FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

U. PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Persistent tracking of ocean sharks with cooperative air vehicles

Filipe Oliveira e Sousa Ferreira de Lemos

Mestrado Integrado em Engenharia Electrotécnica e de Computadores

Supervisor: João Tasso de Figueiredo Borges de Sousa

July 13, 2017

Resumo

Esta tese apresenta uma arquitetura estrutural de um sistema destinado a, de forma persistente, seguir peixes, marcados com dispositivos acústicos, no oceano. O sistema de rastreamento é composto por equipas de *Unmanned Aerial Vehicles* (UAVs). Esta aplicação é sobre tubarões. Cada tubarão é marcado com um transmissor de sinal acústico, colocado na barbatana dorsal, permitindo a medição da intensidade do sinal e sucessiva localização do alvo por parte de recetores montados nos UAVs. Assim sendo, cada UAV deve ter a sua única e pré-definida área de procura de modo a que o sistema seja capaz de cobrir uma área mais extensa. Como se tratam de veículos aéreos, estes têm de pousar na água, permitindo que os seus recetores acústicos ouçam o sinal, transmitido pelo sinalizador colocado no tubarão alvo. Até encontrarem o alvo, devem pousar em diferentes sítios da sua área de procura de modo a maximizar a probabilidade de o encontrar.

Como se trata de um sistema cooperativo, se existirem mais UAVs a seguir o mesmo alvo, devem partilhar dados uns com os outros. Assim, sempre que um UAV receba o sinal, deve informar os outros veículos sobre a sua estimativa da posição do tubarão, permitindo-lhes reduzir a sua área de pesquisa e rastreá-lo com mais facilidade.

O sistema de controlo e estimação foram desenvolvidos segundo a metodologia de sistemas híbridos. Para a estimação, os UAVs obtêm uma lista de todas as posições possíveis do alvo ao medir a distância ao mesmo, através da intensidade do sinal acústico. O número de pontos possíveis pode ser reduzido através da partilha de informação com outros UAVs. Os controladores do veículo alternam entre vários estados (por exemplo, *listen*, *coop search*, *active estimation*, *flying*,...). Os eventos de transição necessários para esta comutação podem ser autoproduzidos ou fornecidos pelo controlador central. Este controlador central é usado para coordenar as missões de veículos cooperativos.

Previsão futura da posição do tubarão foi utilizada para lidar com situações especiais, como o mergulho de tubarões para zonas mais profundas e fora do alcance máximo do dispositivo acústico. Ao prever onde o tubarão estará nos minutos seguintes, é possível definir novas áreas de pesquisa em torno dessa posição, maximizando a probabilidade de encontrá-lo.

Dados reais compostos por rotas de tubarões gravadas e variação da profundidade dos mesmos serão usados para validar o projeto final. A partir das simulações, podemos concluir que é possível rastrear um tubarão no espaço tridimensional com qualquer número de UAVs disponíveis. Além disso, as estratégias de estimativa e rastreamento podem ser melhoradas ao aumentar o número de UAVs com o mesmo alvo. As equipas compostas por três UAVs foram confirmadas como ideais quanto ao consumo de bateria e precisão na estimativa do alvo.

Abstract

This thesis presents the design of a system targeted at persistently tracking fish, tagged with acoustic markers, in the ocean. The tracking system consists of a team of Unmanned Aerial Vehicles (UAV). The application is about sharks. Each shark is tagged with an acoustic signal transmitter, placed on their dorsal fin, allowing receivers mounted on the Unmanned Aerial Systems (UAS) to measure the signal intensity and pin point the location. Therefore, each UAV should have a unique and predefined search area to cover a larger surface. Since they are aerial vehicles they have to land on the water, allowing their acoustic receivers to listen for the acoustic signal, transmitted by the tag placed on the targeted shark. Until they find their target, they should land in different positions of their search area in order to maximize the probability of finding it.

Recall that it is a system of cooperative vehicles. This means that if there are more UAVs tracking down the same target, they should share information with each other. For this reason, every time a UAV receives the signal, it should inform the other vehicles about the shark's estimate location, allowing them to reduce their search area and successfully track it easier.

The control and estimation system was developed in the framework of hybrid systems. For the estimation, the UAVs compute a list of all the possible target's position by measuring the distance to their target with nothing more than the intensity of the acoustic signal. The number of possible points can be reduced by sharing data with other UAVs. The vehicle controllers switch between several states (eg., listen, coop search, active estimation, flying,...). The transition events required for this switching can be self-made or provided by the central controller - hybrid automaton. This central controller is used to coordinate the missions of cooperative vehicles.

Near future prediction of the shark's position was used to deal with special situations such as the shark diving deeper than the maximum range of the acoustic device. By predicting where the shark will be in the following minutes, it is possible to define new search areas around that position, maximizing the probability of finding it.

Real data comprised by recorded shark routes and depth variation will be used to validate the final design. From the simulation runs, we can conclude that it is possible to track a three dimensional shark with any number of available UAVs. Additionally, the estimation and tracking strategies can be improved by increasing the number of cooperative UAVs. Teams composed by three UAVs were confirmed to be ideal regarding battery consumption and accuracy on the target's estimate.

Acknowledgements

First I'd like to thank my supervisor Professor João Borges de Sousa for all his help, availability and enthusiasm. I could always count on him whenever I ran into a trouble spot or had a question about my research or writing.

A very special gratitude goes out to Dr. Nuno Queiroz for sharing real data to validate the system. I would also like to acknowledge Dr. Sujit Baliyarasimhuni for all his valuable comments on the estimation strategies.

Thanks to all my friends and colleagues that helped and encouraged me along the way. I'd also like to express my sincere gratitude to all the wonderful faculty at FEUP that have helped me learn and advance my education over the years.

Finally, I must express my very profound gratitude to my parents and my other family members for providing me with unfailing support throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you all.

Filipe Lemos

“Success is the sum of small efforts,
repeated day in and day out.”

Robert Collier

Contents

List of Figures

List of Tables

Abbreviations and Symbols

ASV	Autonomous Surface Vessel
AUV	Autonomous Underwater Vehicle
CIBIO	Centro de Investigação em Biodiversidade e Recursos Genéticos
FEUP	Faculdade Engenharia da Universidade do Porto
GPS	Global Positioning System
KG	Kalman Gain
LSTS	Laboratório de Sistemas e Tecnologia Subaquática
MATLAB	Matrix Laboratory
NTNU	Norwegian University of Science and Technology
PCA	Principal Component Analysis
PF	Particle Filter
REMUS	Remote Environmental Monitoring Unit System
RPA	Remotely Piloted Aircraft
TL	Track List
UAS	Unammed Aerial Systems
UAV	Unmanned Aerial Vehicle
USBL	Ultra Short Base Line
WHOI	Woods Hole Oceanographic Institution

Glossary

Automated vehicles Vehicles where Sense and Acting are mediated by scripted control procedures . There is no on-board deliberation.

Interoperability Ability of a system to work with or use the parts or equipment of another system

Chapter 1

Introduction

1.1 Context

Covering approximately 70 percent of Earth's surface, the ocean is a powerful force on our planet. It helps to shape the physical features of Earth, makes the Earth habitable - most of the oxygen in the atmosphere originally came from the activities of photosynthetic organisms in the ocean, is a major influence on weather and climate, and supports a great diversity of life and ecosystems.

Due to the ocean's importance to our planet, several sciences have been developed to learn more about it. Physical oceanography, geological oceanography -concerned with marine sediments, chemical oceanography - aim to understand the processes that control the concentration and distribution of elements and their compounds in the ocean, and marine biology, are some of these ocean sciences. Deciphering the mysteries of remote ocean biological systems can reveal new hot spots for medical drugs, food, energy resources, among other products. Ocean exploration incorporates rigorous observations and documentation of these aspects. Unfortunately, it is one of the wealthiest and most underutilised asset on Earth. Moreover, data from deep-ocean exploration can help predict earthquakes and tsunamis and help us understand how we are affecting and being affected by changes in Earth's climate and atmosphere.

Nowadays, with the increasingly sophisticated tools, technologies, and sensors, we learn more about our ocean every day. Ocean exploration and scientific achievements include technologies such as: (1) vessels and submersibles able to descend to depths that are not safe for human divers, which make detailed observations and collect samples of unexplored ecosystems, (2) observing systems and sensors which collect weather and ocean observations such as water temperatures and salinities, the shape of the seafloor, and the speed of currents, (3) communication technologies (that allow scientists to collaborate and transmit data more quickly), and (4) diving technologies that transport us across ocean waters and into the depths, allowing us to record scientific data of the ocean. However, we have still explored less than five percent of this vast underwater world [7].

Often, the basis for technology and engineering innovations are the challenges of exploring the deep ocean. Engineers are continuously searching new ways to study the ocean - using acoustics to reveal important information about the deep-ocean ecosystems, developing floats, robots and autonomous underwater vehicles (AUVs) that can survey the ocean and communicate their findings.

Actually, scientific researches are currently being held with the goal of learning more about the behaviour of undersea life. Shark tracking, which is a particular case of fish tracking, is one of the strategies used by these researches. Their main focus is understanding how fish behaviour is determined by environmental factors (e.g., temperature, food, ...).

