Examining Use of Various Unmanned Vehicles in Single Fish Tracking

Specialization Project in Marine Control Systems

By August Ekanger

Supervisors:
Tor Arne Johansen
Martin Ludvigsen

December 2017
Trondheim, Norway

NTNU
Norwegian University of Science and Technology

Abstract

This paper examine tracking of aquatic animals tagged with a transmitter using a system of unmanned vehicles equipped with acoustic receivers. The robustness of a TDOA localization estimator is directly dependant on the precision of the receiver clocks and receiver position estimates. Other aspects like acoustic noise, communication delays and dilution of precision all influence the performance of the algorithm. An unmanned tracking system needs accurate target position estimates in order to keep its vehicles within transmission range of the tagged animal. These aspects stresses the importance of picking vehicles which optimizes observer performance. This paper aims to look into advantages and disadvantages associated with the use of USV, AUV, AUV glider, and UAV as host-vehicles for an acoustic receiver in an unmanned system for fish tracking. A formation of AUVs or AUV gliders are both able to provide very low geometric dilution of precision values through forming a spherical formation surrounding the target and propagating in evenly shifted sinusoidal patterns, respectively. However, for practical purposes, are positioning, clock synchronization and communication major issues associated with the use of AUVs. These error sources are expected to be less significant for a system of USVs and UAVs as GNSS positioning and radio communication is possible on the surface.

With emphasis on the benefits of using an UAV as an assisting vehicle to a formation of four USVs, a case study with simulation results is presented. A variety of unforeseen events could lead to demanding tracking conditions and potential target loss, e.g predatory-prey interactions. This motivates to develop some kind of support system which would become active and enable sufficient tracking ability during these critical phases of the operation. The results presented in this paper indicate that an unmanned system consisting of four USVs equipped with a UAV have significantly better tracking ability compared to the same system without the UAV. An extended Kalman filter, a formation control algorithm and a situation awareness hierarchy was implemented in an attempt to optimize tracking ability of the system.

Preface

The Department of Engineering Cybernetics at the Norwegian University of Science and Technology (NTNU) have recently received funding for a fish tracking project involving the Atlantic salmon *Salmo salar*. The current outline of the project is to track a fish tagged with an acoustic transmitter by utilizing a formation of four USVs equipped with hydrophones. This paper was written as a part of this project, and as a pre-diploma thesis in the NTNU course "TMR4510 - Marine Control Systems, Specialization Project". The idea of using an Unmanned Aerial Vehicle as a platform and assisting vehicle in a receiver array was introduced to me by professor Tor Arne Johansen.

My desire to explore and understand more about the behavior of fish and mammals served as the main motivation throughout this project. I believe aquatic telemetry is important because it gives us the ability to understand how human life impacts the ocean environment and wildlife. This author have not successfully found any literature of research suggesting use of UAV as a platform for acoustic signal acquisition. My hope is that this work could result in a proof-of-concept experiment.

I want to thank professor Tor Arne Johansen and professor Martin Ludvigsen for supervising the project. I also want to thank Artur Piotr Zolich and Praveen Jain for their assistance. They are both PhD candidates working on related topics within the field of aquatic telemetry at NTNU and University of Porto, respectively. Finally, I want to thank my good friend Axel Berggraf Egenæs for creating the front page of this paper.

Table of Contents

List of Tables

List of Figures

Abbreviations

TDOA	-	Time Difference of Arrival
TOA	-	Time Of Arrival
DOP	-	Dilution Of Precision
GNSS	-	Global Navigations Satellite System
USV	-	Unmanned Surface Vehicle
UAV	-	Unmanned Aerial Vehicle
AUV	-	Autonomous Underwater Vehicle
US	-	Unmanned Systems
LOS	-	Line-of-sight
MEMS	-	Microelectromechanical systems
AIS	-	Automatic Identification System
CTD	-	Instrument used to measure conductivity, temperature, and pressure
ADCP	-	Acoustic Doppler current profiler
RTK	-	Real Time Kinematic, satellite navigation
Ad hoc	-	Approximated value
In situ	-	On site
Et al	-	And others (authors)
E.g.	-	Exempli gratia, which translates to for example

Chapter 1

Introduction

1.1 Aquatic telemetry

In a report from 1987 by NASA, telemetry is defined to be the process of ”reliably and transparently convey measurement information from a remotely located data generating source to users located in space or on Earth.”[1] The word telemetry is derived from two words with Greek origin: tele and metron, which translates to remote and measure, respectively. Aquatic telemetry is the process of collecting measurements in a remote marine environment. Aquatic animals are defined to be the animals which are living solely or chiefly in water.

This part is mainly based upon the 2015 science magazine article *”Aquatic animal telemetry: A panoramic window into the underwater world”* by N. E. Hussey et al.[3] As pointed out by this article, has aquatic telemetry transformed our ability to observe and measure the behaviour of animals living in water environments. Recent technological advances in ultra-low powered electronic systems, digital signal processing and MEMS sensors offers new and exciting possibilities within the field of aquatic telemetry.[2] Telemetry data coupled with genetics, biochemical tracers and biologgers provides innovative opportunities and insight into behaviour of marine species. As radio waves lack the ability to penetrate water, the field of aquatic telemetry have been divided into two principle approaches; satellite and acoustic telemetry.

Satellite telemetry is suitable for tracking a variety of animals ranging from fish to marine mammals and reptiles. It is complementary to acoustic telemetry in the sense that a GNSS array of transmitters provides the data necessary to determine the animal’s position, and therefore only a single receiver is needed. Satellite tags can transmit real-time position estimates and the data logged during dive each time the tagged animal surfaces. Satellite tags are, however, unable to determine the horizontal position of animals during their dives. The size of the tags limit the possibility of tracking smaller fish and mammals.

Acoustic telemetry is based on the principle of using hydrophones to record the transmitted signal from a tagged animal. Acoustic transmitter tags (also known as fish tags), originating from the aquaculture industry, are used to monitor the behavior of fish. These tags are able to measure a variety of variables such as temperature, depth, acceleration, dissolved oxygen, salinity and tilt/inclination. The tags are able to both store and transmit the data. A tag which is solely able to store data is often referred to as a biologger. Fish tags transmit data through an acoustic signal. One or several acoustic receivers, also known as hydrophones, can be used to gather the signal. It is possible to estimate the horizontal location of the fish tag using time of arrival measurements. Time of arrival is defined as the time an acoustic receiver obtains the signal transmitted by the fish tag. A large variety of position estimation techniques exist, but due to the difficulties introduced by the harsh environment of underwater communication channels, TDOA localization methods are recognized as the most rigorous. Acoustic telemetry is divided into two primary approaches differentiated by how the array of receivers are oriented: 1) receivers moored at fixed locations, and; 2) mobile receivers.

Moored receivers are applicable for tracking fish in closed environments, e.g. farm cages in marine aquaculture. Fixed receiver networks have also proven useful in open environments to measure predictable behaviour of animals such as the Atlantic salmon *Salmo Salar*. In an experiment performed in the river *Lærdalselva*, in Western Norway, the differences in behavioural patterns and migration timing between wild and hatchery-reared salmon smolts was examined using a receiver network of this kind.[4] A downside of moored receivers, however, is their inability to reconstruct detailed behaviour pattern for fish migrating over large ocean areas.

In recent years, acoustic receivers have been attached to marine vessels, forming a receiver network able to detect and track tagged animals. These networks are able provide detailed information about the location of the animal also during dives. In august 2011, a tagged leopard shark was tracked by an AUV equipped with two acoustic receivers in the SeaPlane Lagoon in Los Angeles, California.[17] A field experiment in Agdenes, Norway, took place in September 2016 proving a concept of USVs tracking a target AUV equipped with an acoustic transmitter.[2] A study conducted in southern Portugal involved satellite tags in addition to underwater and surface robotic vehicles to measure both the movements and the contextual environment of an Ocean Sun fish *Mola mola*.[7] The study reports ”near-real time monitoring of finescale ($<$ 10 m) behaviour of sunfish”. An AUV was used for video recording of the tracked fish, and a *WaveGlider* USV was used for measuring the contextual environment of the fish. The *WaveGlider* was equipped with a large variety of sensors, including an ADCP, a CTD and an AIS receiver. The project serves a perfect example of how several types of unmanned vehicles can collaborate in obtaining data in an open sea environment.

Another approach, which does not fit under the category satellite nor acoustic telemetry, is the method of estimating target position of a radio transmitter tagged animal based upon signal strength measurements. A study from 2013 suggests using UAVs equipped with radio receivers for fish tracking by using fractional order potential fields and Kalman

filtering.[6] The simulation results showed that localization of fish was possible using two or three UAVs. Tracking methods of this kind are only applicable for fish living close to the surface, as the transmission loss of the electromagnetic waves through water is significant.

1.2 Single fish tracking

A large variety of tracking scenarios can be envisioned, e.g. tracking a pelagic fish along the shore or a coral reef fish in a tropic environment. The behavior of fish is dependent upon many variables such as species, age and surrounding environment. Different constellations and compositions of unmanned vehicles can be configured in order to create an optimal tracking system. The unmanned system should be designed with respect to characteristics of the animal it is supposed to track and examine.

If the target is lost in an open sea environment it is likely to never be detected again. Hypothetically if the target was tagged with a satellite receiver in addition to the acoustic transmitter the odds of reallocating the target would be higher. As the very intention of many acoustic tracking systems is to examine the behavior of species of which satellite telemetry is not applicable for, this is often not the case.

A variety of unforeseen events could lead to demanding tracking conditions and potential target loss. Some examples being predator-prey interactions involving the target fish, sudden malfunction in one of the receivers or increased acoustic noise in the surrounding environment. An unmanned tracking system might be able to track a target with sufficient precision for most of the operation only to experience sudden target loss due to these unforeseen events. This motivates to develop some kind of supporting system which would become active as these unforeseen events occur and the tracking ability of the system is endangered. The support system would ideally only be active during these critical phases of the operation in order to save both energy and cost.

1.3 Research question

Look into advantages and disadvantages associated with the use of different vehicles in a unmanned system for single fish tracking, with emphasis on the benefits of using an Unmanned Aerial Vehicle as an assisting vehicle to a group of Unmanned Surface Vehicles.

1.4 Article structure

The overall structure of the article is presented in this section.

In the second chapter of this paper, the basic theory of TDOA and TOA localization methods are presented. This theory forms a basis of which the estimator and formation control is built. This chapter aims to illustrate how many TOA measurements that are needed to obtain distinct solutions in $\mathbb{R}_2$ and $\mathbb{R}_3$, geometrically.

The third chapter aims to provide some insights regarding receiver platforms and vehicle configurations in an unmanned system for single fish tracking. The advantages and disadvantages associated with use of USVs, AUVs, AUV gliders and UAVs are discussed. Geometric dilution of precision (GDOP) related to the receiver constellation, and how it effects the accuracy of the estimator, is discussed towards the end of the chapter. Two suggested system constellations are presented with corresponding GDOP calculations. The first example aim to highlight the importance of having a depth measurements in a coplanar receiver array. The second example aim to highlight AUV gliders' ability to form receiver constellations associated with low geometric dilution of precision values.

The project of which the Department of Engineering Cybernetics has received funding for is likely to involve a vehicle setup of four *AutoNaut* USVs. As a variety of unforeseen events could lead to demanding tracking conditions and potential target loss for the *AutoNauts*, a case study examining the possibility of using a UAV as a support vehicle is presented in the fourth chapter. An extended Kalman filter algorithm was implemented providing estimated solutions to the time difference of arrival equations. A formation control algorithm and a situation awareness hierarchy was implemented in an attempt to optimize tacking performance.

The results are discussed in the final chapter. Some suggestions for further research and conclusions are provided towards the end of the chapter.

Chapter 2

Target localization

2.1 Intro

A pseudorange is an estimate of the distance between a receiver and a transmitter. The time of transmission (TOT) is the time it takes for a signal to propagate from the transmitter to the receiver. In GNSS, the electromagnetic signals are time-stamped utilizing an atomic clocks aboard the satellites. While a major source of error in GNSS localization is clock drift, this can be estimated and accounted for through a variety of methods.

As GNSS signals are unable to penetrate water other localization methods are needed for tracking of submerged fish. Methods utilizing acoustic signals are widely used in underwater operations. Atomic clocks are large, expensive and not suitable for fish tags. The time of when an acoustic signal is sent from a fish tag is therefore unknown. Time difference of arrival (TDOA) is a localization method which can be used in these situations. By subtracting two time of arrival (TOA) measurements is the unknown term eliminated from the equation. Compared with TOA methods do TDOA methods use measured time difference rather than the absolute time of arrival. The positions of the receivers obtaining the TOA measurements must be known with sufficient precision as they appear in the target positioning equations. Normally, the TOA of one receiver is used as a reference for comparison with measurements from other receivers. The number of TDOA equations is equal to one less than the number of TOA measurements as one measurement is needed as a reference. TDOA methods therefore require an additional TOA measurement compared to TOA localization methods. As for both TDOA and TOA localization methods are the corresponding equations quadratic of nature. An additional equation with regard to the number of dimensions of which one wants to determine a distinct solution in is therefore needed for both methods. There exist various methods for obtaining distinct solutions without this extra receiver, but in certain receiver constellations the methods fail.

A major difference in TOA and TDOA localization methods is the nature of the solutions. In TOA localization methods is the solution found as the intersections of circles in

$\mathbb{R}^2$ and spheres in $\mathbb{R}^3$. In GNSS, several pseudorange measurements correspond to a set of spherical solutions, with the GNSS receiver position in the intersection of their surfaces. A TDOA measurement corresponds to the difference of two pseudorange measurements divided by the speed of the signal. Each TDOA equation correspond to a hyperbolic curve in $\mathbb{R}^2$ or hyperbolic surface in $\mathbb{R}^3$, with the target located in the intersection of these curves or surfaces.

2.2 TDOA localization

2.2.1 Solutions in $\mathbb{R}^2$

Let R1, R2, R3 and R4 be four receivers in two-dimensional plane: $\mathbf{x}_{R_j} = \begin{bmatrix} x_{R_j} \\ y_{R_j} \end{bmatrix} \in \mathbb{R}^2$ for j = 1,2,3,4, and let a target fish at an unknown target be: $\mathbf{x}_t = \begin{bmatrix} x_t \\ y_t \end{bmatrix} \in \mathbb{R}^2$. Let the target emit an acoustic signal with propagation speed c. By assigning the TOA measurement obtained by the receiver in position $\mathbf{x}_{R_1}$ to be the reference, three TDOA measurements are obtained:

$$\text{TDOA}_{j-1} = \text{TOA}_j - \text{TOA}_1 \tag{2.1}$$

for j = 2,3,4. By utilizing the linear relationship between speed, time and position, the resulting equations become:

$$\begin{aligned} \text{TDOA}_{j-1}\, \text{c} = &\sqrt{(x_{R_j} - x_t)^2 + (y_{R_j} - y_t)^2} \\ &- \sqrt{(x_{R_1} - x_t)^2 + (y_{R_1} - y_t)^2} \end{aligned} \tag{2.2}$$

for j = 2,3,4.

Four receivers are needed to determine the distinct solution of a transmitter in $\mathbb{R}^2$. Let R1, R2, R3 and R4 be four receivers in point locations (0,1), (7,1), (2,5) and (5,4), respectively. Let the position of the transmitter be (3,3). By assigning R1 to be the reference receiver, three TDOA equations are obtained. Each of the three equations have two y(x) solutions because of the quadratic nature of the equations. Hence, 6 y(x) solutions in $\mathbb{R}^2$ are obtained. The distinct solution is given in the intersection of three hyperbolas in (3,3) in figure 2.1. The corresponding *MATLAB* script is provided in the appendix section D.1.

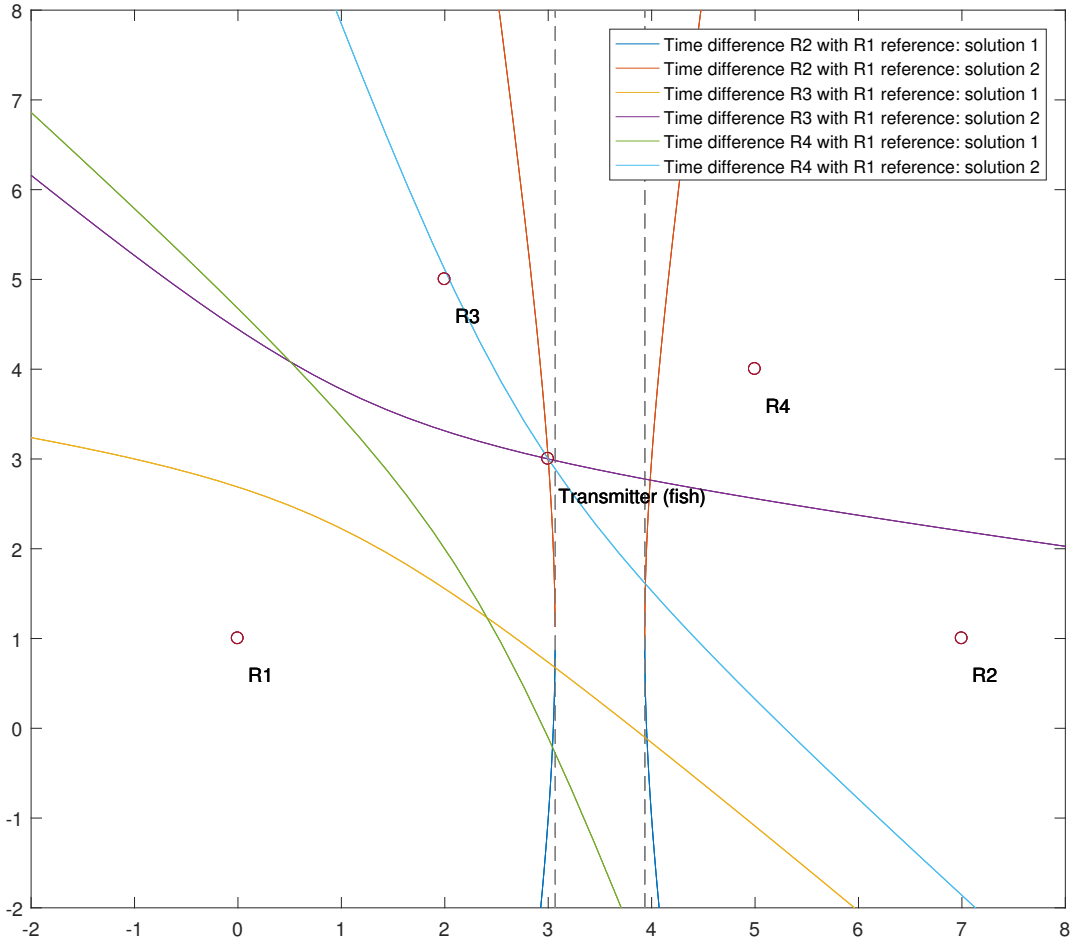

Figure 2.1: Solutions in $\mathbb{R}^2$

2.2.2 Solutions in $\mathbb{R}^3$

Let R1, R2, R3 and R4 be four receivers in three-dimensional space: $\mathbf{x}_{R_j} = \begin{bmatrix} x_{R_j} \\ y_{R_j} \\ z_{R_j} \end{bmatrix} \in \mathbb{R}^3$ for j = 1,2,3,4. Let the position of target fish at an unknown location be: $\mathbf{x}_t = \begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix} \in \mathbb{R}^3$. Given similar constraints as in in equation 2.1, the following TDOA equations are obtained:

$$\begin{aligned} \text{TDOA}_{j-1}\, \text{c} = &\sqrt{(x_{R_j} - x_t)^2 + (y_{R_j} - y_t)^2 + (z_{R_j} - z_t)^2} \\ &- \sqrt{(x_{R_1} - x_t)^2 + (y_{R_1} - y_t)^2 + (z_{R_1} - z_t)^2} \end{aligned} \tag{2.3}$$

for j = 2,3,4.

Three TDOA equations are obtained, each of which have two solutions due to the quadratic nature of the equation. In three dimensions, the solutions of the TDOA equations correspond to hyperbolic surfaces in $\mathbb{R}^3$. Figure 2.2 illustrates four receivers and an unknown target marked with dots colored blue and red, respectively. The corresponding *MATLAB* script is provided in the appendix section D.2. The solutions of the TDOA equations correspond to 3 hyperbolic surfaces in yellow color. A sinusoidal surface plot with mean value 0 in yellow color was included to illustrate a fictive ocean surface. The target location of the fish is given as the intersection of the three TDOA surface solutions.

