NTNU
Norwegian University of
Science and Technology

Underwater Acoustic Positioning System for Real-time Fish Tracking

Jørn Iversen Efteland

Master of Science in Cybernetics and Robotics
Submission date: June 2016
Supervisor: Jo Arve Alfredsen, ITK

Norwegian University of Science and Technology
Department of Engineering Cybernetics

Preface

This thesis is the culmination of the work I have done during my last year as a student of engineering cybernetics. First laying the groundwork in my specialization project and then finishing the main goals as well as presenting my findings for this thesis.

I am grateful for being able to take this particular assignment, as it nicely combines my interests for fish and wildlife with my interests for technical solutions and embedded development.

I would like to thank Jo Arve Alfredsen for supervising this project. I would also like to thank Eivind Brandsæter Hvam as well as Erik Høy at Thelma Biotel for delivering hardware that made it possible to realize this system. Eivind modified the software of their product in order to suit my needs and has also been an invaluable source of technical support. The cooperation between Bjørøya Fiskeoppdrett and the research project CycLus made it possible to put the system to the test, and for that I am grateful to those whom it may concern.

Among my fellow students I want to thank Ove Nesse. He worked on a similar embedded project to mine, and our exchange of ideas for implementation has been valuable.

Lastly I would like to thank the people at the workshop of the Department of Engineering Cybernetics for building the water proof casing for the electronic equipment I developed.

Abstract

This master thesis describes an implementation of an underwater acoustic positioning system for real-time fish tracking in large-scale aquaculture sea cages, as well as experimental tests conducted with system developed. The positioning system is to be based on hydrophones of type TBR700-RT and Thelma Biotel transmitter tags. It is a passive acoustic localization system, which means that the transmitting node is to be located based on signal reception at the hydrophones. For localization, the method of time difference of arrival (TDOA) was used in combination with a pressure sensor.

First, the theoretical groundwork for the system is shown. That includes derivations of the positioning equations, algorithms, methods for software implmentation and considerations for the special case of underwater sound.

As the TDOA method presupposes accurate clock synchronization, a surface support module (SSM) was developed, utilizing GPS technology to synchronize the clocks of the hydrophones. This thesis will describe how to utilize the SSM to bridge the hydrophones to the internet, enabling fish-tracking in real time.

Lastly, experimental tests will be presented in order to evaluate the precision of the system. Some experiments that were done in November 2015 involved fish tracking, and was parsed by the developed algorithm. These results will also be discussed briefly.

The experimental tests showed the system to perform well. Within a sea cage where hydrophones were mounted in an equilateral triangle on the periphery, the system was precise within a margin of $\pm 2m$ for most transmitter positions. However, the deviations on the position estimates are dependent on the transmitter's relative geometry to the receiver array, and is therefore variable. Near the middle of the cage, the precision was about $\pm 1m$. It has thus been shown that the system is useful for monitoring fish motion in sea cages.

Sammendrag

Denne masteroppgaven beskriver implementasjonen av et undervanns akustisk posisjoneringssystem for sanntids-sporing av fisk i oppdrettsmerd, samt eksperimentelle forsøk gjort for å avgjøre systemets presisjon. Posisjoneringssystemet er basert på akustisk telemetri-teknologi levert av Thelma Biotel. Det er et passivt akustisk lokaliseringssystem, som betyr at en lydkilde skal bli posisjonert basert på forskjeller i mottakstidspunkt mellom ulike hydrofoner. Lydkilden er da en akustisk sender som opereres inn i fisken som skal spores.

Først vil det teoretiske grunnlaget for et slik system bli gjennomgått. Det inkluderer utledelse av posisjoneringsligninger, algoritmer og metoder for programvareimplementasjon av disse.

Ettersom tidsforskjell-basert posisjonering krever at mottakerene har synkroniserte klokker, ble en overflatemodul utviklet. Overflatemodulens formål var å kommunisere synkroniseringsmeldinger til hydrofonene ved bruk av GPS-teknologi. I denne oppgaven legges det også fram forslag til hvordan overflatemodulen kan fungere som en bro til internett, slik at kapabiliteter til sanntids-meldingsoverføring fra hydrofonene blir mulig.

Til slutt vil det bli presentert eksperimentelle forsøk. Dette inkluderer både data fra et fiskesporings-eksperiment gjort i 2015, samt en presisjonstest av nyere tid.

Forsøkene viste at posisjoneringssystemet yter bra. Ved å montere hydrofonene til ringen på en oppdrettsmerd, kan man oppnå en presisjon på $\pm 2m$ for de fleste posisjoner. Avvikene i posisjonsestimatene er avhengig av lydkildens relative posisjon i forhold til mottaker-konstellasjonen og er dermed variabel. Så mot midten av merden får man posisjonsestimater helt nede i $\pm 1m$. Det har dermed vist seg at systemet som har blitt uviklet kan være nyttig i sammenheng med overvåkning av bevegelsesmønstre til fisk i fangenskap.

Abbreviations and nomenclature

DPPM - Differential pulse position modulation
TDOA - Time difference of arrival
LOP - Hyperbolic line of position
DOP - Dilution of precision
SNR - Signal to noise ratio
CTD - Conductivity, temperature, salinity for measuring sound speed
GNSS - Global navigations satellite system
GPS - Global positioning system (The american GNSS constellation)
SSM - Surface support module
Timestamp[sic] - Reception time at a single node associated with a signal

Contents

Chapter 1

Introduction

1.1 Background

Currently the aquaculture industry is Norway's second largest export industry. It is however, also one of the youngest, with its modern roots in the seventies. As it is an industry that deals with the complexity of biological systems, it has to withstand the threats of diseases and parasites. One of the major concerns right now is the parasite known as the sea louse. (Lepeophtheirus salmonis) The parasite feeds on its salmonoid host's skin, mucus and blood and is proving to be a detriment to the industry. A common method for fighting the parasite has been using chemicals to treat the salmon. Methods like this are unfortunately susceptible to the evolutionary struggle of the parasite, and resistant lice has now been found in sea cages used for salmon farming. This forces the industry to try out new methods. One of those are the use of so-called cleaner fish. Species of fish that eats lice from the salmon hosts. The Lumpsucker (Cyclopterus lumpus) and the Ballan wrasse (Labrus bergylta) are both species that exhibits the property of eating sea lice off salmons in the wild.

Cleaner fish are now set out in sea cages together with the salmon in the hope that it will suppress the parasite problem. This is a new endeavour for the industry, and little is known about the behaviour of the cleaner fish in captivity. There is also a problem that the death rate among the cleaner fish used in sea cages are quite high. Therefore, the usage of cleaner fish should be subjected to rigorous research in order to determine it's efficiency and performance as a solution for the lice problem.

One step of this research lies in the development of technical equipment to monitor the animals, and gathering data that can be used to infer points about their behaviour. In this project, a technical solution for the use case of tracking fish motion in large-scale aquaculture sea cages, has been developed.

1.2 Previous work

This master thesis provides a follow-up to the work done in my specialization project at the fall of 2015. The purpose of that project was to create an underwater positioning system using already existing hydrophone technology and hydroacoustic transmitter tags developed by Thelma biotel. This positioning system can be used for tracking aquatic animals within a specified physical area covered by the hydrophones. The equipment and algorithms developed has also been used as part of the research project CycLus. CycLus is an interdisciplinary project researching the potency of the use of lumpsuckers as a means for fighting the sea lice problem currently troubling the Norwegian aquaculture industry.

During the specialization project the basic theoretical groundwork was laid for creating an underwater acoustic positioning system. Also, some hardware and software development was done in order to meet the requirements set by the theoretical groundwork. As the system is based on time difference of arrival (TDOA), a surface support module (SSM) dedicated to clock synchronization was made. This utilized GPS technology creating a successful distributed clock synchronization mechanism with an accuracy of at least $\pm 1ms$. The accuracy was probably significantly better, but the testability was constrained by the resolution of the signal processing on the hydrophones.

The system was tested during a month-long experiment as part of CycLus, gathering a large chunk of data. This included one week of tracking data from individual fishes. As the positioning algorithm was not yet finished for the project report, these tracking data will be briefly presented in this thesis.

1.3 Project outline

This thesis will be split in three distinct parts. One part is about the positioning system in itself, including suggestions for improving the system, drawing concepts from signal processing and marine acoustics. Afterwards, the incorporation of real-time monitoring into the system will be described. Lastly there will be results in form of experimental tests. These will be focused on the positioning algorithm and its performance.

1.4 Available equipment

This project is based on the usage of telemetry technology developed by the company Thelma Biotel.[1] Namely the Thelma Biotel Receiver 700 Real Time (TBR700-RT) hydrophone, and the Thelma Biotel transmitter tags. Introductory details to this equipment will now be discussed to provide context for the following chapters.

1.4.1 Thelma Biotel Transmitter tag

This is the tag that is to be implanted in the fish. It is an acoustic tag, that sends acoustic signals in the $69kHz$ frequency band using a piezoelectric transducer. Mounted in the tag there is also a pressure sensor. Signal transmission is done by sending a Differential Pulse Position Modulated (DPPM) signal. This means that the data packages are encoded in the time difference between the pulses. In the current version, the data from the pressure sensor is sent as an 8 bit data package. The tag also sends an identification number as part of this package.

1.4.2 Thelma Biotel Receiver 700

The Thelma Biotel Receiver 700 Real Time, hereby denoted TBR700-RT is the receiver equipment for the system. It is a hydrophone used for decoding the DPPM signals sent from the transmitter tags. The signal reception resolution is at $1ms$ because that is the rate at which it performs the fourier transform on the raw data. Therefore the signals will be timestamped with a $\pm 1ms$ uncertainty. The data package logged at the receiver at each reception is the tag ID, timestamp, signal to noise ratio, and the 8 bit data package from the tag. On the receiver, one can connect to a RS-485 communication line, as well as provide power supply to the submerged hydrophone. This is useful as one can communicate with the hydrophone from a surface module.

1.5 Development tools

During the course of this project, various development tools were utilized. The SSM software development were done using Atmel Studio and the C programming language. The GPS/Tinymesh expansion circuit board was designed using the freeware version of EAGLE. The positioning algorithm was implemented in the Python programming language. Matlab was also used for various tasks of data processing.

Chapter 2

Acoustic hyperbolic positioning

It follows from the available equipment that the method for positioning chosen here needs to be passive localization. This means that an acoustic transmitter is to be located based on time differences of signal reception at different positions in a hydrophone array. This method is called time difference of arrival, or TDOA for short. This chapter will describe an algorithm for utilizing time difference of arrival measurements to determine a transmitter's position. There will be a summary of the mathematical principles behind solving the positioning equations. Also there will be documented a method developed for finding unique positions when the receiver constellations yields ambiguous results.

2.1 Use case for this positioning application

The algorithm developed throughout this section is mainly focused on calculating position estimates within the constrained area of an aquaculture sea cage, thus providing some simplifications that can be used during the development. Later, the placement of the receivers will be discussed considering this particular use case. It should be noted there are specific challenges related to this particular case. For instance one has to consider ship traffic, and sound propagation through large bodies of biomass, thereby limiting the options for receiver placement.

2.2 Principles of TDoA-based hyperbolic positioning

For explaining the principles of hyperbolic positioning, we will consider the two-dimensional case. The length difference between two stationary points in a plane is what defines a mathematical function known as a hyperbola, where the two points are known as the foci points of that particular hyperbola. See Figure 2.1 for illustration.

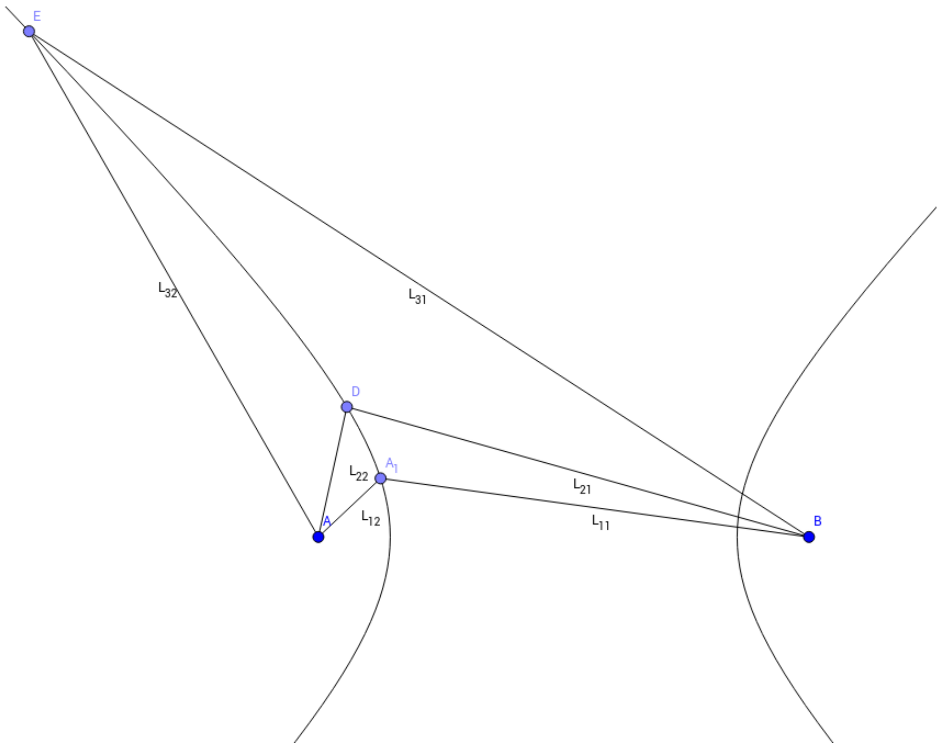


Figure 2.1: Hyperbola defined by a constant difference k and two foci, A at (A_x, A_y) and B at (B_x, B_y): $L_{i1} - L_{i2} = k$

From Figure 2.1 one can write out this in mathematical terms.

$$L_{i1} - L_{i2} = k \tag{2.1}$$

Where k is the difference by which the hyperbola is defined. L_{i1} is the length from the curve to point A, and L_{i2} is the length to point B. From the Pythagorean theorem it follows that:

$$L_{i1} = \sqrt{(x - A_x)^2 + (y - A_y)^2} \tag{2.2}$$

$$L_{i2} = \sqrt{(x - B_x)^2 + (y - B_y)^2} \tag{2.3}$$

Let's say that the two focal points in this case are some sort of sensor equipment. Either RF (radio frequency) antennas or acoustic transducers that are receiving signals from an electromagnetic or acoustic source, respectively. The difference in time between the received signals on those two foci multiplied with the speed of the transmitted signal (e.g. the speed of light or speed of sound) will be a distance in meters that defines a hyperbola on which the transmitter can be placed. With the speed of the transmitted signal c, and the time difference between node A and

B, t_{diff} it follows that $k = c * t_{diff}$. With all this in mind, the final equation for placing a signal on a hyperbola in a plane is:

$$\sqrt{(x - A_x)^2 + (y - A_y)^2} - \sqrt{(x - B_x)^2 + (y - B_y)^2} = c * t_{diff} \tag{2.4}$$

With one more hyperbola, defined by another set of focal points, one can determine two candidate positions for the transmitter. The hyperbolas will intersect at two different points in the plane. One last hyperbola is needed to narrow this down to one point.

In the end, what is measured is the ranges from the receivers to the transmitter. As the ranges are not measured directly, but by the combination of the signal speed, and the time it takes to reach its destinations, they are referred to as pseudo-ranges. The difference which defines a hyperbola are referred to as a pseudo-range difference. The fact that we are working with pseudo-ranges is important to take into consideration when evaluating acoustic signals, as the speed of sound in water can vary significantly by depth.

2.2.1 Hyperbolic positioning in a 3-dimensional space

The same principles as for 2-dimensional positioning still applies for 3-dimensional space. However, a TDOA pseudo-range difference in a 3-dimensional space will define a hyperboloid as opposed to a hyperbola. (See Figure 2.2) Two TDOA measurements will yield a curve on which the transmitter can be placed, whereas three will narrow this curve down to two points in space. A fifth receiver will be necessary to calculate the origin point of the signal. In other words, there will be need for one extra receiver node in order to calculate the point at which the transmitter is located, when considering a 3-dimensional space.

The equation for a hyperboloid defined by signal speed, c and time difference, t_{diff} is

$$\begin{aligned} &\sqrt{(x - A_x)^2 + (y - A_y)^2 - (z - A_z)^2} \dots \\ &- \sqrt{(x - B_x)^2 + (y - B_y)^2 - (z - B_z)^2} = c * t_{diff} \end{aligned} \tag{2.5}$$

Which is the same as Equation 2.4, but with the z-coordinate considered in the two pseudo-ranges.

2.2.2 Altitude measurement to limit the number of required receiver nodes

The depth of the transmitter to be located will define a plane in space. This can be used to limit the number of receivers needed. As the subject of this project is underwater localization, a pressure sensor will be an efficient means to define this plane. In water, the depth, h is linearly correlated to the pressure p by the formula $p = \rho * g * h$. The plane defined by the pressure at the site of the transmitter can replace one TDOA measurement hyperboloid, such that one less receiver is needed to determine the point of origin of the signal.

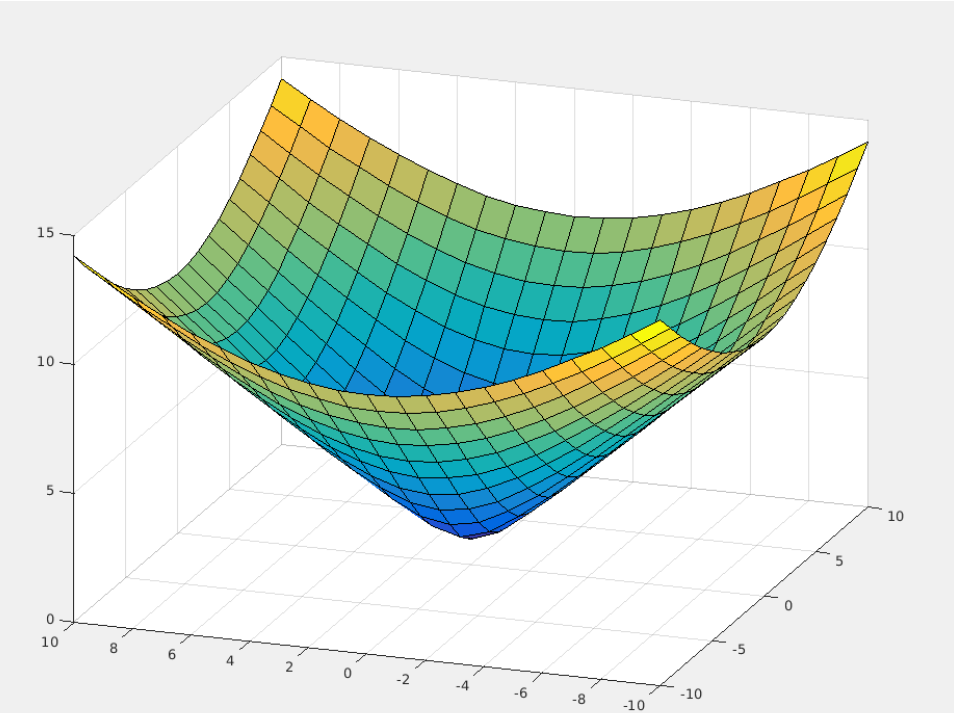


Figure 2.2: Upper half of a hyperboloid

2.3 Finding the time differences

Critical to the a pseudo-ranger application is the ability of the system to calculate the time differences of signal reception at different nodes. The faster the waveforms analysed propagates through space, the more precise it has to be. In sea water, the speed of sound, is roughly $1500m/s$ which means that a sound wave travels at about 1,5 meters every millisecond. For applications using electromagnetic (RF) waves, like the Global Positioning System this is even more critical as electromagnetic waves propagate at the speed of light. $(3 * 10^8 m/s)$

2.3.1 Cross-correlation

Cross-correlation is an efficient and accurate way to determine a time-delay between two signals. It is also the most commonly used method in RF systems. The cross-correlation function of two signals is the measure of similarity between the two signals as a function of the relative time-lag between them. In discrete terms, the cross-correlation product between two signals, f and g is described in Equation 2.6, where f^* is the complex conjugate of f.

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f^*[m]g[m+n] \tag{2.6}$$

As the value of the cross-correlation function determines the similarity between the two signals evaluated at a given time lag, the time lag between the two signals can be found at the peak of that function. This is exemplified in Figure 2.3.

2.3.2 Timestamping by first wavefront

Another way to determine the time delay is to simply find the point of origin of each pulse train, that is find the point in time at which signal originated at the receiver. I will refer to this as the time of the first wave front. In the environment of marine acoustics, this is actually one of the safer methods for guaranteeing accuracy or at least a consistent margin of error. The reason for this is that a signal may look different at different receiver nodes due to the nature of reflection patterns from the seabed and surface often causing multipath interference.

2.4 Solving the equations to get hyperbolic position fixes

It is now established that a hyperboloid can be defined as the difference in pseudo-ranges between two receivers and that this difference in pseudo-ranges is found using time differences. The next step is to find a solution for the equations given by the pseudo-range differences. This is the point at which the two hyperboloids intersects each other as well as a known z-plane defined by the output from a pressure sensor.

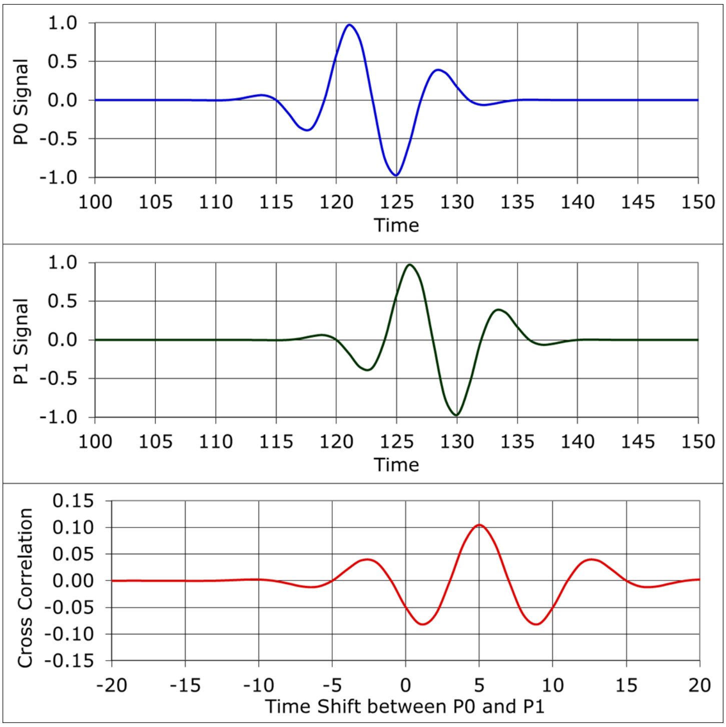


Figure 2.3: Two signals and the cross correlation product. (Source: Wikipedia, image by deleted user)[2]

The two hyperboloids will be defined by Equation 2.5. To create two non-dependent equations one would need at least three points in space, A, B and C. These two equations is to be solved for x and y, that is the points at which the hyperboloids intersect each other and the known z plane.

