INGENIEURE ENAC

Léa Terrin

IENAC23

End of Studies Project Memoir

16/10/2026

Estimation of the angle of arrival of AIS signals with an SDR-based solution

2026

Abstract

With the growing maritime traffic, monitoring vessels accurately is becoming essential. Introduced in 1998, Automatic Identification System (AIS) messages, broadcasted by vessels on the two frequencies 161.975 MHz and 162.025 MHz, are supposed to support this task. However, these messages can be easily falsified, intentionally or not. Therefore, false information, for example, about a position can be broadcasted.
The aim of this work is to develop a solution that detects modified AIS messages based on the Direction of Arrival (DoA) estimation. Therefore, a KrakenSoftware Defined Radio (SDR) is chosen to build an antenna array in order to perform DoA estimations with algorithms such as Multiple Signal Classification (MUSIC). Then, the estimated DoA is compared to the bearing, obtained from the data transmitted by the vessels. In case of a mismatch, the AIS message is considered to be falsified. First, a GNU Radio pipeline is developed to decode and register AIS messages. Then three different arrays, each composed of 5 antennas, are used with the KrakenSDR and the GNU Radio pipeline: an Uniform Linear Array (ULA), a pentagon array and a cross array. These recorded AIS messages are later used in MUSIC to estimate the DoA. The arrays were first tested on the site of Deutsches Zentrum für Luft- und Raumfahrt (DLR) and later tested with two measurement campaigns that both took place in Altwarp. The final results of the two measurement campaigns present the ULA as the best performing configuration, followed by the cross array and then the pentagon array. However, the results still show errors that are too significant to validate the performance of the KrakenSDR setup. To improve its performance, other antennas, such as VHF maritime antennas, could be used.

Keywords: AIS, DoA estimation, MUSIC, antenna arrays, KrakenSDR, position verification

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

Contents

Abstract **3**

Introduction **13**

1 Presentation of the company **17**
1.1 DLR . . . 17
1.2 Institute of Communication and Navigation . . . 18
1.3 Conclusion . . . 18

2 State of the art **19**
2.1 Previous related work . . . 19
2.1.1 Consistency of the message content . . . 19
2.1.2 Anomalies on the transmission protocol . . . 20
2.1.3 Comparison with other sensors . . . 21
2.1.4 Analysis of the received radio signal . . . 21
2.2 DoA method . . . 22
2.2.1 Signal model . . . 22
2.2.2 Bartlett method . . . 24
2.2.3 MUSIC . . . 24
2.2.4 Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT) . . . 25
2.3 Conclusion . . . 26

3 Experimental Setup **27**
3.1 KrakenSDR . . . 27
3.2 The setup . . . 28
3.3 Conclusion . . . 29

4 Calibration of the KrakenSDR **31**
4.1 Calibration with *heimdall daq fw* . . . 31
4.1.1 Presentation of *heimdall daq fw* . . . 31
4.1.2 Values for *heimdall daq fw* . . . 33
4.2 Calibration with an osmocomSDR block . . . 34
4.2.1 Implementation of the new Calibration . . . 34
4.3 Conclusion . . . 35

5 Receiving and decoding the AIS signals **37**
5.1 AIS signals reception range . . . 37
5.2 Encoding processing . . . 38
5.2.1 Cyclic Redundancy Check (Cyclic Redundancy Check (CRC)) . . . 39
5.2.2 Bit stuffing . . . 39
5.2.3 Non-Return to Zero Inverted (Non-Return to Zero Inverted (NRZI)) . . . 39
5.2.4 Gaussian Minimum Shift Keying (Gaussian Minimum Shift Keying (GMSK)) modulation . . . 40
5.3 Decoding process . . . 40
5.4 Conclusion . . . 42

6 Setup for the DoA **43**

6.1 Burst Detection . . . 43
6.2 Antenna arrays . . . 45
6.2.1 ULA . . . 45
6.2.2 Pentagon array . . . 51
6.2.3 Cross array . . . 55
6.3 Comparison of different array geometries . . . 58
6.4 Conclusion . . . 60

7 Experimental results **61**
7.1 Test on DLR site . . . 61
7.2 First measurement campaign . . . 64
7.2.1 Pentagon array . . . 66
7.2.2 cross array . . . 68
7.2.3 ULA . . . 69
7.3 Second measurement campaign . . . 71
7.3.1 ULA . . . 72
7.3.2 Pentagon array . . . 74
7.3.3 Cross array . . . 75
7.4 Limitation of the setup . . . 77
7.5 Conclusion . . . 79

Conclusion **81**

Acronyms **83**

Glossary **85**

Bibliography **89**

List of Figures

I.1 AIS traffic monitoring [1] 13

1.1 logo of DLR [2] 17
1.2 DLR site in Neustrelitz [3] 18

2.1 ULA with M = 5 antennas and sketch of the wave arriving from a far-field source. 22

3.1 KrakenSDR [4] 27
3.2 Overview of the experimental setup 29

4.1 Firmware Block diagram [5] 32
4.2 Delay synchronizer diagram [5] 33

5.1 Maximum distance scheme 38
5.2 AIS signal processing chain [6] 38
5.3 GNU Radio chain to decode the AIS messages 41

6.1 GNU Radio chain of the burst detector 43
6.2 ULA setup at DLR 45
6.3 ULA setup scheme 45
6.4 ULA Root Mean Square Error (RMSE) and Cramér-Rao Bound (CRB) for different level of Signal-to-Noise Ratio (SNR) 46
6.5 Theoretical MUSIC spectrum for an angle of 60° 48
6.6 Theoretical MUSIC spectrum for an angle of 85° and different SNR 49
6.7 Theoretical MUSIC spectrum for an angle of 10°, different SNR and different spacing d 49
6.8 MUSIC spectrum for an ULA for an errors of position of 1% 50
6.9 Pentagon array setup at DLR 51
6.10 Pentagon array setup scheme 51
6.11 Pentagon array RMSE and CRB for different level of SNR 52
6.12 Theoretical MUSIC spectrum for an angle of 60° 53

6.13 Theoretical MUSIC spectrum for an angle of 85° and different SNR 54
6.14 MUSIC spectrum for the pentagon array for an errors of position of 1% 54
6.15 Cross array setup at the first measurement campaign 55
6.16 Cross array setup scheme 55
6.17 Cross array RMSE and CRB for different level of SNR 56
6.18 Theoretical MUSIC spectrum for an angle of 50° 57
6.19 Theoretical MUSIC spectrum for an angle of 10° and different SNR 57
6.20 MUSIC spectrum for the cross array for an errors of position of 1% 58
6.21 Comparison of the precision of MUSIC for the different array 59
6.22 Comparison of the precision of MUSIC for the different array 59

7.1 Measurement campaign site 61
7.2 Final GNU Radio chain 62
7.3 Result of the DoA estimation on DLR site for an ULA at 7.08° 62
7.4 Result of the DoA estimation on DLR site for an ULA at 64.51° 62
7.5 Result of the DoA estimation on DLR site for the pentagon array at 50.1° 63
7.6 Result of the DoA estimation on DLR site for the cross array at 52.0° 63
7.7 Setup of the first measurement campaign in Altwarp 64
7.8 ULA used during the first measurement campaign 64
7.9 Pentagon array used during the first measurement campaign 64
7.10 Cross array used during the first measurement campaign 64
7.11 Altwarp map with the reference used by the three arrays 65
7.12 Altwarp with the reference and the different object identified during the first measurement campaign 65
7.13 DoA over time for the pentagon array - VECTOR, JANA 66
7.14 DoA over time for the pentagon array - Station 261, OHLIS 66
7.15 DoA over time for the pentagon array - 211753900, 211315970 67
7.16 DoA over time for the cross array - 211223190, VECTOR, JANA 68
7.17 DoA over time for the cross array - Station 261, OHLIS 68
7.18 DoA over time for the cross array - 211753900, 211315970 69
7.19 Signal power of each antenna of the pentagon array for boat *211753900* 69
7.20 Signal power of each antenna of the cross array for boat *211753900* 70
7.21 ULA for the second measurement campaign 71
7.22 Pentagon array for the second measurement campaign 71
7.23 Cross array for the second measurement campaign 71

7.24 Altwarp map with the reference and the different objects identified during the second measurement campaign 72
7.25 DoA over time for the ULA - 269057081, LA PERLA, 211324960 72
7.26 DoA over time for the ULA - Station 261, 261182155, 257306000 73
7.27 DoA over time for the pentagon array - Station 261, 261182155 74
7.28 DoA over time for the pentagon array - 211324960 74
7.29 DoA over time for the pentagon array - 269057081, LA PERLA 75
7.30 DoA over time for the cross array - Station 261, 261182155 75
7.31 DoA over time for the cross array - 211324960, 269057081 76
7.32 Comparison of the different arrays 76
7.33 S parameter at 162MHz for an antenna height of 44cm 77
7.34 S parameter for an antenna height of 18.2 cm 77
7.35 S parameter at 300MHz with the theoretical antenna height 78
7.36 S parameter at 350MHz with the theoretical antenna height 78
7.37 S parameter at 400MHz with the theoretical antenna height 78
7.38 S parameter from 500MHz to 1GHz with the theoretical antenna height 78
7.39 Maritime VHF antenna 79

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

List of Tables

I.1 Fields of the AIS Class A message (message types 1, 2 and 3) [7] 16
I.2 AIS Class A reporting intervals according to vessel condition [7] 16

3.1 Key specifications of the KrakenSDR [4] 28

5.1 Estimated maximum AIS reception range for different vessel and coastal antenna heights computed with the vacuum and tropospheric models 38

6.1 Settings of the different arrays 58

7.1 AIS emitters detected during the first measurement campaign 65
7.2 AIS emitters received during the second measurement campaign at Altwarp . . . 71

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

Introduction

Nowadays, around 80–90% of the world's trade volume is carried by sea [1]. Since the density of vessels in major shipping areas continues to increase, better maritime traffic monitoring and safety systems are needed [8]. Localising the vessels accurately is important in order to supervise the traffic.

In 1998, it was decided that AIS messages would be used for the communication between boats, as well as between boats and the shore. While they were initially used in situations of low visibility, they are now also used to send information about a boat's current status, for example whether it is at the harbour or moving, its speed, and more. The AIS messages are transmitted on the two frequencies [6] 161,975 MHz (called channel 87B or channel A) and 162,025 MHz (called channel 88B or channel B).

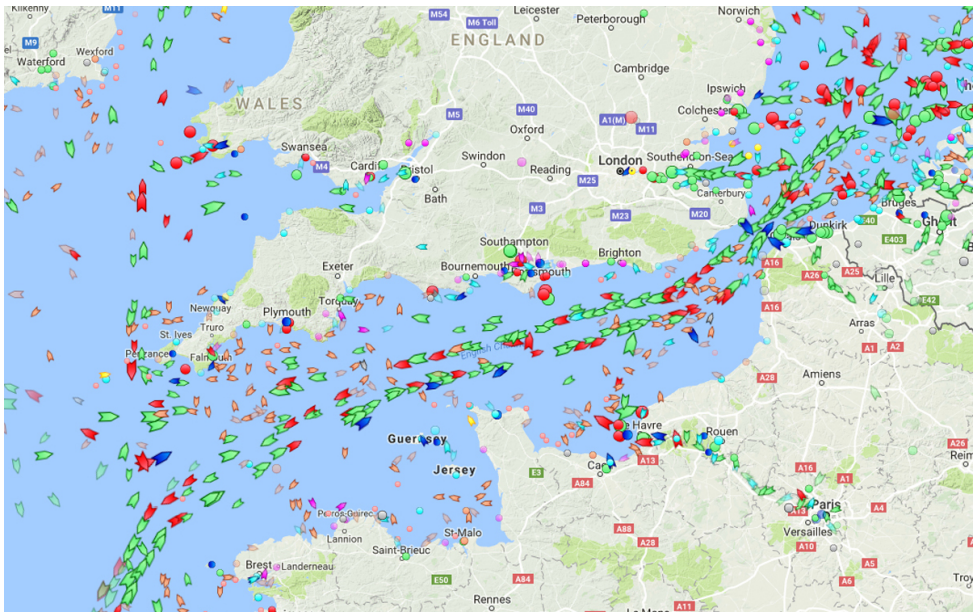


Figure I.1: AIS traffic monitoring [1]

Therefore there are two types of messages:

- Class A : AIS messages are mandatory and sent by boats over 300 GT. They use the Self-Organized Time Division Multiple Access (SOTDMA) protocol, which is an anti-collision protocol. It splits time into 60 second repeating frames that each contain 2,250 slots. Before transmitting, a Class A transponder listens to the channel. It identifies the slots that are already occupied and selects a free one. Afterwards, it announces its future slot reservations in each message, in order for other transponders to avoid these slots. Depending on the boat's

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

status and speed, Class A messages are transmitted every two to ten seconds, if the boat is moving. If the boat is not moving, the messages are transmitted every three minutes [9][10].

- Class B : AIS messages are not mandatory and sent by smaller vessels. They use the Carrier Sense Time Division Multiple Access (CSTDMA) protocol, which is a simpler and a more passive anti-collision protocol than SOTDMA. Instead of announcing slot reservations in advance, a Class B transmitter listens to the channel right before transmitting and only sends its message if the chosen slot seems to be free. Since there is no reservation mechanism as in Class A transmissions, Class B transmissions present a higher risk of collisions in dense areas. When the boat is not moving it transmits AIS every 3 minutes, and every 30 seconds, when it is moving. [11][10].

On TableI.1, the information and size of Class A messages can be seen [7]. TableI.2 also shows the intervals between each AIS message according to its condition. However, even though AIS has become a fundamental data source for maritime surveillance, these messages are not always reliable. Since the protocol does not include any encryption, authentication or integrity check, AIS messages can be easily manipulated. This can be done by purposely switching a transponder off, which will not be further discussed in this project, or by changing the content of a message. The goal of vessels, doing these manipulations, is to hide a vessel's position in order to, for example, avoid being sanctioned [12]. Additionally, AIS messages may contain unintentional errors that can be caused by, for example, a poor installation of the equipment or external factors [13]. Generally, there are three reasons why a transmitted position can be wrong. The first one is that the crew intentionally alters it, in order to, for example, hide the vessel's fishing in a forbidden area. The second reason is that the position is affected by interference with the Global Navigation Satellite System (GNSS) signals in the form of jamming or spoofing. Jamming consists of overpowering the GNSS signal with noise. This leads to the receiver not obtaining a fixed position. In this case, instead of transmitting a false one, the transponder reports that there is no position available or it sends an estimated one. Spoofing consists of transmitting fake GNSS signals that are accepted by the receiver. This one then calculates and broadcasts an incorrect position. Since the transponder works properly and respects the AIS protocol, the error cannot be seen from the message content alone. The final reason is that the position can be affected by equipment faults. These usually produce smaller errors or isolated jumps [13].

Therefore, the information transmitted in AIS messages is not reliable. In order to detect the transmission of false positions, several solutions have already been researched. The first group of methods validates the consistency of the content from the messages. Kinematic tests, for example, compare the distance between two consecutive positions of a vessel to the distance it could have travelled at its reported speed. This can reveal several vessels using the same Maritime Mobile Service Identity (MMSI) [14]. Another method tracks each vessel with a Kalman filter and flag positions and speeds that do not match the predicted motion [15]. Looking at multiple vessels at the same time, another one detects GNSS spoofing when several vessels suddenly jump from the same area to the same false position [16]. While these methods are simple and able to process large amounts of data, they can only detect inconsistencies, and a false track that seems to be realistic is not detected. The second group of methods has a different approach. It rather checks how the messages are transmitted instead of analysing what they contain. One of them detects them by verifying that each message respects the Time Division Multiple Access (TDMA) protocol defined in the AIS standard, specifically the reservation of time slots, since it is known that messages, which are purposely created, rarely follow this protocol. However, this check can fail when a false position is fed to a transponder and the messages still respect the protocol, like in the case of GNSS spoofing [15]. The third group of methods compares AIS data with other sensors, such as coastal radars or satellite Synthetic Aperture Radar (SAR) images. An

example are the Sentinel-1 C and D satellites, which carry an AIS receiver that operates at the same time as the SAR instrument. This makes it easier to match the vessels that are detected on the images to their AIS messages. While these sensors give an independent view of the vessels, they are not always available, and satellite images are additionally only acquired during short passes over a given area [17]. A different method uses coastal radar that can detect the vessels on its own, whether they transmit AIS messages or not. That way, the position reported in each message can be compared to the position of the matching radar echo [18]. The downside of this approach, however, is its coverage, that is limited to a few tens of kilometres from the coast by the radio horizon, and its requirement of a dedicated installation. A few other methods use the received radio signal itself. Since the signal is always sent from the actual position of the vessel, it has information that does not depend on what the vessel declares. One method uses the Received Signal Strength Indicator (RSSI) that is measured at coastal base stations, to decide if the reason for a missing message are the propagation conditions or a transponder that has purposely been switched off [19]. Other methods use a group of spatially distributed receivers, to locate the transmitter directly from its signal. Several stations capture the same signal, and the transmitter's location is determined using multilateration or Time Difference of Arrival (TDOA) techniques. This principle can be applied to AIS signals received by several coastal stations in order to verify the reported positions. Therefore, this approach offers wide coverage and redundancy. However, it requires precise time synchronisation between the receivers and important infrastructure investment [20].

Another approach could be implemented with a single receiver with multiple antennas. Indeed, the phase differences between the antennas depend on the direction of the incoming signal. Thus, DoA estimation algorithms can be used to determine the bearing of a signal received by the antennas of such a receiver. Therefore, the bearing of the transmitter can be compared to its reported position. This leads to the following problematic: How to estimate the DoA of AIS signals with an SDR-based solution in order to track vessels?
To answer this problem and implement this system, a KrakenSDR is used. The reasons to choose an SDR are its flexibility and low-cost, compared to traditional hardware receivers. Since, unlike dedicated hardware solutions, most of the signal processing is handled by the software, parameters such as frequency and bandwidth can be easily reconfigured. Compared to the former four-channel KerberosSDR, the fifth channel of the KrakenSDR allows the use of a five-element antenna array. This improves direction-finding accuracy and multipath rejection compared to smaller arrays. Another reason to choose the KrakenSDR is its five phase-synchronized receivers that share a common clock and reference signal, which is required for any DoA algorithm that is relying on the phase difference between antenna elements [4]. Additionally, its tuning range of 24 MHz to 1766 MHz covers the 162MHz AIS band and the device is based on a well-documented open-source software, including the calibration. For these reasons the Kraken SDR is a suitable choice for the DoA estimation of AIS signals.

First, the KrakenSDR will be presented and calibrated. Then, a pipeline to receive and decode those AIS signals will be implemented, alongside a pipeline to detect and record AIS bursts. After a few theoretical DoA simulations and some practical tests on the site of DLR, the functioning of the GNU Radio pipeline and DoA estimation will be validated. Afterwards, real data will be collected in Altwarp and later analysed.

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

Field name	Size in bits	Description
Message ID	6	Identifier of message 1, 2 or 3.
Repeat Indicator	2	Used by repeaters to indicate how many times this message has been repeated.
User ID	30	Maritime Mobile Service Identity (MMSI for *maritime mobile service identity*).
Navigational Status	4	Indicates the state of the vessel. Several codes are defined, such as fishing, at anchor, moored, aground...
Rate of Turn	8	Rotation rate of the vessel in degrees per minute.
Speed Over Ground	10	Speed of movement in tenths of a knot.
Position Accuracy	1	Indicates whether the positioning accuracy is below or above 10 metres.
Longitude	28	Longitude in ten-thousandths of a minute of arc.
Latitude	27	Latitude in ten-thousandths of a minute of arc.
Course Over Ground	12	True direction in tenths of degrees of the course followed by the vessel subject to drift.
True Heading	9	Orientation in degrees of the vessel relative to north.
Time Stamp	6	Seconds of Coordinated Universal Time (UTC).
Special Manoeuvre	2	Indicates whether the vessel is performing a special manoeuvre.
Reserved	3	Not used.
RAIM Flag	1	Indicates whether the transmitter uses the autonomous integrity monitoring of positioning (RAIM for *receiver autonomous integrity monitoring*).
Radio Status	19	Contains information used for SOTDMA slot allocation and the synchronisation state.

Table I.1: Fields of the AIS Class A message (message types 1, 2 and 3) [7]

Vessel condition	Average period
Vessel at anchor or moored moving at less than 3 knots	3 minutes
Vessel at anchor or moored moving at more than 3 knots	10 seconds
Moving from 0 to 14 knots	10 seconds
Moving from 0 to 14 knots and changing course	3,33 seconds
Moving from 14 to 23 knots	6 seconds
Moving from 14 to 23 knots and changing course	2 seconds
Moving at more than 23 knots	2 seconds
Moving at more than 23 knots and changing course	2 seconds

Table I.2: AIS Class A reporting intervals according to vessel condition [7]

Chapter 1

Presentation of the company

In this chapter, the German aerospace center DLR, where this project was done, is presented.

1.1 DLR

DLR, which stands for Deutsches Zentrum für Luft- und Raumfahrt, was originally founded in 1989 after the merge of several research institutes and ultimately named DLR in 1997. Today DLR employs about 12000 people, has 30 sites across Germany, its headquarters in Cologne and several offices abroad. It is a non-profit association that is mostly funded by the German state and fulfills a public research mission. DLR is the research center for space and technology for the German federal government and has been directed by Anke Kaysser-Pyzalla since 2020. Its projects can be used by companies, administrations or public organizations. The DLR logo can be seen in Figure 1.1 [2]:

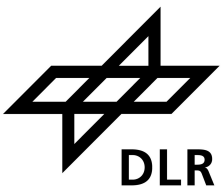


Figure 1.1: logo of DLR [2]

DLR has three main roles. First, it is a research center with its own institutes and facilities. Second, the German Space Agency at DLR plans and executes space activities on behalf of the German federal government. Third, it manages the budget for industry and research with two agencies.
Its main mission is to explore Earth and space and develop sustainable technologies. In order to accomplish that, DLR has 51 institutes and spreads across 30 sites, which work on research that covers space, aeronautics, energy, transport, security and defense. DLR also has international partnerships and offices abroad in Brussels, Paris, Tokyo and Washington D.C. [2].

1.2 Institute of Communication and Navigation

This project was done at the Institute of Communication and Navigation, which is spread across three sites: Oberpfaffenhofen, Neustrelitz and Aachen-Merzbrück. It consists of several departments working on different subjects such as satellite networks, optical satellite links, communication systems, navigation and nautical systems. Their missions focus on global connectivity, autonomy and cooperation in transport, and cybersecurity for radio systems.
The project was developed within the department of Nautical systems based in Neustrelitz. Its aim is to ensure secure maritime traffic at sea and on inland waterways. The department is formed of the three groups: Maritime Services, Multi Sensor Systems and Traffic Systems [3]. Figure 1.2 shows the DLR site, where the project was supervised.



Figure 1.2: DLR site in Neustrelitz [3]

The Maritime Services group develops and tests radio navigation for maritime Positioning, Navigation and Timing (PNT) systems. They adapt them to modern and future GNSS and test them in real conditions in the research port of Rostock. The group's work aims to improve PNT data provision while reducing the resource use and costs. All software is developed by the researchers of the team [3].

The Multi Sensor Systems group develops methods to provide reliable PNT data such as speed, course or altitude on board of vessels. For this, GNSS receivers or inertial sensors with augmentation services and alternative systems like R-Mode are used. Then it estimates the integrity of the result [3].

The Traffic system group is the one where this project was done. The group develops technologies to make reliable the description of the traffic around the vessels. In order to do that the researchers combine AIS messages with radar measurements. Indeed, since AIS messages can be falsified, and the radar itself can be disturbed by clutter, they cannot be trusted. By combining the two, the group tries to reliably track the vessels in difficult conditions. The tests are made in the port of Rostock. This project of detecting falsified AIS messages is aligning with the mission of the group. [3].

1.3 Conclusion

In this chapter DLR and the institute, where the project was developed were presented.

Chapter 2

State of the art

In this chapter, previous work on the detection of anomalies in AIS messages will be explained and several methods that perform DoA will be introduced.

2.1 Previous related work

As stated in the introduction, the proposed methods, used to detect false AIS data, are structured into four groups. While the first group checks the consistency of the message's content, the second one checks the way messages are transmitted, the third one compares AIS data to other sensors and the fourth group uses the received radio signal itself. In the following abstract, each group's most important works, results and limitations are described.

2.1.1 Consistency of the message content

In order to detect false AIS data it is possible to check if the content of the message is physically possible. This is the principle of these methods.

Kinematic tests: The first method checks whether a vessel could have actually traveled from its first reported position to its next reported one. The distance r between these two positions, is being calculated with the haversine formula. Then, it is compared to the distance that could have actually been traveled between the two positions, at the vessel's reported speed:

$$r \leq \bar{v}\,\Delta t + \varepsilon, \tag{2.1}$$

where $\bar{v}$ is the reported speed, Δt the time between the two messages and ε a tolerance. If this test fails, one can assume that these two messages are coming from two different vessels, which are using the same MMSI. The advantage of this method is its simplicity. However, since the tolerance has to be large, small position errors might not be detected while some other vessels are still mistakenly flagged. Additionally, it relies on the reported speed that may be incorrect on its own [14].

Tracking with a Kalman filter: A different method suggests to track each vessel with a Kalman filter that uses a model with a nearly constant velocity [15]. Then, for every new message, the filter predicts the position of the vessel and compares it to the reported one. The

difference ν between these two values is called the innovation and normalized by its innovative covariance S. The message is rejected if:

$$\nu^T S^{-1} \nu > \gamma, \tag{2.2}$$

where the threshold γ is chosen for a given false alarm probability. The same test is also applied to the vessel's reported speed. While it is possible to run this method in real time, it only detects errors larger than the filter's uncertainty, which is approximately 80 m for the position and 4.5 knots for the speed and it fails if the vessel is not following the velocity model [15]

Detection of GNSS spoofing: Another method uses AIS data to detect GNSS spoofing on a larger scale. The first thing this method does, is to look for sudden position jumps that would be impossible given the vessel's speed. Known artifacts, such as bit errors or default MMSI numbers, are removed. Since a spoofing transmitter affects all vessels in its range the same way, an event is only kept if a minimum of two vessels jump from the same area to the same false position. Thus, the disadvantages of this method are, first, that AIS data alone cannot tell the difference between an area affected by spoofing and an area with poor reception coverage, and second, that the method can only detect Spoofing, which affects multiple vessels at the same time [16].

Detection of unusual behaviours: Another group of methods studies the vessels' usual behaviour in a specific area and then flags the tracks that do not follow it. A method can study this behaviour by, for example, a recurring neural network that learns the probabilities of AIS tracks and then flags the tracks that have a low probability compared to the others in the same area. However, most of these methods are based on variations from usual routes and are not evaluated on data with a real ground truth. They detect unusual behaviours not falsified messages, since they always assume that the content of the messages is correct [12].

All these methods only use what the vessels declare. They can notice if a position is physically impossible or unusual, however, they cannot detect a false track that seems to be realistic, for example, when a spoofed position slowly moves away from the true one.

2.1.2 Anomalies on the transmission protocol

Therefore, other methods focus on checking how the messages are transmitted, instead of checking the content of the messages. Since AIS uses a complex anti collision protocol, messages that are created from scratch almost never follow it. This approach is based on two checks. The first one checks if the time between two messages of the same vessel matches the reporting interval that is standardly required. This depends on the speed and navigation status of the vessel. That is why, beforehand, a Kalman filter has already had to confirm that the vessel's reported speed is consistent. The second one checks that each message is received in a time slot that was reserved before by the same vessel. This is done by using the reservation fields contained in the previous messages. From a message that was not announced in advance or reservation fields that are not consistent with the standard, one can assume that the message was probably not produced by a real transponder [15]. This approach runs in real time, but has three important limitations. First, since each message must be assigned to its time slot, this requires a receiver with a precise time stamp. It is, for example, impossible to do it with satellite AIS, due to the propagation delay that is equal to several milliseconds and therefore the loss of many messages in collisions. Second, because messages that are not received also create alerts, a vessel is only considered to be suspicious, when more than 80 % of its messages are flagged for several minutes. Lastly, when

a false position is given to a transponder, like in the case of GNSS spoofing, the messages still follow the protocol and cannot be detected [15].

