

Research Project and Seminar

Informatik-Ingenieurwesen

Orthogonal Codes for Acoustic Underwater Localization

by

Sergej Keller

January 2023

Supervised by

Fabian Steinmetz

Institute of Autonomous Cyber-Physical Systems, Hamburg University of Technology

First Examiner

Prof. Dr.-Ing. Bernd-Christian Renner

Institute of Autonomous Cyber-Physical Systems
Hamburg University of Technology

Table of Contents

1	Introduction	1
1.1	Motivation	2
1.2	Setup and objectives	2
2	Code fundamentals	3
2.1	Pseudo-random codes	3
2.1.1	Gold codes	4
2.1.2	Kasami codes	4
2.2	Comparison	4
3	Implementation	7
3.1	Pulse shaping	7
3.2	Shift to carrier	8
3.3	Band pass filter	9
3.4	Shift to base band	9
3.5	Low-pass filter	9
3.6	Peak detection	9
3.7	Localization method	11
4	Evaluation	13
4.1	Simulation	13
4.1.1	Watermark	13
4.1.2	White noise	14
4.1.3	Localization simulation	14
4.2	Localization field testing	17
5	Conclusion	21
Bibliography		23
A Localization Formula		25

TABLE OF CONTENTS

Introduction

Orthogonal codes are a widely adopted technology in various fields, including telecommunications and wireless communications. One of the most promising areas for the application of orthogonal codes is in underwater acoustic localization. Acoustic signals are widely used for underwater communication due to their ability to propagate well in water. However, the harsh underwater environment poses many challenges for accurate localization, such as multipath propagation and noise.

This research project proposes the use of pseudorandom binary codes for underwater acoustic localization. The codes are processed and then transmitted via hydrophones, and multiple sending anchors are used to increase the accuracy of the localization. Time-difference-of-arrival (TDOA) correlation techniques are employed to estimate the position of a target in 3D space using the signals received by the anchors. The use of orthogonal codes can improve the robustness of the localization system against noise and multipath propagation. The aim is to examine the performance of various code types and design strategies to select the optimal code. This research endeavors to enhance the precision and dependability of underwater localization systems through the utilization of orthogonal codes, with the ultimate goal of facilitating a plethora of applications, including but not limited to oceanographic studies and undersea navigation.

One specific example of this application would be the use in the BlueROV2 1.1, a remotely operated underwater vehicle, which would greatly benefit from this method of localization.



■ **Figure 1.1:** BlueROV2 from Blue Robotics Inc

1 INTRODUCTION

1.1 Motivation

The principle of Time Difference of Arrival (TDOA) has been widely adopted in GPS applications for localizing small devices or mobile vehicles. However, this principle has yet to be fully explored for its potential in underwater localization. Unlike GPS, which utilizes electromagnetic waves, acoustic communication is required for underwater localization due to the high damping of electromagnetic waves underwater. This can be achieved through the use of hydrophones, piezoelectric transducers that are capable of receiving and sending underwater acoustic signals.

The current underwater localization systems rely on the underwater vehicle transmitting signals to anchored beacons. However, this project aims to reverse this approach, allowing multiple underwater vehicles to localize themselves with the help of signals transmitted by the beacons. In theory, multiple underwater vehicles should be able to simultaneously localize themselves without interference, making this project a valuable contribution to the development of more accurate and reliable underwater localization systems.

1.2 Setup and objectives

The study will commence with an examination of orthogonal codes, signal processing and modulation techniques. This will include a review of cross-correlation and auto-correlation concepts to ensure a solid understanding of the fundamental principles. Subsequently, the focus will shift towards understanding underwater communication and the transmission of acoustic signals through piezoelectric transducers.

Based on the knowledge gained, the study will move forward with the implementation of a self-localization system in 3D space using four anchors, which will be based on orthogonal codes and an appropriate peak detection algorithm. The initial phase of development will take place in MATLAB, followed by a switch to Python. The only exception will be the use of a benchmark tool, which will remain in MATLAB. Upon completion of the implementation, the performance of the algorithms will be evaluated in various scenarios. If the results are found to be satisfactory, the evaluation will be extended with a simplified field test in real-world conditions.

In summary, the research project aims to uncover valuable insights into using orthogonal codes for self-localization in underwater environments, and to implement a localization system. The project will conclude with a comprehensive evaluation and comparison of the implemented algorithm.

Code fundamentals

To attain a higher level of localization accuracy, there are two primary goals that must be pursued.

First, the code used for underwater localization should have an auto-correlation function that approaches a Dirac impulse. This is important because it allows for more efficient detection through the use of correlation techniques.

The second factor to consider is the cross-correlation properties of the code. It is essential that these attributes meet certain criteria in order to improve separation from other sequences. Mathematically speaking, this means that the codes should be orthogonal to each other, or at least approaching orthogonality. This will be particularly useful in real-world scenarios where noise, reflections, and other artifacts may be present. In summary, by striving to achieve both of these objectives, it is possible to significantly improve the localization accuracy.

There are a couple of techniques to generate PN sequences. Most of these methods use linear feedback shift registers to generate the codes by an initial condition or seed value. In this project I will concertize my research on gold codes, kasami codes and the basic m-sequences which are used for generating gold codes. All three code types are based on linear shift registers.

2.1 Pseudo-random codes

M-sequences are defined as binary PN codes, which are generated by linear shift registers with feedback. The sequences are periodic, and contain an equal number of zeros and ones [PS08]. Maximum length sequences need to fulfill certain criteria. First its length is defined by $N = 2^n - 1$ where n is the maximum degree of the generator polynomial $f(X)$ [SP80].

$$|u| = 2^n - 1 = N, \text{ from polynomial } h(x) \text{ of degree } n \quad (2.1)$$

2 CODE FUNDAMENTALS

$$\frac{N}{\gcd(N, q)} = N' \quad \text{from decimation polynomials } \widetilde{h(x)} \quad (2.2)$$

Second the cross-correlation between m-sequences must take three values only, which are -1 , $-t(n)$, $t(n) - 2$. With it $t(n)$ is defined by $1 + 2^{\lfloor 0.5(n+2) \rfloor}$ [SP80]. If every pair of m-sequences is considered a preferred pair, they form a maximal connected set, which has a limited cardinality. Experiments conducted by Gold and Koptizke demonstrated that the number of such connected sets is limited, and that a primitive polynomial is required for an m-sequence [GK65].

2.1.1 Gold codes

M-sequences alone may not have optimal cross-correlation properties, which can affect their orthogonality. However, when two m-sequences of the same length are combined through a modulo-2 sum, their orthogonal properties are improved. The resulting codes are known as Gold codes [PS08].