Let's say that receiver A is located a point (0,0) in a plane. Receiver B is located at $(b_x, 0)$ and receiver C at (c_x, c_y). Additionally it is assumed that all receivers are in the same z plane. The time difference between two stations, A and B is described by: $T_{ab} = T_a - T_b$, where T_a and T_b denotes arrival time of the signal at a speed c to be localized. As demonstrated by Bertrand T. Fang[3], the solution for a set of Equation 2.5 can be obtained by

$$y = g * x + h \tag{2.7}$$

$$d * x^2 + e * x + (f - z^2) = 0 \tag{2.8}$$

where

$$R_{ab} = c * T_{ab} \tag{2.9a}$$

$$R_{ac} = c * T_{ac} \tag{2.9b}$$

$$g = (R_{ac} * (b/R_{ab}) - c_x)/c_y \tag{2.9c}$$

$$h = (c^2 - R_{ac}^2 + R_{ac} * R_{ab}(1 - (b/R_{ab})^2)/(2 * c_y) \tag{2.9d}$$

$$d = -(1 - (b/R_{ab})^2 + g^2) \tag{2.9e}$$

$$e = b * (1 - (b/R_{ab})^2) - 2 * g * h \tag{2.9f}$$

$$f = (R_{ab}^2/4) * (1 - (b/R_{ab})^2)^2 - h^2 \tag{2.9g}$$

The solutions of these polynomial equations will output two candidate points at which the transmitter can be placed. This ambiguity can be resolved either by introducing redundancy to the system or a geometrical argument can be constructed to discard one of the candidate points. There is also some cases which these equations does not cover because of a zero division. That is the cases where $T_{ab} = 0$ or $T_{ac} = 0$. The solutions of these cases are of course quite easy to find as they are just the line that is in the middle of the two points on which the transmitter produced that time difference.

This means that if $T_{ab} = 0$ you get

$$x = B_x/2 \tag{2.10}$$

and if $T_{ac} = 0$

$$y = -\frac{c_x}{c_y} * x + \frac{c_x^2 + c_y^2}{2 * c_y} \tag{2.11}$$

Inserting these into Equations 2.7-2.8 yields results for the different cases while avoiding zero-divisions. There are three special cases that needs to be solved when implementing the positioning equations in software. Those are when $R_{ab} = 0$, $R_{ac} = 0$ and $R_{ab} = R_{ac} = 0$. By simple algebra it can be shown that when $R_{ab} = 0$ the candidate positions can be solved by Equation set 2.12.

$$x = B_x/2 \tag{2.12a}$$

$$i = R_{ac}^2 - c^2 + 2 * C_x * x \tag{2.12b}$$

$$(4 * Cy^2 - 4 * R_{ac}^2) * y^2 + 4 * C_y * i * y + (i^2 - (4 * R_{ac}^2) * (z^2 + x^2)) * y = 0 \tag{2.12c}$$

Similarly when $R_{ac} = 0$, assuming x_1 and x_2 are the solutions of the polynomial, one gets

$$k = (C_x^2 + C_y^2)/(2 * C_y) \tag{2.13a}$$

$$l = -2 * (C_x/C_y) * k \tag{2.13b}$$

$$j = R_{ab}^2 - B_x^2 \tag{2.13c}$$

$$\begin{aligned}&(4 * B_x^2 - 4 * (R_{ab}^2) * (1 + (C_x/C_y)^2)) * x^2 \dots \\ &+ (4 * B_x * j - 4 * (R_{ab}^2) * l) * x + j^2 - (4 * R_{ab}^2) * (z^2 + k^2) = 0\end{aligned} \tag{2.13d}$$

$$y_1 = -(C_x/C_y) * x_1 + (C_x^2 + C_y^2)/(2 * C_y) \tag{2.13e}$$

$$y_2 = -(C_x/C_y) * x_2 + (C_x^2 + C_y^2)/(2 * C_y) \tag{2.13f}$$

The case of $R_{ab} = R_{ac} = 0$ yields the trivial solution where the transmitter can be placed in equal distance from all receivers. The position coordinates for that case is found by solving Equations 2.10-2.11.

2.5 Sound speed

In seawater the sound speed varies with different physical parameters. Those are salinity, depth and temperature. Del Grosso's formula [4] is sufficient to illustrate the relationship between the sound speed and those parameters

$$c = 1448.6 + 4.618 * T - 0.0523 * T^2 + 1.25 * (S - 35) + 0.017 * D \tag{2.14}$$

Both salinity and temperature will potentially vary significantly by depth as well, and may cause a high sound speed gradient. This needs to be considered when calculating the pseudo-range difference at which the transmitter is to be located.

Measuring the sound speed can be done by using a CTD probe. This measures the salinity, temperature and depth in the medium.

2.5.1 Ray acoustics and reflections

As the speed of sound in water varies by depth, the paths at which the sound waves propagate will be subjected to bending effects. This means that the sound waves will not take a direct path straight to the receiver, and the bending effects might manifest itself differently at different receivers. Calculating the shortest path lengths at all times from a transmitter to all possible receivers is simply not feasible as it would require a lot of sensor equipment. The problem with multipath transmission is especially a problem for high frequency signals.

Another thing to consider are reflections from the seabed. The same signal might be received multiple times with significant time differences on the receiving node. As the transmitter uses Differential Pulse Position Modulation (DPPM) there is a possibility that the pulse train will collide/interfere with itself. For long range transmissions this is a large problem due to the nature of multipath signals.

2.5.2 Accounting for varying sound speed in the positioning equations

The mathematics behind ray acoustics are complicated, and the sensor equipment needed to monitor the behaviour of the rays needs to be extensive. It follows that considering ray acoustics for the positioning algorithm is not really feasible. Nor is it useful as the bending effects are relatively insignificant over short distances. Instead one can assume that the sound waves propagate in a straight line, spherically from the source. If this is the case, the mathematics to determine the sound speed for the pseudo-ranging algorithm becomes quite easy. As the sound waves will spend the same amount of time in every layer of the sea, one just has to use the average sound speed between the origin of the signal and the receiver. Since the transmitter depth is known from the data it sends, this can be utilized to calculate the average sound speed at which the signal propagates. If z_t is the depth of the transmitter, z_r is the depth of the receiver, and the CTD profile in the required range is known, one can find the sound speed to input into the hyperbolic positioning equations, c_{avg} by

$$c_{avg} = \frac{1}{z_t - z_r} * \sum_{n=z_t}^{z_r} c[n] \tag{2.15}$$

One important closing remark is that this approximation will perform really well if the seawater is mixed. That is when the sound speed-gradient is low or constant. Higher sound speed gradients will lead to more significant bending effects. It is also something one should be weary about when trying to design a long range acoustic positioning system.

2.6 Transformation of reference frames

The solutions to Equation 2.5 as demonstrated by Fang presupposes three receiving nodes where two has to be fixed on the same axis. Let's call that axis the baseline of the system. One of the receiving nodes, A, is placed at the origin of the coordinate frame in which positioning is to be performed. Another one is to be placed at the x-axis, this is point B in the equations at the point $(B_x, 0)$ in the xy-plane. The last node is unconstrained in the algorithm, at point (C_x, C_y).

Suppose a system of more than three nodes, where each different set of three nodes can generate a position estimate. Obviously the conditions for Equations 2.7-2.8 can't be satisfied for each triangle, and it is impractical and limiting to create a new coordinate system for each triangle. A solution to this is to transform the triangles into a reference frame that satisfies the conditions for solving the positioning equations, then transform the position estimates back to the original coordinate system. This is performed by first using a translational, and then a rotational transform of the coordinates for the receiver nodes, and then doing the reverse process on the positions generated.

The translational transform is just the trivial task of moving the triangle such that one of the receiver nodes is placed at the origin of a reference frame. In other words, one has to chose one node position as the origin for the new frame. Let's consider a general case with a point A at (A_x, A_y), B at (B_x, B_y) and C at (C_x, C_y), where A is supposed to be the origin for the positioning equations. The translated points will then be

$$A : \begin{bmatrix} A'_x \\ B'_y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{2.16a}$$

$$B : \begin{bmatrix} C'_x \\ C'_y \end{bmatrix} = \begin{bmatrix} B_x - A_x \\ B_y - A_y \end{bmatrix} \tag{2.16b}$$

$$C : \begin{bmatrix} C'_x \\ C'_y \end{bmatrix} = \begin{bmatrix} C_x - A_x \\ C_y - A_y \end{bmatrix} \tag{2.16c}$$

The step that remains is to rotate the triangle such that point B is aligned along the x axis. A counter-clockwise rotation of angle θ around the origin of an arbitrary point, (x, y) is obtained by

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} * \begin{bmatrix} x \\ y \end{bmatrix} \tag{2.17}$$

The angle at which to rotate the triangle can be found using trigonometric identities. Using the tangent function of an angle, which is the ratio of the length between the opposite side to the length of the adjacent side, using the point found in Equation 2.16b one gets

$$tan\theta = B'_y / B'_x \tag{2.18}$$

Which is the angle at which the line from the origin to point B is aligned counter-clockwise compared to the baseline. This needs to be clockwise rotated such that point B will be at some point $(B_x, 0)$. A clockwise rotation is the same as was shown in Equation 2.17 but with a negative angle which means that the rotation angle for the transformation, θ_t can be written

$$\theta_t = -arctan\frac{B'_y}{B'_x} \tag{2.19}$$

and the transformed points A_t, B_t and C_t are then

$$A_t : \begin{bmatrix} A'_{tx} \\ A'_{ty} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{2.20a}$$

$$B_t : \begin{bmatrix} B'_{tx} \\ B'_{ty} \end{bmatrix} = \begin{bmatrix} cos\theta_t & -sin\theta_t \\ sin\theta_t & cos\theta_t \end{bmatrix} * \begin{bmatrix} B'_x \\ B'_y \end{bmatrix} \tag{2.20b}$$

$$C_t : \begin{bmatrix} C'_{tx} \\ C'_{ty} \end{bmatrix} = \begin{bmatrix} cos\theta_t & -sin\theta_t \\ sin\theta_t & cos\theta_t \end{bmatrix} * \begin{bmatrix} C'_x \\ C'_y \end{bmatrix} \tag{2.20c}$$

Applying the theory from Section 2.4 will then yield a set of candidate positions within the new reference frame. A candidate point from the new reference frame,

T' at (T'_x, T'_y) can then be transformed to the old frame where T at (T_x, T_y) is the the actual position in the original frame. This transformation can be shown as

$$T : \begin{bmatrix} T_x \\ T_y \end{bmatrix} = \begin{bmatrix} cos(-\theta_t) & -sin(-\theta_t) \\ sin(-\theta_t) & cos(-\theta_t) \end{bmatrix} * \begin{bmatrix} T'_x \\ T'_y \end{bmatrix} + \begin{bmatrix} A_x \\ A_y \end{bmatrix} \tag{2.21}$$

Using these transformations one can can calculate position estimates from multiple triangles within the same coordinate system. This is the first step towards providing redundancy in a positioning system. It can also be used to resolve ambiguity by the means of cluster identification.

2.7 Resolving ambiguity from the positioning equations by geometry

Following will be a geometrical method for resolving the aforementioned ambiguities produced by Fang's positioning equations. For simplicity it is here assumed that all receiver nodes are in the same plane, and therefore the 2-dimensional argument holds also for 3-dimensional positioning. Suppose two receiver nodes, A and B receiving signals from the target at times T_a and T_b. By the use of a third receiver node, two candidate positions T_1 and T_2 are generated by Fang's equations. Consider Figure 2.4. It can be easily seen that in this case that the candidate T_2 can be discarded if $Tb < Ta$. Similarly $T_a < T_b$ leads to T_1 being discarded. This is assuming that the sound speed is equal regardless of azimuth angle from the transmitter, which is a common assumption in underwater acoustics.

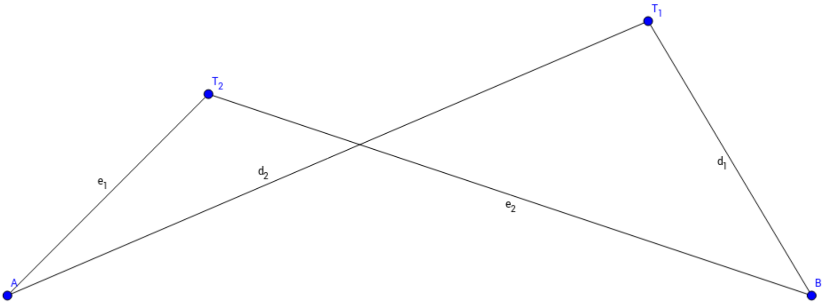


Figure 2.4: Showing two candidate positions, T_1 and T_2, as well as two receiver nodes A and B

Using the notation introduced in Figure 2.4 this can be generalized, where d_1 is the distance from T_1 to B, and d_2 is the distance from T_1 to a. For the other candidate position, T_2, we define e_1 as the distance to A and e_2 as the distance to B. For being able to resolve the ambiguity geometrically we need to know that the two candidates will have their first reception at opposite nodes. In other words, if the potential signal from T_1 arrives first at B, then the potential signal from T_2 should arrive at A first. If not, it is not possible to determine which of the candidate positions are the origin of the signal. Formally the conditions and subsequent results can be written as

```
if $d_1 < d_2$ and $e_1 < e_2$ then
    if $T_a < T_b$ then
        choose $T_2$
    else
        choose $T_1$
else if $d_2 < d_1$ and $e_2 < e_1$ then
    if $T_a < T_b$ then
        choose $T_1$
    else
        choose $T_2$
```

Algorithm 1: Resolving ambiguity from Fang's equations by geometrical interpretation

2.8 Redundancy and cluster identification

There are however a few situations where the ambiguity from Fang's equations can not be resolved using the method described in Section 2.7. Assuming that there are more than three receiver nodes in the system, the ambiguities can be resolved by redundant calculations using different triangles.

Before proposing a solution to this problem, the use-case for the system has to be addressed. As the plan is to utilize the system in an aquaculture sea cage, there are restrictions both to the movement of the fish (potential positions) and the placement of receivers. The optimal triangle for positioning, when constrained to the cage, is an equilateral triangle mounted on the periphery, hereby referred to as the outer triangle. The reason for this triangle's optimality will be explained in Section 2.9. However the knowledge that one triangle performs better than the others for all positions enables one important simplification, which is that the redundant calculations performed on other triangles are done solely to choose between candidates on the outer triangle.

As each other triangle also produces two candidate positions, there will be a significant amount of data points to consider when evaluating which of the candidate points on the outer triangle that is the most likely position of the transmitter. It is to be expected that there will form a cluster around one of the candidate positions, where each calculated triangle contributes to this cluster.

Assuming N inner triangles where calculations are performed, there should be a cluster of N points around the correct candidate position excluding the candidate position itself. This can be solved by assuming that the average distance between the best candidate and the N nearest data is shorter than for the candidate position to be discarded. A pseudo-code implementation of this method can be seen in

Algorithm 2. Due to the importance of relative geometry between the transmitter and the receiver triangle, this method should not be used when there is a candidate position far outside the outer receiver triangle. This is due to the non-linearity of the hyperbolic positioning equations which will significantly increase the distance between the points in the cluster.

```
Calculate candidates $C_{01}$ and $C_{02}$ from outer triangle;
Initialize $C_{1dist} = 0$ and $C_{2dist} = 0$;
foreach Inner triangle do
    Calculate triangle candidates, $C_{i1}$ and $C_{i2}$;
    $d$=shortest distance of $C_{01}$ to one of the inner triangle candidates
    $e$=shortest distance of $C_{02}$ to one of the inner triangle candidates
    $C_{1dist} = C_{1dist} + d$;
    $C_{2dist} = C_{2dist} + e$;
end
if $C_{1dist} < C_{2dist}$ then
    choose $C_{01}$
else
    choose $C_{02}$
end
```

Algorithm 2: Choosing candidate points based on cluster identification

2.9 Dilution of precision

The solutions to the hyperbolic equations are non-linear in the sense that a given uncertainty in time will not produce a given uncertainty in position. This non-linearity depends on the transmitter's relative position to the receivers. In GNSS literature this is referred to as geometric dilution of precision or GDOP for short. The concept of DOP was introduced so one could describe the relationship between uncertainties in the measured data and the consequent uncertainty in the output location as a function of the target's relative position to the receiver array. Formally this can be written as in Equation 2.22.[5]

$$GDOP = \frac{\Delta(Output\ location)}{\Delta(Measured\ data)} \tag{2.22}$$

2.9.1 Background theory for DOP simulation

DOP values can be expressed as a variety of separate measurements, namely HDOP, VDOP, PDOP and TDOP, which are horizontal, vertical, position (3D) and time dilution of precision. As the application developed in this thesis uses a pressure sensor for depth measurements, the VDOP can be omitted. The pressure sensor is of course linear in it's performance for all practical purposes. Since we are interested in the precision of the positioning system, and an effective placement of the receiver triangle, we can look at the positional dilution of precision which can be expressed in terms of HDOP and VDOP. Because VDOP is a meaningless parameter for our application, the precision can be expressed in terms of HDOP. Using derivations from "The Underwater GPS Problem" (Taraldsen, Reinen, Berg) [6] and "Lowest GDOP in 2D scenarios" (N.Levanon) [7], the standard deviation σ of the position estimate is given by

$$\sigma = k_D * \sigma_R \tag{2.23}$$

Where the DOP factor, k_D, is given by the square root of the trace of the error covariance matrix of the two-dimensional system. That is $\sqrt{\sigma_x^2 + \sigma_y^2}$

$$k_D = \sqrt{trace[(M^T * M)^{-1}]} \tag{2.24}$$

The pseudo-ranges are given as

$$PR_i = R_i + \rho \quad i = 1, ..., N \tag{2.25}$$

Where R_i is the measured range $(c * t_{diff})$ and ρ is an arbitrary range offset common to all measurements. The matrix M assuming three measurements is then given by

$$M = \begin{bmatrix} \frac{\partial PR_1}{\partial x} & \frac{\partial PR_1}{\partial y} & \frac{\partial PR_1}{\partial \rho} \\ \frac{\partial PR_2}{\partial x} & \frac{\partial PR_2}{\partial y} & \frac{\partial PR_2}{\partial \rho} \\ \frac{\partial PR_3}{\partial x} & \frac{\partial PR_3}{\partial y} & \frac{\partial PR_3}{\partial \rho} \end{bmatrix} \tag{2.26}$$

and

$$\frac{\partial PR_i}{\partial x} = \frac{x - x_i}{R_i}; \ \frac{\partial PR_i}{\partial y} = \frac{y - y_i}{R_i}; \ \frac{\partial PR_i}{\partial \rho} = 1 \tag{2.27}$$

Where (x, y) are the target/transmitter position and (x_i, y_i) are the receiver positions.

Using this theory one can simulate HDOP for different receiver constellations. Matlab code for this is provided in Appendix B.

2.9.2 Simulated DOP results

The simulated results shows the DOP factor k_D in a contour plot (Figure 2.5-2.6) where the green circle represents the sea cage, and the red dots are receiving nodes. All experiments described in this thesis have been conducted using this set up, that is an equilateral triangle with one receiving node in the middle of it. By running the simulation, and pushing around on the receiver nodes it is quickly apparent that most configurations are much worse. This is also demonstrated mathematically in Levanon's article. [7]. It should also be noted that the HDOP values plotted, are those in the same plane as the receiver triangle

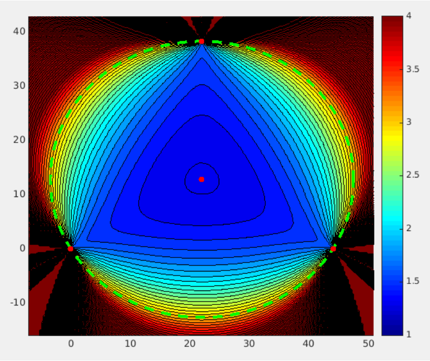


Figure 2.5: HDOP values for an equilateral receiver triangle placed on the periphery of a sea cage

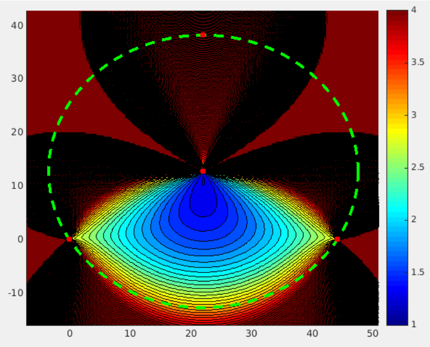


Figure 2.6: HDOP values for an inner triangle, with two receivers at the periphery and one in the middle of the sea cage

2.10 Resolution of the TDoA measurements and potential position fixes

To determine the resolution of a hyperbolic positioning system, one has to consider two parameters; the speed of the signal used for positioning, and the time resolution of the receiver equipment. For this project the signal speed is the speed of sound in water which is roughly $c_w = 1500m/s$ but of course critically reliant on different physical parameters in the sea at the site of measurement. The time resolution of the TBR700-RT receivers is $t_{res} = 1ms$.

The resolution of the positioning system can be found by plotting all the hyperboloids that are possible to generate within a constrained area given a range of time differences from different pairs of receivers. The resolution of the depth data will also have implications, but that could be tuned to diminish in magnitude with a different specification for data transmission.

To create a full resolution grid one has to slice the hyperboloids at a given z-plane.

In Figure 2.7 one can see a full grid on which the transmitter can be placed given certain parameters. The points of intersections are potential position fixes given by the resolution of a receiver and the speed at which the signal propagates.