2.1.3 Comparison with other sensors

A third approach suggests comparing AIS data to an independent sensor, such as a coastal radar or satellite SAR images, that can see the vessels. Then, the AIS data is suspicious, when a vessel that is detected by the sensor does not have a matching AIS message or if the reported position does not match the detection. For example, like on Sentinel-1 C and D, an AIS receiver can be placed on the same satellite as the SAR instrument and operated at the same time as it. The AIS antenna is an array of four elements whose beam is shaped in order to cover the area imaged by the radar. Since each AIS packet is precisely time stamped, it can be matched with the SAR data. However, a receiver in space sees a very large area and therefore has many messages from different vessels arriving at the same time. Thus, these messages can collide and cause some vessels to not be detected [17]. Furthermore, satellite images are only taken during short passes and processing SAR images can take from a few hours to one day. Another method uses coastal radar that can detect the vessels by itself, regardless if they transmit AIS messages or not. That way, the reported position in each message can be compared to the position of the matching radar echo[18]. However, the downside is, its coverage, which is limited to a few tens of kilometres from the coast by the radio horizon, and that these solutions need expensive infrastructure.

2.1.4 Analysis of the received radio signal

The last approach uses the physical AIS signal, not only the decoded message. Since the AIS signal is always sent from the actual position of the vessel, it shows information that cannot be changed by modifying the content of the messages.

Signal strength: The RSSI, which is measured by coastal base stations, can be used to determine whether a message is missing, because of propagation conditions or because a transponder has been purposely switched off. This method uses two models to describe the usually received power. The first one is physical and depends on the distance between the vessel and the station. Here, the power follows the free space model, up to a critical distance:

$$d_c = \frac{4h_T h_R}{\lambda}, \tag{2.3}$$

and then the ITU-R P.1546 model up to the radio horizon:

$$R \approx 4.131 \left(\sqrt{h_T} + \sqrt{h_R} \right), \tag{2.4}$$

where h_T and h_R are the heights of the transmitting and receiving antennas in metres, is the wavelength, and R is given in kilometres. Since the coverage of each station depends on its antenna and surroundings, the second model is learned from data that has been received by the station in the past. When a message is missing both models are used to decide if a message was expected to be received. However, this method was only researched on simulated events and it requires a model for each station. Additionally, propagation effects such as tropospheric ducting are not modelled, even though they could extend the range to multiple hundred kilometres [19].

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

Localization with several receivers: Another method consists of locating the transmitter from its signal and to then compare this position to the reported one. To detect position spoofing, the TDOA of the same AIS message that was received by several coastal stations, combined with a tracking filter, is evaluated [20]. Therefore, one can estimate a position with an accuracy of a few hundred metres. On one hand, this approach offers wide coverage and redundancy. On the other hand, it requires that several stations receive the same message and are precisely synchronized in time. This needs a large investment in the infrastructure [21].

2.2 DoA method

In this section three DoA methods are presented: the Bartlett method, the MUSIC algorithm and the ESPRIT algorithm.

2.2.1 Signal model

The three DoA algorithms presented are using the very same signal model. When a wavefront reaches an array of M antennas, each antenna receives the same signal but with a different time delay that depends on the geometry of the array and on the angle θ, measured from the broadside, from which the wave arrives. These time delays can be seen as measurable phase differences between antennas and the DoA is computed from these phase differences [22]. Consider an ULA of M antennas that are equally spaced by a distance d as seen in Figure2.1:

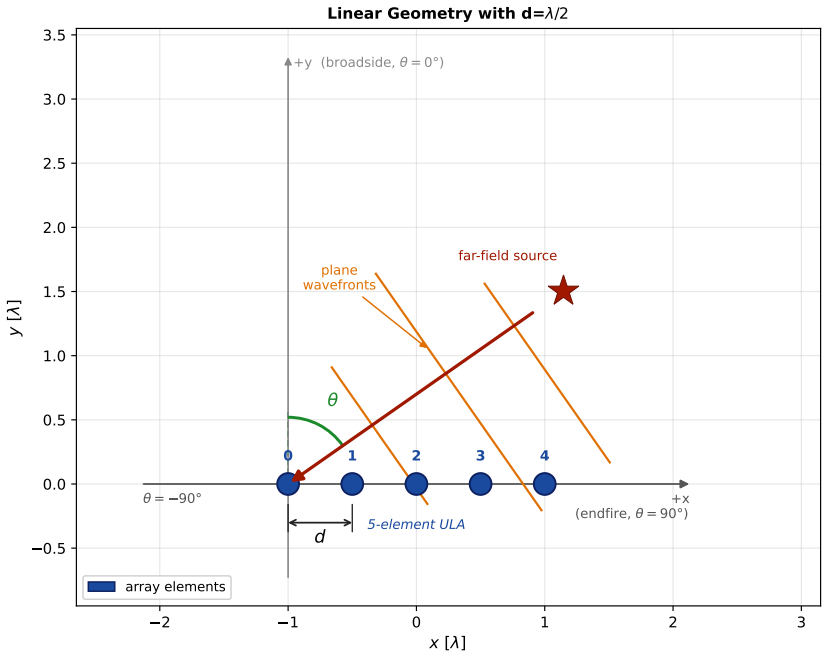


Figure 2.1: ULA with M = 5 antennas and sketch of the wave arriving from a far-field source.

Consider a narrow band signal $s[n]$ that arrives from an angle θ. The signal received on antenna i is the source signal multiplied by a phase shift that depends on the difference of the path length

between that antenna and the reference antenna (antenna 0 in this case) [22]:

$$x_i[n] = a_i(\theta)\, s[n] + w_i[n], \qquad i = 0, 1, \ldots, M-1, \tag{2.5}$$

where $s[n]$ is the AIS signal and $w_i[n]$ is additive white noise on antenna i. In vector form for all antennas simultaneously it gives [22]:

$$X[n] = A(\theta)\, s[n] + W[n], \tag{2.6}$$

where $X[n] = [x_0[n],\, x_1[n],\, \ldots,\, x_{M-1}[n]]^T \in \mathbb{C}^M$ is the snapshot vector at time n.

The vector $A(\theta)$ is called the steering vector. Each of its entries $a_i(\theta)$ is a phase shift representing the propagation delay from the source to antenna i relatively to the reference antenna (antenna 0 in this case). For a wave arriving at angle θ, antenna i, that is spaced $i \cdot d$ from the reference, is closer to the source by a distance $i \cdot d \sin\theta$. Therefore, it receives the wave earlier with a time advance of $\tau_i = i \cdot d \sin\theta \,/\, c$, where c is the speed of light. This gives [23]:

$$a_i(\theta) = e^{j\frac{2\pi}{\lambda} i\, d\, \sin\theta}, \tag{2.7}$$

so the full steering vector is [23]:

$$A(\theta) = \begin{bmatrix} 1 \\ e^{j\frac{2\pi d}{\lambda}\sin\theta} \\ e^{j\frac{4\pi d}{\lambda}\sin\theta} \\ \vdots \\ e^{j\frac{2\pi(M-1)d}{\lambda}\sin\theta} \end{bmatrix}. \tag{2.8}$$

For other geometries the expression of $A(\theta)$ changes, since it depends on the positions of the antennas. When K sources are present their contributions add up and the steering vectors are grouped in the steering matrix $\mathcal{A} = [A(\theta_1), \ldots, A(\theta_K)] \in \mathbb{C}^{M\times K}$:

$$X[n] = \mathcal{A}\, S[n] + W[n], \tag{2.9}$$

where $S[n] \in \mathbb{C}^K$ contains the K source signals.

The spatial covariance matrix R represents the cross-correlations between the antennas. It is estimated from N received snapshots as [23]:

$$\hat{R} = \frac{1}{N}\sum_{n=1}^{N} X[n]\, X^H[n], \tag{2.10}$$

where $\hat{R}$ is an estimation of the true covariance matrix R, calculated from N finite snapshots and converges to R as N increases. It converges if the snapshots are independent and identically distributed. Also $(\cdot)^H$ denotes the Hermitian transpose. The received signals is complex, so the standard transpose is not sufficient and the Hermitian transpose is required for the covariance matrix $\hat{R}$ to be Hermitian positive semi-definite. It is the same as the real symmetric positive semi-definite matrix, but with complex values. Moreover, each entry $R_{ij} = \mathbb{E}[x_i x_j^*]$ is the cross-correlation between antennas i and j. Since both receive the same source signal with a relative phase shift of $e^{j\frac{2\pi d}{\lambda}(i-j)\sin\theta}$, the matrix R is structured around the steering vector $A(\theta)$. For a single source [23]:

$$R = \sigma_s^2\, A(\theta)A^H(\theta) + \sigma_w^2\, I, \tag{2.11}$$

where σ_s^2 is the signal power and σ_w^2 is the noise power on each antenna and I is the identity matrix of the matching dimension [23].

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

2.2.2 Bartlett method

The Bartlett method is one of the most simple DoA estimation methods. The idea is to test each possible direction θ one by one. For a given direction, the signal of each antenna is shifted in phase to cancel the delay that a wave coming from this direction would have and the M signals are then added together. If θ is the true direction of the source, all the signals are in phase after this correction so they add up and the output power is maximal. Otherwise, they partly cancel each other and the power is lower. With the steering vector $A(\theta)$ and the covariance matrix $\hat{R}$ defined in the signal model, this output power is [23]:

$$P_{\mathrm{B}}(\theta) = \frac{A^H(\theta)\,\hat{R}\,A(\theta)}{A^H(\theta)\,A(\theta)}. \tag{2.12}$$

This function is called the Bartlett spectrum, and the estimated DoA is the angle of its highest peak. This method has several advantages. It is simple and does not require important resources, since it does not need a matrix inversion or an eigenvalue decomposition. Additionally, it does not need to know the number of sources. When only one source is present in white noise, the peak of the Bartlett spectrum is the maximum likelihood estimate of its direction [24]. However, the resolution of the Bartlett method is limited by the aperture of the array because the width of the main lobe is around λ/D, where D is the size of the array. Therefore, two sources with close directions cannot be separated and the side lobes of a strong source can hide another one. Finally, it requires a search over a grid of angles [24].

2.2.3 MUSIC

The MUSIC algorithm (MUltiple SIgnal Classification) estimates the DoA by using the Eigenvalue Decomposition (EVD) of the covariance matrix $\hat{R}$ that is defined in the section signal model. The EVD of R can be written as:

$$R = \sum_{i=1}^{M} \lambda_i\, u_i\, u_i^H, \tag{2.13}$$

where λ_i are the eigenvalues and u_i are the corresponding eigenvectors. Each eigenvalue λ_i represents the power that is captured along the direction defined by its associated eigenvector u_i. The eigenvectors form an orthonormal basis of $\mathbb{C}^M$. This means that they point in directions that are mutually uncorrelated in the received signal space. In the ideal case, the K largest eigenvalues correspond to the K sources, while their eigenvectors represent the signal subspace. The remaining $M-K$ (MUSIC imposes $K < M$) eigenvalues are all equal to the noise power σ_w^2 and their eigenvectors represent the noise subspace.

The EVD of R gives [23]:

$$R = E\,\Lambda\,E^H, \tag{2.14}$$

where Λ is the diagonal matrix composed of the eigenvalues of R sorted in decreasing order and E is the matrix of the eigenvectors of R which can be decomposed into two subspaces: E_s, the signal subspace and E_n, the noise subspace.

Once the eigenvalues are obtained they are sorted in decreasing order and a separation appears between the largest eigenvalue(s), which correspond(s) to the signal. The remaining $M-K$ small eigenvalues correspond to the noise. The eigenvectors associated with these small eigenvalues form the noise subspace E_n. MUSIC is using the fact that the true steering vector is orthogonal to this noise subspace [22]:

$$A^H(\theta_{\mathrm{true}})\,E_n \approx 0. \tag{2.15}$$

Then MUSIC tries all the θ that are candidates and computes the pseudo-spectrum which measures how close each candidate steering vector is to being orthogonal to the noise subspace [22]:

$$P_{\text{MUSIC}}(\theta) = \frac{1}{A^H(\theta)\, E_n E_n^H\, A(\theta)}. \tag{2.16}$$

When θ matches the true DoA, the denominator $A^H(\theta)\, E_n E_n^H\, A(\theta)$ approaches zero so P_{MUSIC} increases and produces a peak. The angle at which this peak occurs is the estimated DoA.

MUSIC is a very common DoA algorithm since it only needs the steering vector to be known so it can be used with any array geometry and with several sources at the same time. Its main advantage is its high resolution so it can separate sources that are very close and its precision. MUSIC also has some drawbacks. First, the number of sources K has to be known in advance and it has to be smaller than the number of antennas M. Second, the algorithm performances decrease when the sources are coherent, for example with multipath on nearby structures. Third, the numerical search can take a large part of the computation time [23].

2.2.4 ESPRIT

ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques) is a subspace method, like MUSIC, however, it does not need to search over a grid of angles. Indeed, the directions are obtained directly with a formula [23]. It uses the same signal subspace E_s as MUSIC but instead of using its orthogonality with the noise subspace, it uses a property of the geometry of the array called shift invariance.

Consider an ULA of M antennas spaced by d and split it into two overlapping subarrays of $M-1$ antennas where the first one contains antennas 0 to $M-2$ and the second one antennas 1 to $M-1$. The second subarray is the first one shifted by the distance d. These subarrays are selected with the matrix [23]:

$$J_1 = \begin{bmatrix} I_{M-1} & 0 \end{bmatrix} \qquad J_2 = \begin{bmatrix} 0 & I_{M-1} \end{bmatrix}, \tag{2.17}$$

where I_{M-1} is the identity matrix of size $M-1$ and 0 is a column of zeros. From the expression of the steering vector $A(\theta)$, each entry is equal to the previous one multiplied by the same factor $e^{j\frac{2\pi d}{\lambda}\sin\theta}$. Therefore, the steering vector of the second subarray is the one of the first subarray multiplied by this factor [23]:

$$J_2\, A(\theta) = e^{j\frac{2\pi d}{\lambda}\sin\theta}\, J_1\, A(\theta). \tag{2.18}$$

With K sources and the steering matrix $\mathcal{A}$ defined in the signal model section this relation becomes [23]:

$$J_2\, \mathcal{A} = J_1\, \mathcal{A}\, \Phi, \qquad \Phi = \text{diag}\left(e^{j\frac{2\pi d}{\lambda}\sin\theta_1}, \ldots, e^{j\frac{2\pi d}{\lambda}\sin\theta_K}\right). \tag{2.19}$$

The directions of the sources are contained in the diagonal matrix Φ.

The matrix $\mathcal{A}$ is unknown because the directions are unknown. But as seen for MUSIC, the signal subspace $E_s \in \mathbb{C}^{M\times K}$ is made of the K eigenvectors associated with the largest eigenvalues, spans the same space as the steering vectors of the sources. There is an invertible matrix T of size $K \times K$ such that $E_s = \mathcal{A}\, T$. By applying the two selection matrices to E_s it gives [23]:

$$E_1 = J_1 E_s = J_1 \mathcal{A}\, T, \qquad E_2 = J_2 E_s = J_1 \mathcal{A}\, \Phi\, T, \tag{2.20}$$

which gives

$$E_2 = E_1\, \Psi \qquad \text{with} \qquad \Psi = T^{-1}\, \Phi\, T. \tag{2.21}$$

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

The matrices Ψ and Φ have the same eigenvalues. So even if $\mathcal{A}$ and T are unknown, the eigenvalues of Ψ are exactly the factors $\phi_k = e^{j\frac{2\pi d}{\lambda}\sin\theta_k}$ that contain the directions. And Ψ can be computed from the data. ESPRIT starts by computing the signal subspace E_s from the EVD. Then $E_1 = J_1 E_s$ and $E_2 = J_2 E_s$ are computed. After that, the system $E_2 = E_1\Psi$ is solved by least squares [23]:

$$\hat{\Psi} = \left(E_1^H E_1\right)^{-1} E_1^H E_2. \tag{2.22}$$

Finally the eigenvalues $\hat{\phi}_k$ of $\hat{\Psi}$ are computed and the direction of each source is found from the phase of the corresponding eigenvalue [23]:

$$\hat{\theta}_k = \arcsin\left(\frac{\lambda \arg(\hat{\phi}_k)}{2\pi d}\right). \tag{2.23}$$

ESPRIT only needs two identical subarrays shifted by a known vector, and therefore it is not limited to the ULA. For a cross array with a central antenna, for example, it could work. However, with other arrays such as a pentagon it would not work since it does not contain two subarrays that are shifted copies of each other. The main advantage of ESPRIT is that it gives the directions with a direct formula after the EVD. Indeed, it does not require to scan all the angle like MUSIC, so the calculation is faster. Moreover, its resolution is similar to MUSIC. However, an error that differs from one antenna to another creates errors in the DoA estimation. Like MUSIC, it needs the number of sources K to be known and it degrades when the signals are coherent. Finally, the method is restricted to shift-invariant geometries[23].

2.3 Conclusion

First, this chapter reviewed previous work on this topic by presenting its methods. In summary, methods that check the message do not detect false vessel tracks that seem realistic. Methods that check the TDMA protocol can detect messages that were created from scratch, however, they cannot detect a false position that was sent by a transponder, for example as a result of GNSS spoofing. While methods based on other sensors or data sources, like radars or satellite images, are independent of the message, they have a limited coverage and a high cost. Methods based on the radio signal are also independent of the message, however, localization requires several synchronized stations. Since the AIS signal is always transmitted from the true position of the vessel, estimating its DoA with a single multi-antenna receiver provides an independent check for the reported position and requires only one receiving site. This is the reason why this approach is being used in this project. Thus, several DoA estimation methods were presented. The Bartlett method is simple, since it does the number of sources before its execution and has a good precision. However, its main peak can be quite large as its secondary lobes and therefore, detecting several sources could be a problem. The MUSIC algorithm requires the number of sources in advance, can require more calculation time and is sensible to multipath. However, it shows a better resolution than the Barlett method. Lastly, ESPRIT also requires the number of sources before its execution, is sensible to multipath and does not work with every array. However, its computation time requires less power than other algorithms. Among these three methods, MUSIC was chosen, due to its good documentation, the good knowledge of its limitations and its good precision and resolution.

Chapter 3

Experimental Setup

This chapter presents the hardware and software used during this project. As explained in the introduction the KrakenSDR was chosen. This chapter first describes the KrakenSDR in more detail and then introduces the Raspberry Pi and the setup of the project.

3.1 KrakenSDR

The KrakenSDR is a SDR that can be used for the DoA estimation. It is composed of an RTL2832U chip that is a digital demodulator chip that was manufactured by Realtek and it was originally built for DVB-T television reception on USB dongles. It integrates an eight-bit Analogue-to-Digital Converter (ADC), a digital demodulator, and a USB 2.0 interface on a single die. However, in the 2010's some people, passionate about electronics, discovered that this chip could be modified to give a direct access to raw radio signals. Compared to other SDR, this chip is very affordable. This discovery opened access to radio signal analysis by turning a low-cost TV dongle into a signal processing tool [25].



Figure 3.1: KrakenSDR [4]

The KrakenSDR drives its five channels from a single shared reference clock. Each tuner derives its own local oscillator from this reference, so all five local oscillators run at the same frequency. Thus, the phase offset between channels stays constant over time. The same clock also drives the five ADC so the sample streams do not drift relatively to each other. This coherence is very important to perform a DoA estimation. The main hardware specifications of the KrakenSDR are on Table 3.1:

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

Feature	Description
Receiving channels	5 phase-coherent, each with an SubMiniature version A (SMA) connector
Local oscillator	Shared to ensure inter-channel phase coherence
Calibration	Internal noise source for inter-channel calibration
Frequency range	24 MHz to 1766 MHz
Max sample rate	2.56 MS/s per channel
Interface	USB 2.0

Table 3.1: Key specifications of the KrakenSDR [4]

The KrakenSDR is connected to a computer or a Raspberry Pi by a USB cable which carries the data of all five channels. The processing software runs on this computer or on the Raspberry Pi and provides a web interface, so the system can be used either locally or remotely over a network. Moreover software is designed to work with the KrakenSDR:

- *heimdall daq fw*: the software used to do the calibration. Its functioning will be detailed in the next chapter [5].
- *gr krakensdr*: the software allowing the user to do the link between the KrakenSDR and GNU Radio. It's adding specific blocks, for example a source block [26].
- *kraken DoA*: the software we can use to do the DoA estimation. However, we will investigate later in the project if we will use it in order to make the DoA estimation or if we will implement our own solution [27].

3.2 The setup

The block diagram, representing the experimental setup used in this project, is illustrated in Figure 3.2. It consists of a KrakenSDR connected to a Raspberry Pi 4, via USB, which itself is connected to a laptop via an Ethernet cable. The KrakenSDR streams the five coherent IQ channels to the Raspberry Pi 4, which runs the *heimdall_ daq_ fw* calibration software. The Raspberry Pi 4 was chosen for its low power consumption, its compact form and its compatibility with the software developed around the KrakenSDR, especially *heimdall_ daq_ fw*. Its quad-core ARM Cortex-A72 processor at 1.8 GHz and 4 GB of RAM provides sufficient computational resources to run the calibration in real time [28]. Then, the processed data is accessible from the laptop through GNU Radio thanks to *gr krakensdr* or *osmocomsdr*, allowing connection between the Raspberry Pi 4 and the laptop through the different source blocks. GNU Radio is a free and open-source software that provides signal processing blocks to implement software-defined radios and signal processing systems. It can be used with external RF hardware such as the KrakenSDR or without hardware in a simulation environment. The chain is built by connecting functional blocks such as sources, filters, demodulators, and sinks through the graphical interface of GNU Radio Companion or through python where new blocs can be created. In this project, GNU Radio version 3.10.9.2 is used to receive and decode AIS messages from the raw IQ samples captured by the KrakenSDR.

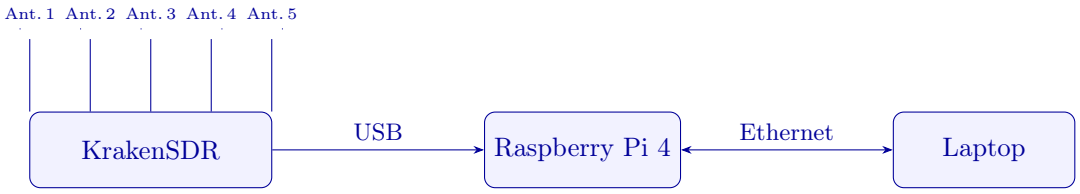


Figure 3.2: Overview of the experimental setup

3.3 Conclusion

In this first chapter the main features of the KrakenSDR were presented with its software. The setup used in this project was also introduced.

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

Chapter 4

Calibration of the KrakenSDR

In this chapter the two different methods to calibrate the KrakenSDR will be presented. Indeed, before performing any DoA estimation, the antenna array must be calibrated since each receiving channel has its own phase and amplitude offset, due to connector quality or SDR hardware variations. If it is left uncorrected, these offsets cannot be differentiated from the geometrical phase differences induced by the antenna arrays which are what DoA algorithms rely on. Without calibration the estimated bearing will be biased and will lead to the wrong estimated angle. The goal of the calibration procedure is to measure and compensate for these offsets. Therefore, only the phase difference observed is the one induced by the antenna array.

4.1 Calibration with *heimdall daq fw*

In this section the calibration of the device is done with *heimdall daq fw*. First, the functioning of *heimdall daq fw* will be explained and then the calibratiom will be tested.

4.1.1 Presentation of *heimdall daq fw*

This section starts with a presentation on the functioning of *heimdall daq fw*. Figure 4.1 shows the different blocks that *heimdall daq fw* is made of. First, there is the Realtek DAQ block, packing raw data of each channel into frames and sending it out at its standard output. Then the Sync block will align the bytes of each channel that could have been shifted previously. However, this does not mean that they are aligned in time The channels are still not coherent. After the sync block, the Rebuffer block follows. This one offers the possibility to use different sample sizes in the data acquisition part. Thus, there is the decimator block that is doing a decimation to decrease the sample rate of the processed signal in order to use less resources. It is also filtering the signal to prevent aliasing. There is also an Squelch block but its use is optional. It allows to capture the short bursts and rejects the useless portion of the full received RF signal. This improves the effectiveness of the program by analysing less signals[5].

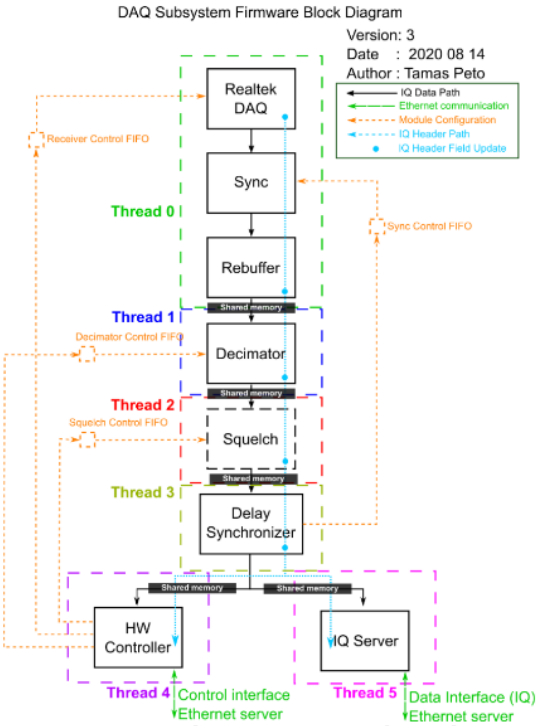


Figure 4. DAQ Subsystem Firmware block diagram

Figure 4.1: Firmware Block diagram [5]

Then, there is the Delay Synchronizer block that calculates and corrects the delay, phases and amplitudes of the error. The Figure 4.2 shows the different steps performed by the delay synchronizer. At first the KrakenSDR waits for the calibration signal to come in STATE INIT. In this case it is the internal noise source of the KrakenSDR. Then in STATE SAMPLE CAL the cross correlation between the standard channel (the reference channel, in our case we choose the channel 0) and all the other channels is calculated to determine their exact sample mismatch. After, the algorithm does an estimation of the peak. If the peak is not high enough, *heimdall daq fw* waits for other data. Otherwise the program then checks the position of the peak:

- if it is not centred at 0, the algorithm will calculate the delay and send this delay and the signal into SYNC WAIT. SYNC WAIT will correct the delay and send the signal back to STATE SAMPLE CAL to check if it's centered at 0.
- if it is centred at 0, the signal will go to the next step: STATE IQ CAL

In the STATE IQ CAL step, the algorithm verifies again that the cross correlation peak is centered at 0. Then it calculates the phase and amplitude errors with this peak and corrects them. Once the correction has been applied, the signal is going to the STATE TRACK LOCK. There, the signal is waiting for the noise source (the calibration signal) to be turned off and waits for the real signal to come. Finally, the signal is going to the STATE TRACK. There, the signal is transmitted to the HW Controller and the IQ server. However, during this phase the cross correlation peak between the standard channel and the others is still calculated. The algorithm still calculates a possible delay, phase or amplitude errors. If these errors are too important and it is happening three times in a row, the system goes back to STATE INIT, where the whole calibration starts again. The diagram of the algorithm is on Figure 4.2:

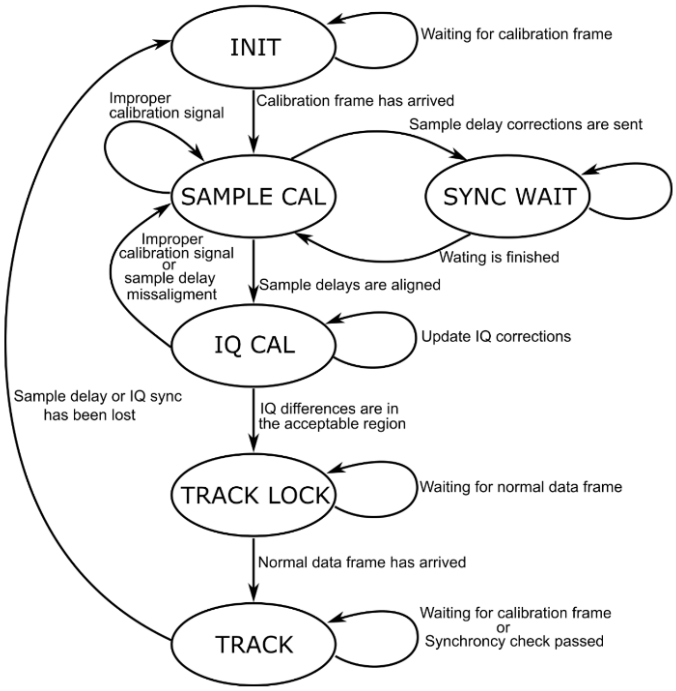


Figure 4.2: Delay synchronizer diagram [5]

After this process the signal is send to the HW Controller and the IQ server where the data is transmitted to other software like *gr krakensdr*[5].