Recent research indicates that some Gold codes have a high similarity to a Gaussian random variable, making them a suitable choice as orthogonal and pseudo-random codes [MD].

$$Gold(u, v) = \{u, v, u \oplus v, u \oplus (v \ll 1), \dots, u \oplus (v \ll N-1)\} \quad (2.3)$$

2.1.2 Kasami codes

Kasami sequences are constructed in the similar fashion by using m-sequences with the exception that a second sequence, which is used in the modulo sum, is formed by decimating the default m-sequence by $2^{m/2}$ [PS08] [SP80] [PPWW72]. As a result, only one generator polynomial is needed, but this also limits the number of code variations in the set.

$$w = u[2^{N/2} + 1] = \{u_1, \dots, u_i, \dots, u_N | \text{take every } i\text{-th bit of } u\} \quad (2.4)$$

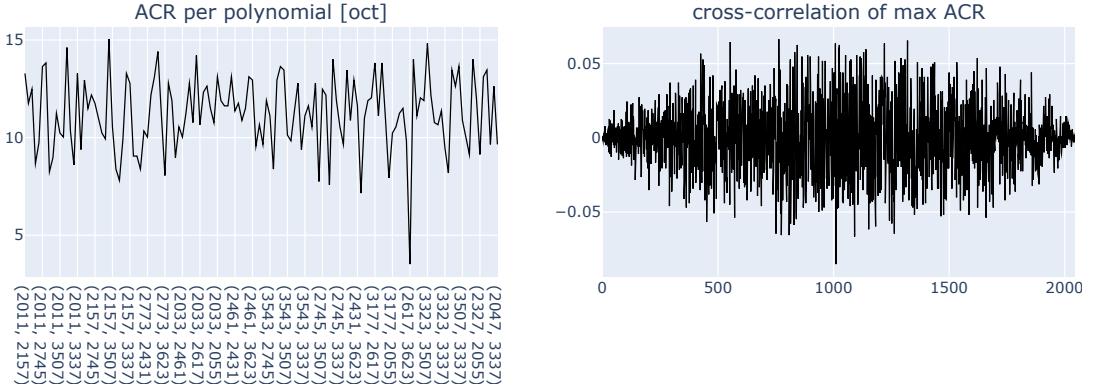
$$Kasami(u) = \{u, u \oplus w, u \oplus (w \ll 1), \dots, u \oplus (w \ll 2^{N/2} - 2)\} \quad (2.5)$$

2.2 Comparison

For the localization process by orthogonal codes certain criteria needs to be met, which were named at the beginning of chapter 2. To compare the before explained code types two measures are introduced.

The first one is the peak to side-lobe ratio (PSR) 2.6. This measure is defined by subtracting the mean from the peak of the auto-correlation. Then this value get divided by the standard deviation of the same auto-correlation. A higher PSR value signifies a lower error between

2.2 COMPARISON



■ **Figure 2.1:** Evaluation of gold sequences by AC ratio

the auto correlation and the perfect Dirac resulting in better detection capability. The second one is the ratio between the auto-correlation peak and the maximum of the cross-correlation (ACR) 2.7. A higher value indicates better code separation qualities.

The comparison is done by sampling pairs from degrees six to ten N times from the set of random sequences. These pairs are then used for generating the wanted pseudo-random codes like Gold or Kasami. Afterwards for all three code types both measures are applied.

$$PSR = \frac{\max\{x_{ac}\} - \bar{x}_{ac}}{\sigma_{ac}} \quad (2.6)$$

$$ACR = \frac{\max\{x_{ac}\}}{\max\{x_{cc}\}} \quad (2.7)$$

In this evaluation of data, three types of codes were compared. Maximum length sequences, Gold codes, and Kasami codes. The performance of each code was assessed using two ratios, the ACR and the PSR.

From preferred polynomial all possible maximum length sequences, Gold sequences and Kasami sequences are generated. Then both measures are applied on the cross-correlation and auto-correlation functions of the random codes. The PSR and ACR measures are compared against the used polynomials. Also the best case of PSR and ACR are plotted by their given correlation function.

Maximum length sequences hold the best auto-correlation properties in comparison to its competitors. But it takes peaks in its cross-correlation, making it a rather unsuitable option for orthogonal separation. The Kasami sequence has a slight superior cross-correlation but still small peaks peeking through. The winning codes are Gold codes because of the solid auto-correlation and very fitting cross-correlation properties 2.1, 2.2. Furthermore, Gold codes yield better scalability compared to Kasami codes, because of its larger set. Only its auto-correlations lags a bit behind its competitors but orthogonality is as much as important.

2 CODE FUNDAMENTALS

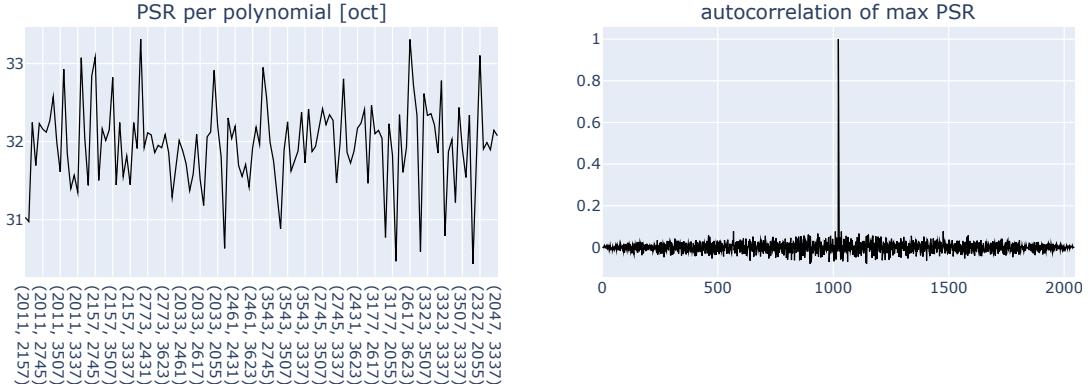


Figure 2.2: Evaluation of gold sequences by PS ratio

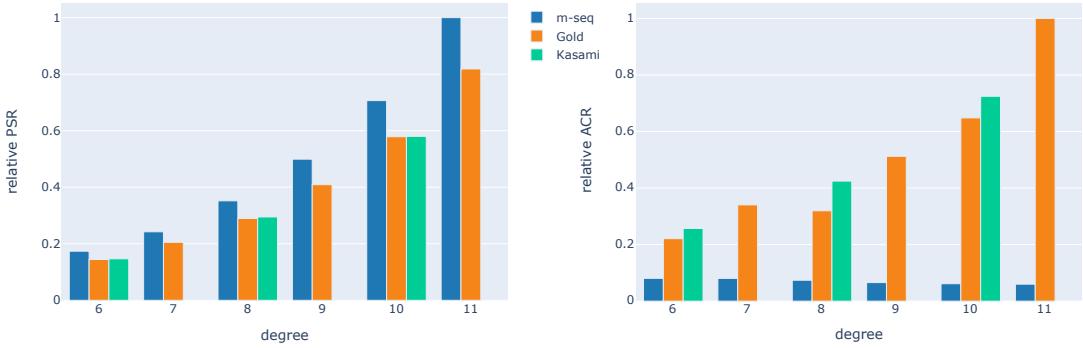


Figure 2.3: Evaluation by PSR and ACR for degrees 6 to 11 of all three code types

To get more valuable data, elements from each set of codes are sampled uniformly ($N = 1000$) and afterwards the evaluation parameters are applied. The results yields that Gold codes had the least increase by PSR, but were the second best by ACR. Kasami codes were only slightly better by both PSR and ACR than Gold codes, but had a smaller set of codes available. Maximum length sequences had the highest PSR, but the worst ACR 2.3.

