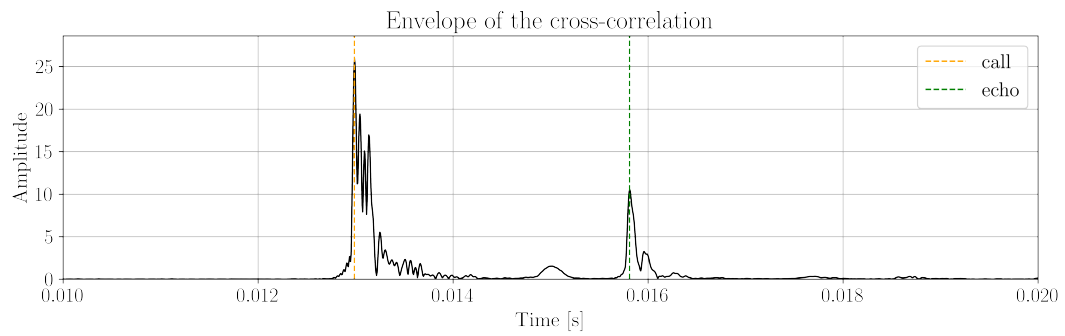
Figure 3.2: Spectrogram of the emitted call and first echo. The audio was extracted from the first channel of the array. The call and the echo are clearly distinguishable in the time-frequency representation of the recorded audio.



(a)



(b)



(c)

Figure 3.3: Example of the ranging process for locating an object placed at 50 cm in front of the robot. The three graphs refer to a single channel in the microphone array. It is easy to see how much the matched filter compresses in time the signals that match the template by comparing (a) and (b). The envelope reported in (c), then, improves the peak-finding algorithm accuracy, which needs to be tailored properly nonetheless to avoid locating the spurious peaks of the call's autocorrelation function as inexistent obstacles.

3.1.3. Direction of Arrival Estimation Algorithms

The following step in the localisation process is the Direction of Arrival estimation. Two of the possible strategies for this task are employing spatial filtering, or parametrizing the covariance structure of the array data to exploit the properties of one of its subspaces. I implemented and tested two algorithms belonging to the spatial filtering class, and a third one belonging to the parametric methods class. One relevant note is that all three of the algorithms implemented are effective on narrow-band signals, while the call emitted by the robot is inherently wide-band. To accommodate this requirement, each algorithm has been extended to a wide-band use case by means of an incoherent average [49, 56]: the input signal is divided into sub-bands which are processed individually. The results obtained by each sub-band are then averaged, and the DOA is computed on the basis of this average. This method is defined as incoherent because a series of power spectra are averaged, without maintaining any phase relationship between sub-bands.

The data fed to these algorithms is constituted of a rectangular window of length 1 ms centred on the cross-correlations portion corresponding to the echo identified in the previous processing step. The advantages of passing the filtered echo with respect to the raw audio data are that the filtered echo is much more isolated in the time history of the signal, reducing the bleeding-in of the emitted call even when the obstacle is very close to the robot, and also it contains the same amount of spectral information in a reduced number of samples, resulting in a faster computation time.

The DOA estimation algorithms implemented are:

- **Delay and Sum (DAS):** this algorithm focuses on passing undistorted the signals coming from a specific direction $\bar{\theta}$, while attenuating the signals coming from all the other directions. The array signal is assumed to be spatially white, meaning that the covariance matrix $\mathbf{R}$ is an identity matrix $\mathbf{I}$ of rank M . The formalisation of the equivalent optimisation problem is written as:

$$\mathbf{h}(\bar{\theta}) = \arg \min_{\mathbf{h}} \mathbf{h}^H \mathbf{h} \quad \text{subject to } \mathbf{h}^H \mathbf{a}(\bar{\theta}) = 1 \quad (3.4)$$

The resulting filter coefficients are defined as:

$$\mathbf{h}(\bar{\theta}) = \frac{\mathbf{a}(\bar{\theta})}{M} \quad (3.5)$$

where $\mathbf{a}(\bar{\theta}) = [1 \quad e^{-j\omega_s} \quad \dots \quad e^{-j(M-1)\omega_s}]^T$ is the propagation vector whose spatial frequency ω_s corresponds to the direction $\bar{\theta}$, and M is the number of microphones

in the array. It is worth noting how the obtained coefficients are scaled complex exponentials, which correspond to pure delays: the output of the filter is the sum of the signals captured by the array, each one of them with an appropriate phase shift to create zones of constructive and destructive interference where it is desired. The name with which this procedure is usually found in literature reflects this very property.

The resulting pseudospectrum, which derives from the power of the filtered signal evaluated at all the angles of interest $\theta \in [-90^\circ, 90^\circ]$ is:

$$p(\theta) = \frac{\mathbf{a}^H(\theta)\hat{\mathbf{R}}\mathbf{a}(\theta)}{M^2} \quad (3.6)$$

where $\hat{\mathbf{R}}$ is the sample estimate of the covariance matrix of the array data. The DOAs of N sources are then estimated as the angles in correspondence of the N highest peaks of the function:

$$p_{avg}(\theta) = \frac{1}{B} \sum_{i=1}^B p_i(\theta) \quad (3.7)$$

where B is the number of processed sub-bands.

- **Capon method:** the second spatial filtering algorithm developed is the so-called Capon method. Contrary to the previous approach, this method puts effort into attenuating the actual signals which impinge on the array from unwanted directions, while passing undistorted the signals coming from the desired direction. This method is then data-dependent, since the corresponding optimisation problem takes the form:

$$\mathbf{h}(\bar{\theta}) = \arg \min_{\mathbf{h}} \mathbf{h}^H \mathbf{R} \mathbf{h} \quad \text{subject to } \mathbf{h}^H \mathbf{a}(\bar{\theta}) = 1 \quad (3.8)$$

The filter coefficients for every sub-bands assume the following form:

$$\mathbf{h}(\bar{\theta}) = \frac{\mathbf{R}^{-1} \mathbf{a}(\bar{\theta})}{\mathbf{a}^H(\bar{\theta}) \mathbf{R}^{-1} \mathbf{a}(\bar{\theta})} \quad (3.9)$$

where $\mathbf{R}$ is the covariance matrix of the array data. The resulting pseudospectrum for this spatial filter is:

$$p(\theta) = \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{R}}^{-1} \mathbf{a}(\theta)} \quad (3.10)$$

where once again the sample estimate of the covariance matrix $\hat{\mathbf{R}}$ is used in place of the actual covariance matrix $\mathbf{R}$.

Also in this case, the pseudospectra obtained from each sub-band elaboration were averaged before seeking the directions of arrival, which are defined as the angles corresponding to the N highest peaks of the averaged pseudospectrum as well.

The trade-off between DAS and Capon is that the Capon method has a better spatial resolution, meaning that it is able to discriminate between sources coming from two very close points in space much better than what DAS can usually do, at the cost of a more complex computation.

- **Multiple Signal Classification (MUSIC):** this algorithm belongs to the signal subspace methods class. The subspace exploited by this method is the one generated by the eigenvectors of the covariance matrix of the array data related to the sensors' noise. It can be proven that such eigenvectors and the propagation vectors associated to N distinct sources are orthogonal. So, once the eigenvector matrix related to noise $\hat{\mathbf{V}}$ is estimated after an eigenvalue decomposition of $\hat{\mathbf{R}}$, the DOAs of N wideband sources can be estimated in correspondence of the N highest peaks of the function:

$$p_{avg}(\theta) = \frac{1}{B} \sum_{i=1}^B p_i(\theta) \quad (3.11)$$

where the pseudospectrum for each sub-band is defined as:

$$p(\theta) = \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{V}} \hat{\mathbf{V}}^H \mathbf{a}(\theta)} \quad (3.12)$$

One important difference between this parametric method and the two spatial filters mentioned before is that the expected number of sources to be detected with MUSIC must be known in advance. This translates well in the case in which only one obstacle at a time is considered, but it would scale poorly when an unknown number of obstacles which is greater than one would need to be treated by the robot.

3.1.4. DOA Algorithms Performance Comparison

Before deploying the full pipeline on the robot while it navigates, I tested and compared the performances of the three algorithms developed in terms of their computation time, spatial resolution, and accuracy for obstacles in various positions around the array. More information on the robotic platform employed are given in Chapter 4, here only the relevant tests results are displayed and explained.

Computation Time

This metric is extremely important for real time use cases of the DOA estimators, since it determines how responsive the system would be to an evolving environment. I ran each algorithm for 20 times on the robot keeping the sampling frequency of the system fixed at 176.4 kHz and feeding the same window of length 1 ms of data to the algorithms. I kept track of the difference in the time stamp between the instant in which the signal is presented to the DOA estimator and the instant in which the pseudospectrum is available for further analysis. The resulting mean computation times are shown in Tab. 3.1. The collected data show that DAS has the highest computation speed among the three, unsurprisingly: the required operations are relatively simple with respect to the other two methods. Capon requires the inversion of a matrix with a large rank, and MUSIC employs an eigenvalue decomposition of the same matrix followed by a sorting of its eigenvalues and corresponding eigenvectors.

DOA estimator	Delay and Sum	Capon method	MUSIC
Computation time [ms]	68	145	162

Table 3.1: Mean execution times for the DOA estimation algorithms implemented

Spatial Resolution

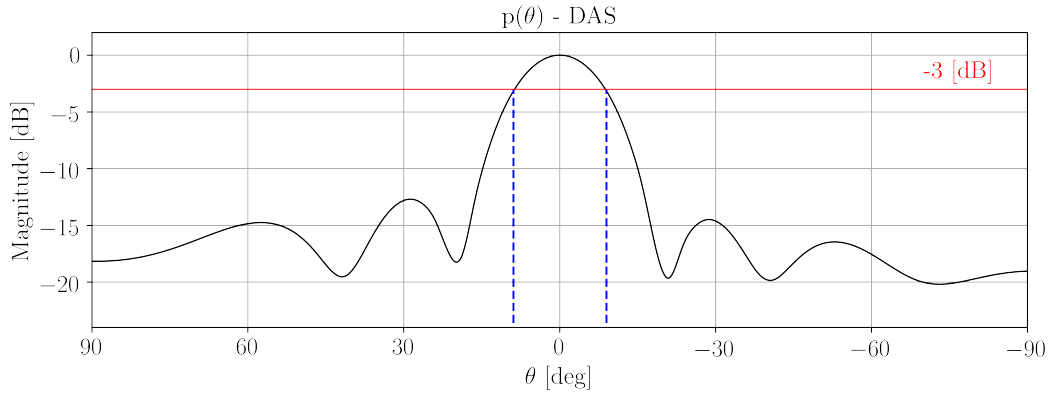
The spatial resolution refers to the minimum angular separation that two sources must have to be identifiable as separate by the DOA estimator. The metric that I chose to give an estimate of the spatial resolution of the algorithms is the Half-Power Beam Width (HPBW), a concept taken from the theory of phased array antennas. The HPBW is defined as the angle between the two points of the main lobe of the pseudospectrum that have a magnitude inferior by 3 dB with respect to its maximum. Empty beer bottles of the NRW (Nordrhein-Westfalen) model (height: 260 mm, base diameter: 64 mm) were used as reflectors, due to their availability and ideal geometrical and material properties in relation to the robot's emission. One bottle was placed at 50 cm of distance from the robot, aligned with its longitudinal axis. The robot emitted its call, and the echo caused by the object was recorded by the microphone array and filtered with a matched filter. The output of the matched filter was windowed with the usual rectangular window of length 1 ms centred on the autocorrelation function representing the echo, and this array signal was then fed to the three DOA estimators. The resulting pseudospectra were interpolated with a cubic spline by a factor of 25, to have a more accurate estimate of the main lobe width, and then normalised with respect to their maximum. The two -3 dB points were

then seeked in the magnitude of the pseudospectra. The interpolated pseudospectra are reported in Fig. 3.4.

DOA estimator	Delay and Sum	Capon method	MUSIC
Half-Power Beam Width [°]	19	4	4.6

Table 3.2: Half-Power Beam Width of the DOA estimators implemented. The cylindrical obstacle was placed at 0° and at 50 cm of distance from the robot

The obtained HPBW values are reported in Tab. 3.2, and they were obtained by averaging five distinct detections. Once again the results are not surprising: DAS has the widest main lobe among the three algorithms, while Capon and MUSIC have comparable performances that exceed by a fair margin the one offered by DAS. This very simple experiment is not intended to be an exhaustive analysis of the resolution capabilities of the three algorithms, since the effect of only one parameter was investigated and just for one single value of it. Instead, it is meant to be an indication of the general trend in the capabilities of the algorithms once they would be deployed on a moving robotic platform: if multiple sources need to be located with a good degree of separation, then more refined algorithms can provide the desired performance. If instead the interest is focused on the presence of obstacles of whatever nature in a general direction, then also a simpler algorithm like DAS could be applied.



(a)

Figure 3.4: Pseudospectra obtained by applying the DOA estimation algorithms to the echo generated by a cylindrical obstacle placed at a distance of 50 cm (part 1).

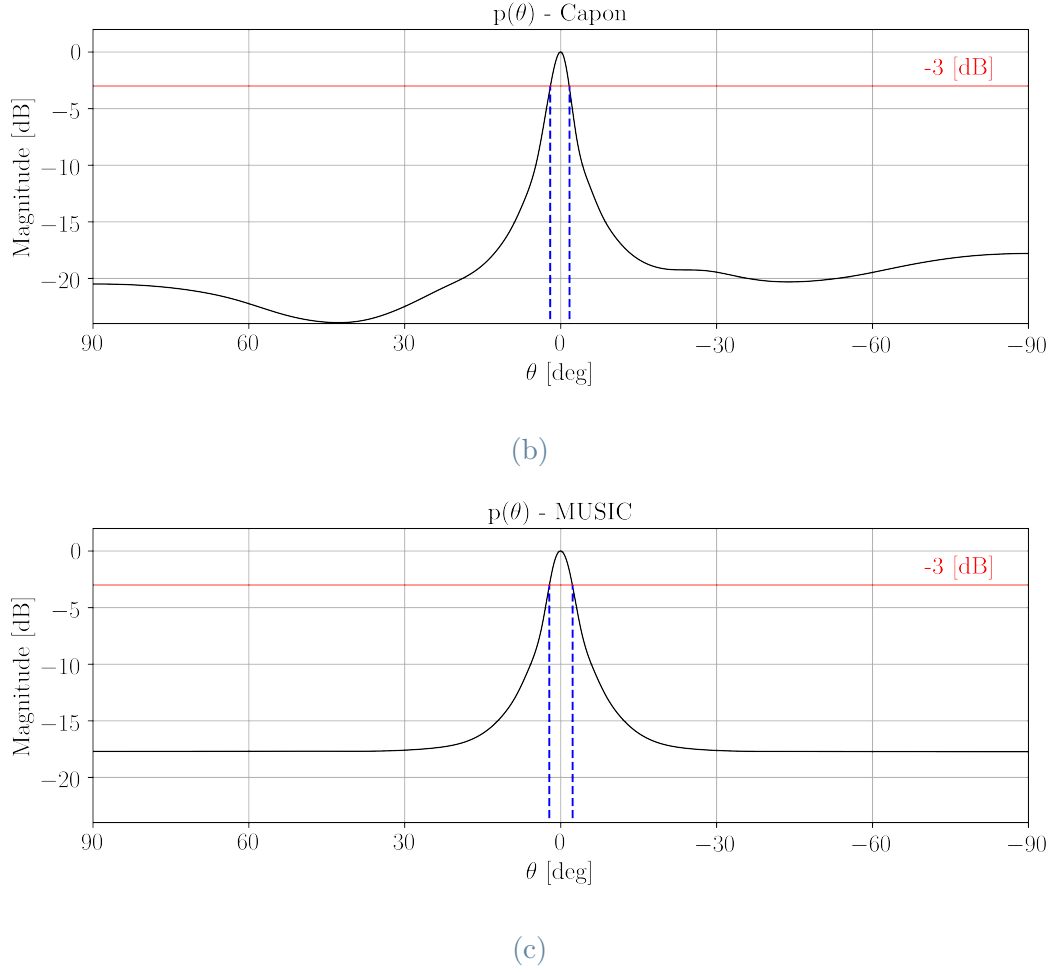


Figure 3.4: (continued) Pseudospectra obtained by applying the DOA estimation algorithms to the echo generated by a cylindrical obstacle placed at a distance of 50 cm. The call employed is a linear descending sine sweep from 60 kHz to 20 kHz, windowed with a Tukey window with roll-off factor α of 0.3.

Obstacle positions

This last verification step takes into account how the position of the obstacles with respect to the robot's axis plays a role in the accuracy of their localisation. To perform this test, two bottles were placed at a distance of 50 cm with an angle separation of 30° , starting from -90° and spanning the entire frontal half-plane. The two tallest peaks were then selected in the obtained pseudospectra as an indication of the position of the two objects.

What was observed for all the algorithms is a progressive decrease in the localisation accuracy of both objects the farther they are positioned with respect to the robot's axis: this is caused by how the acoustic power emitted by the robot is distributed in the

frontal hemisphere. As it will be shown in Chapter 4, the radiation pattern of the source employed is concentrated along its axis, so echoes are stronger if they are generated by obstacles close to the robot's longitudinal axis. The performance of the DOA estimators depends strongly on the SNR of the array signal: to a low signal power corresponds a worse localisation performance in terms of accuracy. A comparison of the pseudospectra obtained for various objects configurations is shown in Fig. 3.5.