4.1.2 Values for *heimdall daq fw*

Since *heimdall daq fw* is working, the right values in its configuration file (daq_chain_config.ini) need to be chosen so it fits the AIS signals. First, the center frequency is set at 162 MHz. Then a sample rate of 240 kS/s is chosen. However a sampling rate directly at 240 kS/s is not compatible with *heimdall daq fw* for different reasons.
First, Heimdall reads the samples in buffers of 262 144 samples. At 240 kS/s filling one buffer takes 1.09 s and the 12-buffer queue of *librtlsdr* can then hold more than 13 s of delayed data. Therefore, any processing delay accumulates. Second, *heimdall daq fw* aligns the channels iteratively and applies only one correction per buffer. At 240 kS/s, a buffer lasts 1.09 s so the alignment would progress very slowly. Third, the channels are considered aligned when the remaining fractional delay is below 0.03 samples which gives 125 ns at 240 kS/s. At this sample rate, the calibration state machine keeps oscillating around this threshold and never reaches the STATE TRACK which is the only state in which IQ data is sent to the output [5]. To avoid these problems, the sample rate is set at 1.2 MS/s and decimate by a factor of 5 to obtain 240 kS/s. So $samp\_rate = 1200000$ and $decimation = 5$.

Then the parameters of the FIR filter have to be set. The sample rate is 240 kS/s after decimation so the Nyquist criteria is at 120 kHz. The FIR relative bandwidth is set to 0.8 so a bandwidth of +/- 96 kHz (120 kHz x 0.8 = 96 kHz). This leaves a transition band of 24 kHz between the passband edge and the Nyquist limit, which is a standard practice in decimation filter design to prevent aliasing and preserving the useful signal bandwidth. A Hann window was chosen because it provides a good compromise between the width of the transition band and the attenuation of

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

the stopband [5]. Then the following parameters are defined: fir_relative_bandwidth = 0.8 and fir_window = hann.

Finally, the right Coherent Processing Interval (CPI) has to be chosen. It is set by default at $2^{20} = 1048576$. So it means that a frame is send every:

$$\frac{\text{cpi size x decimation ratio}}{\text{sample rate}} = \frac{2^{20}\text{x1}}{240000} \approx 4.3\text{s}. \tag{4.1}$$

If a cpi size = $2^{16} = 65536$ is chosen a frame will be send every:

$$\frac{\text{cpi size x decimation ratio}}{\text{sample rate}} = \frac{2^{16}\text{x1}}{240000} \approx 273\text{ms}. \tag{4.2}$$

The following setting is set: cpi_size = 65536. The right parameters of *heimdall daq fw* for AIS messages are now set.

4.2 Calibration with an osmocomSDR block

The calibration with *heimdall daq fw* is working and therefore the KrakenSDR source block that requires it can be used. However, an alternative was explored: another calibration with an osmocomSDR block source which uses the AIS signals as calibration signals.

4.2.1 Implementation of the new Calibration

The first problem encountered by using the osmocomSDR source block is that *librtlsdr* is designed for the KrakenSDR only and the osmocomSDR block is using classic *librtlsdr*. Additionally, some features are not accessible and are described in the following paragraphs.

A function that allows deactivate dithering: Each tuner of the KrakenSDR generates its local oscillator with a fractional-N Phase-Locked Loop (PLL). A fractional-N PLL produces the frequency of the local oscillator by multiplying a fixed reference clock by a non-integer factor N, which allows it to reach almost any frequency with a fine step such as 162 MHz in the case of AIS messages. By default, the PLL uses dithering. Dithering makes the switching of the PLL slightly random so it spreads out the spurs it creates in the spectrum. This is useful for a single receiver but the KrakenSDR has five channels so the dithering is different in each of the five channels. The phase of each local oscillator then varies randomly with respect to the others and the phase differences between the antennas, on which the DoA estimation relies, are no longer reliable. For this reason, the KrakenSDR library provides a function that disables dithering so that the five channels keep a stable phase relationship [29][30]. To determine the impact of the dithering, some test will be run on this new calibration with and without dithering. Regarding the result, it will be decided if it should be activated or deactivated.

Patches to avoid phase distortion caused by the tuners: Some modifications were also made regarding the tuners filters. The analog filter were turned off because it was causing phase distortion. For example in tuner_r82xx.c :

- reg_0a = IFi[i].reg10Lo -> 0x00

- reg_0b = IFi[i].reg11 -> 0x8F

And the ones filtering the noise and allowing a narrower bandwidth to have fixed values to avoid big differences between each channels so the channel are the most coherent possible [29][30]. However this settings could be defined as they are in the osmoSDR library because those filters hasn't been implemented in this library. If needed, their values will be changed.

Improvements of the tuners settings in order to avoid phase distorsion: The KrakenSDR version of the *librtlsdr* library changes the way the tuners are configured so that the five channels stay coherent. In the original osmocom version, each tuner chooses some of its internal settings by itself when it is tuned, such as the settings of its PLL and its mixer. Two tuners set to the same frequency can end up with different internal configurations which cause an unpredictable phase offset between them. The KrakenSDR library removes these automatic choices by forcing the same PLL, mixer, gain and filter settings. Also it forces the same initialisation sequence on all five tuners so they all process the signal in exactly the same way [29][30].

The main impact is a stable relative phase over time and across receivers. In comparison, the osmocom version prioritizes flexibility but lets internal tuner variations happen that can cause unpredictable phase offsets. Some tests should be run to know how much these changes improve the phase distortion problem.

Functions to activate and deactivate bias tee: The bias-tee, which is a DC voltage injector that allows the KrakenSDR to power an active antenna or low-noise amplifier directly through the coaxial cable, support in the KrakenSDR *librtlsdr* library differs from the osmocom *librtlsdr* library. The osmocom library has these functions but there are common to every channel while in the KrakenSDR one these functions allow handling each channel separatly. These functions are mainly used to activate and deactivate the noise source during the calibration. However, since the aim is to use the AIS messages in order to do the calibration, this change is not supposed to be an issue [29][30].

New function for the sampling frequency: The KrakenSDR version of *librtlsdr* also adds a finer correction of the sampling frequency. The original osmocom driver can only correct the sampling frequency in steps of one Parts Per Million (PPM). The KrakenSDR library adds the function `rtlsdr_set_sample_freq_correction_f()`, which accepts a non-integer value and can therefore correct the sampling frequency with a precision better than one PPM. A coherent system needs this precision since even a very small error on the sampling frequency makes the samples drift in time and degrades the phase measurements [29][30].

A calibration of the array based on the received AIS signals was also planned. For each decoded message, the reported position gives the expected direction of the vessel, which can be compared to the DoA estimated from the same signal. Repeated over many messages this comparison would give the correction to apply. This calibration was not implemented due to lack of time.

4.3 Conclusion

In this chapter the calibration with *heimdall daq fw* was explained. Then, the correct values for *heimdall daq fw* regarding the AIS messages were set. Additionally, a new calibration using AIS

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

signals as calibration signals instead of the internal noise source and the osmocomSDR source block was explored but not implemented.

Chapter 5

Receiving and decoding the AIS signals

In this chapter, the maximum distance between the setup and the AIS emitter in order to receive the AIS messages will be calculated. Then, the encoding and decoding process of AIS will be presented.

5.1 AIS signals reception range

To know the limit of the measurements it is important to know the maximum distance between the setup and AIS emitter in order to receive AIS. At 162MHz the radio propagation is governed almost only by Line of Sight (LoS) geometry because at this frequency the signal behaves essentially as an optical ray that bends slightly due to atmospheric refraction. In a perfect vacuum the radio horizon distance d for an antenna at height h above a spherical Earth of radius R_E is obtained with [31]:

$$d_{\text{vacuum}} = \sqrt{2R_E h + h^2} \approx \sqrt{2R_E h} \quad (h \ll R_E), \tag{5.1}$$

In the real atmosphere, the radio horizon is extended beyond the geometric horizon because the refractive index of air decreases with altitude leading radio waves to slightly bend towards the Earth's surface. This effect is modeled by replacing the true Earth radius R_E with an effective Earth radius noted $R_{\text{eff}} = kR_E$. Where $k = \frac{4}{3}$ for a standard reference atmosphere [32]. Then the formula for the distance in the troposphere is [31]:

$$d_{\text{tropo}} = \sqrt{2kR_E h} = \sqrt{\frac{4}{3}} \times \sqrt{2R_E h}. \tag{5.2}$$

For a communication link between a vessel with an AIS transmitter at height h_1 and the antenna array at height h_2, the total maximum distance is the sum of the two distances [31]:

$$d_{\text{total}} = d(h_1) + d(h_2). \tag{5.3}$$

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

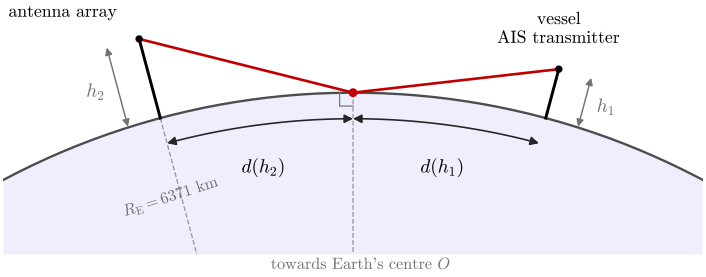


Figure 5.1: Maximum distance scheme

Table 5.1 presents the computed total range for two representative vessel antenna heights and three coastal station antenna heights with the vacuum and the troposheric models and $R_E = 6371$km:

Vessel antenna	**Coastal antenna**	**Vacuum** km	**Tropospheric** km
Large vessel (20 m)	10 m	27.3	31.5
Large vessel (20 m)	50 m	41.2	47.6
Large vessel (20 m)	100 m	51.7	59.7
Small boat (4 m)	10 m	18.4	21.3
Small boat (4 m)	50 m	32.4	37.4
Small boat (4 m)	100 m	42.8	49.5

Table 5.1: Estimated maximum AIS reception range for different vessel and coastal antenna heights computed with the vacuum and tropospheric models

As expected the reception range is larger with the tropospheric model and it is going from 18.4 km to 59.7 km depending on the scenario of the boats and the heights of the antennas.

5.2 Encoding processing

In this section, it will be explained how the AIS messages are encoded. The process can be observed in Figure 5.2:



Figure 5.2: AIS signal processing chain [6]

As it can be seen, the process of encoding is composed of four stages. The first one, the CRC (Cycle Redundancy Check), appends a checksum to data in order to detect transmission errors. Then there is the bit stuffing that is inserting a zero after every sequence of five consecutive ones. After, there is the NRZI (Non-Return to Zero Inverted) encoding in which a binary one is represented by a transition and a binary zero by the absence of transition, making the signal robust to polarity inversions. Finally, the GMSK (Gaussian Minimum Shift Keying) modulation,

which converts the digital bit stream into an analogue signal s(t) ready for transmission on the VHF channel. These steps will be explained in the next sections [7].

5.2.1 Cyclic Redundancy Check (CRC)

The CRC allows the receiver to verify if the frame has been received without errors. For the DoA estimation, this check is what allows the content of an AIS message to be trusted before it is compared to an estimated DoA. The CRC is composed of 16 bits and obtained and computed from n data bits. The n are seen as the coefficients of $D(x)$, either 0 or 1. Then, $D(x)$ is multiplied by x^16 to add 16 zeros at the end of the data, and divided by the generator polynomial of the High-Level Data Link Control (HDLC) standard [7]:

$$G(x) = x^{16} + x^{12} + x^5 + 1. \tag{5.4}$$

Since the bits are added modulo 2, addition and subtraction are the same operation known as the XOR. Therefore, the polynomial division is a long division in which each subtraction is replaced by an XOR. $G(x)$ has degree 16 so the remainder has at most 16 bits and this remainder is the CRC. This division is computed bit by bit with a 16-bit shift register which holds the remainder once the last data bit has been processed. The AIS CRC adds two operations to this remainder. First, the register is initialised to one on each bits, not zero. With a zero initial state, zeros at the beginning of the data would leave the register unchanged, and a frame whose start is shifted by zero would still pass the check. To be able to start with only ones a prefix has to be added to the register before the data. It is defined by the polynomial $P(x)$[7]:

$$\mathrm{CRC}(x) = \left(P(x)\, x^{n+16} + D(x)\, x^{16}\right) \bmod G(x). \tag{5.5}$$

Second, the CRC bits are inverted before being appended to the frame. Thus, when the receiver computes the CRC on the entire frame, data and received CRC, a frame without errors does not give zero but always the same constant:

$$\mathrm{CRC}\left([\mathrm{Data},\ \overline{\mathrm{CRC}(\mathrm{Data})}]\right) = \texttt{0xF0B8}. \tag{5.6}$$

A frame is accepted when this constant is obtained [7].

5.2.2 Bit stuffing

Each AIS frame is delimited by a start flag and an end flag which is: 01111110. This flag contains six consecutive ones so a receiver can locate the frame boundaries correctly only if this sequence never appears inside the frame. Bit stuffing guarantees that by scanning the data and the CRC and inserting a zero after every five consecutive ones. Moreover, the inserted bits carry no information [7].

Bit stuffing has a second benefit. Indeed, since every zero produces a level transition, the encoded data cannot stay at a constant level for more than six bit periods, which helps the receiver to keep its bit synchronisation [7].

5.2.3 Non-Return to Zero Inverted (NRZI)

NRZI is a differential code because the value of a bit is given by what happens to the signal level and not by the level itself. In the convention used by AIS a zero means that there is level

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

change and one means that there is none. Denoting by a_k the input bits and by b_k the encoded bits, the encoder calculates [7]:

$$b_k = \overline{a_k \oplus b_{k-1}}, \tag{5.7}$$

where $\oplus$ is a modulo 2 operation. The decoder only has to compare two consecutive levels, $a_k = \overline{b_k \oplus b_{k-1}}$, so a global inversion of the polarity of the signal has no effect on the decoded bits [7].

5.2.4 Gaussian Minimum Shift Keying (GMSK) modulation

GMSK is a continuous-phase modulation. It means that the signal has a constant envelope and the information is carried by its phase which continuously evolves. In complex baseband, the AIS signal can be written as [7]:

$$s(t) = e^{-j\phi(t;\mathbf{I})}, \qquad \phi(t;\mathbf{I}) = \pi \sum_{k=-\infty}^{n} I_k \, q(t - kT), \quad nT \leqslant t \leqslant (n+1)T, \tag{5.8}$$

where $1/T = R_b = 9600$ bit/s is the bit rate and $I_k \in \{-1, +1\}$ are the symbols obtained from the output of the NRZI process. The factor π corresponds to a modulation index $h = 0.5$ because each bit rotates the phase by $\pm\pi/2$. The phase pulse $q(t)$ is the integral of a frequency pulse $g(t)$, obtained by filtering a rectangular pulse of duration T with a Gaussian filter of -3 dB bandwidth B:

$$q(t) = \int_{-\infty}^{t} g(\tau)\, d\tau, \tag{5.9}$$

$$g(t) = \frac{1}{2T}\left[Q\left(\frac{2\pi B}{\sqrt{\ln 2}}\left(t - \frac{T}{2}\right)\right) - Q\left(\frac{2\pi B}{\sqrt{\ln 2}}\left(t + \frac{T}{2}\right)\right)\right], \tag{5.10}$$

where $Q(\cdot)$ is the tail function of the standard normal distribution. For AIS, $BT = 0.4$ and the pulse is truncated to $L = 3$ bit periods so each bit influences the phase over three bit periods [7].

5.3 Decoding process

In this section, the decoding chain and the result of the decoding chain is presented. It is important to decode the AIS messages since it will be necessary to have access to the ships ID and positions to compare the DoA estimation with their reported position. For that a chain on GNU Radio is implemented and tested with the KrakenSDR, calibrated with *heimdall daq fw* and an AIS emitter. The Figure 5.3 shows this chain for channel A. After the source block a FIR Xlating filter is added with center frequency at 161.975 MHz and a decimation at 5. It shifts the signal of interest from its offset frequency down to baseband and decimates the sample rate the use less of computational ressources. With a sample rate of 240 kHz and a decimation of 5 the new sample rate is 48 kHz. Then, the Quadrature Demodulation block comes, which is doing the demodulation of the signal. The gain is [33]:

$$\text{Gain} = \frac{1}{\Delta\Phi}, \tag{5.11}$$

with :

$$\Delta\Phi = 2\pi\frac{\Delta f}{f_s} \text{ and } \Delta f = \frac{hR_b}{2}, \tag{5.12}$$

here Δf the frequency deviation, $\Delta\Phi$ the phase difference, R_b the bit rate, h the modulation index and f_s the sample rate frequency. In the case of a GMSK modulation for AIS message the standard values are $h = 0.5$, $R_b = 9600 bits/s$ which leads to:

$$\Delta f = \frac{0.5 \text{ x } 9600}{2} = 2400\text{kHz}, \tag{5.13}$$

furthermore after the decimation of the filter $f_s = 48kHz$ so the followings result is:

$$\text{Gain} = \frac{1}{\Delta\Phi} = \frac{f_s}{2\pi\Delta f} = \frac{48000}{2\pi \text{ x } 2400} \approx 3.18. \tag{5.14}$$

Then the Clock Recovery block comes. This block is needed because the clock between the emitter and the receiver is not exactly the same. Both of them are supposed to work at 9600 bits/second but there is always a little gap between the two oscillators. Without correction, sampling window of the receiver drifts from the optimal position and could read bits at the wrong moment and would cause decision errors. It is set that Mu = 0.5, Gain Mu = 0.175 and Omega relative = 0.005 because they are standard values. Therefore [34] :

$$\Omega = \frac{f_s}{R_b} = \frac{48000}{9600} = 5, \tag{5.15}$$

$$\text{Gain Omega} = \frac{\text{Gain Mu x Gain Mu}}{4} = \frac{0.175^2}{4} \approx 0.008. \tag{5.16}$$

After a Binary Slicer is added, it is a decision block that converts the analogue-valued output of the Clock Recovery into a binary bit stream. Any value larger than zero is mapped to a binary one and any value smaller or equal to zero is mapped to a binary zero. It is the final step of the demodulation process, producing the raw bit stream that is then passed to the AIS decoding block.

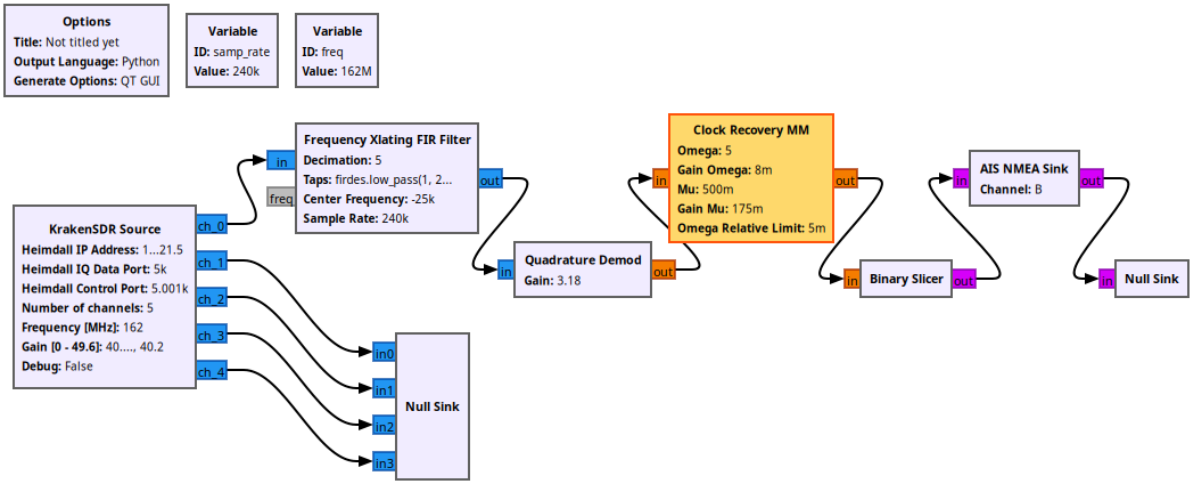


Figure 5.3: GNU Radio chain to decode the AIS messages

Finally, there is the AIS decoding block. It first starts by inverting the NRZI process. Then it looks for the beginning flag of the AIS message: 01111110. After that it takes all the bits from this flag until the end flag, which is also 01111110. Therefore, it removes the bits appended by the bit stuffing and verifies the correct length of the message. Finally, the CRC is verified to

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

ensure that no errors were made during the transmission. If the CRC is correct the message is formatted as a National Marine Electronics Association (NMEA) sentence. NMEA is the standard communication protocol used in the maritime domain to exchange navigation data between onboard instruments. Here is the example of the sentence that was found using an AIS emitter:

!AIVDM,1,1,,B,<eLtkjtP‘<6uSa‘G0PR7t?t11IFEuaj380R280R280P200000004,0*38
!AIVDM,1,1,,B,<eLtkjtP0<6uE‘TGw8n7t?t11IFEuaj380R280R280P200000004,0*58

A functioning receiving and decoding chain was implemented for the AIS messages. The NMEA message obtained can be decoded with the library *pyAIS*[35].

5.4 Conclusion

In this chapter, the AIS reception range was presented. Then, the AIS signal encoding process was explained. Moreover, a GNU Radio chain to decode them was proposed. Receiving and decoding those messages correctly is very important for the next parts of the project in order to estimate the position of the vessels thanks to the position they are transmitting and compare it to the DoA estimation.

Chapter 6

Setup for the DoA

In this chapter the detection of the burst will be presented first. Then the different array configuration will be introduced and then compared.

6.1 Burst Detection

Before running MUSIC the AIS have to be registered. So a GNU Radio chain was implemented to record every burst detected by the KrakenSDR. In theory AIS message should be send one by one especially because they are using anti collision protocol. The GNU radio Radio chain is on the Figure 6.1:

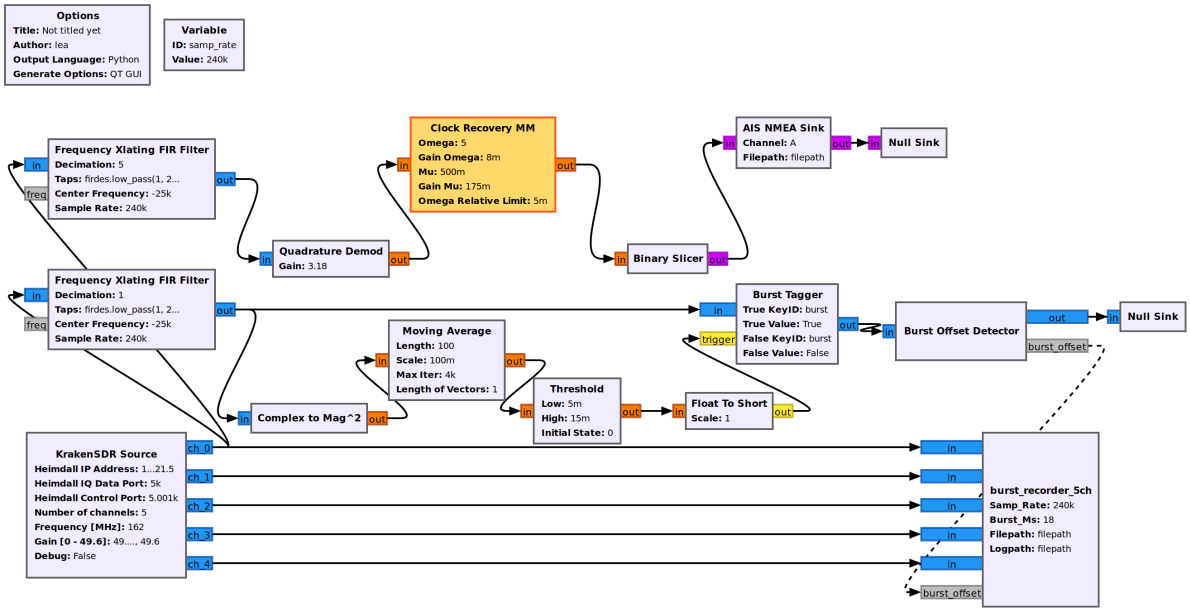


Figure 6.1: GNU Radio chain of the burst detector

The main idea is to detect a burst on channel 0, after which the signal from the five channels will be recorded for 18 ms. The detection relies on a Burst Tagger which has two inputs. The first one is the signal of antenna 0 after a Frequency Xlating FIR filter, which shifts the signal to baseband and filters it with a low-pass at $f_s = 240$ kHz. The second input is the trigger signal, which is derived from the same filtered signal. A Complex to Mag^2 block first computes

the instantaneous power $|x[n]|^2$. Then a Moving Average filter of length $N_{samples} = 100$ samples and scale 0.1 then smooths this power over $N_{samples}/f_s \approx 0.42$ ms after. Hereby, fast amplitude fluctuations are removed, while the burst's envelope is still maintained. Therefore its output is ten times the mean power over the window. After that a Threshold block turns this estimate into a binary decision. Indeed its output switches to 1 when the smoothed power exceeds 0.015 and returns to 0 only when it goes below 0.005. Then a Float To Short block converts this 0 or 1 signal into the integer type expected by the trigger input of the Burst Tagger. Finally, when the trigger switches from 0 to 1, the Burst Tagger attaches a tag (key=burst, value=True) to the data. When the Burst Offset Detector receives the first tag it sends a signal to the Burst Recorder block that will start to record the signal during 18 ms for each channel. Indeed, MUSIC is the algorithm that will be used to perform the DoA as explained in Chapter 2. This algorithm is using the cross correlation matrix $\hat{R}$ defined in Chapter 2:

$$\hat{R} = \frac{1}{N} \sum_{n=1}^{N} X[n]\, X[n]^H. \tag{6.1}$$

$\hat{R}$ requires at least $N \geq M$ snapshots and $N \gg M$ so that the signal and noise subspaces are estimated accurately. Also, the AIS messages that carry the position information, last 24.2 ms. In order to be sure to only capture a part of the AIS signal, only 18 ms will be recorded. The number of snapshots for a complex signal is [36]:

$$N = 2 \times \frac{B}{2} \times T = B \times T, \tag{6.2}$$

with $B = 25$ kHz and $T = 18$ ms it gives:

$$N = 25000 \times 0.018 = 450 \gg 5. \tag{6.3}$$

This recording will then be used to do the DoA estimation.

6.2 Antenna arrays

In this section the result of MUSIC with different antenna pattern are studied: a linear array, also called ULA in this project because the spacing between each antenna is the same, a pentagon array and cross array. Also the parameters of the MUSIC algorithm for this project are: $M = 5$ and $N = 450$. Moreover the wavelength for AIS signal is $\lambda = 1.85$m.

6.2.1 ULA

The ULA is the first one being tested. As shown in Figure 6.2, the five antennas are placed on a metal bar. Figure 6.3 shows the scheme of the setup:



Figure 6.2: ULA setup at DLR

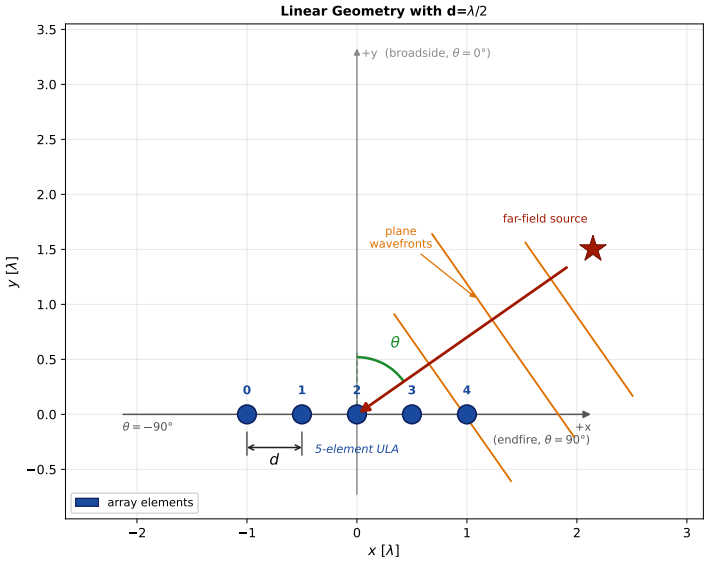


Figure 6.3: ULA setup scheme

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

Even though the implementation of the ULA is simple, there are two main problems. The first one is the ambiguity that results from the spectrum going from -90° to 90°. The second one is that the ULA does not provide uniform DoA estimation performance across all angles. In fact, the accuracy actually degrades significantly when the angle of incidence approaches the endfire direction ($\theta = \pm 90$) [37]. To illustrate this degradation, the two estimators RMSE, which is determined by Monte Carlo simulation and the CRB, that is calculated analytically, can be used.