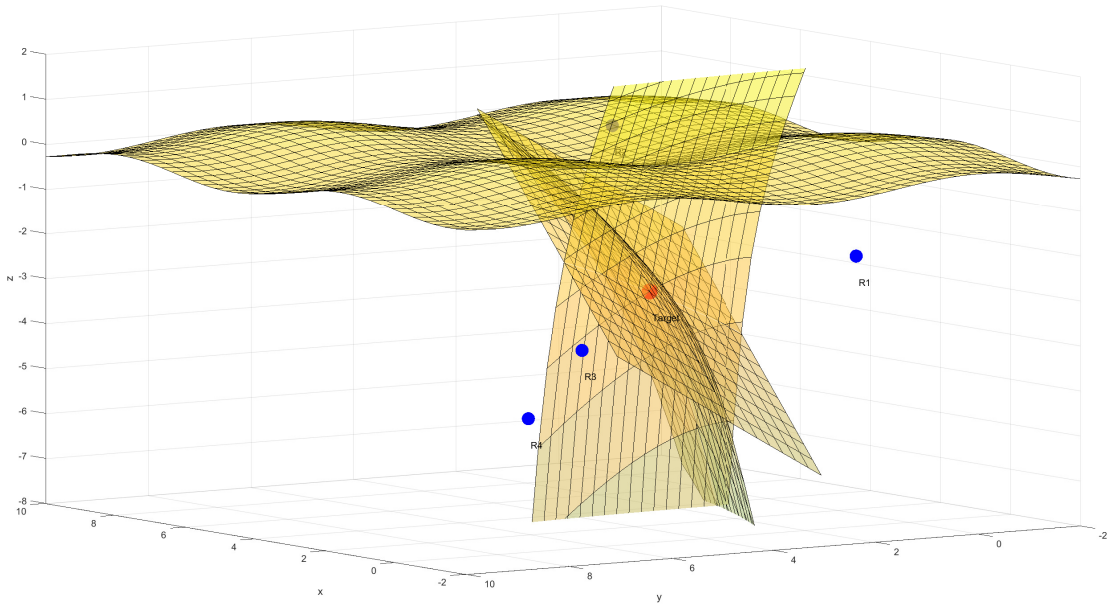

Figure 2.2: Solutions of TDOA equations in $\mathbb{R}^3$ with four acoustic receivers

2.2.3 Solutions in $\mathbb{R}^3$ with a depth measurement

When a depth measurement is available, one less receiver is needed to locate a target fish in three-dimensional space. Given a system equipped with three receivers rather than four, three TOA measurements would result in two TDOA equations. A constellation of three receivers with a depth measurement, and the corresponding solutions surfaces is illustrated in figure 2.3. The corresponding *MATLAB* script is provided in the appendix section D.3. The depth measurement indicates a horizontal plane in which the solution must be within. The solution and location of the target is found as the intersection between two TDOA hyperbolic surface solutions and the horizontal plane.

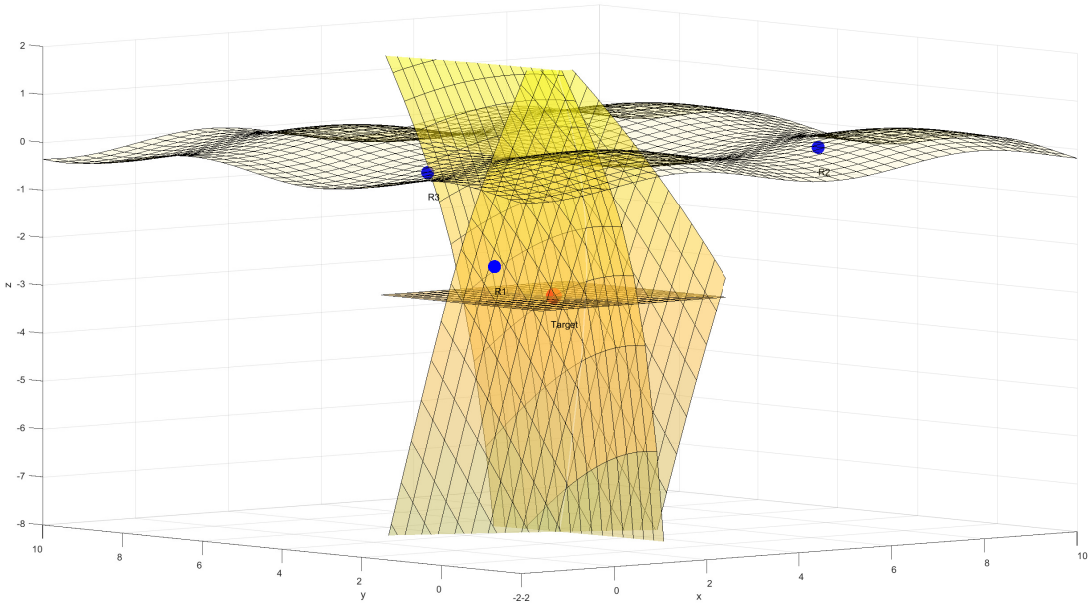

Figure 2.3: Solutions of TDOA equations in $\mathbb{R}^3$ when a depth measurement is available

2.2.4 Noise amplification

Noise is always present in acoustic signals and therefore also present in the distances derived from TDOA measurements. Let y_1 and y_2 and be two TOA measurements:

$$y_1 = \frac{1}{\mathrm{c}}||\mathbf{x}_{R_1} - \mathbf{x}_t|| + w_1 \tag{2.4}$$

$$y_2 = \frac{1}{\mathrm{c}}||\mathbf{x}_{R_2} - \mathbf{x}_t|| + w_2 \tag{2.5}$$

where w_1 and w_2 are zero-mean Gaussian white noise, $\mathbf{x}_{R_1}$ and $\mathbf{x}_{R_2}$ are the positions of two acoustic receivers and $\mathbf{x}_t$ is the position of the target. The resulting TDOA measurement becomes:

$$\begin{aligned} \mathrm{TOA}_{2-1} &= \mathrm{TOA}_2 - \mathrm{TOA}_1 \\ &= (y_2 - y_1) \\ &= \frac{1}{\mathrm{c}}(||\mathbf{x}_{R_2} - \mathbf{x}_t|| - ||\mathbf{x}_{R_1} - \mathbf{x}_t||) + (w_2 - w_1) \end{aligned} \tag{2.6}$$

In the process of differentiating TOA measurements, the mean value of the noise term $(w_2 - w_1)$ increase by a factor $\sqrt{2}$, and the variance increase by a factor of 2. This is one of the disadvantages of utilizing TDOA localization methods. Note that errors in TDOA measurements are scaled with the speed of sound when used in a positioning algorithm.

2.3 Single measurement localization

In some situations, only one receiver might be able to acquire the acoustic signal transmitted by a fish tag. As the principle of TDOA is based upon having at least two TOA measurements, the method is not applicable in this situation. However, a positioning technique based upon signal strength can be applied.

Let P_r and P_t denote the signal strength of the received and transmitted signal, respectively. Let r denote the euclidean distance from the receiver to the transmitter, and λ denote the wavelength of the signal. The path loss in free space of an electromagnetic signal is given as the dB-ratio of the relationship $\frac{P_r}{P_t}$. The path loss in free space is proportional to λ and r through the following relationship $\frac{P_r}{P_t} \propto \frac{\lambda^2}{16\pi^2 r^2} \implies 10\log_{10}(\frac{P_r}{P_t}) \propto 10\log_{10}(\frac{\lambda}{4\pi r})^2$. Let the signal strength loss in decibels be donated by L:

$$L \propto 20\log_{10}(\frac{4\pi r}{\lambda}) \tag{2.7}$$

By assuming that the acoustic signal transmitted by a fish tag decays as electromagnetic signals do in free space, and assume the receiver and transmitter gain to be equal to one,

a pseudorange estimate $\hat{\rho}$ of the distance between an acoustic receiver and a target fish is given as

$$\hat{\rho} = \frac{\lambda}{4\pi} 10^{\frac{L}{20}} \tag{2.8}$$

where λ is the wavelength of the acoustic signal transmitted by the fish tag. The wavelength of the acoustic signal is given through the following relationship $\lambda = \frac{c}{f}$. Where c is the speed of sound in water and f is signal frequency.

Imagine a situation where a single acoustic receiver on a USV or UAV platform is the only node within transmission range of the acoustic transmitter. Assume the received depth measurement to be perfect, and assume the pseudorange measurement derived from received signal strength is perfect. In this situation, the system is able to determine the location of the target to be within a circle set of solutions in $\mathbb{R}^3$. In figure 2.4, the yellow

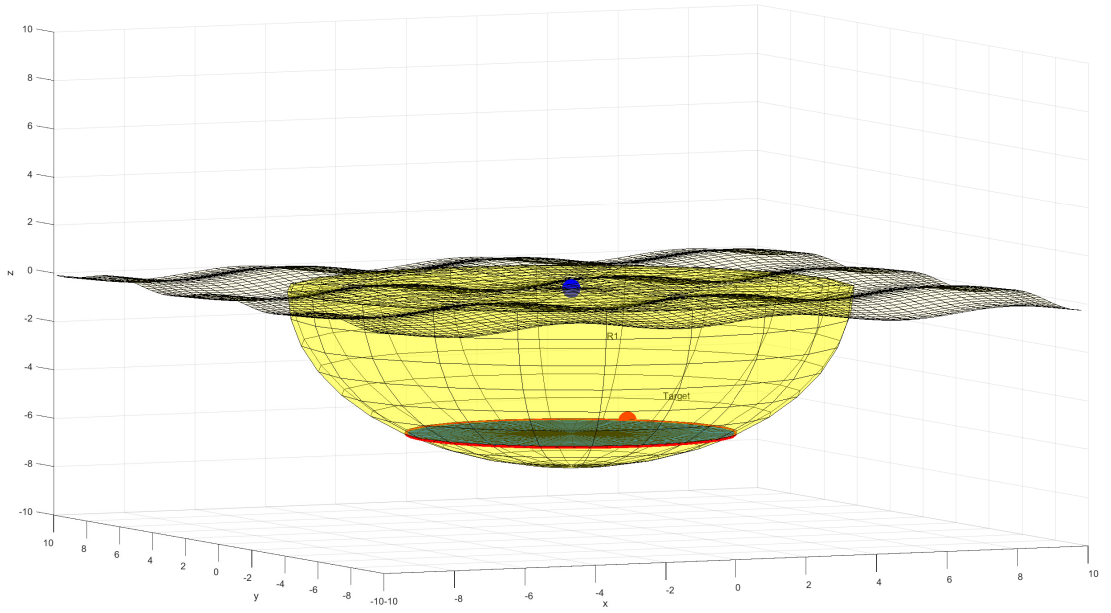

Figure 2.4: One receiver

dome represents a spherical solution set derived by the pseudorange. The corresponding *MATLAB* script is provided in the appendix section D.4. As the fish is not able to be above the surface, locations on the bottom dome are the only ones within the valid solution set. A horizontal plane determines the location of the target in depth direction. The intersecting solution circle set of the dome and horizontal plane, in which the fish must be located within, is indicated with red color in 2.4. The true target and receiver position is indicated by a red and blue dot, respectively.

To estimate pseudoranges based upon signal strength is difficult. The transmission channel in water is constantly changing in time and space, and non linear relationships furthermore complicate the calculations. Imagine a system of unmanned vehicles equipped with acoustic receivers tracking a fish. Through having obtained TDOA measurements for some time, a model of measured signal strength as a function of range can be derived. If suddenly an unforeseen event occur, and only a single receiver is within transmission range, this model could provide a pseudorange estimate based upon the measured signal strength. The model and its parameters would have to be tuned real time as signal noise in the surrounding environment of the system varies throughout operation.

In general is there more uncertainty involved with using a signal strength based technique in comparison with TDOA localization. However, combinations of utilizing both techniques could potentially increase the robustness of the estimator and result in enhanced tracking performance for the system.

Chapter 3

Receiver platforms and vehicle configurations

A variety of different autonomous vehicles and constellations can be applied in a drone tracking system. This chapter aims to provide some insight in the advantages and disadvantages related to each of the platforms. Characteristics like communication, maneuverability, speed and endurance are discussed briefly.

3.1 USV

A receiver array of co-planar receivers would induce geometric dilution of precision errors in the direction of the orthogonal projection in the plane. As a constellation consisting solely of USVs is bound to the surface plane, bad tracking performance is to be expected unless a pressure measurement or TOA measurement from an assisting underwater vehicle is available.

Intermittent and self noise levels are expected to be high for acoustic hydrophones on a USV platform. The book *Autonomous Underwater Vehicles* states three major contributors to noise underwater *"ambient or background noise of the ocean; self noise of the vehicle; and intermittent noise including biological noises such as snapping shrimp, ice cracking and rain."* [8] In harsh weather conditions sound disturbance from breaking waves, rain and hull slamming could potentially influence the localization accuracy. In a study performed at NTNU the transmission range of the acoustic signal was lower than expected and previously observed, with the reason assumed to be noise from an intensive rain shower.[2] Results from the same study showed that the reception of an acoustic receiver on a USV platform is significantly influenced by the depth of the submerged hydrophone, indicating better performance with increased depth. However, dragging a hydrophone with a long line would induce drag forces and corrupt the hydrodynamic properties of the vessel. A study from 1982 [9] indicated that self noise from cavitation effects on the propeller is

significant for higher frequencies. In this context, high frequencies are noted as acoustic signals with a frequency higher than 10 kHz. As most acoustic transmitter signals have a mean frequency of about 70-80 kHz, self noise from propellers are expected to have an impact on the performance of the acoustic receivers. The study from NTNU [2] showed that a system performed satisfactory if the USV travelled at speeds below 3 knots, despite noise generated by the propeller and travelling hull being one of the major concerns ahead of the field trail. A possible solution is to utilize wave propelled USVs in the system in an attempt to minimize the self noise levels.

As stated in the paper *Straight-Line Target Tracking for Unmanned Surface Vehicles* by Breivik et. al [11]: *"USVs are unique in the sense that they are able to communicate with vehicles both above and below the sea surface at the same time, capable of acting as relays between underwater vehicles and vehicles operating on land, in the air, or in space."* The fact that USVs are able to communicate with satellites through a GNSS enables them to clock synchronize hydrophones and time stamp their TOA measurements with high precision. Highly accurate and synchronized clocks is a prerequisite for the TDOA estimation technique to give good estimates.

Harsh weather might prevent a USV from operating safely. Other effects from rapidly changing variables of the surface water like pressure, temperature and salinity might induce a rapidly changing bias in the velocity of sound. In a USV experiment performed at NTNU [2] effects from a fresh water run-off is believed to be one of the reasons as to why estimates of the target AUV position were inaccurate. The run-off *" created a layer of fresh water on the surface of the sea, which affected acoustic wave propagation"*.

Potential USV platform: *AutoNaut*

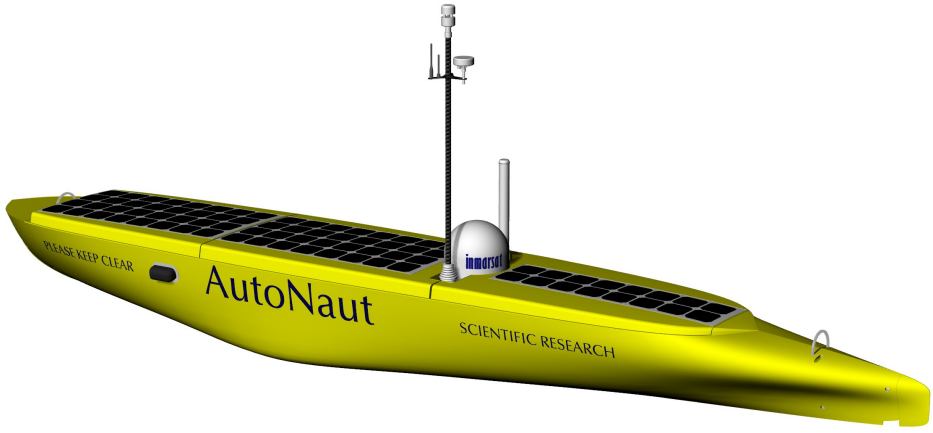

Figure 3.1: AutoNaut

AutoNaut is a wave propelled Unmanned Surface Vessel. The propagation speed is size-dependant, and varies from 1-2 knots for the smallest to 4-5 knots for the largest vessel. The *AutoNaut* could in theory be a potential hosting spot for the UAV.

In order to achieve good acoustic measurements without inducing large drag forces, the AutoNaut could be equipped with the Thin Line Array. This is *"a product family of low profile miniaturised acoustic arrays suitable for low speed towed or static applications. This is a highly adaptable product that provides impressive performance with reduced power consumption, weight, drag and cost when compared with traditional line and towed arrays."[12]*

High endurance is to be expected for the vessel as it is equipped with solar panels able to power sensors and other payloads. Due to the mobility and high signal-to-noise ratio provided by AutoNaut the vessel seem near ideal for fish tracking purposes.

3.2 AUV

A constellation of AUVs equipped with acoustic receivers are in theory able to form a spherical formation surrounding the target minimizing the effects of dilution of precision. An AUV could also provide tracking robustness as an assisting vessel to a swarm of surface vessels in the case of lacking or defective pressure measurement. AUVs have complementary noise characteristics compared to USVs as the intermittent noise and self noise are expected to be low. Noise from rain and waves is damped due to the path loss propagating from the surface to the depth of the AUV. Noise from cavitation on the propellers should be reduced as the effects of the pressure-related phenomena is decreasing for increasing depths.

Uncertainties in the position estimates of submerged AUVs might affect the ability of an AUV system to track a target. AUVs can be localized through excessive use of strap-down INS in addition to other acoustic positioning methods. A tightly coupled positioning algorithm can be used to obtain position estimates. Acoustic positioning systems are necessary to reduce drift in the estimates of horizontal position, velocity and attitude provided by the INS. Some examples of such systems are Doppler Velocity Log (DVL), Long Baseline (LBL) and Ultra-short baseline (USBL). As the position of the receivers mounted to the AUVs are directly used in the estimation technique for target position, errors could corrupt and limit the ability of the estimator to converge and determine the position of the target.

Many communication-related problems could occur while utilizing UAVs in a unmanned tracking system. AUVs must broadcast their TOA measurements and be able to receive formation control instructions through an acoustic signal. An acoustic two-way communication link is therefore needed. As a result, delays and additional acoustic noise is introduced in the system. It is important that the AUV communicate with other nodes in the systems on frequencies sufficiently different from the frequency of the target transmitter. Frequencies which might disturb and corrupt the natural behavior of surrounding animals should also be avoided. Marine mammals are able to hear on a large spectrum of frequen-

cies and acoustic communication serves a vital role in their behavior.

As the methods utilizing GNSS data in order to synchronize clocks is not applicable for submerged AUVs, other methods must be utilized to synchronize the clocks in the AUV nodes of the system. A study from 2015 claims that *"terrestrial clock synchronization protocols are not readily applicable to underwater acoustic networks, because of long propagation times, low packet delivery success rates, communication ranges that vary over time in an unpredictable manner, and, in the presence of mobile nodes, the ad hoc nature of the composition of the network."*[10] A robust clock synchronization algorithm for ad hoc underwater acoustic networks was presented in the paper. Alternative clock synchronization methods for underwater acoustic networks are however not expect to be as accurate as the ones applicable on the surface. The accuracy of the TDOA localization method is directly dependant on the quality of the clock synchronization system.

As the dynamic forces induced by surface waves decay exponentially with increasing depth, submerged AUVs are likely to have higher availability and a larger operational window than USVs. AUVs could prove to be useful during harsh environment tracking situations like storms.

Potential AUV platform: LAUV

Figure 3.2: LAUV

The LAUV is an AUV with a maximum speed of 5 knots. With a cruising speed of 3 knots could the AUV serve as a relatively fast receiver node in the system. When underwater is it able to use an acoustic modem to exchange messages with the acoustic modem installed in the gateway or other vehicles. [14]

3.3 AUV glider

By being equipped with a set of wings do AUV gliders obtain forward speed through gradually adjusting buoyancy. In comparison to regular AUV designs with electrically driven propellers are the self noise values of AUV gliders expected to be even lower, and the vessels expected to have higher endurance.

The gliders are however constrained to propagate through water in distinct sinusoidal patterns. This is a limiting factor with respect to the maneuverability of the vessels. AUV gliders are relatively slow normally propagating at speeds ranging from a half knot to a full knot, or approximately 0.25 to 0.5 m/s. However, for tracking slow predictable animals or other phenomena, the maneuverability and speed of the vessels might be sufficient, granting possibility of surveilling long distance migration patterns.

Potential AUV glider platform: *Seaglider*

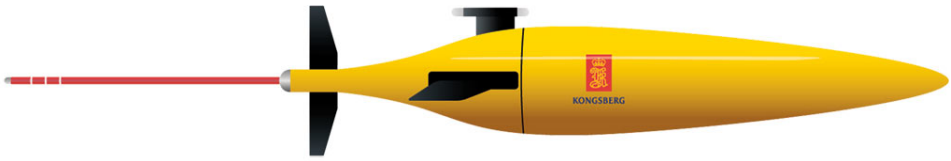

Figure 3.3: Seaglider

With a maximum travel range of 4,600 km could the Seaglider enable surveilling of the migration patterns of slow aquatic animals. The vessel has a typical speed of 0.25 m/s, with a glide angle varying from 16-45 degrees. [13]

3.4 UAV

A UAV mounted with an acoustic receiver might be able to assist an unmanned system. As fixed winged UAVs are not applicable for rapid landing and takeoff in water, a muliticopter UAV would better serve the task. New ”waterproof” multicopter designs have emerged on the market, granting new exciting possibilities within the field of acoustic telemetry.