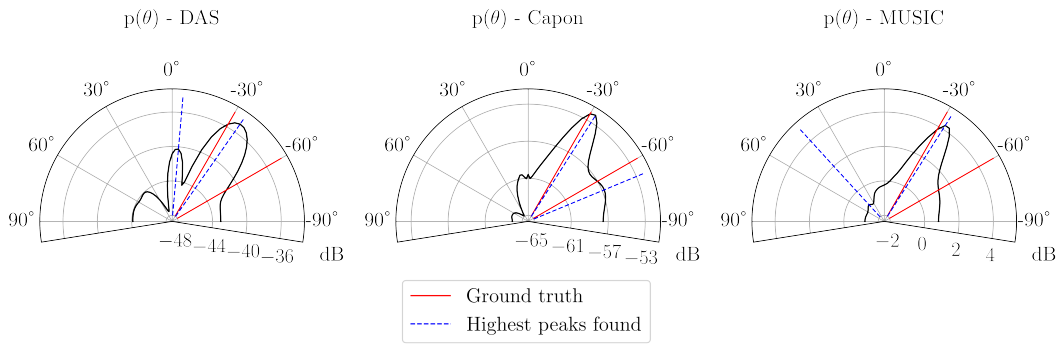
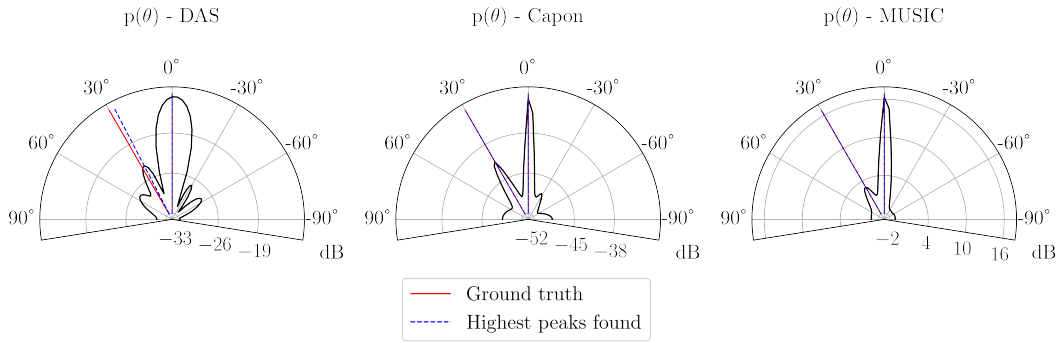


Figure 3.5: Pseudospectra obtained by echoes coming from pairs of objects placed in asymmetrical configurations around the robot (part 1).

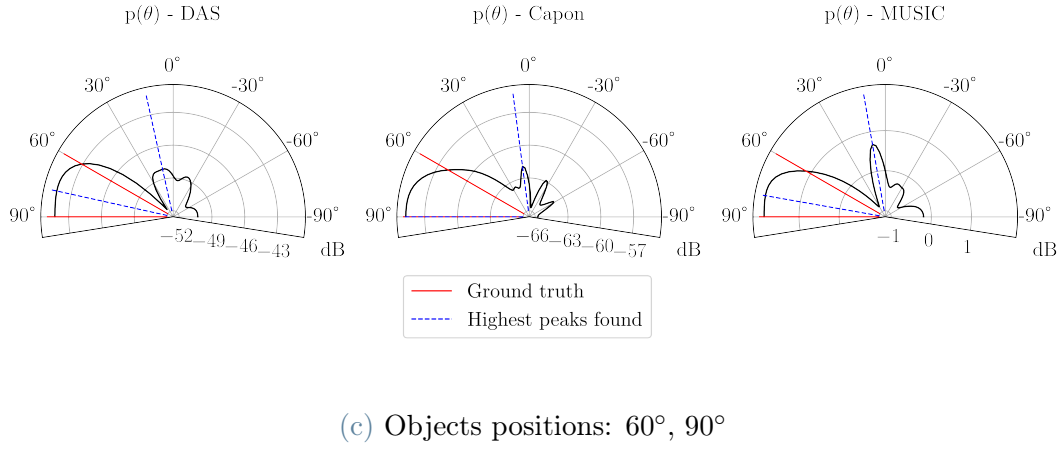


Figure 3.5: (continued) Pseudospectra obtained by echoes coming from pairs of objects placed in asymmetrical configurations around the robot.

The pair of objects placed in configuration 1 (0°, 30°) are located with good accuracy by all three algorithms, despite the echo at 0° producing a much more prominent peak in the pseudospectra than the one coming from the side. In configuration 2 (−30°, −60°) the object closest to the front of the robot can still be recognised, but the obstacle placed at −60° is completely invisible to the robot. In the third and final configuration (60°, 90°) the localisation performance is poor for both obstacles. The overall level of the pseudospectra obtained can also give an indication about the received echoes strength: comparing the pseudospectra in 3.5a with the ones in 3.5c, there is an excursion of about 20 dB between their maxima.

3.2. Final Remarks on the Processing Pipeline

In the previous Section the full signal processing chain that gives the robot the ability to echolocate objects around itself has been detailed, and in particular great attention was dedicated to the Direction of Arrival estimation algorithms that have been tested as the mechanisms with which the robot would identify the directions from which echoes arrive to the microphone array. Since the application for such a pipeline is on a mobile platform whose focus is a real time reactive behaviour, in the end the decision to favour reaction speed over detection accuracy was taken: DAS was the only algorithm which was implemented and tested while the robot was performing an autonomous navigation of a controlled environment. The other two algorithms, i.e. Capon and MUSIC, are still valid as a more accurate alternative, when the computational power of the platform they are

run on is enough to reduce their execution time to an acceptable level.

4 | Robot Design and Implementation

This Chapter is dedicated to explaining the architecture of the robot that I built and programmed, the active sensing Ro-BAT. I both present the details of every single component and give an overview on how the system is intended to work as a whole. In Section 4.1, I go over the hardware components that constitute the robot, starting from its base configuration and detailing the added components for the intended functionalities. In Section 4.2, I detail the acoustic measurements carried out to evaluate key aspects of the transducer and microphones employed, such as their frequency response and directivity, and how these quantities can influence the robot's capability to sense the world. In Section 4.3, finally, I explain how the signal processing pipeline described in Chapter 3 is integrated into the robot's behaviour.

4.1. Hardware

The robot design process was strongly influenced by the decision to use consumer-grade hardware wherever possible. This approach has a number of advantages: first, costs and development time for the realisation of a single robot are decreased. Secondly, scalability is also improved, since once a satisfying prototype is established it can be steadily replicated. These two features are well suited for a swarm robotics scenario. The selection process for the single components and the design stage to reach a final working prototype are explained. A picture of the final robot and a block diagram highlighting the main components and their connections is shown in Fig. 4.1.

4.1.1. Robot Base Configuration

The starting point for the development of my Ro-BAT consists of a Thymio II wireless robot [37] by Mobsya [36] equipped with a Raspberry Pi Model 4B single-board computer (SBC) [29], powered by an external power bank through the Pi's USB-C socket. It is a

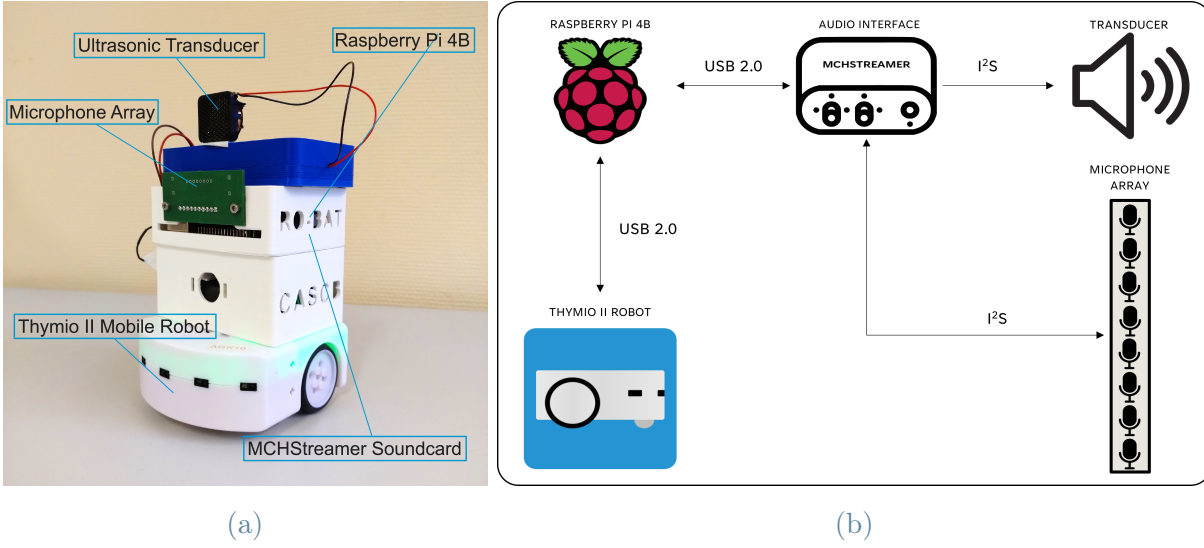


Figure 4.1: The active sensing Ro-BAT developed. (a) portrays the robot highlighting its main components, while (b) is a functional block diagram indicating the main components of the robot and their connections. As shown, the Raspberry Pi communicates with both the robot and the audio interface, which expands the capabilities of the single-board computer regarding audio emission and reception.

popular robotic platform in the field of swarm robotics [9, 19, 33, 55] and already in use in the Cyber-Physical-Systems Group at the University of Konstanz [40]. The reasons for such a combination are the contained cost and dimensions, the ease of communication with the robot over Wi-Fi networks for efficient remote development, and the possibility to expand its capabilities with additional sensors and actuators. A custom 3-D printed case was also developed to hold the Raspberry Pi, the audio interface, the emission circuit and the microphone array in place on top of the robot.

The Processing Unit

The Raspberry Pi Model 4B employed had the following specifications:

- **CPU:** Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.8GHz
- **RAM:** 8GB LPDDR4-3200 SDRAM
- **Network:** 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE, Gigabit Ethernet
- **Peripherals connectivity:** 2 USB 3.0 ports; 2 USB 2.0 ports; Raspberry Pi standard 40 pin GPIO header

- **OS:** Raspberry Pi Os with Desktop, 64bit

The Raspberry Pi was interfaced with an external laptop via both the SSH protocol and the VNC protocol for remote development. Two USB ports were needed to connect the audio interface and the Thymio II robot to the computer, respectively, while the GPIO header was used as both a power supply from pins 1 and 2, and for the I²C bus located on pins 3 and 5.

The Mobile Platform

The Thymio II robot is an open source differential drive robot developed for education and research purposes [37]. Its dimensions are contained, measuring 11 by 11.2 by 5.3 cm, so several robots can fit in a relatively small arena. It offers a multitude of sensors, among which the most relevant for my use case are two reflected light sensors on the bottom of the robot, which can be used to recognise the boundaries of an arena traced with white tape on a dark carpet. As far as actuators are concerned, the two wheels motors are fundamental for the robot's movement, and also the many RGB LEDs installed on the robot can be exploited to give visual feedbacks to the user effectively. As previously stated, all the equipment is fitted on top of the robot with a 3-D printed case which latches onto the lego attachments provided.

4.1.2. Audio Equipment

The core of all audio data management is a miniDSP MCHStreamer Kit USB interface [34]. This tiny USB Audio Class 2 compliant device hosts an XMOS XCore-200 microcontroller as its streaming engine, and it can communicate with Linux devices via an Advanced Linux Sound Architecture (ALSA) compliant driver. It is capable of handling digital audio in various encoding formats, such as I²S, PDM, TDM, DSD, ADAT, TOSLINK, and S/PDIF up to sample rates of 384 kHz and bit depths of 32 bit. The digital audio format can be selected by loading different firmwares onto the device. It comes packaged with two optical ports for ADAT/TOSLINK input and output, and two JST two-pin housings for S/PDIF connection. For the logic-level audio protocols three twelve-pin headers are placed on the PCB, and cable harnesses are used to wire suitable audio devices.

This particular audio interface model was picked as a starting point in the development of the passive sensing Ro-BAT [9], and proved itself to be a reliable solution for audio recording on small platforms, so I followed the same path, extending its usage for audio emission as well.

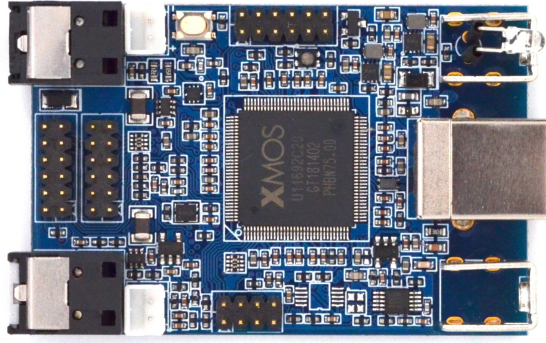


Figure 4.2: The miniDSP MCHStreamer Kit audio interface

Given that the robot should mimic the acoustic emission of a wide variety of bat species, it must be able to emit and listen to frequencies reaching even 90 kHz. The most suitable firmware among the ones provided by the interface’s manufacturers is then the one labelled AllRate [35], which is able to handle bidirectional I²S audio with a sample rate of up to 192 kHz. Set this way pins 2, 4, 6 and 8 of header J1 each carry two input channels, while pins 1, 3, 5, and 7 provide eight output channels in total. The bit clock line is located on pin 10, while the frame synchronisation clock line is located on pin 12. Finally, pin 11 provides a ground voltage reference. The input channels are connected to an eight-microphone array, while the first output channel is connected to a digital-to-analog converter that drives the emission circuit.

Audio Emission

Given their use in the literature [22, 45], we employed a Polaroid-type electrostatic transducer, in particular the Senscomp series 7000 Ultrasonic Sensor [44]. This device is properly called a transceiver because it acts as both an emitter and a receiver in compatible ranging modules. Ranging modules work by emitting narrow-band tones which once reflected by an object placed in the direction of emission excite again the thin metallic membrane of the transducer. Since bat calls are inherently wide-band and with many possible trends in the time-frequency space, the ranging module is not suitable for this use case. The idea is then to exploit the aforementioned transceiver only as an emitter, in combination with a custom built high-voltage amplifier which can be driven by any signal coming from a digital-to-analog converter connected to the sound card. With this configuration, the Ro-BAT is capable of emitting a wide range of signals through an actuator specifically designed to work at ultrasonic frequencies.

As a power source for the amplifier, standard 9 V batteries were chosen after trying other strategies which involved a 5 V power bank and a step up voltage regulator. The power

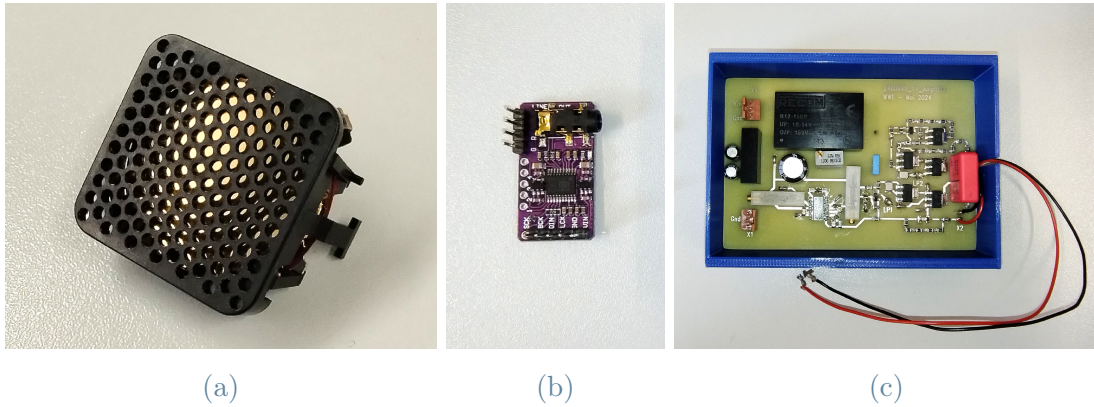


Figure 4.3: The components which constitute the audio source for the Ro-BAT. (a) represents the Senscomp Series 7000 ultrasonic transducer; (b) shows the breakout board mounting the TI PCM5102 DAC chip, and (c) the custom-built high-voltage amplifier. The transducer is then connected to the leads departing from the left side of the amplifier.

bank did not work as the amplifier could not drain steady current from it, resulting in the emission of pulsating tones instead of a continuous signal. The circuit comprising the transducer and the amplifier had already been developed before I started to work on the project, but it was my job to actually test its functioning together with the audio interface. This meant that a digital-to-analog converter was needed to bridge the gap between the audio interface and the amplifier. The choice fell on a breakout board developed around a TI PCM5102 Audio Stereo DAC with PCM interface [50]. This board is convenient for driving analog audio devices with Pulse Code Modulation audio data, especially at higher sampling rates. It offers solder pads directly connected to the configuration pins of the converter, and both input and output connections are realised with male pin headers. The transducer and the components driving it can be seen in Fig. 4.3.

Audio Reception

Following what has been done before for the passive localisation approach [9], a Uniform Linear microphone Array was elected as the audio receiving device for the active sensing Ro-BAT as well.

Microphone Selection Once the ultrasound source was set up, the first approach was to try to record emissions and echoes with an array that was already available, shown in Fig. 4.4. It consisted of eight Knowles SPH0645LM4H-B I²S digital microphones [24]. The inter-element distance of the array was 2.70 mm, which corresponds to an upper limit in the frequency range of 63.5 kHz before the occurrence of spatial aliasing, but the

microphones themselves were not intended to use at ultrasonic frequencies: with sampling rates higher than 48 kHz, the dynamic range of the microphones was severely limited by the amount of internal noise.

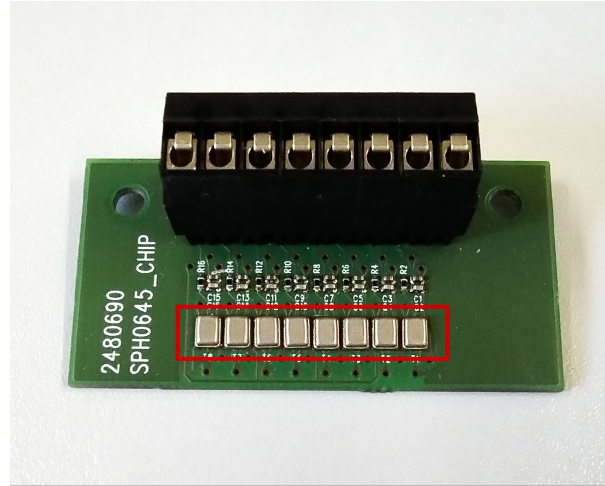


Figure 4.4: The eight-channel microphone array that was already available. The actual microphones are the ones highlighted by the red frame.