The RMSE is computed by running $K_{\text{trials}} = 500$ independent Monte Carlo trials for each angle of incidence θ and each SNR. In each trial, an AIS signal is generated with $N = 450$ snapshots, $M = 5$ array elements spaced at $d = 0.4\lambda$ and a white Gaussian noise. The MUSIC algorithm is applied to this signal to obtain an estimate $\hat{\theta}_i$ of the true angle for each trial i. The RMSE is then defined as follows:

$$\text{RMSE}(\theta) = \sqrt{\frac{1}{K_{\text{trials}}} \sum_{i=1}^{K_{\text{trials}}} \left(\hat{\theta}_i - \theta\right)^2} \tag{6.4}$$

Unlike the CRB, the RMSE measures the empirical behavior of the MUSIC estimator, which is including all effects that occur outside the asymptotic regime, for example, at a low SNR or near the endfire.

The CRB is the theoretical lower bound on the variance of any unbiased estimator of θ. For a CRB with a single uncorrelated narrowband source, the corresponding expression is [38]:

$$\sqrt{\text{CRB}(\theta)} = \sqrt{\frac{6}{\text{SNR} \cdot N \cdot (2\pi d/\lambda)^2 \cdot \cos^2\theta \cdot M(M^2-1)}} \tag{6.5}$$

Figure 6.4 plots the RMSE and CRB for a ULA at a distance of $d = 0.4\lambda$, with $M = 5$ antennas, $N = 450$ snapshots, $K_{trials} = 500$ and different SNR values:

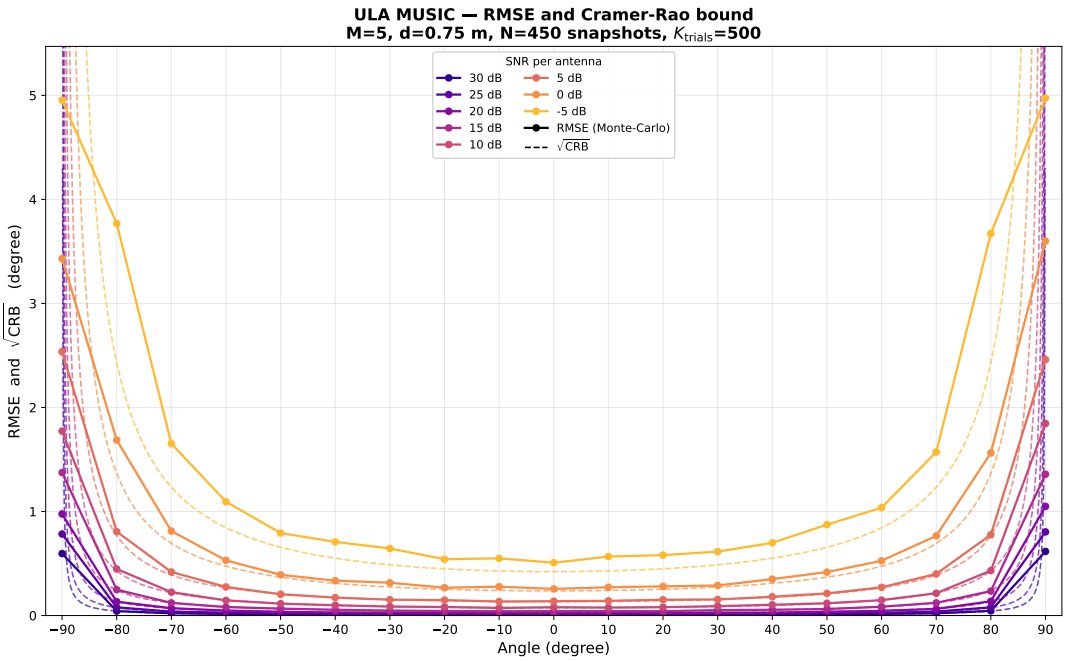


Figure 6.4: ULA RMSE and CRB for different level of SNR

The solid lines show the RMSE of the DoA estimates, and the dashed lines show the corresponding square root of the CRB given in Equation (6.5). As explained before the more the angle is close to the endfire the less MUSIC algorithm will be precise. Also if the SNR is lower the performances of MUSIC are affected. In order to test the limit of the setup presented on figure 6.2 two simulations were made with the AIS emitter at the following position: 10° and 85°. Also different values of SNR will be tested. Moreover in order to select the optimal distance between each antenna, different spacing were tested. But first the maximum d that can be selected has to be calculated. A plane wave is arriving from an angle θ that travels an extra distance $d\sin\theta$ between two adjacent antenna so it produces a phase shift:

$$\varphi(\theta) = \frac{2\pi d}{\lambda}\sin\theta. \tag{6.6}$$

MUSIC is using the θ from this phase shift, but a phase is only measurable modulo 2π so the array will alias if the phase shift exceeds one full turn. For $\theta \in [-90°, 90°]$:

The continuous lines show the RMSE of the DoA estimates, and the dashed lines show the corresponding square root of the CRB given in Equation (6.5). As explained before, the closer the angle is to the endfire, the less precise the MUSIC algorithm is. If the SNR is lower, the performance of MUSIC is also affected. In order to test the limitations of the setup shown in Figure 6.2, two simulations are made with the AIS emitter at positions 10° and 85°. Additionally, different values of SNR are tested. To determine the optimal distance between each antenna, different distances are also tested. However, first, the maximum d that can be selected, has to be calculated. A plane wave that is arriving from an angle θ, travels an extra distance $d\sin\theta$ between two adjacent antennas and therefore produces a phase shift:

$$\varphi(\theta) = \frac{2\pi d}{\lambda}\sin\theta. \tag{6.7}$$

MUSIC uses the from this phase shift. Since a phase is only measurable modulo 2π, the array will alias if the phase shift exceeds one full turn. For $\theta \in [-90°, 90°]$:

$$-1 \leq \sin\theta \leq 1 \qquad \Longrightarrow \qquad -\frac{2\pi d}{\lambda} \leq \varphi(\theta) \leq \frac{2\pi d}{\lambda}, \tag{6.8}$$

so φ total width is $\frac{4\pi d}{\lambda}$. As explained before the width can't exceed 2π:

$$\frac{4\pi d}{\lambda} \leq 2\pi, \tag{6.9}$$

which gives:

$$d \leq \frac{\lambda}{2} \tag{6.10}$$

with $f = 162$ MHz it gives $d_{max} = 0.93$ m. To illustrate that a theoretical simulation is done with the following d: 2λ, λ, $\frac{\lambda}{2}$, 0.4λ. The parameters are $N = 450$ and $K_{trials} = 500$ for the Monte-Carlo simulation on Figure 6.5:

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

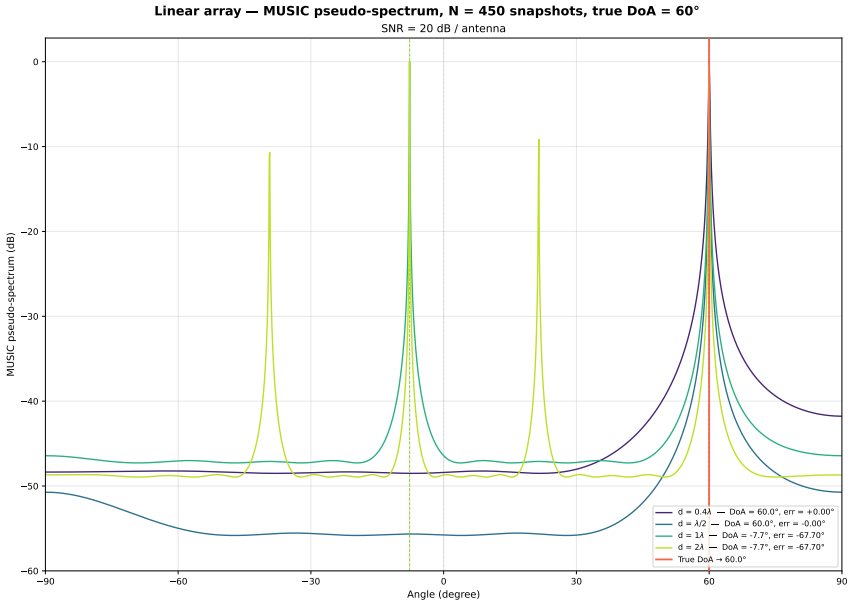


Figure 6.5: Theoretical MUSIC spectrum for an angle of 60°

As explained for $d = 2\lambda$ and $d = \lambda$ the phenomenon of aliasing is observed with the appearance of new peaks that can lead to a wrong DoA estimation. Therefore, the four following d are selected: $\frac{\lambda}{2}$, 0.4λ, $\frac{\lambda}{4}$ and $\frac{\lambda}{6}$.
At first, the four distances are simulated with an angle of 10° from the broadside. Figure 6.7 shows the theoretical result that should be obtained with the four different distance d, $N = 450$, and the four different values of SNR: 15 dB, 20 dB, 25 dB and 30 dB. A Monte-Carlo simulation of $K_{trials} = 500$ is run for each simulation. It can be seen that the MUSIC algorithm shows a good performance in all four cases for any value of d, especially when the SNR is high. In fact, all the peaks are sharp and almost at 10°. However, $d = \frac{\lambda}{6}$ leads to the biggest errors in the case of a lower SNR. The errors are quite small, between 0.01° and 0.05°. This is coherent with the CRB and RMSE values that were calculated before.
Then, a simulation with an angle of 85° is made. Since this angle is close to the endfire, according to the Figure 6.4, the errors should be higher. As in the previous simulations, the parameters for the Monte-Carlo simulation are $N = 450$ and $K_{trials} = 500$ and the spectrum is shown in Figure 6.6. These plots correspond to Figure 6.4. Indeed, the errors increase near the endfire, specifically when the SNR is lower for $d = \frac{\lambda}{4}$ and $d = \frac{\lambda}{6}$. Furthermore, d=2 and d=0.4 give the best performance.
In general, $d = \frac{\lambda}{2}$ and $d = 0.4\lambda$ give the best performance for the different values of SNR and near the endfire. For space reasons, $d = 0.4\lambda$ is chosen.
Finally, a simulation is made to evaluate the impact of geometric errors. The position of the antennas in the array may present small geometric errors when MUSIC uses their exact position. In order to evaluate the sensitivity of geometric errors, four simulations are made for different SNR levels with $d = 0.4\lambda$, $N = 450$ snapshots, $K_{trials} = 100$ Monte-Carlo trials and an error of position of 1%, shown in Figure 6.8. This error corresponds to an error of 1% in spacing and is randomly distributed along the x and y axis. Then, the signal is generated with the position offset. After that, the algorithm computes the DoA with the correct position in MUSIC, however, with the signal generated with a position offset. As it can be seen, the errors caused by the geometry are higher for a SNR of 15 dB and 20 dB. However, since the errors are less than one degree, small errors in the geometry are not causing large errors on the DoA estimation.

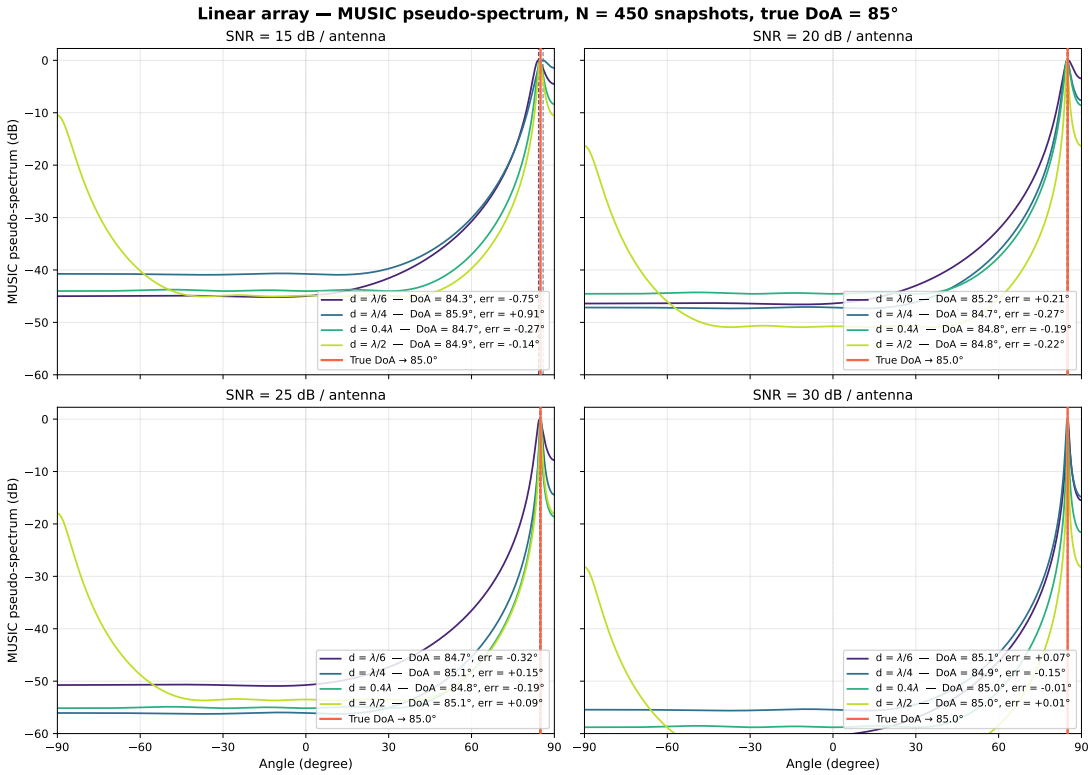


Figure 6.6: Theoretical MUSIC spectrum for an angle of 85° and different SNR

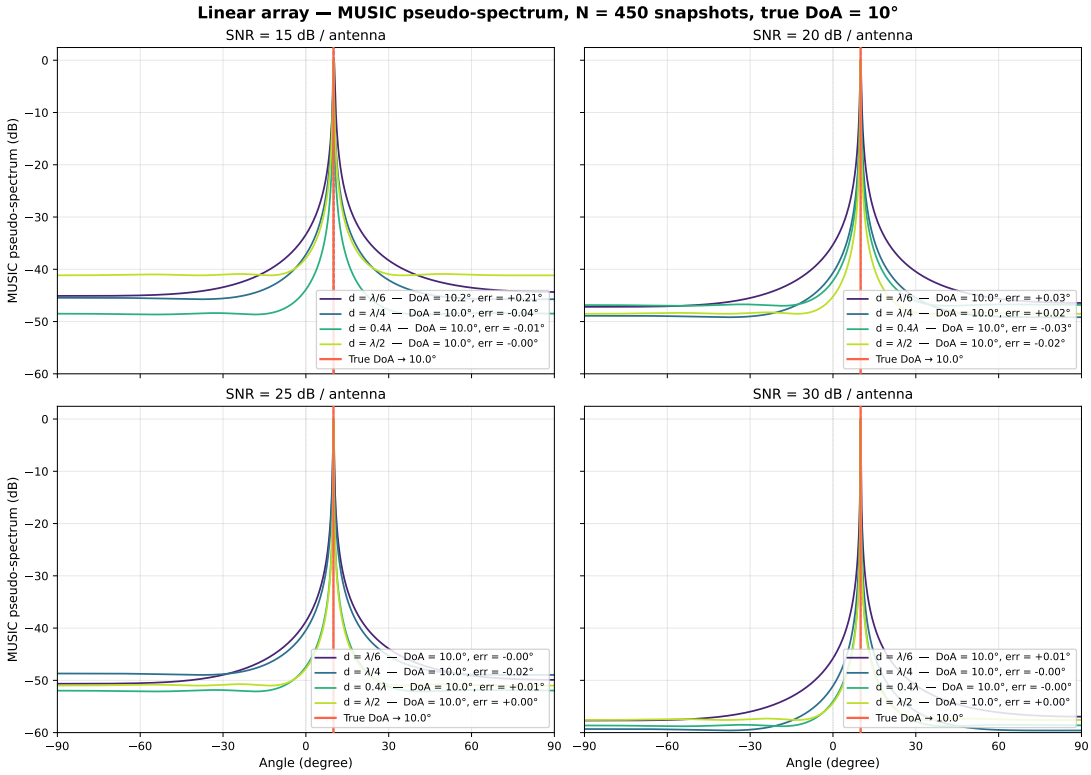


Figure 6.7: Theoretical MUSIC spectrum for an angle of 10°, different SNR and different spacing d

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

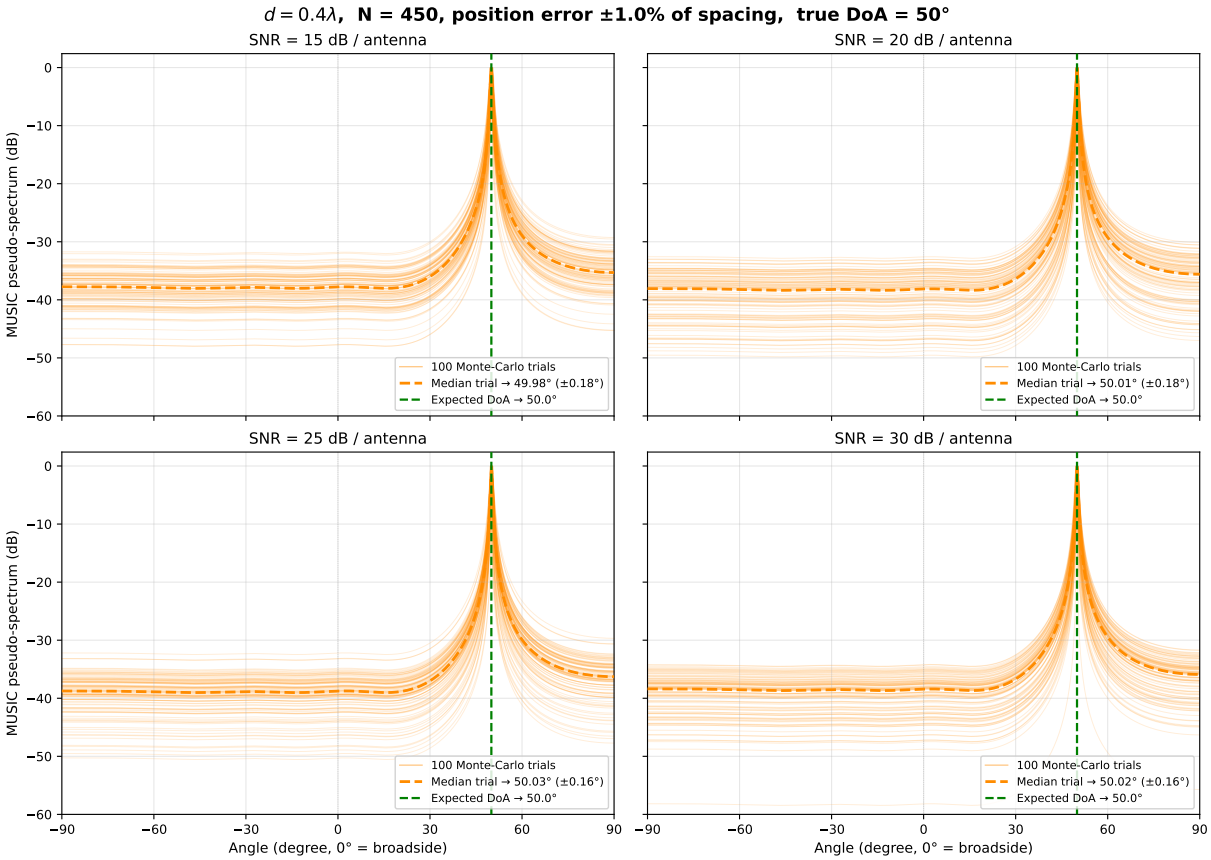


Figure 6.8: MUSIC spectrum for an ULA for an errors of position of 1%

6.2.2 Pentagon array

The second antenna array being tested is the pentagon. As shown in Figure 6.9, the five antennas are placed on a wooden plank. Figure 6.10 shows the scheme of the setup:

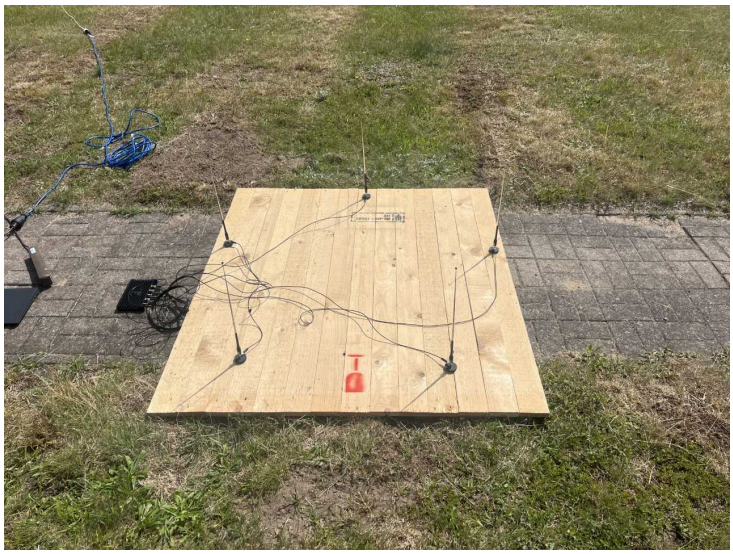


Figure 6.9: Pentagon array setup at DLR

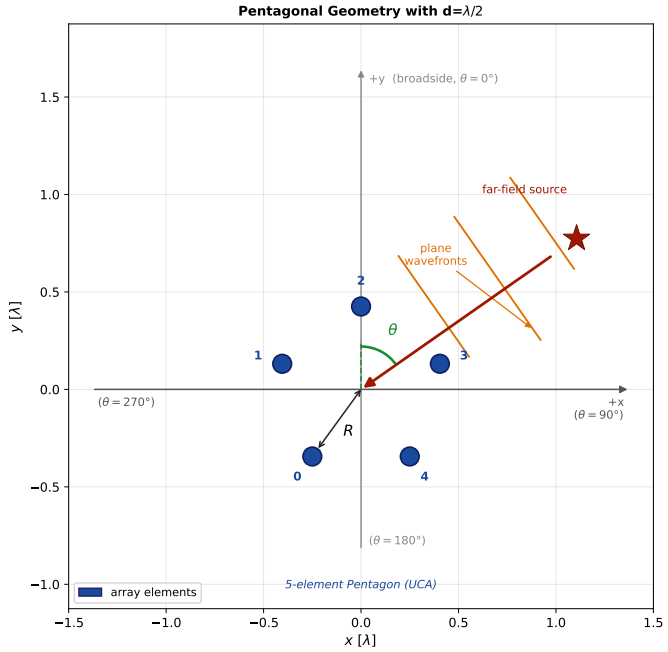


Figure 6.10: Pentagon array setup scheme

The main advantages of this setup are the absence of ambiguity and the consistent performance of the MUSIC algorithm. To estimate the performance, the RMSE and the CRB are calculated. For the RMSE, Monte Carlo simulations for the pentagon are done with the same parameters as for the ULA. The CRB is obtained from the expression [38]:

$$\sqrt{\mathrm{CRB}(\theta)} = \sqrt{\frac{1}{2K \cdot \mathrm{SNR} \cdot h(\theta)}}, \qquad h(\theta) = \dot{A}^H(\theta)\, E_n E_n^H\, \dot{A}(\theta) \tag{6.11}$$

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

where $A(\theta)$ is the steering vector, $\dot{A}(\theta) = \partial A(\theta)/\partial\theta$ is the derivative of the steering vector, and E_n the noise subspace. Furthermore $E_n E_n^H = I_M - A(\theta)\left[A^H(\theta)A(\theta)\right]^{-1} A^H(\theta)$ is the orthogonal projector onto the noise subspace and $a_m(\theta) = \exp\left[j\frac{2\pi}{\lambda}\left(x_m \sin\theta + y_m \cos\theta\right)\right]$, $m = 0, \ldots, M-1$ and (x_m, y_m) is the known position of the M antenna.
The geometry is isotropic so $h(\theta)$ is constant and can be calculated. Thus $h(\theta) = h \approx 11.75$ for every incidence angle θ, making the resulting CRB independent of θ:

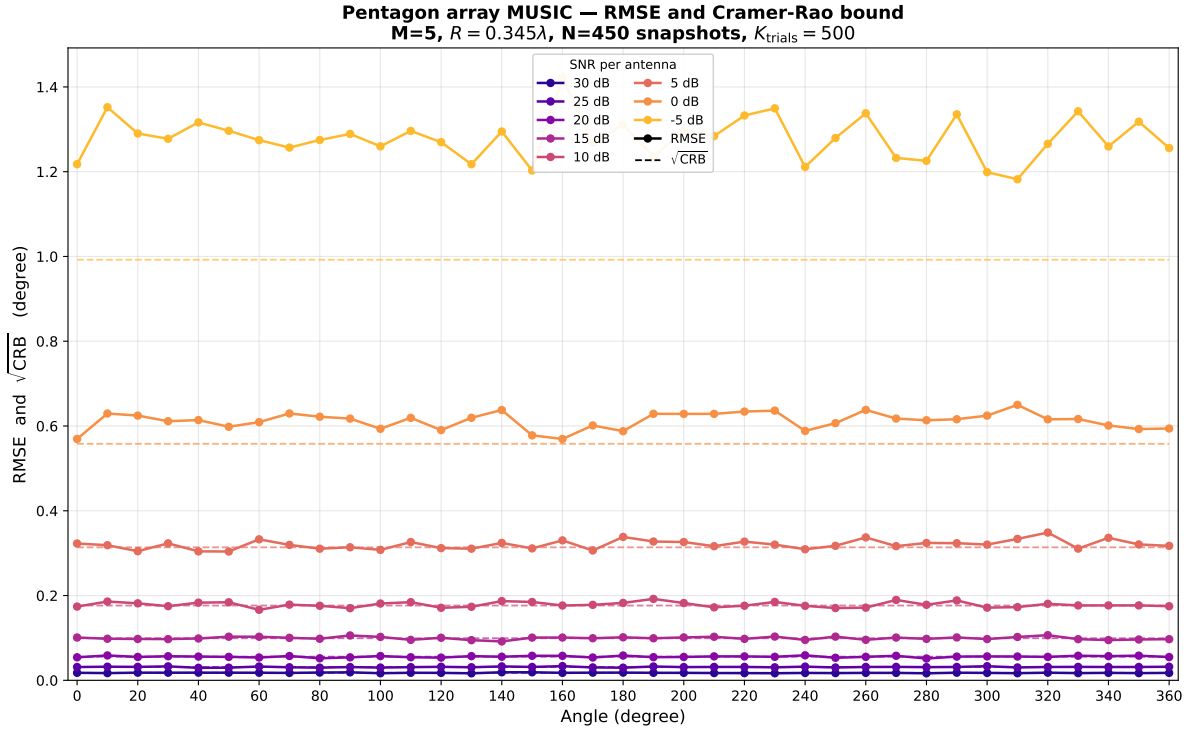


Figure 6.11: Pentagon array RMSE and CRB for different level of SNR

Compared to the ULA, the CRB is constant and the RMSE is not depending on the angle. Additionally, the CRB is lower than the RMSE, since the CRB is the deterministic bound and is therefore the lower bound for any other unbiased estimator. Moreover, up until 65°, the two estimators are both higher than the one for the ULA. After that it is becoming higher for the ULA because it is closer to the endfire. Therefore, the errors of the pentagon are not depending on the angle, since the geometry is isotropic, but the errors are slightly higher near the broadside than with the ULA.
As the geometry is isotropic, only one theoretical simulation with an angle of 10° is performed. The maximum distance d remains the same:

$$d \leq \frac{\lambda}{2} \tag{6.12}$$

However $d = 2R\sin(\pi/5)$ and therefore the condition $d \leq \lambda/2$ becomes:

$$R \leq \frac{\lambda}{4\sin(\pi/5)} \approx 0.425\,\lambda \tag{6.13}$$

To illustrate that a theoretical simulation with the following R is performed: λ, 0.54λ, 0.425λ, 0.345λ. The parameters used for the Monte-Carlo simulation shown in Figure 6.12 are $N = 450$ and $K_{trials} = 500$:

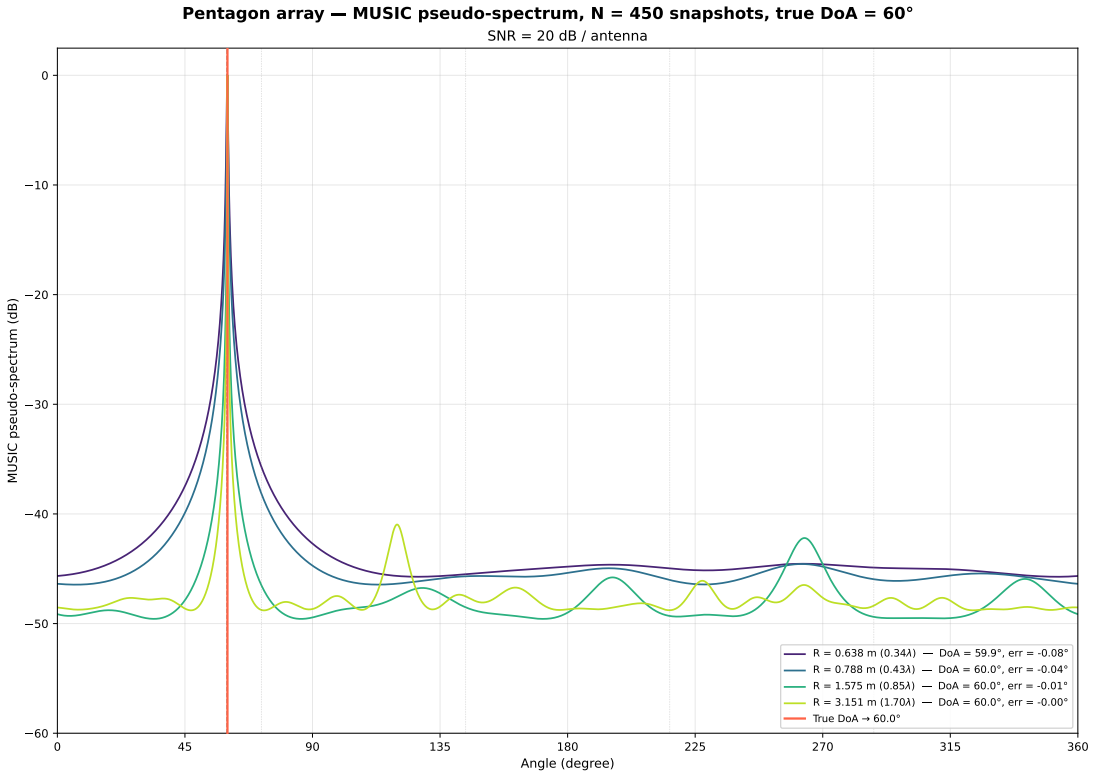


Figure 6.12: Theoretical MUSIC spectrum for an angle of 60°

For R over 0.425λ some undulations are appearing and could cause secondary peak. Therefore the distance needs to stay under 0.425λ. However, it can be noted that those undulations are not as strong as for the ULA and no second peaks are appearing.
The four distances tested are: $R = 0.17\lambda$, $R = 0.25\lambda$, $R = 0.345\lambda$ and $R = 0.425\lambda$. For the simulation, similar to the ULA, the parameters that are used for the Monte-Carlo simulation shown in Figure 6.13 are $N = 450$ snapshots and $K_{trials} = 500$. Thus, the higher the SNR, the less important are the errors, especially for smaller R. The values of R that deliver the best overall performances are $R = 0.345\lambda$ and $R = 0.425\lambda$. However, due to space limitation $R = 0.345\lambda$ is selected.
Finally, as shown in Figure 6.14, the impact of the error of 1% in the geometry is simulated with $N = 450$ and $K_{trials} = 100$ Monte-Carlo simulations. They are simulated with the same method as for the ULA. The only difference is the simulation of the errors on R and not on d. As with the ULA the impact of the geometry is less than 1° regardless of the SNR and is not causing any significant errors in the direction of arrival.