The high speed, mobility and maneuverability of multicopters could potentially serve an unmanned system in critical situations whereas the system is about to loose track of the target. Through being able to quickly fly to the estimated position of the target and land, and then act as a node in the receiver network by obtaining and broadcasting in situ measurements on the surface, target loss could potentially be prevented.

Many other potential uses for the UAV exist. The UAV could serve as communication link between an AUV and a surface vessel, e.g deliver formation control messages or assist in localization of the AUV. UAVs could potentially pick up and deliver different

objects in the surface or on/off a moving platform. Two examples of potential objects are lightweight drifting sonobuoys and AUVs. These objects can act as nodes in the receiver array by broadcasting TOA measurements through radio. As drift buoys lack the ability of positioning themselves the UAV could serve as transporter adjusting their position in the water or reallocating them to a moving platform.

Many of these operations would demand a robust pickup/dropoff system, as well as high precision navigation and calm weather conditions. Enhanced variations of GNSS e.g. RTK and sensor fusion with radar or Lidar systems could be used to obtain higher precision of position estimates.

Multicopters have a very limited endurance and the use should therefore be limited to situations where target loss is likely to occur. Dependant on location and duration of the operation, a floating platform for the UAV to take off and land on might be needed. An ideal situation might be to utilize one of the existing USVs in the system as a host. A landing and attachment procedure could however be very complex and time demanding and potentially not possible in hazardous weather conditions.

Potential UAV platform: *HEX H20*

Figure 3.4: Hex H20 UAV

The *HEX H20* is a UAV with a propagation speed of up to 35 mph. It is ”water safe” and should therefore be able to land and take off in water. The UAV is able to a carry a payload of up to 2 kg. An example of an acoustic receiver the *Hex H20* is able to carry is TBR-700 by *Thelma Biotel*, which weighs 1140 g.[16] The UAV should therefore be able to participate in the search for signals transmitted by the fish tag through making in situ measurements while landed in water. The UAV could take off from a host platform, e.g USV, or ground-level - depending on the localization of the operation.

3.5 Geometric dilution of precision

Geometric dilution of precision (GDOP) is a measure which represents the degree the geometry of the transmitter and receivers dilutes the solution accuracy of an estimator.[18] GDOP is derived from a co-factor matrix $\mathbf{G}$:

$$\begin{aligned}\mathbf{G} &= (\mathbf{H}^T\mathbf{H})^{-1} \\ &= \begin{bmatrix} q_{xx} & q_{xy} & q_{xz} \\ q_{xy} & q_{yy} & q_{yz} \\ q_{xz} & q_{yz} & q_{zz} \end{bmatrix}\end{aligned} \tag{3.1}$$

Where $\mathbf{H}^T\mathbf{H}$ is the Grammian matrix of which $\mathbf{H}$ is the geometry matrix which provides a connection between the TDOA measurements and the states, as given in equation 4.7. As a result of one receiver being used as a reference in obtaining the TDOA equations, the dilution of precision values increase due to poorer geometry.

The optimal receiver constellation is the one which minimizes the sum of diagonal elements in the dilution of precision matrix $\mathbf{G}$:

$$\min_{\mathbf{x_R}\in\mathbb{R}^3} (Tr(\mathbf{H}^T\mathbf{H})^{-1}) \tag{3.2}$$

where Tr is the trace operator of the matrix (sum of its diagonal elements), and $\mathbf{x_R} \in \mathbb{R}^3$ is the positions of receivers in 3-space, e.g. a USV constellation. The constellation which minimizes the trace values happens to be a group of receivers evenly distributed on the surface of a sphere surrounding the target in the center. Constellations with aligned or co-planar receivers should ideally be avoided, as singularities in the H matrix would result in large dilutions of precision values.

3.5.1 USV constellation example

A system consisting of co-planar receivers would have large problems in determining the range of the transmitting target in the direction of the orthogonal projection in the plane. A relevant case is fish tracking utilizing surface vehicles. The USVs are co-planar at the surface resulting in dilution in the depth direction.

Imagine a system of four USV receivers in locations (0,1,0), (1,0,0), (0,-1,0), (-1,0,0) and a target in position (0,0,-1). The resulting $\mathbf{H}$ matrix is found utilizing equation 4.7, and is given as

$$\mathbf{H} = \begin{bmatrix} 0.7071 & -0.7071 & 0 \\ 0 & -1.4142 & 0 \\ -0.7071 & -0.7071 & 0 \end{bmatrix} \tag{3.3}$$

and the resulting measurement Gram matrix $\mathbf{H}^T\mathbf{H}$ becomes singular

$$\mathbf{H}^T\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.4}$$

hence, the matrix is not invertible. The elements of the dilution of precision matrix becomes enormous, compromising the estimators ability to determine the position of the target.

However, by utilizing a depth measurement transmitted by the acoustic signal, the singularity can be avoided and enhance system performance. The depth measurement enters as a element of the value 1 in the H matrix location H(4,3), as explained thoroughly in equation 4.12. Now the H matrix takes the form:

$$\mathbf{H} = \begin{bmatrix} 0.7071 & -0.7071 & 0 \\ 0 & -1.4142 & 0 \\ -0.7071 & -0.7071 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.5}$$

$$\mathbf{H}^T\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.6}$$

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.3333 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.7}$$

The constellation of 4 co-planar USVs obtain satisfactory vales of dilution of precision as long as the depth measurement is available. However, this is not necessarily always the case as situations where the USVs receivers have received a ping with defect or missing depth signal could occur.

To increase the robustness of the USV tracking system, as well as reducing the effects of dilution of precision, underwater vehicles equipped with acoustic receivers could participate as nodes in the system. Without an AUV to support the system during these situations, near singular matrices like in the case of equation 3.4 might occur. Dilution of precision values will grow large, which furthermore will effect the estimator. Over time, the error of the estimated position could grow large and potentially lead to the drone system losing track of the target.

Restricted maneuverability and communication range of the AUVs could limit the possibility of obtaining certain receiver array constellations. A constellation of AUV gliders would be limited to propagate through water in large sinusoidal patterns. In some fish tracking applications with a faster and more unpredictable target a regular AUV might be more suited for the operation.

3.5.2 AUV glider constellation example

The authors of the paper *Tracking and Following a Tagged Leopard Shark with an Autonomous Underwater Vehicle*[17] suggest incorporating multiple AUVs to achieve better tracking and improve the system. A suggested method of incorporating multiple AUV gliders with the goal of minimizing the dilution of precision values follows in this part of the paper.

Imagine a constellation of eight glider AUVs equipped with acoustic receivers tracking a tagged target moving in a straight line at constant depth. Ideally, the glider AUVs would form a spherical shape surrounding the target fish, however, due to limitations with respect to maneuverability of the vessels this is not possible. The glider AUVs are constrained to propagate through the water in large sinusoidal patterns. In an attempt to minimize the errors associated with dilution of precision, a control strategy for the AUVs is purposed.

In order to evenly distribute the eight receivers in xy plane, the AUV gliders should be evenly distributed on a circle surrounding the target. This corresponds to a 45 degree angle between each receiver. In depth direction, the AUVs should propagate in phase shifted sinusoidal patterns, in an attempt to evenly distribute the AUVs. The amplitude and mean value of the sinusoidal could vary upon several parameters, and would constantly change in real tracking system as the target moves. Limiting factors of the amplitude of the sinusoidal could be the surface, seabed or range of acoustic communication. Let T denote the time of a dive,

$$T = \frac{d}{v} \tag{3.8}$$

where d is the distance travelled in x direction during a single dive, and v is the speed in x direction, then the dive frequency is determined as f = $\frac{1}{T}$, $[h^{-1}]$ dives per hour. In an attempt to obtain even distribution in depth the AUVs could propagate in the following trajectories:

$$z_j = -d + Asin(2\pi f + \phi_j) \tag{3.9}$$

for j = 1...8, where z is the mean depth of the AUVs and ϕ is the phase shift vector:

$$\phi = \begin{bmatrix} 0 & \frac{\pi}{4} & \frac{\pi}{2} & \frac{3\pi}{4} & \pi & \frac{5\pi}{4} & \frac{3\pi}{2} & \frac{7\pi}{4} \end{bmatrix}^T \tag{3.10}$$

The operating depth range of the *Seaglider* AUV from *Kongsberg*, shown in figure 3.3, is from 50 to a 1000 meters. By assuming a linear relationship of the specifications given by *Kongsberg*, the *Seaglider* should be able to travel 460 km performing 65 dives to a hundred meters depth. Each dive corresponds to 7 km propagating at the speed of 0.25 m/s. The dive frequency of the *Seaglider* under these conditions would be $f = \frac{1}{7{,}77}[\frac{1}{hour}]$ using equation 3.8 and 3.9.

Imagine a target fish, swimming in a straight line at a constant depth of 100 meters traveling 7 km in 7,77 hours, which is tagged with an acoustic transmitter not able to provide depth estimates. Imagine a constellation of 8 *Seagliders* evenly distributed on a circle with

a radius of 100 meters in the xy plane surrounding the target fish as indicated in figure 3.5. Through forcing the *Seagliders* to propagate in evenly phase shifted sinusoidal curves in the same direction as the fish even distribution of vessels in direction z can be obtained. These sinusoidal curves should have a mean depth of 50 meters equal to the fish. In order to stay within the operating range of the *Seaglider*, as well as maximize the vessel distribution in all three directions, the amplitude of the sinusoidal curves should be 50 meters. The initial position of the vessels are indicated in figure 3.5 and 3.6. Plots of AUV glider positions in x and z direction as functions of time is provided in figure 3.7, as well as the dilution of precision values associated with each constellation at each time instant. Please note that there is an error in figure 3.7, the third plot is marked with an indicator ”GDOP in z” which should have been ”GDOP in x”. The corresponding *MATLAB* script is provided in the appendix section D.5.

The values of geometric dilution of precision (GDOP) are low throughout the entire simulation indicating good tracking capabilities for the AUV glider system. The last plot in figure 3.7 ”the sum of GDOP in x,y and z” indicates the sum of GDOP in all directions. A low sum induce good position estimates of the fish. The sum is varying due to the AUVs constant change in z direction position throughout the simulation. Please note that the constellation of AUVs with respect to the target in x and y direction is constant throughout the simulation, by forming a perfect circle surrounding the fish in the xy-plane.

In order to make some comparisons is a GDOP value between 4-6 considered to be good for GNSS receiver. Note that what is considered to be a ”good” GDOP value would be dependant on the operation. High precision tracking require very low GDOP values. However, in general for GNSS, is GDOP values below four considered to be excellent. Figure 3.7 therefore shows that the AUV glider system would provide excellent GDOP values in comparison with GNSS. This is expected as a GNSS receiver on the surface of the earth at best is able to receive signals from directions covering half a sphere. This is due to the fact that GNSS signals are not able penetrate through the earth.

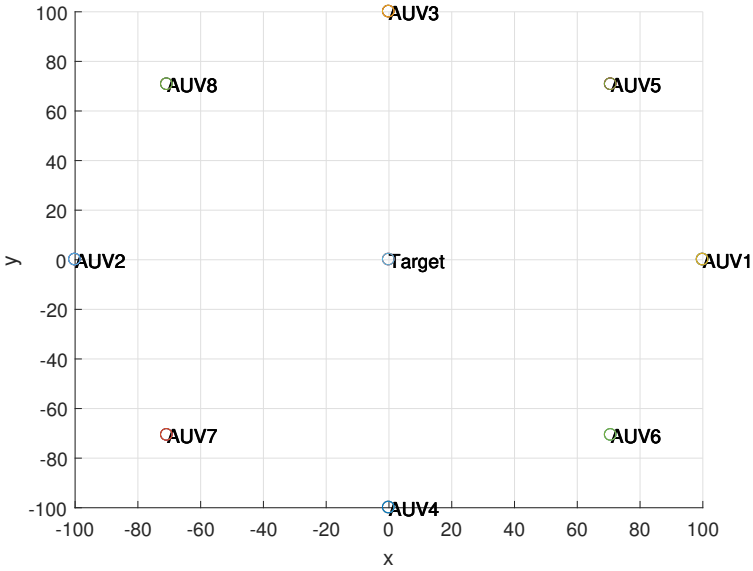

Figure 3.5: Initial AUV glider constellation from above

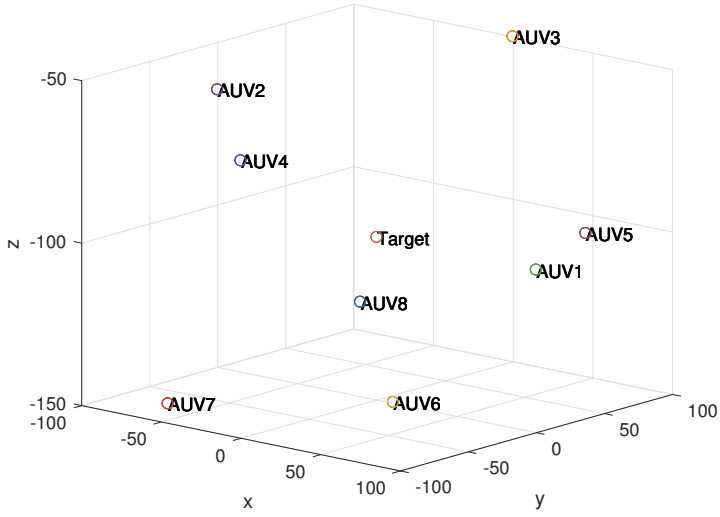

Figure 3.6: Initial AUV glider constellation

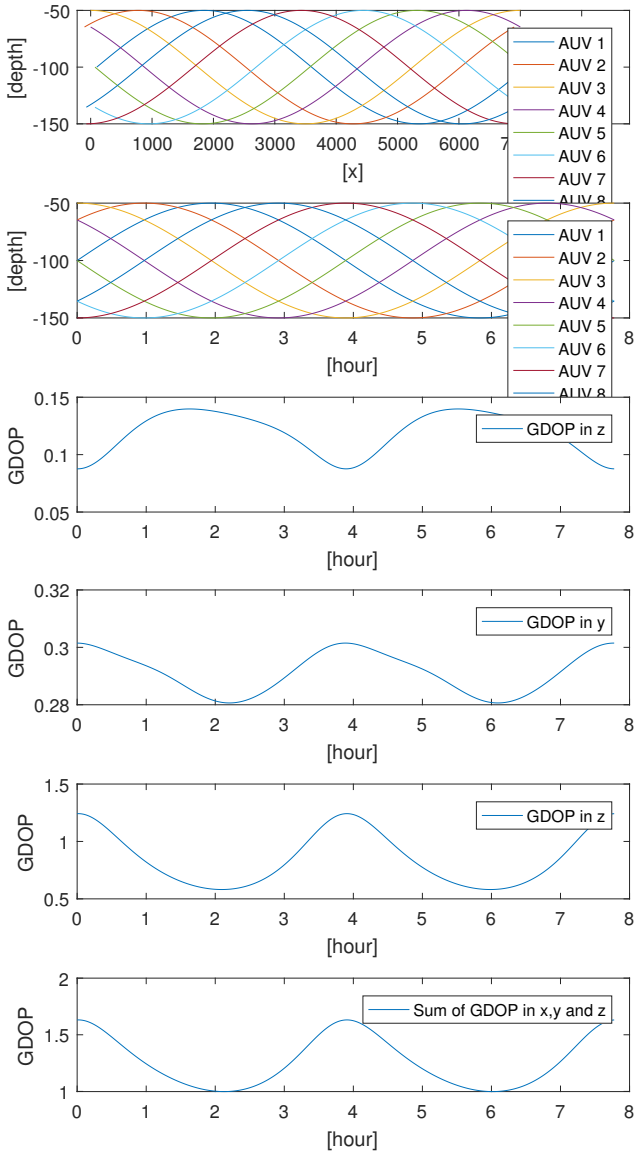

Figure 3.7: AUV Seaglider positions and GDOP plots

Chapter 4

Case study

4.1 Goal

The overall goal for this case study was to compare the tracking performance of a drone system consisting of four USVs with and without an assisting UAV. A demanding tracking scenario of a fast target fish was designed with the intention of pinpointing the advantages of using an UAV.

4.2 Simulation specifications and assumptions

All vehicles were simplified as point locations $\mathbf{x} \in \mathbb{R}^3$ with a corresponding orientation heading angle $\psi \in \mathbb{R}$. The position and orientation estimates of the vehicles were assumed to be perfect. Their velocity and heading was directly manipulated: $u = \frac{dx}{dt}$, $v = \frac{dy}{dt}$ and $w = \frac{dz}{dt}$. All the vehicles were assumed to host an acoustic receiver. Receiver clocks were assumed to be perfectly synchronized. The target fish was equipped with an transmitter transmitting a signal propagating at the speed of sound in water set equal to 1484 $\frac{m}{s}$. The speed of the acoustic signal was assumed to be constant, and detectable by receiver nodes within a euclidean distance of less or equal to 500 meters. The transmission range is an ad hoc value, as in reality, the range would vary as it is dependant on the receiver, transmitter and propagation channel.

The four USV nodes, i.e four *AutoNauts*, create a formation surrounding the estimated position of the fish. The maximum velocity of the vessels was set to 2 $\frac{m}{s}$. The UAV, i.e a *Hex H20*, was initially landed one of the USVs. The maximum velocity of the UAV is set to 15 $\frac{m}{s}$. The UAV was assumed to be able to land and take off from both the USV and water near instantaneously. In reality, this process could be very time demanding and complex.

The acoustic transmitter tag was set to transmit a signal every eight seconds. A depth mea-

surement was set to be available as long as at least one receiver has acquired the acoustic signal, hence potentially available as frequent as the acoustic signal it self. The acoustic transmission range was set to 500 meters.

Radio communication links between the USVs and UAV were assumed to be stable and instantaneous. In real systems communication loss, damaged data packages, path and group delay are to be expected.

The initial error of the estimator (Kalman filter) was set to be 1 meter in direction x, y and z.

Forces and disturbances from waves, wind and current was neglected.

The TOA and depth measurements were assumed to subject to zero-mean Gaussian white noise. In reality, slowly varying biases and other disturbances are expected. As the depth measurement is assumed to be more accurate than the TOA measurements the power its associated noise was multiplied with a factor of 0.1. The matrix elements Q(3,3) and R(3,3) corresponds to the associated uncertainty in the process model and measurement for depth, respectively. Q(3,3) was set to hold a large value in order to make the Kalman filter weight the depth measurement. R(3,3) was set to hold a lower value than the other elements in order to make the Kalman filter weight the depth measurement higher than the other measurement.

4.3 Fish trajectory generation

The main goal of this section was to create a fish trajectory which could represent a fish which is involved in a predator-prey interaction, such as described in this paper in section 1.2. Fish behavior can be modelled as a random walk process in 3-space. This model could be fitting for fish which is lurking around in a reef not propagating in any specific direction. Another approach is to model the fish movement as a straight line or a parametric curve in space. However, as another goal was to create a trajectory which could represent the behavior of a real fish in the best way possible, a different method granting a more natural trajectory was developed.

A curve consisting of several connected sinusoidal trajectories in $\mathbb{R}^2$ was used as a basis for the trajectory. The target moves along the trajectory with varying speed. A bias from a slowly varying 1st order Markov process in $\mathbb{R}^3$ was created. The final trajectory was set to be equal the sum of the sinusoidal trajectory and the slowly varying Markov process. This method creates a trajectory which seems to represent "the natural behavior of a fish". The slowly varying Markov process is created by allowing white noise to travel through a low-pass filter:

$$\dot{\mathbf{x}}(t) = \frac{-1}{T}\mathbf{x}(t) + \mathbf{w}(t) \tag{4.1}$$

where $\dot{\mathbf{x}} \in \mathbb{R}^3$ is the velocity of the target, $\mathbf{x} \in \mathbb{R}^3$ is the position of the target, T is the Markov time constant and $\mathbf{w} \in \mathbb{R}^3$ is Gaussian white noise. The resulting trajectory is

presented in figure 4.1. The trajectory was created using the *MATLAB* script provided in appendix section E.2. The duration of the simulation is 6000 seconds. The trajectory is

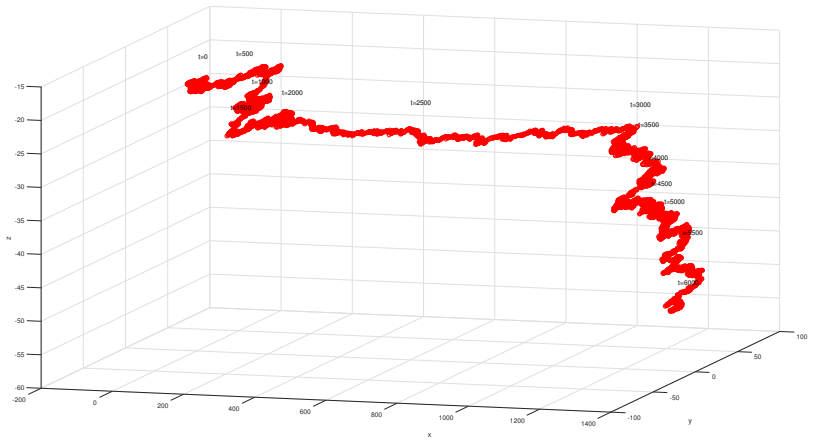

Figure 4.1: Trajectory of a fish

representing a fish propagating mainly in one direction relatively close to the surface. As indicated by the time tags in figure 4.1, the fish is propagating at low speeds in the time interval t = $[0, 2000]$ and t = $[3000, 6000]$, performing a short dive in the latter. In time interval t = $[2000, 3000]$ the fish is propagating at an average speed 2.668 [m/s] in x direction. This could represent the time span of which the fish is involved in a predator-prey interaction. Due to the speed constraints of the USVs of 2 [m/s] is the target propagating in a speed which is on the borderline of what should be possible for the system to track.