From Figure 2.7 one can easily see that a given uncertainty in the TDOA measurements will manifest itself as a larger factor in the horizontal positioning error when the depth difference between the transmitter and the receiver array gets higher. From the horizontal slice, one can see that the position fixes in the middle of the receiver array triangle will be the best ones. The precision will decrease as the transmitter moves away from the center, and outside of the triangle one can get really high deviations from a time difference deviation of just $1ms$. It is fairly easy to see the similarity between the simulated DOP values, and the resolution grid. Also, the vertical slice of the resolution grid explains how the DOP values

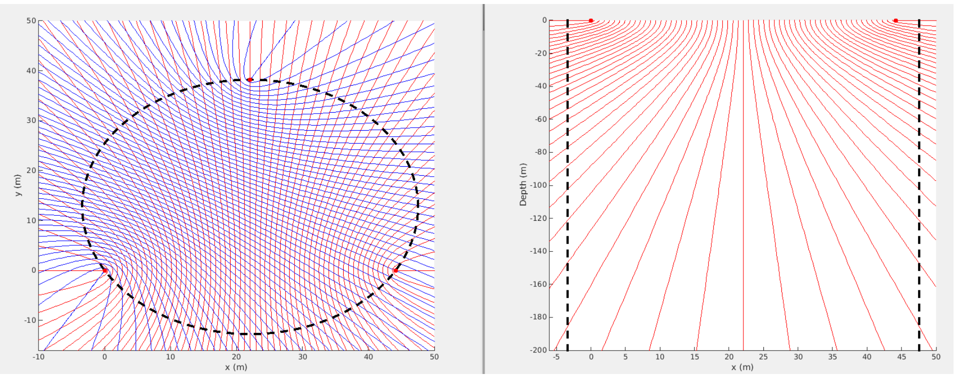


Figure 2.7: Grid on which the transmitter can be placed given $t_{res} = 1ms$ and $c_w = 1490m/s$, Horizontal position plot sliced at $z = 0$, depth plot sliced at $y = 0$, the circle in horizontal position plot and the lines in the depth plot represents an aquaculture sea cage of radius, $r = 25.48m$. The receivers are shown as red dots.

will deteriorate with depth.

Chapter 3

System overview

This chapter will provide an overview of hardware and software implementations for the purpose of doing real-time fish tracking with the available equipment described in Chapter 1.

3.1 Surface Support Module

One important part of the proposed system, is the surface support module hereby denoted as the SSM. Originally this was intended as a solution to the clock synchronization problem. However it has since been expanded to accommodate for other uses such as real-time tracking.

3.1.1 Clock synchronization

As shown in Section 2.2 all the receiving nodes in the system needs to have synchronized clocks. In this case it means that the submerged hydrophones needs to agree on which time it is. The reason that clock synchronization poses a problem is that the crystal oscillators usually implemented for digital clocks are imperfect. They tend to drift slightly such that millisecond precision won't be available for more than a limited time frame after they have been synchronized. That is why the GPS satellites for instance uses atomic clocks. Atomic clocks are unfortunately expensive, and not a valid solution for synchronizing these hydrophones.

There are different ways to synchronize clocks. One is to use acoustic signals sent between hydrophones in known positions. Another way which is highly effective is the use of GPS technology. The GPS system provides time synchronization down to $\pm 60ns$. This is well within the reception time resolution of the hydrophones used in this project, $1ms$, and therefore an efficient means to provide accurate time synchronization.

An advantage of using an SSM with GPS for timekeeping is that the system becomes distributed, which means that each node is decoupled and independent from the other nodes.

3.1.2 Network integration

Besides the fact that the system becomes distributed, there is a another advantage with choosing a solution involving an SSM. The SSM can also be used as a bridge between the hydrophones and the internet, thus providing the possibility for real-time data harvesting. A multi-hop mesh network can utilize all SSMs in order to reach it's destination, and is therefore well suited for this kind of application.

3.2 Server

The server is the backbone of the real-time fish tracking system. All data decoded at the hydrophones are forwarded to the server. It will then organize this data into log files. Using the theory developed in Chapter 2 it will perform positioning on the received data, and write this to seperate log files. It also has a webserver which ensures easy connection for anyone who wish to look at the log files generated. The general proposed system for real-time fish tracking can be seen in Figure 3.1.

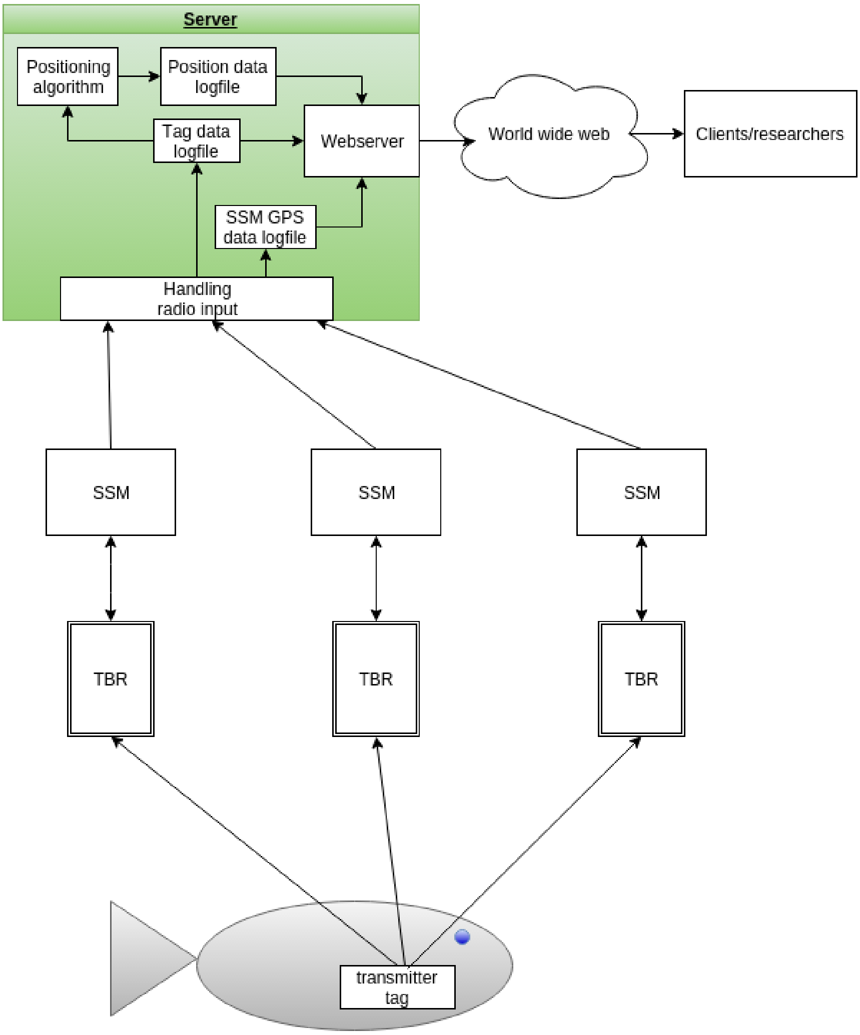


Figure 3.1: Overview of the proposed system for real-time fish tracking

Chapter 4

Surface support module implementation

As seen in Chapter 3, a surface support module (SSM) is not only vital to the design for real time tracking, but also advantageous as a means for clock synchronization. Clock synchronization is as explained in the aforementioned chapter a necessity for doing TDOA-based positioning. This chapter concerns the implementation of the SSM, and the incorporation of a network module in order to bridge the hydrophones to an internet server.

As the GPS-based time synchronization software for the SSM was built for my project report it will not be discussed in great detail. But the synchronization software will run together with the network software on the microcontroller. So for the cohesiveness of this report, as well as context for further discussion, the basic principles of the time synchronization mechanism will be described.

The software implementation for the network is yet to be developed, and will thus be described only conceptionally. That means that specific details surrounding its implementation will be omitted.

4.1 System overview: individual nodes

One node in the context of this project will be defined as a hydrophone, TBR700-RT with its corresponding surface support module (SSM), including the communication line between them. In Figure 4.1 the components that comprise a single node can be seen.

4.1.1 Transmission line: RS-485

The transmission line between the SSM and the hydrophone is an RS-485 line. RS-485 is a standard for multipoint data transmission that can be used to transmit

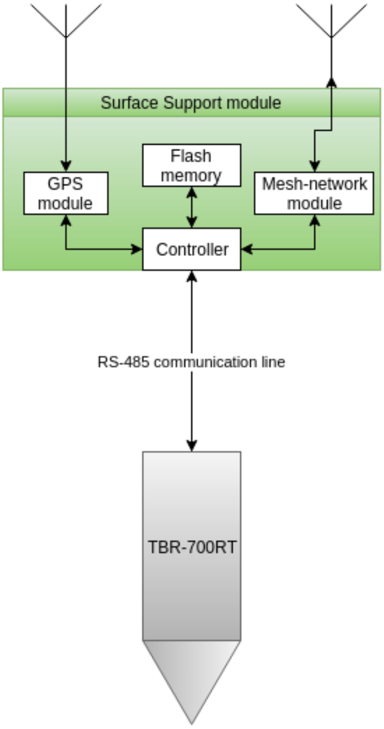


Figure 4.1: A single node in the positioning system

data over relatively long distances. RS-485 defines electrical standards, but not communication protocols. This means that the communication protocol needs to be specified by the system developer, including safety precautions for error-free transmission.[8]

4.1.2 Hydrophone: Basic features

As mentioned in the introduction, the hydrophones used are the TBR700-RT. The hydrophones receives the acoustic signals from the transmitter tags, and decodes them. It then timestamps the data, provides data on the SNR of the signal and writes this information to its memory. It also sends freshly decoded signals automatically through the RS-485 line.

4.1.3 SSM: Basic features

The most critical task of the surface support module is to send synchronization messages through the RS-485 line to the hydrophone. It uses GPS for timekeeping. The SSM should also forward data from the hydrophone to the server through radio communication. Whenever it receives a tag detection message from the hydrophone, this message should be written to its flash memory. It will write certain GPS data upon tag detections, keeping track of the quality of the signal, as well as the SSM position data at the time of reception.

4.2 SSM requirements

Following there will be a list of requirements for the SSM. The hard requirements for the SSM in order to enable the possibility of a real-time fish tracking system are:

- Accurate internal clock, synchronized by GPS every second. $\pm 0.5ms$ precision in clock synchronization. (better than half of the hydrophone resolution)
- RS-485 interface for communication with the hydrophone
- Network module for forwarding data received from the hydrophone.
- Internal flash memory to keep data in storage, such that the network module can sleep for most of the time, and only unload data to the server at a set periodicity.

Thelma Biotel provided me with the development boards used for the hydrophones. These boards already had the required RS-485 driver and microcontroller. In addition it had a flash memory chip, micro usb interface and some other interesting features that might be used at a later stage. The board delivered by Thelma Biotel will hereby be denoted the TBR board.

This board had to be interfaced with a GPS module as well as a network module, which required the creation of an expansion board. There were possibilities to connect this expansion board to both SPI/UART for communication and an interrupt pin on the microcontroller.

4.3 Introduction to tinymesh

The network solution chosen for this application is tinymesh.

Tinymesh is a self-healing, self-forming and self-optimizing mesh network[9], providing multi-hop message transmission between nodes. It is quite easily installed with embedded systems. The only thing that needs doing is to connect the radio to the UART interface of the microcontrollers you are using, and it is ready for use.

A tinymesh network comprises of three different types of nodes

- Gateway: This is the endpoint for transmission. In this application, the gateway will be connected to the internet server.
- End devices: these are devices on the outskirts of the mesh network. They are not active in forwarding messages from other nodes, but does transmit messages to the network.

- Routers: These nodes are the backbone of the mesh network as these forward messages through the network in an efficient way to the gateway.

A simple illustration of how these nodes are connected can be seen in Figure 4.2.

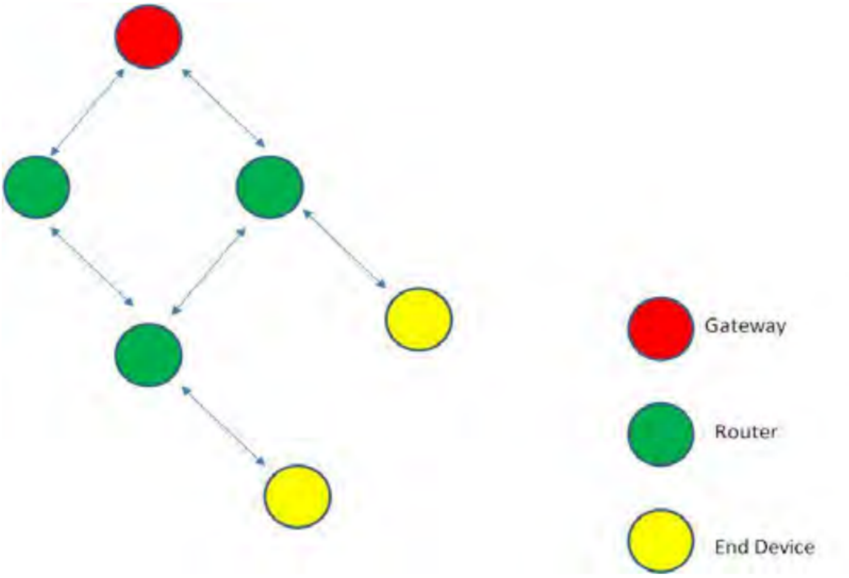


Figure 4.2: Connectivity between nodes in a tinymesh network. (taken from the Radiocrafts data sheet, page 6[10])

Details around the technical specifications can be found in the data sheet.[10] But some important features should be noted. By default the radio uses the UHF band, between $865 - 867mHz$ which means that it is configurable to license free bands in most countries, including Norway. Output power is at $27dBm$, translating to up to $800m$ of transmission distance. Data rate is at $76.8kbit/s$, but throughput is also related to the structuring of the network and should be determined by experimental tests.

4.4 SSM hardware

Following is a brief overview of the most important hardware utilized in the system. An overview of how the SSM hardware is connected can be seen in Figure 4.3. Here the PPS signal is connected to the interrupt pin on the microcontroller, as the PPS signal is what guarantees the accuracy of the time synchronization. Full schematics for how the expansion board was designed can be found in Appendix A.

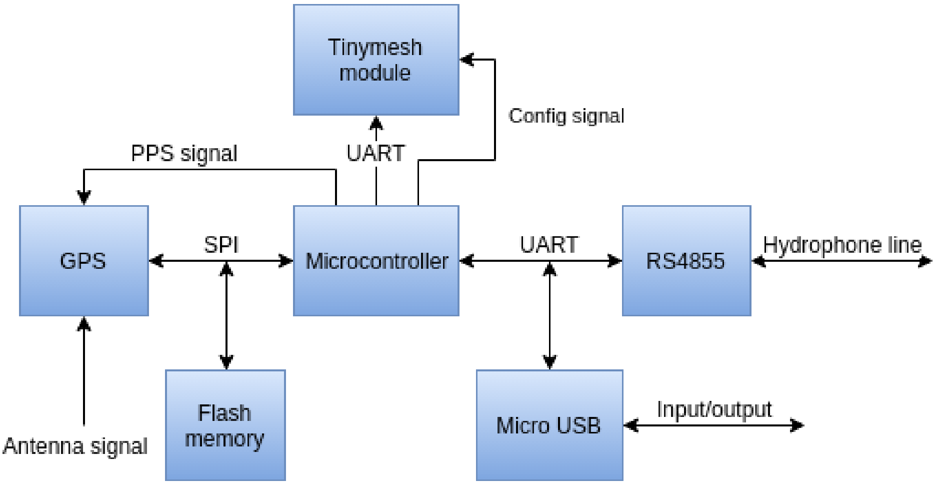


Figure 4.3: Coupling of the SSM hardware

Microcontroller

The microcontroller used was a 32 bits Atmel AVR controller.[11] For communication it provides both UART and SPI interfaces. It has interrupt pins that can be used for time critical applications and the processing power needed in order to implement the SSM.

GPS module

The GPS module chosen was the u-blox NEO-7P GNSS module. This particular GPS module provides a pulse per second (PPS) signal with an accuracy of $\pm 60ns$ according to its datasheet.[12] Additionally it has a SPI interface that can be accessed by the microcontroller. It has a lot of interesting features that might be useful for further development, e.g. Precise Point Positioning, which can provide position fixes down to about $1m$ accuracy. An active antenna, GPS-P1MAM was chosen for this application, providing a low fix acquisition time.

4.5 Time synchronization software

The time synchronization software can be described as one independent module. This module is interrupt driven and listens to the pulse per second (PPS) signal from the GPS and communicates synchronization to the hydrophone. I will not go into detail about how exactly the protocols for SPI communication to the GPS module is done, I will however refer to the name of the sentences polled. Details can be found in the ublox protocol specification.[13]

4.5.1 Time synchronization protocol

For clarity, first let's define what the format of the timestamp is. The timestamp used is Unix time. That is the number of seconds since 1 january 1970. The timestamp is a 32 bits signed variable.[14]

The protocol for sending synchronization messages was proposed by Eivind Brandsæter Hvam at Thelma Biotel. He wrote code for the hydrophone so that the hydrophone could utilize these messages. The synchronization messages are to be sent at a whole 10 second. That is when $timestamp\%10$ is equal to 0.(% is here the modulo division) There are two types of synchronization messages. The basic message is simply a request for the hydrophone to round up or down its timestamp to the nearest ten as well as resetting the millisecond value of its clock. The second message also sends that request, but in addition it sends a full timestamp and a checksum to set the clock of the hydrophone. A full overview of the synchronization related messages sent over the RS-485 communication line is provided in Table 4.1.

Sent from	Message	Description	Stream length
SSM	(+)	Basic synchronization message, round hydrophone to nearest 10. Reset millisecond value.	3 bytes
SSM	(+)123456789X	Advanced synchronization message. Sends the basic sync message, as well as the first 9 signs of a unix timestamp. X is a checksum calculated by the Luhn checksum algorithm.	13 bytes
Hydrophone	ack01	Acknowledgement of basic synchronization message	5 bytes
Hydrophone	ack02	Acknowledgement of advanced synchronization message	5 bytes

Table 4.1: Synchronization related messages sent over the RS-485 line.

All messages are sent through ASCII representation. This provides some extra security as it may produce "illegal" numbers if something goes wrong during transmission. Since a timestamp constitutes of 32 bits, it has 10 numbers in the decimal system. The timestamp is sent as a stream of nine bytes. The last number is of course not sent as the protocol will only set the clock at a whole 10 second. At last the Luhn checksum number is sent.

4.5.2 Keeping accurate time and sending synchronization messages

One of the modules that runs on the SSM is the time keeping module. This module is interrupt driven. It runs as a clock that sends synchronization messages every

10 seconds. In the current version it sends the basic messages every 10 seconds, as mentioned, and advanced messages with full timestamps every 60 seconds. The flowchart for this process can be seen in Figure 4.4.

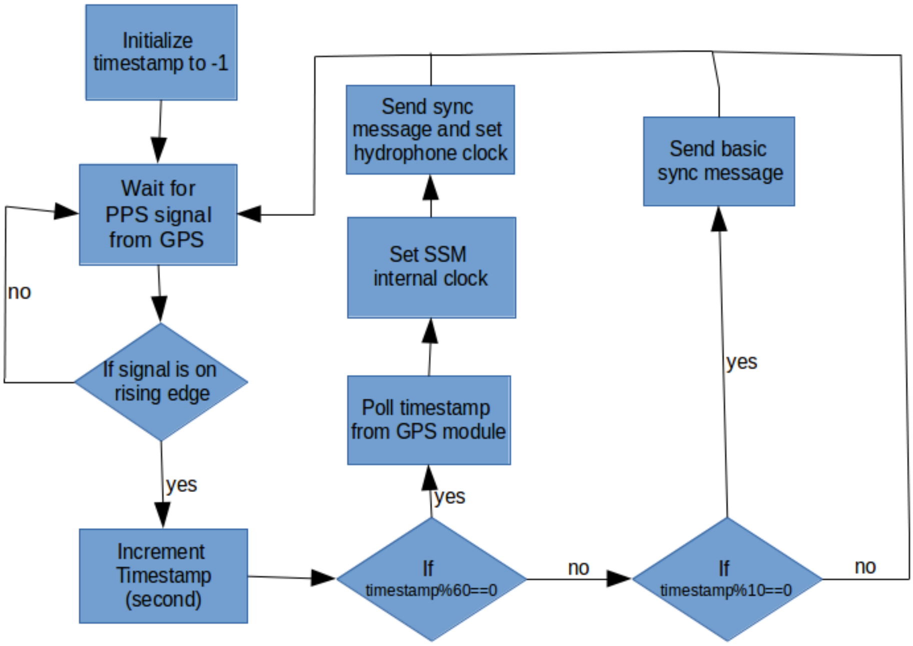

Initialize timestamp to -1
Wait for PPS signal from GPS
no
If signal is on rising edge
yes
Increment Timestamp (second)
If timestamp%60==0
yes
Poll timestamp from GPS module
Set SSM internal clock
Send sync message and set hydrophone clock
no
If timestamp%10==0
yes
Send basic sync message
no

Figure 4.4: Inner workings of the time synchronization software

The clock is kept by listening to the PPS signal of the GPS module. This signal is connected to an interrupt pin on the microcontroller, such that it has priority over other tasks that run in software. It is an atomic action that triggers once every second. That is when a rising edge comes in through the PPS line. The action is non-interruptible when it has started, and occupies key resources like the SPI bus for its duration. If the system has not found a GPS fix, this module will become idle, as no rising edges can happen on the PPS line. That particular behaviour is configurable through the GPS communication protocol used.

4.6 Network implementation

Network implementation is fairly straightforward as tinymesh comes equipped with a lot of features to simplify incorporations of its radio into embedded designs. The tinymesh protocol takes care of any collisions that might occur. It also provides methods like CRC for ensuring error free transmission.

There are as mentioned earlier two important types of configurations for the tinymesh radio. First it is the router. This is the configuration which makes up the bulk of a tinymesh network. The routers are the nodes that are part of the sensor array, and their purpose is to send and forward data to the gateway. The mentioned gateway is the second important configuration. There can only be one gateway in a tinymesh network as it is the endpoint for data transmission. Messages transmitted from the gateway will be received and delivered to the microcontroller at every router. For the messages sent from the routers, however, they will only be received by the controller at the gateway. This enables the gateway to control its routers. The gateway can for example communicate a message when it has finished receiving data, enabling the microcontrollers to turn of their corresponding radio in order to save power.