Based on these findings, it can be concluded that Gold codes are the best choice for this application. While maximum length sequences had the highest PSR, they performed poorly in terms of ACR. Gold codes, on the other hand, had a good balance of performance in both ratios, and also had a large set of codes available. In addition, Gold codes demonstrated better cross-correlated detection compared to maximum length sequences.

Implementation

The Implementation covers the process of implementing a localization system using orthogonal codes in underwater environments. It starts with the description of pulse shaping using a cosine FIR filter. Then, the signal is shifted to the carrier frequency of 62.5 kHz. To ensure the desired frequency characteristics, a band pass filter of 5th order Butterworth is applied. The signal is then shifted back to baseband, followed by a low-pass filter to remove any high-frequency components. The peak detection is performed using a CFAR (Constant False Alarm Rate) algorithm. Finally, the localization method is presented that uses the information obtained from peak detection to determine the position.

The whole chain of processes, including pulse shaping, shift to carrier, band pass filtering, shift to baseband, low-pass filtering, peak detection, and the localization method, is implemented in Python.

3.1 Pulse shaping

The raw code which was previously generated is first put through an sign function, which sets its mean to zero. The discontinuous signal holds an infinite bandwidth because it consists of rectangular pulses. These pulse spans have infinity frequency, which are impossible to implement for acoustic transmission or at all. Therefore a restriction to a certain bandwidth must be introduced. Such a reduction is possible by applying an low-pass filter.

Because limiting the signals bandwidth introduces a damped oscillation, which leads to incorrect decoding of received data [Gen07], a appropriate choice of filter would be a the raised cosine 3.1. Such a pulse filter is defined by a squared cosine, which decreases its amplitude in frequency. In our case the roll off factor α is set to 0.125 and symbol length T_{sym} to the inverse of our target bandwidth 20 kHz.

Before applying the filter the generated codes need to be up-scaled for our target sampling rate. The relation between sampling rate f_s and up-sampling factor Sp_s is $Sp_s = f_s \cdot T_{sym}$.

3 IMPLEMENTATION

$$H(\omega) = T_{sym} \cdot \cos^2 \left(\frac{T_{sym}(\omega - \pi(1 - \alpha)/T_{sym})}{4\alpha} \right), \quad \omega = 2\pi f \quad (3.1)$$

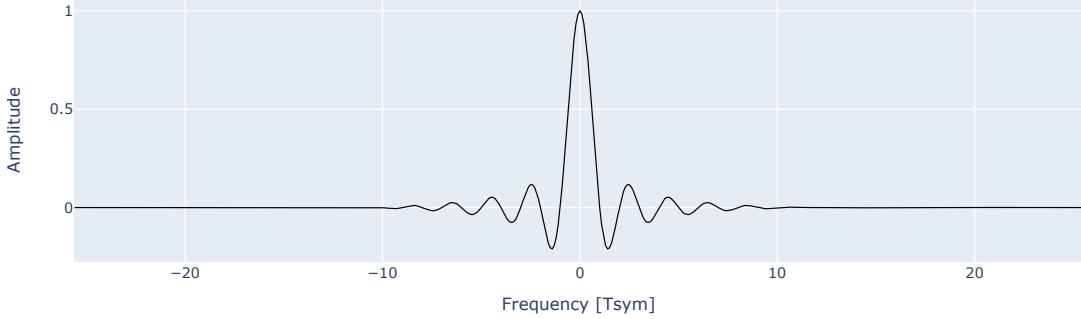


Figure 3.1: Section of cosine FIR with a resolution of 1024

3.2 Shift to carrier

Now that our base band signal $tSigBB[k]$ has its target bandwidth and sample rate only the frequency shift to the carrier frequency f_c of 62.5 kHz is necessary. That procedure is accomplished by multiplying our sampled signal by the exponential function, where f_c is passed to its exponent. An Frequency shift of the transmitted signal can be achieved by multiplying it with a complex exponential function of the form $e^{-2\pi j f_c k}$. This results in a phase rotation in the frequency domain and effectively shifts the spectrum of the signal by f_c . The resulting signal could hold imaginary parts, hence only the real part is passed through for transmission.

The signal is now in an almost appropriate state for the use inside an localization algorithm, where the delays of the signals can be detected by the help of cross-correlation techniques.

$$x_{tSigTB}[k] = Re\{x_{tSigBB}[k] \cdot e^{-2\pi j f_c k}\} \quad (3.2)$$

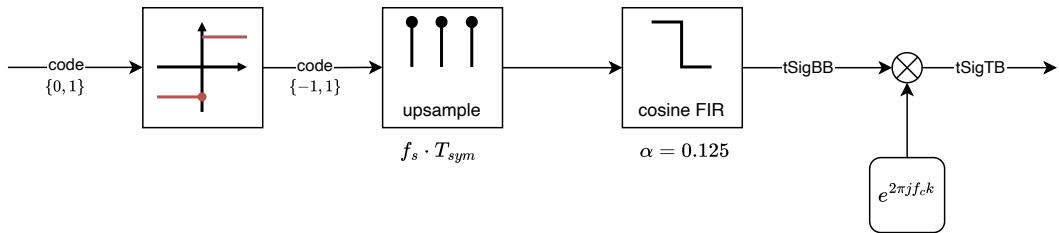


Figure 3.2: Processing of generated signal

3.3 Band pass filter

On the receiver side all signals are received as a sum, which is mixed by artifacts of signal reflections and random noise from the electronics.

The received signal may also have noise around its bandwidth because of its frequency shift. Thus, we only pass through frequencies inside our frequency band by applying a butterworth band pass filter. A flat magnitude is favorable because only frequencies of the base-band should be passed through. The filter gets applied after shifting back to the base-band. Such a filter, namely a maximally flat magnitude filter, approximates this goal. The roll-off decreases by increasing the order of the system.

The critical frequencies of the applied filter are $f_c \pm \frac{bw}{2}$ by an Order of 5. Thus, frequencies get removed which are not inside the spectrum of interest. To remove shifts in time the filter is applied forwards and backwards following a doubling of its order.

3.4 Shift to base band

Shifting the transmitted signal back to its original frequency is feasible by just changing the sign of f_c . In this case also the imaginary part can be retained.

$$x_{tSigBB}[k] = x_{tSigTB}[k] \cdot e^{2\pi j f_c k} \quad (3.3)$$

3.5 Low-pass filter

Due to the presence of residuals within the transfer-band originating from the product of frequency shift, a low-pass filter was utilized. This filter is a Butterworth filter, which had previously been employed by the same order. Additionally, the filter is applied in both forward and backward directions, thereby eliminating any time shift.

3.6 Peak detection

The received signal, consisting of summed delayed signals, cross-correlated by every anchor. If the signal is not reflected the peak in cross-correlation would be obvious. But by the introduction of noise and water reflections a higher rate of similar peaks appear. To suppress these effects a CFAR Algorithm [Roh11] is applied to only detect the first reflected peak resulting in lower false alarms of peaks.