It was clear that the only available strategy was to keep the same array geometry but to pick a different MEMS microphone model specifically designed for ultrasound recording. Also for this choice, the suggestion came from the literature [21]: the Knowles SPH0641LU4H-1 PDM digital microphones [23] are an example of MEMS microphones with a built-in ADC whose bandwidth extends in the ultrasonic range. The new challenge was then to bridge the gap between the audio encoding formats used by the microphones and the interface: since the interface's only firmware which supported audio emission at ultrasonic frequencies was based on the I²S protocol, which encodes audio with Pulse Code Modulation, and these microphones encoded audio with Pulse Density Modulation, specific format converters were needed to make the two kinds of devices compatible with each other. Following this strategy, I designed a microphone array suited for ultrasound recording through an iterative process. The first model I designed demonstrated the feasibility of such a solution, but due to unrevised datasheets suboptimal choices were made both in the design and in the realisation process. The second model I devised, instead, worked seamlessly and was ultimately installed on the robot as its audio receiving device.

In the design and realisation process I was helped by the Electronics Service Centre located at the University of Konstanz Workshop: the workflow was generally initiated by my summary drawing of the needed connections to implement the desired data flow, at which followed a lay down of the schematics to accommodate all electrical requirements

of the components involved. After a common revision of the schematics, the PCB layout stage followed, and once the printed circuits were delivered back to the workshop, the components were soldered on the board and the final product could be tested accordingly.

First Ultrasonic Array Prototype The first array model designed, represented in Fig. 4.5 integrated eight microphones of the aforementioned model and two TI PCMD3140 Quad-Channel PDM input to I²S output converters [52]. These devices can be programmed via an I²C or SPI interface by modifying the content of their registers. I chose to wire them so that they could be controlled via one of the I²C bus interfaces located on the GPIO header of the Raspberry Pi. When working with such an interface, it is common practice to have each device correspond to a different I²C address so that they could be controlled individually. This was not the case for the two converters, since for this particular model the address is fixed by the manufacturers. The consequence is that each I²C instruction was imparted to both converters at the same time, with no possibility to interact with them individually. Each converter was wired to four microphones, taking advantage of the capabilities of the chips to the maximum. The main issue revealed itself when devising the audio connections between the converters and the sound card: while the latter works with a more common two channels per data line implementation of the I²S protocol, the converters are able to send more than two channels on each data line, so the manufacturers provided only one serial data out pin which is intended to carry all the converted audio to the host device. This meant that the sound card could see data coming only from two microphones connected to each converter. What I had devised was then to route out audio data both from the SDOUT1 pin and from the GPIO1 pin, which is supposed to be General Purpose and capable of being used as a secondary audio data bus according to the latest revision of the device's datasheet. It was then revealed that this was not feasible because there was not any readily available option to program the converters to route audio data out of the GPIO1 pin.

In the end, the first microphone array prototype was working in concept, since at least four out of the eight microphones installed were actually reachable by the sound card and able to capture audio at ultrasonic frequencies with a higher Signal-to-Noise ratio with respect to the previous array model. However, it did not constitute a Uniform Linear Array, since the four working microphones were alternated in pairs with the four unreachable ones. It was not suited for ULA audio processing, but it still represented a useful tool since in combination with a working ultrasonic audio source I was able to work on the ranging capabilities of the robot nonetheless.

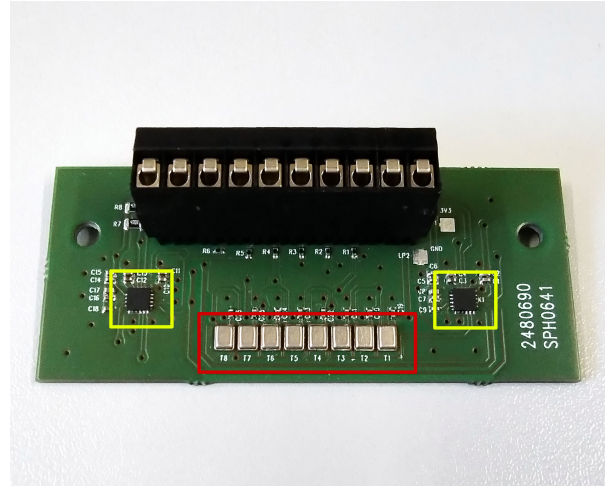


Figure 4.5: The first model of a microphone array for ultrasound recording. Microphones are highlighted by the red frame, while the converter chips are inside the yellow frames. Due to ill-designed connections only microphones 1, 2, 5 and 6 (from the right) could send data to the audio interface.

Second Ultrasonic Array Prototype Building on what was working in the previous model and trying to improve what failed, the second prototype was devised with four TI PCMD3180 Octal-Channel, PDM input to I²S output converters [51], one for each pair of microphones, depicted in Fig. 4.6 These devices are a more capable model of the previous ones, being able to handle eight microphones at once for each device. Furthermore, their I²C address is not fixed, but can be set by connecting two dedicated pins to a high- or low-voltage reference to modify its last two bits. This ensures four possible addresses for the same device, meaning that I could interact with each converter on the board independently if needed. If instead I wanted to impart an instruction to all the converters at once, the broadcast mode could be enabled by setting the content of a specific register, so that all converters would switch their address to a common one. Once again, this is not usually done when working with the I²C bus, but in my case it suited all my needs without compromising the device's performance. The final working version of this new array model was achieved after a revision of the initial PCB layout to accommodate the converters' new requirements: contrary to their less capable version, these devices exhibit a hardware shutdown pin which must be connected to a high-voltage reference in order to boot the device, and their analog low-voltage reference connection is located on the thermal pad on the bottom of the device, and not on a pin. These two specifications were overlooked in the first schematics design, and so that had repercussions on the first iteration of the PCB layout and fabrication.

After troubleshooting these inconveniences, the objective was reached: a uniform linear array of MEMS microphones capable of recording ultrasonic audio with a suitable dynamic range was operative. The last array model is the one that was installed in the robot and constituted its audio receiving device.

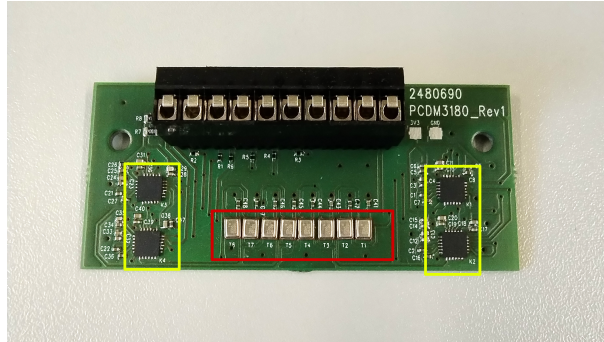


Figure 4.6: The second and final microphone array model. Each converter serves a single pair of microphones, so that the two-channels-per-data-line format of the audio interface is met. Microphones in the red frame, converter chips in the yellow ones.

4.2. Acoustic Characterisation of the System

After the robot was assembled, the acoustic properties of the system needed to be assessed properly, to evaluate how they could influence its echolocation capabilities. A series of measurements were then carried out to characterise both the transducer and the sensors in the microphone array in terms of their frequency response and directivity patterns. All the measurements were carried out in room Z723 located in building Z at the University of Konstanz campus. The room's dimensions are approximately 4.7 m by 7 m by 3 m, and it does not have any special acoustic treatment. The floor is carpeted with a square grid pattern whose side measures 0.5 m, the ceiling is made out of perforated metallic sheets, and the walls are flat hard painted surfaces. The room is mostly empty, except for a desk and a chair in the far-left corner from the entrance door, and a rack containing the measurement equipment. All the measurements were carried out with a sampling rate of 192 kHz. The tools at disposal for this task are listed here below:

- External PC.
- RME Fireface 802 USB audio interface (Audio AG, Heimausen, Germany).
- GRAS 12AA power module (GRAS Sound & Vibration, Holte, Denmark).
- GRAS 26AC 1/4" preamplifier (GRAS Sound & Vibration, Holte, Denmark).

- GRAS 40BF 1/4" free-field microphone (GRAS Sound & Vibration, Holte, Denmark).
- Brüel & Kjær Type 4231 sound level calibrator (Brüel & Kjær, Nærum, Denmark).
- Sanken CO-100K microphone (SANKEN MICROPHONE Co., Ltd., Tokyo, Japan).
- Harman Kardon AVR 445 OM audio receiver and amplifier (Harman International Industries, Inc., Stamford, Connecticut, USA).
- Peerless XT25SC90-04 tweeter (Tympany, San Rafael, California, USA).

4.2.1. Transducer Frequency Response

To measure the frequency response of the emission apparatus the robot is placed on the ground, and the GRAS calibration microphone is set at 1 m of distance, aligned with the centre of the transducer. The recording chain consists of the GRAS microphone and preamplifier polarised by the 12AA power module. The output of the power module is sent to channel 9 of the Fireface audio interface. The distance was checked with a Bosch UniversalDistance 30 laser range finder (Bosch, Gerlingen, Germany), and the vertical and horizontal alignment with a Bosch Quigo laser crosshair (Bosch, Gerlingen, Germany). The robot emitted five linear descending sine sweeps of length 6 ms, from 95 kHz to 15 kHz, one second apart from each other. The sweeps were windowed with a Tukey window with roll-off factor α of 0.2. Each sweep was then time-gated in the recording, to discard reflections that arrived at the calibration microphone. It was possible to do so because the duration of the sweeps was shorter than the shortest time that a reflected wave front would have taken to arrive at the microphone. With this strategy, measurements could be considered as performed in an anechoic environment. After isolating each sweep, their unilateral Discrete Fourier Transform is retrieved, and the RMS spectrum of the sweeps is obtained by dividing each frequency bin by $\sqrt{2}$. The mean RMS spectrum is then computed.

The measurement microphone is calibrated with the Brüel and Kjær calibrator, which emits a reference tone at 1 kHz with a Sound Pressure Level (SPL) of 94 dB. After computing the dB SPL to dB RMS conversion factor, the RMS spectrum of the transducer is converted to a SPL spectrum, to finally estimate the frequency response of the emission chain in terms of the actual pressure level generated.

The result is shown in Fig. 4.7. The transducer exhibits a plateau centred at 50 kHz, and spanning approximately 10 kHz. This is not surprising, since according to the datasheet of the device its intended frequency of operation is exactly 50 kHz. However, the device

is capable of producing a satisfying sound pressure level starting from around 30 kHz up to 90 kHz, if its -20 dB bandwidth is considered.

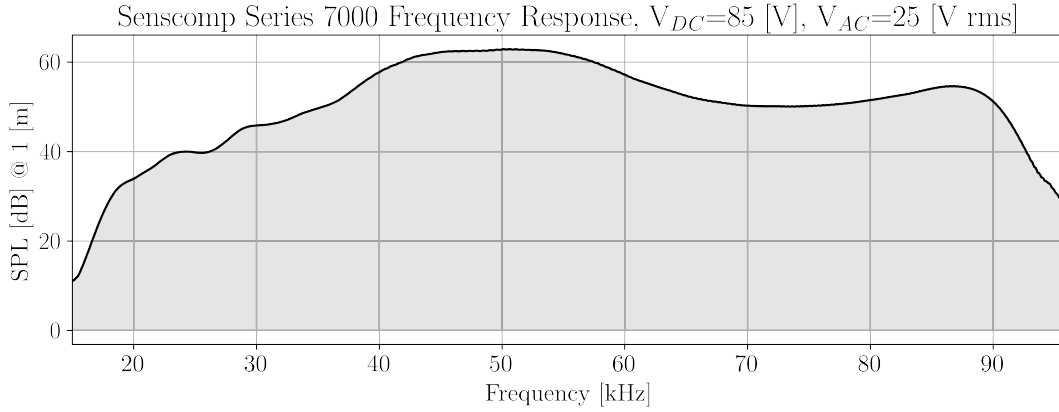


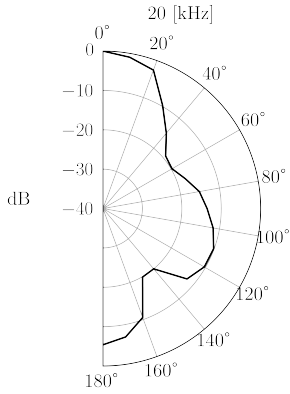
Figure 4.7: Frequency response of the Senscomp Series 7000 transducer, powered by the custom built high-voltage amplifier.

4.2.2. Transducer Directivity

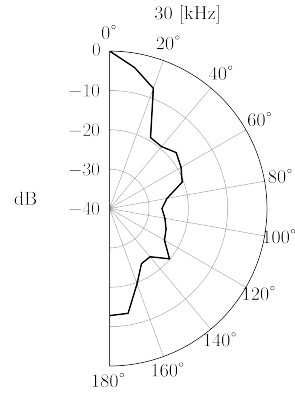
To characterise the radiance pattern of the transducer, the robot was placed on the ground, with a protractor printed on a sheet of paper underneath. The Sanken CO-100K recording microphone was placed at 1 m of distance and was plugged into input channel 9 of the Fireface audio interface. The Sanken CO-100K microphone was used in place of the GRAS calibration microphone because the GRAS microphone did not have a large enough dynamic range to make it possible to record at an acceptable level emissions from all sides of the transducer. The improved dynamic range of the Sanken microphone was necessary, even though its frequency response is not as flat as the calibration microphone [42]. Since only a relative characterisation is looked after, this does not constitute an issue for the integrity of the measurements taken, though. The robot emitted five sweeps as described in Section 4.2.1, to maintain the anechoic propagation condition, which are recorded through the Sanken microphone. Then, the robot is manually rotated in steps of 10° . The radiance pattern of the transducer is assumed to be left-right symmetrical, so only measurements between 0° and 180° are taken.

The recorded sweeps are isolated and time-gated in the recording, and their DFT is taken. To properly display the directivity pattern of the transducer as an angle and frequency dependent function, a series of central frequencies have been defined from 20 kHz to 90 kHz in steps of 10 kHz. For each central frequency f_c , bins belonging to the interval $[f_c - \Delta f, f_c + \Delta f]$ where $\Delta f = 2$ kHz contribute to the directivity computation.

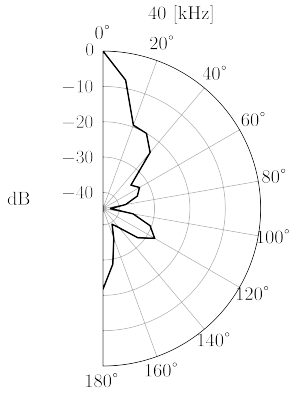
Results are displayed in Fig. 4.8. As expected, the higher the frequency the narrower the main lobe of the transducer becomes, meaning that less and less energy will impinge on objects which are located to the sides of the robot. It is worth noting that up to 40 kHz the transducer has a prominent emission also on its back side, a feature that could be exploited by other robots in a hypothetical swarm configuration to locate other agents which are positioned in front of them and with the same orientation passively. This backside emission is then much fainter for higher frequencies.



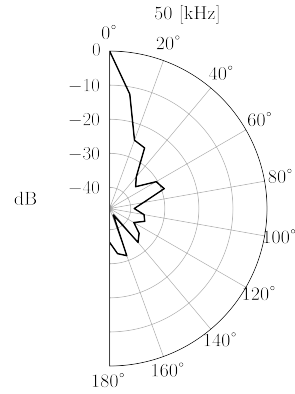
(a) Directivity pattern at 20 kHz



(b) Directivity pattern at 30 kHz

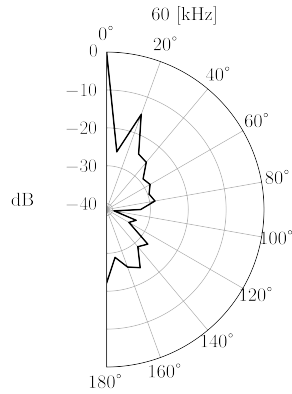


(c) Directivity pattern at 40 kHz

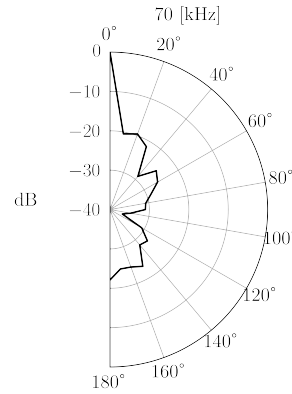


(d) Directivity pattern at 50 kHz

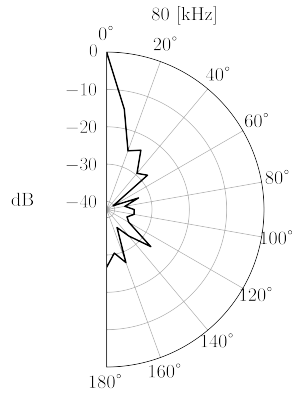
Figure 4.8: Directivity patterns of the Senscomp Series 7000 transducer as function of frequency and emission angle (part 1).



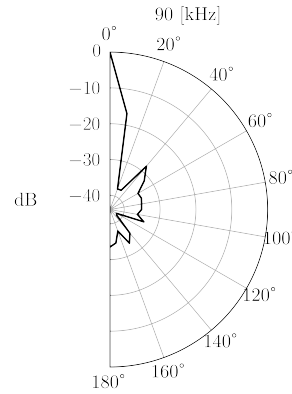
(e) Directivity pattern at 60 kHz



(f) Directivity pattern at 70 kHz



(g) Directivity pattern at 80 kHz



(h) Directivity pattern at 90 kHz

Figure 4.8: (continued) Directivity patterns of the Senscomp Series 7000 transducer as function of frequency and emission angle. Each pattern is computed considering the central frequency ± 2 kHz.