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

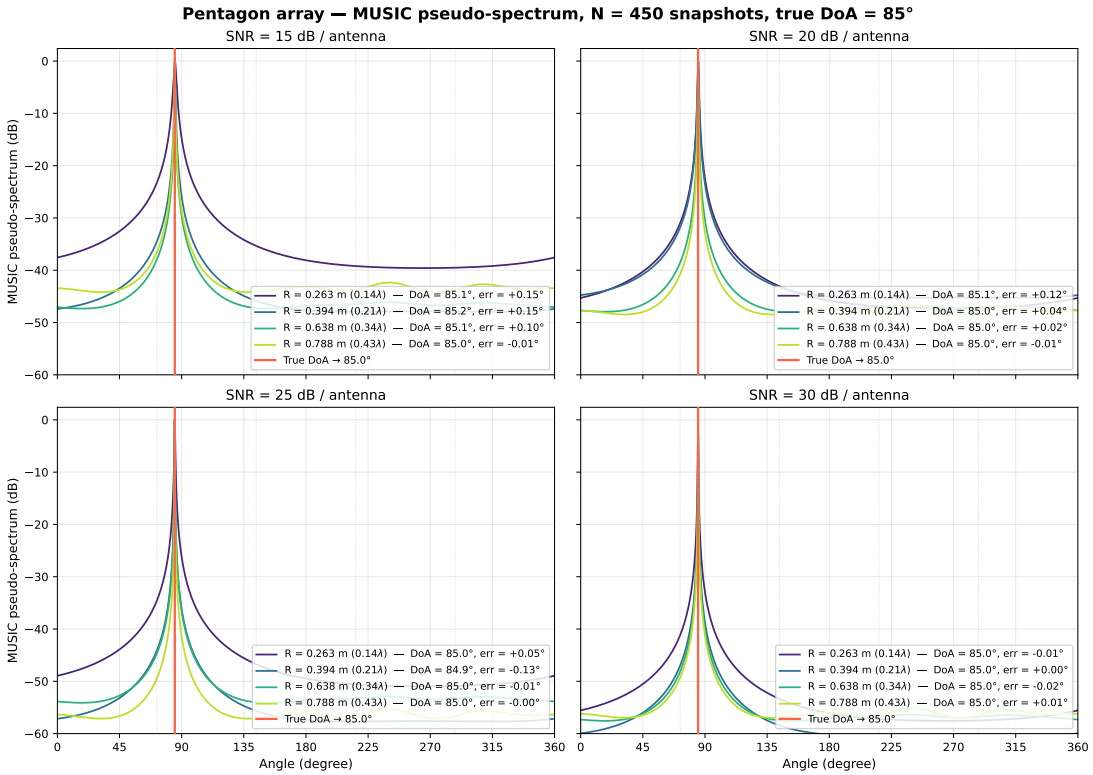


Figure 6.13: Theoretical MUSIC spectrum for an angle of 85° and different SNR

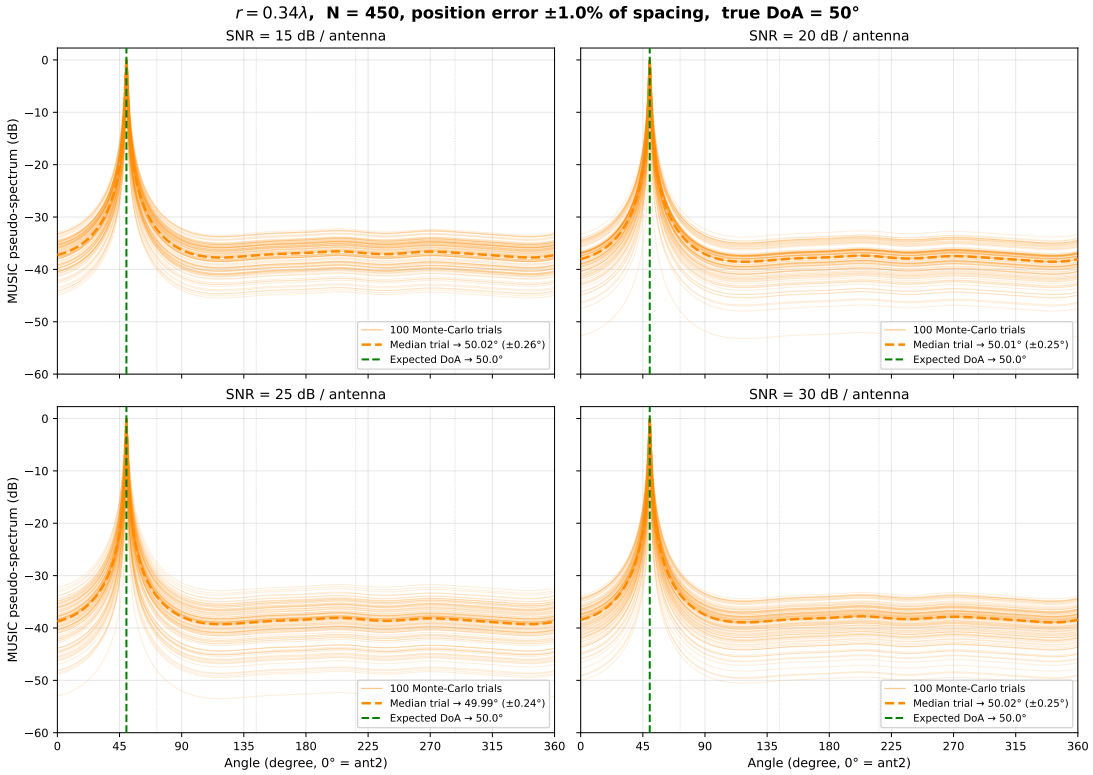


Figure 6.14: MUSIC spectrum for the pentagon array for an errors of position of 1%

6.2.3 Cross array

Finally, the cross array is tested. The setup is done on the same wooden plank as the pentagon array and it is shown with its scheme in Figure 6.15 and Figure 6.16:



Figure 6.15: Cross array setup at the first measurement campaign

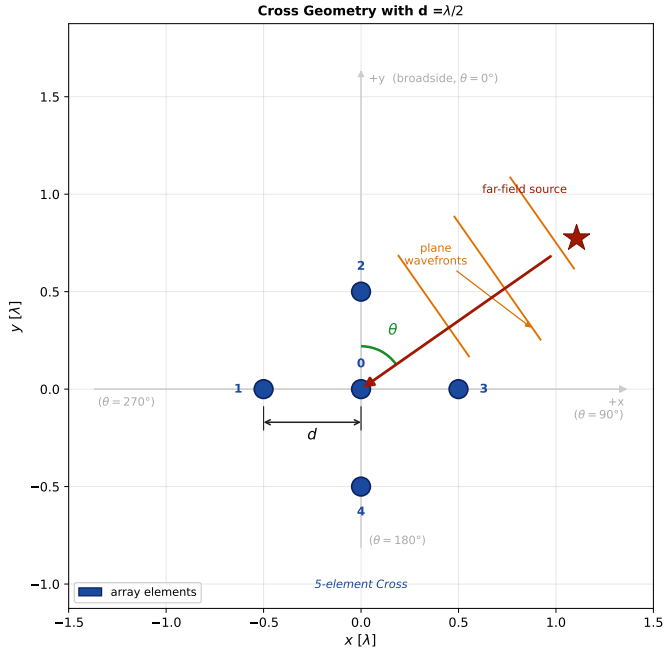


Figure 6.16: Cross array setup scheme

This array presents the same advantages as the pentagon one. The RMSE Monte Carlo simulations for a cross array are done with the same parameters as for the ULA. Moreover, the CRB is obtained from the expression [38]:

$$\sqrt{\mathrm{CRB}(\theta)} = \sqrt{\frac{1}{2K \cdot \mathrm{SNR} \cdot h(\theta)}}, \qquad h(\theta) = \dot{A}^H(\theta)\, E_n E_n^H\, \dot{A}(\theta) \tag{6.14}$$

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

where $A(\theta)$ is the steering vector, $\dot{A}(\theta) = \partial A(\theta)/\partial\theta$, and E_n the noise subspace. Also $E_n E_n^H = I_M - A(\theta)\left[A^H(\theta)A(\theta)\right]^{-1} A^H(\theta)$ is the orthogonal projector onto the noise subspace and $a_m(\theta) = \exp\left[j\frac{2\pi}{\lambda}\left(x_m \sin\theta + y_m \cos\theta\right)\right]$, $m = 0, \ldots, M-1$ and (x_m, y_m) is the known position of the M antenna. Since the geometry is isotropic, $h(\theta)$ is constant so $h(\theta) = h \approx 12.63$ for every incidence angle θ. The following graph is obtained:

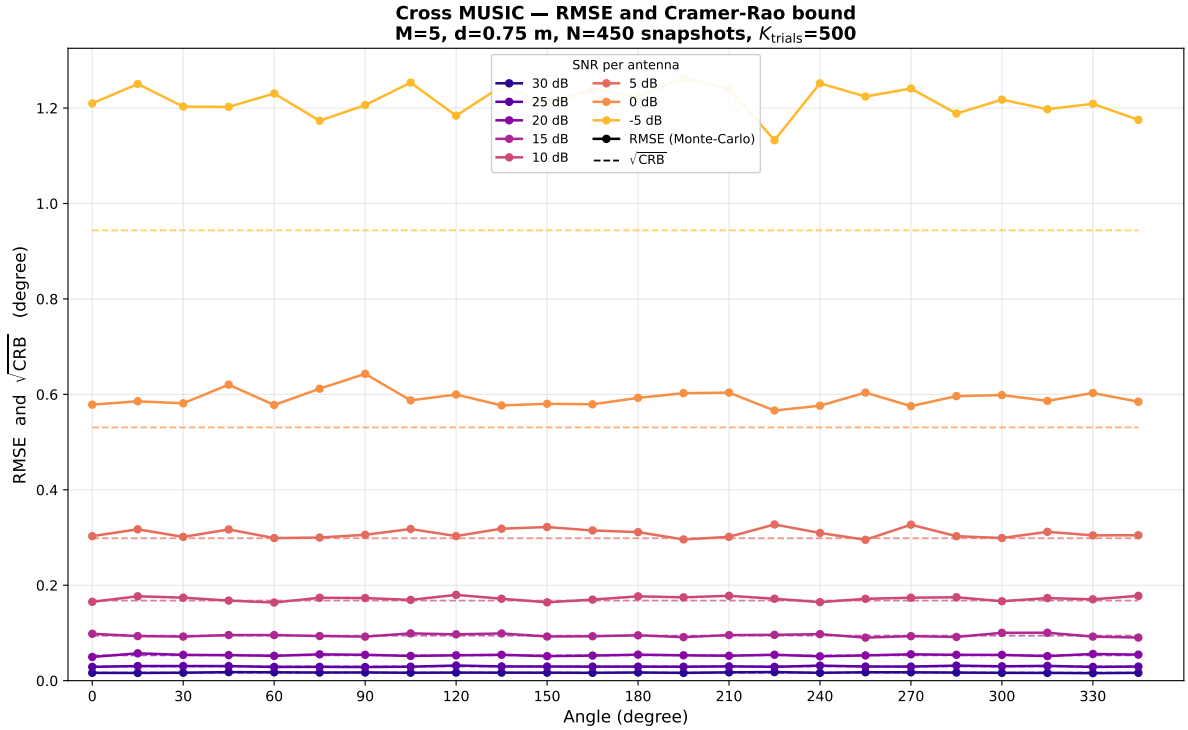


Figure 6.17: Cross array RMSE and CRB for different level of SNR

As for the pentagon array, the CRB is constant, smaller than the RMSE and it is not dependent on the angle. It is also higher than the ULA until an angle of 65°. The spacing limitation is the same as for the ULA:

$$d \leq \frac{\lambda}{2} \tag{6.15}$$

To illustrate that, a theoretical simulation with the following d: 2λ, λ, $\frac{\lambda}{2}$ is done. The parameters for the Monte-Carlo simulation are $N = 450$ and $K_{trials} = 500$. In Figure 6.18 the phenomenon of aliasing, as for the ULA, is shown.
Therefore, the different d distance are: $\frac{\lambda}{2}$, 0.4λ, $\frac{\lambda}{4}$ and $\frac{\lambda}{6}$. Like the pentagon, the cross geometry is isotropic, so only one simulation is made for an angle of 85° with $N = 450$ snapshots and $K_{trials} = 500$ for the Monte-Carlo simulation. As shown in Figure 6.19, the higher the SNR, the smaller the error. However, it can be seen that a secondary peak appears for $\frac{\lambda}{2}$. To explain this, the cross has to be seen as two ULA, each with three antennas. When an angle is close to the endfire of one of these two arrays, the ambiguity that can be seen for an ULA, appears. Thus, around 0°, 90°, 180° and 270° second peaks can be observed for $\frac{\lambda}{2}$. For this reason, $d = \frac{\lambda}{4}$ is chosen, as this value produces one peak and has a good performance regardless of the SNR.
Finally, like it was done for the ULA, the impact of the error of 1% in the geometry is simulated with $N = 450$ and $K_{trials} = 100$ Monte-Carlo simulations. Figure 6.20 shows that for both the ULA and the pentagon array, the impact of the geometry is less than 1° regardless of the SNR level, meaning that it does not cause significant errors in the DoA estimation.

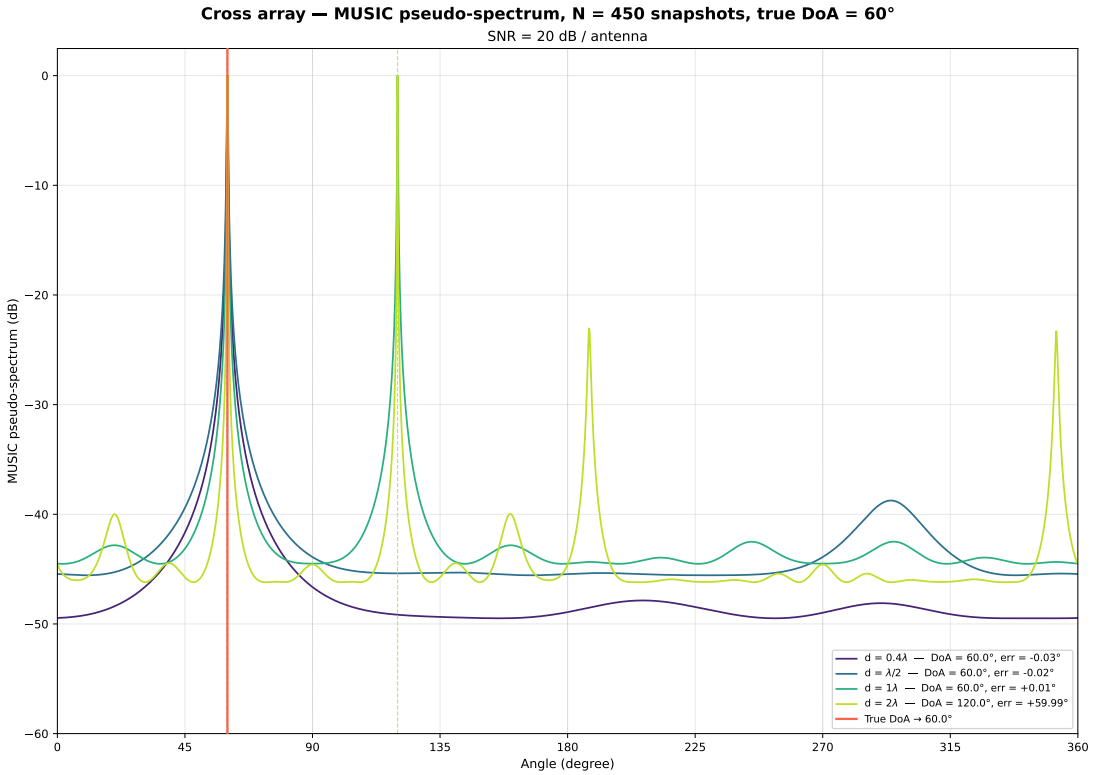


Figure 6.18: Theoretical MUSIC spectrum for an angle of 50°

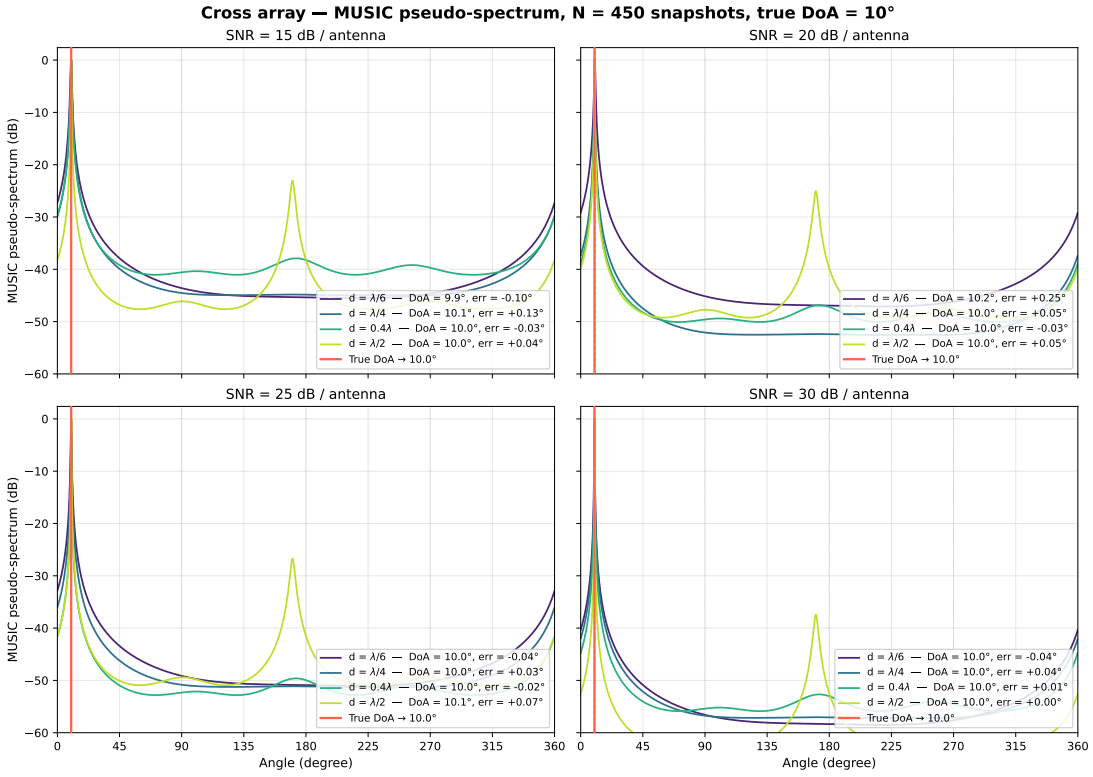


Figure 6.19: Theoretical MUSIC spectrum for an angle of 10° and different SNR

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

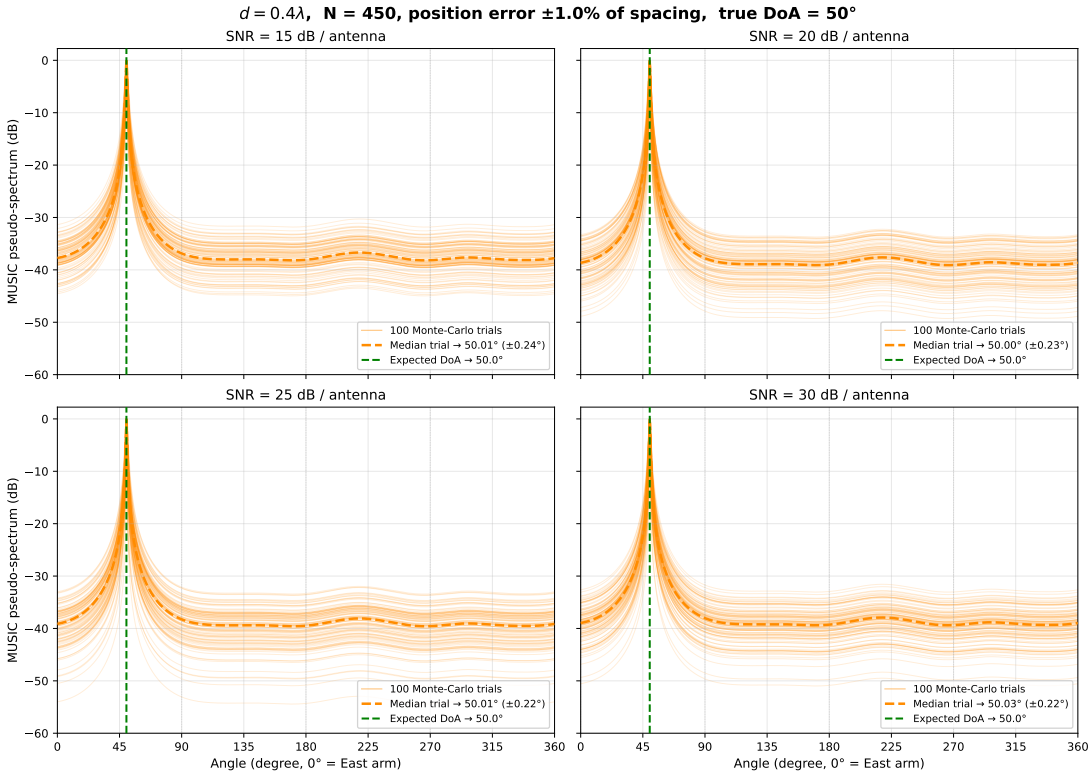


Figure 6.20: MUSIC spectrum for the cross array for an errors of position of 1%

6.3 Comparison of different array geometries

First, a summary of the different arrays and their characteristics:

Array	Chosen spacing
Linear	$d = 0.4\lambda$
Cross	$d = 0.4\lambda$
Pentagon	$R = 0.345\lambda$

Table 6.1: Settings of the different arrays

In order to compare the performance of each array, the CRB can be used. Figure 6.21 and Figure 6.22 show the angular information $h(\theta)/\pi^2$ of the three arrays as a function of the direction of arrival. Since the CRB is inversely proportional to $h(\theta)$, a higher curve indicates a better reachable precision.
In Figure 6.22 and Figure 6.22, it appears that the ULA is more precise than the cross array for angles between 0° and 63.3°. The ULA is also more precise than the pentagon array for angles between 0° and 64.4°. However, after these values, the performance of the ULA decreases and becomes worse than the performance of the two other arrays. Moreover, the cross array has a slightly better overall precision compared to the pentagon one and does not present the ambiguity of the ULA. Thus, in theory, the cross array seems to be the best compromise.