CFAR works by using multiple values intervals. The most outer one could be described as a train bin and is used to get an estimation of the signals noise. Especially CA-CFAR (cell-averaging constant false alarm rate) uses averaging to estimate the noise by measured

3 IMPLEMENTATION

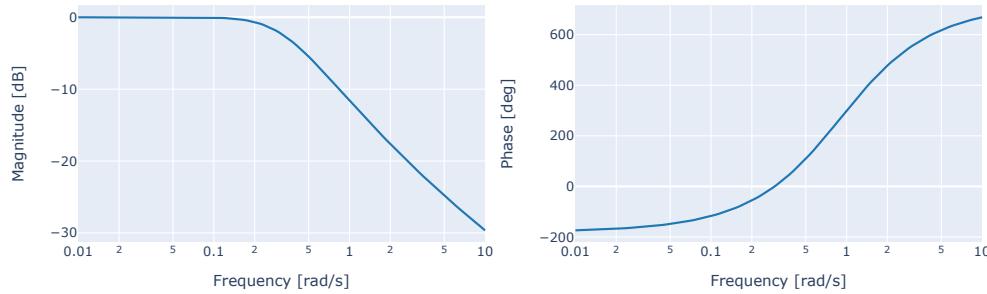


Figure 3.3: Bode plot of 5th order Butterworth low-pass filter

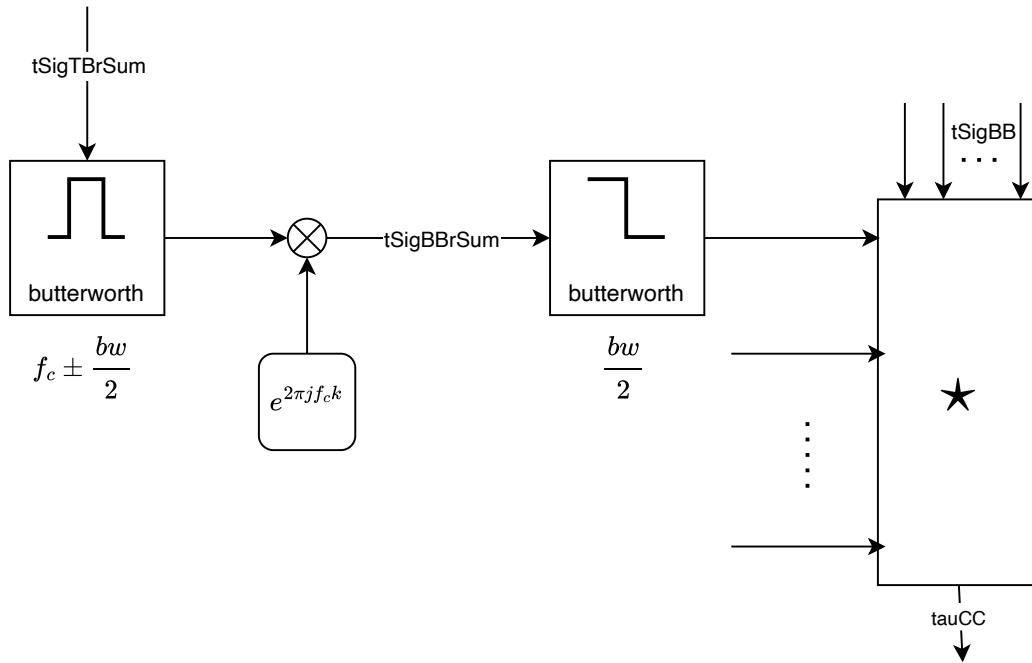
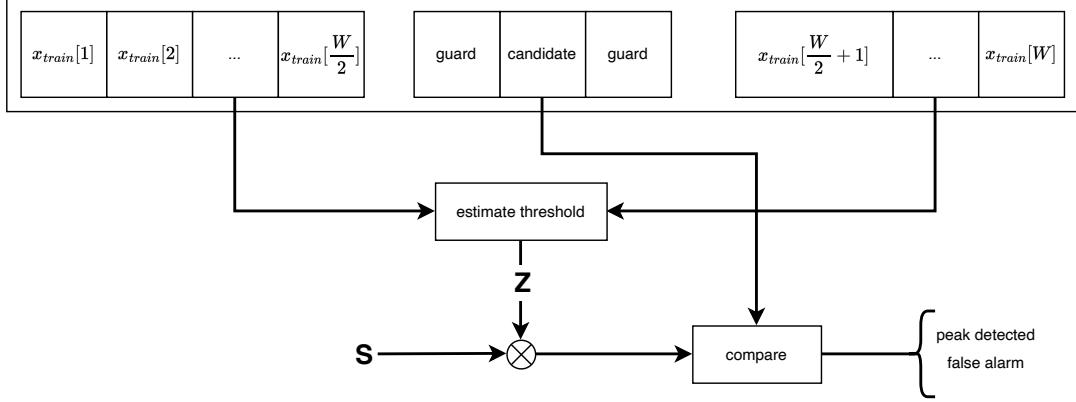


Figure 3.4: Processing of received signal

cells. The bordering bin, defined as the guard cells, is used to reduce self-interference of the peaks. Thus, increasing window sizes W results in better noise estimating but overall detectability is still limited by the sample rate [Roh11][rad]. By knowledge of measured peak widths a optimal guard interval can be figured.

The calculated threshold is than scaled by factor S depending on a formula based on the false alarm rate η . The higher the false alarm rate, the weaker high amplitude peaks gets included by the estimated threshold 3.4.

$$S = 2W \left(\eta^{-1/2W} - 1 \right) \quad (3.4)$$



■ **Figure 3.5:** CFAR threshold peak detection procedure [Roh11]

$$T = S \cdot Z, \quad Z_{CA} = \sum_{i=1}^W \frac{1}{W} x_{train} \quad (3.5)$$

3.7 Localization method

The initial condition for the localization are four anchors S_i with their coordinates $\{x_i, y_i, z_i\}$ and the target S which is to be located. By multiplying relative delays by the speed of sound c which is approximately set to 1500 m s^{-1} , the distance d_{ij} between the reference anchor S_0 and S_i is calculated [WXX11].

$$d_{ij} = c \cdot \tau_{ij} = c \cdot (t_i - t_j), \quad \text{absolute delays } t_k, k \in \{0, 1, 2, 3\} \quad (3.6)$$

$$x_{ji} := x_j - x_i, \quad y_{ji} := y_j - y_i, \quad z_{ji} := z_j - z_i \quad (3.7)$$

Every TDOA estimate creates hyperbolic curves which anchors are placed at its foci. By rearranging the derivation of hyperbola intersections the following substitutes can be defined [BM02].

$$A = \frac{d_{02}x_{10} - d_{01}x_{20}}{d_{01}y_{20} - d_{02}y_{10}}, \quad B = \frac{d_{02}z_{10} - d_{01}z_{20}}{d_{01}y_{20} - d_{02}y_{10}} \quad (3.8)$$

(See appendix for complete formula)

A downside of this approach is the uncertainty of position z . Thus, additional information on bounds is necessary. The target won't get above sea level. Consequently, at least one

3 IMPLEMENTATION

boundary $z_{surface}$ which acts like a maximum can be set. The minimum value z_{ground} can be assumed as the lowest position achievable underwater.

$$z_{a,b} = \frac{N}{2M} \pm \sqrt{\left(\frac{N}{2M}\right)^2 - \frac{O}{M}} \quad (3.9)$$

$$z = \min \{ \max \{ z_a, z_b, z_{surface} \}, z_{ground} \} \quad (3.10)$$

One other solution to this issue is to utilize information about our previous location z' . Specifically, we can calculate the distance between our current position and the two potential candidates for the next estimate of z , and choose the candidate with the shorter distance.

$$z = \begin{cases} z_a & \text{if } |z_a - z'| < |z_b - z'| \\ z_b & \text{else} \end{cases} \quad (3.11)$$

The resulting x and y values of our target can then be calculated by the following formula using the selected z , completing the desired position vector \vec{x} .

$$\vec{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} Gz + J \\ Iz + H \\ z \end{bmatrix} \quad (3.12)$$

Evaluation

This evaluation outlines the process of testing the performance of the localization system using orthogonal codes in underwater environments. The chapter covers the simulation of the system, in which a benchmark is used to assess the performance with different levels of SNR by adding white noise.