Sharks are able to swim from a few meters to thousands of meters per day. Some are fast, others are slower, but the way they decide their path is probably identical. They take into account the temperature gradient, the amount of plankton and other factors to know where to go.

We still do not know much about what drives the behaviour of most fish. This is in part because we do not know how ocean features affect fish behaviour. This is why we need to track fish continuously for long periods and to measure environmental parameters that may influence fish behaviour.

During May of 2014, an experiment was made in order to retrieve data from sunfish and its habitat [8]. The goal is to understand the environment in which the fish lives and how it affects its behaviour. This experiment was conducted by *Laboratório de Sistemas e Tecnologia Subaquática* (LSTS) and *Centro de Investigação em Biodiversidade e Recursos Genéticos* (CBIO) and it consisted in tracking tagged ocean sunfish with different varieties of cooperative autonomous vehicles. These vehicles searched around the coastal area of Olhão, near Algarve, for these tagged fish, recording video footage and ping information. These data would be then transmitted to a central station allowing live streaming of the sunfish behaviour.

Within the scope of the course - Preparation of the Dissertation - inserted in the integrated master of electrical engineering of *Faculdade Engenharia da Universidade do Porto* (FEUP), Professor João Borges de Sousa presented a challenge, which consisted in the development of a cooperative system of unmanned air vehicles capable of persistently following sharks in the ocean. Tags that emit signals will be placed on the targeted sharks, allowing the vehicles to follow it. Since sharks swim at an extremely high speed for most vehicles to keep up with them, it is necessary that a set of these vehicles exist in such a way that, in case the tracking signal is lost, it will be more efficient to find it back. This is possible since they can contemplate a larger area of demand in less time.

This topic is in the area of control and optimisation, being also related to animal biology, in particular with the subject previously discussed.

1.2 Motivation

By now you must be wondering why tracking sharks is so important. In fact, several plants and animals around the world are currently experiencing rapid declines. Unfortunately, sharks are not an exception and, for that reason, scientists are worried about their future. Tracking these endangered species makes it possible to understand the reasons behind the decrease in their numbers and, with that, elaborate prevention mechanisms. Actually, in [9], a team of researchers, while tracking satellite tagged sharks, found out that there is overlap of long-line-vessel fishing fleets with shark habitat hot spots, across the North Atlantic. They have also confirmed that this association is spatially and temporally persistent between years, which suggests that their hot spots are at risk. This information could be used to argue for the introduction of international catch limits, preventing these sharks extinction.

Additionally, biologists would like to extend their knowledge regarding migration paths and shoals size of some sharks; the maximum distance, from their home habitat, achieved when hunting; their preferred habitats or even how is shark behaviour determined by environmental factors. Luckily, every topic just mentioned can be explored by persistently tracking this species.

Also note that, knowing sharks current location or where they are heading may prevent shark attacks close to the beach, saving human lives.

Clearly, it is important to develop a system capable of tracking fish. Since their speed varies from species to species, this system should be designed for fast fish because then it will work for all types of fish. Unfortunately tracking fast fish for extended periods of time is not an easy task. For example, some sharks move too fast, reaching approximately 8 m/s on vertical transitions, and dive too deep to be tracked by Autonomous Underwater Vehicles (AUVs). Estimation and prediction mechanisms combined with knowledge of the fish behaviour must be used to overcome these difficulties.

At the moment, the most common approaches for tracking fish are (1) satellite tracking and (2) acoustic tracking. Some commonly used methods, regarding the second approach, are moving boats, fixed acoustic receivers or single/cooperative AUV system. Undoubtedly, every system has their own flaws and merits. Satellite tracking tags, used in [9], require the fish to be very close to the surface in order to transmit information. For that reason, tag transmissions are not regular which makes it very difficult to evaluate shark behaviour between two consecutive transmissions.

Moving boats, which were used in [10], are normally too expensive for long missions since they have to be constantly moving to keep up with the targeted fish.

When using stationary acoustic receivers, as in [11], it is only possible to observe shark's behaviour when they are within their range. Therefore, if the surveillance area is not big enough to contemplate different types of habitats, not much information can be derived from the behaviour of a given shark. As it was shown in that work, some tagged sharks might not even appear within the covered area.

Other methods like [12] and [5], use autonomous underwater vehicles for tracking fish. However they are not fast enough for tracking some species of sharks when sharks go on burst swims or dive too deep. Nevertheless, these vehicles, like every autonomous vehicle, have battery constraints which means that their missions are normally cancelled when they run out of battery.

1.3 Objectives

The main goal of this dissertation is to implement a control system for a team of unmanned vehicles in such a way that it is able to follow any type of fish in the ocean. Initially, we considered that this animal would be the sun-fish, however, the ultimate goal will be to design the system so that it is possible to follow sharks.

This goal can be achieved with a system of cooperative UAV's. These aerial vehicles have to land on the water, allowing their signal receivers to listen for pings, transmitted by the tag placed on the targeted shark. Additionally, they can travel faster than the common AUV, which is required since sharks usually go for burst swims, attaining velocities impossible for an AUV to keep up with.

Basically, each UAV should have a unique and predefined search area to cover larger surface. Until they find their target, they should land in different positions of their search area in order to maximise the probability of finding it.

Recall that it is a system of cooperative vehicles. This means that if there are more UAV's tracking down the same target, they should share information with each other. For this reason, every time a UAV receives the signal, it should inform the other vehicles about the shark's estimate location, allowing them to reduce their search area and successfully track it easier.

After finding the shark, regular future position estimation should be used for two reasons: (1) the signal has dead zones so even if the vehicle is on range, there will be no signal to listen to and, (2) shark can dive deeper than the maximum range of the receiver, losing the signal for good. Therefore, in order to deal with these type of signal loss situations, the system should be able to predict where the shark would be going in the following minutes, making it possible to define new search areas for the UAV's. This would maximise the probability of finding the target.

Moreover, this thesis should address the following investigation questions:

- Can we estimate the position of the fish based on real observations?
- Can we improve the quality of this estimation with multiple vehicles?
- Can we improve the tracking strategy by using multiple UAVs?

1.4 Thesis overview

In the following chapter, background regarding some topics such as unmanned aircraft systems, fish tracking methods, estimation procedures and others will be given. Chapter 3 briefly describes the problem, how it will be tackled and the functional architecture.

In Chapter 4, current fish tracking methodologies will be analysed to the very detail, exploring how they estimate fish position and how they use the fish behaviour to achieve better results. The distributed control architecture of the UAVs system will also be presented.

Chapter 5 starts by explaining the overall distributed architecture of the UAVs team system. Then, it will be possible to find out how and when do vehicles communicate. Throughout this chapter, the selected estimation techniques, the models used to simulate the fish behaviour and the design of the controllers will be given.

Finally, Chapter 6 will explore the algorithm and simulation environment used to validate the final system.

1.5 Contributions

The contributions of this work include:

- A control architecture for cooperative tracking.
- Controllers modelled in the framework of hybrid automata.
- Estimation and tracking strategies for tracking a tagged shark at any depth with nothing more than the intensity of the acoustic signal.
- A system that can track fast fish.
- Analysis of fish tracks.

Chapter 2

Background

In the last chapter, I mentioned how LSTS and CBIO have been tackling the main thematic of this thesis – fish tracking. Also, explained why it is so important to achieve a system capable of continuously chasing fish in the open ocean.

I will begin by examining the definition of unmanned aircraft systems for marine operations, which will be used on this work. Then we will explore how are fish being tracked nowadays. This will be followed by knowledge regarding hybrid control architectures and estimation methodologies. In this final topic, Kalman and Particle Filters will be briefly explained.

2.1 Unmanned aerial system

An UAS is an aircraft that does not require a qualified human pilot on board. It can perform manoeuvres unrealistic for human pilots and does not require expensive life-support systems. Their flight can be conduct either under the remote control of human operator or by on-board computers. For these reasons, they are preferred over manned aircraft for dangerous or dull missions for the humans, such as military operations.

Nowadays, higher level functions such as path planning supervision or target tracking, are still performed by remote operators due to the low level of automation of the majority of these UASs. Normally the operation of an individual unmanned aircraft requires no less than one ground station, a few operators, and a data connection to the vehicle. In the most straightforward remotely piloted aircraft (RPA), the ground station uses the data link to interact with the aircraft.

Note that, the interoperability goal for Unmanned Systems is not just to provide data, material, and services to others systems but also to accept the same from them, using the exchanged data to improve the cooperative efficiency of the global system.

Moreover, although existing operational cost comparisons between manned and unmanned aircraft have not been necessarily favourable to UAS, the value of human life has been a major consideration for the use of military UAS in missions. [13]

Figure 2.1: Mariner UAV landing on the water

2.2 Fish trackers

Typical methods for tracking fish include sensing GPS tags, active manual tracking, static receivers and robotic autonomous systems. This methods have the particularity of not disturbing the animal natural behaviour while fulfilling their mission.