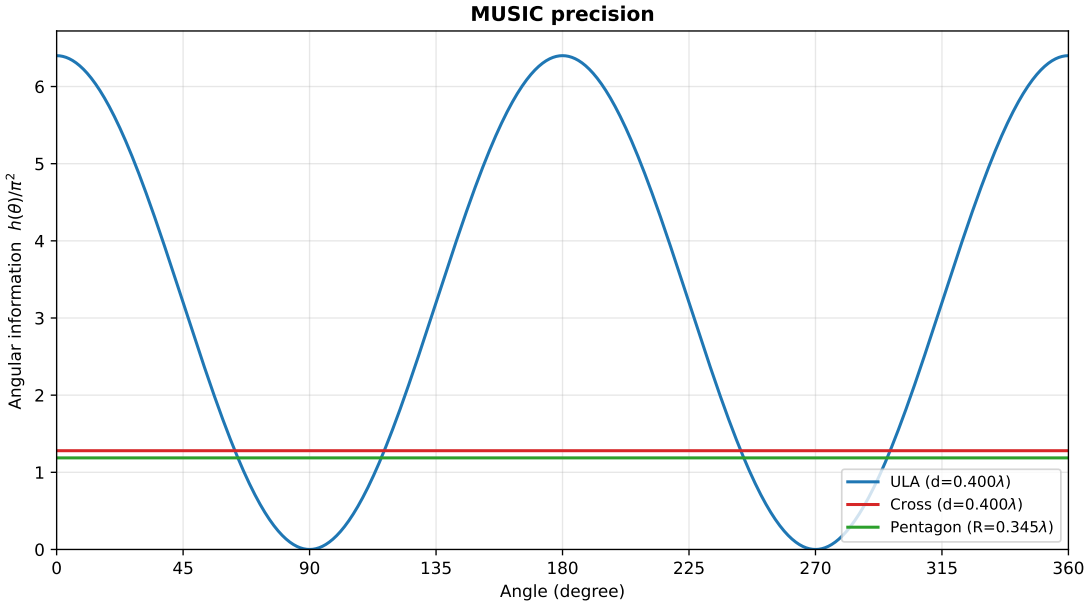


Figure 6.21: Comparison of the precision of MUSIC for the different array

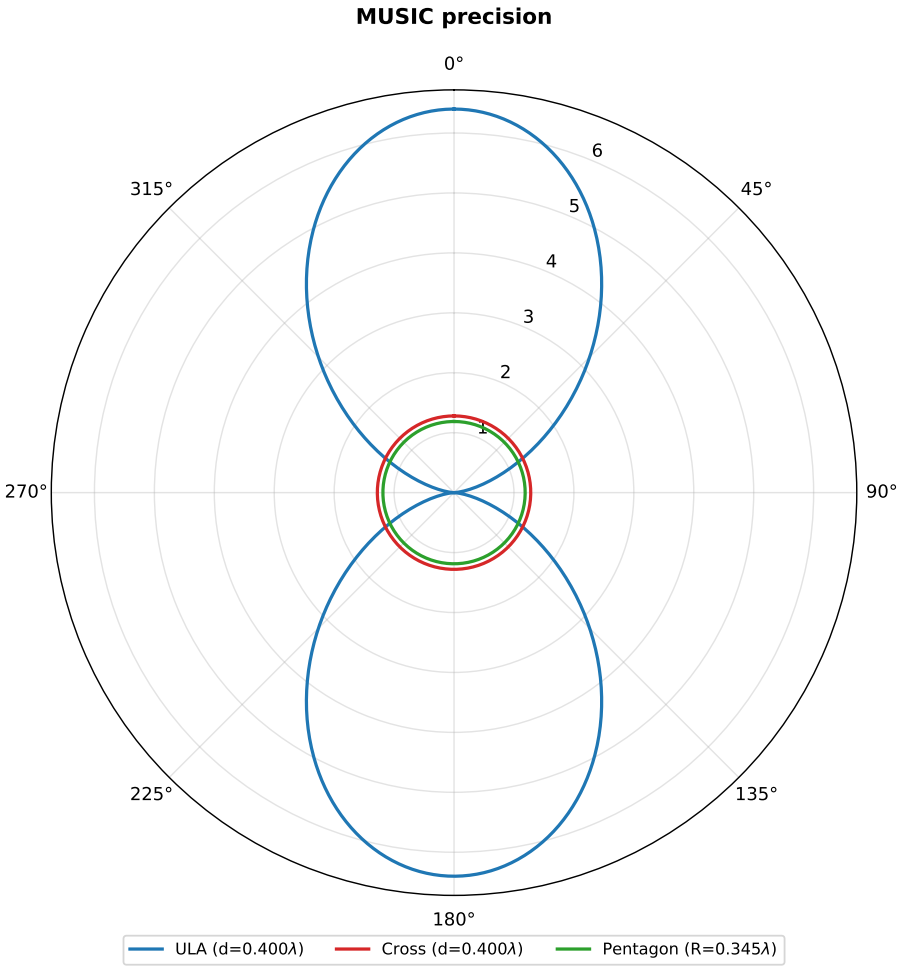


Figure 6.22: Comparison of the precision of MUSIC for the different array

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

6.4 Conclusion

In this chapter, the algorithm for the DoA was explained and the GNU Radio chain, which captures the useful signal, was presented. Moreover, the selected arrays were presented and theoretically analysed. The ULA presents the best results near the broadside, however, it has an ambiguity between the front and back and shows errors that increase near the endfire. The cross and the pentagon arrays are less precise near the broadside compared to the ULA, but they have a constant precision even close to the endfire and show no ambiguity. The three of them will be tested in the next chapter.

Chapter 7

Experimental results

This chapter presents the conducted measurements. The three setups, already described in the previous chapter, were tested near the coast. Indeed, two measurement campaigns took place in Altwarp, which is located in the North East of Germany, as shown in the Figure 7.1. In this area, there are Class A and Class B boats transmitting AIS messages, which were recorded and analyzed.

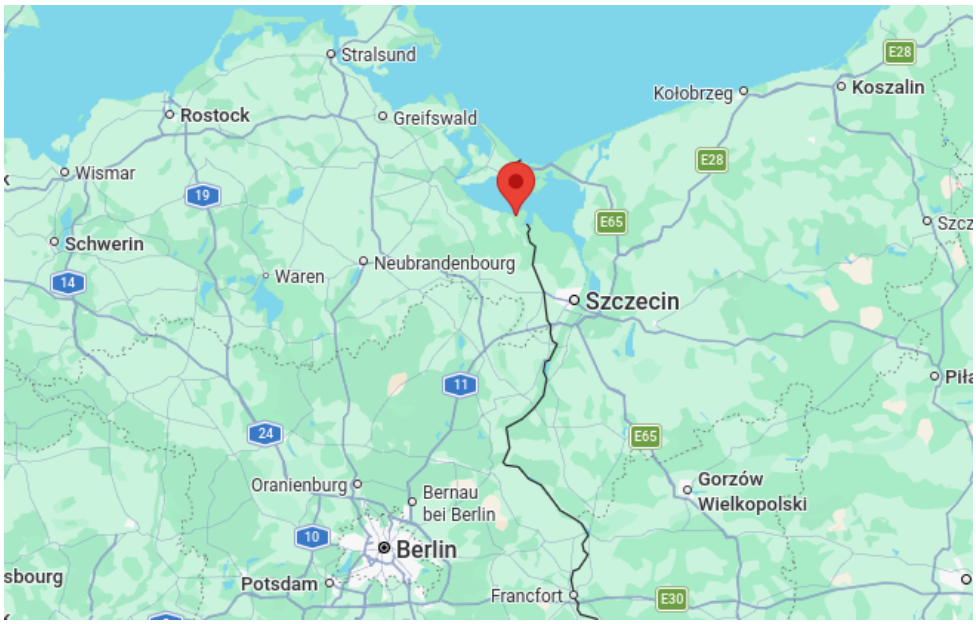


Figure 7.1: Measurement campaign site

7.1 Test on DLR site

Prior to the measurement campaigns, the setup as well as the GNU radio chain were tested on the site of DLR. Therefore, the arrays were installed outside and some AIS emitters were used to generate the useful signals.The final GNU Radio chain can be seen in Figure 7.2.

For readability reasons, only half of the chain is depicted, and the AIS decoding chain and the burst detector are put together. The chain, shown in Figure 7.2, is detecting AIS messages and bursts for channel A. In the file, the same chain is also present for channel B, so messages

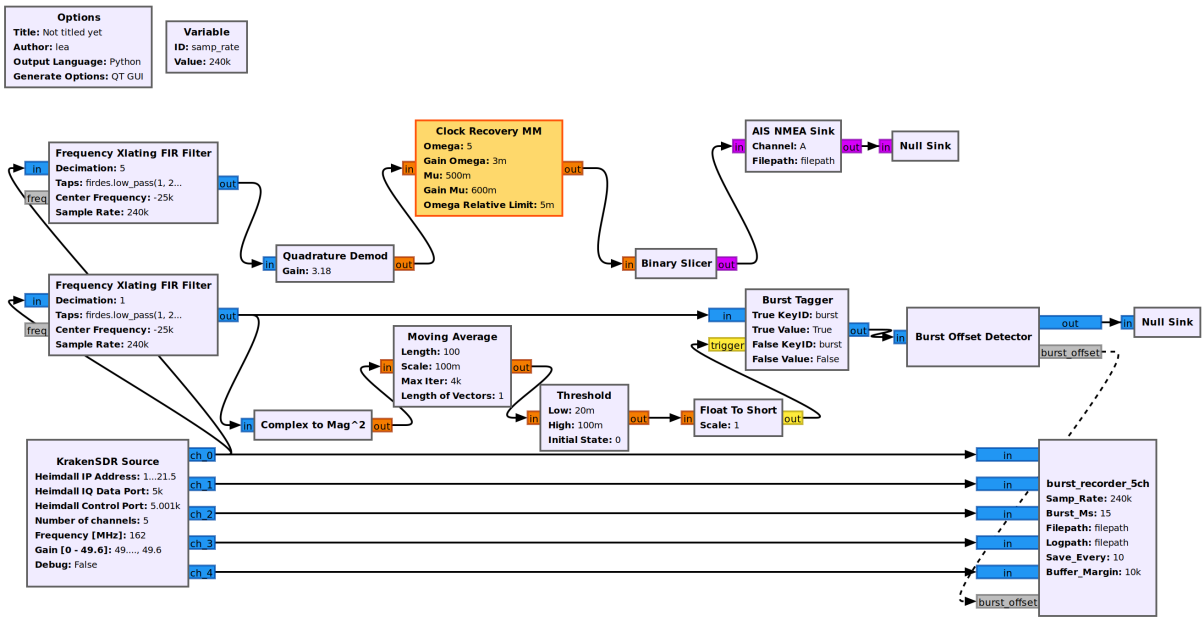


Figure 7.2: Final GNU Radio chain

from both channels are received. This final chain first records the burst and its associated AIS messages, and then performs the DoA estimation post processing with Python. In order to be sure that the messages are detected and recorded with the different arrays, the chain was tested with an emitter on the site of DLR. The used setups were the same as the ones used for the first measurement campaign and are shown in Figure 6.2, Figure 6.9 and Figure 6.15. For the ULA, the antennas were placed on a metal bar. For the cross and the pentagon arrays, the position of the antennas were first marked with large screws on a wooden plank. Then, the antennas were placed at these positions.

With $d = 0.4\lambda$ for the cross array and at $R = 0.345\lambda$ for the ULA, multiple tests were done, while an emitter was placed about one hundred meters away from the arrays. Regarding the ULA, one test was done near the broadside and the other was done closer to the endfire. Figure 7.3 and Figure 7.4 depict the results:

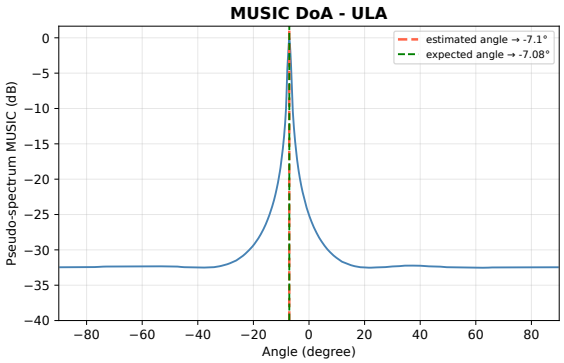


Figure 7.3: Result of the DoA estimation on DLR site for an ULA at 7.08°

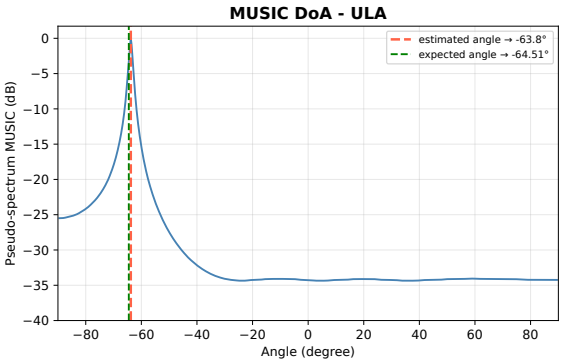


Figure 7.4: Result of the DoA estimation on DLR site for an ULA at 64.51°

While, near the broadside, the angle estimation is very close to the actual angle, the accuracy decreases with the angle coming close to the endfire. Indeed, in Figure 7.4 the error is around

0.8°, compared to the error of 0.02° in Figure 7.3. This is coherent with previous simulations made in Chapter five. Then, as shown in Figure 7.5 and Figure 7.6, the cross and the pentagon arrays were tested. The pentagon array shows an error of 0.8°, while the cross array shows one of 1.3°. As expected, they are larger than the error observed for the ULA near the broadside. However, they are also larger than those obtained in the simulations. This could be explained by possible errors made during the estimation of the true angle or small errors in the geometries. After these tests, the first measurement campaign took place in Altwarp.

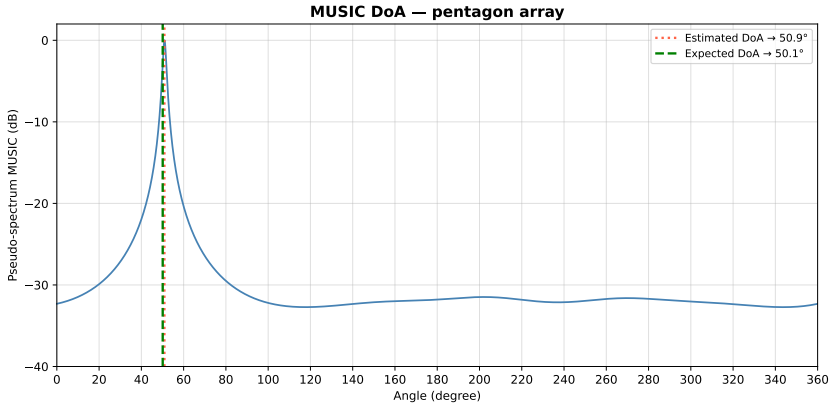


Figure 7.5: Result of the DoA estimation on DLR site for the pentagon array at 50.1°

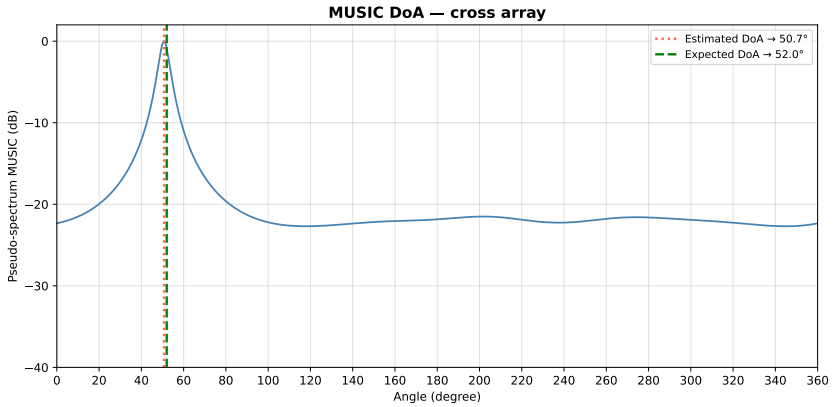


Figure 7.6: Result of the DoA estimation on DLR site for the cross array at 52.0°

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

7.2 First measurement campaign

At this first measurement campaign in Altwarp, the three arrays were tested with the same spacing that was used for the test in the previous section, as shown in Figure 7.8, Figure 7.9 and Figure 7.10. While running the experiments, the van provided the necessary power supply for the computer, the Raspberry Pi 4 and the KrakenSDR, as shown in Figure 7.7.



Figure 7.7: Setup of the first measurement campaign in Altwarp



Figure 7.8: ULA used during the first measurement campaign



Figure 7.9: Pentagon array used during the first measurement campaign



Figure 7.10: Cross array used during the first measurement campaign

All three arrays have the same reference. The broadside is the axis between the center of the array and some bridge ruins. This reference illustrated in Figure 7.11 will be used for the rest of this chapter.
In the following, the results of the first measurement campaign are presented. Three recording sessions were made, one for each array. Each recording session lasted ten minutes. During these sessions six boats or stations were detected, as shown in Table 7.1. While among them, there are three motionless objects: the *Station 261*, the boat *211753900* and the boat *211315970*, three of them are in motion: *OLHIS*, *JANA* and *VECTOR*. The positions of these objects are shown in the map 7.12. Points of the same color are representing the same boat and its motion. Additionally, the average SNR is written next to the MMSI or the name of the boat.

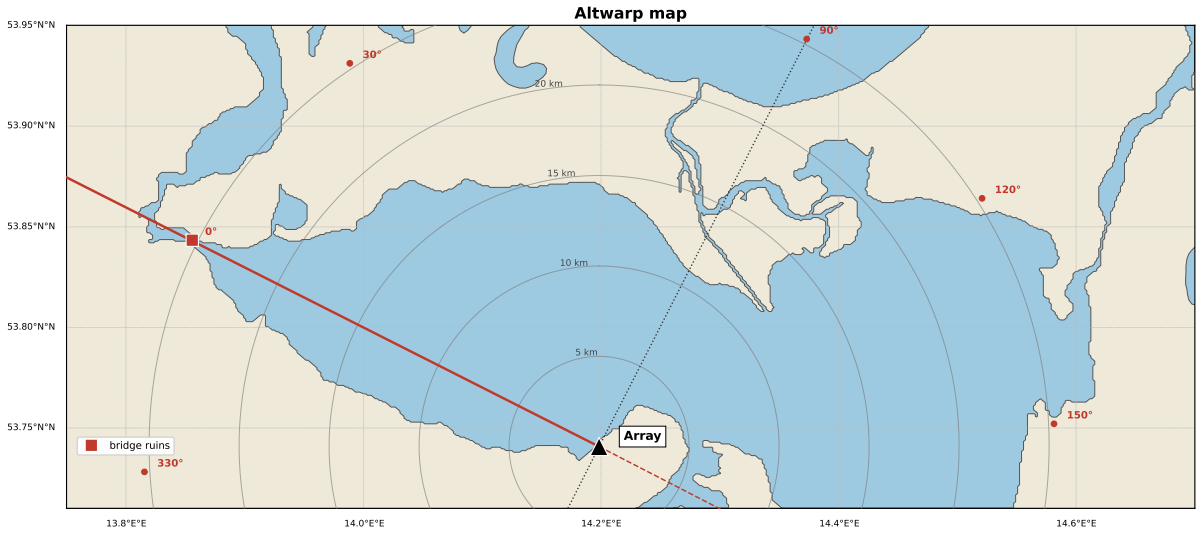


Figure 7.11: Altwarp map with the reference used by the three arrays

				Bearing		
Name	MMSI	Class	Dist. (km)	Linear session	Cross session	Pentagon session
VECTOR	211275110	Class B	6.3 – 9.1	–	17.0° – 24.0°	28.6° – 36.4°
OHLIS	211521550	Class B	6.4 – 7.3	–	56.9° – 64.3°	71.0° – 82.2°
Station 261	2610800	Base station	20.7	–	motionless (79.2°)	motionless (79.2°)
JANA	211499050	Class B	5.7 – 7.1	–	80.5° – 90.0°	94.7° – 105.2°
–	211315970	Class A	9.7	–	motionless (330.4°)	motionless (330.4°)
–	211753900	Class A	9.7	–	motionless (330.5°)	motionless (330.5°)

Table 7.1: AIS emitters detected during the first measurement campaign

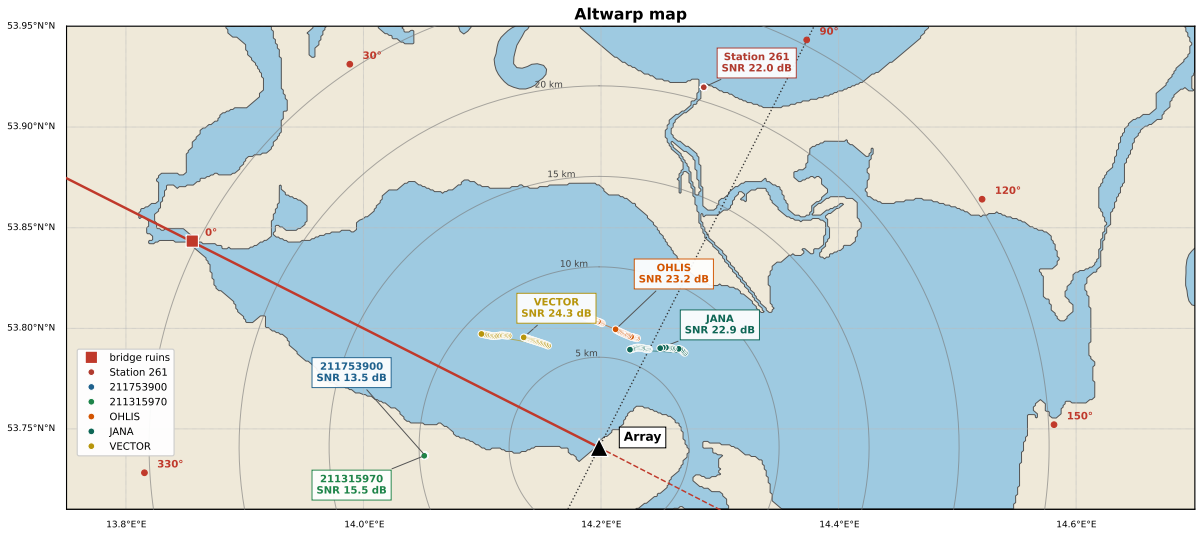


Figure 7.12: Altwarp with the reference and the different object identified during the first measurement campaign

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

7.2.1 Pentagon array

The first recording was performed with the pentagon array. It lasted ten minutes and 283 bursts were detected. The DoA is shown in three figures. The small lines of different colors represent the true bearing of each boat that should be obtained, according to the position they are transmitting. The dots represent the DoA, estimated by MUSIC, over time and can be observed in Figure 7.13, Figure 7.14 and Figure 7.15. The numbers in the legend correspond to the MMSI of the boat, which is its ID. In order to evaluate the errors, the Root Mean Square (RMS), bias, absolute median error and absolute median error are computed for each boat.

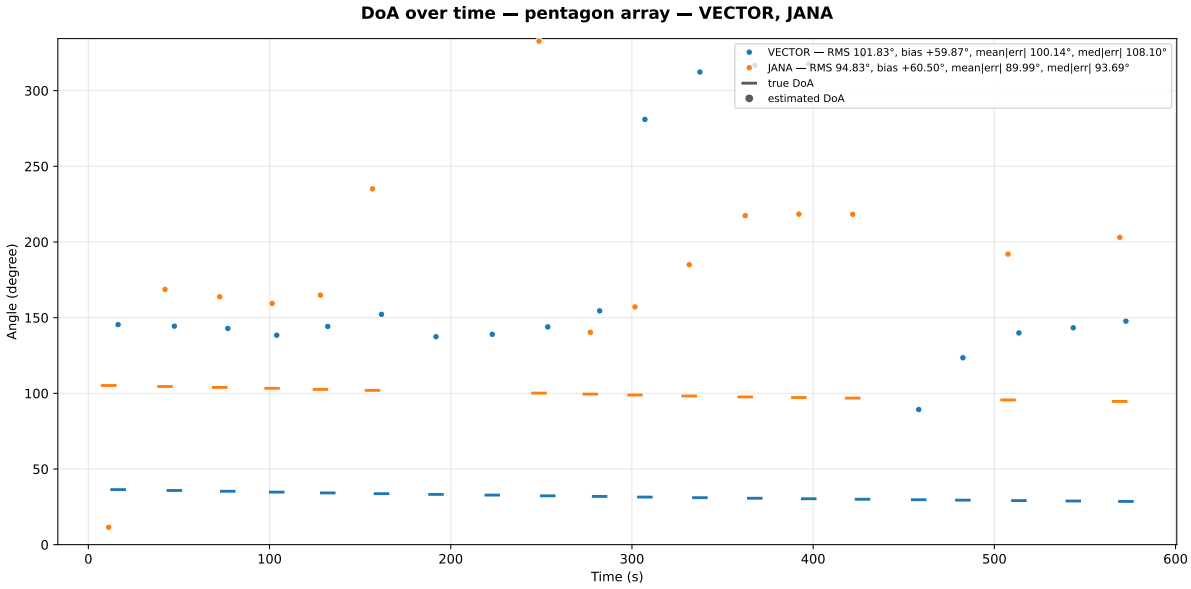


Figure 7.13: DoA over time for the pentagon array - VECTOR, JANA

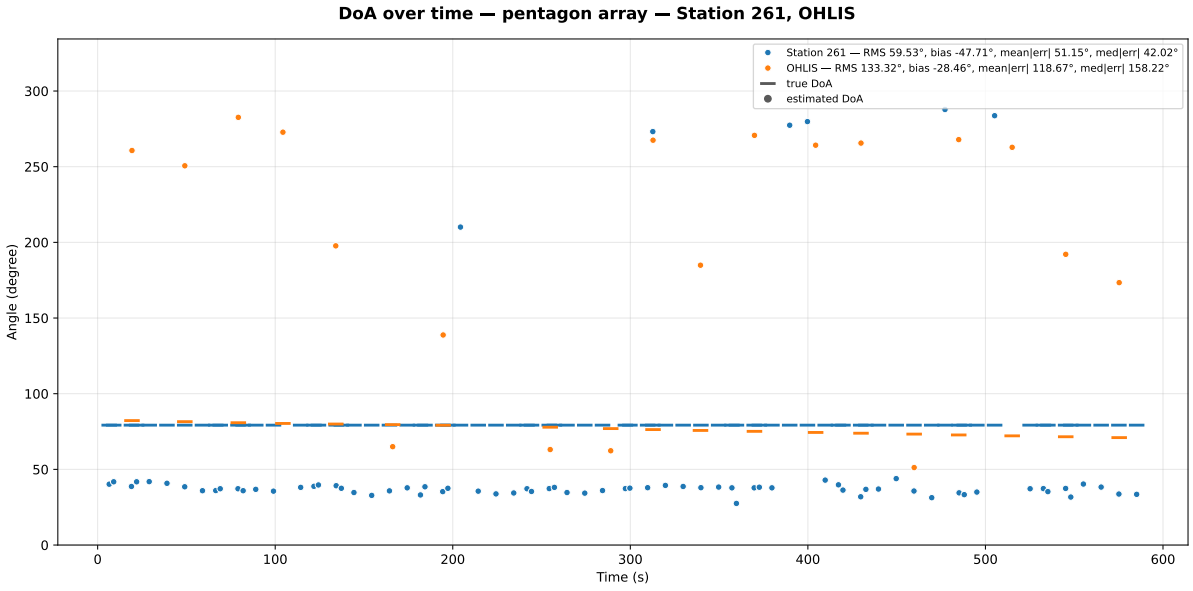


Figure 7.14: DoA over time for the pentagon array - Station 261, OHLIS

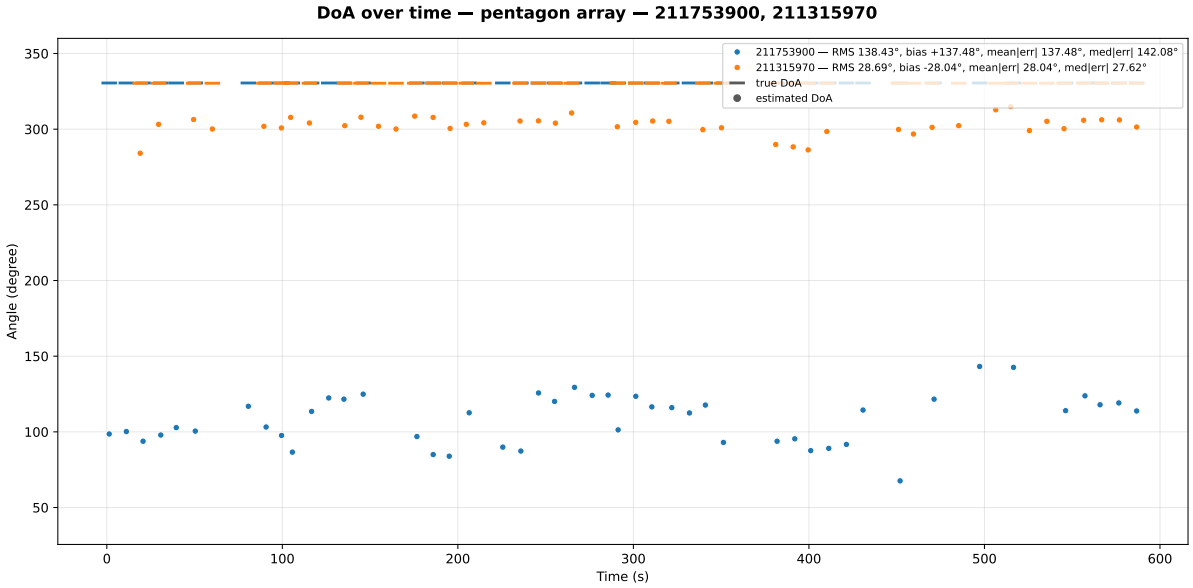


Figure 7.15: DoA over time for the pentagon array - 211753900, 211315970

The result were not the one expected. Indeed, in Figure 7.13 the boats *VECTOR* and *JANA* could not be tracked. The errors have a large distribution with a RMS over 90°.
The same phenomenon is observed in Figure 7.14 for the boat *OHLIS*. Even though, for *Station261* the error is not negligible, the bursts seem to have the same systematic error. Indeed, the bias is close to the absolute mean error. The same can be seen in Figure 7.15 with the boat *211315970*, while the errors for the boat *211753900* have a larger distribution. This difference in the distribution of the errors depends on the motion of the objects. Indeed, the signals from motionless objects are received from the same direction, and therefore the multipath or errors due to setup errors are very similar. When the object is moving it is changing, and therefore the distribution of the errors is higher.