4.4 Observer

4.4.1 Determining an estimation technique

Estimating the position of the target fish using TDOA measurements is a non-linear problem. A large variety of estimation techniques exist and could be employed in order to solve the problem. A robust estimation technique based on a dynamic model is preferable as signal loss and out-of-range situations are expected.

The Kalman filter is an optimal processing algorithm, with optimal meaning it minimizes the associated errors in some respect. The Kalman filter is based upon two main assumptions[19]:

1. Both the target motion and measurement model is linear
2. The probability distribution of the associated errors of the motion and measurement model is Gaussian

In the case of tracking fish using TDOA measurements, neither the motion nor measurement model is linear. However, a variant of the Extended Kalman Filter can be employed to estimate solutions of the non-linear equations.[20] A drawback with the method is that only the first order approximations of the optimal errors are provided which might lead to sub-optimal performance and divergence.

The Kalman filter is a well know method to estimate states dealing with noisy data. Through implementing a dynamic model the extended Kalman filter should be able to estimate the transmitter position when the acoustic receiver data-set is changing in size. The extended Kalman filter technique was implemented based upon these characteristics.

4.4.2 Extended Kalman Filter estimation of the TDOA solutions

The Extended Kalman Filter equations are given as

$$\begin{cases} \hat{\mathbf{x}}_{k+1|k} & = \mathbf{F}\,\hat{\mathbf{x}}_k \\ \mathbf{P}_{k+1|k} & = \mathbf{F}\,\mathbf{P}_{k|k}\,\mathbf{F}^T + \mathbf{Q}_k \end{cases} \tag{4.2}$$

$$\begin{cases} \mathbf{K}_k & = \mathbf{P}_{k+1|k}\,\mathbf{H}^T\,(\mathbf{H}\,\mathbf{P}_{k+1|k}\,\mathbf{H}^T + \mathbf{R}_k) \\ \hat{\mathbf{x}}_{k|k} & = \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_k(\mathbf{y} - \mathbf{h}(\hat{\mathbf{x}}_{k+1|k})) \\ \mathbf{P}_{k|k} & = (\mathbf{I} - \mathbf{K}_k\,\mathbf{H}_k)\mathbf{P}_{k+1|k} \end{cases} \tag{4.3}$$

where the prediction step is given in equation 4.2, and the measurement update step is given in equation 4.3. First, the predicted state vector $\hat{\mathbf{x}}_{k+1|k}$ and projected error covariance matrix $\mathbf{P}_{k+1|k}$ are calculated. In the next step the Kalman gain $\mathbf{K}_k$ is computed, and the state estimate $\hat{\mathbf{x}}_{k|k}$ and error covariance matrix $\mathbf{P}_{k|k}$ updated.

The state estimate vector $\hat{\mathbf{x}}_k$ is a 6x1 vector with estimates of the position and velocities of the target in $\mathbb{R}^3$ at time step k:

$$\hat{\mathbf{x}}_k = \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \\ \hat{u} \\ \hat{v} \\ \hat{w} \end{bmatrix} \tag{4.4}$$

$\mathbf{F}_k$ and $\mathbf{H}_k$ are the state transition and observation matrix, respectively. $\mathbf{F}_k$ is the linearized nonlinear process function matrix and $\mathbf{H}_k$ is the linearized nonlinear measurement/observation matrix.

$$\begin{aligned} \mathbf{F}_k &= \frac{\delta \mathbf{f}}{\delta \hat{\mathbf{x}}}\Big|_{\hat{\mathbf{x}}=\bar{\hat{\mathbf{x}}}_k} \\ &= \begin{bmatrix} 1 & 0 & 0 & \Delta T & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta T & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta T \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{4.5}$$

Where the $\bar{\hat{\mathbf{x}}}_k$ is the priori state estimate at time step iteration k, and Δ T is the size of the time step.

$$\begin{aligned}\mathbf{H}_k &= \frac{\delta \mathbf{h}}{\delta \hat{\mathbf{x}}}|_{\hat{\mathbf{x}}=\hat{\mathbf{x}}_\mathbf{k}{}^-} \\ &= M \begin{Bmatrix} \begin{bmatrix} \frac{\delta h_j}{\delta \hat{x}} & \frac{\delta h_j}{\delta \hat{y}} & \frac{\delta h_j}{\delta \hat{z}} & \frac{\delta h_j}{\delta \hat{u}} & \frac{\delta h_j}{\delta \hat{v}} & \frac{\delta h_j}{\delta \hat{w}} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ \frac{\delta h_N}{\delta \hat{x}} & \frac{\delta h_N}{\delta \hat{y}} & \frac{\delta h_N}{\delta \hat{z}} & \frac{\delta h_N}{\delta \hat{u}} & \frac{\delta h_N}{\delta \hat{v}} & \frac{\delta h_N}{\delta \hat{w}} \\ \frac{\delta h_D}{\delta \hat{x}} & \frac{\delta h_D}{\delta \hat{y}} & \frac{\delta h_D}{\delta \hat{z}} & \frac{\delta h_D}{\delta \hat{u}} & \frac{\delta h_D}{\delta \hat{v}} & \frac{\delta h_D}{\delta \hat{w}} \end{bmatrix} \end{Bmatrix}\end{aligned} \tag{4.6}$$

for $j \in \mathbb{J}$.

Where $\hat{\mathbf{x}}_\mathbf{k}{}^-$ denotes the posterior state estimate at time step k, and $\mathbb{J}$ denotes a vector of size Nx1 with N begin the total number of TDOA measurements: $\mathbb{J} = \begin{bmatrix} j & \dots & N \end{bmatrix}$.

The measurement function elements $\mathbf{h}_j$ ... $\mathbf{h}_N$ are created in the following manner:

$$\begin{aligned}\mathbf{h}_j(\hat{\mathbf{x}}) = &\sqrt{(x_{R_j} - \hat{x})^2 + (y_{R_j} - \hat{y})^2 + (z_{R_j} - \hat{z})^2} \\ &- \sqrt{(x_{R_{ref}} - \hat{x})^2 + (y_{R_{ref}} - \hat{y})^2 + (z_{R_{ref}} - \hat{z})^2}\end{aligned} \tag{4.7}$$

for $j \in \mathbb{J}$.

With $x_{R_{ref}}$, $y_{R_{ref}}$ and $z_{R_{ref}}$ being the positions of the reference receiver $\mathbf{x}_{R_{ref}} \in \mathbb{R}^3$:

$$\mathbf{x}_{R_{ref}} = \begin{bmatrix} x_{R_{ref}} \\ y_{R_{ref}} \\ z_{R_{ref}} \end{bmatrix} \tag{4.8}$$

In equation 4.16, $\mathbf{h}_D$ and M denotes the depth measurement function and sum of total of TDOA measurements in addition to the depth measurement, respectfully. M is given by the following relationship:

$$M = N + D \tag{4.9}$$

where D holds the value of 0 or 1 depending of whether the depth measurement is available or not. The width of the H matrix is equal to the number of states in the state matrix; 6.

As a result of lacking velocity measurements of the target, the elements $\frac{\delta h}{\delta \hat{u}}$, $\frac{\delta h}{\delta \hat{v}}$ and $\frac{\delta h}{\delta \hat{w}}$ become zero for both $\mathbf{h}_j$ ... $\mathbf{h}_N$ and $\mathbf{h}_D$.

The depth measurement is derived from a pressure measurement, and obtained through the following relationship:

$$\begin{aligned} p_m &= \rho g z_m + p_{atm} \\ \Rightarrow z_m &= \frac{1}{\rho g}(p_m - p_{atm}) \end{aligned} \tag{4.10}$$

where p_m is the measured pressure, ρ is the density of seawater and p_{atm} is the atmospheric pressure. By assuming the direct depth measurement is provided through the

emitted signal from the fish tag, the resulting depth measurement equation $\mathbf{h}_D$ is given as:

$$\mathbf{h}_D = z_m \tag{4.11}$$

which further results in:

$$\begin{aligned} \frac{\delta \mathbf{h}_D}{\delta \hat{x}} &= 0 \\ \frac{\delta \mathbf{h}_D}{\delta \hat{y}} &= 0 \\ \frac{\delta \mathbf{h}_D}{\delta \hat{z}} &= 1 \end{aligned} \tag{4.12}$$

The observation matrix $\mathbf{H}_k$ is now reduced to:

$$\mathbf{H}_k = \begin{bmatrix} \frac{\delta h_j}{\delta \hat{x}} & \frac{\delta h_j}{\delta \hat{y}} & \frac{\delta h_j}{\delta \hat{z}} & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\delta h_N}{\delta \hat{x}} & \frac{\delta h_N}{\delta \hat{y}} & \frac{\delta h_N}{\delta \hat{z}} & \vdots & \vdots & \vdots \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{4.13}$$

Where the elements $\frac{\delta h_j}{\delta \hat{x}}$, $\frac{\delta h_j}{\delta \hat{y}}$ and $\frac{\delta h_j}{\delta \hat{z}}$ are given by the following equations:

$$\begin{bmatrix} \frac{\delta h_j}{\delta \hat{x}} \\ \frac{\delta h_j}{\delta \hat{y}} \\ \frac{\delta h_j}{\delta \hat{z}} \end{bmatrix}^T = \begin{bmatrix} \frac{\sqrt{2}(x_{R_{ref}} - \hat{x})}{\sqrt{(x_{R_{ref}} - \hat{x})^2 + (y_{R_{ref}} - \hat{y})^2 + (z_{R_{ref}} - \hat{z})^2}} - \frac{\sqrt{2}(x_{R_j} - \hat{x})}{\sqrt{(x_{R_j} - \hat{x})^2 + (y_{R_j} - \hat{y})^2 + (x_{R_j} - \hat{z})^2}} \\ \frac{\sqrt{2}(y_{R_{ref}} - \hat{y})}{\sqrt{(x_{R_{ref}} - \hat{x})^2 + (y_{R_{ref}} - \hat{y})^2 + (z_{R_{ref}} - \hat{z})^2}} - \frac{\sqrt{2}(y_{R_j} - \hat{y})}{\sqrt{(x_{R_j} - \hat{x})^2 + (y_{R_j} - \hat{y})^2 + (x_{R_j} - \hat{z})^2}} \\ \frac{\sqrt{2}(z_{R_{ref}} - \hat{z})}{(\sqrt{((x_{R_{ref}} - \hat{x})^2 + (y_{R_{ref}} - \hat{y})^2 + (z_{R_{ref}} - \hat{z})^2}} - \frac{\sqrt{2}(x_{R_j} - \hat{z})}{\sqrt{(x_{R_j} - \hat{x})^2 + (y_{R_j} - \hat{y})^2 + (x_{R_j} - \hat{z})^2}} \end{bmatrix}^T \tag{4.14}$$

for $j \in \mathbb{J}$

The measurement is a vector of TDOA measurements and a depth measurement.

$$\mathbf{y} = M \left\{ \begin{bmatrix} \text{c } (\text{TOA}_j - \text{TOA}_{ref}) \\ \vdots \\ \text{c } (\text{TOA}_N - \text{TOA}_{ref}) \\ z_m \end{bmatrix} \right. \tag{4.15}$$

for $j \in \mathbb{J}$. Where M is the number from equation 4.9, c is the propagation speed of the acoustic signal, TOA_j is the time of arrival at receiver j, and TOA_{ref} is the time of arrival at the reference reciever. The measurement vector is naturally only available when the receivers have received a signal.

The estimated measurements vector $\mathbf{h}(\hat{\mathbf{x}}_{k+1|k})$ is calculated in the following manner:

$$\mathbf{h}(\hat{\mathbf{x}}_{k+1|k}) = M \left\{ \begin{bmatrix} \mathbf{h}_j(\hat{x}_{k+1|k}) \\ \vdots \\ \mathbf{h}_N(\hat{x}_{k+1|k}) \\ \hat{z}_{k+1|k} \end{bmatrix} \right. \tag{4.16}$$

for $j \in \mathbb{J}$. Where $\mathbf{h}_j$ is the estimated range difference between receiver j and the reference receiver given by the measurement function from equation 4.7, and $\hat{z}_{k+1|k}$ is the predicted depth of the target provided by the Kalman filter.

$\mathbf{P}_{k+1|k}$ and $\mathbf{P}_{k|k}$ is the predicted and updated error co-variance matrix, respectively. The values represents the confidence level of the solution.

$\mathbf{Q}_k$ is the process noise co-variance matrix. The elements of the matrix amounts to the associated uncertainty in the process model. The size of the matrix is equal to the number of states in the system.

$\mathbf{R}_k$ is the measurement noise spectral density matrix. The elements of the matrix amounts to the associated confidence in each measurement. The size of the matrix is equal to the number of available measurements $\mathbf{R}_{MxM}$.

$\mathbf{K}_k$ is the Kalman gain matrix, which is a weighting factor matrix. The factor elements of the matrix amounts to a relation between the filter's use of predicted state estimate and the available measurement.

The Kalman filter equations are included in the *MATLAB* script in appendix section E.

4.5 Formation control and guidance

The objective of the formation controller is to provide the vehicles of the unmanned system with waypoints optimizing tracking performance. The formation controller can be set to optimize with respect to many different variables, e.g. fuel consumption and dilution of precision values. Fuel consumption can be reduced by weighting the distance from current USVs positions to new waypoints minimizing the distance travelled by the AUVs. The optimal constellation with respect to the dilution of precision, is the one which minimizes the diagonal values of the matrix $\min_{\mathbf{x_R} \in \mathbb{R}^3} (Tr(\mathbf{H}^T\mathbf{H})^{-1})$. Other constraints like vehicle speed and maneuverability must be taken in to account.

The formation control system consist of two sub systems: the USV formation control system and the UAV situation awareness model. The USV formation controller is independent of the UAV, whereas the UAV situation awareness model provides guidance commands dependant of the tracking ability of the USVs.

The main goal for the UAV is to assist the USV formation array in critical situations through obtaining in situ measurements while being landed in water. A critical situation is defined as a situation where the unmanned system could potentially lose track of the target.

4.5.1 USV formation control and guidance

The USV formation controller has two main objectives.

1. Keep USVs within acoustic transmission range
2. Form a quadrilateral shape surrounding the target's projection on the surface plane

The reason to why the USVs are set to form a quadrilateral rather than square shape started out as a misunderstanding. The author of this paper was initially convinced that the estimator would experience some difficulties in estimating the target's position if the receiver array formed a perfect square. The reason being that equal distance to the target would result in equal TOA measurements for all receivers. The USVs should preferably form a square formation in order to minimize the dilution of precision values. However, small offsets and misalignments are always present in real systems and the receiver array is able to track a target despite not forming a perfect square shape.

The USV formation geometry is determined by a 4x2 offset matrix $\mathbf{O}$

$$\mathbf{O} = \begin{bmatrix} -250 & 350 \\ 250 & -250 \\ -350 & -250 \\ 250 & 250 \end{bmatrix} \tag{4.17}$$

where the values of the matrix corresponds to the offsets in meters in direction x and y each USV should obtain with respect to the estimated position of the target.

In an attempt of limiting the distance travelled by the USVs a simple constraint was implemented. Initially, both a waypoint $\bar{\mathbf{x}} \in \mathbb{R}^2$ and a radius of acceptance r_a is defined. As long as the projection of the estimated position of the target in the xy-plane $\hat{\mathbf{x}} \in \mathbb{R}^2$ is within the radius of acceptance, the waypoint should not change. The radius of acceptance is a design parameter. Determining the radius is in general a trade-off between energy consumption and the tracking ability of the system.

In extreme cases where the target is moving towards the deep ocean, the vehicle formation radius might have to decrease in order for the USVs to stay within signal range.

A very simple line-of-sight LOS guidance law was implemented for the USVs. Other guidance laws like pure pursuit (PP) and constant bearing (CB) could also have been implemented.

Pseudocode for the USV formation controller and guidance is provided in algorithm 1. The *MATLAB* script is provided in the appendix section E.8.

4.5.2 UAV situation awareness hierarchy and guidance

The main objective of the UAV situation awareness model is to detect critical situations where the unmanned system is about to lose track of target. Pseudocode for the UAVs situation awareness hierarchy is provided in algorithm 2. The *MATLAB* script is included in appendix section E. Ideally should the UAV be on standby continuously during the operation while the situation awareness algorithm is searching for indicators of endangered

tracking ability. The situation awareness hierarchy should be installed in the system node of which the estimator is hosted. In this way, the algorithm has all TOA data available in order to make adjudicate decisions. Through broadcasting waypoints to the UAV over radio link is the situation awareness hierarchy able to change the flight plan of the UAV near instantaneously.

In order to be able to cope with the complex behavior of the aircraft some states, properties and design variables are defined in table 1, 2 and 3 in the appendix, respectively.

Active

The situation awareness hierarchy is set to change the UAVs status from inactive to active if the warning value is larger or equal to a set limit. The warning value is initially set to zero, and increments by one for each consecutive transmitter period in which only partial of the acoustic receiver data-set is available. The limit of what is defined as an insufficient data-set could vary. In this simulation the limit is set to be situations where less than three USV-mounted receivers have acquired a signal. This is based upon the result in section 2.2.3 showing that three acoustic receivers and a depth measurement is sufficient to determine transmitter location $\hat{\mathbf{x}} \in \mathbb{R}^3$. In this simulation the transmitter period is set to eight seconds resulting in an increase of the warning value by one every eight second less than three receivers have acquired the signal. The warning limit is a design parameter, whereas a high value could result in loss of target and a low limit could result in large energy consumption and unnecessary use. While inactive, the UAV is landed on a host USV, and its position is set equal to the USV position.

Airbourne

The UAV is airborne due to one of the following reasons:

1. The UAV is travelling from the USV to a waypoint in the ocean
2. The UAV is travelling from a waypoint in the ocean towards another waypoint in the location
3. The UAV is travelling from a waypoint in the ocean towards the USV for landing

The functions guidanceUAVtoUSV() and guidanceUAVtoTarget() are implemented with a simple line-of-sight (LOS) guidance law, similar to the one for the USVs in algorithm 1. The flight dynamics are simplified allowing the velocity of the UAV to be manipulated directly. The *MATLAB* script is included in appendix sections E.6 and E.7.

The warning value is constantly checked throughout the flight which enables the situation awareness hierarchy to change plan while airborne. This feature might be useful is the following cases:

1. The UAV is travelling towards a waypoint in the ocean as the number of USVs receiving the emitted signal changes from being insufficient to sufficient.
2. The UAV is travelling towards its host/carrier USV as the number of USVs receiving the emitted signal changes from being sufficient to insufficient.

Landing procedure

The water-landing location of the UAV is denoted as a waypoint $\bar{x} \in \mathbb{R}^2$. Two possible approaches for determining waypoints follows

1. Minimize the dilution of precision values of the resulting receiver array. Pick $\bar{x} \in \mathbb{R}^2$ such that $\min_{\bar{x} \in \mathbb{R}^2} (Tr(\mathbf{H}^T \mathbf{H})^{-1})$
2. Land as close to estimated target position as possible. Pick $\bar{x} \in \mathbb{R}^2$ such that $\min_{\bar{\mathbf{x}} \in \mathbb{R}^2} (|\bar{\mathbf{x}} - \hat{\mathbf{x}}|)$. Where $\hat{\mathbf{x}}$ is the projection of the estimated target position in the surface place.

In this simulation, the UAV is set to land using the latter strategy. This strategy was picked based upon two main arguments.

1. Uncertainties associated with target position estimates are expected to be high during UAV operation, therefore the UAV should land as close to the estimated value to ensure signal acquisition.
2. As the target velocity is expected to be high during critical situations, the fish might move out of the acoustic transmission range of the landed UAV. As the UAV is forced to fly to a new waypoint, and unable to obtain signal during flight, the risk of loosing track of the fish increases.

However in real tracking situations landing close to the target might scare and induce unwanted animal behavior and should be taken into account when the system is designed. The landing procedures are simplified in the simulation allowing the UAV to land in both water and USV within 0.5 seconds. The UAV is set to land as the euclidean distance between $\mathbf{x}_{UAV}$ and the landing waypoint in water $\bar{\mathbf{x}}$ or USV position $\mathbf{x}_{R_j}$ is smaller than the radius of acceptance for landing. Where j is the number of the USV vessel or vessels able to host the UAV.