All these methods have their flaws and merits. Even though GPS tags (Figure 2.2) provide accurate positional info, the tags only transmit when the fish is at the surface. This leads to data gaps when the animal is not close to the surface. One possible solution for this, was the introduction of acoustic sensing technologies. Acoustic signals can travel through water but, since they have a maximum range, receivers must always be within their range in order to receive new information. This methods involves tagging a fish with an acoustic device, which is capable of transmitting information almost continuously, depending on the tag frequency.

Figure 2.2: GPS satellite tracking

One common tracking method, involves mounting a hydrophone acoustic receiver on a moving boat for following tagged fishes. However, since they have to be constantly moving to keep up with the target, this missions tend to be very expensive and, therefore, not practical. It also requires human to operate the vehicle throughout the entire mission.

Stationary acoustic sensors can track the movement of tagged animals when they are within their range. Since they are stationary, when the target leaves their localised area it is not possible to record data anymore. Figure 2.3 shows how this method works.

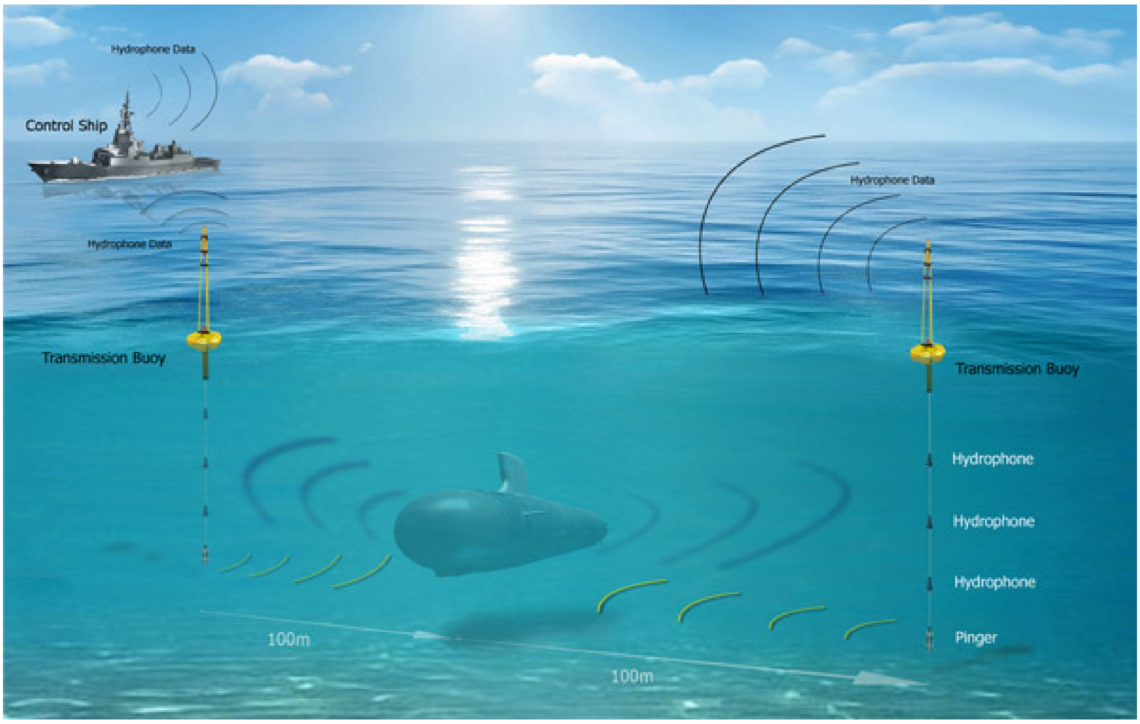


Figure 2.3: Stationary acoustic fish tracking

Nevertheless, both these methods are affected by environmental factors and require human operation. Unlike them, robotic autonomous systems do not have these disadvantages. Consequently, autonomous underwater vehicles have been used to follow tagged animals, Figure 2.4. Basically, they also have acoustic receivers ($\approx$ 4.5 kg) mounted on them to track down acoustic tags. Moreover, they can achieve considerable depths without damaging the vehicle, giving them a greater chance to keep up with their target when it dives. However, some fishes swim too fast or achieve depths inconceivable for this vehicles to successfully follow. More recent solutions, use a system of cooperative vehicles to provide a bigger surveillance area but, for long term missions, it is still not enough since they can't tackle the problems described above.

(a) View from the AUV

(b) Shark acoustic tag

Figure 2.4: AUV shark tracking

Additionally, it is important to know that the tags are placed adjacent to the dorsal fin via an intramuscular dart, Figure 2.4b. Most of them, have also a remotely activate pop-up mechanism, allowing the retrieval of the tag without disturbing the animal.

Model 795 & 800 Acoustic Tags

0.5 gm, 15-20 d life
0.67 gm, 20-35 d life
1.4 gm, 40-50 d life
3.1 gm, 45-65 d life
13 gm, > 6 mo life
23 gm, > 3 yr life

Tags operate at 307 kHz and are fully programmable
for pulse width, period, and activation.

T. Steig AFS 2012 Presentation
August 22, 2012

Figure 2.5: Size, weight and life period of acoustic tags

2.3 Hybrid System Models

Controlling a multi-vehicle system is not a trivial task. Therefore, the authors in [14] had to introduce a formal specification to operate such a system. They have modelled their controllers in the framework of a hybrid transition system, described by Definition 1.

Definition 1. A transition system T is a tuple

$$T = (Q, \rightarrow, I, O, \mathrm{I}nit, \mathrm{F}inal),$$

where

- Q is the set of states
- I and O is the set of inputs and outputs, respectively
- $\rightarrow \subset Q \times I \times Q \times O$ is the transition relation
- $\mathrm{I}nit \in Q$ is the initial state

- F*inal* $\in Q$ is the final state

The interpretation is that an input $i \in I$ cause the system to move from one state $q \in Q$ to another state $q' \in Q$ producing the output $o \in O$. It is convenient to write $q \overset{i/o}{\rightarrow} q'$ instead of $(q,i,q',o) \in \rightarrow$. The graphical representation of T is a directed graph with vertices representing Q and arcs representing $\overset{i/o}{\rightarrow}$, an arc with empty origin representing I*nit* and a vertex with an extra circle representing F*inal*.

Lets explore the following example of an hybrid automaton in order to better understand this thematic. Basically, it describes the control mechanism of a thermostat in a room.

Denote Q = {on, off} as the set of all discrete states that describe the thermostat modes. Additionally, x is a continuous state-space variable that stands for the mean temperature of the room.

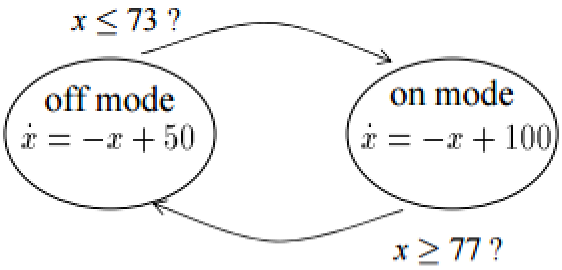


Figure 2.6: Hybrid Automaton, retrieved from [1]

$$o = \begin{cases} -x+50, & \text{q=off} \\ -x+100, & \text{q=on} \end{cases} \tag{2.1}$$

$$q' = \begin{cases} on, & \text{q=off}\ ,\ x \leq 73 \\ off, & \text{q=off}\ ,\ x > 73 \\ off, & \text{q=on}\ ,\ x \geq 77 \\ on, & \text{q=on}\ ,\ x < 77 \end{cases} \tag{2.2}$$

In this system, the discrete transition in Equation 2.2, derives the new state q' by taking both the continuous and the discrete variables, q and x, as the input. For example, if the current state q of the thermostat is off and the mean temperature x drops below 73, the heating should be turned on, leading to the new state q'.

Moreover, note that while switching between the discrete states, it produces the output o, which is the rate of change of x. Therefore, the state space variable x will be updated every time a transition occurs, by following Equation 2.1.

2.4 Fish Models

A system capable of continuously tracking fish, must be able to predict its location when the system deals with sparse data. Thus, having knowledge regarding the behaviour and swimming pattern of their target, is required to successfully achieve this goal.

Like every irrational animal, fishes are ruled by their instinct to survive and reproduce. This does not only influence their behaviour, but also their swimming nature. Considering our system's target – the shark – it exhibits two special styles of motion: (1) the "V" and (2) the "U" shaped swimming patterns [2]. The first one, represented by Figure 2.7, describes the exploratory behaviour of a shark. It consists in fast and continuous depth transitions which reminds of a sequence of the letter "V", when analysing the depth variation of the shark in respect to time. This constant depth switching behaviour, may provide the shark with a greater perception and knowledge on how to approach their prey.