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

7.2.2 cross array

The second recording session was made with the cross array. As for the pentagon array, it lasted for ten minutes and 234 bursts were detected. The DoA estimations for the different objects can be seen in Figure 7.16, Figure 7.17 and Figure 7.18. Regarding motionless and moving objects, the same observations can be made. However, the result for *Station261* is better. Indeed, the bias is close to 0°, which indicates a dispersion around the object, and a RMS, absolute mean error and absolute median error, very close to each other, around 3°. Ultimately, this object is the only one, where the DoA estimation worked.

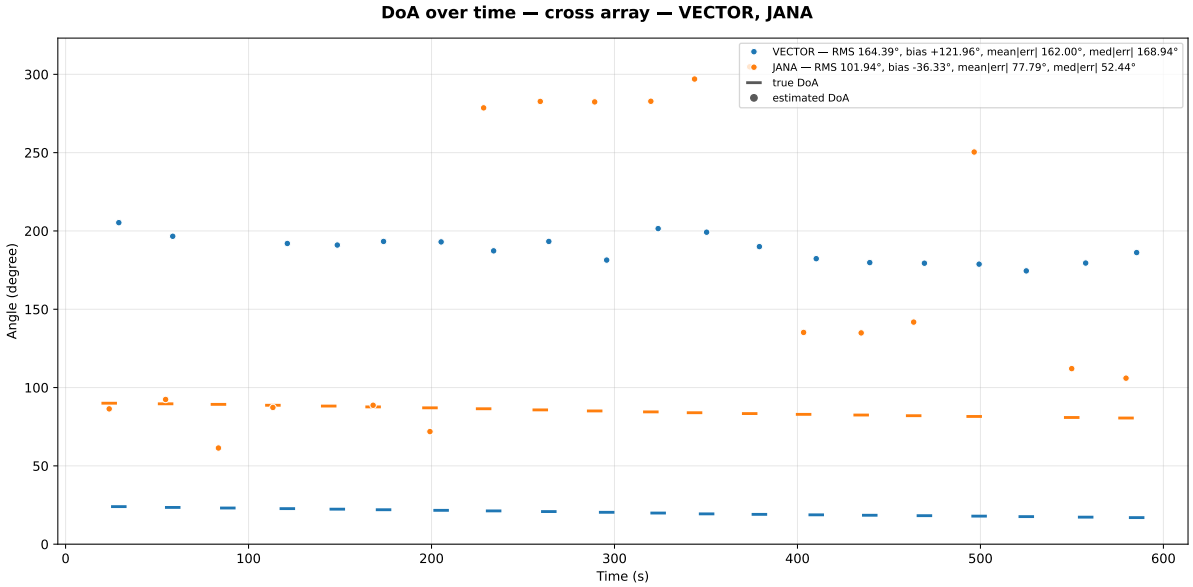


Figure 7.16: DoA over time for the cross array - 211223190, VECTOR, JANA

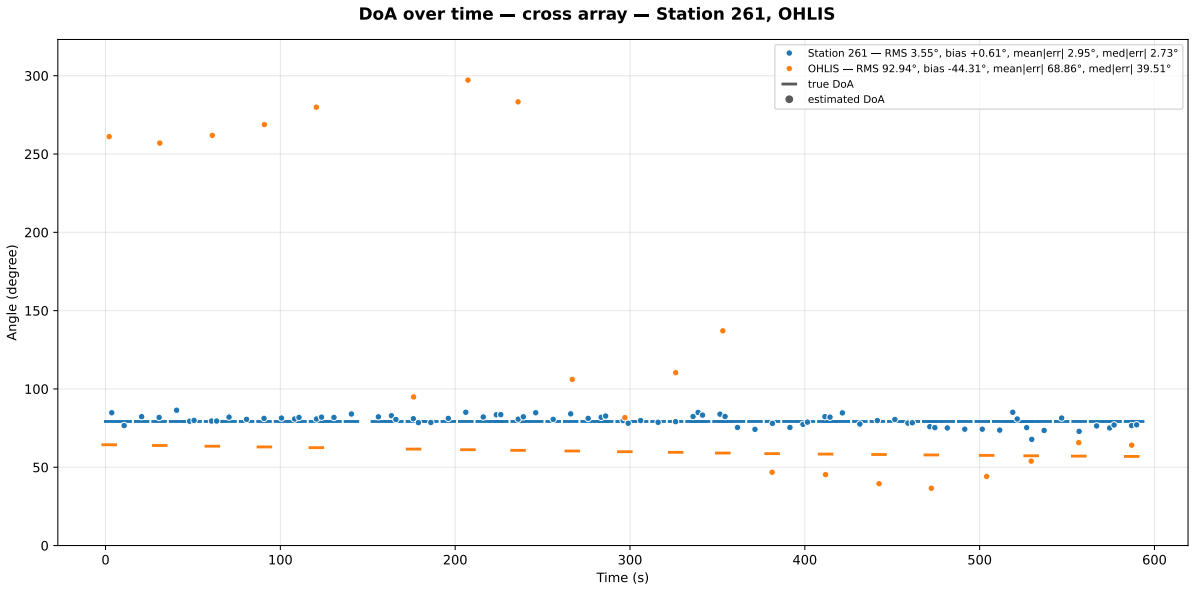


Figure 7.17: DoA over time for the cross array - Station 261, OHLIS

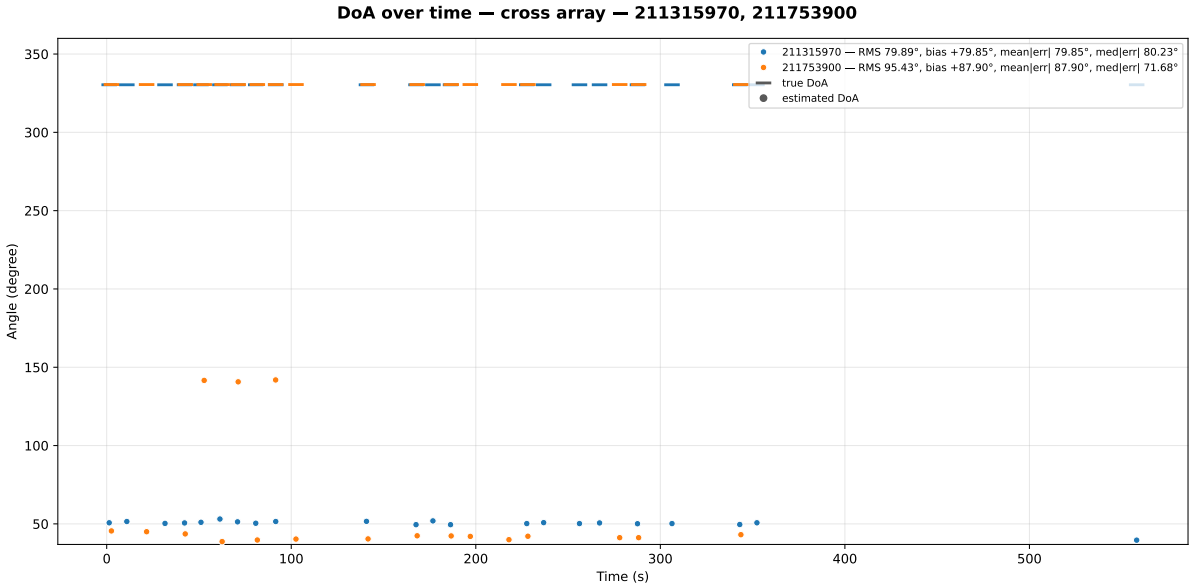


Figure 7.18: DoA over time for the cross array - 211753900, 211315970

7.2.3 ULA

The third session was recorded with the ULA. However, as shown in the Table 7.1, there were no emitters to be found. The hypothesis why nothing was detected during the recording lies in the antennas. Since the default KrakenSDR antennas are quarter waves antennas, their height needs to be 4 and they need a conductive ground plane extending at least /4 around their base to behave as expected. These conditions were not met during the experiments. Having only the screw or the metal bar as the dielectric surface was not enough. Since without a proper grounding of the antennas, the phase and gain deviate from the ideal model, which can affect the DoA, it did not work. These effects are not the same considering the angle of arrival of the signal. After analysis, as shown in Figure 7.19 and Figure 7.20, the SNR is not the same on each channel and the noise level is different depending on the antenna. This is a problem for MUSIC, since it requires the same noise level for every antenna.

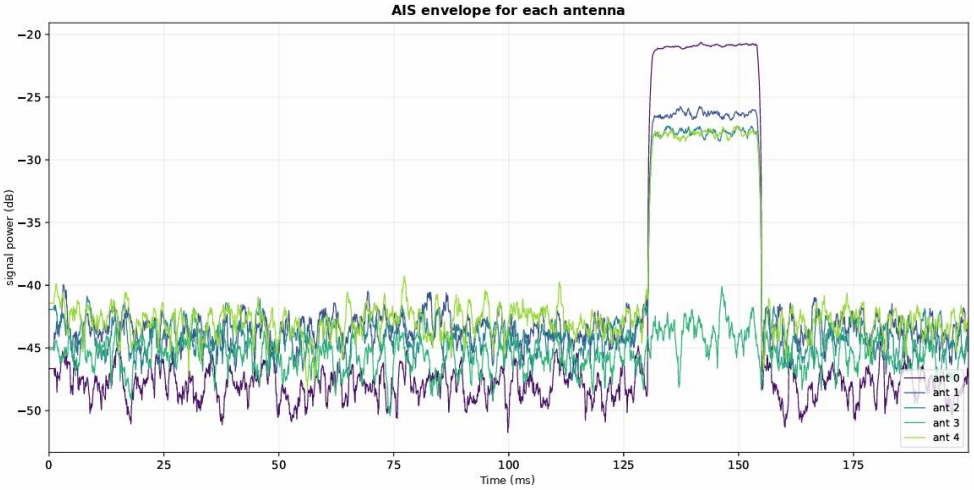


Figure 7.19: Signal power of each antenna of the pentagon array for boat *211753900*

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

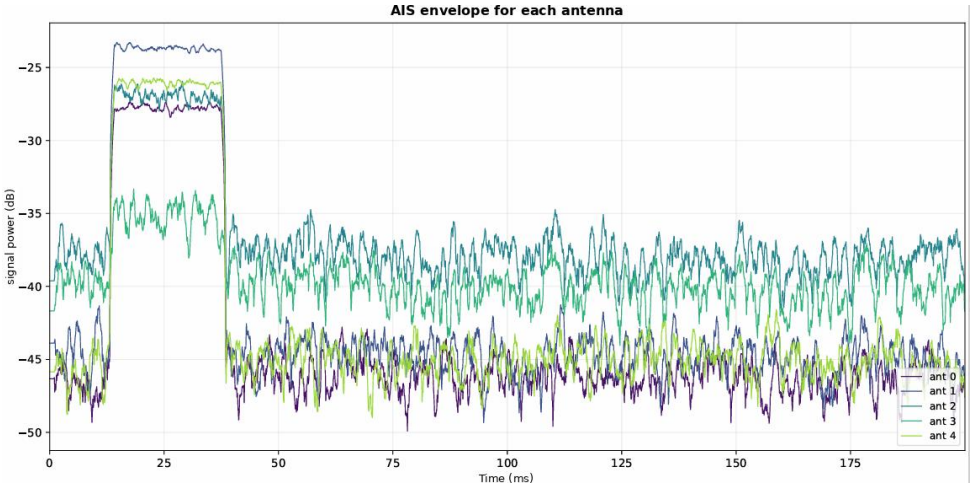


Figure 7.20: Signal power of each antenna of the cross array for boat *211753900*

The Figure 7.19 and Figure 7.20, both show an example of the signal power in one burst. It shows the existence of different levels of noise and the different amplitude levels that can be seen. The differences between the channels are too high, especially when the signal on one of the channels is barely detectable, such as channel 3 in this case. The DoA still works since not all channels are broken. However, the error in the DoA estimation comes from these differences between channels that could be caused by the absence of a common ground plane.

7.3 Second measurement campaign

For the second measurement campaign, the grounding of the antennas was fixed, by using a metal sheet. However, the spacing between each antenna had to be changed due to spatial constraints that were introduced by the new metal sheet. The new spacings are: $d = 35\text{cm} \approx 0.2\lambda$ for the cross and the ULA and $R = 36.5\text{cm} \approx 0.19\lambda$. The measurements were made in Altwarp, exactly where the first measurement campaign was done, and with the same reference, as shown in Figure 7.11. The new arrays can be seen in Figure 7.21, Figure 7.22 and Figure 7.23:



Figure 7.21: ULA for the second measurement campaign



Figure 7.22: Pentagon array for the second measurement campaign



Figure 7.23: Cross array for the second measurement campaign

For the second measurement campaign, one recording session of ten minute was made per array. As shown in Table 7.2, six boats or stations were detected, during these sessions.

			Bearing		
Name	MMSI	Dist. (km)	Linear session	Cross session	Pentagon session
LA PERLA	211637160	6.4 – 8.8	−14.79° – −6.49°	not detected	338.32° – 338.35°
–	269057081	7.3 – 9.7	−3.45° – 15.03°	331.04° – 331.07°	335.90° – 341.26°
–	211324960	7.3 – 12.7	54.80° – 63.01°	340.06° – 350.42°	19.05° – 38.44°
–	257306000	17.5 – 20.7	77.71° – 78.65°	not detected	one report (78.55°)
Station 261	2610800	20.7	motionless (79.22°)	motionless (79.22°)	motionless (79.22°)
–	261182155	14.6	motionless (86.30°)	motionless (86.30°)	motionless (86.30°)

Table 7.2: AIS emitters received during the second measurement campaign at Altwarp

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

Among them, there are two motionless objects: the *Station 261* and the boat *261182155*, and four objects in motion: *LA PERLA*, *269057081*, *211324960*, *257306000*. The positions of these can be seen in the map 7.24. Various points of the same color represent the same boat and its motion. Additionally, the average SNR is written next to the MMSI or the name of the boat.

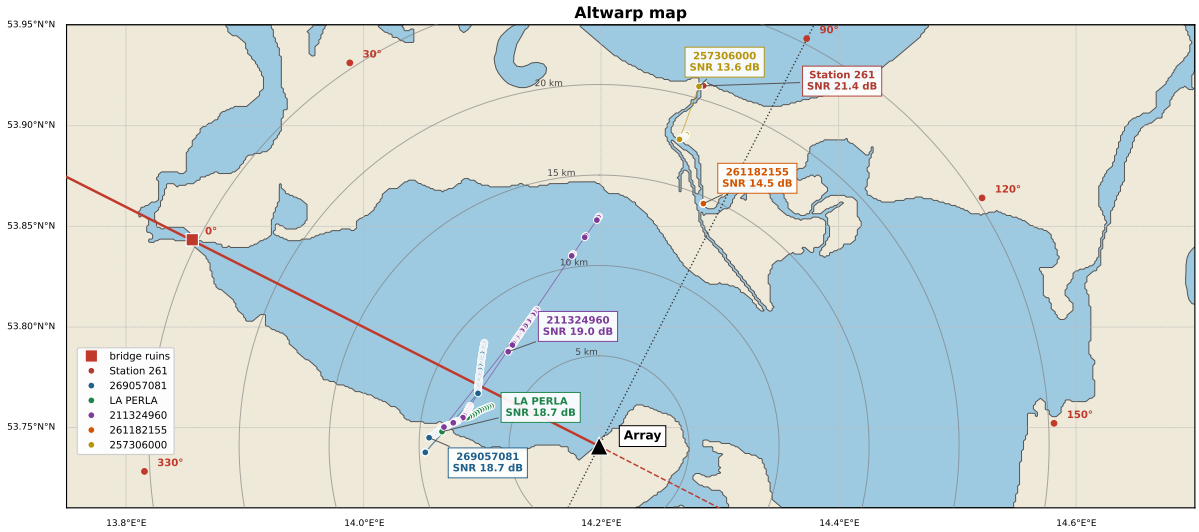


Figure 7.24: Altwarp map with the reference and the different objects identified during the second measurement campaign

7.3.1 ULA

The first session was made with the ULA and recorded bursts during ten minutes. As the previous section the result are presented in the form of DoA over time in Figure 7.25 and Figure 7.26:

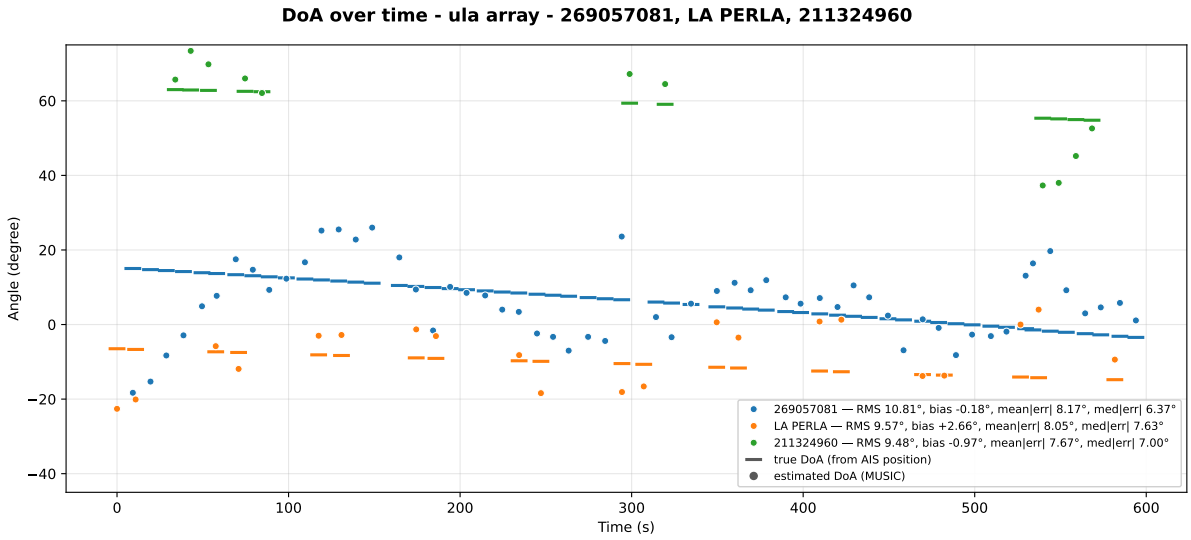


Figure 7.25: DoA over time for the ULA - 269057081, LA PERLA, 211324960

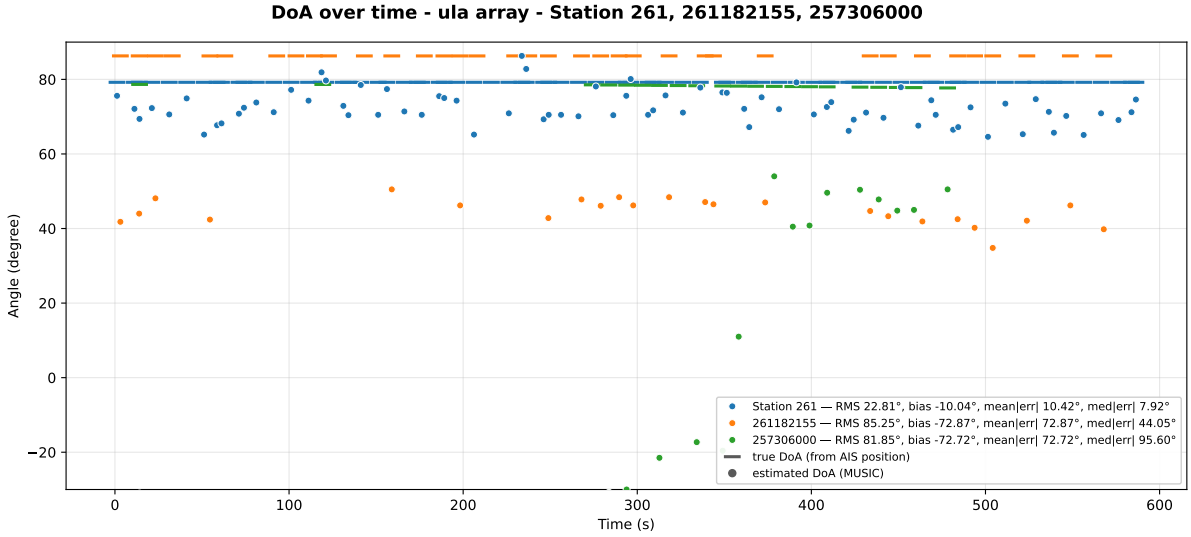


Figure 7.26: DoA over time for the ULA - Station 261, 261182155, 257306000

First, one can see that the tracking works better in comparison to the first measurement campaign. As shown in Figure 7.25, the absolute median errors of *269057081*, *211324960* and *LA PERLA* are below 8°. The RMS and the absolute mean error are close, suggesting that there are only few outliers. Additionally, *269057081* and *211324960* both have a bias close to 0°, which indicates a distribution of the errors around the true position. Therefore, the errors appear to be more random than systematic. *LA PERLA* also presents a small bias of 2.66°. Moreover, as shown in Figure 7.26, the tracking for *Station 261* also works. However, the errors are slightly larger than the emitters near the broadside, which is coherent with the theory. Indeed, the absolute mean error and the median error are approximately 1° and 2° higher. While the higher RMS is indicating a larger distribution of the errors. For *261182155* and *257306000* the tracking is not working. This is shown by the large values of RMS, bias, absolute mean and median error. Finally, for this array, the tracking works better than in the first measurement campaign near the broadside, even if boats are in motion. However, near the endfire, due to larger errors, the tracking is less efficient. Additionally, the errors are too high to consider the estimates as precise enough.

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

7.3.2 Pentagon array

Following this, the pentagon array was tested and recorded bursts for ten minutes. The boat *257306000* was not detected. As in the previous section, the results are presented in Figure 7.27, Figure 7.28 and Figure 7.29 in the form of DoA over time.

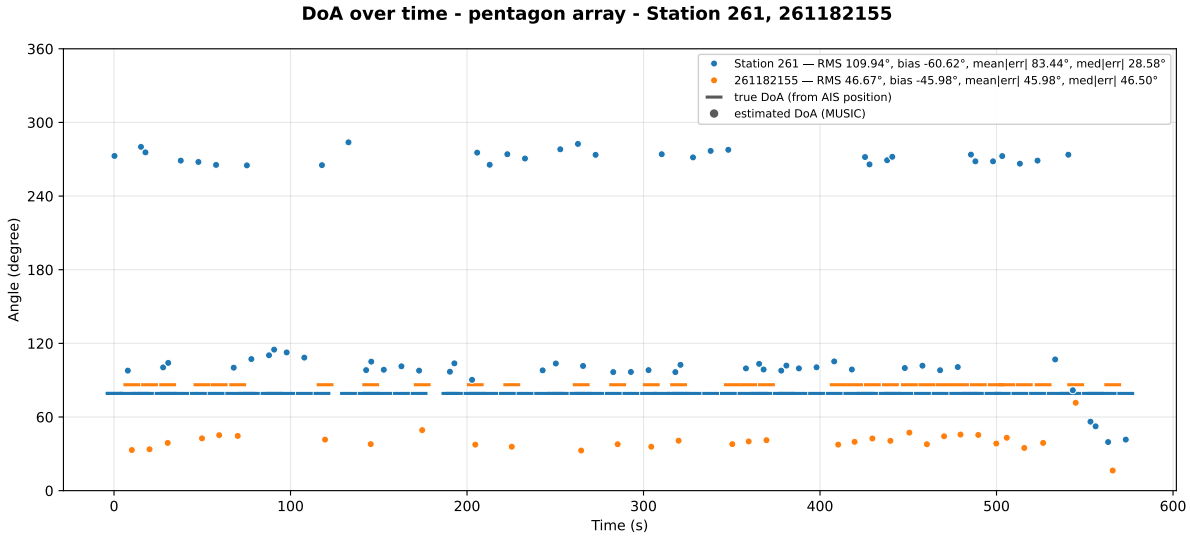


Figure 7.27: DoA over time for the pentagon array - Station 261, 261182155

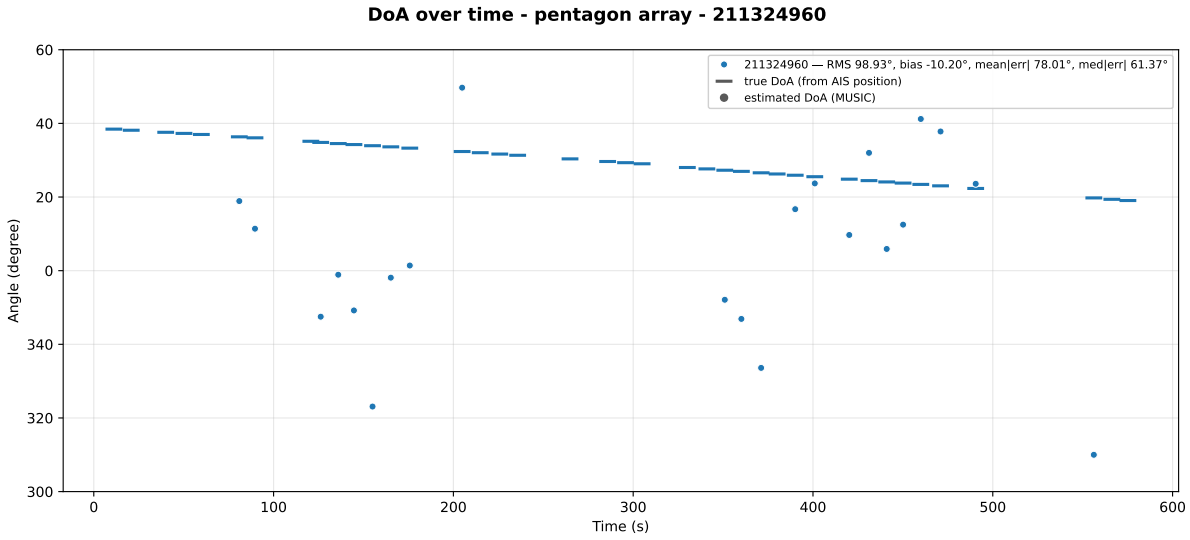


Figure 7.28: DoA over time for the pentagon array - 211324960

For the pentagon, the tracking is less efficient than with the ULA. Indeed, in Figure 7.27,*261182155* shows a systematic error around 46°, which is too high for reliable vessel tracking. For *Station 261*, two groups of bursts show a different systematic error, leading to a RMS equal to almost 110° and an absolute mean error of almost 84°. The tracking is not working. The same goes for *211324960* in Figure 7.28, since the errors are too large. Finally, in Figure 7.29, one can see better results. While, for *269057081* the RMS is around 22°, the bias around 1° indicates a distribution of the errors around the true bearing. Additionally, the absolute median error equal to 12.50° shows a better tracking than other vessels. Since only three

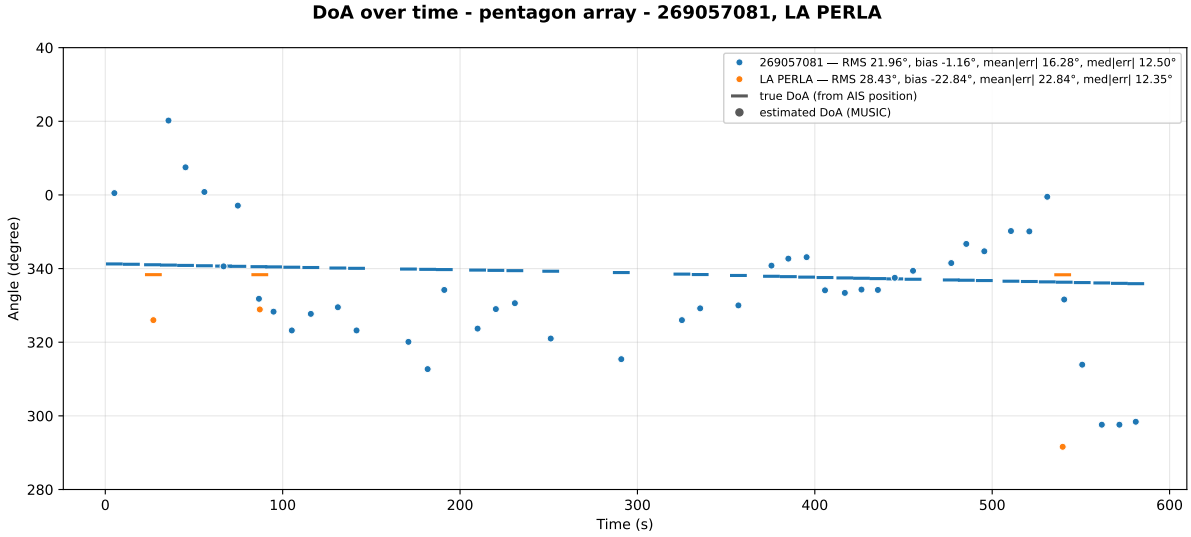


Figure 7.29: DoA over time for the pentagon array - 269057081, LA PERLA

bursts were recorded for *LA PERLA*, it is difficult to determine if the tracking works, with such little data. However, out of the three bursts, two present an error around 10°, which is better than most of DoA estimations made with the pentagon.
In conclusion, the majority of the objects cannot be tracked with the pentagon array, due to big systematic errors and outliers. The exceptions are the boats *LA PERLA* and *269057081* that present an absolute median error around 12°. However, even for these boats, it is not possible to validate a precise tracking.