4.6.1 Tinymesh set-up

The tinymesh can be set in configuration mode by pulling the CONFIG pin low. This is acknowledged by a $' >'$ sent on the UART from the network module. All SSMs should be set as routers by sending a 'R' (ASCII) to the tinymesh module. It is also important that all are configured to the same channel and frequency. The tinymesh module is set back to normal operation mode by sending an 'X' through the UART interface. Radio set up should be done upon start-up of the SSM.

4.6.2 Transmission of messages

When the tinymesh module is in operating mode, the SSM transmits messages by simply sending a stream of characters on the UART to the radio. These characters will in turn be forwarded automatically to the gateway when the maximum buffer size of the tinymesh has been reached, or a $20ms$ time out period since last message reception has been passed.

4.6.3 Memory entries

The memory entries in the flash memory of the SSM should contain both GPS data as well as the data received from the TBR. These memory entries are written upon reception of tag data from the hydrophone. The current incarnation of the SSM only writes the GPS data, but that is fairly easy to expand upon.

GPS memory entry

The GPS module outputs data through the NMEA protocol.[15] This protocol includes various sentences with different types of data. The SSM developed in this project uses the so called GGA sentence for logging position fixes and fix quality data. Full specifications for this sentence can be found in the NMEA protocol referenced above. The GGA sentence provides essential fix and accuracy data, and is well suited for the application of the SSM. In memory, the GPS log has the format

```
$<ID>,<timestamp>,<latitude>,<longitude>,<fix quality>...
,<No. of satellites>,<HDOP>
```

For example

```
$SSM25,1446716612,6435.99677,N,01051.23965,E,1,12,0.6
```

Where Table 4.2 explains the different parts of the log entry.

Entry	Description
$	Sync sign such that the software knows where a new memory entry starts
SSM25	Identification of type of memory entry (SSM), as well as node identification(25)
1446716612	Unix timestamp for fix data
6435.99677,N	Latitude on the form 64^o and 35.99677 minutes
01051.23965,E	longitude on the form 10^o and 51.239625 minutes
1	Fix quality, 0=invalid, 1=GPS fix, 2=DGPS fix 3=PPS fix (see NMEA protocol for elaboration[15])
12	Number of satellites tracked at the time of fix acquisition
0.6	Horizontal dilution of precision

Table 4.2: GPS memory entry description

Tag data memory entry

The tag data memory entries can be written to the SSM flash as follows

```
$<TBR ID>,<timestamp[s]>,<timestamp[ms]>,<code type>,<tagID>...
,<Pressure data>,<SNR>
```

For example

```
$TBR05,1446716612,123,S256,2,233,50
```

Where Table 4.3 explains the different parts of the log entry.

Entry	Description
$	Sync sign such that the software knows where a new memory entry starts
TBR05	Identification of memory type and TBR serial number
1446716612	Unix timestamp for tag detection
123	Millisecond timestamp
S256	Code type. S256= 8bit ID and 8bit data
2	Tag ID
233	Data value from pressure sensor (0-255)
50	Signal to noise ratio (SNR)

Table 4.3: TBR memory entry description

4.6.4 Protocol for transmission

The plan is to unload bursts of data at a set periodicity. This will be done by clocking the memory entries out of flash with the SPI bus and transmit them to the tinymesh module on the UART directly. Each memory entry starts with a $-sign which provides a synchronization method for the gateway when it receives the packages. If there is more data to unload than the buffer size of the tinymesh module, one should specify a delay time for the microcontroller until it continues transmission. The delay needs to be long enough so that one is sure that all entries in the buffer has been transmitted to the gateway. This is due to the minimal hardware design causing the CTS pin of the tinymesh module to be unavailable to the microcontroller.

4.6.5 Limiting power consumption

It is preferrable to keep the tinymesh modules in sleep mode as much as possible, due to their power consumption. This can be done by toggling the CONFIG pin, and sending a sleep command on the UART.

As the SSMs have perfectly synchronized clocks, one should be able to use their clock in order to accurately wake up all tinymesh devices at the same time. Waking up is done by toggling the config pin. After waking up, the devices will need some time to form the network. After that guard time, memory entries should be clocked out of the flash and sent through the network.

When the SSMs receive a ”sleep” message from the gateway, the tinymesh modules will be put to sleep by toggling the config pin as described above.

The periodicity of the wake up of the tinymesh modules is implemented by a modulo division. Such that if one for example were to choose 1 minute as the period, the wake up procedure should be done when $timestamp\%60 == 0$ in the main while loop of the software.

4.6.6 Avoiding scheduling conflicts

The SPI bus is used both by the flash memory and the GPS unit. As the GPS is interrupt driven, it can take control of the SPI bus, even when the bus is in use for writing/reading the flash memory. This problem should be avoided by preventing the interrupt routine to access the SPI bus when the flash memory is in use. For implementation of this one can use a global variable as a semamphore, which is set each time the flash is accessed in software. It should then be cleared alongside with the SPI chip select when the software has finished its usage of the flash. This semaphore can be used to prevent sending of advanced synchronization messages, and thus make sure that the SPI bus won’t be taken by the interrupt routine.

As the advanced synchronization message is just a safety mechanism in case something goes wrong, it will probably never be of any consequence to replace it with the basic sync message for those special cases.

Chapter 5

Server implementation

This chapter will briefly explain the the software which has been developed for the server, as well as describing the server hardware. These are Python implementations. At the current time, the positioning algorithm, as well as the web server is fully developed. What remains on the server is to develop the module that is supposed to handle input from the tinymesh radio. The basic principles and software of such a module is still described.

5.1 Server concept

A Raspberry Pi 3 Model B (RPI3) is the server core. It uses a linux Debian-based operating system called Rasbpian. The RPI3 comes with both Wi-fi and an ethernet interface for internet connection. To communicate with the tinymesh radio, it uses an UART interface found on the GPIO pins of the RPI3 as well as other GPIO pins for connection to the CONIFG/CTS/RTS pins on the radio.

The radio configuration at the server should be set to gateway.

A flowchart of the modules in the server and how they communicate can be seen in Figure 3.1.

5.2 Radio event handler

The radio event handler is the module that is listening to the Tx pin on the tinymesh radio. It should parse all data received on the radio, and communicate a sleep message to the routers once the the SSM data is finished uploading. A flowchart of this process can be seen in Figure 5.1.

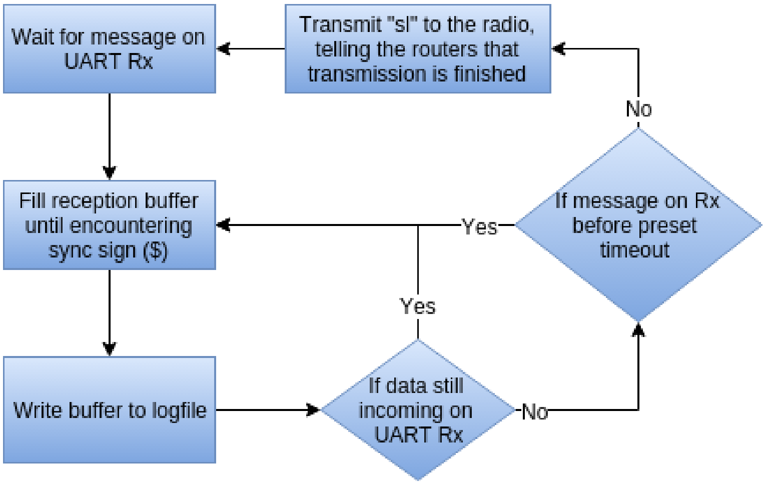

Wait for message on UART Rx
Transmit "sl" to the radio, telling the routers that transmission is finished
No
Fill reception buffer until encountering sync sign ($)
Yes
If message on Rx before preset timeout
Yes
Write buffer to logfile
If data still incoming on UART Rx
No

Figure 5.1: Flowchart describing the general work flow of the radio event handler

5.2.1 Log files

The log files are organized as follows

- One SSM log file for each system node.
- One hydrophone log file for each system node.
- The log files are named by the identification of their originating module.

The data is stored in '.txt' files with filenames taken from the identification of the memory entry. (For example 'SSM25.txt' and 'TBR25.txt') Each time a memory entry is received from the radio, the radio handler writes it to a new line in its corresponding log file.

5.3 Positioning algorithm

The positioning algorithm utilizes the methods derived in Chapter 2. First the system has to search the hydrophone log files, and organize the signals that are present on more than three hydrophones. This is done by filtering on timestamps, and ensuring that the signals have transmitted the same data package. The signal is then passed into a struct within the software which contains timestamp and SNR data from all nodes, as well as the depth measurement. This will be done to all positioning eligible signals.

The process of decoding such a struct into a position estimate is shown in Figure 5.2.

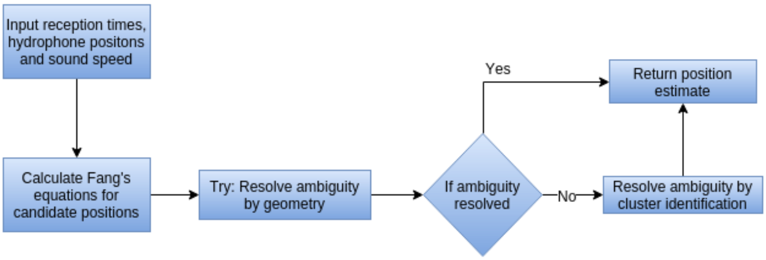

Input reception times, hydrophone positons and sound speed
Calculate Fang's equations for candidate positions
Try: Resolve ambiguity by geometry
If ambiguity resolved
Yes
No
Return position estimate
Resolve ambiguity by cluster identification

Figure 5.2: Flowchart describing the positioning of a single signal

The position estimates are then passed into an array of structs, and the information of those is written to a log file with a name indicating tag ID. A position data entry contains the following

```
<timestamp>,<x>,<y>,<z>,<Tab>,<Tac>,<Tad>,<snrA>,<snrB>,<snrC>,<snrD>
```

Where

```
<timestamp>,<x>,<y>,<z>
```

is the position data for the signal. The other parameters will be explained in Chapter 6 as they are mainly intended for enabling more data analysis.

The source code for this entire process can be found in the attachment submitted with this master thesis, as it is too extensive to provide in the appendix.

5.4 Web server

The web server is a simple python implementation of TCP communication using the HTTP protocol. It allows a user to access the log files through a web browser. The source code for this server is provided in Appendix C. An example of accessing such a log file is shown below

```
http://xxx.xxx.xxx.xxx:port/position_log_71.txt
```

Where ”xxx.xxx.xxx.xxx” is the IP of the server, port is the port number used (4083 in the provided source code) and 71 is the tag id of the position log to be examined.

Chapter 6

Experimental tests for hyperbolic positioning

The surface support modules have been tested at two separate occasions at Bjørøya Fiskeoppdrett.

First as part of a larger research project named CycLus. The resarch project is about evaluating lumpsuckers as cleaner fish in sea cages. CycLus is a joint project between different universities and companies in Norway. Among them Bjørøya AS, NTNU, the University of Nordland, INAQ AS and Thelma Biotel AS. Integral to the field tests conducted at Bjørøya was to gather behavioural characteristics of the lumpsuckers in relation to the salmon.

36 lumpsuckers and 36 salmon were implanted with acoustic transmitter tags, and evenly distributed among three sea cages. That is 12 tagged salmon and 12 tagged lumpsuckers in every cage. Four hydrophones were placed in each sea cage. Three at the periphery, with equal spacing between them, and one in the middle of the cage. This field test, which involved fish tracking, went from the 4. of November to the 4. of December 2015.

There were installed SSMs in one of the cages. Because of some unfortunate technical issues with the deployment, they only generated positioning data for the last week of the field test. Two of the hydrophones were synchronized throughout the whole period. But of course, three is needed for providing position fixes. These results were discussed in my project report from the fall of 2015, that was however at a point where the positioning algorithm wasn't fully developed. So they will be briefly re-examined in this chapter.

A second test was done in May 2016. As the first test contained moving targets, it was not well suited for determining the precision or accuracy of the system. The second test however was done by using transmitter tags at known locations with fairly high data transmitting frequency to ensure a large data set. The second test should have been the first, but that wasn't possible due to time constraints. However, the second test was a good way to verify the data collected in November.

As such, there will in this chapter be presented both a study of the quality of the positioning system, and later some real fish tracking data will be discussed.

When discussing terms like precision and accuracy in this chapter the ISO definitions will be used. Accuracy will be used to describe the trueness of the measurements, which may relate to systematic errors or statistical bias. Precision is used to describe the statistical variability of the measurements. This is according to ISO 5725.[16] Note that accuracy by this definition includes precision.

However, the "true" positions of the transmitter tags, as well as the hydrophone positions was found using a GPS measuring instrument which may itself produce an error, thus limiting our ability to discuss trueness. It was an ikeGPS300 and according to its specification, the error should be no more than $60cmHRMS$, so it will still serve as a good indicator for accuracy. Also note that during this chapter, the measurement data itself, the time differences, will emphasized in the discussion regarding precision. The position data will of course also be discussed but more in the context of the performance of the algorithm developed as the relation between the time differences and the position estimates is already detailed in Section 2.9.

Finally it should be noted that this chapter is mainly concerned with displaying the data. Features of the data will be commented to set up for the discussion following in the next chapter.

6.1 Tools for data analysis

All data generated through the experiment were parsed by the positioning algorithm (using Python) described in Chapter 5 and thus organized into a position logfile. The logfile contains strings of the format:

```
<timestamp>,<x>,<y>,<z>,<Tab>,<Tac>,<Tad>,<snrA>,<snrB>,<snrC>,<snrD>
```

Where each entry is one signal with position fix, SNR data of the signal for all receivers, and some selected time differences. The data showcased in this chapter is taken directly from that logfile and analysed using Matlab.

6.2 Hydrophone set up

As established in Section 2.9 the best way to organize the system is to use an equilateral triangle of receivers. The experiments were conducted in a sea cage, and therefore the SSMs and hydrophones were mounted on the periphery ring of the cage. Also one receiving node was placed in the middle of the cage in order to enable the possibility of redundancy in the position calculations as well as ensuring wider coverage. An acoustic signal's travelling distance is limited by absorption and spreading loss. It is therefore not possible to assume that a given signal will reach all hydrophones. Equally distributed spacing between the hydrophones in the cage should hence be used in order to gather as much data as possible.

All hydrophones were set at the same depth, $z = 3$. Their positions were measured using the GPS instrument, and transformed to meaningful coordinates for the positioning algorithm by a flat earth projection. (using the matlab function lla2flat [17].) The same was done for the transmitter locations. The hydrophone set up can be seen in Figure 6.1, where the red circles (A, B, C and D) are the receiver nodes. For the fish tracking experiment, the middle node was not measured by GPS, and is therefore just assumed to be exactly in the middle as intended.

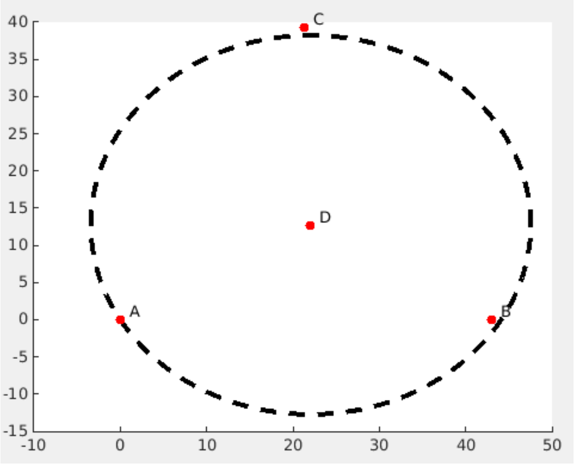


Figure 6.1: Hydrophone set up

6.3 Positioning performance

This section will investigate the positioning performance of the system using data gathered during the second experiment at Bjørøya on the 26. and 27. of May 2016. The performance will be investigated in terms of TDOA accuracy/precision as well as output from the positioning algorithm. It should be noted that for this section, there were no fish present in the sea cage. It was thus a favourable environment for high precision positioning, compared to what would usually be the case.

6.3.1 Experiment plan and set up

There were two phases of this experiment. The major one was to investigate the positioning performance of the system using the developed algorithm and the same receiver node configuration as for the earlier fish tracking experiment. Thus enabling discussion around the precision of the fish tracking data generated by the algorithm.

Secondly, a new experimental firmware for the TBR was briefly tested. The firmware was supposed to increase the accuracy of the timestamping mechanism at the hydrophone. One of the findings in my project report of last fall was that timestamp accuracy would decrease with lower signal to noise ratio. (SNR) As the details surrounding the different timestamping methods is the property of Thelma

Biotel they will not be discussed in great detail. However, the point of the new experimental firmware was to compensate in the timestamping method when encountering signals with low SNR values.

The set up for the first phase can be seen in Figure 6.2, where the red circles (A, B, C and D) are the receiver nodes and the '+'-signs (pos1,pos2...) are GPS measured transmitter positions.

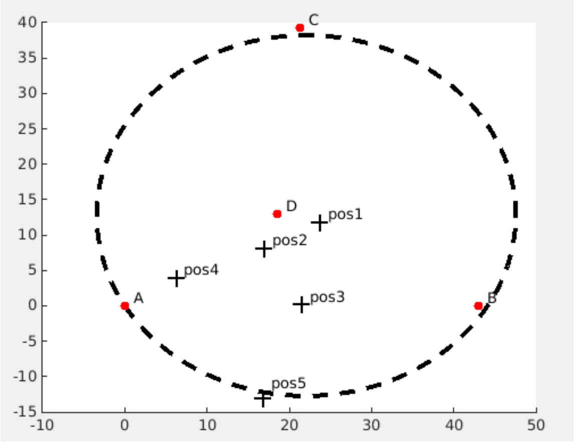


Figure 6.2: Experiment set up for the first phase

During the first phase, the middle receiver were at two different positions. First it was at the point denoted as $pos1$ in Figure 6.2, as the transmitter and receiver were anchored to the same buoy. After the transmitter was recollected for testing other positions, the middle receiver was deployed again and measured to be at the position denoted as D in the figure.

6.3.2 Data set and presentation

The data set was generated using a $9mm$ Thelma Biotel transmitter tag designed for test purposes. It essentially is the same type of tag that was used for the fish tracking experiment, except that the DPPM modulated signals are sent more often. The transmitter tag was attached to a buoy and anchored down to a depth of $3m$, then dragged into position using ropes. When the transmitter was in position, the buoy's position was measured using GPS. These positions are found in Figure 6.2, and the data generated at each position is displayed in Table 6.1. Here position data points refers to amount of position fixes that were found when searching the hydrophone logs using the positioning algorithm developed in Chapter 5.

Position	Time (minutes)	Position data points
1	1073	3455
2	38	155
3	33	121
4	24	74
5	22	77

Table 6.1: Data generated for each position

Timestamping performance

I will throughout this section give plots on the statistical spread for the timestamping performance. As the timestamps are discrete with a $1ms$ resolution, their statistical likelihood will be displayed in histograms. In the notation used for describing the plots, $T_{ab} = T_a - T_b$, is the time difference, where T_a is the signal reception time at node A and T_b is the reception time at B. Same applies for T_{ac} and T_{ad}. The reason for choosing to display these particular time differences, is first of all that they contain all the data from the outer triangle. T_{bc} will not produce any new position data as it is dependent on T_{ab} and T_{ac} by $T_{bc} = T_{ac} - T_{ab}$. Also, this data includes all there is to know from the lower inner triangle. For the positions tested in this experiment, the lower inner triangle performs the best considering DOP, and is the natural triangle to include when evaluating the data. The nature of how the other triangles will perform is already shown in Section 2.9. It is purely reliant on mathematics and therefore not necessary to evaluate through experiments.

The theoretical accurate timestamps are calculated using the accurate position measured by GPS, and the speed of sound as measured by the CTD probe. For a target position (x, y) using the sound speed c, the theoretical accurate time difference for T_{ab} where node A is at (A_x, A_y) can be found by

$$T_{ab} = \frac{\sqrt{(x - A_x)^2 + (y - A_y)^2}}{c} - \frac{\sqrt{(x - B_x)^2 + (y - B_y)^2}}{c} \tag{6.1}$$

As the resolution of the hydrophones are set at $1ms$ any data from the same position within a margin of $\pm 1ms$ is considered precise. Also the GPS inaccuracy should be addressed. The sound speed that was measured for this experiment was $c = 1482m/s$. With a GPS accuracy of $0.6mHRMS$, the maximum time difference inaccuracy that would stem from GPS measurement error would be time it takes for a sound signal to travel $1.2m$. The GPS inaccuracy then translates into a maximum $\pm 0.8ms$ time difference inaccuracy

Position plots

Position data will be shown in two dimensions as the transmitter and hydrophones were at the same depth throughout the experiment. In the position plots one should note that many of the points lay on top of each other. This is due to the resolution of the timestamping. There are a finite amount of positions that will be output by the algorithm in a given area. Along with the position plots, there will also be shown some hyperbolic lines of position (LOP) defined by a specified time difference. The axes shown in the position plots are always in meters.

6.3.3 Position 1: middle of the sea cage

This position is where we have the largest data set, due to leaving the tag in this position overnight during the experiments. It is in many respects the best case scenario for the positioning system due to two important factors. Sound originating middle position has an equal travel distance to all receiving nodes of the outer triangle, which should in theory give similar SNR values at all reception nodes used for hyperbolic positioning. This is important as the SNR values and timestamping accuracy are related. Also, signals from the middle position is optimal when considering DOP, giving relatively small deviations in the position estimates.

In Figure 6.3-6.4 the output from the positioning algorithm can be seen with the blue circles indicating found positions in the data set. The red, blue and green lines are the hyperbolic LOPs defined by the most common time differences for each node pair investigated. Notice that the green line does not have the typical shape of a hyperbola in this plot. That is because the time difference T_{ad} is greater than or equal to the travelling time of a signal from node D to A and thus the transmitter has to be placed more or less directly behind node D.