4.2.3. Microphones Frequency Response

To evaluate the frequency response of the microphones constituting the array, the free-field calibration method was employed [18]. This calibration method involves the subsequent recording of the same source by a calibrated microphone with a known frequency response, and by the microphone(s) of which the frequency response is looked after. By properly comparing the Fourier Transforms of the two recordings, it is possible to estimate the sensitivity of the target microphone. For this method it is necessary to be in a free-field condition, which can be obtained with the usual method of employing very short sine sweeps as the source signal. The audio source for this experiment is a Peerless XT25SC90-04 tweeter, driven by signals coming from the Harman Kardon amplifier, which is connected to the Fireface soundcard as its main audio output. The sweeps used are

3 ms long, and they ascend linearly from 15 kHz to 95 kHz. These are recorded both by the GRAS calibration microphone, and by the MEMS microphone array installed on the robot. After the sweeps are isolated in both the calibration microphone and the target microphone recording, their RMS spectra are computed. The calibration microphone RMS spectrum is turned into a pressure RMS equivalent spectrum thanks to the sound level calibrator, and then the sensitivity of the target microphones in terms of RMS over Pascal is obtained by taking the ratio of their RMS spectra and the pressure RMS equivalent spectrum of the calibration microphone. The results of the calibration are shown in Fig. 4.9.

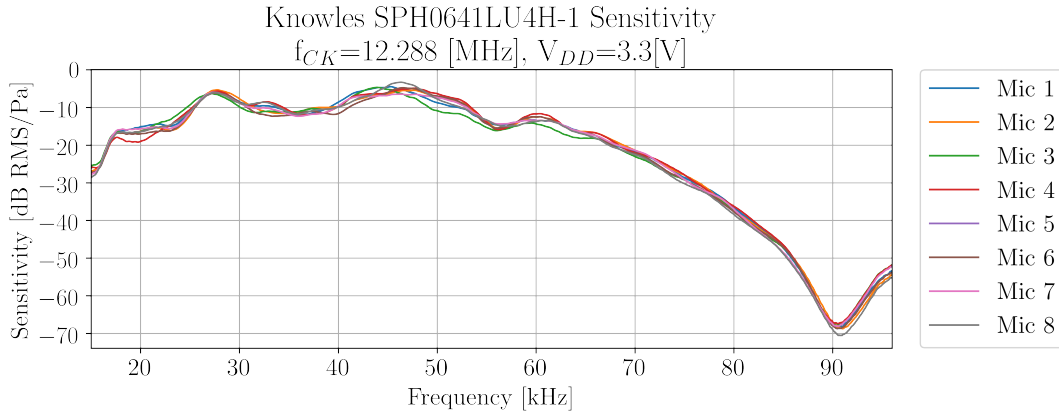


Figure 4.9: The frequency response in terms of dB RMS of the eight microphones which constitute the array installed on the robot.

The response of each microphone is slightly different, but they all follow a very similar trend. Their -20 dB bandwidth spans from 15 kHz to about 72 kHz, meaning that they are not able to pick up the higher end of the ultrasonic emission of the transducer with a satisfying level. This however is not of interest for this particular array geometry, since as previously stated spatial aliasing would limit the use of the array up to 63 kHz.

4.2.4. Microphones Directivity

To measure the directivity of the microphones in the array configuration, the robot and the tweeter are put 1 m apart, and the very same sine sweep signal as in the previous Section is adopted. For every angle, five sweeps are recorded by the robot. Since an estimate of the directivity of the microphones is needed for every possible angle of incidence, the robot was rotated manually in steps of 10° until it accomplished a full turn, with the usual protractor printed on a sheet of paper as a reference. The results are expressed in terms of the mean received signal power, and for each microphone they are normalised with

respect to the maximum power received across each direction. The directivity patterns are shown in Fig. 4.10. Even though the single MEMS microphones are omnidirectional according to the datasheet, the body of the robot in contact with their backside introduces a significant amount of attenuation, resulting in a cardioid polar pattern across most of the frequencies. The polar pattern obtained for 90 kHz differs from the others because, as it is possible to observe in Fig. 4.9, the microphones are not able to pick up those frequencies with a sufficiently high output level. The pattern obtained is therefore highly biased by a signal-to-noise ratio that is too low. For the other frequency bands, it is shown that the microphones can pick up sounds coming from angles up to $\pm 80^\circ$ without major attenuation. At the endfire direction, e.g. $\pm 90^\circ$, the overall picked up level is lower by at least 6 dB. This is because the microphones are of the bottom-port type: the wavefronts must travel through holes in the PCB to reach the microphones' diaphragms, and as the entrance angle gets steeper and steeper, they physically cannot traverse the silicon layer to reach the microphones any more.

4.2.5. Conclusions

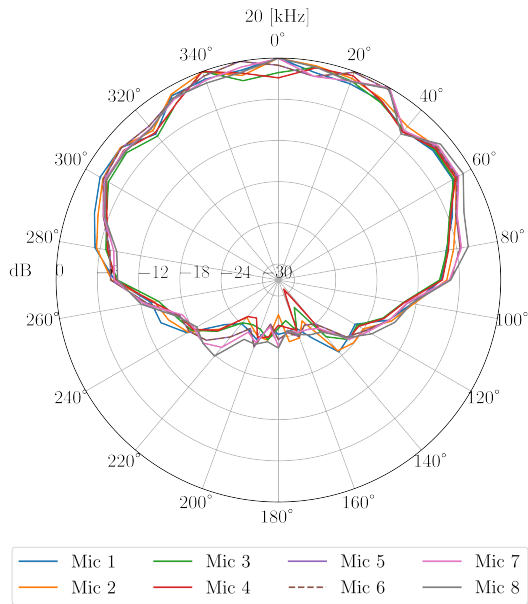
The measurements done attest the validity of the system built in relation to its task: the robot is capable of emitting loud wide-band signals, and is capable of receiving the echoes generated by its own calls with an acceptable level for further processing.

One major weakness in the system could be individuated in the directivity of the ultrasonic transducer: having a narrow beam concentrated in the frontal direction means that the system will be biased towards objects which are close to the longitudinal axis of the robot, making it more challenging to reveal obstacles which are placed on its sides.

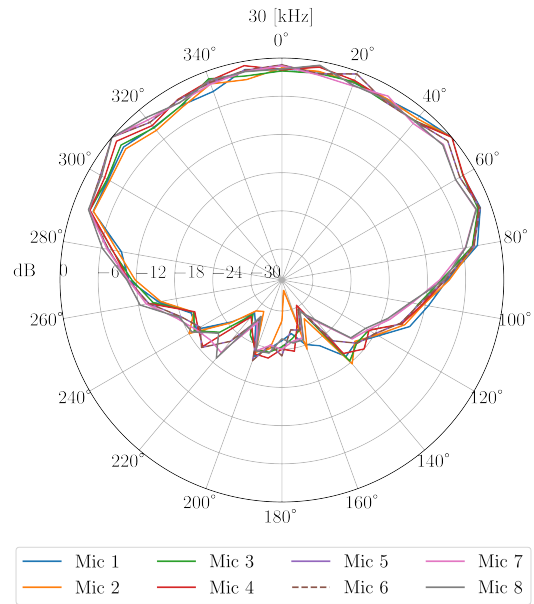
4.3. Software

The language chosen for development is Python, specifically the version 3.12, mainly for its versatility and portability. Being a high-level language, it offers many powerful methods which hide the complexity of the single operations, at the cost of sacrificing optimisation, which is an acceptable trade-off while prototyping. Some notable libraries which greatly helped in the software implementation are listed here below:

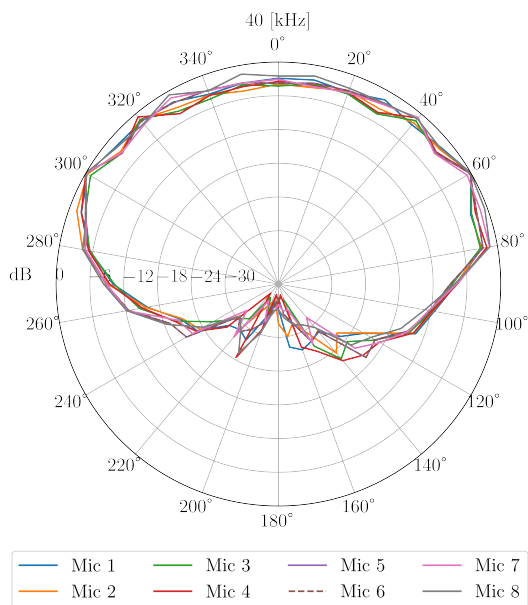
- **python-sounddevice:** [14] offers methods for controlling equipment for audio playback and recording.
- **python-soundfile:** [3] this module was used to read and write audio files.
- **thymiodirect:** [17] communication between the Raspberry Pi and the Thymio II



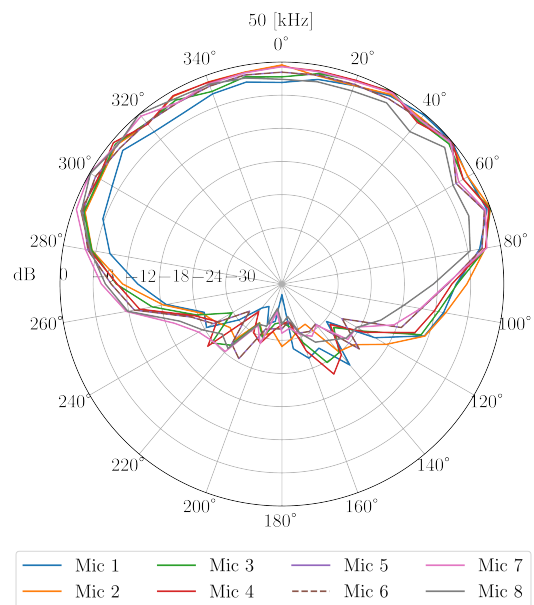
(a) Directivity pattern at 20 kHz



(b) Directivity pattern at 30 kHz

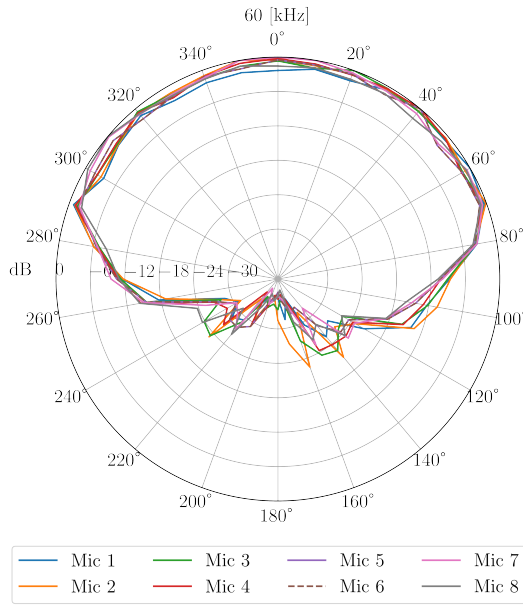


(c) Directivity pattern at 40 kHz

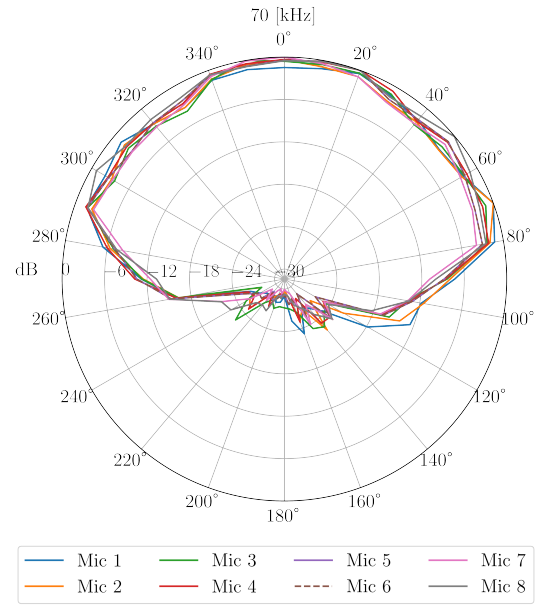


(d) Directivity pattern at 50 kHz

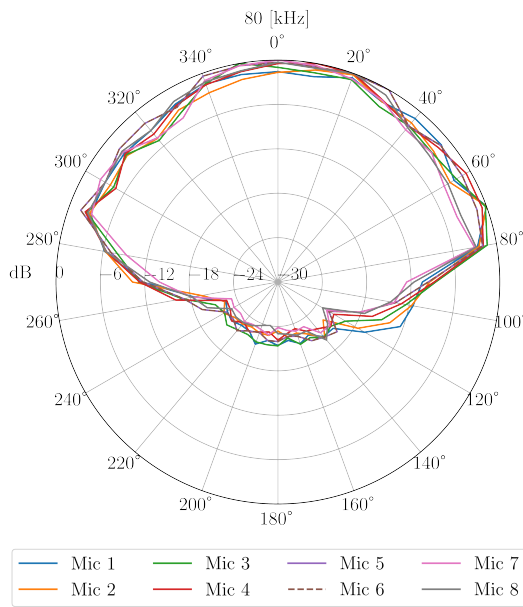
Figure 4.10: Directivity patterns of the Knowles SPH0641LU4H-1 microphones (part 1).



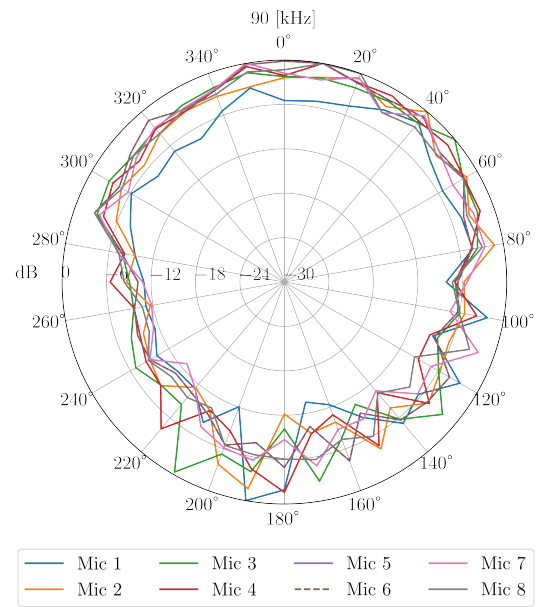
(e) Directivity pattern at 60 kHz



(f) Directivity pattern at 70 kHz



(g) Directivity pattern at 80 kHz



(h) Directivity pattern at 90 kHz

Figure 4.10: (continued) Directivity patterns of the Knowles SPH0641LU4H-1 microphones. Each pattern is computed considering the central frequency ± 2 kHz.

robot. The module enabled reading values from the robot's sensors and controlling its actuators.

The software implementation followed a modular approach, developing the robot's capabilities independently and then putting them together in a cohesive script to achieve the desired behaviour.

4.3.1. Latency Rating of the System

The first challenge which was presented to me once the hardware was set up was to quantify its latency, meaning the interval of time between the moment in which an output stream is initiated in the code and the moment in which a signal emitted from the transducer is recorded again by the microphone array and is available to the code for further processing. This time lag is of crucial importance for the responsiveness of the system: too high of a delay between calls and the system does not have an update rate suitable for real time scenarios.

To conduct an analysis of the performance of the system, I initiated a bidirectional audio stream on the robot, which recorded continuously while at the same time emitting a playback signal of length 30s. The latency can be obtained by cross-correlating the sample signal and the recording of one of the channels of the array. I conducted this test multiple times, refining the parameters which influenced the responsiveness of the audio equipment in the robot, and the results showed that a latency below 15ms was achievable in most of the cases without discarding a significant amount of input data frames. The results obtained are positive, but not optimal: the system takes around 15ms just for emitting a call which usually must be a few milliseconds long, meaning that the update rate of the system cannot exceed 50Hz in any case. It can still be used for real time applications, but with an expected lower update rate with respect to other sensors.

4.3.2. Obstacle Avoidance Behaviour

Once established that the system built could work in a real time scenario, it was possible to implement and test the obstacle avoidance behaviour of the robot accordingly.

The code governing the robot's behaviour is multi-threaded. One thread is responsible for acquiring audio data through the microphone array and appending it to a growing `.wav` file, which will be used at a later stage for post processing and evaluation of the robot's performance. A second thread executes the main instruction loop, whose structure can be described in the following steps.

1. **Seek current end of audio file:** the number of frames already written into the audio file is checked and stored.
2. **Call emission:** the Ro-BAT emits a linear descending sine sweep of duration 3 ms, from 60 kHz to 20 kHz, with a sweeping rate of 7 Hz. The sweep is windowed with a Tukey window with roll-off factor α of 0.3. This call will cause echoes when reflected by nearby obstacles. The microphone array records both the emitted call and the echoes that come back to the robot. The sampling rate for both emission and recording is 176.4 kHz, enough to respect the Nyquist-Shannon sampling theorem. The Nyquist frequency would be 88.2 kHz, which is greater than the maximum frequency elaborated by the robot (60 kHz).
3. **Forward advancement:** after the call emission and recording, the robot checks again the number of samples written in the audio file. If the audio file contains more frames with respect to the value found at step 1, that means that new audio data to be analysed is present, and the robot can make a decision on its next move based on the audio read from the file. The index at which the audio file starts to be read and the number of samples read are stored in an array, which at the end of the run will be dumped into a YAML file for post processing purposes. If the two values are the same, that means that the thread writing to the audio file was not executed while the call was emitted, and so no audio data was captured by the robot. If this is the case, the robot stops in place, since it was effectively "blind", or better "deaf" to the environment. This latter case was extremely rare during experiments and it affected the robot only for a fraction of a second.
4. **Echolocation of the nearest obstacle and consequent reaction:** the robot's call was powerful enough to cause perceivable echoes also from walls up to 3 m of distance from the robot. In the experiments, though, since it would have been difficult to tag the walls as well for post processing analysis, the FOV of the robot was limited to 1 m, to focus its detecting capabilities on tagged obstacles nearby. For the obstacle avoidance behaviour, a "danger zone" was defined as in Fig. 4.11: it consists of a sector of a semicircle with radius 20 cm, delimited on its left side at 40° and on its right side at -60° . The reason for these boundaries is the asymmetry of the robot: cables which could not be contained inside the case protrude from its right side, extending the robot's width in an asymmetric fashion. If no obstacles are found within the robot's FOV, or if the nearest obstacle found is outside of the "danger zone", then the robot proceeds forward on a straight line at a constant speed of about 8 cm s^{-1} . If the obstacle is located in the danger zone, instead, the robot reacts by turning in place in the opposite direction to avoid a collision with it. The

robot's turning motion consists of a rotation in place of either 20° or 15° , depending on whether the obstacle is located at an angle respectively greater or smaller than 30° on either side. The robot turns by rotating its wheels with the same speed but in opposite directions. The robot turns until its way is clear again, and it can resume its path. While the robot is rotating in place, the call emission stops until the robot resumes its forward movement. This is because the only reliable solution to make the robot accomplish precise controlled turns was to set the wheels' speed and make the system sleep for the amount of time necessary to achieve the desired turn, before waking it up again when the turning motion was accomplished. This means that while the robot turns, it cannot sense nearby obstacles.