7.3.3 Cross array

Ultimately, the cross array was tested and recorded bursts for eight minutes. Due to weather conditions, two interruptions were made during the recording. The boats *257306000* and *LA PERLA* were not detected. As in the previous sections, the results are presented in Figure 7.30 and Figure 7.31 in the form of DoA over time:

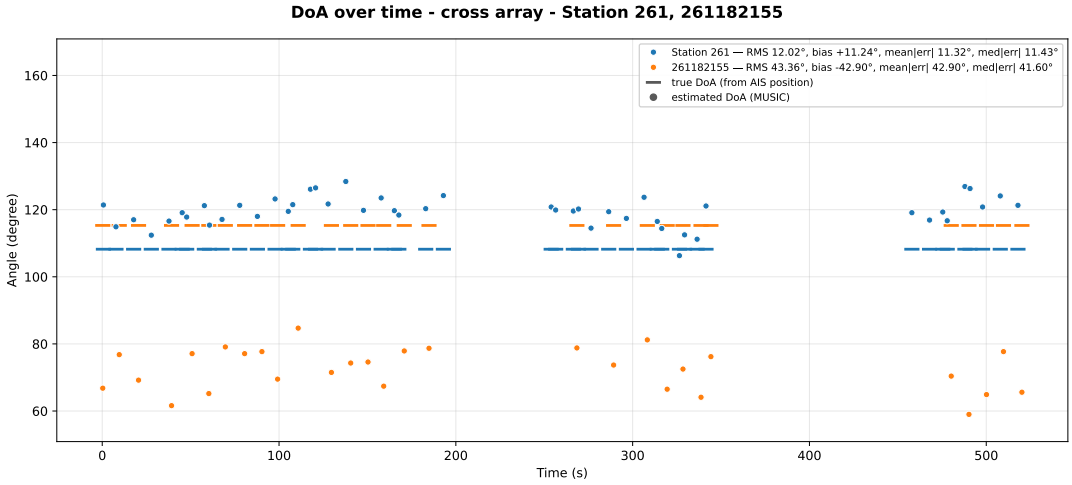


Figure 7.30: DoA over time for the cross array - Station 261, 261182155

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

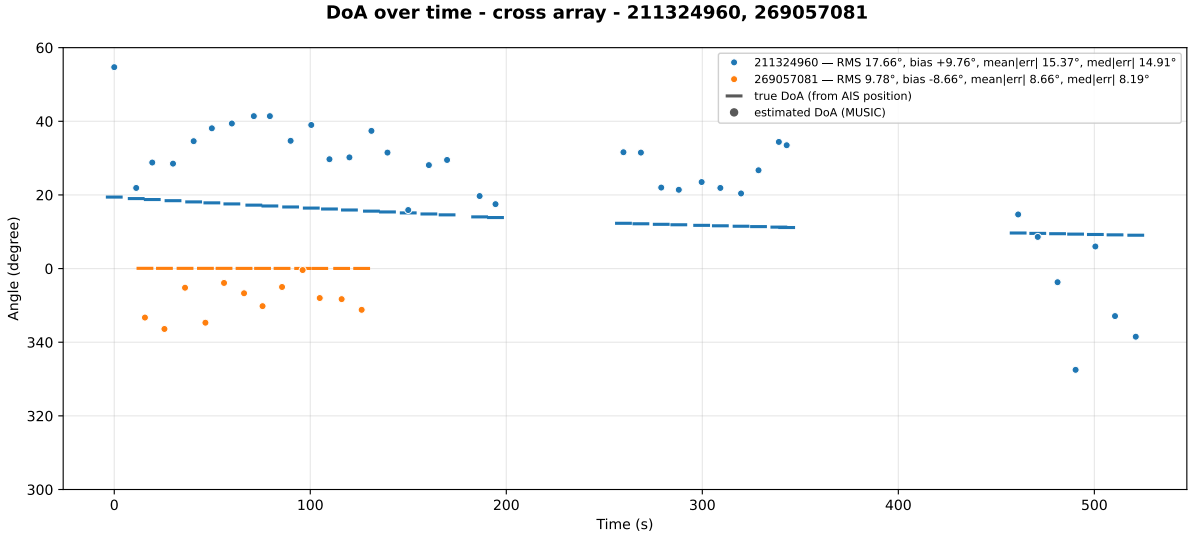


Figure 7.31: DoA over time for the cross array - 211324960, 269057081

For the cross array, the results show that some objects can be tracked. Indeed, in Figure 7.30, *Station 261* shows a systematic error with few outlier around 11°. Therefore, it is better than the performance of the pentagon array and close to the ULA. However, for the vessel *261182155* the errors are larger, and with an absolute mean and median error around 41°, the tracking is not working here. In Figure 7.31, the RMS of the ship *211324960* is around 17°, due to some outliers. The absolute mean error and median error are approximately 15°, while the bias is around 10°. Therefore, the DoA estimation seems to track the vessel with a systematic error. Ultimately, the boat *269057081* shows the same results as *Station 261*, with an approximately 3° better precision.
To conclude, the cross array shows regular systematic errors for all objects. While three out of the four objects detected can be considered as tracked, the precision is not enough to validate its functioning.

In order to compare the three arrays, the RMS, the mean absolute error and the median absolute error for *Station 261*, *269057081*, *211324960* and *LA PERLA* are compared in Figure 7.32.

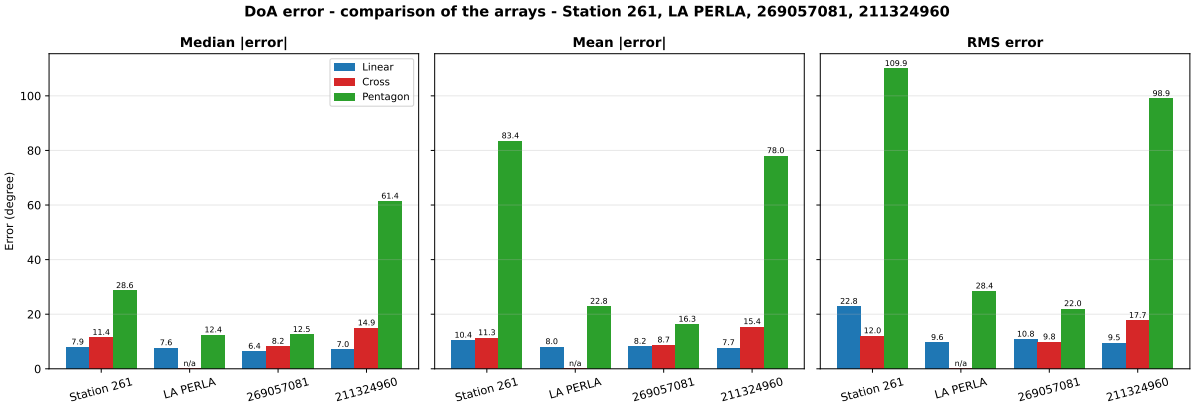


Figure 7.32: Comparison of the different arrays

This graph indicates that the ULA is the best overall performing array. After that comes the cross array and then the pentagon array with the worst performance. The better performance of

the ULA can be explained by the fact that its aperture is two times larger than the one of the pentagon or the cross array, and therefore a same small phase error between the antennas will be two times smaller. Furthermore, it can be seen that the performance of the ULA decreases near the endfire. Finally, all those errors are around or above 6°, which is too large to precisely estimate the bearing of a boat. Therefore, the functioning of the setup cannot be validated.

7.4 Limitation of the setup

Even though some objects were tracked, the errors remain significant. After verifying the whole software and calibration, the antennas were finally tested. The data sheet indicates that such antennas can work from 100MHz to 1GHz [4]. To verify that a Vector Network Analyser (VNA) was used, in order to measure the S parameter. The S parameter is describing the capacity of an antenna to reflect and transmit signal at each of its ports. In this case $S11$ was evaluated. This parameter measures the fraction of the signal that is reflected by the antenna. The lower the value, the better the antenna is adapted to the frequency. For the theoretically correct height of the antenna placed on a metallic surface, the following result was obtained in Figure 7.33:

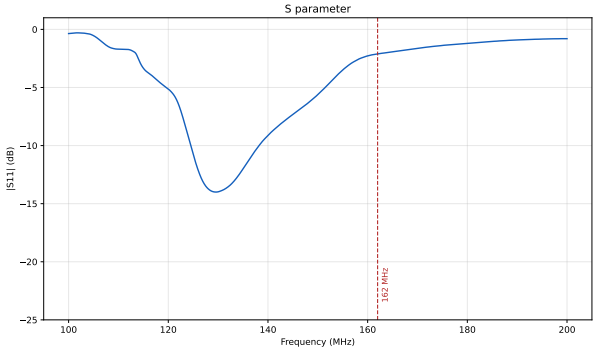


Figure 7.33: S parameter at 162MHz for an antenna height of 44cm

As shown in Figure 7.33, the antenna is far from being optimal for the given frequency. In fact, the minimum value should be at 162 MHz instead of around 129 MHz. In order to achieve the minimum value at 162 MHz, the antenna should be 18.2cm long, as shown in Figure 7.34:

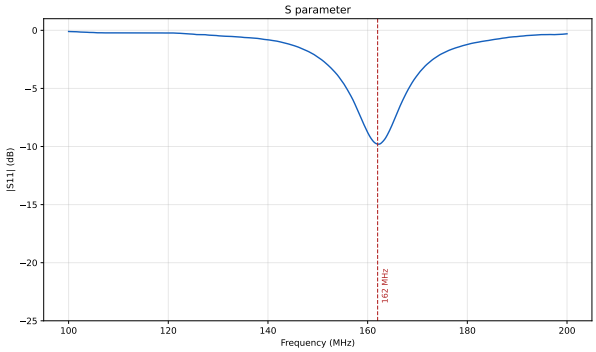


Figure 7.34: S parameter for an antenna height of 18.2 cm

It appears that the anomalies in the S parameters are caused by parasite effect of the magnetic mount of the antennas. Indeed, when the magnetic mount is on a metallic surface, the parasite effects increase and change the S parameter. However even if the antenna is not not placed

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

on any metallic surface, the S parameter would still not be the one expected at 162 MHz for the theoretical height of the antenna. This could explain the errors in the DoA estimation. At 162MHz, the antennas show $|S11| = -2.1$ dB, while this value is supposed to be around the minimum of the curve. Since all five antennas have the same value for $|S11|$, this problem is common within all channels. It causes a weaker SNR and reflections of the signals that may increase mutual coupling between the antennas. This could contribute to the observed angle errors. Further tests on the $S11$ parameter show that in between 400MHz and 1GHz, the antenna functions according to the theory. This is shown in the following Figure 7.35, Figure 7.36, Figure 7.37 and Figure 7.38:

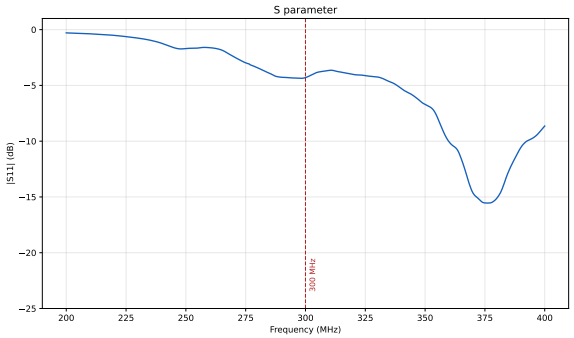


Figure 7.35: S parameter at 300MHz with the theoretical antenna height

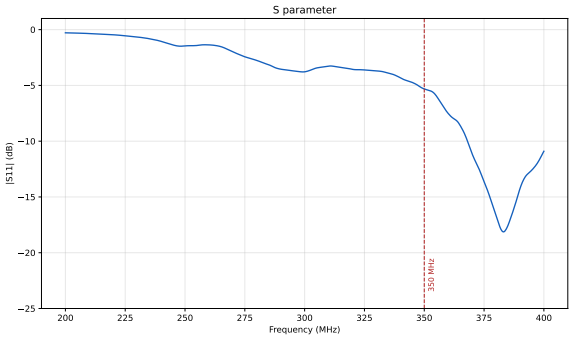


Figure 7.36: S parameter at 350MHz with the theoretical antenna height

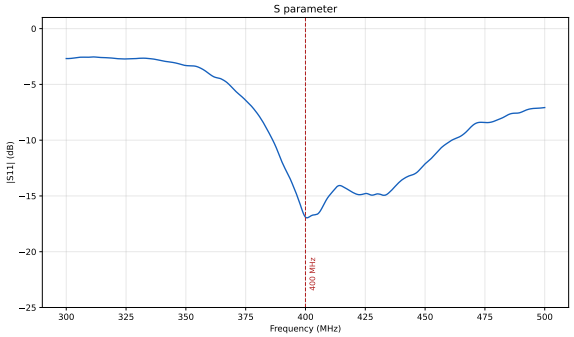


Figure 7.37: S parameter at 400MHz with the theoretical antenna height

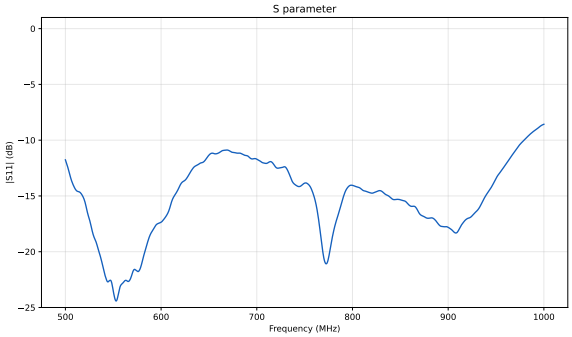


Figure 7.38: S parameter from 500MHz to 1GHz with the theoretical antenna height

A possible alternative for these antennas could have been a maritime VHF antenna, as shown in Figure 7.39. Unfortunately, only one of this antenna was available.



Figure 7.39: Maritime VHF antenna

7.5 Conclusion

In this chapter, the setups were tested on the site of DLR and validated. Then, the first measurement campaign, which did not achieve useful results due to the absence of a common ground plane for the antennas, was performed. After creating a new setup with a suitable grounding, the second measurement campaign took place. Here, the overall results were better, while showing the best performance for the ULA in contrast to the cross or the pentagon array. Further investigations show that the antennas appear to be a part of the problem since their S parameter is not optimal for their theoretical height at 162 MHz.

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

Conclusion

Since the monitoring of vessels has become essential, due to the constantly growing maritime traffic, AIS messages were introduced in 1998. However, these messages are neither authenticated nor verified. This results in the problem that vessels can purposely broadcast a false position. Although there are methods that detect those anomalies, they either only check the consistency of the messages themselves or require a large infrastructure, such as several synchronised stations, satellite images or radars.

The aim of this project was to provide a detection of false positions, that is independent and only relies on the received radio signal. Since the AIS signal is always emitted from the vessel's actual position, its DoA can be measured by an antenna array. This can be further compared to the direction that is expected because of the position that is reported in the message. For the implementation, a KrakenSDR was used at 162 MHz. While the three DoA estimation methods Bartlett, MUSIC and ESPRIT were presented, ultimately MUSIC was chosen. Additionally, with the help of theoretical simulations, ULA, cross array and pentagon array were compared. After that, the processing pipeline, allowing to receive and decode AIS messages and record the AIS burst from each channel, was developed. After some testing on the site of DLR the three arrays were compared with real data. Indeed two measurement campaigns took place in Altwarp. The first measurements in Altwarp did not show any good results, since the antennas did not have the right setup. After correcting this, the second measurements showed better results. Even though the DoA of some target estimates contained errors, it still followed their motion. The measurements also showed that the ULA works the best. These results are encouraging, especially since there was a design flaw of the antennas that was found later.

However, the results are unfortunately not precise enough yet to determine if a mismatch between the reported and estimated position is actually coming from falsified AIS data or from an estimation error. Based on the analysis of the antenna, the next step would be to use different antennas, since with a better signal captured, the DoA may be sufficiently accurate to detect the mismatch. Another step could be the development of a system that detects collisions between AIS messages. In order to adapt MUSIC and perform the correct DoA estimation in particularly dense traffic zones, it is important to know if two AIS messages are present in the same burst. As a final step, the DOA estimation could be performed using three KrakenSDR at the same time at different locations, in order to triangulate the vessel and obtain a more accurate position. The entire process would need to be automated for it to be fully operational.

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

Acronyms

ADC	Analogue-to-Digital Converter
AIS	Automatic Identification System
CPI	Coherent Processing Interval
CRB	Cramér-Rao Bound
CRC	Cyclic Redundancy Check
CSTDMA	Carrier Sense Time Division Multiple Access
DLR	Deutsches Zentrum für Luft- und Raumfahrt
DoA	Direction of Arrival
ESPRIT	Estimation of Signal Parameters via Rotational Invariance Techniques
EVD	Eigenvalue Decomposition
GMSK	Gaussian Minimum Shift Keying
GNSS	Global Navigation Satellite System
HDLC	High-Level Data Link Control
PNT	Positioning, Navigation and Timing
LoS	Line of Sight
MMSI	Maritime Mobile Service Identity
MUSIC	Multiple Signal Classification
NMEA	National Marine Electronics Association
NRZI	Non-Return to Zero Inverted
PLL	Phase-Locked Loop
PPM	Parts Per Million
RMS	Root Mean Square
RMSE	Root Mean Square Error
RSSI	Received Signal Strength Indicator
SAR	Synthetic Aperture Radar
SDR	Software Defined Radio
SMA	SubMiniature version A
SNR	Signal-to-Noise Ratio
SOTDMA	Self-Organized Time Division Multiple Access
TDMA	Time Division Multiple Access

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

TDOA	Time Difference of Arrival
ULA	Uniform Linear Array
VNA	Vector Network Analyser

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

Bibliography

[1] Worldwide AIS, "Understanding marine traffic density and its impact on global trade." https://www.worldwideais.org/post/understanding-marine-traffic-density-global-trade, 2025.

[2] German Aerospace Center (DLR), "About us." https://www.dlr.de/en/dlr/about-us/, 2026.

[3] German Aerospace Center (DLR), "Department: Nautical systems." https://www.dlr.de/en/kn/about-us/departments/nautical-systems, 2026.

[4] KrakenRF, "KrakenSDR." https://www.krakenrf.com/about-krakensdr, 2020.

[5] KrakenRF, "HeIMDALL DAQ firmware." https://github.com/krakenrf/heimdall_daq_fw, 2021.

[6] E. S. Raymond, "AIVDM/AIVDO protocol decoding." https://gpsd.gitlab.io/gpsd/AIVDM.html, 2023.

[7] R. Prévost, *Décodage et localisation AIS par satellite*. PhD thesis, Institut National Polytechnique de Toulouse, 2013. HAL.

[8] Z. Xie, E. Tu, X. Fu, G. Yuan, and Y. Han, "AIS data-driven maritime monitoring based on transformer: A comprehensive review." https://arxiv.org/abs/2505.07374, 2025.

[9] International Electrotechnical Commission, "Maritime navigation and radiocommunication equipment and systems — automatic identification systems (AIS) — part 2: Class a shipborne equipment of the automatic identification system (AIS) — operational and performance requirements, methods of test and required test results," Standard IEC 61993-2, International Electrotechnical Commission, 2018.

[10] International Telecommunication Union, "Technical characteristics for an automatic identification system using time division multiple access in the VHF maritime mobile frequency band," Recommendation ITU-R M.1371-5, International Telecommunication Union, 2014.

[11] International Electrotechnical Commission, "Maritime navigation and radiocommunication equipment and systems — class b shipborne equipment of the automatic identification system (AIS) — part 1: Carrier-sense time division multiple access (CSTDMA) techniques," Standard IEC 62287-1, International Electrotechnical Commission, 2017.

[12] K. Wolsing, L. Roepert, J. Bauer, and K. Wehrle, "Anomaly detection in maritime AIS tracks: A review of recent approaches," *Journal of Marine Science and Engineering*, vol. 10, no. 1, p. 112, 2022.

[13] T. Emmens, C. Amrit, A. Abdi, and M. Ghosh, "The promises and perils of Automatic Identification System data," *Expert Systems with Applications*, vol. 178, p. 114975, 2021.

[14] A. Karim, S. Utama, and M. A. Arifin, "Algorithms for detecting vessel spoofing of space-based AIS data," in *AIP Conference Proceedings*, vol. 2366, p. 060006, 2021.

[15] M. Louart, J.-J. Szkolnik, A.-O. Boudraa, J.-C. Le Lann, and F. Le Roy, "Detection of AIS messages falsifications and spoofing by checking messages compliance with TDMA protocol," *Digital Signal Processing*, vol. 136, p. 103983, 2023.

[16] J. Winkel, T. Willems, C. O'Driscoll, and I. Fernández-Hernández, "SeaSpoofFinder – potential GNSS spoofing event detection using AIS," 2026. arXiv:2602.16257.

[17] F. Spataro, P. Pavia, R. Roscigno, R. Torres, D. Bibby, and M. Cossu, "AIS P/L on SAR satellite: the Copernicus Sentinel-1 solution," in *Proceedings of the European Conference on Synthetic Aperture Radar (EUSAR 2021)*, pp. 329–334, VDE Verlag, 2021.

[18] F. Katsilieris, P. Braca, and S. Coraluppi, "Detection of malicious AIS position spoofing by exploiting radar information," in *Proceedings of the 16th International Conference on Information Fusion (FUSION)*, pp. 1196–1203, 2013.

[19] F. Mazzarella, M. Vespe, A. Alessandrini, D. Tarchi, G. Aulicino, and A. Vollero, "A novel anomaly detection approach to identify intentional AIS on-off switching," *Expert Systems with Applications*, vol. 78, pp. 110–123, 2017.

[20] F. Papi, D. Tarchi, M. Vespe, F. Oliveri, F. Borghese, G. Aulicino, and A. Vollero, "Radiolocation and tracking of automatic identification system signals for maritime situational awareness," *IET Radar, Sonar & Navigation*, vol. 9, no. 5, pp. 568–580, 2015.

[21] A. Winter, N. Skolova, A. Morrison, O. Hasler, and T. A. Johansen, "Low-cost angle of arrival-based auxiliary navigation system for UAV using signals of opportunity," in *Proceedings of the 9th European Conference for Aeronautics and Space Sciences (EUCASS)*, 2022.

[22] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[23] H. Krim and M. Viberg, "Two decades of array signal processing research: The parametric approach," *IEEE Signal Processing Magazine*, vol. 13, no. 4, pp. 67–94, 1996.

[24] P. Stoica and R. L. Moses, *Spectral Analysis of Signals*. Upper Saddle River, NJ: Pearson Prentice Hall, 2005.

[25] T. Nardi, "RTL-SDR: Seven years later." `https://hackaday.com/2019/07/31/rtl-sdr-seven-years-later/`, 2019.

[26] KrakenRF, "gr-krakensdr: GNU Radio blocks for the KrakenSDR." `https://github.com/krakenrf/gr-krakensdr`, 2024.

[27] KrakenRF, "krakensdr_doa: Direction of arrival software." `https://github.com/krakenrf/krakensdr_doa`, 2026.

[28] Raspberry Pi Ltd., "Raspberry Pi 4 Model B — specifications." `https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/`, 2019.

[29] Osmocom, "rtl-sdr: Osmocom librtlsdr library." `https://github.com/osmocom/rtl-sdr`, 2026.

[30] KrakenRF, "KrakenRF librtlsdr library." `https://github.com/krakenrf/librtlsdr`, 2026.

[31] J. S. Seybold, *Introduction to RF Propagation*. Hoboken, NJ: John Wiley & Sons, 2005.

[32] International Telecommunication Union, "Effects of tropospheric refraction on radiowave propagation," Recommendation ITU-R P.834-9, International Telecommunication Union, 2017.

[33] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*. Upper Saddle River, NJ: Pearson Prentice Hall, 4 ed., 2007.

[34] K. H. Mueller and M. Müller, "Timing recovery in digital synchronous data receivers," *IEEE Transactions on Communications*, vol. 24, no. 5, pp. 516–531, 1976.

[35] L. M. Richter, "pyais: AIS message decoding and encoding in Python (AIVDM/AIVDO)." `https://github.com/M0r13n/pyais`, 2026.

[36] C. E. Shannon, "Communication in the presence of noise," *Proceedings of the IRE*, vol. 37, no. 1, pp. 10–21, 1949.

[37] S. Shirvani-Moghaddam and F. Akbari, "A novel ULA-based geometry for improving AOA estimation," *EURASIP Journal on Advances in Signal Processing*, vol. 2011, p. 39, 2011.

[38] P. Stoica and A. Nehorai, "MUSIC, maximum likelihood, and Cramér–Rao bound," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 5, pp. 720–741, 1989.

This document may not be reproduced, modified, adapted, published, translated in any way, in whole or in part, nor disclosed to a third party without the prior written consent of ENAC. © 2026, ENAC

RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

CONFIDENTIALITY QUESTIONNAIRE & AUTHORIZATION TO DISSEMINATE A STUDENT MEMOIR

Full name: Michailas Romanovas **Promotion:** IENAC23

Email address: michailas.romanovas@dlr.de

Memoir title: Estimation of the angle of arrival of AIS signals with an SDR-based solution

Company: German Aerospace Center - DLR **Represented by:** Michailas Romanovas

Confidentiality questionnaire:

Please indicate the confidentiality level of this memoir and its defense. If you do not respond, the memoir will be considered non-confidential and its defense will be public.

We draw your attention to the fact that a memoir is a valuable source of information on companies and professions for students at our school. It is also a way to promote your organization to future trainees.

A confidential memoir will be referenced but not disseminated, despite all the interest it represents. We encourage students to include confidential information with regard to your company in appendices, thus allowing access to the summary of the work carried out during the internship.

Important information

The access to ENAC's memoir portal is exclusively granted to ENAC or DGAC members on authentication.

The information displayed may be different according to the confidentiality level and the authorization granted:

- for the confidential memoirs: **only the metadata** - title, abstract and keywords
- for the non-confidential memoirs and those for which the five years confidentiality period is over: **PDF file + metadata**.

⚠ If you do not want the memoir metadata to be included in ENAC's database, please contact us by e-mail at the following address: bibliotheque@enac.fr

RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

The PDF files of the memoirs are archived as follows:

- those for which the confidentiality period is unlimited are not archived
- those for which the confidentiality period of five years is not over are archived on a secure server located in ENAC
- at the end of the confidentiality period, the memoirs are processed and archived on a secure server located in the service provider's building.

To be completed by the company

☒ This memoir is not confidential

☐ This memoir is confidential

☐ For a limited period of 5 years from the defense

☐ For an unlimited period

☐ Only the appendices of this memoir are confidential

☐ For a limited period of 5 years from the defense

☐ For an unlimited period

Confidential defense : ☐ Yes ☒ No

The company, read and approved

At Neustrelitz, Germany

Precede the signature with the words "read and approved"

Date 01./10./2026

Read and approved.

Michailas Romenskis

RÉPUBLIQUE
FRANÇAISE
Liberté
Égalité
Fraternité

ENAC

Authorization for dissemination:

The purpose of this authorization is to define the terms of dissemination of the memoir, in order to facilitate access to knowledge and to contribute both to the recognition of the work carried out by the students and to the reputation of the school.

To be completed by the student

☐ I authorize the dissemination of my memoir on the Internet

☐ I authorize the dissemination of my memoir only to the ENAC and DGAC community

☑ I do not authorize the dissemination of my memoir

ENAC undertakes, in all cases where it makes use of this authorization for dissemination, to include the name of the author. The student is however aware of the fact that, beyond the indication of the prohibition, given the state of technology, ENAC does not have the means to prevent the total or partial reproduction of the work from a website.

In accordance with the French *Law No. 78-17 of January 6, 1978 relating to information technology, files and freedoms*, students may modify their personal information at any time, as well as this authorization for dissemination, by sending a simple letter or email to the ENAC library: bibliotheque@enac.fr

The student,

At Paris

Precede the signature with the words "read and approved"

Date 06/10/2026

« Read and approved »