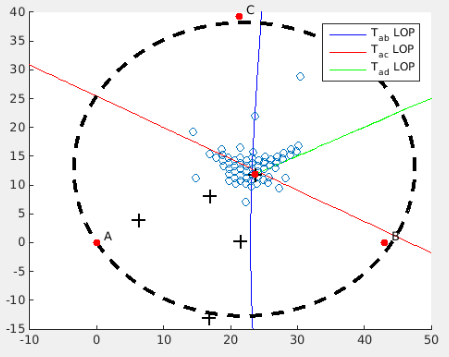


Figure 6.3: Position data for pos1

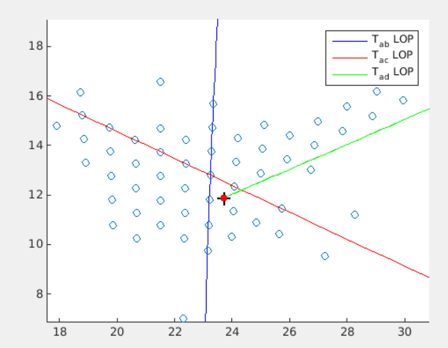


Figure 6.4: Position data for pos1, zoomed

Timestamping performance

The probability mass function of the time differences at selected node pairs is shown in Figure 6.5-6.7. The red line indicates the calculated accurate time difference assuming the GPS measured position to be correct. There's a statistical bias towards the values that are lower than the mode of the time differences. This can possibly be attributed to how the actual time is skewed within the discrete value it is given.

In Figure 6.8 a bivariate histogram is shown. It displays the amount of signals as a function of conjoint time differences. This means that each bin (pillar) of the histogram represents a position estimate and its frequency. Here, T_{ab} and T_{ac} are shown as they are the ones used in the positioning calculation. The reason for showing a bivariate histogram and not only the two-dimensional ones is that it cannot be assumed that the measured time differences of different node pairs are disjoint occurrences. One example of this is that if for a given positioning eligible signal, the timestamping at A is inaccurate, it will affect all three time differences for that particular signal. This actually means that if T_{ab} is accurate, T_{ac} is more likely to be accurate. Other factors related to the acoustic environment might also prove to have an impact, and this will be discussed later.

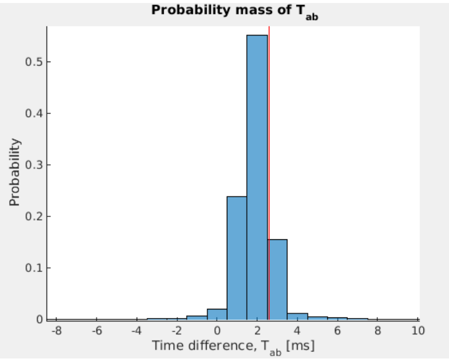


Figure 6.5: Time difference of arrival between node A and B

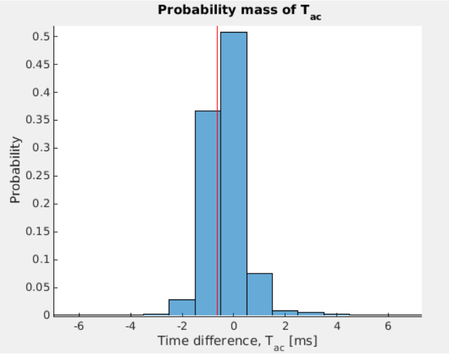


Figure 6.6: Time difference of arrival between node A and C

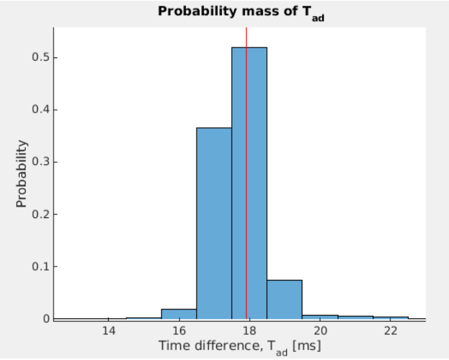


Figure 6.7: Time difference of arrival between node A and D

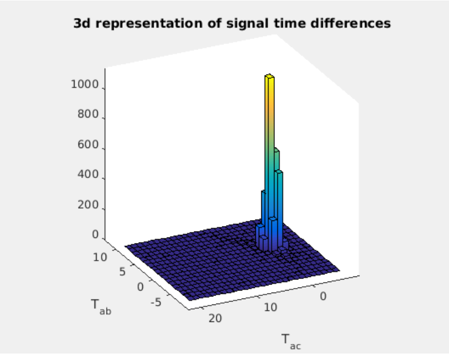


Figure 6.8: Bivariate histogram of time differences

In order to evaluate the precision of the position estimates one can consider the bivariate histogram. The most occurring pair of time differences is seen as the highest pillar in Figure 6.8 at $(\hat{T_{ab}}, \hat{T_{ac}})$. We define the $\pm 1ms$ region as the collection of data points for both $T_{ab} = \hat{T_{ab}} \pm 1$ and $T_{ac} = \hat{T_{ac}} \pm 1$. For seeing how these regions look when translated to the position plots, $(\hat{T_{ab}}, \hat{T_{ac}})$ is at the point of intersection between the LOPs of T_{ab} and T_{ac}. The $\pm 1ms$ region are the six points in closest proximity to $(\hat{T_{ab}}, \hat{T_{ac}})$ as there are 6 different combinations of time differences defined for that region. The cumulative distribution of the position data within these regions can be seen in Table 6.2.

Region	Cumulative distribution	Cumulative percentage
$(\hat{T_{ab}}, \hat{T_{ac}})$	1147	33.2%
$(\hat{T_{ab}} \pm 1ms, \hat{T_{ac}} \pm 1ms)$	3166	91.6%
$> \pm 2ms$	3455	100%

Table 6.2: Cumulative distribution of position fixes from a time domain perspective for position 1

Signal to noise ratio

Signal to noise ratios can be seen in Figure 6.9-6.10. The SNR values are simply a dimensionless logarithmic measure of the signal quality and I am thus only able to describe them in relation to each other.

For the case of a transmitter located in the middle of the cage, it was expected that the SNR values would be quite similar at the nodes of the outer triangle. This is largely true, at least when compared to other transmitter positions that were tested. However acoustic environments are fairly unpredictable, and there were other transmitters interfering, as well as some ship traffic and echo sounders which may have skewed the SNR values somewhat.

An interesting thing to point out for this position is that the SNR values at D seems unrealistically low compared to the receivers that are farther from the transmitter. As the transmitter was mounted at the hydrophone above the transducer, it is probable that the casing of the hydrophone acted as an acoustic shield to the transducer for the direct arrival rays and thus lowering the signal strength at the transducer.

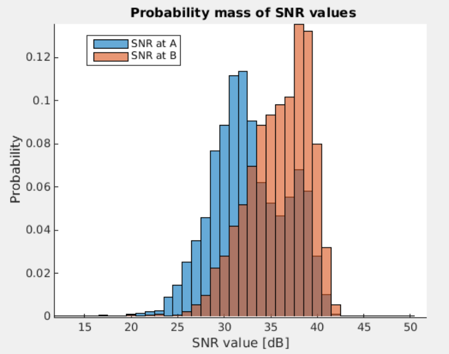


Figure 6.9: Signal to noise ratio at A and B

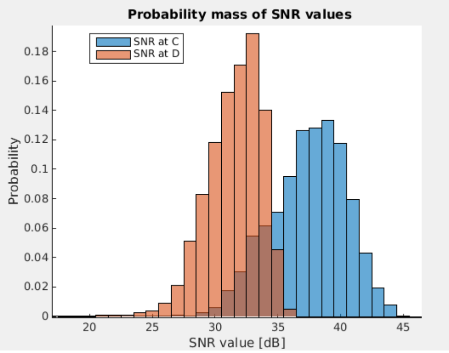


Figure 6.10: Signal to noise ratio at C and D

6.3.4 Position 2: Middle of inner triangle

The data for this position is shown in Figures 6.11-6.18 and Table 6.3. The point was to place the transmitter in optimal DOP conditions for placement by the lower inner triangle.

Here the position estimates are skewed significantly away from the coordinates measured by GPS. There's a bit of mystery as to why that is. It could be due to some corruption in the GPS measurement. Another explanation is that something went wrong at node B as T_{ab} is the only time difference that seems out of balance. The SNR values at B also seems small when compared to those of A. The third explanation is that it could be movement in the sea cage itself, causing the receivers mounted at the periphery to be moved with it.

As for the precision it is similar to that of position 1. Everything mostly lay within the $\pm 1ms$ (96.8%) margin as seen in the table, which translates into $\pm 2m$.

The LOPs for the most occurring time differences also require comment. Notice that the green line, defined by T_{ad} has no intersection with the red line. In practical terms this means that the position is in a shadow zone for the triangle defined by the node pairs of the non-intersecting LOPs. In this case it means the triangle A, C, D. However at the triangle A, C, D, the DOP conditions are really good, and it produces position estimates of almost equal quality to that of the outer triangle. This can be seen by the T_{ad} LOP as well as the probability mass distribution.

The transmitter is close to D, and as a result, the SNR values at D are comparatively higher than the others by a significant margin.

Region	Cumulative distribution	Cumulative percentage
$(\hat{T_{ab}}, \hat{T_{ac}})$	51	32.9%
$(\hat{T_{ab}} \pm 1ms, \hat{T_{ac}} \pm 1ms)$	150	96.8%
$> \pm 2ms$	155	100%

Table 6.3: Cumulative distribution of position fixes from a time domain perspective for position 2

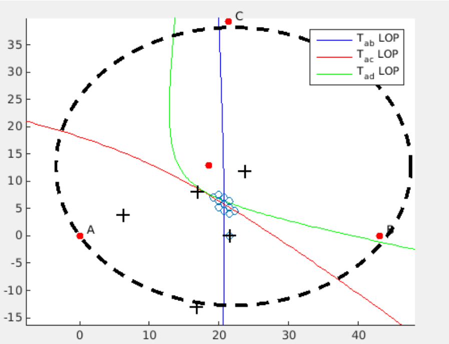


Figure 6.11: Position data

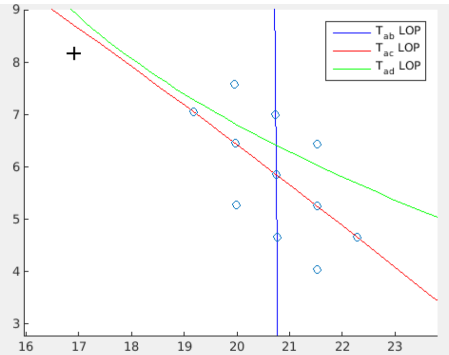


Figure 6.12: Position data zoomed

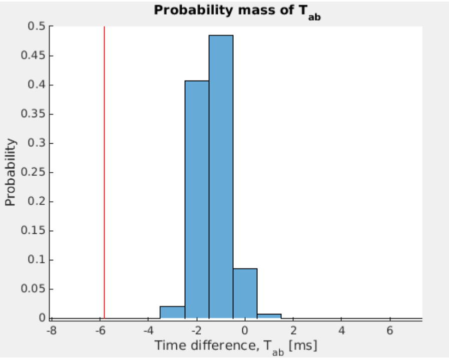


Figure 6.13: Time difference of arrival between node A and B

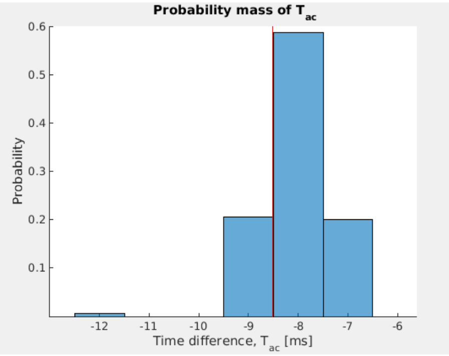


Figure 6.14: Time difference of arrival between node A and C

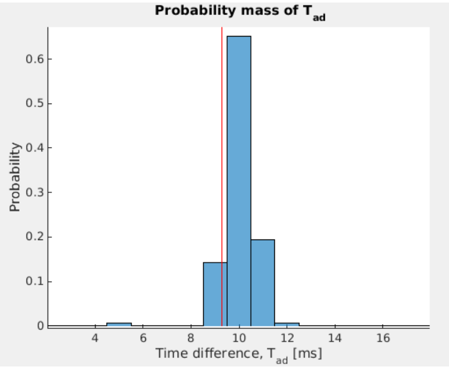


Figure 6.15: Time difference of arrival between node A and D

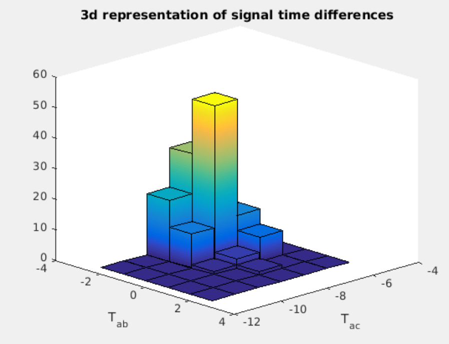


Figure 6.16: Bivariate histogram of time differences

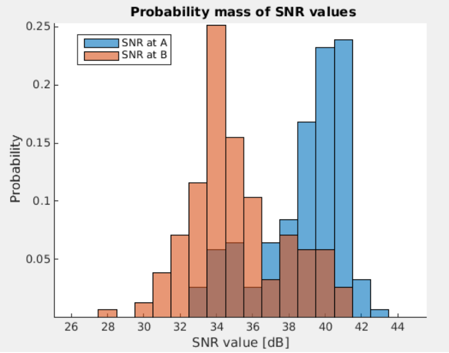


Figure 6.17: SNR at receivers

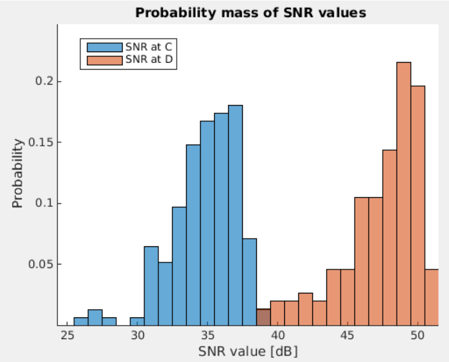


Figure 6.18: SNR at receivers

6.3.5 Position 3: Middle of the baseline of the triangles

The data for position 3 can be seen in Figures 6.19-6.26 and Table 6.4. It was placed at the baseline of the triangles, a relatively good spot when considering DOP. As seen by the position plots, the acoustic positioning system is close to hitting the mark of the GPS measured coordinate. From the table it is seen that 94.2% of the signals are within $\pm 1ms$ which translates roughly to within $\pm 1.5m$.

The same indication is reflected in the time difference plots. As for the SNR values, the highest are found at D which is logical as that is the receiver in closest proximity to the transmitter. The values at A and B can be seen as fairly similar, which makes sense as they have an equal distance between themselves and the transmitter.

Region	Cumulative distribution	Cumulative percentage
$(\hat{T_{ab}}, \hat{T_{ac}})$	35	28.9%
$(\hat{T_{ab}} \pm 1ms, \hat{T_{ac}} \pm 1ms)$	114	94.2%
$> \pm 2ms$	121	100%

Table 6.4: Cumulative distribution of position fixes from a time domain perspective for position 3

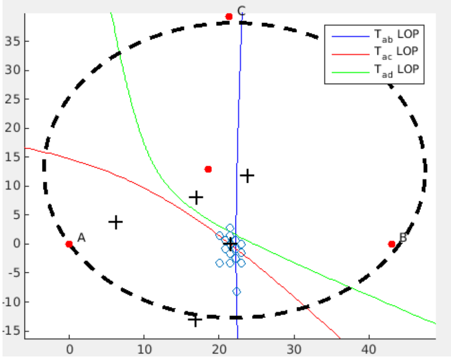


Figure 6.19: Position data

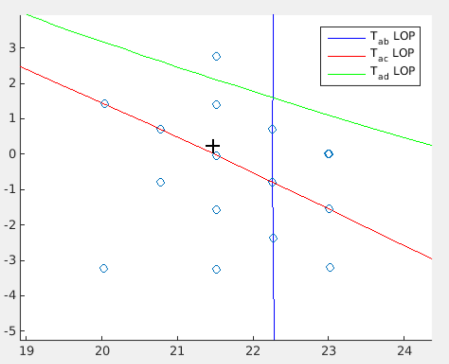


Figure 6.20: Position data zoomed

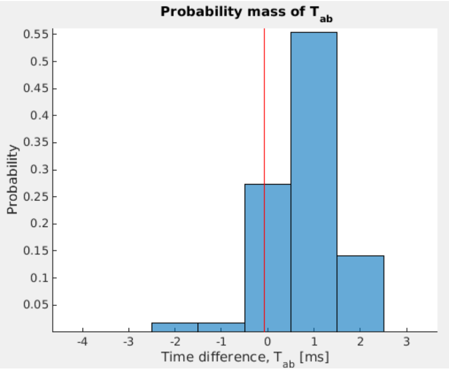


Figure 6.21: Time difference of arrival between node A and B

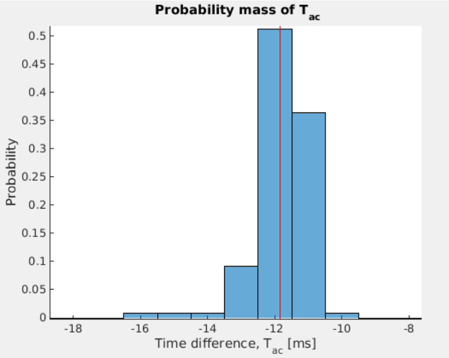


Figure 6.22: Time difference of arrival between node A and C

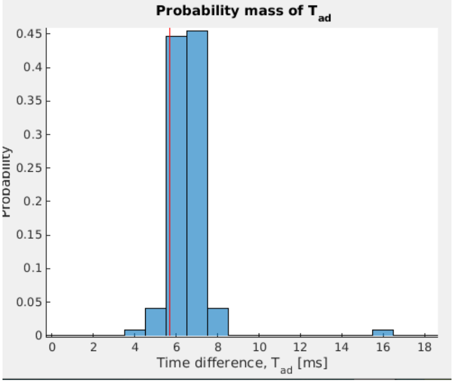


Figure 6.23: Time difference of arrival between node A and D

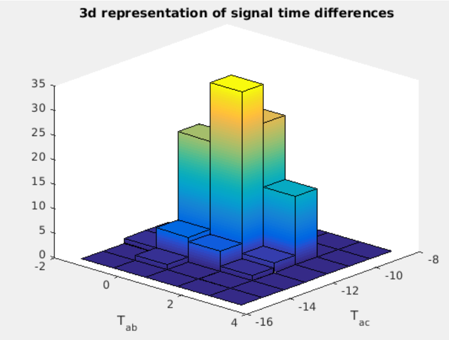


Figure 6.24: Bivariate histogram of time differences

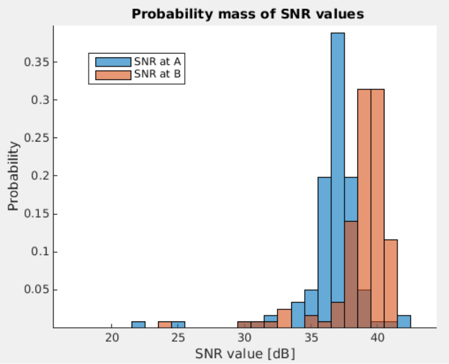


Figure 6.25: SNR at receivers

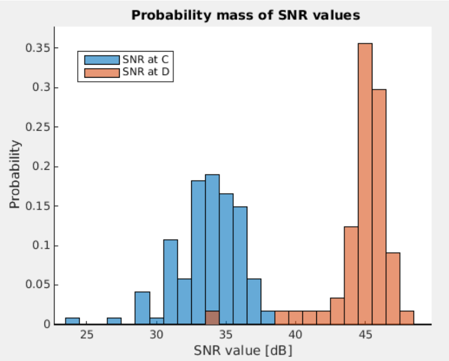


Figure 6.26: SNR at receivers

6.3.6 Position 4: Near node A

The data for position 4 can be seen in Figures 6.27-6.34 and Table 6.5. For this part of the experiment, the transmitter was placed close to a hydrophone near the periphery of the cage. At this placement, the DOP for the inner triangles are high. Another factor that makes this position interesting is the variance of SNR values across the receiver nodes. As the transmitter is placed close to A, the signal strength at A will be significantly higher than at the other receivers. The problem with weak signals is that they might be detected later than their arrival. It is therefore a possibility that the time differences will be displaced to lower values than they should have, as signals at A will be timestamped comparatively early. When T_a becomes smaller, all time differences, T_{ab}, T_{ac} and T_{ad} becomes smaller too, thus providing a bias to all relevant measurements. Lastly it can be noted that at this position, the transmitter is close to, but not inside, regions of high DOP as seen from the outer triangle.

There can be seen some deviations in the time difference plots from the red line. Of course, the possibility of inaccuracies in the GPS measurement should not be discarded. But it may also be the aforementioned effect of T_a being timestamped earlier at the time of signal arrival due to high SNR. All the time differences are skewed to the left which is explained by that hypothesis.

Still the positioning system performs well in terms of precision, and the most common position estimate (occurring in 44.6% of the signals) is within $1m$ of the GPS position.