Additionally, the chapter discusses the field testing of the algorithm, which is performed in a real-world scenario using three hydrophones configured as two sending anchors and one receiver. The simulation and field testing results provide valuable insights into the accuracy and reliability of the localization system.

4.1 Simulation

The simulation part involves the use of a Watermark benchmark [vWOJ] and the addition of Gaussian white noise with varying levels of signal-to-noise ratio (SNR) to the signals. The simulation then evaluates the performance by following a simulated path of positions in 3D space, using four anchors.

4.1.1 Watermark

The Watermark Simulation consists of a convolution or channel replay by an selected channel TVIR estimate. The channels consist of multiple dirac impulses of different strengths. Thus, reflections and reduced signal strength are simulated.

$$x_{tSigTB_r}[k] = \sum_{i=0}^N h[k, i] \cdot x_{tSigTB}[k - i] \quad (4.1)$$

4 EVALUATION

4.1.2 White noise

A additive Gaussian White Noise (GWN) generated by a desired Signal to Noise Ratio (SNR) between -20 dB and 20 dB in steps of 5 dB. From the general equation of the Signal to Noise Ratio we derive our noise standard deviation by transforming this ratio. The white noise is added after the simulation and before receiver filtering. To estimate the power of our signal a standard deviation estimation is used, which consists of all incoming signals by using its expected value. The Gaussian noise is generated by using a normal distributed random variable with its mean at zero and its standard deviation at $\frac{\bar{\sigma}}{SNR}$. Thus, M denotes the number of total anchors and N is the length of the corresponding signal.

$$\bar{\sigma} = \frac{1}{M} \sum_{i=0}^M \sigma_i, \quad \sigma_j = \sum_{k=0}^N tSigTBr_j^2[k] \quad (4.2)$$

$$n[k] = f_{GWN}[k] \cdot \frac{\bar{\sigma}}{SNR}, \quad GWN \sim \text{Normal}(0, 1) \quad (4.3)$$

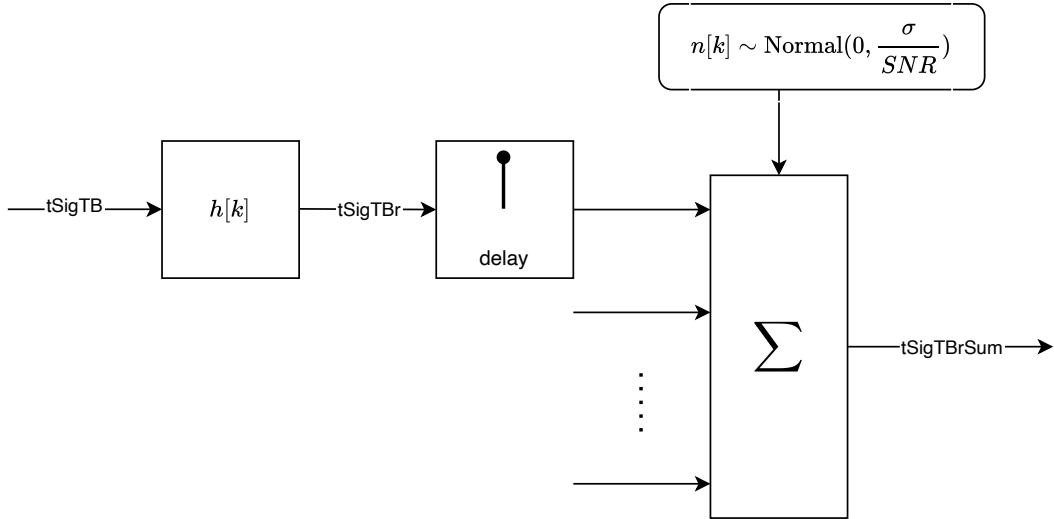


Figure 4.1: Simulation of acoustic signal underwater propagation

4.1.3 Localization simulation

To test the localization algorithm, several 3-dimensional space paths were simulated, wherein the generated points formed a helical curve that expanded in the z -direction 4.4. The positions of four anchors were utilized to calculate the TDOA values using multilateration techniques, which were subsequently incorporated into the localization algorithm 4.5. Further, an approximate value of 1500 m s^{-1} for the speed of sound c in water was used as a parameter in the

4.1 SIMULATION

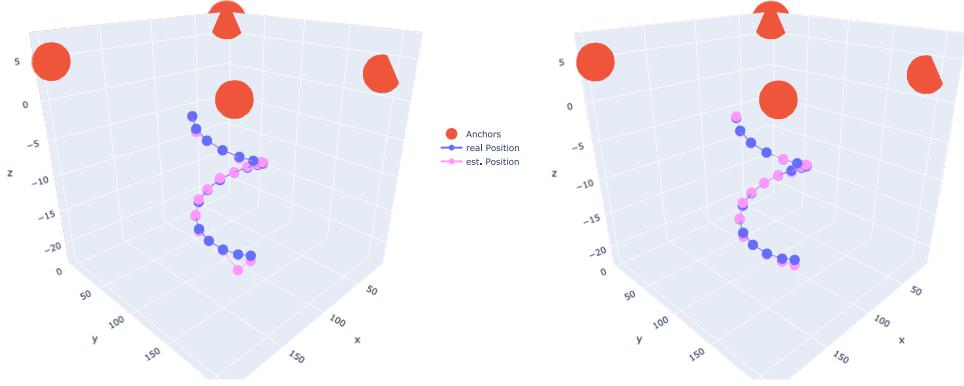


Figure 4.2: 3D position evaluation with code of 10th degree. Left one shows -5 dB and right one 20 dB signal strength created by additive noise.

calculations. Multiple runs, utilizing varying SNR's and watermark channels were conducted to evaluate the performance of the peak detection method.

$$\vec{x}_\phi(\alpha, \beta) = \begin{bmatrix} \alpha \cdot \sin \phi + \beta \\ \alpha \cdot \cos \phi + \beta \\ -|\phi| \leq z \leq -1 \end{bmatrix} \quad (4.4)$$

$$\tau_i = \frac{1}{c} \cdot (|S - S_0| - |S - S_i|) \quad (4.5)$$

The simulation system is based on 20 simulated positions. An almost rectangle coverage for the target is created using four anchor coordinates $S_0 = (15 \text{ m}, 1 \text{ m}, 17 \text{ m})$, $S_1 = (200 \text{ m}, 10 \text{ m}, 5 \text{ m})$, $S_2 = (195 \text{ m}, 210 \text{ m}, 6 \text{ m})$ and $S_3 = (16 \text{ m}, 190 \text{ m}, 3 \text{ m})$ to be located in. The circular curve is scaled by 30 m and has an offset of 100 m from the point of origin. The CA-FAR lower threshold is set to 0.2 to exclude accidental peak detection in low amplitude correlation phases.