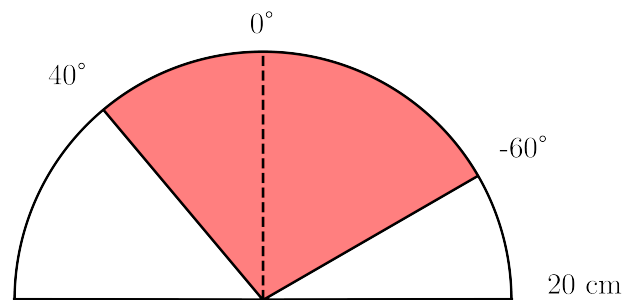


Figure 4.11: A representation of the Ro-BAT's "danger zone", as seen by the robot observed from the top. The asymmetry is introduced to accommodate the form factor of the robot: on its right side a thick USB cable protrudes, and can bump into obstacles if not accounted for.

5 | Tests and Results

In this Chapter the experimental setup employed to assess the robot's performance is detailed, and the consequent results are presented.

The robot's obstacle avoidance behaviour was tested in a controlled environment consisting of a rectangular arena with static obstacles. The metrics chosen for evaluating the robot's performance are the localisation error, in terms of both distance and angle, and after dividing the arena space into a grid, also the number of visits for each grid cell was taken into account to present the effect that the presence of obstacles has on the robot's trajectory.

Section 5.1 details the experimental procedure followed; Section 5.2 describes how data was extracted from the experiments' footage and audio recording, and finally section 5.3 shows the results which quantify the robot's performance, together with a discussion highlighting strengths and weaknesses of the system built.

5.1. Experimental Setup

The experiments were performed at the University of Konstanz campus, in room Z723 located in building Z, which is the same room used for the acoustics measurements carried out on the robot. A description of the room was already given in Section 4.2.

A black carpet measuring 2m by 1.55m was used as the arena to be explored by the robot. The edges of the carpet were marked with white tape, and four additional ticks were placed on the edges to redirect the robot to the inside of the arena. As obstacles, eleven empty glass beer bottles of the NRW (Nordrhein-Westfalen) model (height: 260 mm, base diameter: 64 mm) were placed to randomly cover the carpet's surface, leaving enough room between every pair of obstacles and along the edges of the arena to fit the robot's width. The reasons for choosing such obstacles are explained by the material and geometrical properties of the bottles: they are made out of a hard, reflective material that causes strong reflections of the emitted call, which propagate back to the robot immediately because of their cylindrical shape. A GoPro HERO11 Black camera

was mounted on the ceiling and was used to record the footage of the robot's runs in the arena. The relevant camera settings used are the following:

- Resolution: 2.7K (2704 x 1524).
- Frames per second (FPS): 60.
- Lens: linearised.
- Aspect ratio: 16:9.
- Audio sampling frequency: 48 kHz.

A picture of the arena and the camera setup is reported in Fig. 5.1. The bottles, the Ro-BAT and the corners of the arena were all tagged with ArUco markers [13] belonging to the `aruco.DICT_6X6_250` predefined dictionary. The robot and obstacle markers were used to obtain their respective positions, while the markers on the arena corners were used to obtain a pixel-to-metre conversion factor. In total, four obstacle configurations were tested, each one of them twice starting the robot's exploration at different points chosen at random in the arena. Tab. 5.1 resumes the run duration for every obstacle configuration.

The script governing the robot's obstacle avoidance behaviour was launched on it via Wi-Fi with an external PC, and the camera recording was started shortly after. At the end of each run, the output consisted of the run footage, the multichannel audio file recorded onboard by the robot via the microphone array, and a YAML file containing the reading points in the audio file expressed as indices and the number of frames loaded for analysis at each reading.

Obstacles Configuration	Run	Duration[min : sec]
A	A.1	8:35
	A.2	9:21
B	B.1	9:42
	B.2	9:59
C	C.1	10:05
	C.2	10:11
D	D.1	10:03
	D.2	10:03

Table 5.1: Duration of the runs in the arena for every obstacle configuration.

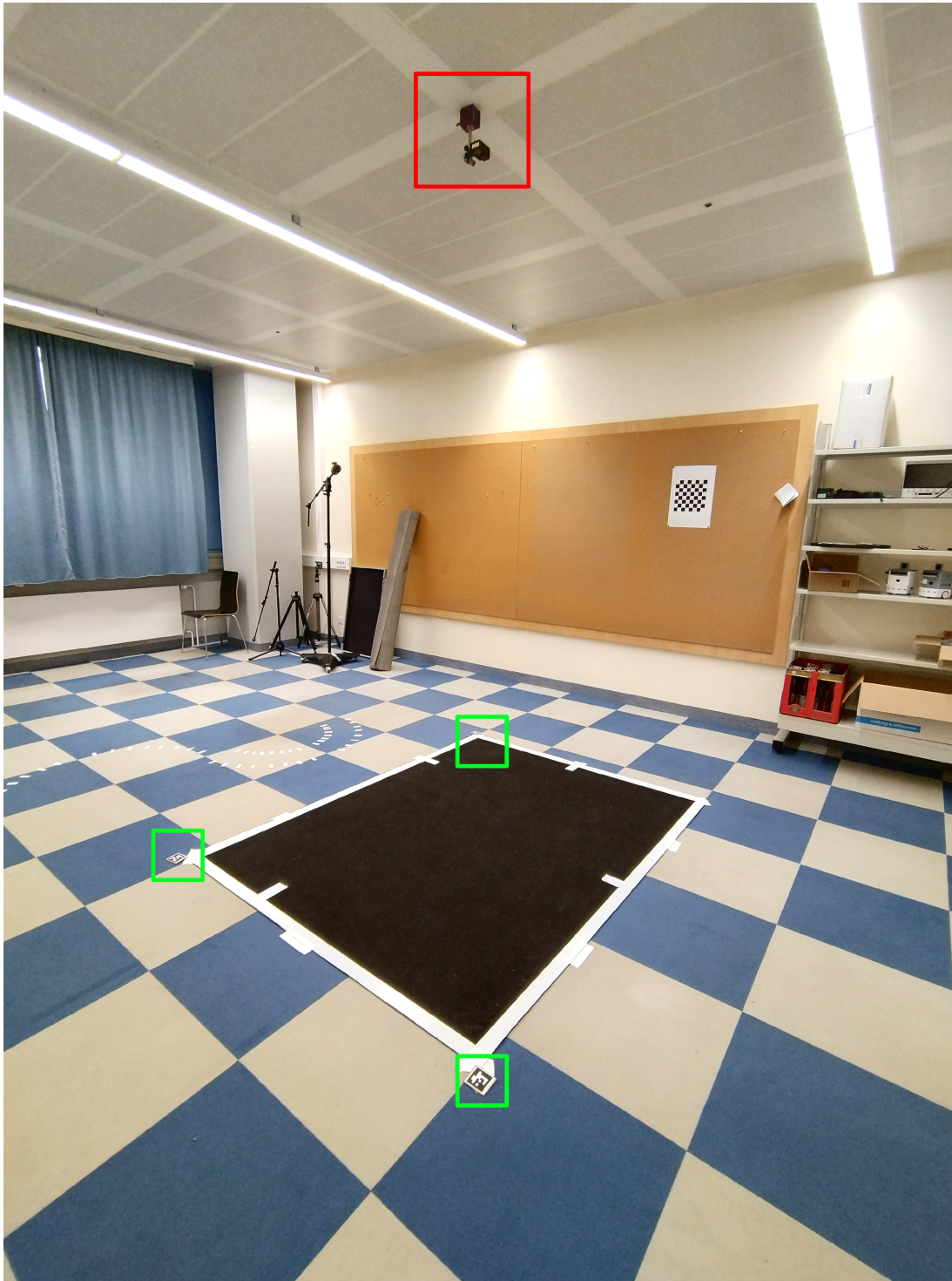


Figure 5.1: The experimental setup for recording the robot while it navigates autonomously. The camera on the ceiling is highlighted by the red frame, while the ArUco markers on the arena corners are inside the green frames.

5.2. Data Extraction and Post Processing

The video analysis and post-processing was performed with the help of the open source computer vision library OpenCV [6], in particular its Python bindings. The librosa library [32] and the python-ffmpeg library [25], which provides bindings for the FFmpeg multimedia framework [11], were used instead to extract the audio from GoPro footage.

The tracking process that was performed on each video starts with synchronising the GoPro video frames to the audio recorded by the robot: assuming that the audio recorded by the camera is synchronised to the relative video frames, by cross-correlating the audio recorded by the GoPro and the audio recorded by one of the microphones on the robot the video frames can also be synchronised with the audio recorded by the robot. This operation is fundamental to then compare at each occurrence the position of the obstacle found by the robot to the ground truth. The audio of the GoPro footage is extracted from the videos and piped into the PC memory thanks to the ffmpeg module. The audio data is loaded thanks to librosa and resampled at 176.4 kHz, which is the native frequency of the audio recorded by the robot. After cross-correlating the two audio signals the delay between the two is found, and the starting frame for the analysis is computed by multiplying the delay by the video native FPS count. The successive frames in which the robot performed active echolocation are then computed by taking the reading points listed in the YAML file generated by the robot, dividing them by the robot's recorded audio sample rate (i.e. 176.4 kHz), and multiplying them by the video FPS count. Adding the start frame index to each of the values found gives an array containing the desired frame indices. The frame indices were interpolated by including three more frames extracted from the GoPro footage if two consecutive entries in the array had a difference greater than ten. For each of the video frames collected, the following operations are then applied:

1. **Robot's position retrieval:** the position of the robot in the frame is retrieved as the coordinates of the centre of its ArUco marker. The tuple containing the coordinates is then appended to an array, which will be exported as the trajectory the robot followed during its exploration at the end of the run.
2. **Audio re-analysis:** the corresponding multichannel audio file portion is read again and processed with the same parameters used during the run, to obtain the same information that the robot exploited during its navigation.
3. **Ground truth comparison and errors computation:** if the robot sensed an obstacle in that video frame, a ground truth position value is retrieved. This information is computed starting from the data generated by the robot, since an a-

priori ground truth individuation lead to too many misleading individuations made by the tracking code, which in turn gave a non-plausible representation of the robot's performance.

For each obstacle its distance and direction with respect to the robot is computed. In particular, the reference point on the robot is the centre of the microphone array, and the distance from the robot to the obstacles is also reduced by 3.2 cm, which is the radius of the beer bottles used. Obstacles which are placed outside of the microphone array's FOV, i.e. outside the range $\theta \in [-90^\circ, 90^\circ]$ are immediately discarded. For each obstacle, a combined echolocation error is computed according to:

$$\varepsilon_{tot} = \sqrt{\varepsilon_d^2 + (\gamma \cdot \varepsilon_\theta)^2} \quad (5.1)$$

where ε_{tot} is the combined error, ε_d is the distance error expressed in centimetres, and ε_θ is the angle error expressed in degrees. The coefficient γ is equal to $0.5 \text{ cm}^\circ{}^{-1}$, and it is needed to adjust the weight of the angle error with respect to the distance error. The distance and angle error are defined as the difference between the coordinates estimated by the robot through echolocation and the proposed ground truth coordinates which express the actual position of the obstacle. In formulas, the two errors are expressed by:

$$\varepsilon_d = \bar{d} - d_{GT} \quad (5.2)$$

and

$$\varepsilon_\theta = \bar{\theta} - \theta_{GT} \quad (5.3)$$

where the subscript $_{GT}$ indicates the proposed ground truth coordinates, while the barred symbols indicate the robot's estimation. The obstacle whose position is labelled as the ground truth is the one with the lowest combined error among all suitable obstacles.

The combined error involved can be seen as the norm of a vector that exist in the 2-dimensional space spanned by the distance error, and the weighted angle error. The ground truth individuation process then searches the vector in the so-defined space with the lowest norm among all the ones given by the position of the obstacles, with respect to the location pointed at by the Ro-BAT. This process ensures that a plausible obstacle with respect to the position estimated by the robot is selected as the ground truth, so that the echolocation performance of the robot can be more reliably assessed.

Once the ground truth coordinates are established, the two values (i.e. range and

angle) are stored in a YAML file, together with the corresponding computed errors.

4. **Frame post processing:** the trajectory of the robot is highlighted by a green line traced by connecting all the consecutive robot's positions. For the frames in which an obstacle is detected, a yellow arrow pointing to the location computed by the robot and a cyan arrow pointing to the ground truth position are also drawn.

The resulting post-processed videos are available at [2].

5.3. Quantitative Results

The information extracted from each obstacle-avoidance trial is combined into one large dataset, containing the ground truth for what concerns the obstacle positions together with the echolocation errors, and the trajectories that the robot followed during its autonomous navigation. The results will be presented in the following sections.

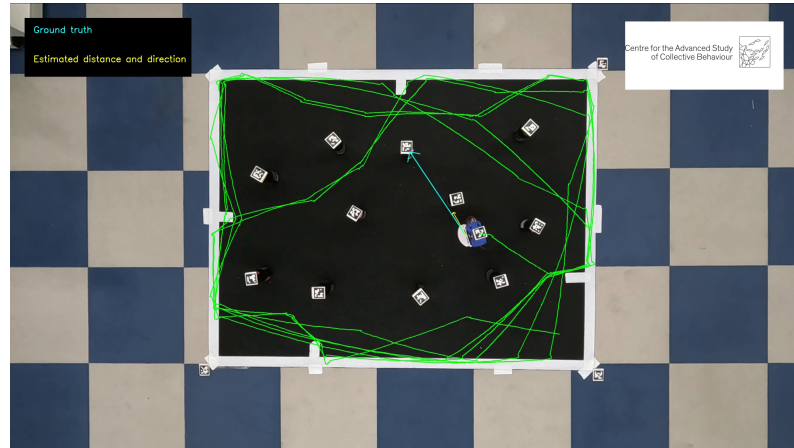
5.3.1. Echolocation Error

To better represent the real capabilities of the robot, an operation of outlier removal was necessary after the collection of the echolocation errors. I attribute the generation of outliers in the data acquired mainly to two factors, one related to the tracking system employed, and the other to the robot's signal processing itself.

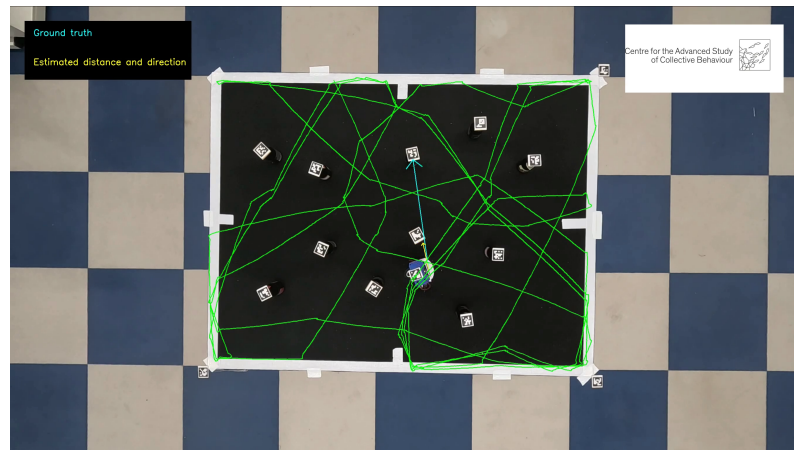
The first cause for the presence of outliers in the collected data is given by rare cases in which the robot locates an obstacle at the extremes of its field of view, but according to the tracking system the centre of the marker on the obstacle is already behind the position of the microphone array. In this cases the yellow arrow indicating the result of the echolocation process performed by the robot and the cyan arrow indicating the ground truth retrieved by the tracking system point to two completely different positions in the arena. Despite that, from Fig. 5.2 it is evident how the robot located an actual obstacle on its path.

The second cause of outliers generation is the presence of other peaks, outside of the maximum, in the envelope of the autocorrelation function of the emitted call that are treated by the robot as echoes. A safety margin after the autocorrelation function of the call is taken for the determination of the onset of the echoes, but in some cases there are still peaks belonging to the emitted call's autocorrelation function that escape the safety window and are treated by the robot as reflections caused by nearby obstacles. To make it simple, the robot tries to locate its own call rather than an actual echo. This is evident by looking at Fig. 5.3, in which the yellow arrow points straight in front of the robot

at a distance of about 12.5 cm, while the tracking system still points the cyan arrow at the object with the lowest combined error, which is in a completely different spot in the arena.