Region	Cumulative distribution	Cumulative percentage
$(\hat{T_{ab}}, \hat{T_{ac}})$	33	44.6%
$(\hat{T_{ab}} \pm 1ms, \hat{T_{ac}} \pm 1ms)$	72	97.3%
$> \pm 2ms$	74	100%

Table 6.5: Cumulative distribution of position fixes from a time domain perspective for position 4

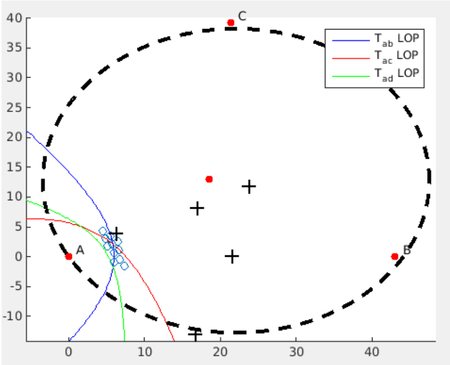


Figure 6.27: Position data

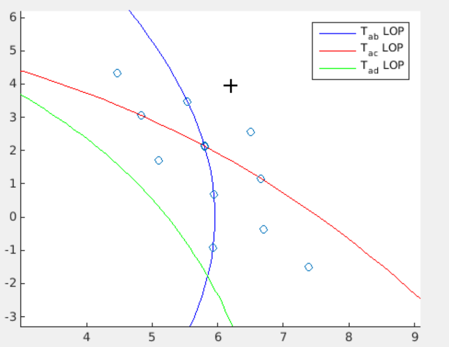


Figure 6.28: Position data zoomed

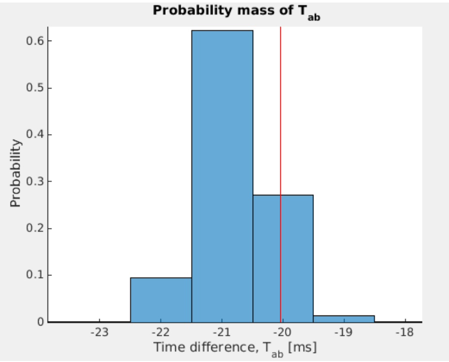


Figure 6.29: Time difference of arrival between node A and B

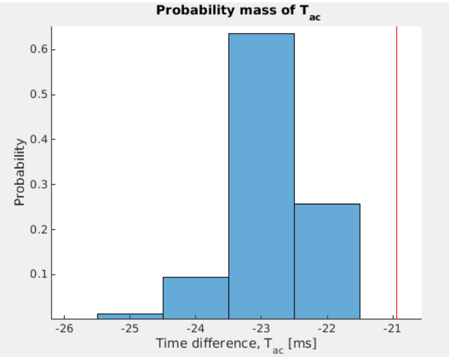


Figure 6.30: Time difference of arrival between node A and C

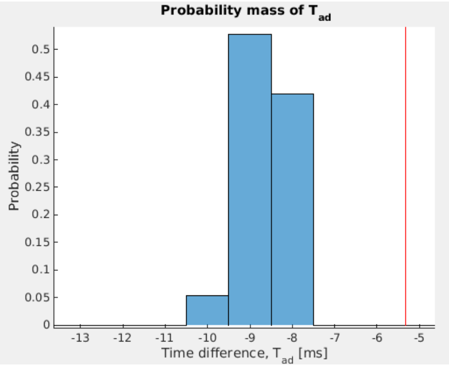


Figure 6.31: Time difference of arrival between node A and D

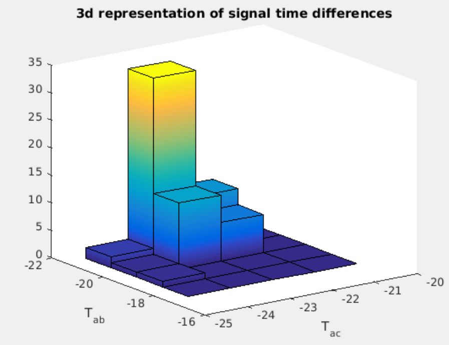


Figure 6.32: Bivariate histogram of time differences

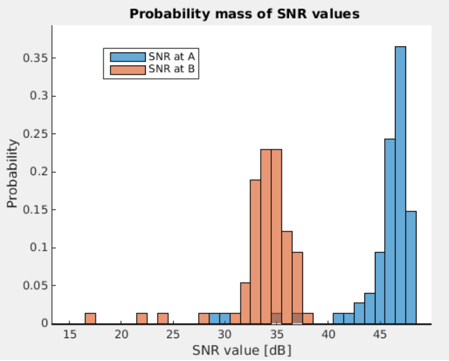


Figure 6.33: SNR at receivers

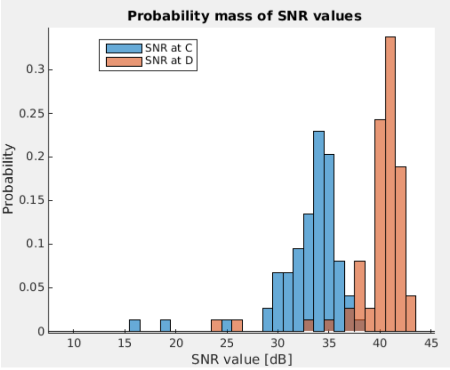


Figure 6.34: SNR at receivers

6.3.7 Position 5: On seacage ring below the triangle baseline

The data for position 5 can be seen in Figures 6.11-6.42 and Table 6.6. Position 5 is at a point with high DOP, and is in that respect a bad case for positioning precision and accuracy. However the positions seems to be accurate. One significant outlier point can be seen at the bottom of Figure 6.11, at more than 30 meters from the transmitter placement. The outlier illustrates the problem of DOP as a similar time difference deviation near the middle of the cage would be a lot less significant.

Note that the DOP parameters will impose larger deviations in the y-direction than the x-direction. This is not only seen from the scattered positions in the plot, but can be seen analytically from the error covariance matrix as described in Section 2.9.

The time differences measured for this position seems to be on point, and close to the theoretical values shown by the red lines. For T_{ac} and T_{ad} there can be seen peaks that are almost equal, and this deviates from most of the other time differences. As the data set is quite small, this could just be a random occurrence. But it is probably similar to the clearly seen statistical bias in the time differences at position 1. The outlier point mentioned above can be seen in the time differences as the point where $T_{ac} = -25$ in Figure 6.38. This value is most likely due to a late detection at node C.

Region	Cumulative distribution	Cumulative percentage
$(\hat{T_{ab}}, \hat{T_{ac}})$	14	18.2%
$(\hat{T_{ab}} \pm 1ms, \hat{T_{ac}} \pm 1ms)$	71	92.2%
$> \pm 2ms$	77	100%

Table 6.6: Cumulative distribution of position fixes from a time domain perspective for position 5

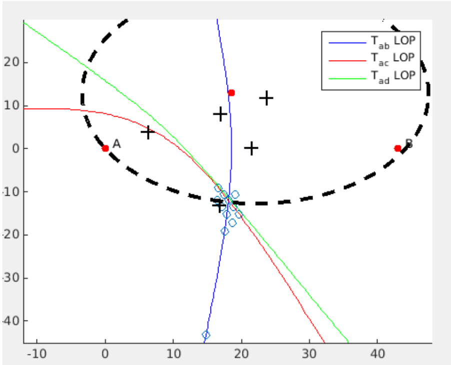


Figure 6.35: Position data

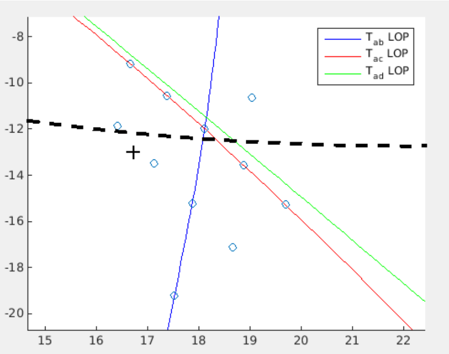


Figure 6.36: Position data zoomed

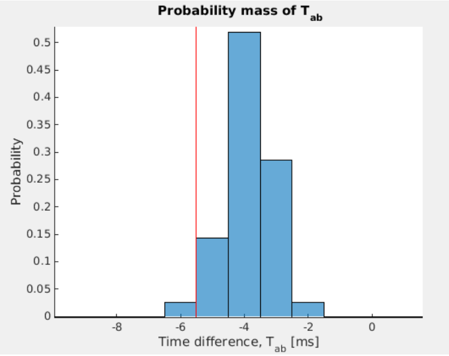


Figure 6.37: Time difference of arrival between node A and B

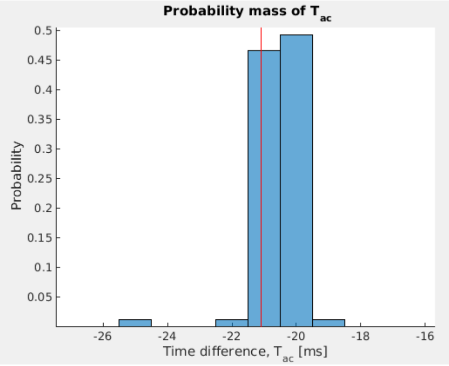


Figure 6.38: Time difference of arrival between node A and C

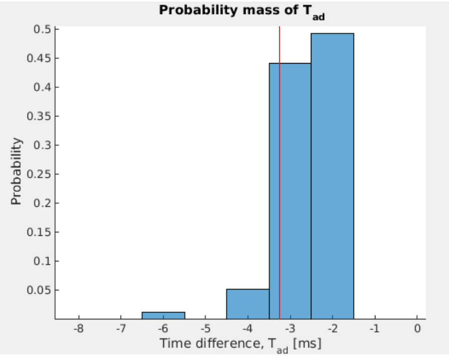


Figure 6.39: Time difference of arrival between node A and D

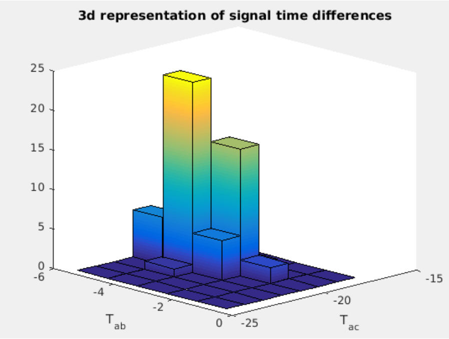


Figure 6.40: Bivariate histogram of time differences

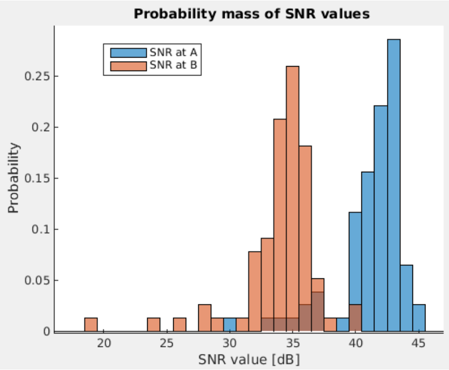


Figure 6.41: SNR at receivers

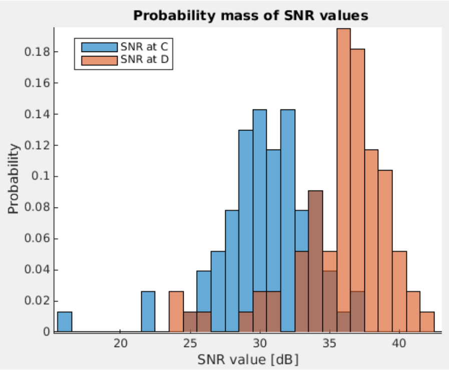


Figure 6.42: SNR at receivers

6.4 Testing the new timestamping method

Plots from the second phase, testing an experimental method for timestamping can be found in Appendix D. The plots are organized in the same fashion as for the first phase and pushed back to the appendix as they don't provide significant additional information. The new method was meant to compensate for SNR when timestamping. It was tested in three positions over short periods causing the data set to be quite small. Also it should have been tested in positions with significant SNR differences among the receiver nodes. Position 4 from phase 1 of the experiment is a good example of such a position.

One thing to note from these plots is that the acoustic positioning system deviates from that of the GPS, in the positions inside of the sea cage. This deviation is quite similar to that of position 2 in the first phase. As stated earlier it might be explained by movements in the sea cage itself. Just before the second phase of the experiment was conducted, a relatively large ship fastened to the sea cage ring making an impact in the process of doing so. That could be enough to displace the ring and subsequently causing the deviations seen in the plots.

6.5 Fish tracking

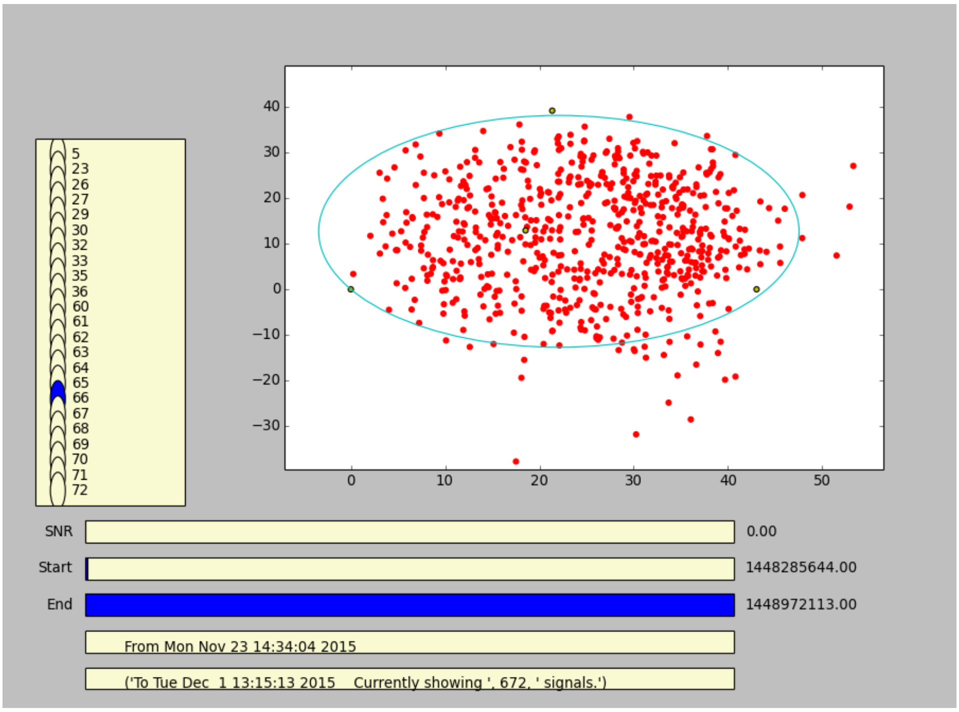
For this section there will be presented tracking data for selected fish monitored during the CycLus experiment in November 2015. Only one week worth of data from time synchronized hydrophones was available. Out of the 12 transmitter tags inserted into the lumpsuckers in the cage, only one remained inside an alive specimen for the last week. The other tags for the most part had sunken to the bottom, presumably due to the fishes dying. Most of the salmon survived to the end of the experiment, with intact transmitter tags within them.

SNR values for the fish-tracking experiment are provided with each of the tracked fishes. The values shown are the lowest among all receiver nodes for each signal. Notice that the probability peaks at about $20dB$ for both of them. This is significantly lower than for any of the receptions in the experiments described above. The low SNR values are due to there being a lot of tags in the water, elevating the noise floor. In total there were 72 tags at the location, where 24 were located in the examined cage.

6.5.1 Salmon

In Figure 6.43, a position cloud for the salmon with tag id 66 is shown. The position data were gathered over a one week period, and there are 672 positions shown in the cloud. This data can also be seen in the figure, as it displays a minimal GUI I developed in order to quickly analyse results. The GUI contains sliders for filtering time periods shown by the plot, as well as one slider to filter out signals with low SNR values. At the left side of the GUI one can select tag ID within the dataset.

The position data for the salmon shows pretty much what you would expect. It is an active species which is constantly on the move within the sea cage. This plot does of course not show the whole story, as there might be behavioural characteristics tied to night time, or when there is feeding in the cage.



SNR 0.00

Start 1448285644.00

End 1448972113.00

From Mon Nov 23 14:34:04 2015

('To Tue Dec 1 13:15:13 2015   Currently showing ', 672, ' signals.')

Figure 6.43: Salmon position cloud

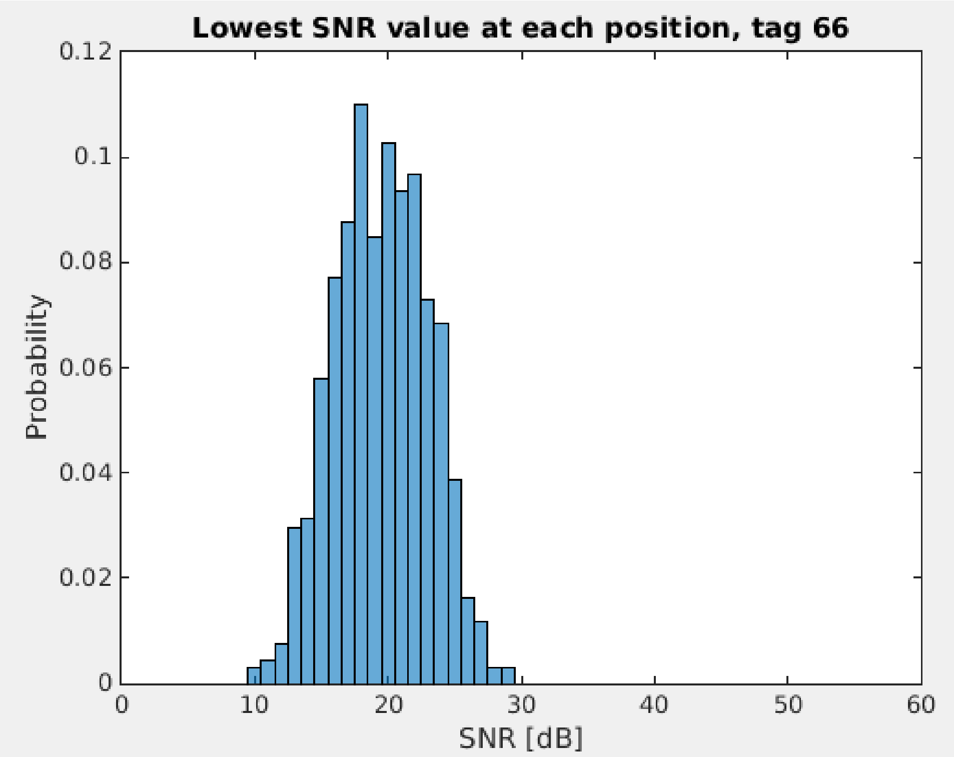


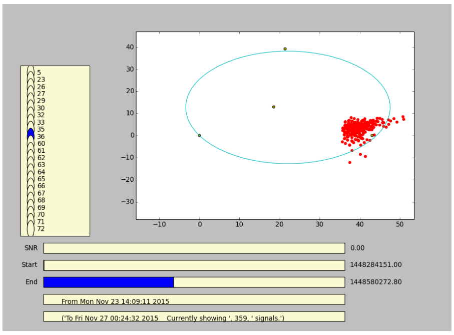
Figure 6.44: Distribution of SNR values for lowest SNR of each positioning eligible signal

6.5.2 Lumpsucker

The lumpsucker has modifield pelvic fins in the forms of adhesive discs which enables it to cling onto objects like for example rocks. This adaptation enables the fish to stay stationary for extended periods of time. In the sea cage there are shelters were they may do that. The lumpsucker are also considered poor swimmers.

Lumpsucker position plots from two different time frames can be seen in Figure 6.45-6.46. One can easily infer behavioural differences between this fish, and the salmon. For the first time frame it is shown that the fish stays in the same region for more than three days. It then migrates to a new region at night, and spends the rest of its time there until the conclusion of the experiment.

A study of the depth log gives concrete proof that the fish is not entirely stationary during the periods the position estimates are contained within certain regions.



SNR 0.00

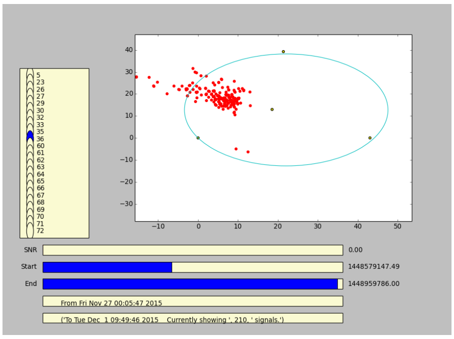
Start 1448284151.00

End 1448580272.80

From Mon Nov 23 14:09:11 2015

('To Fri Nov 27 00:24:32 2015   Currently showing ', 359, ' signals.')

Figure 6.45: Lumpfish positions in time frame 1



SNR 0.00

Start 1448579147.49

End 1448959786.00

From Fri Nov 27 00:05:47 2015

('To Tue Dec 1 09:49:46 2015   Currently showing ', 210, ' signals.')

Figure 6.46: Lumpfish positions in time frame 2

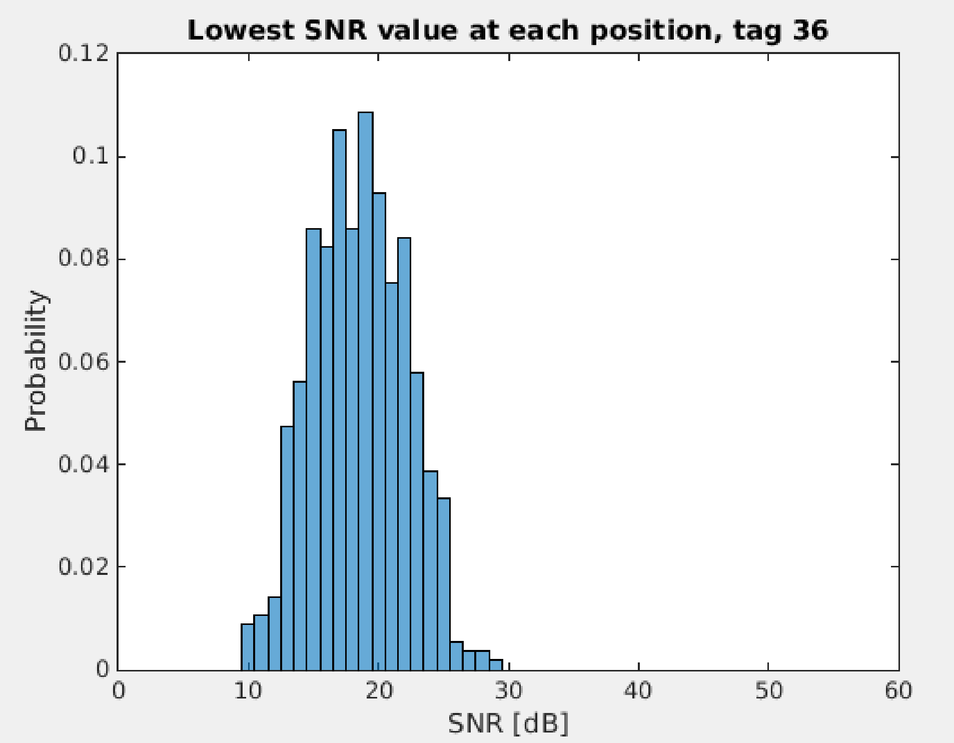


Figure 6.47: Distribution of SNR values for lowest SNR of each positioning eligible signal

Chapter 7

Discussion

In this chapter, the preceding results will be discussed along with proposals for further testing and development. This will be done both in context of the experiment itself, as well as the equipment used.