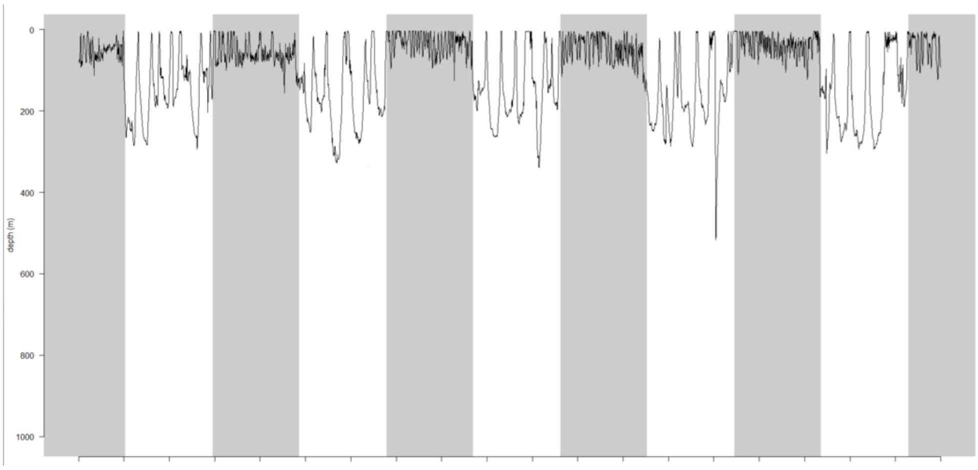


Figure 2.7: V shaped swimming pattern, retrieved from [2]

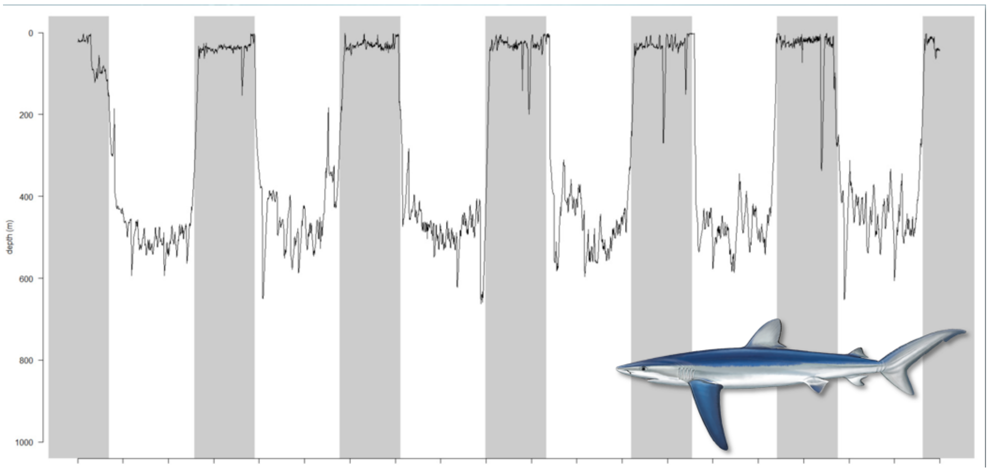


Figure 2.8: U shaped swimming pattern, retrieved from [2]

On the other hand, the "U" shaped swimming pattern, represented by Figure 2.8, is normally associated with the foraging behaviour of this predator, when faced with large groups of prey in discrete horizontal depth layers. Foraging is the animal's ability to search for food resources. Because of its prey spatial distribution, it doesn't have to keep changing depth, staying for longer periods at each layer, leading to the so called "U" shaped pattern.

Fortunately, scientists have developed mathematical and probabilistic models to associate these swimming patterns with the fish behaviour. These models are represented in Figure 2.9. The Brownian motion model, illustrates the behaviour of the shark, when encountered with an abundant habitat regarding the number of preys. In this situation, the animal tends to change the direction of its course quite often, granting a more global picture of the spatial distribution of its prey. Mixing this with fast depth transitions, allows this predator to achieve vantage points, used to approach and capture its prey through their blind spots. As you can see, this model takes into account the "V" shaped swimming pattern of a shark.

The Levy motion model, highlights the behaviour of the animal, when facing a habitat with sparser prey. Here, the shark tends to move in a straight line, for long periods of time. Changes in the direction of its course is very rare, specially when compared to the Brownian model, probably because the shark has a hard time locating preys to chase. It is also normally associated with the "U" shaped swimming pattern.

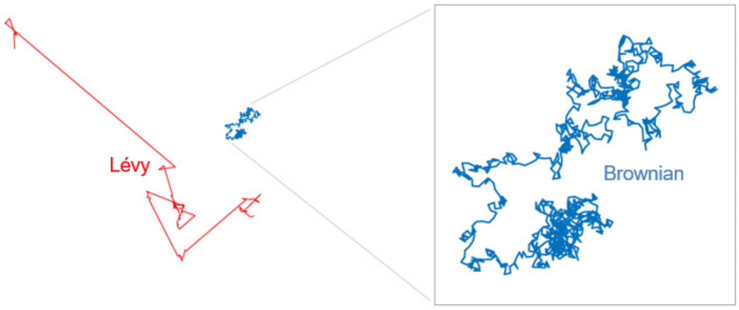


Figure 2.9: Levy and Brownian motion models, retrieved from [2]

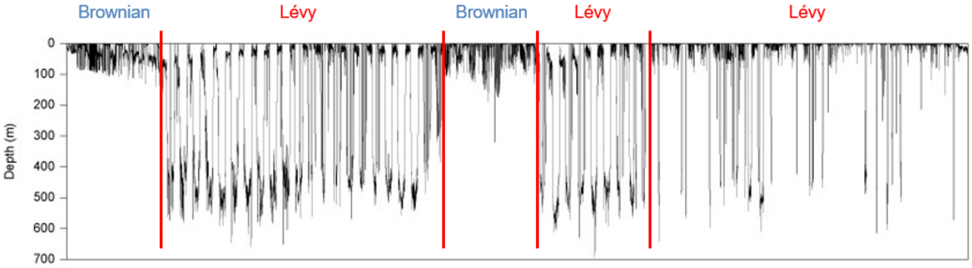


Figure 2.10: Levy and Brownian depth variation, retrieved from [2]

However, why are these models important when tracking fish? Certainly, while we are tracking fish, we are collecting position and depth data. If after a few observations we could identify the motion model, it would be very useful for future prediction. For example, if the observations made until time t could represent a straight movement, we could assume that the shark was using a Levy motion model and, therefore, it would be plausible to assume that the fish should maintain its current route. Also note that, this motion model is associated with high depth transitions. Figure 2.10 represents this association. After identifying the model, if the system is dealing with a Levy motion model, the vehicles should predict the target future location and wait there. This would maximise the probability of finding it in case of sudden signal lost.

On the other hand, if the data points were too scattered, then we would be dealing with the Brownian motion model. Here the tracking should be done with less movement since the shark stays in the same area for quite a while.

2.5 Estimation

Estimation is a process that, given multiple observations of a certain object (eg., position of fish), produces good approximations of the real value, even when dealing with incomplete data or high measurement/process errors. Even though, there are several techniques, we will only introduce two of them : the Kalman and the Particle filter.

2.5.1 Kalman Filter

The Kalman filter is an iterative linear mathematical technique used to quickly estimate the true value of the target being measured, when the measuring device is in the presence of disturbance or unpredicted random error [15]. It requires two methods for estimating the real value: (1) a state estimator using the kinematics equations of motion and (2) a measuring device. For the first one, an initial position, velocity and acceleration must be defined. Additionally, the errors regarding each method must be well known and they should be Gaussian distributed.

Sometimes, it is crucial to, quickly, obtain accurate approximations regarding the object we are measuring. Thus, if we used just one of the methods it could be hard to achieve this goal. For example, the target could slow down too quickly, for the state estimator to be able to precisely follow it or, due to interference from other objects, the measuring device could measure something else, providing unrealistic results. For this reason, an optimal solution is achieved by doing a weighted average of the results derived by both estimators. This is where the Kalman filter enters. Note that, both estimators describe their estimations using a probability density function, defined by its mean and variance. It is this function that allows the filter to decide which estimator we should trust more.

How does it work? Basically, this algorithm has three major steps. The first step is determining the Kalman Gain. This gain is a numeric value, between 0 and 1, that combines the error from in the state space estimator with the error in the measurement. It is given by Equation 2.3, where E_{EST} is the error in the state estimator and E_{MEA} is the error in the measurement.

$$KG = \frac{E_{EST}}{E_{EST} + E_{MEA}} \tag{2.3}$$

It is possible to observe that, the smaller the Kalman Gain, the less accurate the measurements are and the more stable the state estimator is. On the other hand, for bigger gains, the state estimator tends to be less stable and the measurements more precise.

After this step, it is time to calculate the current estimate which is given by Equation 2.4. This denotes the optimal solution at each iteration with respect to the initial or previous results. As you can see, the Kalman Gain is the weight that will decide in which method should we place more faith, every time a new measurement is obtained. For example, if a new measurement is completely odd (high error), the gain will be very small, making the current estimate very similar to the previous one, ignoring the new measurement. This shows how smoothly this filter reacts to the presence of noise. In Equation 2.4, EST_{t-1} is the old estimate.

$$EST_t = EST_{t-1} + KG(MEA - EST_{t-1}) \tag{2.4}$$

Finally, the error in the estimate must be updated. This is done by following Equation 2.5. It is interesting to note that, the more we use the Kalman filter, the more the error in the estimate will decrease. Therefore, this method can quickly zoom in the true value, with few iterations.

$$E_{EST_t} = (1 - KG)E_{EST_{t-1}} \tag{2.5}$$

Certainly, more advanced equations must be taken into account to fully compute the Kalman filter but, this already provides the reader with a global idea of how it works and why it is so useful.