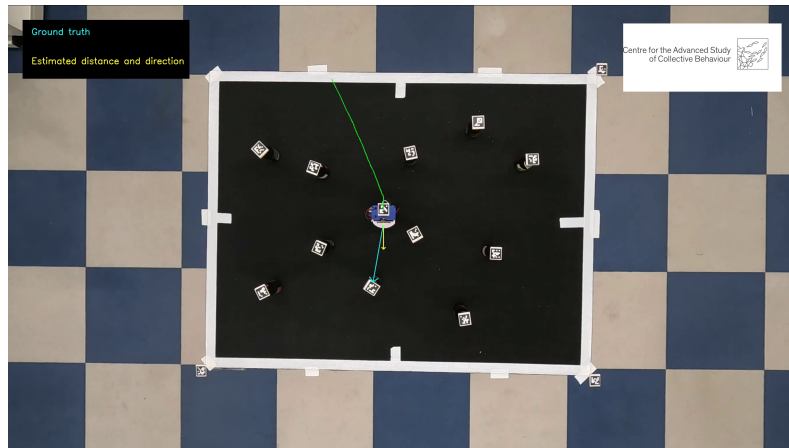


(a)

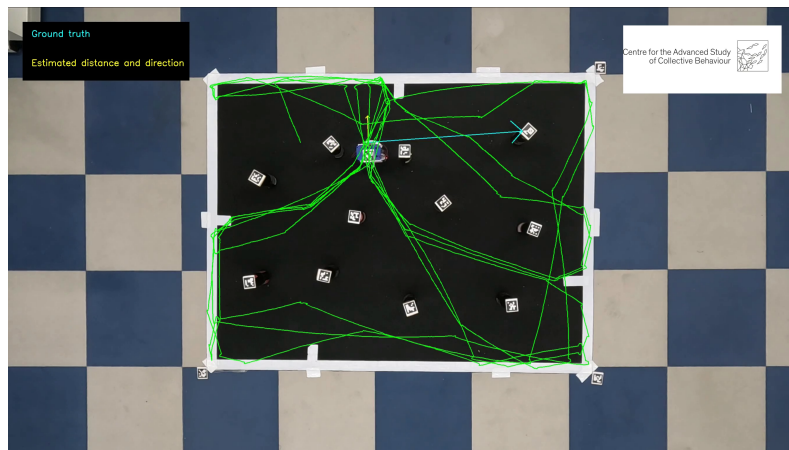


(b)

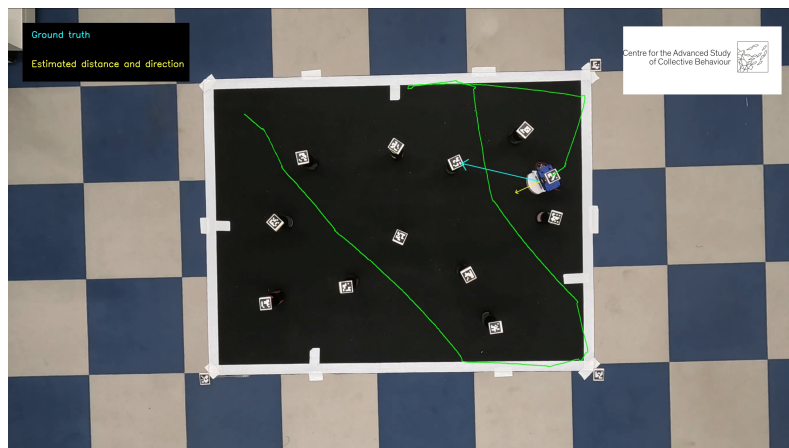
Figure 5.2: Examples of misleading ground truth retrieval by the tracking system. The Ro-BAT clearly points at an actual obstacle placed on its side, but since the centre of the marker is already behind the microphone array according to the tracking system, the next ground truth candidate is selected, resulting in an inflated distance error.



(a)



(b)



(c)

Figure 5.3: Examples of fake positive detections due to peaks related to the call envelope itself, rather than the echo. These instances of incorrect obstacle detection can be distinguished by the location of the detected obstacle, given by the yellow arrow: it is placed around 12.5 cm in front of the robot, aligned with the 0° direction.

The outlier identification and removal process is then based on the distance error value: all the data points corresponding to a distance error greater than 10 cm are treated as outliers, and are not considered for further analyses. The angle data referred to the outlier obstacles is also not considered. Fig. 5.4 shows the echolocation errors in terms of distance and angle as functions of each of the two variables, highlighting with two different colours the points that contributed to a more deep analysis of the performance of the system, and those that were considered as outliers and discarded. The discarded data points are 0.8% of the total detections performed by the robot.

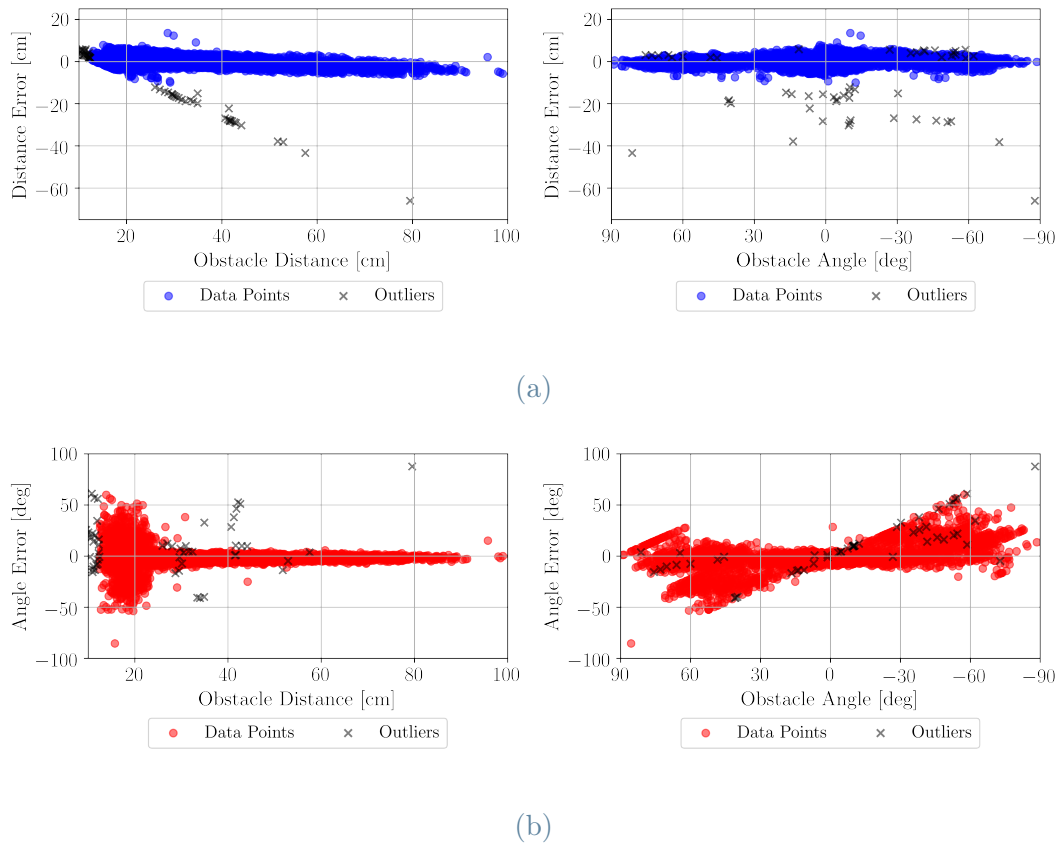


Figure 5.4: Echolocation errors distributions. The outliers, which were not considered in further analyses, are shown as black crosses. The data points are obtained collecting all the runs results in a single dataset. The distance error is contained in a thin distribution oscillating around 0 cm, outside of the outlier points. The angle error is more spread, especially for closer obstacles, and at the extremes of the field of view of the array.

The data points kept are used to estimate a distribution of the echolocation error of the robot, both in terms of range from the obstacles and direction from which the echoes come back to the robot. These distributions are also plotted in Fig. 5.5 as a function of

both spatial variables, to give a more indicative picture of the robot’s capabilities.

The statistics obtained for the two errors are reported in Tab. 5.2.

Error	Mean	Median	Standard Deviation
Distance	0.00 cm	−0.01 cm	2.24 cm
Angle	−1.51°	−2.26°	10.44°

Table 5.2: Echolocation error statistics.

The results indicate that the robot can establish the relative position of the nearest obstacle with a good level of accuracy, both in terms of distance and angle, and within an acceptable tolerance range. Looking at Fig. 5.5b, and comparing it with Tab. 5.2 it is interesting to notice how the angle error mean with respect to the obstacle range is biased towards the left side of the robot, albeit being pretty close to 0°. This means that the robot tends to slightly underestimate the obstacle’s angle when it is placed on its left side, while it overestimates it when the obstacle is located on its right. This might be a consequence of a slightly de-centred placement of the microphone array on the robot.

Obstacles placed towards the extremes of the array’s field of view (FOV) are located less accurately, instead. This behaviour is expected, since the acoustic radiation is concentrated towards the axis of the transducer, meaning that echoes coming from sharp angles will have a lower signal-to-noise ratio, making it difficult for the robot to separate them from the noise floor. Another significant finding is related to the dispersion of the angle error around its mean within the distance range 12.5 cm - 30 cm. In this region it is more dispersed than for greater obstacle distances. By manually reviewing the experiments’ processed footage it is possible to see how the robot seems to estimate the position of obstacles after a turning motion with a sort of delay: it may be possible that the robot reacts to an echo recorded in a certain position while in the meantime its orientation in space has changed. Also, there is a general trend in Uniform Linear Arrays (ULAs) regarding the relationship between the source-to-array distance and the angular resolution, reported and investigated also in [8]. For equal source positions, a shorter distance from the array corresponds to a greater angle with respect to the array axis, leading to a worsening of the resolution and consequent angular accuracy, especially in this case where other nearby objects cause echoes which are in turn captured by the microphone array.

Regarding the distance error reported in Fig. 5.5a, once again the overall distribution both against the ground truth position and the ground truth obstacle angle proves that the system is capable of locating obstacles at various distances, even though with a pro-

gressive decrease in the mean error at larger distances. A negative distance error means that the robot underestimates the range in which the obstacle is located, confirming the conservative approach adopted in the distance estimation. Regarding the relationship between the distance error and the ground truth obstacle direction, its trend oscillates around 0 cm. The general trend is confirmed in this case too: the Ro-BAT works best when localizing obstacles that are close to the direction corresponding to 0° , because most of the energy of the emission is concentrated there.

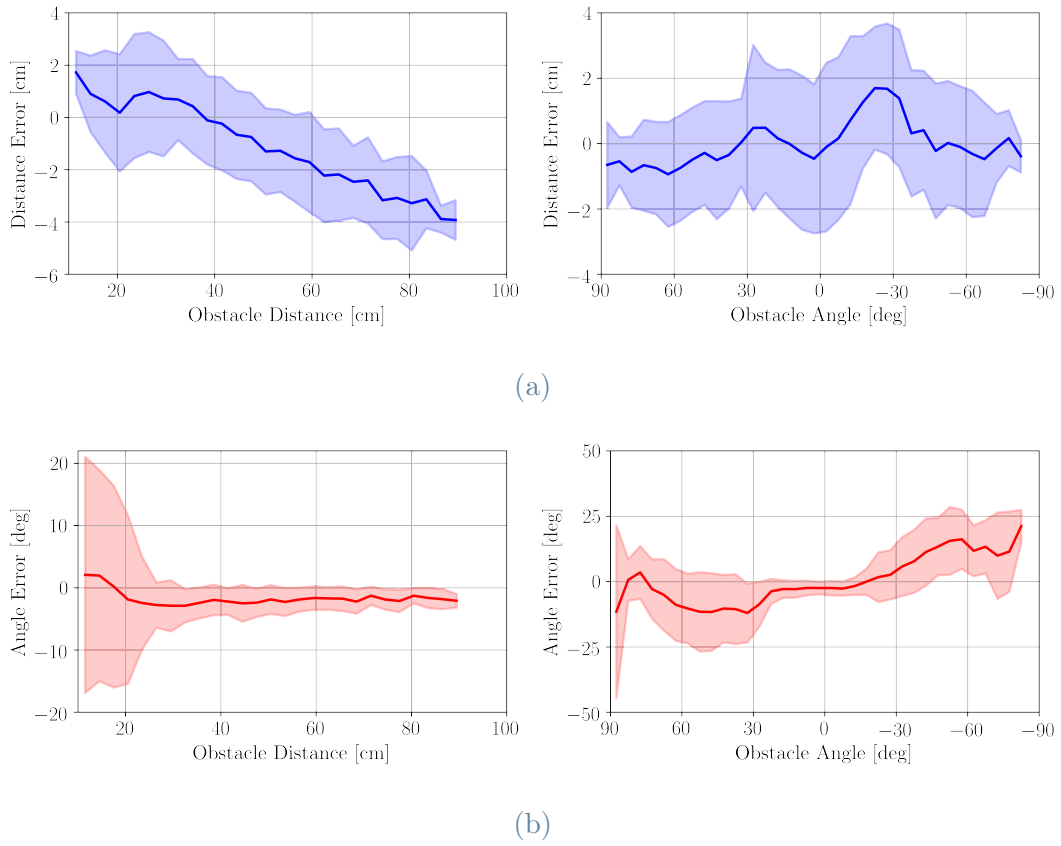


Figure 5.5: Echolocation errors statistics, after the outliers removal operation. Looking at the distance error plots, the robot tends to overestimate the range for closer obstacles, while it underestimates it for objects placed farther. The mean angle error is instead stably positioned at about -1.5° , meaning that the Ro-BAT tends to always locate obstacles slightly more towards the left than they are in reality. For obstacles placed at the very sides of the robot, instead, the localisation accuracy decreases significantly, as the robot tends to estimate their position as shifted closer to its centre.

The resulting kernel density functions are reported in the violin plots of Fig. 5.6. The distance error is well distributed around the 0 cm line, meaning that in general overesti-

mations and underestimations of the obstacle range compensate one another during the robot's navigation. This can be easily explained by the fact the robot in general tends to underestimate its distance from the nearest obstacle, but since closer obstacles are sensed with much more frequency than very far ones, the mean results close to zero.

For the angle error, instead, the situation is slightly different. Most of its density function is concentrated below 0° , meaning that the robot tends to estimate the direction of each obstacle slightly more towards the left than it is in reality. This confirms the trend seen in previous plots. The angle error is also much more dispersed around its mean value, another confirmation that the DOA estimation based on the echoes recorded is a delicate process, which can be subject to non negligible errors under certain conditions, despite still working fine for the majority of the cases. My hypothesis on the reason of this left-side bias is that the array is not perfectly aligned with the robot's axis: the 3D-printed case is composed of two separate stages placed one on top of the other, without any tight locking mechanism. The top stage may have slid on the bottom stage slightly towards the right hand side of the robot in most of the runs, causing this misalignment between the ground truth and the robot's estimations.

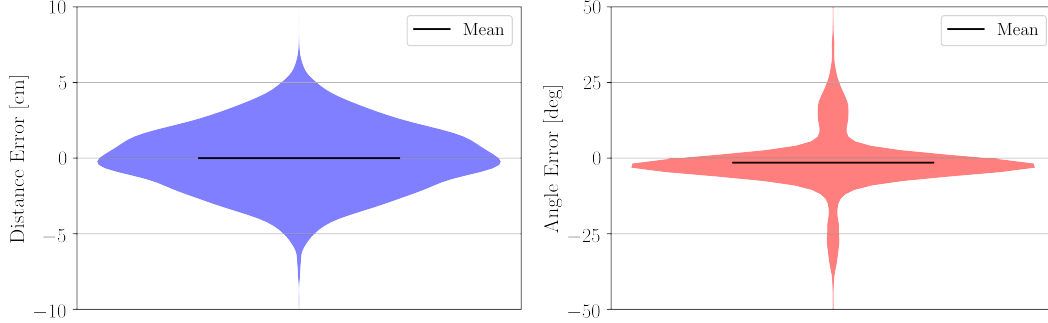


Figure 5.6: Kernel density functions of the echolocation errors. The distance error density is depicted on the left, while the angle error is plotted on the right. In the distance error distribution, the overestimations of the position of closer obstacles are balanced by the underestimations of farther obstacles, which happen less frequently. The resulting mean error is very close to being null. The angle error is much more spread, instead, and it is concentrated around -1.51° , indicating a polarisation in the estimates made by the robot. This may be explained by the positioning of the microphone array slightly towards the right of the robot's axis.

5.3.2. Robot Trajectory Analysis

The other metric used to evaluate the performance of the robot is an analysis on its trajectory while it explores the arena, to see the effect that the presence of obstacles has on the robot's navigation. The rationale behind this analysis is to verify the actual echolocation and reaction process exhibited by the robot, and to exclude that the robot reacts only to the white lines that trace the arena boundaries. The trajectories are extracted by the video frames as an array of coordinates of the centre of the marker placed on the robot. Then, the arena surface is divided into a grid with squared cells with edge length equal to the radius of the beer bottles used, i.e. 3.2 cm. Heatmaps are then generated by computing a two-dimensional histogram with the robot's trajectory coordinates. In total, four heatmaps are generated by the four different obstacles configurations, to which two more heatmaps are added as a control case by letting the robot explore the arena empty of obstacles. The resulting heatmaps can be seen in Fig. 5.7.

What is seen in the heatmaps is that the robot breaks its trajectory in many small rectilinear movements in presence of obstacles, while instead in the case of an empty arena it moves along long, almost straight lines until the edges of the arena are touched, at which point the robot turns to another direction. The slight curvature in the trajectories, especially in the empty arena, is given by slightly bent axle shafts in the Thymio II robot used for the experiments.

It is observable that the robot often repeats the same trajectories for each obstacle configuration. This proves the robot's capability of moving reliably along collision free paths. Furthermore, the trajectories followed by the robot approximate to a certain extent the tessellation lines that would be generated by a Voronoi partition [1] of the arena's surface. This means that the robot is often capable of finding routes that maximise the distance from the closest obstacles encountered, by only localising the nearest obstacle each time thanks to the echoes caused by its emission. The Voronoi partitions for each obstacle configuration are overlapped to the heatmaps in Fig. 5.7 for reference.