There will also be a brief discussion regarding the network implementation, its merits and shortcomings.

7.1 Sources of error

When conducting field tests of equipment such as this, it is of vital importance to know your testing environment. The positioning system is reliant on multiple independent measurments at multiple points.

The amount of coupled measurements that might interfere with the system causes the errors to be hard to describe in terms of positioning error. DOP serves well to illustrate this. As a result, small errors in the measurements may grant butterfly effect-like outcomes.

7.1.1 GPS measurements

The GPS measurements taken for last chapter's data set promised to be within $\pm 60cm$. It should be considered here that the GPS was used to measure all hydrophone positions, as well as all target positions. These positions are interlinked, and as such, the output related to GPS measurement errors are hard to describe mathematically.

7.1.2 Movement of sea cage

The sea cage is often anchored to the bottom of the sea. It is of course given some leeway and is thus moveable. Also, the amount of leeway it has in terms of horizontal freedom of movement is affected by the tide. The horizontal freedom of movement is greater at low tides as the rope it is anchored by does not have to be stretched as much in the vertical direction.

There are two potential ways that sea cage movement can cause trouble. The first is what seemed to be appearant in the precision testing. Namely where the GPS measured targets were skewed in relation to the GPS measured hydrophone positions. This is solely a problem related to precision testing, as the acoustic positioning system moves with the cage, keeping its relative geometry wthin the cage intact.

When conducting precision experiments using GPS for accurate positions, a possible solution to this is to do control measurements on the hydrophones each time one measures the test target in the cage. Assuming that the sea cage does not move while these measurements are being done, that should be sufficient.

For example in the precision test from the last chapter, the GPS measured hydrophone positions were taken at the same time as measuring the target position in the middle (Position 1). That was one of the positions where the acoustic system and the GPS agreed the most. The last tests however, shown in Appendix D, were done after an hour of no testing. The reason that the transmitter at the periphery stayed consistent could be that it was mounted to the ring of the cage, causing it to be moved with it. Its position was taken more than an hour prior to the start of the test.

The second type of movement that should be considered is bending effects on the ring. For example in high waves, one side of the sea cage will be placed higher than the other side, taking the hydrophone with it. The hydrophones will then not all be at the same depth, which causes a shift in the output from the positioning algorithm.

7.1.3 Sound speed error

The sound speed was measured by a CTD probe. It is probably fairly accurate. However there were not done continuous monitoring during the CycLus experiment, so the results from those might be affected significantly. This is also one of those errors that are quite hard to deduce from the output. First of all, the speed stays relatively consistent in sea water, so it won't have impacts which makes the position estimates seem unrealistic. Secondly the error has properties which makes the position estimates reliant on the transmitter's relative position to the receivers. For instance if the transmitter is located in equal distance to two receivers, the sound speed will have zero impact as the time difference will be zero. The larger the time differences are, the more the pseudo-ranges will change as a function of sound speed.

7.1.4 Time synchronization errors

The hydrophone has a $1ms$ cycle time when in listening mode, which means that it can only receive a message once every millisecond. This can be seen as a waiting time before it performs synchronization on its clock. There's probably no way to synchronize these waiting times using the current hardware.

7.1.5 Timestamping inaccuracies

The timestamps were found to be within ±$1ms$ at a rate of over 90%. This spread is connected to the errors from the time synchronization mechanism. It is however difficult for me to say too much about it as I don't have detailed knowledge of the software inside the TBR-700RT. What is known is that the timestamping is performed as soon the acoustic energy in one time frame of the hydrophone's resolution, passes a certain threshold. This is what is causing lower timestamping precision at low SNR values as the signal will not get recognized as such during its first impact on the receiver.

The inaccuracies that stems from weak signals are fairly predictable as they will always be timestamped later than they should. Meaning that you can't say it is $\pm x$ as you often do for measurement errors. It is more like $+x$. Let's say you have a signal arriving at two nodes A and B, and it is as strong as can be on A whilst barely detectable at B. The timestamp T_a is then probably correct, while T_b is on average higher than it should be. Meaning that the time difference T_{ab} will be lower than it should be. Though high SNR values always are to prefer, this might mean that in some cases, the time differences will be more accurate (but not precise) if the SNR values at the receivers are similar. Whereas receptions of high difference SNR values will have the property that the time difference will be skewed from the accurate one. It is stressed here that this can skew the accuracy. The precision is the term used for the statistical spread of the time difference values. Meaning that the precision will probably be lower if the SNR values are significantly different. This accuracy bias is commented on for Position 4 (Section 6.3.6). The data for that position might actually show the bias, though one should not conclude as those results might also be the results of other sources of errors. At least it serves as a good illustration for how the skewed accuracy would manifest itself.

7.1.6 Dilution of precision

The concept of DOP is not in itself a source of error, but it explains the relation of the timestamping errors to the positioning errors, and how they vary with the target's relative position to the receiver array. The reason it is mentioned here, is that it can serve as an amplification of other errors. Also, the non-linear relation between the time difference input and the position estimate output serves as a good illustrator for the unpredictability of the system.

7.1.7 Multipath acoustic propagation

High frequency acoustic waves in water are often modelled as rays emanating from the source. This serves as a good model for explaining how acoustic signals can take multiple paths through the water to a given receiver. As an example we can consider an omnidirectional spread acoustic signal from a transmitter tag which is close to the surface. The receiver in this system is also close to the surface. We can model the system as three rays which are angled towards the receiver. One that will be reflected from the surface before reaching the hydrophone, one that take the shortest path, and one bottom-reflected ray. Those three rays will contain the same acoustic pressure when they arrive at the receiver. However, they will arrive at different times. First the direct arrival ray will impact the transducer, after some time delay, the surface reflected ray will come through. The last ray to arrive is the bottom reflected ray. In some cases, the signal might not be detectable by just the direct arrival ray, and the surface/bottom reflected signals might be necessary to make it so. This can in effect cause a slight time delay on the signal.

7.2 Performance of the positioning system

When discussing the performance of the system it is important to be weary of the sources of error and have a basic understanding of DOP. For evaluation of the system accuracy it is not easy to conclude as none of the experiments provided sufficiently accurate measurements of the test conditions. However, when taking into consideration the consistency of the GPS-measured positions in relation to the ones found by the acoustic positioning system it is apparent that the system performs well.

As such, it is obvious that the system performs within a standard which makes it useful for tracking fish motion. Determination of the system accuracy should still be done, and formalized mathematically, as speculative data often can be of poor scientific value. The accuracy should be described in terms of time differences as they are quantitatively easy to measure. Conversion of time difference inaccuracies to positioning inaccuracies is done using DOP theory.

The precision was quantified in the preceding chapter and gives significant suggestions towards the usefulness of the system. Let's for simplicity assume that the most common combination of time differences in the bivariate system, also is the most true set of time differences. That is probably a realistic assumption even though it has not been established formally. The reason for believing so, is that it fits with the theory, and what the system is supposed to do. Also, linear offset-biases which may be happening in the system at its present state, are normally easy to compensate for in software. Using that assumption, the accuracy is illustrated in Figures 7.1-7.3.

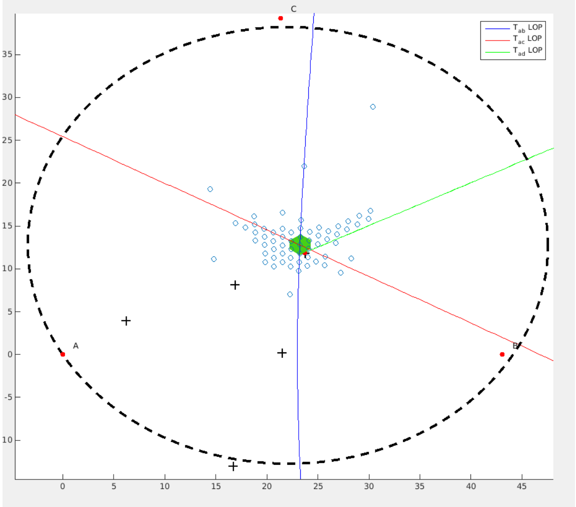


Figure 7.1: Position 1: 91.6% of all signals placed within the green region

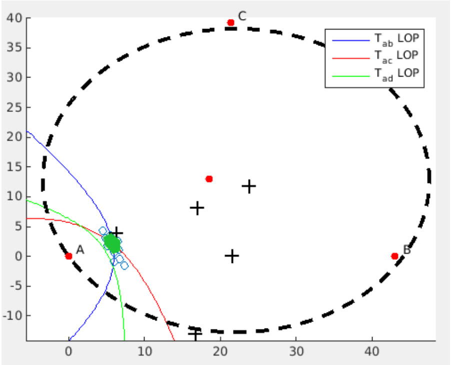


Figure 7.2: Position 4: 97.3% of all signals placed within the green region

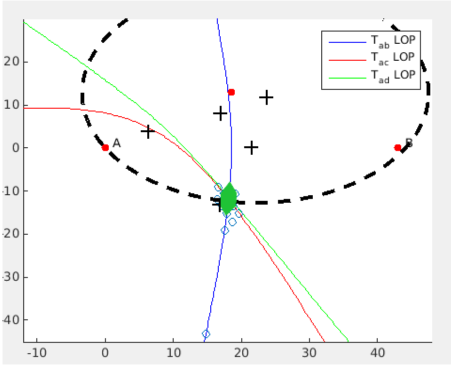


Figure 7.3: Position 5: 92.2% of all signals placed within the green region

The figures shows the same positions as displayed in the preceding chapter, but with the ±1ms region coloured green. They have the shape of a hexagon as there are six different combinations of errors in this region. Using the aforementioned assumption of timestamping trueness, and receiving a signal with estimated position in the middle of the hexagon, one is able to say that there's a more than 90% likelihood of the signal originating from within the green region. These values (likelihood over 90%) were consistent for all positions tested during the experiment. These plots also illustrate DOP as the hexagons becomes more stretched the farther the target is from the middle of the outer triangle.

It should be taken into account that the precision will change with SNR values.

The precision found in the above plots should only be assumed when given higher or equal SNR values to those of the experiment. Meaning that the precision should be re-evaluated if one tries to do fish tracking in an environment of high acoustic noise. Factors that contributes to this noise can be rainfall, ship traffic, echo sounders and most importantly other tags. The acoustic signals will also be affected if there is a lot of fish present in the cage. There is high difference in acoustic impedance between air and water, meaning that sound waves will be reflected upon impacting such an interface. Salmon has swim bladders that are filled with air, causing scattering of the acoustic field and thereby limiting the acoustic energy that reaches the hydrophones. Lumpsuckers do actually not have swim bladders, and are not causing that problem.

7.3 Fish tracking experiments

Given the SNR conditions as in the precision experiment, fish tracking would be equally precise. However as can be seen in the section where the fish tracking data were presented, that was not the case for the particular experiment at Bjørøya in November. It is therefore hard to say anything about the precision of that experiment. Testing the system in noisy environments should therefore be a priority. Especially if one is to track many fishes in a constrained area simultaneously as was done at Bjørøya. Other options include using multiple frequency bands. For example one could use different frequency bands for each sea cage. This is within the capability of the TBR700-RT. Using different frequency bands will not only lower the noise floor, but also suppress the problem of code collisions.

But even though one is unable to say much about the precision of the experiments in a mathematical way, it is obvious that the position estimates tells a story. The difference between the salmon movement and the lumpsucker movement is easily seen. And especially the consistency of the lumpsucker's stationary behaviour gives reason to believe that one can at least recognize movement patterns, even at that low SNR.

It is a difficult task to determine how one should set up experiments such as these. From a technical viewpoint one should limit the transmitter density in the water as much as possible. That would ensure more reliable position estimates and a larger amount of successfully received signals. But for behavioural research, one of course want to track as many individuals as possible. How one balances these two considerations is of vital importance when doing research.

7.4 Positioning system limitations

The capabilities of doing positioning with this system is mainly limited by the fact that it is designed for low power consumption. The $1ms$ cycle time of the hydrophone is an example of this, and is limiting the timestamping capabilities to a discrete amount of milliseconds. If one did not have the demand of low power consumption, the system hardware could also incorporate new things. For example an SRAM and a more powerful micro-controller could enable more sophisticated signal processing methods.

The low power consumption is in many respects advantageous, especially for wildfish surveillance. But at most aquaculture facilities there are power outlets at the sea cages. Thus one is not constrained by low-power demands.

7.5 SSM performance and capabilites

The SSM developed for time synchronization has now been extensively tested at multiple instances. Its precision is by all standards optimal, and the clock synchronization seems to only be constrained by the cycle time of the hydrophone. Also it has shown stable performance when deployed for a month-long period on a sea cage.

When it comes to network integration of the SSM, the hardware is now ready, and tested with the old time synchronization software. What remains is to create software for the network module according to the specifications laid out in Chapter 4. Power consumption also needs to be evaluated. Using the stable SSM software, the SSMs has been proven to be functional for at least one month on a $36Ah$ battery. But the network module will draw power as well. For example its radiated effect of $27dBm$ translates to $0.5W$ when it is transmitting data packages. When the network is active, it is supposed to draw about $24mA$ which will substantially limit the battery time. However as mentioned earlier it can be set in sleep mode, drawing only $0.3 - 1\mu A$.

The fact that the system already has synchronized clocks from the GPS, makes power conservation significantly easier as the modules can be simultaneously woken up to form the network. The proposal now states that the network modules of the SSM should be woken up at a set periodicity. Waking up every 30 minutes is probably sufficient for effective power usage, but it should be investigated how the wake up periodicity affects the power consumption.

One thing which would be a significant improvement of the SSM is to incorporate a more precise GPS module in it. That way, the SSM could hold its own position, and would not be reliant on measurements from proprietary equipment. If it finds its own position, one would also be able to analyse sea cage movement, and probably get rid of that source of error.

7.6 Server and positioning algorithm

7.6.1 Web server

The server developed for this thesis is fairly simple, and satisfies the needs one has when wanting to access the logs and outputs from the positioning system. It could be expanded to handle user input. For example is it technically doable to set the SSMs data unloading periodicity from the server. But other than that, the web server itself is just a bridge for accessing the data logs remotely.

7.6.2 Positioning algorithm

The positioning algorithm is already discussed in terms of performance. However, the algorithm developed for this thesis is constrained by the fact that it was meant to monitor motion within a sea cage. Most of the concepts, algorithms and derivations from Chapter 2 is applicable if the algorithm were to be generalized. By generalization I mean that the algorithm will itself determine which triangles to use for positioning when parsing hydrophone logs. This can be done by calculating likely DOP coefficients in order to find the best triangle.

Also, receiver geometry is as per now hard-coded into the software. Should the receivers in the future know their own positions, this should be changed.

The algorithm in itself is purely mathematical, and there is not much one could do to reliably optimize position estimates within a sea cage. There is the possibility of using information from other than the two time differences of the outer triangle. Though as seen, the outer triangle will always perform the best in terms of DOP. But using one of the other triangles in a sea cage, might optimize the SNR values across the receiving nodes. Such an optimization is hard to do though, as the relation between the SNR values and the time differences has to be experimentally determined.

Chapter 8

Conclusions and future work

8.1 Positioning performance

The developed positioning system has proven to be a potent tool for investigating motion patterns of aquatic animals. In a controlled experiment for determining the system precision, it produced a stable set of position estimates. The output of the acoustic positioning system was accurate in the sense that the most likely estimate for each position tested, was within what you would expect, given the sources of error for the experiment. Most important to consider here is the errors stemming from sea cage movement and uncertainty in the GPS measurement.

The middle position, which was the one with the largest data set, had its GPS measured position taken at the same time as the receiver positions were measured. Thus negating the error source of sea cage movement. The most likely position estimates for that position were within the GPS measurement error of $\pm 0.6m$. (which translates to a rms time difference error of $\pm 0.8ms$)

A short form of the most important findings are listed below:

- The system is stable and precise for SNR values higher than than a mean of $30dB$, with more than 90% of the signals being timestamped within $\pm 1ms$. This corresponds to $\pm 1m$ to $\pm 2m$ within a sea cage, depending on the relative geometry between the transmitter and the receiver array.
- For optimizing the precision of the position estimates, one should take measures to limit the acoustic noise.
- The noise floor can be lowered by reducing the amount of tags in the water, and/or take use of other frequency channels.
- Different signal strengths across the receivers can yield statistically skewed position estimates, due to signal detection being delayed at weaker SNR values.

8.2 Surface support module and real-time monitoring

A network module has been incorporated and tested with the SSM hardware. Proposals for software implementation of this network module with the goal of low power consumption has been laid out. The SSM that was developed for my project report last fall has been extensively tested, and its performance has been thoroughly evaluated.

The challenge with incorporating the network module into the existing design is one of resource management. One can not access the GPS and the flash memory simultaneously due to the fact that they both uses the same SPI bus. This causes the system to not be able to poll GPS data when it unloads data to the server, as the server unloading is reliant on the flash memory. Implementation of a more sophisticated scheduler like for example freeRTOS[18], should be considered.

Important points regarding the current status of the SSM is listed below

- Time synchronization is constrained by the cycle time of the hydrophone and is thus optimal seen from the perspective of the SSM.
- Extensive testing has proven the time synchronization to be stable over long periods of time.
- Networking capabilities is added to the SSM hardware and the SSM is verified to have functioning time synchronization with the new hardware.
- Network incorporation to the SSM is feasible, and should not impose significant power consumption to the system.

8.3 Recommendations for future work

The prospect of creating a better positioning system for fish tracking is of course possible. But within the constraints imposed by the demand of low power consumption, one can actually achieve close-to-optimal precision from the system already developed. It is all a matter of controlling and monitoring the acoustic environment. If one is able to ensure high signal strength at all receivers, it is probably not worthwhile trying to optimize the system. However, when encountering low signal strengths there is room for improvement.

Especially the findings regarding biases in timestamping when the SNR difference between the receivers is high, needs to be addressed. Estimates of position errors, and their statistical bias towards higher or lower time differences should be possible to find experimentally.

The positioning software is at the moment constrained by the fact that it only handles a constellation of four hydrophones, where the triangle with best DOP properties has to be hard-coded into the algorithm. This should be generalized.

The SSMs should have their hardware upgraded with a more precise GPS module. This will enable the SSMs to hold their own positions that can be sent to the positioning software and thus further advancing the generalization of the system.

Concrete recommendations for further work is given below:

- Investigate methods for compensation in the timestamping method when encountering weak signals.
- If compensating for low SNR values is not possible, it should be investigated whether other modulation schemes can be used to optimize timestamping performance.
- Describe biased deviations in the position estimates in relation to SNR difference between the receiving nodes. This should be investigated experimentally.
- Finish software implementation of tinymesh into the SSM, and test extensively.
- Generalize the positioning software.

Bibliography

[1] "Thelma biotel website." http://www.thelmabiotel.com/. [Online, accessed 19-December-2015].

[2] Unknown, "Cross correlation pulse." https://en.wikipedia.org/wiki/File:CrossCorr_Pulse.png.

[3] B. Fang, "Simple solutions for hyperbolic and related position fixes," *IEEE Transactions on aerospace and electronic systems*, vol. 26, no. 5, pp. 748–749, 1989.

[4] V. Del Grosso, "New equations for the speed of sound in natural waters (with comaparisons to other equations)," 1974.

[5] G. Dudek and M. Jenkin, *Computational principles of mobile robotics.* Cambridge university press, 2000.

[6] G. Taraldsen, T. A. Reinen, and T. Berg, "The underwater gps problem," 2011. [Online, accessed 14-June-2016]: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6003649.

[7] N. Levanon, "Lowest gdop in 2-d scenarios," 2000. [Online, accessed 14-June-2016]: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=850811.

[8] M. Soltero, J. Zhang, and C. Cockril, "Rs-422 and rs-485 standards overview and system configurations." http://www.ti.com/lit/an/slla070d/slla070d.pdf. [Online, accessed 16-June-2016].

[9] "Tinymesh website." https://tiny-mesh.com/. [Online, accessed 10-June-2016].

[10] "Rc11xx(hp)/25xx(hp)-tm data sheet rev. 1.42." https://tiny-mesh.com/wireless-mesh-network/pdf/RCxxxx(HP)-TM_Data_Sheet_1_42.pdf. [Online, accessed 10-June-2016].

[11] "Atmel avr 32 data sheet." http://www.atmel.com/images/atmel-32145-32-bit-flash-mcu-uc10_summary.pdf. [Online, accessed 6-June-2016].

[12] u blox, "Neo-7p u-blox 7 precise point positioning gnss module data sheet." https://www.u-blox.com/sites/default/files/products/documents/NEO-7P_DataSheet_(UBX-13003787).pdf. [Online, accessed 13-December-2015].

[13] u blox, "u-blox 7 receiver description including protocol specification v14." https://www.u-blox.com/sites/default/files/products/documents/u-blox7-V14_ReceiverDescrProtSpec_(GPS.G7-SW-12001)_Public.pdf. [Online, accessed 13-December-2015].

[14] T. O. Group, "The open group base specifications issue 7." http://pubs.opengroup.org/onlinepubs/9699919799/basedefs/V1_chap04.html#tag_04_15. [Online, accessed 13-December-2015].

[15] D. DePriest, "Nmea data." http://www.gpsinformation.org/dale/nmea.htm. [Online, accessed 12-June-2016].

[16] "International vocabulary of metrology — basic and general concepts and associated terms (vim)," *JCGM 200*, 2008.

[17] "lla2flat matlab documentation." http://se.mathworks.com/help/aerotbx/ug/lla2flat.html. [Online, accessed 5-June-2016].

[18] "freertos website." http://www.freertos.org/. [Online, accessed 27-June-2016].