2.5.2 Particle Filter

Particle Filters are techniques used in several works to estimate the state of the targets [3], [5], [16], [17]. They use a collection of particles to represent a probabilistic distribution of the target's potential states. Each potential state is represented by one particle. These particles can contemplate many characteristics such as position, orientation, velocity, among others. Initially, these characteristics are randomly selected from a uniform random distribution. Note that, the particles should be spatially placed around the measuring device. Moreover, each particle is associated with a weight. This weight is a numeric value that defines the likelihood of a particle. In other words, it illustrates how well each particle fits with the measured value. At the beginning, every particle has the same weight.

Since this is a continuous process, every time a new observation is made, it is used to update the particles. By comparing each particle with the new measurement, it is possible to update the weight of the particles. The more likely a particle is found the bigger the weight should be. Based on the same reasoning, if a particle doesn't explain the measurements that well, it should be associated with a smaller weight.

After doing this for every particle, it is time for the re-sampling phase. Basically, the particles are re-generated, normalising their weights, but instead of being placed around the entire region, they must be placed based on the weight of the previous particles. Thus, regions which had more likely particles, will now be placed with a lot of new particles. On the other hand, those which had several unlikely particles will disappear or have fewer particles.

Besides this, the particles must also be spatially updated. For this several motion models can be selected. By applying this models, the particles are propagated in time. This is the final step of the iteration. Therefore, the algorithm should now re-start by updating the weights with the new measurement and so on.

Eventually, the spatial distribution of the particles will get smaller and smaller, providing a good estimation for the position of the target. This method is very efficient and commonly used for non-linear problems.

Chapter 3

Fish Tracking Problem

In this chapter, I will define the fish tracking problem. Firstly, some assumptions are made in order to be possible to tackle it. Then, the use cases and the functional requirements of the system will be explored. Finally, the sub-problems such as motion, estimation, prediction, derived from the main problem, will be presented, proving a more precise view of the fish tracking problem complexity.

This work addresses the following problem: Given a set of UAVs, equipped with acoustic receivers, capable of landing and taking off from water, while measuring distance with respect to tagged fish, under some assumptions, design vehicle and global controllers and an estimator to enable the UAVs, operating either in isolation or as a team, to persistently find and track tagged fish.

3.1 Assumptions

Formerly, this thesis goal was to design and implement a system able to follow sunfish in the ocean. Theoretically, this system would eventually be adapted to chase faster and more challenging fishes. However, after a meeting with Dr. Nuno Queiroz, Biologist specialised in the study of large fishes and researcher at CIBIO [18], the type of fish was changed to a much faster fish - the shark.

Since the fish tracking problem is an extensive and complex problem, a list of all the assumptions that define this problem had to be made. This list can be seen below:

Assumption 1 (Tags)**.** The tags placed on the sharks generate acoustic signals and have a maximum range of 500 meters. Moreover, they stick to the fish until the end of the mission.

Assumption 2 (Environmental Conditions)**.** The environmental conditions are ideal throughout the entire mission.

Assumption 3 (UAV)**.** The UAVs may land anywhere on the sea and are aware of their battery level. They can charge their batteries, with solar panels placed on them, when they are not moving.

Assumption 4 (Communications)**.** Communications between UAVs have distance boundaries. Within these boundaries, the transmitted messages have no delays and always arrive at their destination. Additionally, the UAVs can only transmit data when they are not moving.

3.2 Definitions

We need some definitions. Let f, α, x and y denote, respectively, the frequency of the tag, the intensity of the acoustic signal and the coordinates of the UAV.

Definition 2 (Tag distance)**.** Every tag has their own frequency $f \in \mathbb{R}^{+}$, which is how we can distinguish the different sharks. The set of all these frequencies is denoted by F.

The UAVs have devices that, given the intensity of the acoustic signal – α, can measure the distance d to the tag f.

Definition 3 (Observation)**.** Observations are made by UAVs when they listen to the acoustic pings generated by the tags. Each one is characterised by its unique frequency f, the time t when the observation was made, the UAV *(x,y)* location and the measured distance d to the tag. Therefore, it can be represented as follows:

$$observation = (f, t, x, y, d) \tag{3.1}$$

3.3 Use cases

A use case diagram is a modelling tool that is used to describe the different actors interaction with the system (Figure 3.1). The system is composed by the UAVs, the central control and the sharks. However, only the first two are controllable. The functions can be executed by stand-alone UAVs or teams. In this thesis, a team is a group of vehicles that help each other to achieve a common goal: follow or search for the same target.

The system should be able to run with just one vehicle. The UAV remains in an idle state until the central control assigns a target to it. Afterwards, since the vehicle is pursuing its mission alone, it does not require to communicate with other system components. In this case, the vehicle should be able to distinguish a fish from a group of targeted fishes (if the number of targets is bigger than one) and follow or search for it. Note that, with only one vehicle we can only pursue one target at a time.

On the other hand, we have a system composed by a group of vehicles. These vehicles can have the same or different target but they must be able to communicate with any vehicle they encounter. The ability to share information with the other system components is one of the roles that distinguish this system from a system composed by a single vehicle. Additionally, since the number of vehicles is bigger than one, it is possible to chase multiple targets at the same time.

In this cooperative mode, there can be teams - several vehicles with the same target - and vehicles tracking alone, running at the same time. The main goal of this type of system is to

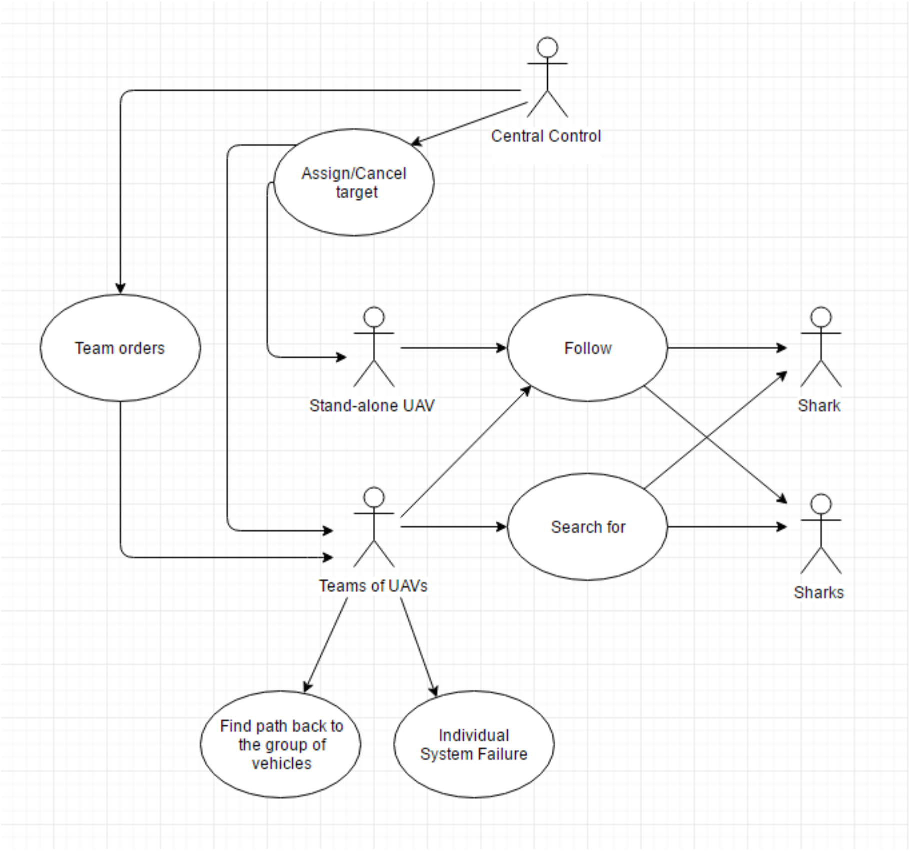


Figure 3.1: Use case diagram

increase redundancy and surveillance area. For this reason, when we have a team of vehicles, we maximise the probability of finding a missing target. Also, since the vehicles communicate, by sharing their observations they can estimate the position of their target, faster than a single vehicle. Note that, it is the central control that coordinates teams. Therefore, the UAVs have to wait and follow its orders, throughout their entire mission.

Moreover, the vehicles must be able to find the path back to their team, whether they went out of the communication range or they have just stopped for charging their batteries. It would also be interesting to make sure the system can deal with vehicle failure, situation when the vehicle can't proceed with its current mission. However, this situation can only occur on cooperative mode since the system must always remain with at least one functional vehicle.

3.4 Functional Requirements

Requirements are a set of characteristics that must be satisfied by a system in order to be useful and bring value to the customer. There are several types of requirements but we are only interested in the functional ones. These define a function of a system or its component.