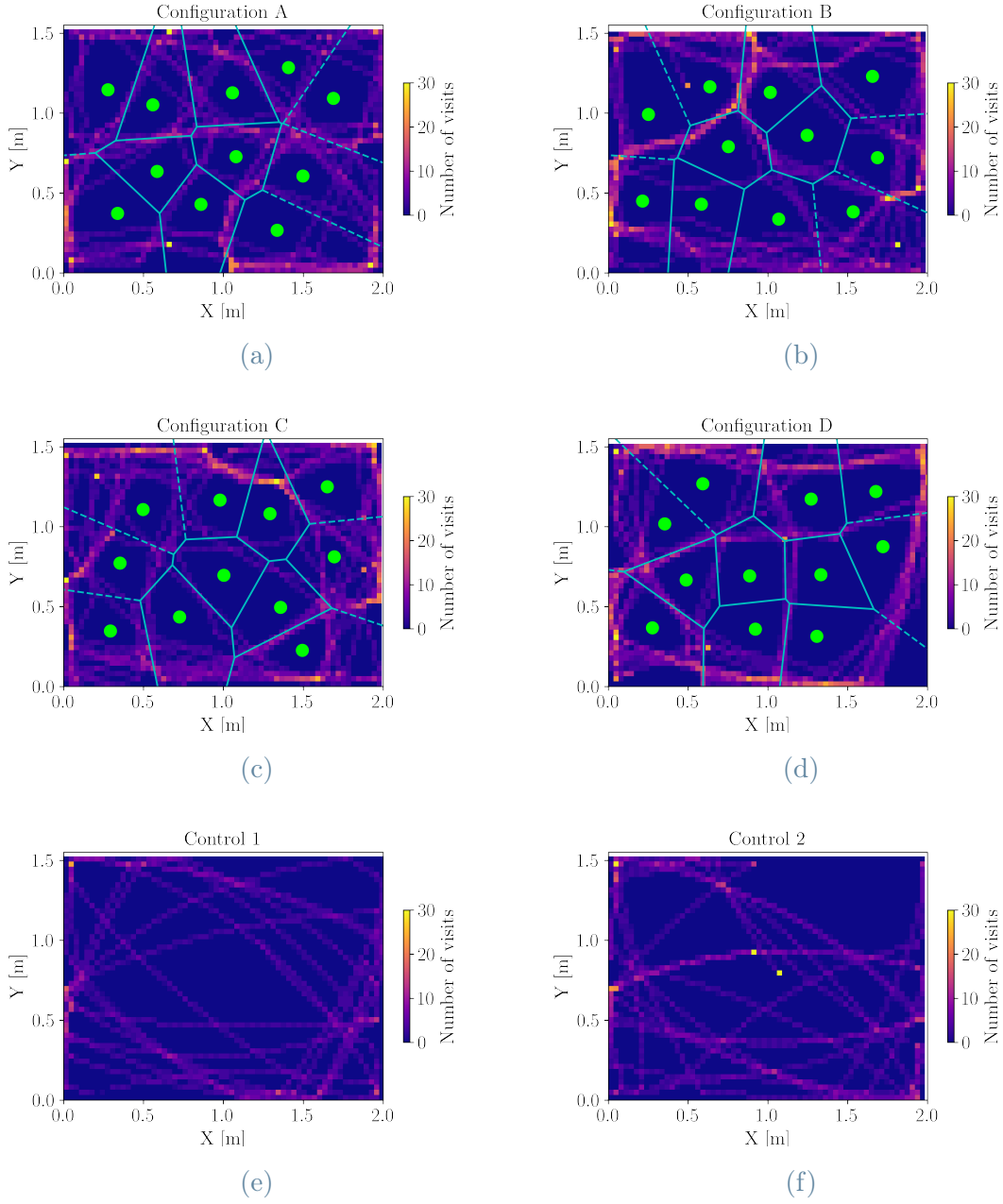


Figure 5.7: Robot's trajectory heatmaps. From (a) to (d) heatmaps are generated by combining the trajectories obtained in both runs for each configuration. The obstacle positions are highlighted in the plots. The Voronoi partitions of the arena's surface are also reported for each obstacle configuration. The two control runs are reported in (e) and (f). The duration of the two control runs is respectively 10 min 4 sec and 11 min 8 sec.

5.4. Future Directions

The results presented in the previous Section show that the active sensing Ro-BAT is capable of navigating autonomously a controlled environment, avoiding obstacles sensed exclusively through echolocation. While being generally successful in such task, the robot proposed in this thesis presents two areas of improvement.

The first one is related to its sensorimotor rate, and how the reaction to obstacles influences the sensing capabilities of the system. When moving on a straight line, the robot has a steady update rate of about 7 Hz, which is an acceptable value for a sensor deployed in real time scenarios, but it could definitely be improved for a more detailed scanning of the environment, especially if the robot moves at increased velocities or it should face a dynamic scenario in which obstacles also move at a considerable speed with respect to the robot. When the robot has to turn to avoid a collision, it momentarily becomes completely blind to the environment, since the only way to reliably synchronise a turning motion in place was to set the wheels velocities to the values giving the desired rotation, and then make the system sleep until the desired angle was covered by the rotation of the robot. The execution is then resumed setting the wheels velocity value for a rectilinear motion, and all the sensory-motor loop is then repeated again. This very same strategy is also employed to contain the robot within the boundaries of the arena: while the robot steers back towards the inside of the arena, it cannot sense the obstacles around itself until it resumes its rectilinear motion. This leads to a notable decrease in the robot's update rate in proximity of an obstacle which could cause a collision, which was compensated in the robot's behaviour with a large angle of turning.

The cause of this lower than desired update rate is the sum of all the execution times of the blocks of code which perform the necessary operations for the robot's navigation. Around 50% of the total loop time is consumed by the DOA algorithm, and another operation which has a heavy impact on the robot's performance is the file access operation to append new audio data to the audio file needed for the post-processing and analysis, and also the reading of the same file to extract audio data to be elaborated. If there is no need for the robot to keep track of the whole experiment, then these time consumptions could be avoided. The emission latency, instead, is hardware dependent and cannot be lowered without modifications on the devices' drivers involved. An implementation of the code written in a programming language closer to the machine representation than Python could definitely speed up the computation, with a benefit on the overall update rate. Even with an optimised code, the bottleneck would still remain the emission latency, that is the time taken by converting the signal from digital to an analog representation and

the subsequent amplification and emission, which is around 15 ms for the system built.

Another point which could be greatly improved about the active sensing Ro-BAT is its form factor, in relation to its echolocation capabilities. The asymmetry introduced by the cables on the right side of the robot caused scenarios in which the robot could not avoid a lateral collision with obstacles on its right side: as a matter of fact, runs A.1 and B.1 were ended abruptly by the robot taking down a bottle with the protrusion on its right. The collision instants are reported in Fig. 5.8. Conversely, no collisions were recorded frontally or on the left side of the robot, indicating that overall the robot is capable of avoiding obstacles based purely on echolocation. A solution to this undesired outcome would be a more refined mapping of the position of the obstacles with respect to the robot's shape and extension on its sides, or a refactoring of the robot's build which grants symmetry, and so a more manageable shape to work with.

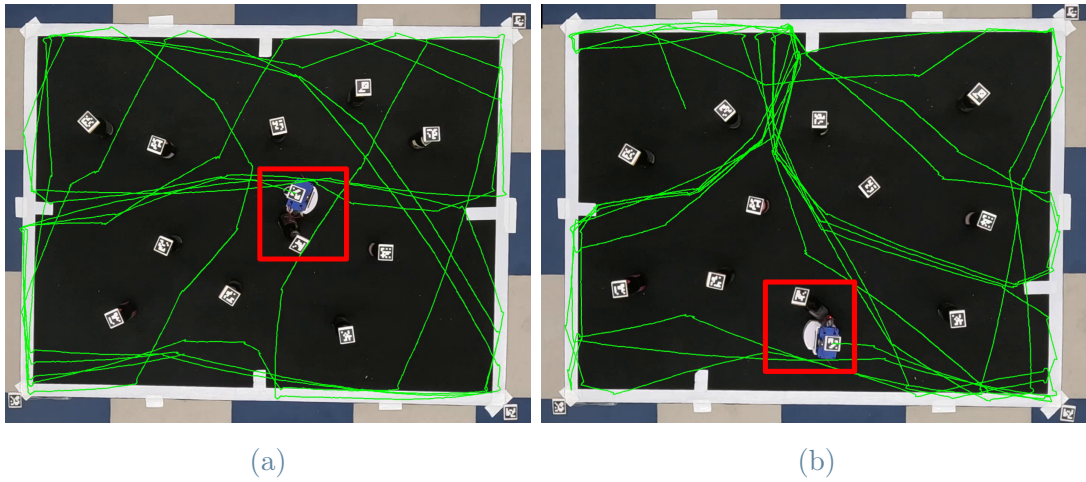


Figure 5.8: Collisions reported due to the Ro-BAT form factor. The collision in (a) was caused the robot echolocating its own call and turning towards the bottle hit, which was then in a position outside of the FOV of the robot. The collision shown in (b) was caused instead by an initial detection of the obstacle by the robot and a subsequent rotation, which at that point put the bottle in a position which was not able to be sensed by the robot.

6 | Conclusions and Future Developments

The aim of this thesis was to equip a small robot with the necessary hardware to make it perform active echolocation, and to design a control system which can make use of the echoes generated by the robot's emission to navigate autonomously a controlled environment without collisions. With this aim in mind, a suitable ultrasonic source consisting of an electrostatic transducer was set up in conjunction with already tested audio equipment for small robotic applications, such as the MCHStreamer audio interface. A Uniform Linear microphone Array specialised for ultrasound recording was also designed, produced and deployed on the robot as the audio receiving device. The built hardware parts were then characterised acoustically to understand how their properties affected the overall echolocation capabilities of the system. The necessary code to make the robot perform echolocation was also implemented: a ranging mechanism which detects the distance from the robot to the nearest obstacle, and also Direction of Arrival estimation algorithms were exploited to complete the localisation process based on the echoes recorded by the robot. In particular, three different DOA estimation algorithms were adapted to wide-band sources and tested in static conditions to establish which one was the most reliable for real-time deployment. The three algorithms developed were Delay and Sum (DAS), Capon method, and MUSIC. Finally, DAS was included in the signal processing chain because of its lower execution time, while offering the same localisation accuracy of Capon and MUSIC for this application. Despite the higher spatial resolution offered by Capon and MUSIC, their higher complexity and increased computation time with respect to DAS lead to their exclusion from further testing.

The robot was tested in an arena with reflecting obstacles, and its obstacle avoidance behaviour during autonomous navigation was demonstrated to be successful. The robot was able to locate most of the obstacles with a good degree of accuracy, despite the limitations imposed by the high directivity of the ultrasonic transducer. A non-optimised form factor of the robot lead to the possibility of encountering lateral collisions during the robot's navigation, which would have been otherwise not recorded.

Overall, this thesis demonstrates the feasibility of building cost-effective small echolocating robots with mostly consumer-grade hardware equipment, with the larger objective to scale the proposed solutions to multiple robots to realise a robotic model of a swarm of active sensing agents. This flexible swarm could then be exploited by biologists and engineers to deepen the understanding of the cocktail party problem. This sensory challenge is experienced in groups of both social animals which base their environment perception on echolocation, such as many bats species, and groups of decentralised robots who exploit audio for communication and coordination. The greater objective in the long run is therefore to fuse active echolocation and passive sound source localisation, which is expected to be a key component in a swarm of active sensing agents.

In the field of robotics, echolocation represents a cost-effective sensing strategy which could complement or substitute vision-based navigation in environments where the visibility is compromised due to fog, dust, or lack of light.

Future developments of the robotic platform proposed could include an experimentation of the works proposed in [20] and [28], to respectively widen the directivity pattern of the ultrasonic transducer employed, and to raise the frequency range in which the microphone array can operate without introducing spatial aliasing. These two modifications would make the robotic platform more similar to the acoustic characteristics of actual bats, since they exhibit a wide radiation pattern, and are able to produce and listen back to frequencies that are much higher than the maximum frequency used by the Ro-BAT. Also, different array geometries and more refined DOA estimation algorithms could be explored in conjunction with an improved processing pipeline, in order to be able to handle more than one reflector at a time efficiently and robustly. Object classification based on the echoes features might be another interesting development for this project.

Bibliography

- [1] F. Aurenhammer. Voronoi diagrams—a survey of a fundamental geometric data structure. *ACM Computing Surveys*, 23(3):345–405, Sept. 1991. ISSN 0360-0300, 1557-7341. doi: 10.1145/116873.116880. URL <https://dl.acm.org/doi/10.1145/116873.116880>.
- [2] G. Baroli. Ro-BAT Project: active sensing Ro-BAT lab tests., June 2025. URL <https://zenodo.org/records/15767326>.
- [3] B. Bechtold. python-soundfile. <https://pysoundfile.readthedocs.io/>, 2023. Version 0.12.1.
- [4] M. A. Bee and C. Micheyl. The cocktail party problem: What is it? How can it be solved? And why should animal behaviorists study it? *Journal of Comparative Psychology*, 122(3):235–251, 2008. ISSN 1939-2087. doi: 10.1037/0735-7036.122.3.235.
- [5] T. Beleyur and H. R. Goerlitz. Modeling active sensing reveals echo detection even in large groups of bats. *Proceedings of the National Academy of Sciences*, 116(52):26662–26668, Dec. 2019. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.1821722116. URL <https://pnas.org/doi/full/10.1073/pnas.1821722116>.
- [6] G. Bradski. The OpenCV Library. *Dr. Dobb’s Journal of Software Tools*, 2000.
- [7] J. Capon. High-resolution frequency-wavenumber spectrum analysis. *Proceedings of the IEEE*, 57(8):1408–1418, 1969. ISSN 0018-9219. doi: 10.1109/PROC.1969.7278. URL <http://ieeexplore.ieee.org/document/1449208/>.
- [8] A. Cigada, F. Ripamonti, and M. Vanali. The delay & sum algorithm applied to microphone array measurements: Numerical analysis and experimental validation. *Mechanical Systems and Signal Processing*, 21(6):2645–2664, Aug. 2007. ISSN 08883270. doi: 10.1016/j.ymssp.2007.01.001. URL <https://linkinghub.elsevier.com/retrieve/pii/S0888327007000052>.
- [9] A. Doimo. RO-BAT: A bat-inspired approach on mobile robot navigation using

- Direction of Arrival estimation - Thesis. Master's thesis, Politecnico di Milano, 2024. URL <https://hdl.handle.net/10589/231608>.
- [10] I. Eliakim, Z. Cohen, G. Kosa, and Y. Yovel. A fully autonomous terrestrial bat-like acoustic robot. *PLOS Computational Biology*, 14(9):e1006406, Sept. 2018. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1006406. URL <https://dx.plos.org/10.1371/journal.pcbi.1006406>.
- [11] FFmpeg Developers. Ffmpeg. <https://ffmpeg.org/>, 2000–2025.
- [12] H. Friis and C. Feldman. A Multiple Unit Steerable Antenna for Short-Wave Reception. *Proceedings of the IRE*, 25(7):841–917, July 1937. ISSN 0096-8390. doi: 10.1109/JRPROC.1937.228354. URL <http://ieeexplore.ieee.org/document/1686435/>.
- [13] S. Garrido-Jurado, R. Muñoz-Salinas, F. Madrid-Cuevas, and M. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, June 2014. ISSN 00313203. doi: 10.1016/j.patcog.2014.01.005. URL <https://linkinghub.elsevier.com/retrieve/pii/S0031320314000235>.
- [14] M. Geier. python-sounddevice. <https://python-sounddevice.readthedocs.io/>, 2024. Version 0.5.1.
- [15] S. Greif and B. M. Siemers. Innate recognition of water bodies in echolocating bats. *Nature Communications*, 1(1):107, Nov. 2010. ISSN 2041-1723. doi: 10.1038/ncomms1110. URL <https://www.nature.com/articles/ncomms1110>.
- [16] S. Greif, S. Zsebők, D. Schmieder, and B. M. Siemers. Acoustic mirrors as sensory traps for bats. *Science*, 357(6355):1045–1047, Sept. 2017. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.aam7817. URL <https://www.science.org/doi/10.1126/science.aam7817>.
- [17] E. M. Group. thymio-python. <https://github.com/epfl-mobots/thymio-python>, 2020. Version 0.1.2.
- [18] G. C. Herring, A. J. Zuckerwar, and B. R. Elbing. Free-field calibration of the pressure sensitivity of air-condenser, electret, MEMS, and piezoresistive microphones at frequencies up to 80 kHz. *The Journal of the Acoustical Society of America*, 116(4_Supplement):2512–2512, Oct. 2004. ISSN 0001-4966, 1520-8524. doi: 10.1121/1.4785022. URL https://pubs.aip.org/jasa/article/116/4_Supplement/2512/536679/Free-field-calibration-of-the-pressure-sensitivity.