Appendix A

Expansion board

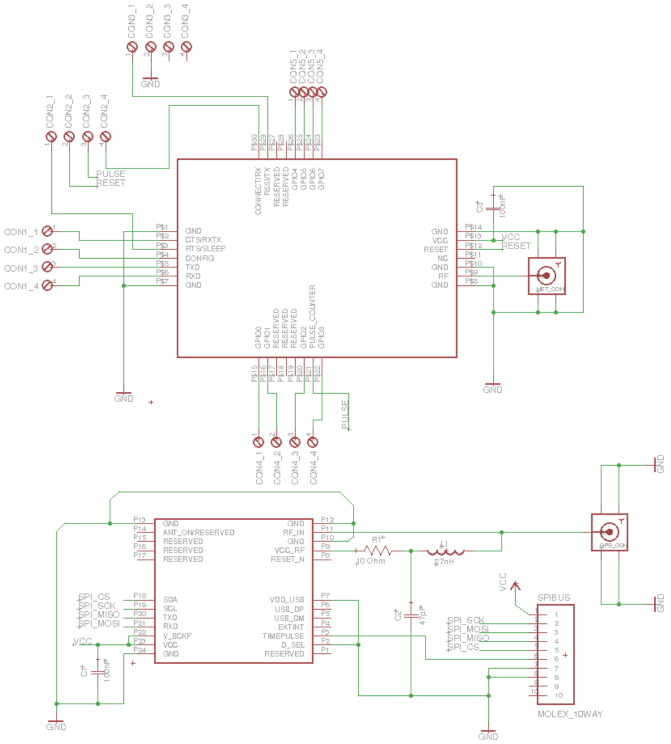


Figure A.1: Expansion board schematics

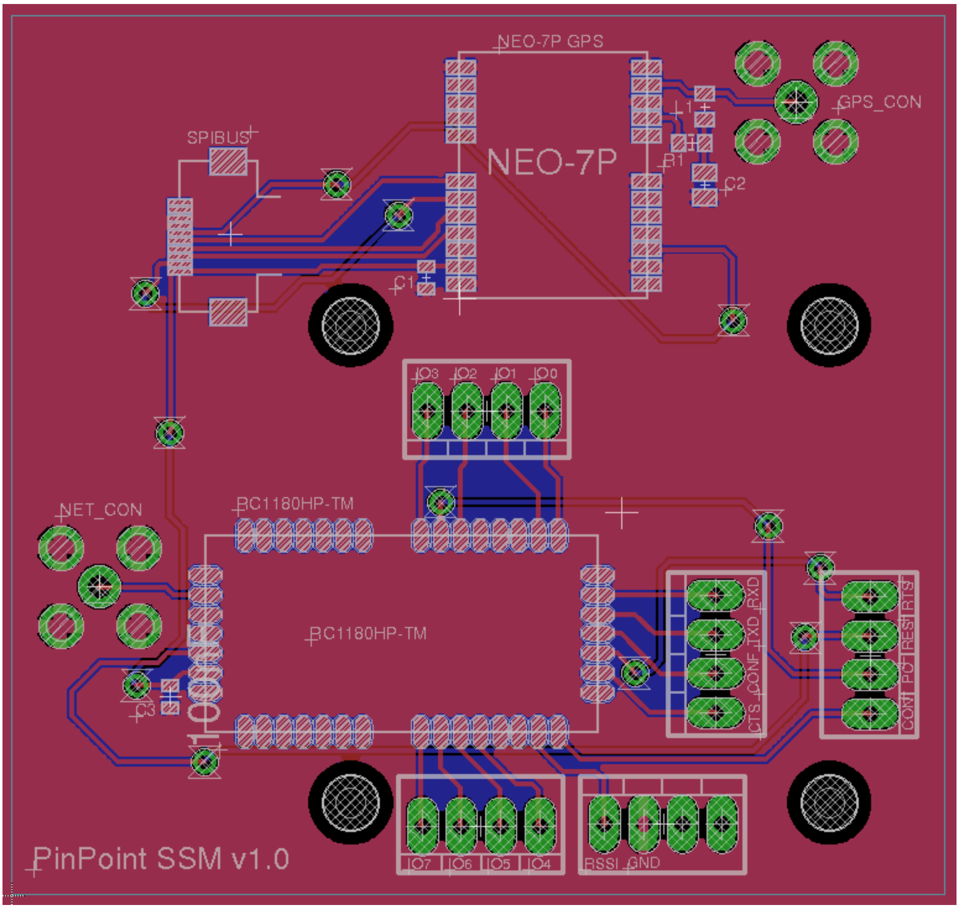


Figure A.2: Expansion board layout

Appendix B

DOP simulation

```
clear all;
c=1500;
rho=1;


%%%Triangle%%%
p1=[0, 0];
p2=[44.133, 0];
p3=[22.066, 38.22];
p4=[22.066, 12.74];


p=[p1,
    p2,
    p3];


step=0.3;
xmin=-7;
xmax=51;
ymin=-16;
ymax=43;



counter1=1;
for u=ymin:step:ymax;

    y=u;
    counter2=1;
    for v=xmin:step:xmax
        x=v;
        for i=1:length(p)
            xi=p(i,1);
            yi=p(i,2);
            zi=1;
            Ri=sqrt((xi-x)^2+(yi-y)^2);
            [(xi-x)/Ri (yi-i)/Ri];
            M(i,:) = [(xi-x)/Ri (yi-y)/Ri 1];
            sqrt(M(i,1)^2+M(i,2)^2);
        end
        kd=sqrt(trace(inv(transpose(M)*M)));
        if(kd<30) %Saturation to prevent contourf(..) from crashing
            dopx(counter2)=kd;
        else
            dopx(counter2)=30;
        end
        counter2=counter2+1;
    end
    doptot(counter1,:)=dopx;
    counter1=counter1+1;
end

x=[xmin:step:xmax];
y=[ymin:step:ymax];
figure(1); clf(1);
hold on;

[C, h]=contourf(x,y,doptot);
h.LevelStep=0.1; %Adjust step size for color axis
colorbar;
caxis([1,4]); %cap loss  values at 20 for more readable scale
colormap(jet); %Color ranges from blue to red

%%%%CIRCLE%%%%
r=25.4777;
x_offset=44.133/2;
y_offset=12.74;
th = 0:pi/100:2*pi;
xunit = r*cos(th)+x_offset;
yunit = r*sin(th)+y_offset;
plot(xunit, yunit, ’--g’, ’linewidth’, 3);

%%%Scatter plot of receiver nodes%%%%
```

```
scatter(p1(1),p1(2),'filled', 'red');
scatter(p2(1),p2(2),'filled', 'red');
scatter(p3(1),p3(2),'filled', 'red');
scatter(p4(1),p4(2),'filled', 'red');
```

Appendix C

Python implementation of web server

```
from socket import *
serverSocket = socket(AF_INET, SOCK_STREAM)
#Prepare a sever socket
serverSocket.bind(('', 4083))
serverSocket.listen(1)

while True:
    #Establish the connection
    print 'Ready to serve...'
    connectionSocket, addr = serverSocket.accept()
    try:
        message = connectionSocket.recv(1024)
        filename = message.split()[1]
        f = open(filename[1:])
        outputdata = f.read()
        connectionSocket.send("\nHTTP/1.1 200 OK")
        connectionSocket.send("\ntext/html\n\n")
        for i in range(0, len(outputdata)):
            connectionSocket.send(outputdata[i])
        connectionSocket.close()
    except IOError:
        connectionSocket.send("\nHTTP/1.1 404 Not Found\n")
        connectionSocket.close()
serverSocket.close()
```

Appendix D

Plots for experimental timestamping mechanism

D.1 Set up

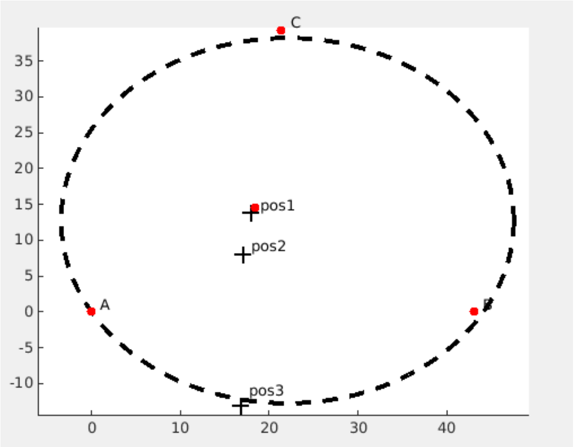


Figure D.1: Experiment set up for the second phase

Position	Time (minutes)	Position data points
1	17	55
2	16	54
3	15	44

Table D.1: Data generated for each position

D.2 Position 1

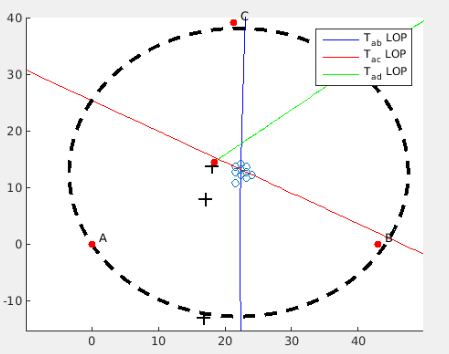


Figure D.2: Position data

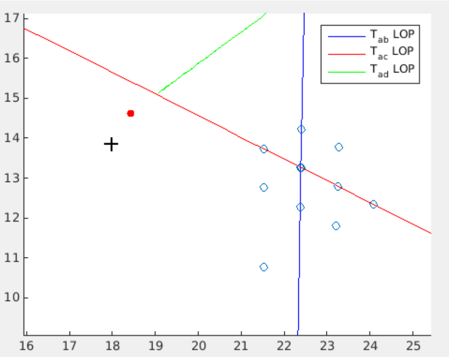


Figure D.3: Position data zoomed

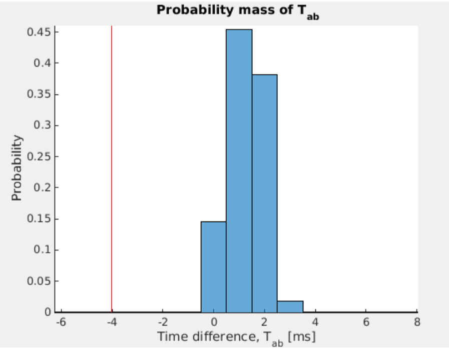


Figure D.4: Time difference of arrival between node A and B

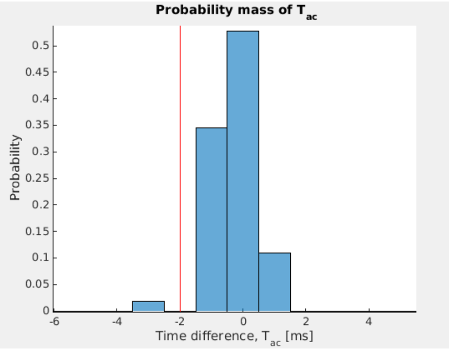


Figure D.5: Time difference of arrival between node A and C

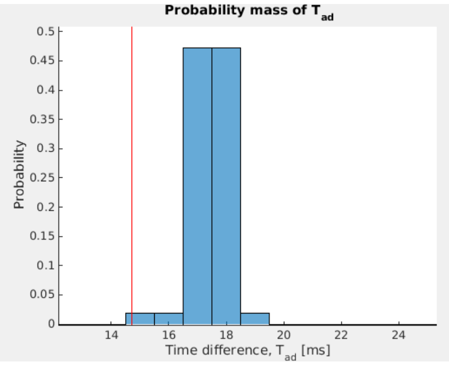


Figure D.6: Time difference of arrival between node A and D

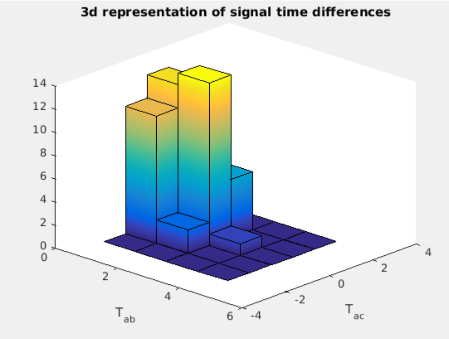


Figure D.7: Bivariate histogram of time differences

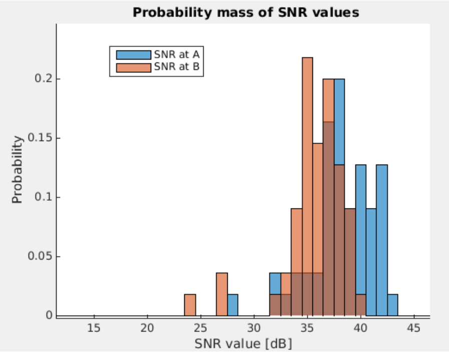


Figure D.8: SNR at receivers

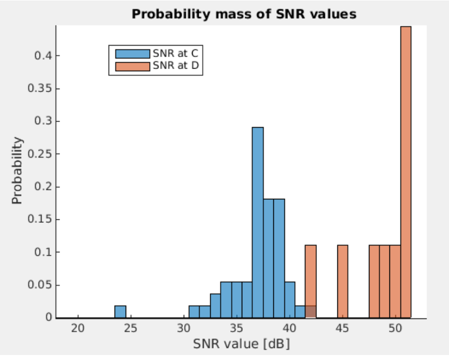


Figure D.9: SNR at receivers

D.3 Position 2

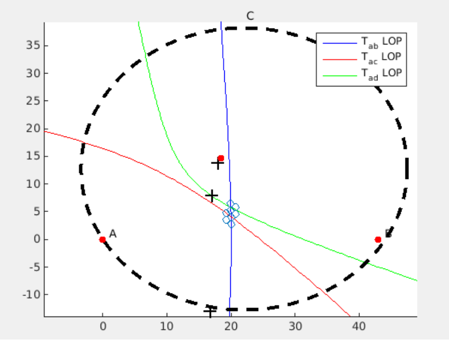


Figure D.10: Position data

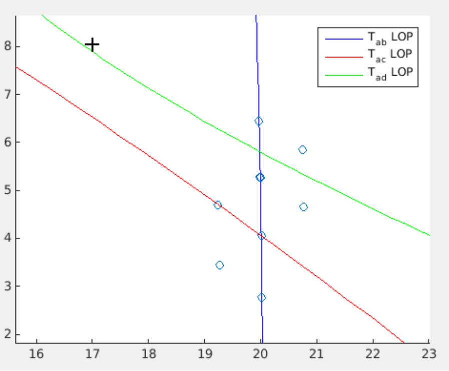


Figure D.11: Position data zoomed

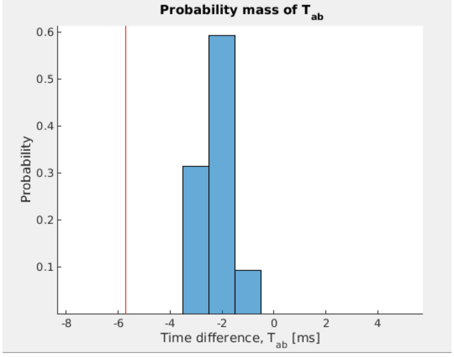


Figure D.12: Time difference of arrival between node A and B

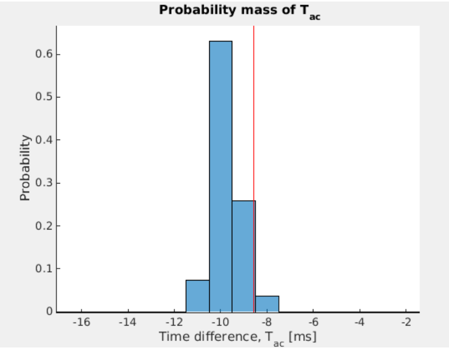


Figure D.13: Time difference of arrival between node A and C

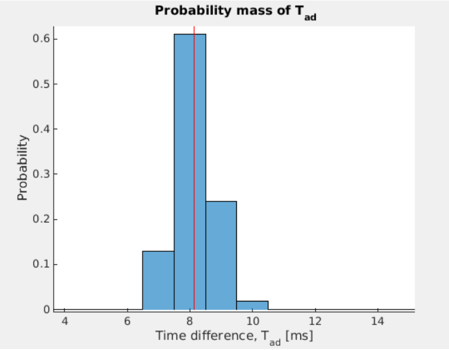


Figure D.14: Time difference of arrival between node A and D

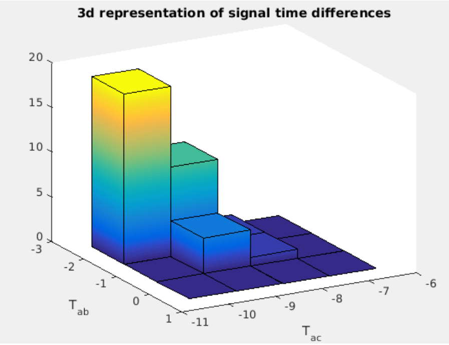


Figure D.15: Bivariate histogram of time differences

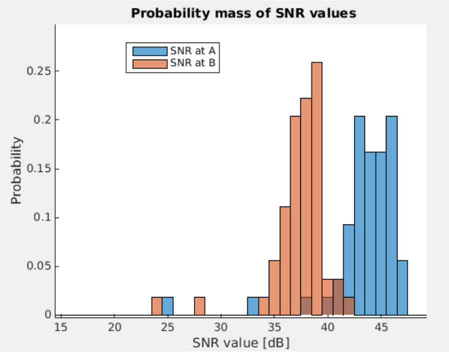


Figure D.16: SNR at receivers

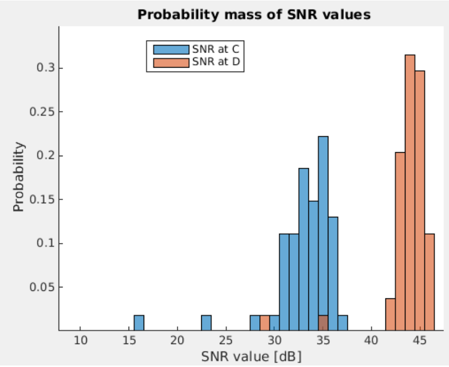


Figure D.17: SNR at receivers

D.4 Position 3

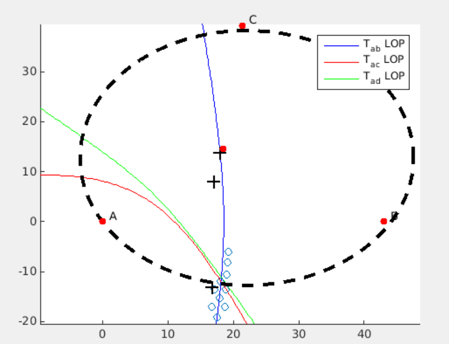


Figure D.18: Position data

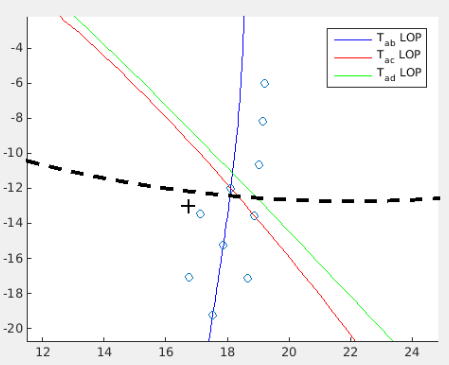


Figure D.19: Position data zoomed

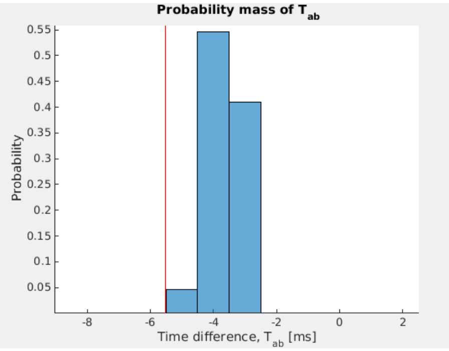


Figure D.20: Time difference of arrival between node A and B

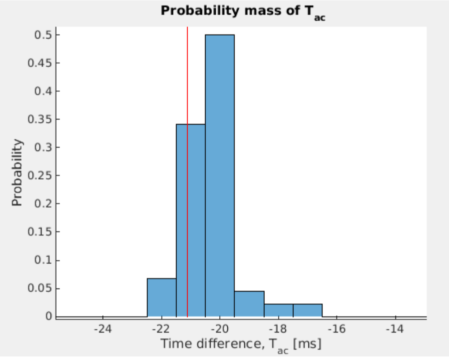


Figure D.21: Time difference of arrival between node A and C

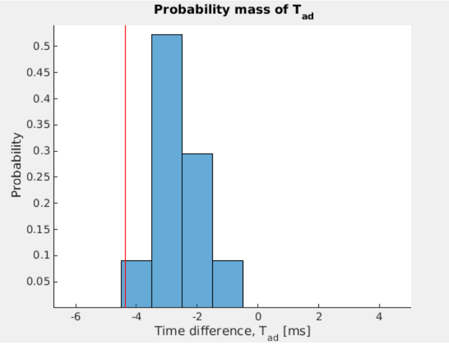


Figure D.22: Time difference of arrival between node A and D

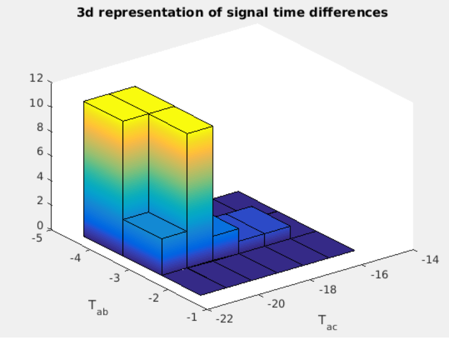


Figure D.23: Bivariate histogram of time differences

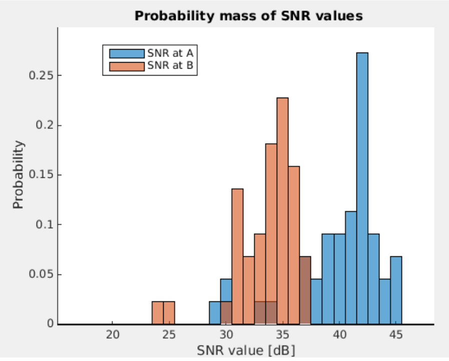


Figure D.24: SNR at receivers

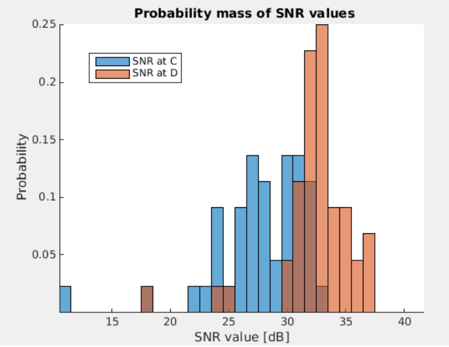


Figure D.25: SNR at receivers