Landed in water

The UAV is set to stay in the water for as long as both the UAV is receiving signal and the number of USVs receiving the emitted signal is insufficient. However, in the case where the UAV has waited for a consecutive number of seconds τ larger than or equal to the limit τ_{tau}: $\tau_{tau} \leq \tau$, without receiving a signal, it should become airborne and fly to the most recent position estimate $\hat{\mathbf{x}}$. τ_{lim} is a design parameter and could be dependent on many variables. In challenging acoustic environments with a lot of noise present, a high value might be preferable. As a minimum, it must be higher than the transmitter period of the fish tag. The number of seconds waited τ is reset to zero each time the UAV becomes airborne and its value increases only while the UAV is landed in water.

4.6 Simulation results

4.6.1 Color codes

The simulation plots are not perfectly intuitive and need a little explanation. The target position of the fish is marked with red in figure 4.1. The estimated position of the target is marked with five different colors, all corresponding to an equally large time span of 1200 seconds. The colors where introduced in an attempt to show how the estimated position is changing in time. The USV positions are indicated with black squares. The UAV position is indicated with yellow when active, and red when inactive. A red and green vertical arrow indicate UAV takeoff and landing, respectively. The color codes are presented in table 4.

4.6.2 Drone system tracking results: 4 USVs and 1 UAV

Figure 4.2 display an unmanned system consisting of four USVs and one UAV tracking the fish from figure 4.1. Two additional plots from other angles of view are provided in figure 1 and 2 in the appendix. The *MATLAB* script generating the simulation is provided in appendix section E.

Initially, all four USV receivers are within transmission range forming a quadrilateral formation surround the target. The UAV is inactive in the beginning and towards the end of the simulation indicated with red color on the host USV. As the target is propagating at an average velocity of 2.668 m/s in the time interval t = $[2000, 3000]$, and the USVs have a maximum velocity of 2 m/s, the two USVs furthest to the left in the figures fall out of transmission range of the signal. This correspond to receiver 1 and 3 after approximately 2400 seconds of the simulation. A figure indicating which of the receivers that acquired signal at given time instants during the simulation is provided in figure 4.3.

As only two receivers are acquiring the acoustic signal after approximately 2400 seconds the warning value starts increasing. As the warning value grows larger than the warning limit the situation awareness hierarchy calls for the UAV to become active. The UAV takes off from its host/carrier USV and lands close to the estimated position as indicated in figure 4.4. UAV related data; euclidean distance to estimated target, airborne status, seconds waited τ, signal acquisition and warning value is indicated in this figure.

After approximately 2800 seconds has the fish propagated out of transmission range with respect to the UAV. When the UAV has waited for a sufficient number of seconds it becomes airborne, flies for a short time period, and lands at a new waypoint close to the target position. The target is slowing down after approximately 3000 seconds and after 5100 seconds USV R1 is within acquisition range once more. As a result of having three USV receivers within range the situation awareness hierarchy resets the warning value to zero, and the UAV is called back to the host USV as the tracking situation is under control.

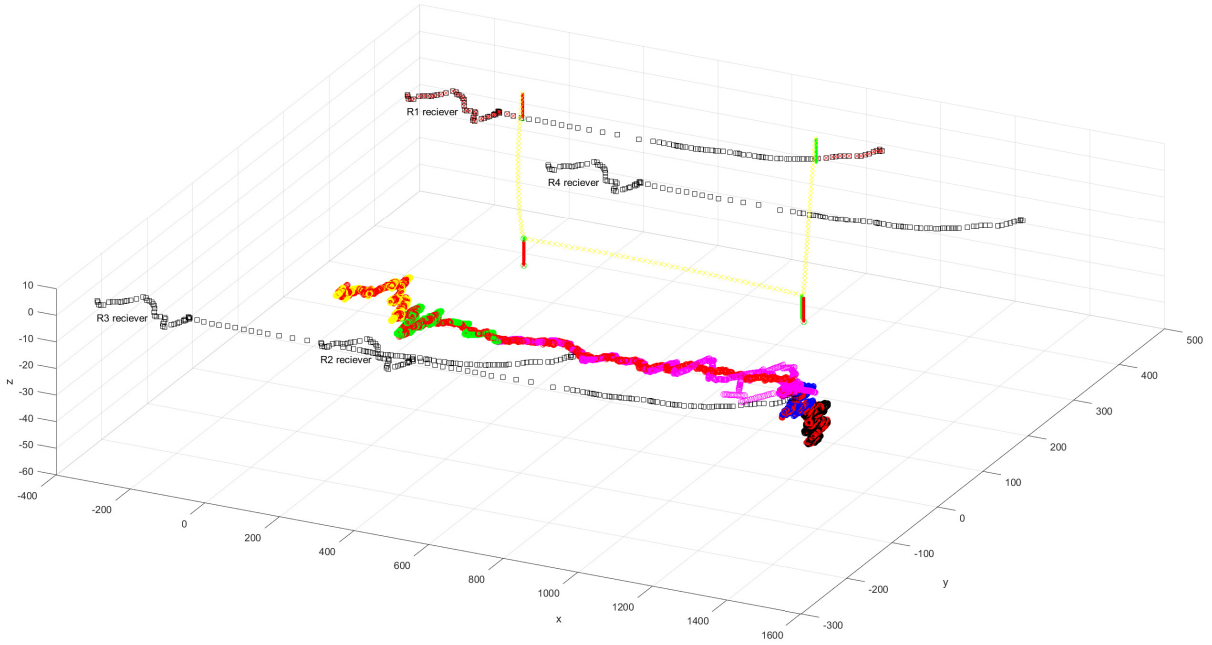

Figure 4.2: 4 USVs and UAV tracking the target fish

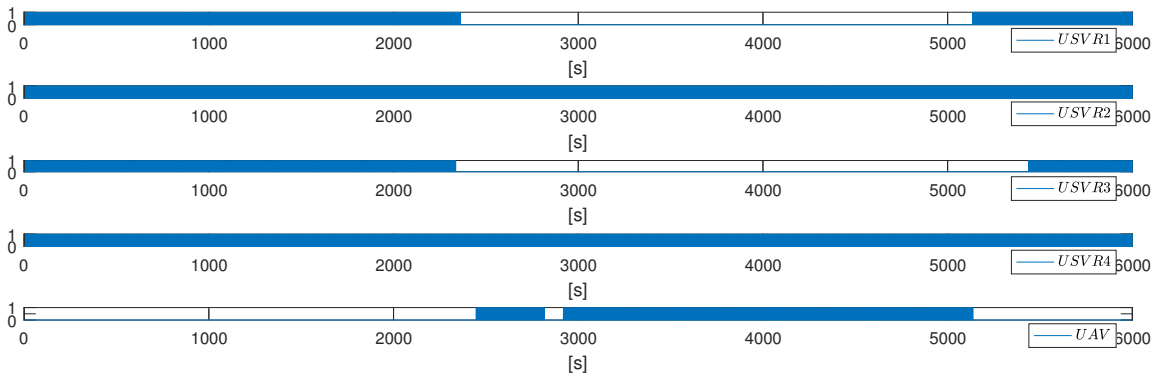

Figure 4.3: TOA measurements acquired throughout the simulation for the various receivers

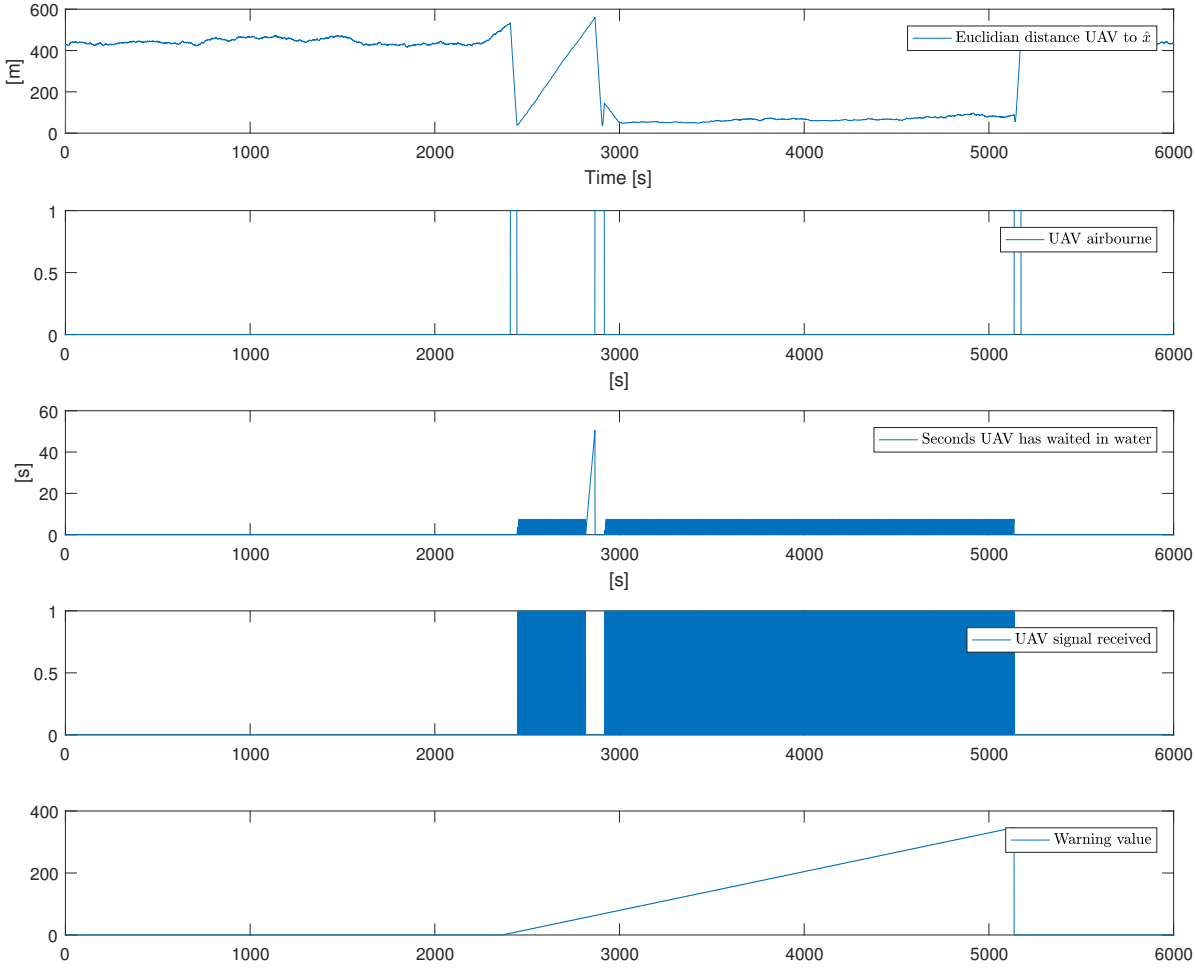

Figure 4.4: UAV data

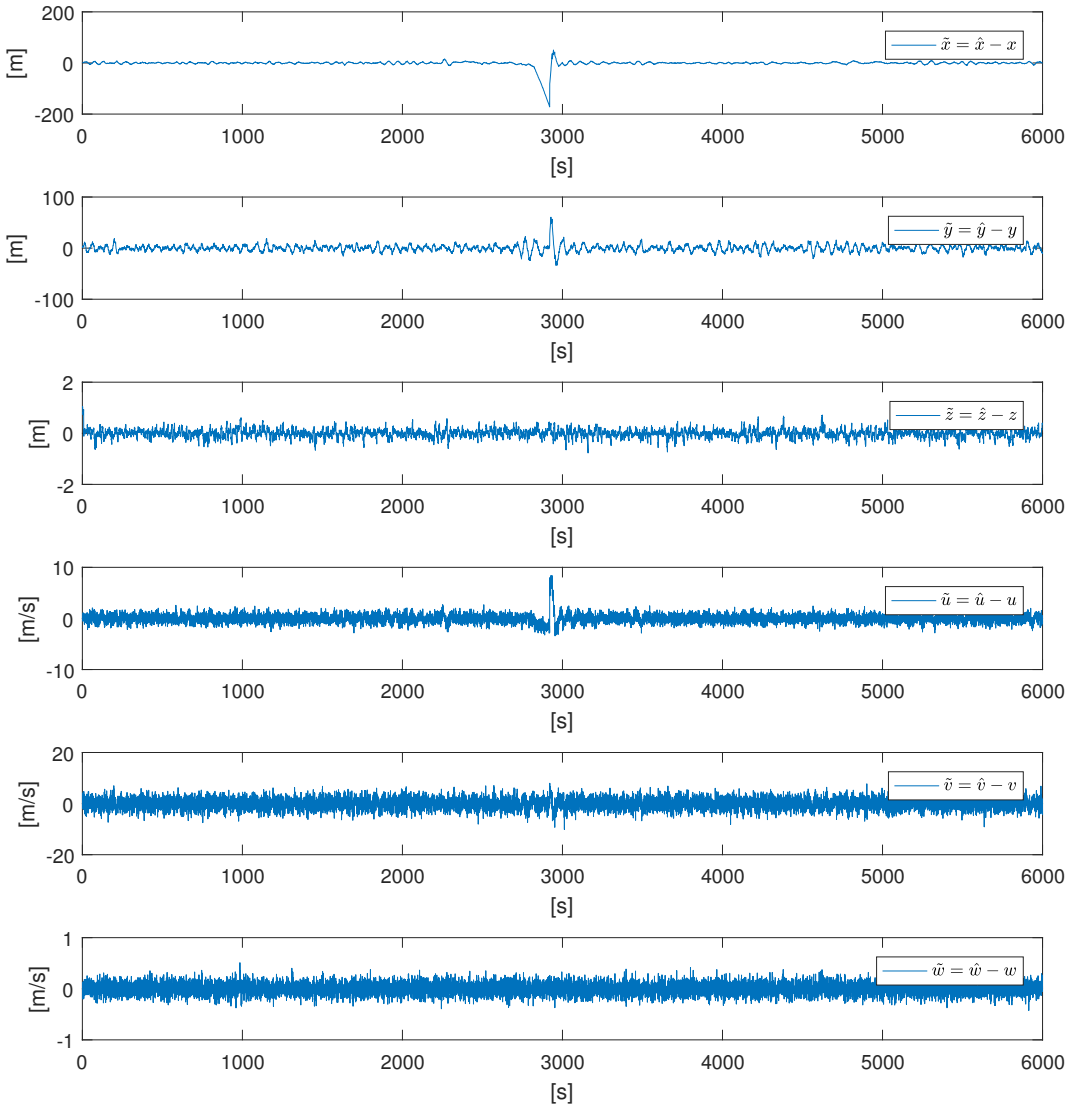

Figure 4.5: Error in estimated target position and velocities in directions x, y and z for system of 4 USVs and 1 UAV

The errors in estimated target position and velocities are given in figure 4.5. The

position errors in x and y direction are significantly larger than in z direction throughout the entire simulation. This is, however, natural as the depth measurement provides better estimates in this direction. The error plots show that the critical point of the simulation occur around 2800 seconds. The error states increase drastically as the UAV suddenly is not able to acquire the signal. The error states peak at approximately -200 in x direction and 50 in z direction. The errors are quickly reduced as the UAV lands for the second time and its receiver once more is able to provide in situ TOA measurements.

4.6.3 Drone system tracking results: 4 USVs

In order to make some comparison between the ability of the unmanned systems to track the target a second simulation is presented. In this simulation the very same USV formation from figure 4.2 is trying to track the same target fish from figure 4.1, but without an assisting UAV. The results are presented in figure 4.6, 4.7 and 4.8. Figure 4.6 is showing the estimated position in changing colors and the true target position indicated with red color. Due to the large errors in estimated position the receivers indicated with black color becomes indistinguishable. Figure 4.7 indicates receiver signal acquisition throughout the situation. Figure 4.8 indicates the estimation errors. In the second half of the simulation, none of the USVs are within transmission range. As there is no measurements available to correct the estimates, the Kalman filter keeps integrating the estimated velocities resulting in enormous errors.

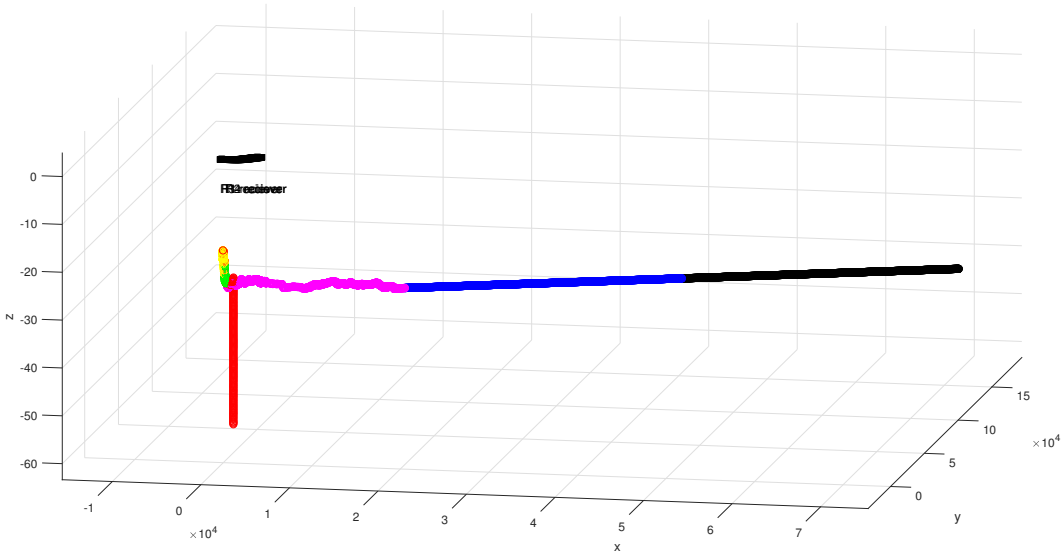

Figure 4.6: Drone system without UAV tracking target

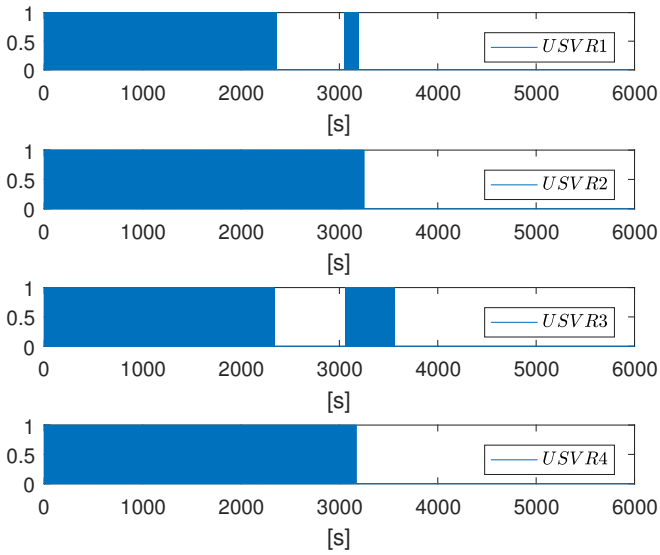

Figure 4.7: TOA measurements acquired throughout the simulation for 4 USVs without UAV

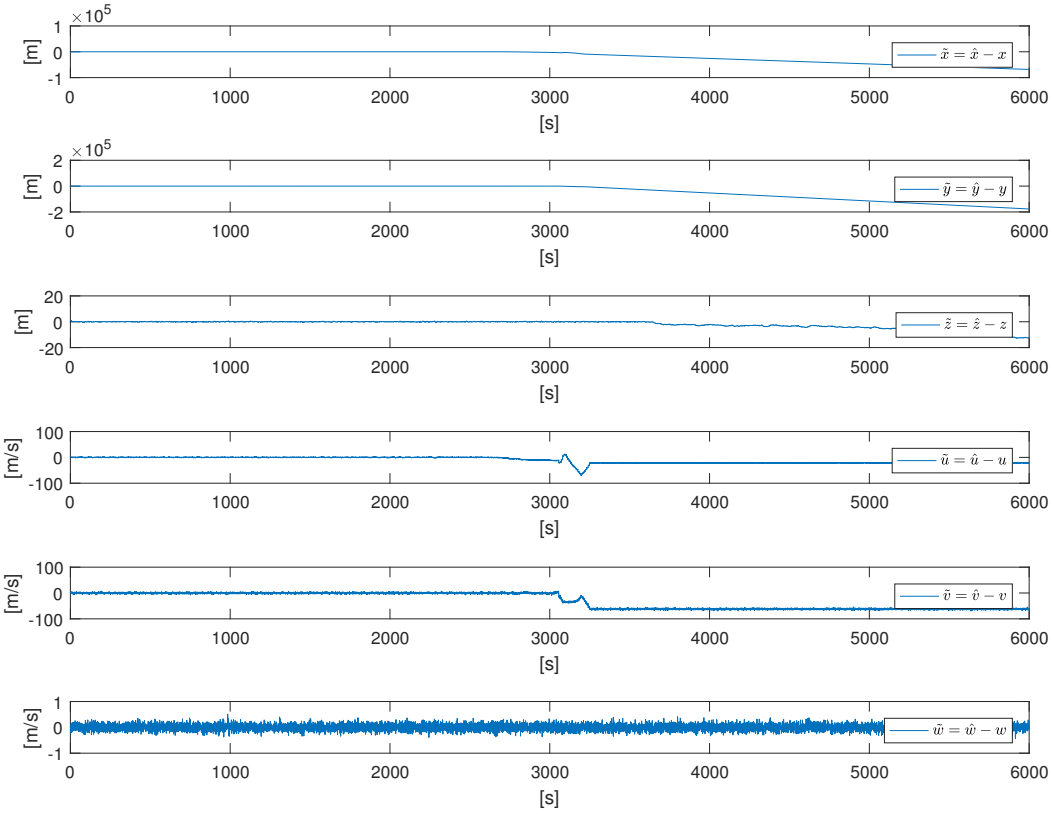

Figure 4.8: Error in estimated target position and velocities in directions x, y and z for a system of 4 USVs

Chapter 5

Discussions and conclusion

5.1 Receiver platforms and vehicle configurations

From a theoretical point of view, does a constellation of AUVs enable near ideal tracking performance through being able to obtain a spherical formation surrounding the target. In addition to provide outstanding endurance capabilities, could a formation of AUV gliders also provide very low geometric dilution of precision values through propagating in evenly shifted sinusoidal patterns. To what degree AUVs and AUV gliders are able to maintain these preferable constellations during operations with an unpredictable target is, however, questionable because of the limitations associated with speed and maneuverability of the vessels. For practical purposes are positioning, clock synchronization and communication major issues associated with the use of AUVs. A TOA measurement obtained by a AUV-fixed receiver is useless unless the AUV is able to broadcast its data to the estimator node in the system.