Table 3.1: Functional Requirements

ID	Component	Requirement
1	Vehicle	Must be able to actively estimate and track the tagged shark
2	Vehicle	Must have a search strategy for locating a given tagged shark
3	Vehicle	Must record observations regarding any received acoustic ping
4	Vehicle	Must be able to disseminate data with other vehicles with the same target
5	Vehicle	Must be able to share collected observations with any vehicle it encounters
6	Vehicle	Must be able to share the target's collected observations with the central control
7	Vehicle	Must be able to take off when requested
8	Central Control	The team control must grant way points that respect the communication constraints
9	Central Control	Must be able to estimate the position and direction of the target with only two vehicles
10	Central Control	Must be able to estimate the position and direction of the target with at least three vehicles
11	Central Control	Must have a cooperative search mechanism for locating a missing target
12	Central Control	Must be able to address targets to vehicles
13	Central Control	Must be able to abort vehicle missions

3.5 Problem statement

The fish tracking problem can be decomposed into the following sub-problems: (1) motion, (2) estimation, (3) prediction and (4) team control. The first one, explains how the vehicle should move, what are the known and control variables, the constraints to the its movement, and so on. The second problem, describes the a general approach for estimating the shark position from the intensity of the acoustic signal. Then, the prediction problem will explain how to act, in theory, when the shark signal is lost. Finally, the last one illustrates how the controllers for the team coordination should be designed, while respecting the vehicles specifications.

3.5.1 Motion

Each vehicle is characterised by its X_v and Y_v coordinates and orientation θ_v. These variables define the vehicle's position, which is given by Equation 3.2. In a similar way, the shark's position is given by Equation 3.3. Note that, instead of the orientation, we only need the rate of change of the shark's position. The linear and angular velocity of the vehicle are the control variables, described by Equation 3.4.

$$S_{vehicle} = [X_v, Y_v, \theta_v] \tag{3.2}$$

$$S_{shark} = [X_s, Y_s, \dot{X}_s, \dot{Y}_s] \tag{3.3}$$

$$U = [V_v, W_v] \tag{3.4}$$

When we are talking about the vehicle's motion, the goal would be to minimise the vehicle's battery consumption without losing track of the target. Since the battery consumption of an UAV is proportional to their instantaneous velocity, this goal can be achieved by minimising the vehicle's velocity. Therefore, the cost function of this problem can be defined by Equation 3.5.

$$min\ consumo \propto V_v \tag{3.5}$$

Additionally, some constraints have to be taken into account. In fact, the control variables, linear and angular velocity, must have a maximum value. This is expressed by Equations 3.6 and 3.7. Besides that, the vehicles must follow the kinematic equations of motion which are represented by Equations 3.8, 3.9, 3.10.

$$V_v \leq V_{max} \tag{3.6}$$

$$W_v \leq W_{max} \tag{3.7}$$

$$X_t = X_{t-1} + \Delta t V_v cos(\theta_{t-1}) \tag{3.8}$$

$$Y_t = Y_{t-1} + \Delta t V_v sin(\theta_{t-1}) \tag{3.9}$$

$$\theta_t = \theta_{t-1} + \Delta t W_v \tag{3.10}$$

Finally, the vehicles should not interfere with the shark's natural behaviour. For that reason, they must always stay γ meters away from the shark. However, they must make sure they are always inside the signal range. This constraints are defined by Equations 3.11 and 3.12, respectively.

$$\sqrt{(X_s - X_v)^2 + (Y_s - Y_v)^2} \geq \gamma \tag{3.11}$$

$$\sqrt{(X_s - X_v)^2 + (Y_s - Y_v)^2} < \kappa \tag{3.12}$$

In the control hierarchy of the UAV, there are low level and high level functions. Low level functions deal with continuous or sampled signals while high level functions treat events. The motion of the UAV is an example of a low level function. In this thesis, we want to find a way to control the motion of the vehicle without being concerned with UAV's dynamics. The solution should be compatible with the high level team controllers, designed as a result of the fourth problem.

3.5.2 Estimation

In this work, I am just going to use the intensity of the acoustic signals to estimate the shark position. From the moment the vehicle gets its first measurement, it must keep collecting new measurements. With this, it is possible to continuously update the estimation, reducing its error.

Lets present the following variables:

$X_{s,t}$ → The estimated position of the shark at time t
$X_{s,t-1}$ → The estimated position of the shark at time t-1
β → The weight given to the previous estimated values.
I → The intensity of the acoustic signal

Certainly, for this type of problem, the goal is to minimise the error ε of the estimator – Equation 3.13.

$$min\ \varepsilon \tag{3.13}$$

Additionally, the estimation variable should be updated through Equation 3.14. It is possible to observe that, by decreasing the weight β, we will be giving more relevance to the new measurement than to the past estimated position. The *F(I)*, in this equation, stands for the new position measurement of the shark computed by using the intensity of the acoustic signal as input.

$$X_{s,t} = \beta X_{s,t-1} + (\beta - 1)F(I) \tag{3.14}$$

After computing the new estimated (x,y) position – $X_{s,t}$, a new observation is made by the UAV through Equation 3.1.

3.5.3 Prediction

Unlike the previous problem, the prediction of the shark location is not a continuous process. Actually, this prediction is just preformed once before calculating the desired way points. It contemplates the following variables:

α_{ij} → Possible position *i* of fish *j*
t_j → Track list of fish *j*
δ_{ij} → Binary decision variable. It takes the value 1 if the shark *j* should be at position *i*. Otherwise it has the value 0.

The predicted position is given by Equation 3.15. Basically, for every possible fish position *i*, the vehicles must compute their probability regarding the fish track list and the fish natural behaviour. The solution is the position with the maximum probability for the set of possible positions. However, only one position can be selected, which is why the constraint, given by Equation 3.16, must be presented.

$$argmax \quad P(\alpha_{ij}|t_j)\delta_{ij} \tag{3.15}$$

$$\sum_{i=1}^{N} \delta_{ij} = 1 \tag{3.16}$$

3.5.4 Team Control

The controllers must be designed in such a way that they respect the vehicle's specification. In other words, they should respect several modes such as random search along a predefined area until they find the fish and, while listening to the signal, the vehicles should try to move in the direction of it by measuring the signal intensity. Additionally, communication protocols should be implemented in each vehicle to allow cooperative search: informing other vehicles about the fish estimated location and good spatial distribution of the vehicles in order to cover larger surface – critical when finding the tagged fish.

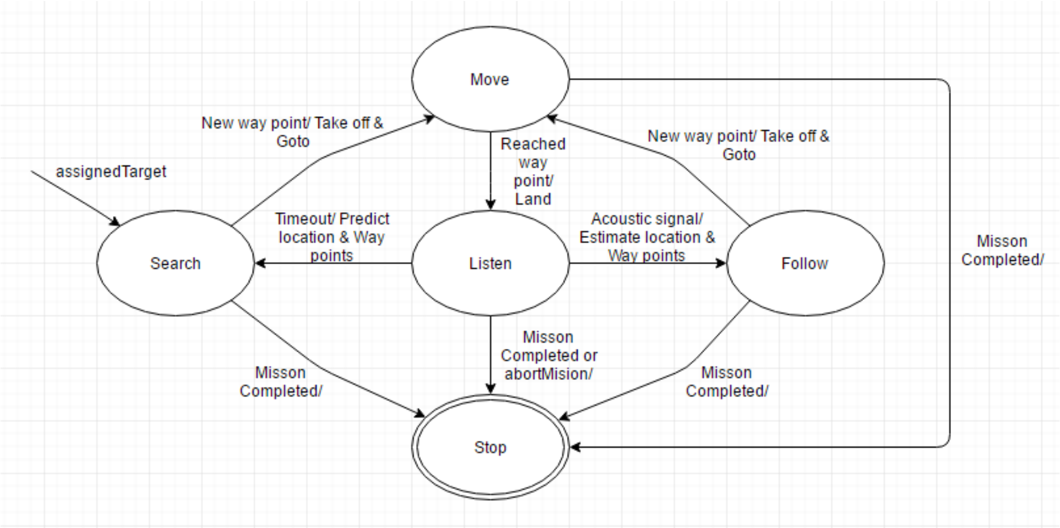


Figure 3.2: Vehicle's Specification

The system starts in the Search state. From this state, the transition to the Stop state takes place when the mission completed condition is true. Otherwise, the vehicles must follow their generated way points by heading to Move state.

At the Move state, vehicles have to reach their way points. Once they do that, they will be able to land and check for acoustic signals, which is the goal of the Listen state. If the vehicles are unable to receive any signal before the pre-defined time runs out, they must use the prediction algorithm to generate new way points and go back to the Search state. If else, the vehicles should start the estimation algorithm and define way points to follow the shark – Follow state.

When the missions are completed, the state automatically jumps to the Stop state, no matter what state the vehicles are currently at.

Additionally, the controller of the central station should also be designed in order to respect its specification. Assigning targets to vehicles, defining way points and aborting missions are some of the functions of this station.