The results of the simulation of position detection indicate that, as the noise level increases and the signal-to-noise ratio decreases, some positions cannot be accurately located 4.2. This intrusion of noise leads to a failure of the peak detection due to new peaks with high similarity to the real ones. The correlations reveal that, with increased noise, the side lobes of the correlation peaks become more prominent, making it difficult to detect peaks 4.3. Additionally, a higher CFAR threshold, which is a side effect of the additive noise, may result in the rejection of valid peaks.

By iterating through SNR's between -5 dB and 20 dB in steps of 1 dB shows that the mean error, which is the average absolute difference between the calculated and expected positions, increases when the SNR is less than 8 dB and jumps to even higher levels due to

4 EVALUATION

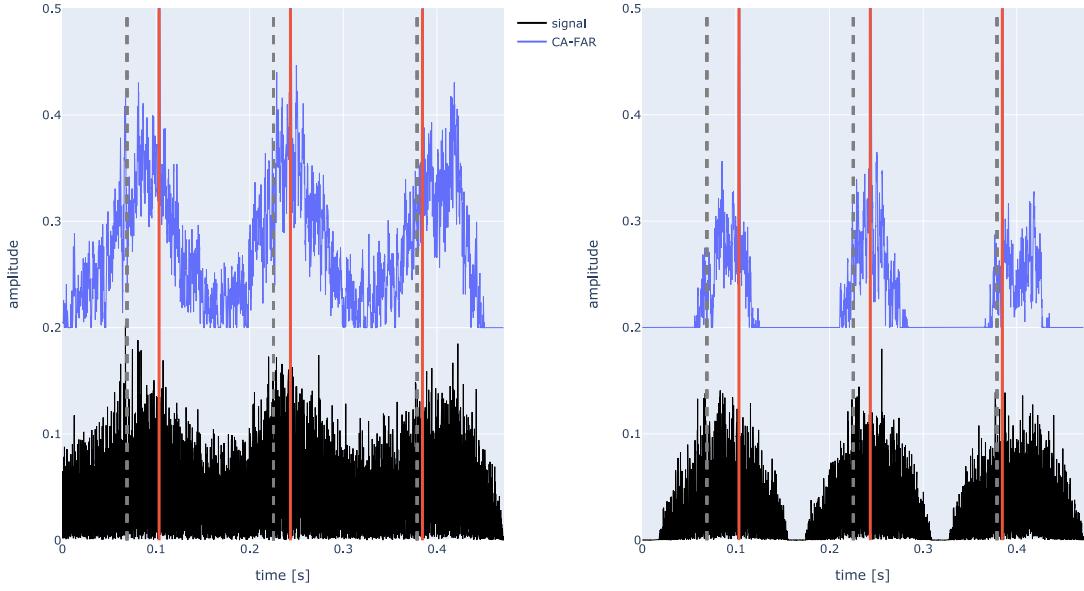


Figure 4.3: Correlation of an anchor for first three positions. The gray dotted line marks the first peak of the period, the red line one of the current correlating anchor.

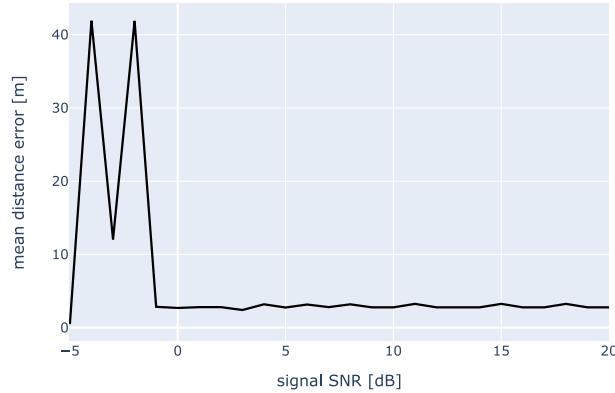


Figure 4.4: Relationship between SNR and mean distance error

the increasing likelihood of false peak detection 4.4. Continuing with the use of the -5 dB SNR the position detection system now incorporates watermark simulations as well. Due to multipath propagation caused by reflections from surfaces, the correlation amplitudes become even higher than those from additive noise 4.5. As a result, peak detection becomes less reliable, and it becomes necessary to adjust the false alarm rate.

4.2 LOCALIZATION FIELD TESTING

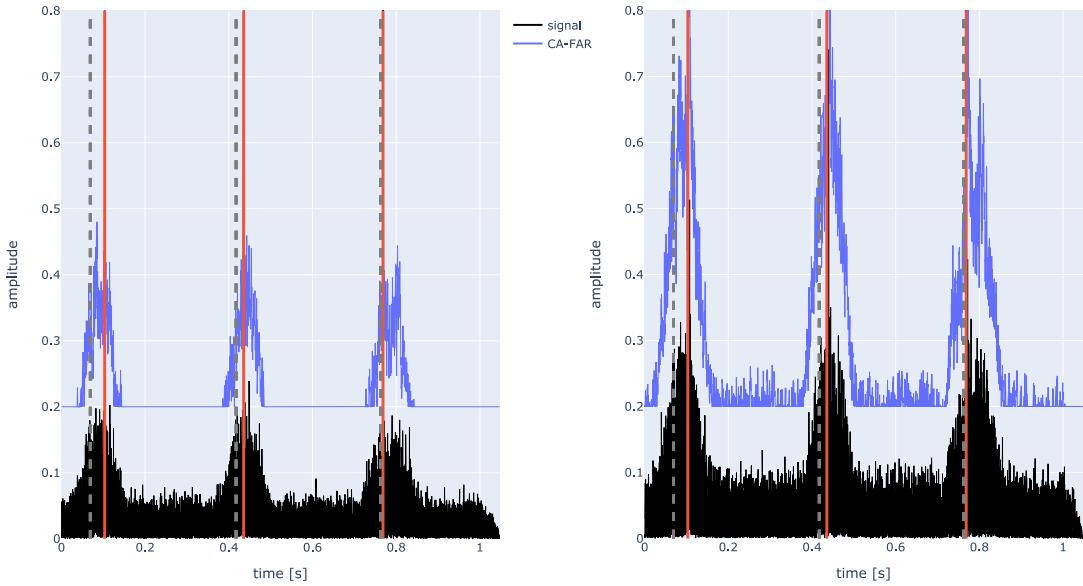


Figure 4.5: Correlation of an anchor for first three positions using watermark simulation channel PLANE1 at the right and CASTLE2 at the left, both with additive white noise of -5 dB

4.2 Localization field testing

A field test was conducted at a shoreline location. Three hydrophones were deployed at a depth of one meter below sea level. Anchors A and B were positioned with a distance of 4.1 m between them, with Anchor B located 10.74 m from the receiving hydrophone. During the test, the underwater speed of sound, as measured using a CTD Sensor (Conductivity, Temperature, and Depth), was 1430.3 m s^{-1} . The receiving hydrophone was also relocated to a second location, 5.56 m from Anchor B.