- [19] T. K. Kaiser and H. Hamann. Innate Motivation for Robot Swarms by Minimizing Surprise: From Simple Simulations to Real-World Experiments. *IEEE Transactions on Robotics*, 38(6):3582–3601, Dec. 2022. ISSN 1941-0468. doi: 10.1109/TRO.2022.3181004. URL <https://ieeexplore.ieee.org/document/9813360/>.
- [20] R. Kerstens, D. Laurijssen, W. Daems, and J. Steckel. Widening the Directivity Patterns of Ultrasound Transducers Using 3-D-Printed Baffles. *IEEE Sensors Journal*, 17(5):1454–1462, Mar. 2017. ISSN 1558-1748. doi: 10.1109/JSEN.2016.2642303. URL <https://ieeexplore.ieee.org/document/7792174/>.
- [21] R. Kerstens, D. Laurijssen, and J. Steckel. Low-cost one-bit MEMS microphone arrays for in-air acoustic imaging using FPGA’s. In *2017 IEEE SENSORS*, pages 1–3, Glasgow, Oct. 2017. IEEE. ISBN 978-1-5090-1012-7. doi: 10.1109/ICSENS.2017.8234087. URL <http://ieeexplore.ieee.org/document/8234087/>.
- [22] R. Kerstens, D. Laurijssen, and J. Steckel. eRTIS: A Fully Embedded Real Time 3D Imaging Sonar Sensor for Robotic Applications. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 1438–1443, Montreal, QC, Canada, May 2019. IEEE. ISBN 978-1-5386-6027-0. doi: 10.1109/ICRA.2019.8794419. URL <https://ieeexplore.ieee.org/document/8794419/>.
- [23] Knowles. *SPH0641LU4H-1 PDM Output Digital Microphone datasheet*, June 2014. URL https://www.mouser.de/datasheet/2/218/know_s_a0010889928_1-2271691.pdf. Rev. A.
- [24] Knowles. *SPH0645LM4H-B I2S Output Digital Microphone datasheet*, Feb. 2017. URL <https://mm.digikey.com/Volume0/opasdata/d220001/medias/docus/908/SPH0645LM4H-B.pdf>. Rev. C.
- [25] K. Kroening. *ffmpeg-python: Python bindings for FFmpeg*, 2025. URL <https://github.com/kkroening/ffmpeg-python>. Version 0.2.0.
- [26] H. Kuttruff. *Acoustics: An Introduction*. CRC Press, London, Nov. 2006. ISBN 978-0-367-80769-6. doi: 10.1201/9780367807696.
- [27] D. Laurijssen, R. Kerstens, G. Schouten, W. Daems, and J. Steckel. A Flexible Low-Cost Biologically Inspired Sonar Sensor Platform for Robotic Applications. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 9617–9623, Montreal, QC, Canada, May 2019. IEEE. ISBN 978-1-5386-6027-0. doi: 10.1109/ICRA.2019.8794165. URL <https://ieeexplore.ieee.org/document/8794165/>.
- [28] D. Laurijssen, W. Daems, and J. Steckel. Broadband MEMS Microphone Arrays

- with Reduced Aperture Through 3D-Printed Waveguides, June 2024. URL <http://arxiv.org/abs/2406.07663>. Issue: arXiv:2406.07663 arXiv: 2406.07663 [eess].
- [29] R. P. Ltd. Buy a Raspberry Pi 4 Model B. URL <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>.
- [30] C. B. Mansour, E. Koreman, and D. Laurijssen. Robotic models of obstacle avoidance in bats.
- [31] G. F. McCracken and M. F. Gassel. Genetic Structure in Migratory and Nonmigratory Populations of Brazilian Free-Tailed Bats. *Journal of Mammalogy*, 78(2): 348–357, 1997. ISSN 0022-2372. doi: 10.2307/1382888. URL <https://www.jstor.org/stable/1382888>. Publisher: [American Society of Mammalogists, Oxford University Press].
- [32] B. McFee, M. McVicar, D. Faronbi, I. Roman, M. Gover, S. Balke, S. Seyfarth, A. Malek, C. Raffel, V. Lostanlen, B. v. Niekirk, D. Lee, F. Cwitkowitz, F. Zalkow, O. Nieto, D. Ellis, J. Mason, K. Lee, B. Steers, E. Halvachs, C. Thom  , F. Robert-St  ter, R. Bittner, Z. Wei, A. Weiss, E. Battenberg, K. Choi, R. Yamamoto, C. J. Carr, A. Metsai, S. Sullivan, P. Friesch, A. Krishnakumar, S. Hidaka, S. Kowalik, F. Keller, D. Mazur, A. Chabot-Leclerc, C. Hawthorne, C. Ramaprasad, M. Keum, J. Gomez, W. Monroe, V. A. Morozov, K. Eliasi, nullmightybofo, P. Biberstein, N. D. Sergin, R. Hennequin, R. Naktinis, beantowel, T. Kim, J. P.   sen, J. Lim, A. Malins, D. Here    , S. v. d. Struijk, L. Nickel, J. Wu, Z. Wang, T. Gates, M. Vollrath, A. Sarroff, Xiao-Ming, A. Porter, S. Kranzler, Voodoohop, M. D. Gangi, H. Jinoz, C. Guerrero, A. Mazhar, toddrme2178, Z. Baratz, A. Kostin, X. Zhuang, C. T. Lo, P. Campr, E. Semeniuc, M. Biswal, S. Moura, P. Brossier, H. Lee, and W. Pimenta. librosa/librosa: 0.10.2.post1, May 2024. URL <https://zenodo.org/records/11192913>.
- [33] D. Mezey, R. Bastien, Y. Zheng, N. McKee, D. Stoll, H. Hamann, and P. Romanczuk. Purely vision-based collective movement of robots. *npj Robotics*, 3(1):11, June 2025. ISSN 2731-4278. doi: 10.1038/s44182-025-00027-2. URL <https://www.nature.com/articles/s44182-025-00027-2>.
- [34] miniDSP. MCHStreamer - USB Audio Swiss-Army knife - Multi-channel PCM/DSD/PDM/ADAT/TDM/SPDIF interface. URL https://www.minidsp.com/products/usb-audio-interface/mchstreamer?srsltid=AfmB0oqF4uG7Mrwtn10zdJ2CAVGLGPD3-HdTzuc5tYGUnzC-jE1E9_ss.

- [35] miniDSP Ltd. *MCHStreamer Kit User Manual*, Jan. 2024. URL <https://www.minidsp.com/images/documents/MCHStreamer%20User%20Manual.pdf>. Rev. 1.7.
- [36] Mobsya. Home. URL <https://www.thymio.org/>.
- [37] F. Mondada, M. Bonani, F. Riedo, M. Briod, L. Pereyre, P. Retornaz, and S. Maguenat. Bringing Robotics to Formal Education: The Thymio Open-Source Hardware Robot. *IEEE Robotics & Automation Magazine*, 24(1):77–85, Mar. 2017. ISSN 1070-9932. doi: 10.1109/MRA.2016.2636372. URL <http://ieeexplore.ieee.org/document/7859350/>.
- [38] C. F. Moss, C. Cechetto, M. Warnecke, C. Chiu, W. Xian, and B. Falk. Echolocating bats face a cocktail party nightmare when they fly together in cluttered environments. *The Journal of the Acoustical Society of America*, 135(4_Supplement):2150–2151, Apr. 2014. ISSN 0001-4966. doi: 10.1121/1.4876970. URL <https://doi.org/10.1121/1.4876970>.
- [39] U. of Antwerp. Cosys-Lab | Cosys-Lab | University of Antwerp. URL <https://www.uantwerpen.be/en/research-groups/cosys-lab/>.
- [40] U. of Konstanz. Cyber-physical Systems. URL <https://www.cps.uni-konstanz.de/>.
- [41] R. Roy and T. Kailath. ESPRIT-estimation of signal parameters via rotational invariance techniques. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(7):984–995, July 1989. ISSN 0096-3518. doi: 10.1109/29.32276. URL <https://ieeexplore.ieee.org/document/32276>.
- [42] Sanken. SANKEN MICROPHONE CO.,LTD. | Products [CO-100K]. URL <https://sanken-mic.com/en/product/freqpola.cfm/3.1000400>.
- [43] R. Schmidt. Multiple emitter location and signal parameter estimation. *IEEE Transactions on Antennas and Propagation*, 34(3):276–280, Mar. 1986. ISSN 0018-926X, 1558-2221. doi: 10.1109/TAP.1986.1143830. URL <https://ieeexplore.ieee.org/document/1143830/>.
- [44] Senscomp, Inc. *Senscomp Series 7000 Ultrasonic Sensor datasheet*, July 2014. URL https://senscomp.com/wp-content/uploads/2019/12/V2_Series-7000-Ultrasonic-Sensor-Spec-31July14.pdf. Rev. 2014-7-31.
- [45] J. Steckel and H. Peremans. BatSLAM: Simultaneous Localization and Mapping Using Biomimetic Sonar. *PLoS ONE*, 8(1):e54076, Jan. 2013. ISSN 1932-

6203. doi: 10.1371/journal.pone.0054076. URL <https://dx.plos.org/10.1371/journal.pone.0054076>.
- [46] J. Steckel and H. Peremans. Spatial sampling strategy for a 3D sonar sensor supporting BatSLAM. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 723–728, Sept. 2015. doi: 10.1109/IROS.2015.7353452. URL <https://ieeexplore.ieee.org/document/7353452/>.
- [47] J. Steckel, F. Schillebeeckx, and H. Peremans. Biomimetic sonar, outer ears versus arrays. In *2011 IEEE SENSORS*, pages 821–824, Oct. 2011. doi: 10.1109/ICSENS.2011.6126961. URL <https://ieeexplore.ieee.org/document/6126961/>. ISSN: 1930-0395.
- [48] J. Steckel, A. Boen, and H. Peremans. Broadband 3-D Sonar System Using a Sparse Array for Indoor Navigation. *IEEE Transactions on Robotics*, 29(1):161–171, Feb. 2013. ISSN 1552-3098, 1941-0468. doi: 10.1109/TRO.2012.2221313. URL <http://ieeexplore.ieee.org/document/6331017/>.
- [49] G. Su and M. Morf. The signal subspace approach for multiple wide-band emitter location. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 31(6): 1502–1522, Dec. 1983. ISSN 0096-3518. doi: 10.1109/TASSP.1983.1164233. URL <https://ieeexplore.ieee.org/document/1164233/>.
- [50] Texas Instruments. *PCM5102 Audio Stereo DAC datasheet*, July 2012. URL https://www.ti.com/lit/ds/symlink/pcm5102.pdf?ts=1737132261218&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252Fde-de%252FPCM5102. Rev. 2012.
- [51] Texas Instruments. *PCMD3180 Octal-Channel, PDM Input to TDM or I2S Output Converter datasheet*, June 2020. URL https://www.ti.com/lit/ds/symlink/pcmd3180.pdf?ts=1731401592762&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FPCMD3180.
- [52] Texas Instruments. *PCMD3140 Quad-Channel, PDM Input to TDM or I2S Output Converter datasheet*, June 2021. URL https://www.ti.com/lit/ds/symlink/pcmd3140.pdf?ts=1731322734696&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FPCMD3140. Rev. A.
- [53] T. Verellen, R. Kerstens, D. Laurijssen, and J. Steckel. Urtis: a Small 3d Imaging Sonar Sensor for Robotic Applications. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4801–4805, Barcelona, Spain, May 2020. IEEE. ISBN 978-1-5090-6631-5.

- doi: 10.1109/ICASSP40776.2020.9053536. URL <https://ieeexplore.ieee.org/document/9053536/>.
- [54] T. Verellen, R. Kerstens, and J. Steckel. High-Resolution Ultrasound Sensing for Robotics Using Dense Microphone Arrays. *IEEE Access*, 8:190083–190093, 2020. ISSN 2169-3536. doi: 10.1109/ACCESS.2020.3032177. URL <https://ieeexplore.ieee.org/document/9229384/>.
- [55] M. Wahby, J. Petzold, C. Eschke, T. Schmickl, and H. Hamann. Collective Change Detection: Adaptivity to Dynamic Swarm Densities and Light Conditions in Robot Swarms. In *ALIFE 2019: The 2019 Conference on Artificial Life*, pages 642–649, July 2019. doi: 10.1162/isal_a_00233. URL https://doi.org/10.1162/isal_a_00233.
- [56] M. Wax, T.-J. Shan, and T. Kailath. Spatio-temporal spectral analysis by eigenstructure methods. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(4):817–827, Aug. 1984. ISSN 0096-3518. doi: 10.1109/TASSP.1984.1164400. URL <https://ieeexplore.ieee.org/document/1164400/>.
- [57] Y. Yamada, K. Ito, A. Oka, S. Tateiwa, T. Ohta, R. Kobayashi, S. Hiryu, and Y. Watanabe. Obstacle-Avoidance Navigation by an Autonomous Vehicle Inspired by a Bat Biosonar Strategy. In S. P. Wilson, P. F. Verschure, A. Mura, and T. J. Prescott, editors, *Biomimetic and Biohybrid Systems*, pages 135–144, Cham, 2015. Springer International Publishing. ISBN 978-3-319-22979-9. doi: 10.1007/978-3-319-22979-9_14.

List of Figures

2.1	Source impinging from the direction θ on a Uniform Linear Array.	7
3.1	Time trend and spectrogram of the emitted signal.	21
3.2	Spectrogram of the emitted call and first echo.	22
3.3	Example of the ranging process for locating an object placed at 50 cm in front of the robot.	23
3.4	Pseudospectra obtained by applying the DOA estimation algorithms to the echo generated by a cylindrical obstacle placed at a distance of 50 cm (part 1).	28
3.4	(continued) Pseudospectra obtained by applying the DOA estimation algorithms to the echo generated by a cylindrical obstacle placed at a distance of 50 cm.	29
3.5	Pseudospectra obtained by echoes coming from pairs of objects placed in asymmetrical configurations around the robot (part 1).	30
3.5	(continued) Pseudospectra obtained by echoes coming from pairs of objects placed in asymmetrical configurations around the robot.	31
4.1	The active sensing Ro-BAT developed.	34
4.2	The miniDSP MCHStreamer Kit audio interface	36
4.3	The components which constitute the audio source for the Ro-BAT.	37
4.4	The eight-channel microphone array that was already available. The actual microphones are the ones highlighted by the red frame.	38
4.5	The first model of a microphone array for ultrasound recording.	40
4.6	The second and final microphone array model.	41
4.7	Frequency response of the Senscomp Series 7000 transducer, powered by the custom built high-voltage amplifier.	43
4.8	Directivity patterns of the Senscomp Series 7000 transducer as function of frequency and emission angle (part 1).	44
4.8	(continued) Directivity patterns of the Senscomp Series 7000 transducer as function of frequency and receiving angle.	45

4.9	The frequency response in terms of dB RMS of the eight microphones which constitute the array installed on the robot.	46
4.10	Directivity patterns of the Knowles SPH0641LU4H-1 microphones.	48
4.10	(continued) Directivity patterns of the Knowles SPH0641LU4H-1 microphones. Each pattern is computed considering the central frequency $\pm$ 2 kHz.	49
4.11	A representation of the Ro-BAT's "danger zone", as seen by the robot observed from the top.	52
5.1	The experimental setup for recording the robot while it navigates autonomously.	55
5.2	Examples of misleading ground truth retrieval by the tracking system. . . .	59
5.3	Examples of fake positive detections due to peaks related to the call envelope itself, rather than the echo.	60
5.4	Echolocation errors distributions.	61
5.5	Echolocation errors statistics.	63
5.6	Kernel density functions of the echolocation errors.	64
5.7	Robot's trajectory heatmaps.	66
5.8	Collisions reported due to the Ro-BAT form factor.	68

List of Tables

3.1	Mean execution times for the DOA estimation algorithms implemented . .	27
3.2	Half-Power Beam Width of the DOA estimators implemented. The cylindrical obstacle was placed at 0° and at 50 cm of distance from the robot . .	28
5.1	Duration of the runs in the arena for every obstacle configuration.	54
5.2	Echolocation error statistics.	62

List of Symbols

Variable	Description	SI unit
$\mathbf{a}$	propagation vector	
$\mathbf{A}$	propagation matrix	m
c	sound velocity in air	m s ⁻¹
d	inter-element distance	m
$\bar{d}$	estimated obstacle distance	m
d_{GT}	ground truth obstacle distance	m
f_{max}	maximum frequency before spatial aliasing	Hz
f_{min}	minimum frequency recorded effectively by a microphone array	Hz
$\mathbf{h}$	spatial filter weights	
M	number of elements in the array	
p	pseudospectrum	
$\mathbf{R}$	covariance matrix of the array data	m
$\hat{\mathbf{R}}$	sample estimate of the covariance matrix	m
$\mathbf{V}$	noise related eigenvector matrix	m

Variable	Description	SI unit
γ	angle error weighting coefficient	$\text{m}^{\circ-1}$
Δd	obstacle range	m
Δt	time interval between the call and the echo	s
ε_d	distance error	m
ε_θ	angle error	$^\circ$
θ	direction of a sound source	$^\circ$
$\bar{\theta}$	estimated obstacle direction	$^\circ$
θ_{GT}	ground truth obstacle direction	$^\circ$
λ	wavelength	m
$\mathbf{\Lambda}_y$	eigenvalues diagonal matrix of the covariance matrix	
τ_k	propagation delay	s
ω_c	carrier angular frequency of a narrow-band signal	rad s^{-1}
ω_s	spatial frequency	rad

List of Acronyms

Acronym	Full Term
ADC	Analog-to-Digital Converter
DAC	Digital-to-Analog Converter
DAS	Delay and Sum
DFT	Discrete Fourier Transform
DOA	Direction of Arrival
DSD	Direct Stream Digital
FOV	Field-of-View
HPBW	Half-Power Beam Width
I ² C	Inter-Integrated Circuit
I ² S	Inter-IC Sound
MEMS	Micro-Electro-Mechanical System
MUSIC	Multiple Signal Classification
PCM	Pulse Code Modulation
PDM	Pulse Density Modulation
RMS	Root Mean Square
SNR	Signal-to-Noise Ratio
SONAR	Sound Navigation and Ranging
SPL	Sound Pressure Level
TDM	Time Division Multiplexing
TDOA	Time Difference of Arrival
ULA	Uniform Linear Array

Acknowledgements

I would like to thank the University of Konstanz, and in particular the Centre for the Advanced Study of Collective Behaviour and the University Workshop for the opportunity given me to do research in such a stimulating and vibrant environment. It was an invaluable experience, which made me grow on so many levels that is difficult to summarize in these few lines. Special thanks to my co-supervisors, Dr. Andreagiovanni Reina, Dr. Thejasvi Beleyur, and Prof. Dr.-Ing. Heiko Hamann for their guidance during the eight months I spent there.

I am thankful for all the colleagues met in the offices corridors: it was a pleasure to get to know each one of you and the time spent with you was always quality time. Furthermore, the help I received when I was in need testified to your kindness and availability even for a newbie like me despite all the hard work that was carried out constantly.

I also thank my supervisor, Prof. Fabio Antonacci, for the opportunity to live this experience.

Huge thanks also go to all the colleagues met during this master's years, both in Milano and in Cremona. Especially in Cremona, I found the most welcoming and friendly group of people that I could have wished for. I will bring back in my heart the memories of the two years spent there for the rest of my life.

I am also grateful for my long-time friends, who despite me being far most of the time, stayed in contact and were always there for a joke, a laugh and just a general good time together when I was back in town.

Finally, thanks to my family: my mother, my father, and my younger brother. To you I owe everything.

This thesis marks the end of a journey, and the beginning of new exciting ones. I am eager to commence them, bringing with me everything that these years have taught me.