The precision of clocks and position estimates of acoustics receivers mounted on vehicles operating near surface are expected to be better than AUVs, as GNSS is available during the entire operation. Vehicles on the surface are able to communicate through radio link which enables the receivers to broadcast their TOA measurement data quickly. UAVs provide unique maneuverability and speed characteristics compared to the other vehicles. The tracking ability of an unmanned tracking system as presented in the case study is, however, very dependant on obtaining accurate depth measurements. An unmanned system equipped with solely UAVs and USVs leads to a near co-planar receiver array and thus in the case of a lacking depth measurement is a near-singular measurement Gram matrix $\mathbf{H}^T\mathbf{H}$ to be expected. This would furthermore lead to enormous dilution of precision values and bad estimator performance, despite the fact that a sufficient number of TOA measurements are available. In the case of unpredictable depth measurements could a system of surface vehicles and UAV be supported by an AUV for increased robustness in depth direction.

The performance of the estimator is directly dependant on the precision of both the receiver clocks and position estimates of the vehicles as well as the dilution of precision values and acoustic noise. In the process of designing a tracking system with the intention of optimizing these variables some trade-offs related to the characteristics of the vehicles must be done. The optimal constellations could potentially be the ones with a vast variety of vehicles with complimentary characteristics such as speed, endurance, maneuverability, positioning and communication. However, with an increasing variety of vehicles in the constellation comes increased system complexity with respect to communication and formation control.

5.2 Performance comparison of case study systems

In the chapter sections 4.6.2 and 4.6.3 in this paper, simulations of two unmanned systems tracking a fish involved in a hypothetical predator-prey interaction was presented. The first and second system consisted of four USVs and a UAV, and four USVs, respectively. Let the first system be denoted as system 1, and the latter as system 2. Initially, both systems were able to track the target with sufficient precision.

However, as the fish quickly propagates in x direction during time interval t = $[2000, 3000]$, system 2 experienced severe tracking difficulties. System 2 loss track of the target after approximately 3000 seconds, leading it astray for the remaining half of the simulation. The estimator was not fed any TOA measurements as the distances between the USVs and target became larger than the transmission range of the acoustic signal, set to 500 meters. As a result, the estimated velocity of the target grew very large in the extended Kalman filter. Their values could probably have been limited by implementing a saturation speed of the target fish, which would be dependant upon the species type. The error in position grew almost linearly as the near constant error in estimated velocity was integrated with every time-step.

System 1, however, was able to track the fish with sufficient precision throughout the entire simulation. The critical point of this simulation occurred after approximately 2800 seconds as the distance between the UAV and target grew larger than 500 meters. The UAV was forced to reallocate to a waypoint closer to the target. Despite experiencing severe errors in the target position estimates of approximately 160 meters in x direction and 70 meters in y direction, the estimator was able recover and track the fish with sufficient precision for the rest of the simulation after the UAV had landed and three TOA measurement was available once more.

The results from simulation of system 1 indicates that the results provided in section 2.2.3 about TDOA localization with 3 TOA and a depth measurement is correct. The poor tracking performance of system 1 as only two TOA measurements are available stresses the importance of always having three receivers within transmission range. The results from the simulations indicate that a UAV could serve an important role as an assisting vehicle in an unmanned tracking system during predator-prey interactions.

5.3 Suggestions for improvements and further work

5.3.1 Case study simulations

Many assumptions were made in order to create the simulation results presented in section 4.6. In order to create a more realistic simulation the following adjustments and system dynamics could be implemented:

- Introduce uncertainties in the positions of the vessels. The level of uncertainty should have basis in what is to be expected in a real system. e.g the accuracy of the GNSS.
- Create a model where the velocities of the vehicles are not directly manipulative. Model added mass and linear and non-linear damping phenomena. A first or second order Nomoto model for heading for the USVs can be implemented.
- Model first and second order induced forces on the vehicles from waves, wind and current.
- Create a ”realistic” target trajectory based upon the characteristics of a specific aquatic animal. It is, however, the very intention of a unmanned tracking system to provide the necessary data in order to create models of animal behavior. And as not too many of these tracking system exist and data set are lacking, many assumptions must be made.
- The landing procedures are simplified in the simulation through allowing the UAV to land within the duration of a time step, set to 0.5 seconds. The AUV is set to land on the AUV or in water as long as it is within a radius. The landing and take-off dynamics should be subject of further study.
- The acoustic measurements are assumed to be available at the instant the UAV lands in the water. Some estimates of how long time it would take for an UAV-mounted receiver to obtain acoustic signals after landing should be made.

5.3.2 Enhanced system tracking performance

Some suggestions of algorithms which could have been implemented to induce better tracking performance of system 1 in section 4.6, is presented in this section.

- Implement reference models for position, velocity and attitude to prevent large steps in controllers.
- If more than one of the USVs are able to host the UAV a simple constraint allowing the UAV to land on the one closest could be implemented in order to reduce energy consumption.

- A very simple constraint could be implemented in the estimator to filter out TOA measurements with large errors. If the position of the receivers are know exact, a TDOA measurement should never hold a value larger than the distance in between the receivers divided by the speed of sound.

5.3.3 Suggestions for further work

- Experiments examining the possibility of mounting an acoustic receiver to a water-proof UAV and test if it is able to receive signals while landed in water should be performed. An approach which might be beneficial is simply to allow the acoustic receiver to be attached to the UAV through an enforced cable, which allows it to sink a little while the UAV is landed in water. This might, however, complicate the landing procedure of landing on a moving platform.
- Examine the possibilities of landing a UAV on a surface vessel or some other moving platform. Look into methods for achieving better positioning accuracy to enable safe landing, e.g. RTK methods for enhanced GPS positioning
- Examine the benefits of utilizing other guidance laws, such as pure pursuit (PP) and constant bearing (CB).

5.4 Conclusions to research question

Many different vehicles can be utilized in an unmanned system for single fish tracking and the system should be carefully designed with respect to the characteristics of the target. While underwater vehicles have the ability to form constellations inducing low dilution of precision values, do surface and aerial vehicles have better characteristics with respect to communication and positioning. During events inducing difficult tracking conditions for a short time period, can an unmanned aerial vehicle serve as an assisting vehicle to ensure good tracking performance. The speed and maneuverability characteristics of the UAV enables it reallocate to new locations and obtain in situ measurements in the water rapidly. The simulations presented in this paper indicate that an unmanned system consisting of four USVs and a UAV have significantly better tracking ability than a system without the UAV. This result motivates further research in determining the possibility of obtaining acoustic measurements with a UAV, and to validate the concept through a proof-of-concept experiment.

Bibliography

[1] Communications and Data Systems Division, National Aeronautics and Space Administration (NASA). *Telemetry Summary Of Concept And Rationale*. CCSDS Plenary Meeting, Frascati, Italy. Green Book, Issue 1. December 1987.

[2] Artur Zolich, Tor Arne Johansen, Jo Arve Alfredsen et al. *A Formation of Unmanned Vehicles for Tracking of an Acoustic Fish-Tag*. Norwegian University of Science and Technology Trondheim, Norway. Oceans, 2017.

[3] Nigel E. Hussey, Steven T. Kessel, Kim Aarestrup, Steven J. Cooke, Paul D. Cowley, Aaron T. Fisk, Robert G. Harcourt, Kim N. Holland, Sara J. Iverson,* John F. Kocik, Joanna E. Mills Flemming, Fred G. Whoriskey. *Aquatic animal telemetry: A panoramic window into the underwater world*. Science magazine. Vol 348 Issue 6240. 12 June 2015.

[4] H. A. Urke, T. Kristensen, J. B. Ulvund. J. A. Alfredsen *Riverine and fjord migration of wild and hatchery-reared Atlantic salmon smolts*. Fisheries Management and Ecology. Pages 544 to 552. 2013.

[5] Deepak Bhadauria, Volkan Isler, Andrew Studenski and Pratap Tokekar. *A Robotic Sensor Network for Monitoring Carp in Minnesota Lakes*. 2010 IEEE International Conference on Robotics and Automation Anchorage Convention District, Anchorage, Alaska, USA. 3-8 May, 2010.

[6] Austin Jensen, YangQuan Chen *Tracking Tagged Fish With Swarming Unmanned Aerial Vehicles Using Fractional Order Potential Fields and Kalman Filtering* 2013 International Conference on Unmanned Aircraft Systems (ICUAS), Grand Hyatt Atlanta, Atlanta, GA. 28-31 May, 2013.

[7] Sousa LL, Lpez-Castejn F, Gilabert J, Relvas P, Couto A, Queiroz N, et al. *Integrated Monitoring of Mola mola Behaviour in Space and Time* PLoS ONE 11(8): e0160404. doi:10.1371/ journal.pone.0160404. 2016.

[8] Gunilla Burrowes and Jamil Y. Khan. A. Cruz . *Autonomous Underwater Vehicles: Chapter 8 Short-Range Underwater Acoustic Communication Networks*. The University of Newcastle Australia InTech, Croatia. 2011.

[9] Kinsler, L., Frey, A., Coppens, A. Sanders, J. *Fundamentals of acoustics*, John Wiley and Sons, 1982.

[10] A. Vermeij and A. Munaf. *A Robust, Opportunistic Clock Synchronization Algorithm for Ad Hoc Underwater Acoustic Networks*. In IEEE Journal of Oceanic Engineering, vol. 40, no. 4, pp. 841-852, Oct. 2015.

[11] Morten Breivik, Vegard E. Hovstein, Thor I. Fossen. *Straight-Line Target Tracking for Unmanned Surface Vehicles*. Modeling, Identification and Control, Vol. 29, No. 4, pp. 131 to 149. 2008.

[12] AutoNaut, *Web page of the producer of AutoNaut USV*. Dec, 2017. http://www.autonautusv.com/

[13] Kongsberg Gruppen, *Web page of the producer of Seaglider AUV*. Dec, 2017. https://www.km.kongsberg.com/ks/web/nokbg0240.nsf/AllWeb/EC2FF8B58CA491A4C1257B870048

[14] OceanScan, *Web page of the producer of LAUV*. Dec, 2017. http://www.oceanscan-mst.com/

[15] Quad H2O Waterproof Multirotors, *Web page of the producer of Hex H2O UAV* Dec, 2017. https://www.quadh2o.com/hexh2o/hexh2o-kit/

[16] Thelma Biotel *Web page of the producer of the acoustic receiver TBR 700* Dec, 2017. http://www.thelmabiotel.com/tbr-700/

[17] C. Forney, E. Manii, M. Farris, M. A. Moline, C. G. Lowe and C. M. Clark *Tracking of a tagged leopard shark with an AUV: Sensor calibration and state estimation*. 2012 IEEE International Conference on Robotics and Automation, Saint Paul, MN, pp. 5315-5321. 2012.

[18] Bjørnar Vik. *Integrated Satellite and Inertial Navigation Systems*. Department of Engineering Cybernetics, NTNU. 2014.

[19] R.E. Kalman *A new approach to linear filtering prediction problems*. ASME Journal of Basic Engineering, Vol. 82, pp. 34-45, 1960.

[20] V. M. Becerra, P. D. Roberts, G. W. Griffiths *Applying the extended Kalman filter to systems described by nonlinear differential-algebraic equations*. Control Engineering Practice, Volume 9, Issue 3, pp 267-281. March 2001.

Appendix

A Tables

A.1 UAV states

States	Value	Description
UAV position	$\mathbf{x}_{UAV} \in \mathbb{R}^3$	Position of the UAV in three-space
UAV velocity	$\mathbf{v}_{UAV} \in \mathbb{R}^3$	Velocity of the UAV in three-space
Active	1/0	The UAV is defined to be active if it is no longer stationed and landed at its host/carrier USV.
Airbourne	1/0	The UAV is defined to be airbourne if it is in the air.
Seconds waited in water τ	[s]	Number of consecutive seconds the UAV have spent landed in water without retrieving a signal.

Table 1: UAV states

A.2 UAV properties

Properties	Value	Description
UAV maximum velocity	$v_{max} \in \mathbb{R}$	The maximium absolute velocity of the UAV
UAV landing time	[s]	The number of seconds it takes to safely land the UAV on the host/carrier USV
UAV takeoff time	[s]	The number of seconds it takes to take off the host/carrier USV

Table 2: UAV properties

A.3 UAV design variables

Design variables	Value	Description
Radius of acceptance	[m]	A radius of acceptance within which it is acceptable for the UAV to land in water
Fly height	[m]	The number of meters above sea level the UAV flies
UAV sick of waiting: τ_{lim}	[s]	The maximum number of seconds the UAV should wait in water without receiving signal before it takes off
Radius of acceptance for landing	[m]	A radius of acceptance for which within the UAV have the ability to land on the USV within UAV landing time. This is a hypothetical value in order to simplify the simulation.
Warning value	x	The warning value is defined as the number consecutive transmitter periods less than three USV have received a signal transmitted signal.
Warning value limit: : $warning_{lim}$	x	While airborne, any warning values lower than this limit will guide the UAV to fly towards its host/carrier USV. While inactive, any warning value equal or larger than this limit will make the UAV take off. Hence, changing from deactive to active state.

Table 3: UAV design variables

A.4 Simulation colors

Object	**Color**	**Marker symbol**
USV positions	Black	Square
UAV inactive on host/carrier USV	Red	Circle
UAV airborne	Yellow	Circle
UAV take off	Red	Arrow
UAV landing	Green	Arrow
True target fish positions	Red	Circle
Estimated target position $\hat{\mathbf{x}}$ for t $\in [0, 1200]$	Yellow	Circle
Estimated target position $\hat{\mathbf{x}}$ for t $\in [1200, 2400]$	Green	Circle
Estimated target position $\hat{\mathbf{x}}$ for t $\in [2400, 3600]$	Pink	Circle
Estimated target position $\hat{\mathbf{x}}$ for t $\in [3600, 4800]$	Blue	Circle
Estimated target position $\hat{\mathbf{x}}$ for t $\in [4800, 6000]$	Black	Circle

Table 4: Color explanation for plots

B Additional plots

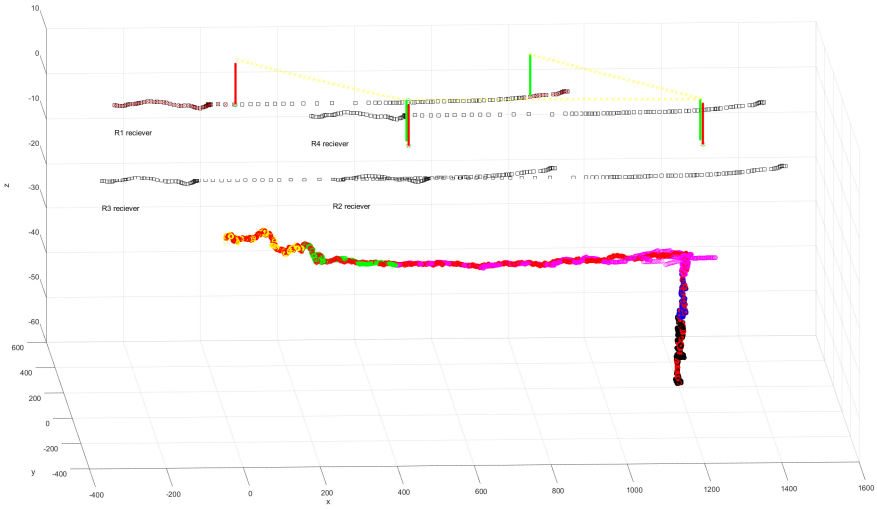

Figure 1: Drone system with UAV tracking target angle II

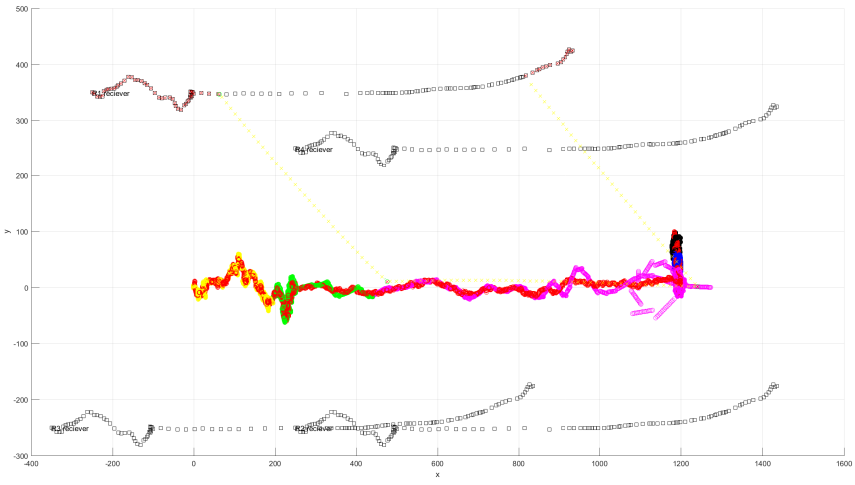

Figure 2: Drone system with UAV tracking target angle III

C Pseudocode

C.1 USV formation control and guidance

Data: USV horizontal positions $\mathbf{x}_{R_{j,i}} \in \mathbb{R}^2$, current way point $\bar{\mathbf{x}}_i \in \mathbb{R}^2$, maximum velocity of the USVs $v_{max} \in \mathbb{R}$, estimated target position projection $\hat{\mathbf{x}}_i \in \mathbb{R}^2$ and radius of acceptance r_a

Result: Desired heading angle $\psi_{d_j} \in \mathbb{R}$ for each of the four USVs: j = 1..4, $\bar{\mathbf{x}}_{i+1}$ and $\mathbf{x}_{R_{j,i+1}}$

if $(euclidean2norm(\bar{\mathbf{x}}, \hat{\mathbf{x}}) > r_a)$ **then**
  for $j \leftarrow 1$ **to** 4 **do**
    $\psi_{d_j} = \text{atan2}([\hat{x} + \mathbf{O}(j,1) - x_{R_j}]\,, [\hat{y} + \mathbf{O}(j,2) - y_{R_j}])$
  end
  for $j \leftarrow 1$ **to** 4 **do**

$$\mathbf{x}_{R_{j,i+1}} = \begin{bmatrix} x_{R_{j,i+1}} \\ y_{R_{j,i+1}} \end{bmatrix} = \begin{bmatrix} x_{R_{j,i}} + sin(\psi_{d_j}) v_{max} \Delta T \\ y_{R_{j,i}} + cos(\psi_{d_j}) v_{max} \Delta T \end{bmatrix}$$

  end
  $\bar{\mathbf{x}}_{i+1} = \hat{\mathbf{x}}_i$;
else
  $\bar{\mathbf{x}}_{i+1} = \bar{\mathbf{x}}_i$
  $\mathbf{x}_{R_{j,i+1}} = \mathbf{x}_{R_{j,i}}$
end