In the first run codes of degree ten were transmitted for a total time of 50 s. Afterwards the receiving hydrophone was moved 5.18 m further away from the anchors. Then again three test runs were done with code degrees of ten, nine and seven. Every run was repeated with a more decreased signal intensity for testing lower SNR's. The figures presented here show the measured and expected values of a position over time for a standard and decreased signal strength for all three degrees of code.

It is observed that with a code of 10th degree, there is a significant offset of approximately 60 cm between the measured and expected values 4.7. This offset is likely a result of synchronization problems between the oscilloscopes used in the experiment. However, it is also worth noting that despite this offset, the position demonstrates a high degree of stability with minimal variations of less than 10 cm. Additionally, by decreasing the SNR, it is observed that the position is shifted somewhat by under 20 cm. In conclusion, this demonstrates that despite the offset, the overall position remains relatively consistent. The next measure presents

4 EVALUATION



Figure 4.6: Location of field test, the markings represent the positions of the hydrophones. From left to right: receiver, sending anchor B and A. The CTD Sensor is depicted in the middle.

4.2 LOCALIZATION FIELD TESTING

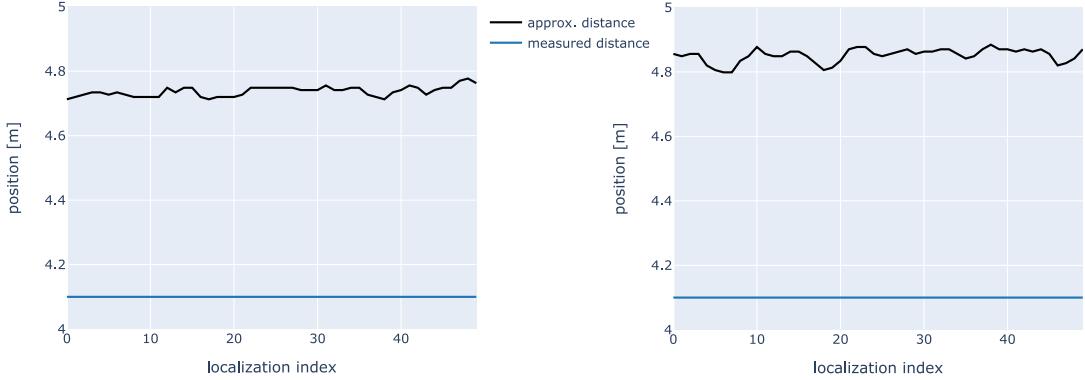


Figure 4.7: Evaluation of 10th degree code. High SNR at the left and low SNR at the right.

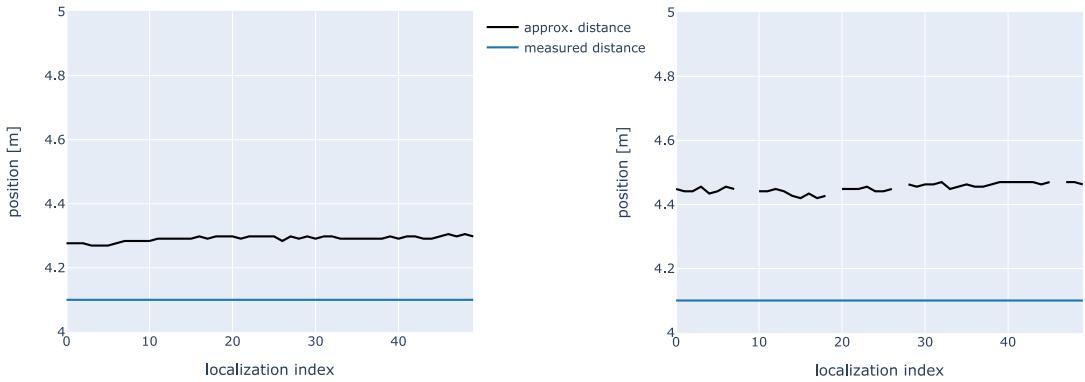


Figure 4.8: Evaluation of 9th degree code

the results of the position measurement when the code of 9th degree is used. It is observed that when the default SNR is in use, similar results are obtained as with codes of 10th degree. However, when the SNR is decreased, the peak detection fails 4.8. This is an important issue to consider, as the ability to accurately detect peaks is crucial for obtaining accurate position measurements. Additionally, values outside the interval from 4 to 5 were excluded. The failure of peak detection in this measure is likely due to the enlarged sidelobes, which are known to distract the CFAR algorithm. As a result, the performance of the system is worse compared to when the code is at 10 degrees. This highlights the importance of maintaining a high SNR and the potential impact of sidelobes on the performance of the algorithm.

The final measure presents the results of the position measurement with code of 8th degree. It is observed that this measure follows the downwards trend of performance that was previously noted with code of 9th degree. Specifically, it is found that both low and high SNR position detection's could not be completed without outliers resulting from failed peak detection 4.9. This further emphasizes the importance of maintaining a high SNR and the potential impact of sidelobes on the performance of the algorithm.

It is important to note that the false alarm rate and the minimum threshold of the CFAR

4 EVALUATION

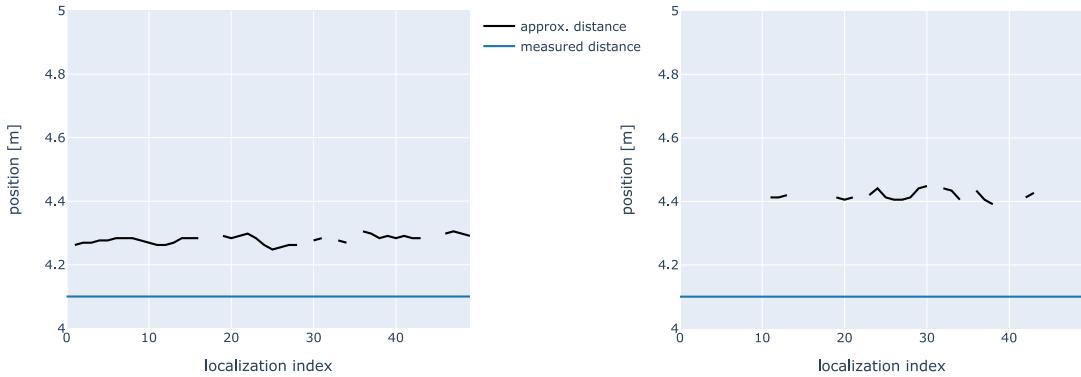


Figure 4.9: Evaluation of 8th degree code

code degree	SNR	false alarm rate	minimum threshold
10	high	9%	0.2
10	low	12%	0.4
9	high	12%	0.2
9	low	17%	0.4
8	high	26%	0.4
8	low	26%	0.4

Table 4.1: Used false alarm rate and minimum threshold in CFAR for different degrees and SNR's

algorithm must be adjusted to maintain successful peak detection in varying code degrees and SNR levels. As code degree and SNR decrease, the false alarm rate and minimum threshold must be increased to compensate for decreased system performance 4.1.