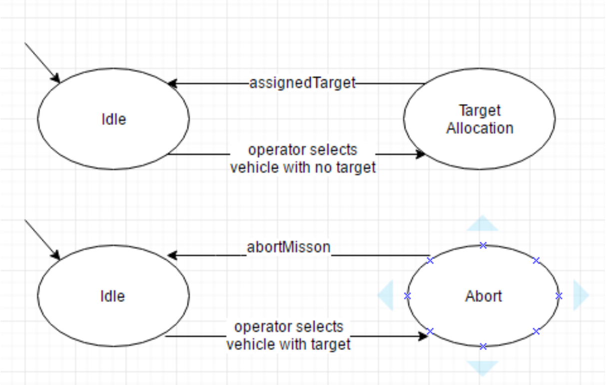


Figure 3.3: Control's Specification

Moreover, due to Assumption 4, we can have a synchronous composition model of the team controllers. Then, we will have to prove that this model satisfies the vehicles specification. This can be done by proving that the Equation 3.17 is satisfied.

$$A_1 \sim A_2 \tag{3.17}$$

In Equation 3.17, A_1 is the specification of the individual vehicles and A_2 is the composition of the team controllers.

Chapter 4

State of the art

The goal of this thesis is to design a system of cooperative vehicles capable of chasing fish in the ocean. This system should be capable of estimating the position of fish tagged with acoustic markers, of tracking acoustic signals, and of searching for fish in the absence of acoustic signals.

Techniques and methods from [14] will be used in the synthesis and verification of properties, as well as to derive the formal models. In what follows I will discuss a few selected papers on fish tracking with unmanned vehicles.

4.1 Estimation of fish position

Prior to [3] tracking and following large marine animals was very inefficient since it was only based on acoustic signals from moving boats. The introduction of underwater vehicles like the *Oceanserver IVER2* allowed a greater insight of this animal behaviour and ecology. These researchers designed a system capable of following acoustic tagged targets, one at a time. Additionally, while doing this, it can also record data from other acoustic tags that work on the same frequency. They used a Lotek MAP600RT stereo-hydrophone receiver system, a GPS receiver and a digital compass to be able to estimate the position, orientation and velocity of a tagged shark. All these components are mounted on the AUVs.

The Oceanserver IVER2 AUV is equipped with a GPS to estimate its current position. However, since the GPS doesn't work under water, the AUV has to come to the surface to use it. Additionally, the Oceanserver IVER2 has two processors. The first one was used to generate way points and communicate with the sensors and actuators. The other one was responsible for the state estimation, acoustic receiver software and controller loop. Note that, these researchers weren't interested on the depth of the shark. Regarding the position, the AUV only had to estimate the two dimensional location of its target (x,y).

How does the system work? Due to the spatial separation of the two stereo-hydrophones placed on the AUV, they are able to measure the bearing to the tag α and the intensity of the signal ρ. This is possible because the signal does not reach both hydrophones at the exact same time. However, this system cannot determine weather the measurement is related to the angle α or $-\alpha$.

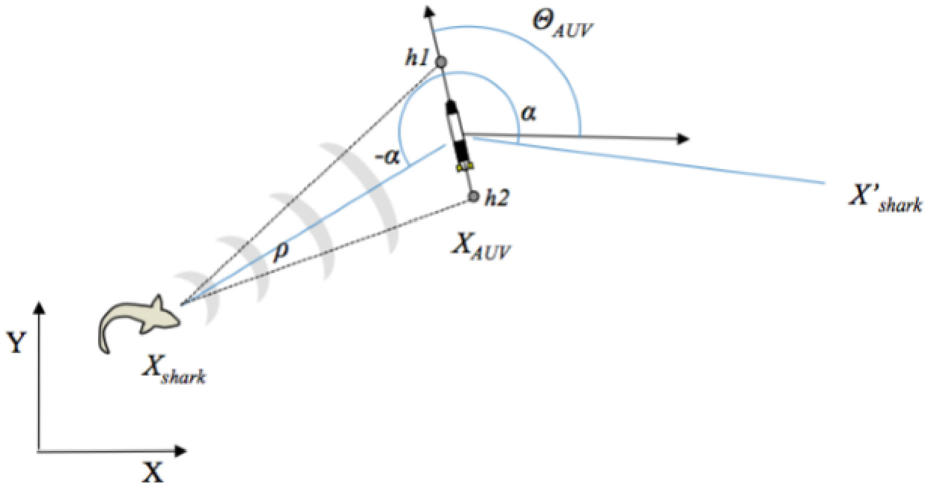


Figure 4.1: Top Down View of Sample Measurement, taken from [3]

This sign ambiguity is shown in Figure 4.1. In order to be able to pinpoint the true location of the estimate, the vehicle has to perform zig-zag movement.

After receiving this measurements, it is time for the state estimation. In this work, they used a particle filter, shown in Figure 4.2, to estimate the shark state.

Finally, the controller takes the computed state of the shark and elaborates the course plan. This plan is nothing more than a sequence of way points that define the best approach for the vehicle.

Later the same year, other researchers published an article describing a similar work. In [4], they used a *REMUS-100* AUV to track a tagged white shark. This shark had a transponder, equipped with an omni directional USBL acoustic receiver, which could respond to messages from the vehicle and provide its current depth. The transponder had also a release mechanism to allow its retrieval without disturbing the shark natural behaviour.

Unlike the AUV from [3], this one could not only follow the tag but perform other manoeuvres such as fly bys or fly overs. Considering that the vehicle had four cameras, these manoeuvres were particularly interesting since they could provide footage regarding the shark swimming behaviour, from different points of view. Also, remote operators could control the vehicle by adjusting its speed or depth. This would allow the AUV to achieve different vantage points to achieve its goal.

When the vehicle is within range of the transponder, it uses the USBL acoustic navigation system to, continuously, interrogate the transponder. Using the communication fly-time it is able to measure the range and bearing to the tag. Then, the controller can estimate the track, course and speed of the tagged animal and re-program the mission path in order to successfully track it down. Even though, it can estimate the three dimensional position of the shark, it requires an initial assumption of the tag location.

```
Algorithm 1 PF_Shark_State_Estimator($\{X^p\}$, $X_{auv}$, $Z_\alpha$)
//$Prediction$
for all $p$ particles do
    $v^p_{rand} \leftarrow v^p + randn(0, \sigma_v)$
    $\theta^p_{rand} \leftarrow \theta^p + randn(0, \sigma_\theta)$
    $x^p_{shark} \leftarrow x^p_{shark} + v^p_{rand} * \cos(\theta^p_{rand}) * \Delta t$
    $y^p_{shark} \leftarrow y^p_{shark} + v^p_{rand} * \sin(\theta^p_{rand}) * \Delta t$
    $v^p \leftarrow \gamma_{vt} * v^p + (1 - \gamma_{vt}) * \frac{\sqrt{(y^p_{shark}-y^p_{prev})^2+(x^p_{shark}-x^p_{prev})^2}}{\Delta t}$
    $\theta^p \leftarrow \theta^p_{rand}$
    if $\alpha$ is valid then
        $\alpha^p_{exp} \leftarrow \text{atan2}(y_{auv} - y^p_{shark}, x_{auv} - x^p_{shark}) - \theta_{auv}$
        $\alpha^p_{exp} \leftarrow g(\alpha^p_{exp})$
        $w^p \leftarrow h(Z_\alpha, \alpha^p_{exp})$
    end if
end for

//$Correction$
if $\alpha$ is valid then
    $\{X^p\}_{temp} \leftarrow \{X^p\}$ for all $p$
    for all $p$ particles do
        $X^p \leftarrow RandParticle(\{X^p\}_{temp})$
    end for
end if
```

Figure 4.2: Particle Filter Algorithm, taken from [3]

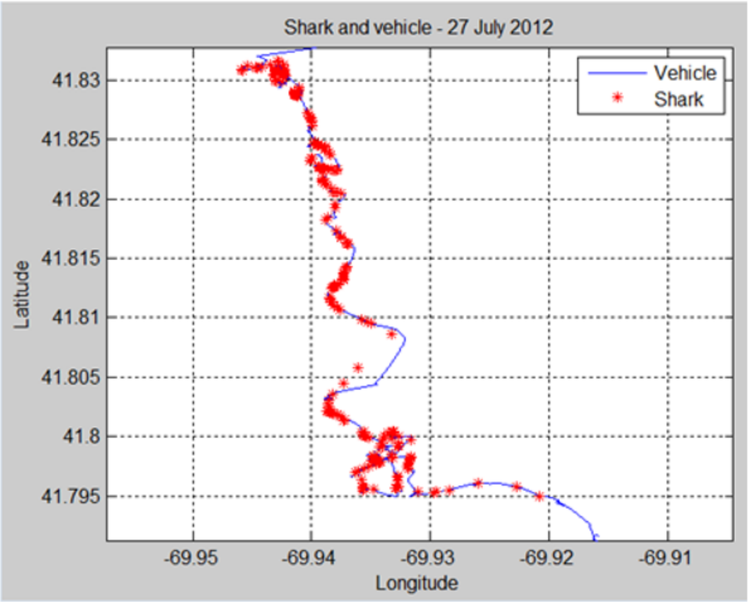


Figure 4.3: Following mechanism, taken from [4]

As you can see from Figure 4.3, where the red asterisks are the computed pings from the transponder, the vehicle is capable of continuously following the tagged fish without losing it.

Actually, the same authors from [4] have significantly improved their system, regarding the software and hardware, in order to fulfil more ambitious missions. Even though the vehicle remained the same, it had six high definition video cameras allowing a panoramic view of the tagged animal from the vehicle. This work can be seen in [19].