Algorithm 1: Pseudocode for USV formation control and guidance

C.2 UAV situation awareness hierarchy and guidance

Data: All states from 1 for iteration i. All properties from 2. All design variables from 3
Result: All states from 1 for iteration i = i + 1
if $active$ **then**
  if $airbourne$ **then**
    $\tau = 0$
    if $(warning \geq warning_{lim})$ **then**
      $[\mathbf{x}_{UAV,i+1}, \mathbf{v}_{UAV,i+1}, active, airbourne]$ = $guidaceUAVtoUSV()$
    else
      $[\mathbf{x}_{UAV,i+1}, \mathbf{v}_{UAV,i+1}, airbourne]$ = $guidaceUAVtoTarget()$
    end
  else
    if $warning \geq warning_{lim}$ **then**
      if $signal$ **then**
        $\tau = 0$
      else
        $\tau = \tau + \Delta T$
        if $\tau \geq \tau_{lim}$ **then**
          $airbourne = 1$
        end
      end
    else
      $airbourne = 1$
    end
  end
else
  if $warning \geq warning_{lim}$ **then**
    $\mathbf{x}_{uav,i+1} = \mathbf{x}_{R_{j,i+1}}$
  else
    $active = 1$
    $airbourne = 1$
  end
end

Algorithm 2: Situation awareness hierarchy for UAV

D MATLAB Script: Chapter 2 and 3

D.1 TDOA localization in $\mathbb{R}^2$

Listing 1:

```
1 clear ; clc ;
```

```
c = 1484; %[m/s] speed of sound in water
%point locations
A=[0,1];
B = [7,1];
C=[2,5];
D=[5,4];
T = [3,3]; %pretended unknow target.
x_points=  [0,7,2,5,3];
y_points= [1,1,5,4,3];
scatter(x_points,y_points)
hold on
ab = euclid(A(1),A(2),B(1),B(2));
ac = euclid(A(1),A(2),C(1),C(2));
bc = euclid(B(1),B(2),C(1),C(2));
at = euclid(A(1),A(2),T(1),T(2));
bt = euclid(B(1),B(2),T(1),T(2));
ct = euclid(C(1),C(2),T(1),T(2));
dt = euclid(D(1),D(2),T(1),T(2));

%calculating the actual difference of arrival time
    experienced by the receivers. Delta T is the difference
    in distance travelled by the actual signal divided by
    the value of the speed of the signal.
deltaTab = (at-bt) / c;
deltaTac = (at-ct) / c;
deltaTad = (dt-at) / c;

syms x;syms y;
y_solved_ab =   solve(c*deltaTab ==  sqrt( (x-A(1)).^2 + (y
    -A(2)).^2) - sqrt((x-B(1)).^2+(y-B(2)).^2), y);
fplot(y_solved_ab(1))
hold on
fplot(y_solved_ab(2))
hold on

y_solved_ac =   solve(c*deltaTac ==  sqrt( (x-A(1)).^2 + (y
    -A(2)).^2) - sqrt((x-C(1)).^2+(y-C(2)).^2), y);
fplot(y_solved_ac(1))
hold on
fplot(y_solved_ac(2))
hold on

y_solved_ad =   solve(c*deltaTad == sqrt( (x-A(1)).^2 + (y-
    A(2)).^2) - sqrt((x-D(1)).^2+(y-D(2)).^2), y);
fplot(y_solved_ad(1))
```

```
41 hold on
42 fplot(y_solved_ad(2))
43 hold on
44 legend('Time difference R1 and R2: solution 1','Time
       difference R1 and R2: solution 2','Time difference R1
       and R3: solution 1','Time difference R1 and R3: solution
        2','Time difference R2 and R3: solution 1','Time
       difference R2 and R3: solution 2')
45 axis([-2,8,-2,8])
```

D.2 TDOA localization in $\mathbb{R}^3$ with 4 receivers

Listing 2:

```
1  clear;clc; clf;
2  %% Initialization
3  c = 1484;
4  receiver_pos = [0,1,-2;
5  7,1,0;
6  2,5,-4;
7  5,4,-6];
8  target_pos = [3,3,-3];
9  range=zeros(4,1);
10  for j = 1:4
11                  range(j) = euclid3( target_pos,
                        receiver_pos(j,:));
12  end
13 TDOA = zeros(3,j);
14 range_ref = range(1);
15  for j = 2:4
16 TDOA(j-1) = (range(j) - range_ref)/c;
17  end
18
19 %% Calculating true solutions
20 for j = 2:4
21 syms x y z;
22 S = solve(c*TDOA(j-1) == sqrt((x-receiver_pos(j,1))^2+(y-
       receiver_pos(j,2))^2+(z-receiver_pos(j,3))^2) - sqrt((x-
       receiver_pos(1,1))^2+(y-receiver_pos(1,2))^2+(z-
       receiver_pos(1,3))^2), z );
23 f(x,y)=S(1);
24 r=1:0.25:5;
25 [X,Y] = meshgrid(r);
26 Z = double(f(X,Y));
27 surf(X,Y,Z,'FaceAlpha',0.5);
28 zlim([-3,-1]);
29 hold on
```

```
end

%% Plots
scatter3(receiver_pos(:,1),receiver_pos(:,2),receiver_pos
    (:,3),200,'b','filled')
hold on
scatter3(target_pos(1),target_pos(2),target_pos(3),300,'r',
    'filled')
hold on
object_names = {'R1','R2','R3','R4','Target'} ;
cellstring = cellstr(object_names);
for j = 1:4
text(receiver_pos(j,1)-3,receiver_pos(j,2)+2,receiver_pos(j
    ,3), cellstring(j)); %name plotting.
hold on
end
text(target_pos(1)-3,target_pos(2)+2,target_pos(3),
    cellstring(5)); %name plotting.
axis([-2,10,-2,10,-8,2])
xlabel('x')
ylabel('y')
zlabel('z')
```

D.3 TDOA localization in $\mathbb{R}^3$ with 3 receivers and a depth measurement

Listing 3:

```
clear;clc; clf;
%% Initialization
c = 1484;
receiver_pos = [0,1,-2;
7,1,0;
2,5,-0.5 ;];
target_pos = [3,3,-3];
range=zeros(4,1);
for j = 1:3
        range(j) = euclid3( target_pos , receiver_pos(j,:));
end
TDOA = zeros(3,j);
range_ref = range(1);
for j = 2:3
TDOA(j-1) = (range(j) - range_ref)/c;
end

%% Calculating true solutions
```

```
for j = 2:3
syms x y z;
S = solve(c*TDOA(j-1) == sqrt((x-receiver_pos(j,1))^2+(y-
    receiver_pos(j,2))^2+(z-receiver_pos(j,3))^2) - sqrt((x-
    receiver_pos(1,1))^2+(y-receiver_pos(1,2))^2+(z-
    receiver_pos(1,3))^2), z );
f(x,y)=S(1);
r= 1:0.25:5;
[X,Y] = meshgrid(r);
Z = double(f(X,Y));
surf(X,Y,Z,'FaceAlpha',0.5);
hold on
end
[X,Y] = meshgrid(-2:0.3:10,-2:0.3:10);
height = 0.2;
Z = height*sin(X) + height*cos(Y);
C = X*Z*Y;
s= surf(X,Y,Z,'FaceAlpha',0.1);

%% Depth
r= 1:0.25:5;
[X,Y] = meshgrid(r);
Z = ones(size(X))*target_pos(3);
surf(X,Y,Z,'FaceAlpha',0.4);

%% Plots
scatter3(receiver_pos(:,1),receiver_pos(:,2),receiver_pos
    (:,3),200,'b','filled')
hold on
scatter3(target_pos(1),target_pos(2),target_pos(3),300,'r',
    'filled')
hold on
object_names = {'R1','R2','R3','Target'} ;
cellstring = cellstr(object_names);
dx=-0.;
for j = 1:3
text(receiver_pos(j,1)+dx,receiver_pos(j,2)+dx,receiver_pos
    (j,3)-0.5, cellstring(j)); %name plotting.
hold on
end
text(target_pos(1)-0.5,target_pos(2)-0.5+dx,target_pos(3)
    -0.5, cellstring(4)); %name plotting.
axis([-2,10,-2,10,-8,2])
xlabel('x')
ylabel('y')
```

```
zlabel('z')
```

D.4 Single measurement localization in $\mathbb{R}^3$

Listing 4:

```
clear;clc; clf;
%% Initialization
c = 1484;
receiver_pos = [0,0,0];
target_pos = [3,3,-6];
domeRadius = euclid3( target_pos, receiver_pos);
%% Dome plot
[x,y,z] = sphere(20);
xEast  = domeRadius * x;
yNorth = domeRadius * y;
zUp    = domeRadius * z;
zUp(zUp < 0) = 0;
figure('Renderer','opengl')
surf(xEast, yNorth, -zUp,'FaceColor','yellow','FaceAlpha'
    ,0.3)
hold on
[X,Y] = meshgrid(-10:0.3:10,-10:0.3:10);
height = 0.2;
Z = height*sin(X) + height*cos(Y);
s= surf(X,Y,Z,'FaceAlpha',0.1);
%% Disc plot
radius  = sqrt(target_pos(1)^2 + target_pos(2)^2);
[T,R] = meshgrid(linspace(0,2*pi,64),linspace(0,radius,16))
    ;
X = R.*cos(T);
Y = R.*sin(T);
Z = ones(size(X))*target_pos(3);
surf(X,Y,Z)
%% Receiver and transmitter plot
scatter3(receiver_pos(:,1),receiver_pos(:,2),receiver_pos
    (:,3),400,'b','filled')
hold on
scatter3(target_pos(1),target_pos(2),target_pos(3),400,'r',
    'filled')
hold on
object_names = {'R1','Target'} ;
cellstring = cellstr(object_names);
dx=-0.;
text(receiver_pos(1)+1+dx,receiver_pos(2)+dx,receiver_pos
    (3)-2, cellstring(1)); %name plot.
hold on
```

```
37 text(target_pos(1)+1,target_pos(2)+dx,target_pos(3)+1,
       cellstring(2)); %name plot.
38 hold on
39 %% Red line circle plot
40 theta = 0:pi/50:2*pi;
41 xunit = radius * cos(theta);
42 yunit = radius * sin(theta);
43 h = plot3(xunit, yunit, target_pos(3)*ones(size(theta)),'r'
       );
44 set(h,'linewidth',3);
45 axis([-10,10,-10,10,-10,10])
46 xlabel('x')
47 ylabel('y')
48 zlabel('z')
```

D.5 AUV glider formation

Listing 5:

```
1  %% Initialization
2  iterations =  28000 ;
3  x_vec = linspace(0,7000,iterations);
4  A =50;
5  frec = (2*pi) / 28000;
6  phaseShiftVec =[0;pi/4;pi/2;3*pi/4;pi;5*pi/4; 3*pi/2;7*pi
       /4];
7  time_step = 1;
8  z = zeros(8,iterations/time_step);
9  %% create AUV glider and target trajectories
10 for i = 1:iterations/time_step +1
11 for j = 1:8
12 z(j,i) = -100 + A*sin(frec*i+phaseShiftVec(j));
13 end
14 end
15 x_vec_save = x_vec;
16 y = [0,0,1,-1, 0.70711,- 0.70711,- 0.70711, 0.70711]*100;
17 deflection = 0.70711*100;
18 x_vec = [x_vec+100;x_vec-100;x_vec;x_vec;x_vec+deflection;
       x_vec+deflection;x_vec-deflection;x_vec-deflection];
19 receiver_pos = zeros(8,3,iterations);
20 x_hat = zeros(1,3,iterations);
21 x_hat(1,1,:) = x_vec_save;
22 for i = 1:iterations
23 x_hat(1,3,:) = -100;
24 end
25 receiver_pos(:,1,:) = x_vec;
26 for i = 1:iterations
```

```
27 receiver_pos(:,2,i) = y';
28 receiver_pos(:,3,i) = z(:,i);
29 end
30
31 %% Calculate dilution of precition values
32 j_vec = create_j_vec([1;1;1;1;1;1;1;1]); %All receivers
       obtain measurements
33 GramMatrix_inv_data = zeros(3,3,iterations);
34 for i = 1:iterations
35 h2 = createJacobian(x_hat(1,:,i), receiver_pos(:,:,i),j_vec
       );
36 GramMatrix_inv  = inv(h2' *h2);
37 GramMatrix_inv_data(:,:,i) = GramMatrix_inv;
38 end
39
40 gdop_values=zeros(3,iterations);
41 for i =1:iterations
42 for j = 1:3
43 gdop_values(j,i) = GramMatrix_inv_data(j,j,i);
44 end
45 end
46
47 sum_gdop=zeros(1,iterations);
48 for i = 1:iterations
49 sum_gdop(i) = sum(gdop_values(:,i));
50 end
51 %% Plots
52 figure(1)
53 tot=6;
54 subplot(tot,1,1)
55 time = [0:time_step:iterations];
56 time = time/3600; %convert to hours
57 rex = zeros(8,iterations);
58 for i = 1:iterations
59 for j = 1:8
60 rex (j,i)= receiver_pos(j,1,i);
61 end
62 end
63 for j = 1:8
64 plot(rex(j,:),z(j,1:28000)');
65 hold on
66 end
67 legend('AUV 1', 'AUV 2', 'AUV 3', 'AUV 4', 'AUV 5', 'AUV 6'
       , 'AUV 7', 'AUV 8')
68 xlabel('[x]')
```

```
69  ylabel('[depth]')
70  subplot(tot,1,2)
71  time = [0:time_step:iterations];
72  time = time/3600;
73  plot(time,z(1,:))
74  hold on
75  plot(time,z(2,:))
76  hold on
77  plot(time,z(3,:))
78  hold on
79  plot(time,z(4,:))
80  hold on
81  plot(time,z(5,:))
82  hold on
83  plot(time,z(6,:))
84  hold on
85  plot(time,z(7,:))
86  hold on
87  plot(time,z(8,:))
88  legend('AUV 1', 'AUV 2', 'AUV 3', 'AUV 4', 'AUV 5', 'AUV 6'
        , 'AUV 7', 'AUV 8')
89  xlabel('[hour]')
90  ylabel('[depth]')
91
92  subplot(tot,1,3)
93  plot(time(1:28000), gdop_values(1,:))
94  legend ('GDOP in z')
95  xlabel('[hour]')
96  ylabel('GDOP')
97  hold on
98
99  subplot(tot,1,4)
100 plot(time(1:28000), gdop_values(2,:))
101 legend ('GDOP in y')
102 xlabel('[hour]')
103 ylabel('GDOP')
104 hold on
105
106 subplot(tot,1,5)
107 plot(time(1:28000), gdop_values(3,:))
108 legend ('GDOP in z')
109 xlabel('[hour]')
110 ylabel('GDOP')
111
112 subplot(tot,1,6)
```

```
113 plot(time(1:28000), sum_gdop(1,:))
114 legend ('Sum of GDOP in x,y and z')
115 xlabel('[hour]')
116 ylabel('GDOP')
117
118 figure(2)
119 %initial constillation
120 stri = {'Target','AUV1','AUV2','AUV3','AUV4','AUV5','AUV6',
        'AUV7','AUV8'} ;
121 cellstring = cellstr(stri);
122 scatter3(0,0,-100)
123 text(0,0,-100,cellstring(1));
124 i=1;
125 for j = 1:8
126 scatter3(receiver_pos(j,1,i),receiver_pos(j,2,i),
        receiver_pos(j,3,i));
127 text(receiver_pos(j,1,i),receiver_pos(j,2,i),receiver_pos(j
        ,3,i),cellstring(j+1));
128 hold on
129 end
130 xlabel('x')
131 ylabel('y')
132 zlabel('z')
```