In conclusion, the results of these measures demonstrate the importance of maintaining a high SNR for accurate peak detection and position measurement. As the code degree decreases, the performance of the system worsens, highlighting the need for codes of higher degrees to achieve better results. Additionally, a small shift in position under 20 cm is observable on all results when the SNR is decreased, emphasizing the importance of a high SNR in maintaining the accuracy of the position measurements.

Conclusion

In this research project, the optimal orthogonal pseudorandom code, named gold codes, have been selected due to their satisfactory correlation properties and scalability. The signal processing, which includes modulation, filtering, and peak detection, has been successfully implemented in python. As a result, a 3D position localization algorithm has been developed that operates effectively in a simulated environment with only 4 anchors.

To sum up the evaluation, the results of the data in this research project have demonstrated the effectiveness of using orthogonal codes for acoustic underwater localization. The simulation results have shown a stable localization performance, with higher SNR levels resulting in improved stability in peak detection. The principle of sidelobe increase has been observed to have a significant impact on the localization performance at lower SNR levels. The field testing results have confirmed the stability of the localization performance, particularly for codes of 10th degree. The CA-FAR algorithm has been found to be an effective tool for peak detection, but its performance still needed improvements by adjusting its false alarm rate based on the SNR and code degree.

In terms of future research, there is room for improvement in the CFAR algorithm, such as the use of sorted data bins (OS-CFAR). Besides that, also alternative peak detection method could be tested in combination with CFAR.

Moreover, exploring other options for increasing the gold code length, such as a method for concatenating codes without negatively impacting their correlation properties, could be investigated in future studies. This would provide more opportunities for improving the localization performance and may achieve even higher noise and multipath propagation tolerances.

Additionally, the effectiveness of the simulated setup with four anchors for 3D position localization, such as with the BlueROV2 underwater vehicle, could be further evaluated in real-world scenarios to gain a better understanding of its practical performance and limitations.

5 CONCLUSION

Bibliography

- [BM02] Ralph Bucher and Durga Misra. A synthesizable vhdl model of the exact solution for three-dimensional hyperbolic positioning system. *VLSI Design*, 15, 2002.
- [Gen07] Ken Gentile. Digital pulse-shaping filter basics, 2007.
- [GK65] R. Gold and E. Kopitzke. Study of correlation properties of binary sequences. *Interim Tech. Rep.*, 1–4(1), 1965.
- [MD] Steven J. Merrifield and John C. Devlin. The implementation of a multiplexing gold codegenerator using a xilinx logic cell array. *School of Electronic Engineering, La Trobe University*.
- [PPWW72] William Wesley Peterson, Wesley Peterson, Edward J Weldon, and Edward J Weldon. *Error-correcting codes*. MIT press, 1972.
- [PS08] John G. Proakis and Masoud Salehi. *Digital Communications*. McGraw-Hill Higher Education, New York, 2008.
- [rad] Falschalarmrate - radar basics. <https://www.radartutorial.eu/01.basics/Falschalarmrate.de.html>. Accessed: 2022-11-08.
- [Roh11] Hermann Rohling. Ordered statistic cfar technique - an overview. In *2011 12th International Radar Symposium (IRS)*, pages 631–638, 2011.
- [SP80] Dilip V Sarwate and Michael B Pursley. Crosscorrelation properties of pseudorandom and related sequences. *Proceedings of the IEEE*, 68(5):593–619, 1980.
- [vWOJ] Paul van Walree, Roald Otnes, and Trond Jenserud. Watermark.
- [WXX11] Yang Weng, Wendong Xiao, and Lihua Xie. Total least squares method for robust source localization in sensor networks using tdoa measurements. *International Journal of Distributed Sensor Networks*, 7(1):172–902, 2011.

BIBLIOGRAPHY

Localization Formula

$$x_{ji} := x_j - x_i, \quad y_{ji} := y_j - y_i, \quad z_{ji} := z_j - z_i \quad (\text{A.1})$$

$$A = \frac{d_{02}x_{10} - d_{01}x_{20}}{d_{01}y_{20} - d_{02}y_{10}}, \quad B = \frac{d_{02}z_{10} - d_{01}z_{20}}{d_{01}y_{20} - d_{02}y_{10}} \quad (\text{A.2})$$

$$C = \frac{d_{02}(d_{01}^2 + x_0^2 - x_1^2 + y_0^2 - y_1^2 + z_0^2 - z_1^2) - d_{01}(d_{02}^2 + x_0^2 - x_2^2 + y_0^2 - y_2^2 + z_0^2 - z_2^2)}{2(d_{01}y_{20} - d_{02}y_{10})} \quad (\text{A.3})$$

$$D = \frac{d_{23}x_{12} - d_{21}x_{32}}{d_{21}y_{32} - d_{23}y_{12}}, \quad E = \frac{d_{23}z_{12} - d_{21}z_{32}}{d_{21}y_{32} - d_{23}y_{12}} \quad (\text{A.4})$$

$$F = \frac{d_{23}(d_{21}^2 + x_1^2 - x_2^2 + y_1^2 - y_2^2 + z_1^2 - z_2^2) - d_{21}(d_{23}^2 + x_1^2 - x_3^2 + y_1^2 - y_3^2 + z_1^2 - z_3^2)}{2(d_{21}y_{32} - d_{23}y_{12})} \quad (\text{A.5})$$

$$G = \frac{E - B}{A - D}, \quad H = \frac{F - C}{A - D}, \quad I = A \cdot G + B, \quad J = A \cdot H + C \quad (\text{A.6})$$

$$K = d_{02}^2 + x_0^2 - x_2^2 + y_0^2 - y_2^2 + z_0^2 - z_2^2 + 2x_{20}H + 2y_{20}J \quad (\text{A.7})$$

$$L = 2(x_{20}G + y_{20}I + z_{20}) \quad (\text{A.8})$$

$$M = 4d_{02}^2(G^2 + I^2 + 1) - L^2 \quad (\text{A.9})$$

$$N = 8d_{02}^2[G(x_0 - H) + I(y_0 - J) + z_0] + 2L \cdot K \quad (\text{A.10})$$

A LOCALIZATION FORMULA

$$O = 4d_{02}^2 \left[(x_0 - H)^2 + (y_0 - J)^2 + z_0^2 \right] - K^2 \quad (\text{A.11})$$

$$z_{a,b} = \frac{N}{2M} \pm \sqrt{\left(\frac{N}{2M}\right)^2 - \frac{O}{M}} \quad (\text{A.12})$$

$$z = \min \left\{ \max \left\{ z_a, z_b, z_{surface} \right\}, z_{ground} \right\} \quad (\text{A.13})$$

$$z = \begin{cases} z_a & \text{if } |z_a - z'| < |z_b - z'| \\ z_b & \text{else} \end{cases} \quad (\text{A.14})$$

$$\vec{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} Gz + J \\ Iz + H \\ z \end{bmatrix} \quad (\text{A.15})$$