E MATLAB Script: Fish tracking simulation, 4 USVs and 1 UAV

E.1 Run simulation

Listing 6:

```
1  %% 4USV + UAV Fish Trajectory Tracking
2  clear;clc;clf;
3
4  %% Initializing General Constants
5  iterations = 12000; %6000 seconds
6  c = 1484;  %speed of sound
7  noise_power=0.0000005;
8  time_step = 0.5;
9  signal_range = 500; %Acoustic transission range
10 signal_period = 8; % Acoustic transmittor period [s]
11
12 %% Transmitter signal pulse
13 signal_pulse =zeros(1, iterations);
14 for i=16:signal_period/time_step:iterations
15     signal_pulse (i) = 1;
```

```
16 end
17
18 %% Fish trajectory
19 %CreateFishTrajectory()
20 target_pos_load = load('target_pos.mat'); %The trajectory
       from case study
21 target_pos= target_pos_load.target_pos1;
22
23 %% Guidance initialization
24 usv_pos = zeros (4, 3);
25 usv_pos_measurements = zeros(4,3,iterations);
26 converg_points= zeros(3, iterations);
27 uav_pos =  [-250  350  0]; % Initially hosted on USV
28 uav_pos_measurements = zeros(1,3, iterations);
29 uav_pos_measurements(1,:,1) = uav_pos;
30 radius_of_acceptance_landing = 30;
31 radius_of_acceptance_target = 3;
32 radius_of_acceptance_vehicles = 10;
33 currentConvergPoint = zeros(3,1);
34 velocity_USVs = 2;
35 velocity_UAV = 15;
36 radius_of_acceptance_target_UAV = 50;
37 uav_sick_of_waiting = 100;
38
39 %% Initial position of USVs
40 A=[-250, 350, 0];
41 B = [250, -250, 0];
42 C=[-350, -250, -0];
43 D=[250, 250, -0];
44 offset = [A(1),A(2);
45 B(1), B(2);
46 C(1), C(2);
47 D(1), D(2)];
48 usv_pos (1,:)= [A(1),A(2),A(3)];
49 usv_pos (2,:)= [B(1),B(2),B(3)];
50 usv_pos (3,:)= [C(1),C(2),C(3)];
51 usv_pos (4,:)= [D(1),D(2),D(3)];
52 usv_pos_measurements (:,:,1) = usv_pos;
53 %% Depth measurements without noise
54 depth = zeros(1, iterations);
55 depth(1,:) = target_pos(3,:);
56 %% TOA measurements without noise a
57 TOA =zeros(4,1);
58 range_real = zeros(4,1);
59 pseudorange = zeros(4,1);
```

```
60 TOA_measurements = zeros(5,iterations);
61 %% initializing the Kalman filter
62 x_hat = zeros(6,1); % [x y z x_dot y_dot z_dot]'
63 x_hat(1) =target_pos(1)-1; x_hat(2) = target_pos(2)-1;
       x_hat(3) = target_pos(3)-1; %initia
64 x_hat_measurements = zeros(6,iterations); %for data storage
65 x_hat_measurements(1:3,1) = x_hat(1:3);
66 D = time_step*eye(6);
67 F = [1 0 0 time_step 0 0;
68          0 1 0 0 time_step 0;
69          0 0 1 0 0 time_step
70          0 0 0 1 0 0
71          0 0 0 0 1 0
72          0 0 0 0 0 1];
73 P_hat = eye(6)*0.01;
74 Q = eye(6)*0.01;
75 Q(3,3)=1000;
76 x_hat_pred = x_hat;
77 H=zeros(4,6);
78 uav_active = 0;
79 uav_in_air = 0;
80 warning = 0; %initial warning value
81 iterations_waited = 0;
82 uav_in_air_data = zeros(1,iterations);
83 iterations_waited_qount = zeros(1,iterations);
84 distanceUAVtoTarget = zeros(1,iterations);
85 UAV_signal_received_measurements= zeros(1,iterations);
86 warning_data=zeros(1,iterations);
87 uav_landing_data=zeros(1,iterations);
88 uav_take_off_data=zeros(1,iterations);
89 uav_active_data = zeros(1,iterations);
90 available_measurements = zeros(5,iterations);
91
92 %% Simulation loop
93 for i = 2:iterations
94
95 %Saving x_hat measurement receivers position
96 usv_pos_measurements(:,:,i) = usv_pos;
97 uav_pos_measurements(1,:,i) = uav_pos;
98 x_hat_measurements(:,i) = x_hat;
99 converg_points(:,i)= currentConvergPoint;
100 %Creating pos vec
101 x_hat_pos = [x_hat(1),x_hat(2),x_hat(3)];
102 distanceUAVtoTarget(i)= (euclid3(uav_pos, target_pos(:,i))
       );
```

```
103 %% UAV
104 UAV_signal_received = 0; %auto
105 iterations_waited_qount(i) = iterations_waited;
106
107 if (uav_active)
108         if (uav_in_air)
109             iterations_waited = 0;
110             uav_pos(3)=10;
111             uav_in_air_data(i) = 1; %saves for plotting.
112                 if (warning<3) %we have more than two usv
                        recieving target signal and can fly back
113                 [uav_pos, uav_in_air, uav_active,
                        uav_landing] = guidanceUAVtoUSV(uav_pos,
                         usv_pos, time_step, velocity_UAV,
                        radius_of_acceptance_landing);
114                 uav_landing_data(i) = uav_landing;
115                 else
116                 [uav_pos, uav_in_air,uav_landing] =
                        guidanceUAVtoTarget(x_hat, uav_pos,
                        time_step, velocity_UAV,
                        radius_of_acceptance_target_UAV);
117                 uav_landing_data(i) = uav_landing;
118                 end
119
120         else %UAV is in water landed
121             if (warning>1)
122               uav_pos(3)=0;
123                 if ((euclid3(uav_pos, target_pos(:,i)))
                        <500 && signal_pulse(i))
124                     UAV_signal_received = 1;
125                     iterations_waited = 0; %reset and
                            wait intill no signal is
                            reached or 3 usv have singal
                            received..
126                   else
127                   iterations_waited = iterations_waited
                        + 1;
128                     if (iterations_waited >
                            uav_sick_of_waiting)
129                      uav_in_air = 1;
130                      uav_take_off_data(i) = 1;
131                      %
                                radius_of_acceptance_target_UAV
                                 =
                                radius_of_acceptance_target_UAV
```

```
                                        ;
132                                 end
133                             end
134                 else %The situation is concidered to be under
                        control
135                     uav_in_air = 1; %UAV takeoff for flight
                            home
136                     uav_take_off_data(i) = 1;
137                 end
138             end
139
140     else
141         if (warning<6)
142             uav_pos = usv_pos(1,:); % UAV stays on 1st USV in
                    the case of no urgency regarding loosing target
143         else
144             uav_active = 1;
145             uav_in_air =1;%take off
146             uav_take_off_data(i) = 1;
147         end
148     end
149
150     UAV_signal_received_measurements(i) = UAV_signal_received;
151     warning_data(i) = warning;
152     uav_active_data(i) = uav_active;
153
154 for n = 1:4
155 range_real (n) = euclid3( target_pos(:,i)', usv_pos(n,:));
156 TOA (n) =range_real(n)/c + wgn(1,1,1)*noise_power +
        time_step*i;
157 end
158
159 if (UAV_signal_received)
160 range_real(5) = euclid3(target_pos(:,i),uav_pos);
161 TOA(5) = range_real(5)/c + wgn(1,1,1)*noise_power +
        time_step*i;
162 pseudorange (5) = euclid3( x_hat_pos, uav_pos);
163 TOA_measurements (5,i)=TOA(5);
164 available_measurements(5,i) = 1;
165 end
166
167 if (signal_pulse(i))
168 TOA_measurements (1:4,i)=TOA(1:4);
169 end
170
```

```
171 if (signal_pulse(i))
172 m_available = m_avail(range_real,signal_range);
173 m_available_usv = m_avail(range_real(1:4),signal_range);
174     if (sum(m_available_usv) < 3)
175     warning = warning + 1;
176     else
177     warning = 0;
178     end
179 else
180 m_available = zeros(4,1);
181 end
182
183 j_vec = create_j_vec(m_available);
184 [m,n] = size(j_vec);
185
186 for n = 1:4
187 pseudorange (n) = euclid3( x_hat_pos, usv_pos(n,:));
188 end
189
190 if ((sum(j_vec) ==0))
191 y_hat = 0;
192 y = 0;
193 else if ((dybdemaaling(i) ==1) && (sum(j_vec) ==0) )
194 y_hat = 0;
195 y = 0;
196 H = [];
197 else
198 y_hat = produce_y_hat(pseudorange,j_vec);
199 y = TDOA(TOA,j_vec) *c;
200
201 if (UAV_signal_received)
202 H =createJacobian(x_hat_pos, [usv_pos;uav_pos],j_vec);
203 else
204 H =createJacobian(x_hat_pos, [usv_pos],j_vec);
205 end
206 end
207
208
209 for j = 1:4
210 if(range_real(j)<500 && signal_pulse(i))
211 dybdemaaling(i) =1;
212 available_measurements(j,i) = 1;
213 end
214 end
215
```

```
216 if (dybdemaaling(i))
217 R = eye(m);
218 R(m,m) = 0.001; % We trust the depth measurement alot
219 y_hat(m,1) = target_pos(3,i) +wgn(1,1,1)*noise_power*0.1;
220 y(m,1) = x_hat_pred(3);
221 H(m,:) = [0 0 1 0 0 0];
222 else
223 R = eye(m-1);
224 end
225 end
226 %% Kalman filter equations
227 %Time update
228 x_hat_pred = F*x_hat;
229 P_pred = F*P_hat*F' + D*Q*D';
230 if ((sum(j_vec) + dybdemaaling(i))>0)
231 %Measurement update
232 K = P_pred*H'*inv(H*P_pred*H' + R);
233 P_hat = P_pred - K*H*P_pred;
234 else
235 K = 0;
236 end
237 x_hat = x_hat_pred + K*(y_hat-y);
238
239 %% USV Guidance
240 [ usv_pos, currentConvergPoint ] = USVguidance( x_hat(1:3),
      currentConvergPoint, usv_pos, time_step, velocity_USVs,
      radius_of_acceptance_target,offset );
241
242 end %End of simulation loop
243
244
245 %% True velocities of target
246 target_velocity = zeros(3,iterations);
247 for i = 2:iterations
248     for j = 1:3
249         target_velocity(j,i) = (1/time_step)*(target_pos(j,
              i)-target_pos(j,i-1));
250     end
251 end
252
253 %% Creating error vectors
254 error = [target_pos;target_velocity] - x_hat_measurements;
255
256 %% Plots
257 time = [time_step:time_step:time_step*iterations];
```

```
%% Erros plots (estimated target position errors)
figure(1)
tot=6;
title('Error plots')
subplot(tot,1,1)
plot (time,error(1,:))
xlabel ('[s]')
ylabel('[m]')
legend('$\tilde{x} = \hat{x} - x$')
set(legend,'Interpreter','latex')

subplot (tot,1,2)
plot (time,error(2,:))
xlabel ('[s]')
ylabel('[m]')
legend('$\tilde{y} = \hat{y} - y$')
set(legend,'Interpreter','latex')

subplot (tot,1,3)
plot (time,error(3,:))
xlabel ('[s]')
ylabel('[m]')
legend('$\tilde{z} = \hat{z} - z$')
set(legend,'Interpreter','latex')

subplot(tot,1,4)
plot (time,error(4,:))
xlabel ('[s]')
ylabel('[m/s]')
legend('$\tilde{u} = \hat{u} - u$')
set(legend,'Interpreter','latex')

subplot (tot,1,5)
plot (time,error(5,:))
xlabel ('[s]')
ylabel('[m/s]')
legend('$\tilde{v} = \hat{v} - v$')
set(legend,'Interpreter','latex')

subplot (tot,1,6)
plot (time,error(6,:))
xlabel ('[s]')
ylabel('[m/s]')
legend('$\tilde{w} = \hat{w} - w$')
set(legend,'Interpreter','latex')
```

```
303
304 %% 3D figure plots
305 figure (2)
306 % USV name plot
307 stri = {'R1 receiver','R2 receiver','R3 receiver','R4
        receiver'} ;
308 cellstring = cellstr(stri);
309 stri2 = {'1,5 [s]',',1498 [s]','2997,5 [s]'} ;
310 cellstring2 = cellstr(stri2);
311 i = 3;
312 for j = 1:4
313         text(usv_pos_measurements(j,1,i),
                usv_pos_measurements(j,2,i),usv_pos_measurements
                (j,3,i)-6, cellstring(j))
314         hold on
315 end
316
317 % Color plot for estimated target position
318 rest = mod( iterations, numbersOfColors);
319 antall = iterations - rest;
320 hver = antall/numbersOfColors;
321 scatter3(x_hat_measurements(1,1:hver),x_hat_measurements
        (2,1:hver),x_hat_measurements(3,1:hver),'y')
322 hold on
323 scatter3(x_hat_measurements(1,hver:2*hver),
        x_hat_measurements(2,hver:2*hver),x_hat_measurements(3,
        hver:2*hver),'g')
324 hold on
325 scatter3(x_hat_measurements(1,2*hver:3*hver),
        x_hat_measurements(2,2*hver:3*hver),x_hat_measurements
        (3,2*hver:3*hver),'m')
326 hold on
327 scatter3(x_hat_measurements(1,3*hver:4*hver),
        x_hat_measurements(2,3*hver:4*hver),x_hat_measurements
        (3,3*hver:4*hver),'b')
328 hold on
329 scatter3(x_hat_measurements(1,4*hver:5*hver+rest),
        x_hat_measurements(2,4*hver:5*hver+rest),
        x_hat_measurements(3,4*hver:5*hver),'k')
330 hold on
331 scatter3(target_pos(1,1:iterations),target_pos(2,1:
        iterations),target_pos(3,1:iterations),'r')
332
333 %USV plots
334 hold on
```

```
335 for i = 1:100:iterations
336     for j = 1:4
337     scatter3(usv_pos_measurements(j,1,i) ,
            usv_pos_measurements(j,2,i),usv_pos_measurements(j
            ,3,i),'k','square')
338     hold on
339     end
340     scatter3(uav_pos_measurements(1,1,i) ,
            uav_pos_measurements(1,2,i),uav_pos_measurements
            (1,3,i),'x','r')
341     hold on
342 end
343
344 % Ploting UAV position
345 for i = 1:2:iterations
346     if (uav_in_air_data(i))
347         scatter3(uav_pos_measurements(1,1,i) ,
                uav_pos_measurements(1,2,i),uav_pos_measurements
                (1,3,i) ,'x','y')
348         hold on
349     end
350 end
351
352 for i = 1:2:iterations
353     if (uav_in_air_data(i)==0 && uav_active_data(i) )
354         scatter3(uav_pos_measurements(1,1,i) +noise_power*
                wgn(1,1,1) ,uav_pos_measurements(1,2,i)+
                noise_power*wgn(1,1,1),uav_pos_measurements(1,3,
                i) +noise_power*wgn(1,1,1),'o','g')
355         hold on
356     end
357 end
358
359 % Plot take off and landing arrows
360 for i =1:iterations
361     if (uav_take_off_data(i))
362         a= [uav_pos_measurements(1,1,i),
                uav_pos_measurements(1,2,i),0];
363         b = [uav_pos_measurements(1,1,i),
                uav_pos_measurements(1,2,i),10];
364         a_b = b-a;
365         quiver3( a(1), a(2), a(3), a_b(1), a_b(2),a_b(3),'r
                ','linewidth',3)
366         hold on
367     end
```

```
368
369     if (uav_landing_data(i))
370         a = [uav_pos_measurements(1,1,i),
                uav_pos_measurements(1,2,i),10];
371         b = [uav_pos_measurements(1,1,i),
                uav_pos_measurements(1,2,i),0];
372         a_b = b-a;
373         quiver3( a(1)+1, a(2)+1, a(3), a_b(1), a_b(2),a_b
                (3),'g','linewidth',3)
374         hold on
375     end
376 end
377 xlabel('x')
378 ylabel('y')
379 zlabel('z')
380 grid on
381
382 %% UAV data
383 figure(3)
384 tot =5;
385 time = [time_step:time_step:time_step*iterations];
386 subplot(tot,1,1)
387 plot(time,distanceUAVtoTarget)
388 ylabel('[m]')
389 xlabel('Time [s]')
390 legend('Euclidian distance UAV to $\hat{x}$');
391 set(legend,'Interpreter','latex')
392
393 subplot(tot,1,2)
394 plot(time,uav_in_air_data)
395 xlabel('[s]')
396 legend('UAV airbourne')
397 set(legend,'Interpreter','latex')
398
399 subplot(tot,1,3)
400 plot(time,iterations_waited_qount*time_step)
401 ylabel('[s]')
402 xlabel('[s]')
403 legend('Seconds UAV has waited in water')
404 set(legend,'Interpreter','latex')
405
406 subplot(tot,1,4)
407 xlabel('Time [s]')
408 plot(time,UAV_signal_received_measurements)
409 legend('UAV signal received')
```

```
410 set(legend,'Interpreter','latex')
411
412 subplot(tot,1,5)
413 xlabel('Time [s]')
414 plot(time, warning_data)
415 legend('Warning value')
416 set(legend,'Interpreter','latex')
417
418
419 %% Ploting real and estimated target position
420 figure(4)
421 tot=6;
422 title('Real and estimated values')
423 subplot(tot,1,1)
424 plot (time,x_hat_measurements(1,:))
425 hold on
426 plot(time, target_pos(1,:))
427 hold on
428 xlabel ('[s]')
429 ylabel('[m]')
430 legend('$\hat{x}$', '$x$')
431 set(legend,'Interpreter','latex')
432
433
434 subplot (tot,1,2)
435 plot (time,x_hat_measurements(2,:))
436 hold on
437 plot(time, target_pos(2,:))
438 hold on
439 xlabel ('[s]')
440 ylabel('[m]')
441 legend('$\hat{y}$', '$y$')
442 set(legend,'Interpreter','latex')
443
444 subplot (tot,1,3)
445 plot (time,x_hat_measurements(3,:))
446 hold on
447 plot(time, target_pos(3,:))
448 hold on
449 xlabel ('[s]')
450 ylabel('[m]')
451 legend('$\hat{z}$', '$z$')
452 set(legend,'Interpreter','latex')
453
454
```

```
455 subplot(tot,1,4)
456 plot (time,x_hat_measurements(4,:))
457 hold on
458 plot(time, target_velocity(1,:))
459 hold on
460 xlabel ('[s]')
461 ylabel('[m/s]')
462 legend('$\hat{u}$', '$u$')
463 set(legend,'Interpreter','latex')
464
465
466 subplot (tot,1,5)
467 plot (time,x_hat_measurements(5,:))
468 hold on
469 plot(time, target_velocity(2,:))
470 hold on
471 xlabel ('[s]')
472 ylabel('[m/s]')
473 legend('$\tilde{v}$','$v$')
474 set(legend,'Interpreter','latex')
475
476 subplot (tot,1,6)
477 plot (time,x_hat_measurements(6,:))
478 hold on
479 plot(time, target_velocity(3,:))
480 hold on
481 xlabel ('[s]')
482 ylabel('[m/s]')
483 legend('$\tilde{w}$','$w$')
484 set(legend,'Interpreter','latex')
485
486 %% Available measurements plot
487 figure(5)
488
489 tot=5;
490 subplot(tot,1,1)
491 plot (time,available_measurements(1,:))
492 hold on
493 xlabel ('[s]')
494 legend('$USV R1$')
495 set(legend,'Interpreter','latex')
496
497 subplot (tot,1,2)
498 plot (time,available_measurements(2,:))
499 hold on
```

```
xlabel ('[s]')
legend('$USV R2$')
set(legend,'Interpreter','latex')

subplot (tot,1,3)
plot (time,available_measurements(3,:))
hold on
xlabel ('[s]')
legend('$USV R3$')
set(legend,'Interpreter','latex')

subplot(tot,1,4)
plot (time,available_measurements(4,:))
hold on
xlabel ('[s]')
legend('$USV R4$')
set(legend,'Interpreter','latex')

subplot(tot,1,5)
plot (time,available_measurements(5,:))
hold on
xlabel ('[s]')
legend('$UAV$')
set(legend,'Interpreter','latex')
```

E.2 Create fish trajectory

Listing 7:

```
%% Initialization & Allocating Space
target_pos= zeros(3,iterations); %Fish position
target_pos(3)=-20; % Initial depth
d_target_pos  = zeros(3,iterations);
w=zeros(3,1); %Gausian white noise
T=10; %Markov time constant
noise_power=zeros(3,1);
noise_power(1) =0.3;
noise_power(2) =0.9;
noise_power(3) =0.05;
target_pos_out = zeros(3,iterations);

%% Creating piecewise trajectory with varying speed in x-
    direction
%Create a fish trajectory with high speed first half of
    simualtion
iterations = iterations/2;
x=zeros(1,iterations);
```

```
17 x(1:iterations/2)= linspace (0,250,iterations/2);
18 x((iterations/2):((3*iterations)/4)-1)= linspace (250,260,(
       iterations/4));
19 x(3*iterations/4:iterations) = linspace(260,1200,iterations
       /4 +1);
20 x(iterations:iterations*2)=linspace(1200,1200,iterations+1)
       ;
21
22 %% Creating trajectory in y and z direction
23 %Path consisting of piecewise sinus waves and slowly
       varying Markov chain
24 Amplitude=1;
25 y = Amplitude*sin(x/100);
26 for i = 2:half_of_iterations
27 for j= 1:3
28     w(j) = wgn(1,1,1) * noise_power(j);
29     d_target_pos(j,i) =  (1/T)* target_pos(j,i-1)  + w(j);
30     target_pos(j,i) = target_pos(j,i-1) + d_target_pos(j,i)
           ;
31 end
32 end
33
34 for i = 2:half_of_iterations
35 target_pos_out(1,i) = target_pos(1,i) + x(i);
36 target_pos_out(2,i) = target_pos(2,i) + y(i);
37 end
38 target_pos_out(3,:) = target_pos(3,:);
39
40 %% Plot path
41 figure(1)
42 scatter3(target_pos_out(1,:),target_pos_out(2,:),
       target_pos_out(3,:))
43 axis([min(target_pos_out(1,:)), max(target_pos_out(1,:)),
       min(target_pos_out(2,:)), max(target_pos_out(2,:)), min(
       target_pos_out(3,:)), max(target_pos_out(3,:))])
44 xlabel('x');
45 ylabel('y')
46 zlabel('z')
47 axis equal
48 legend('Fish movement')
```

E.3 Available measurements

Listing 8:

```
1 function [ m_avail_vec ] = m_avail(range_real, signal_range
      )
```

```
[m,n] =  size(range_real);
m_avail_vec = zeros(m,1);
    for j =1:m
        if (range_real(j) <= signal_range)
            m_avail_vec(j) = 1;
        end
    end
end
```

E.4 Create vector j-vec from available measurement

Listing 9:

```
function [ j_vec ] = create_j_vec( m_avail )
[m,n] = size(m_avail);
j_vec=[];
for i=1:m
    if (m_avail(i))
        j_vec=[j_vec;i];
    end
end
end
```

E.5 Euclidean distance in 3-space

Listing 10:

```
function [d] = euclid3(A,B)
    d=sqrt((A(1)-B(1))^2+(A(2)-B(2))^2+(A(3)-B(3))^2);
end
```

E.6 UAV guidance to target

Listing 11:

```
function [ uav_pos_out, uav_in_air_out,uav_landing ] =
    guidanceUAVtoTarget( x_hat, uav_pos, time_step,
    velocity_UAV, radius_of_acceptance_target_UAV )
temp = uav_pos;

%% guidance function
deltaX = x_hat(1) - uav_pos(1);
deltaY = x_hat(2) - uav_pos(2);
heading_des = atan2(deltaX,deltaY);
temp(1) = temp(1) + sin(heading_des)*velocity_UAV*time_step
    ;
temp(2) = temp(2) + cos(heading_des)*velocity_UAV*time_step
    ;
```

```
10 uav_pos_out  = temp;
11
12 if ((euclid3(uav_pos, x_hat)) >
       radius_of_acceptance_target_UAV)
13     uav_in_air_out=1;
14     uav_landing = 0;
15 else
16     uav_in_air_out=0; % UAV lands
17     uav_landing = 1;
18 end
19
20 end
```

E.7 UAV guidance to USV

Listing 12:

```
1  function   [uav_pos_out, uav_in_air_out, uav_active_out,
       uav_landing] = guidanceUAVtoUSV(uav_pos,usv_pos,
       time_step, velocity_UAV, radius_of_acceptance_landing);
2  temp = uav_pos;
3
4  %% guidance function
5  deltaX =usv_pos(1,1) - uav_pos(1);
6  deltaY = usv_pos(1,2) - uav_pos(2);
7  heading_des = atan2(deltaX,deltaY);
8  temp(1) = temp(1) + sin(heading_des)*velocity_UAV*time_step
       ;
9  temp(2) = temp(2) + cos(heading_des)*velocity_UAV*time_step
       ;
10 uav_pos_out  = temp;
11
12 if ((euclid3(uav_pos, usv_pos(1,:))) >
       radius_of_acceptance_landing)
13 uav_in_air_out=1;
14 uav_active_out = 1;
15 uav_landing = 0;
16 else
17 uav_in_air_out = 0; % UAV not airborne
18 uav_active_out = 0; % UAV safely landed in USV.
19 uav_landing = 1; %UAV landed
20 end
21 end
```

E.8 USV formation control and guidance

Listing 13:

```
function [ receivers_pos_out, currentConvergPoint_out ] =
    guidance2( x_hat, currentConvergPoint, receiver_pos,
    time_step, velocity_USVs, radius_of_acceptance_target,
    offset )

if ((euclid3(currentConvergPoint, x_hat)) >
    radius_of_acceptance_target)
temp = receiver_pos;
currentConvergPoint = x_hat;
currentConvergPoint_out = currentConvergPoint;
heading_des = zeros(4,1);
target_pos = zeros (4,2);

    for j = 1:4
        for i = 1:2
        target_pos(j,i)= currentConvergPoint(i) + offset(j,
            i); %offset formation matrix
        end
    end
    %inherit depth:
    target_pos(:,3) =  receiver_pos(:,3);
    %% guidance function
    for j = 1:4
    deltaX = target_pos(j,1) - receiver_pos (j,1);
    deltaY = target_pos(j,2) - receiver_pos (j,2);
    heading_des(j) = atan2(deltaX,deltaY);
    temp(j,1) = temp(j,1) + sin(heading_des(j))*
        velocity_USVs*time_step;
    temp(j,2) = temp(j,2) + cos(heading_des(j))*
        velocity_USVs*time_step;
    %  end
    end
    receivers_pos_out  = temp;
    return;
    end
    currentConvergPoint_out = currentConvergPoint;
    receivers_pos_out = receiver_pos;
end
```