Also, the transponder tag could now endure depths up to 350 meters, which was a great achievement since white sharks on that region hardly go deeper than that value. Its battery life could also withstand multi-day tracking. Apart from that, another improvement was the introduction of an effective release mechanism. It would active whether in response to an acoustic order or when it reached the rated depth of 350m. Finally, they added a camera to the tag to provide a close look of the shark behaviour.

These researchers have also developed a shipboard tracking system to individually track the AUV and the tagged shark. This was important because if the AUV was compromised and had to be taken out of the water, remote operators could still keep track of the shark. Even though, they were only interested in following one tagged fish, it had a tracking interface that could pinpoint multiple transponders, which could be useful for multiple fish tracking in the near future. Like the vehicle, it uses a circular USBL acoustic array and a GPS. In addition to this, it also had an WHOI micro-modem to retrieve data regarding the global system status and a magnetic compass for updating the position of the AUV and the tagged shark. Another interface could provide range predictions from the AUV and the shark.

What were the biggest changes regarding software? Well, for starters, it could now adjust its speed autonomously. When the vehicle was too far from the tag (bigger than a given distance threshold), it would speed up and, when it was too close it could either slow down or fly over to achieve other points of view. Like in the previous work, remote operators could still change the vehicle velocity. Another important improvement, was the future fish prediction based on past movements. Using future prediction allows the vehicle to respond faster to shark quick changes in its course, making it easier to follow. Unfortunately, there is nothing in the article explaining how they did this. As in [4], the vehicle would interrogate the tag continuously, and, by measuring the time of flight between the request and the response messages, it would estimate the relative range. After the first response, the tag would wait a certain amount of time, proportional to the shark current depth, before sending another ping. Knowing the communication speed and the time delay between responses, it would be possible to retrieve a good approximation for the shark's depth.

However, as it was already explained in a previous chapter, tracking with a single vehicle has several problems. Actually, the researchers in [5] implemented a system of multiple coordinating AUV's since several modular vehicles can improve the system in terms of redundancy and performance.

This system consisted of two *Oceanserver IVER2* AUVs equipped with GPS and a digital compass, used to estimate the position of each vehicle. Each AUV had two processors used for the same purpose as in [3]. They also have a Lotek MAP600RT stereo-hydrophone set and receiver to deal with the acoustic communications.

Moreover, the fish tags broadcast messages at a frequency of 76 kHz and this would allow the AUVs to know their relative position and orientation to those tags.

Additionally, the communication between the AUVs is done by a WHOI Micro-Modem. The first one receives messages from every AUV in the system, containing their position, orientation and their Lotek values. This values are the computed range and bearing, measured by the hydrophones. Still regarding the communications, each AUV could only send message to the other vehicles if it had received a clear order from the computer. Only when it sent confirmation to the computer would this send the same command to another AUV.

For the state estimation, this researchers used a multi-AUV Particle Filter (PF). Basically, the PF has a set of particles P that represent the shark's position, orientation, velocity and, also, the weight of this particle. Like the particle filter from [3], it has two steps: (1) prediction and (2) correction. During the first step, each particle is updated using a stochastic motion model, specifically, a random walk algorithm. If the AUV received a valid measurement it would compute the expected angle according to its position and the particles position. The weight is updated after comparing the expected and the actual Lotek angle. The other step, consists of normalising all the particle weights and select new randomise particles with probability respecting their old weight.

For the tracking itself, they used a decentralised target circumnavigation. This can be shown in Figure 4.4. Basically, instead of mimicking the fish movement, each AUV would drive around the target using a circular path. By doing this, they hope to minimise any changes in behaviour of the targeted shark. In order to cover larger ranges, the path radius was different for each AUV. Also, each vehicle had to have phase difference minimising the probability of collision with other vehicles.

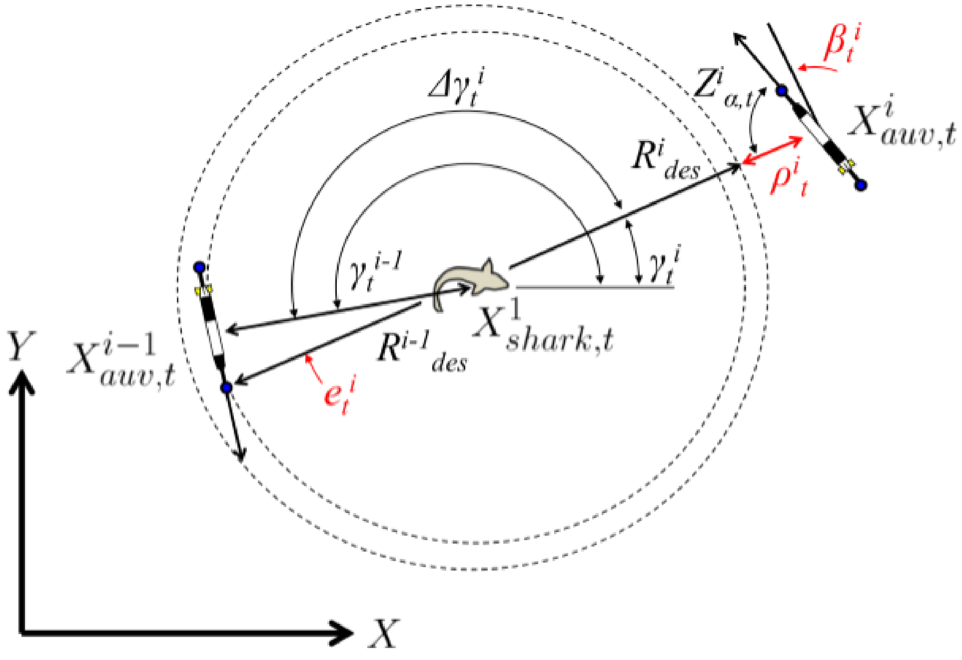
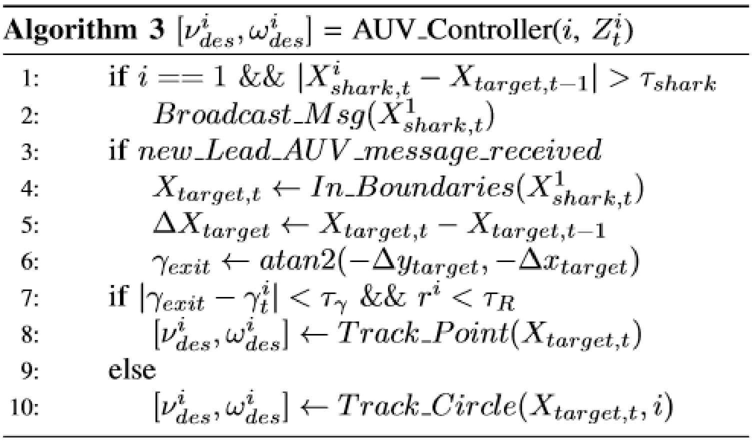


Figure 4.4: Circumnavigation mechanism, taken from [5]

Even though they used a system of multiple vehicles throughout the missions, there was always a leading vehicle. The leader AUV was responsible for the targeted shark. Its role was to update the target's position estimate and inform the others about it. The controller algorithm, which is going to be explained now, is summarised in Figure 4.5. The leader would keep checking if the shark had moved enough relatively to the previous tracked position. If positive, the leader should broadcast the new shark position to every AUV in the system. The other vehicles will then reset their target and update with the closest position to the received location. This means that, if the new position is outside their predefined surveillance area, they should circumnavigate the closest point on the boundary of their area.



Algorithm 3 $[\nu^i_{des}, \omega^i_{des}]$ = AUV_Controller(i, Z^i_t)

```
if $i == 1$ && $|X^i_{shark,t} - X_{target,t-1}| > \tau_{shark}$
    $Broadcast\_Msg(X^1_{shark,t})$
if $new\_Lead\_AUV\_message\_received$
    $X_{target,t} \leftarrow In\_Boundaries(X^1_{shark,t})$
    $\Delta X_{target} \leftarrow X_{target,t} - X_{target,t-1}$
    $\gamma_{exit} \leftarrow atan2(-\Delta y_{target}, -\Delta x_{target})$
if $|\gamma_{exit} - \gamma^i_t| < \tau_\gamma$ && $r^i < \tau_R$
    $[\nu^i_{des}, \omega^i_{des}] \leftarrow Track\_Point(X_{target,t})$
else
    $[\nu^i_{des}, \omega^i_{des}] \leftarrow Track\_Circle(X_{target,t}, i)$
```

Figure 4.5: AUV controller algorithm, taken from [5]

One year later, in 2014, significant upgrades were done to this system. In [6], a digital compass, a wireless antenna, a GPS receiver, and a 6-beam Doppler Velocity Logger are required to accurately estimate the vehicle state. Even though the actual tracking mechanism remained the same, it took into account the intensity of the signal Z^i_β and the depth of the tag Z^i_γ to estimate the state of the shark. Unlike any of its predecessors, with this new values the system is able to estimate the three dimensional coordinates of the shark.

Furthermore, they have discussed several models to update the particles of their filter, namely the Brownian, the Levy Flight random walk and the Hybrid motion model. During extended periods without sensor measurements, the Brownian model throws the particles into areas far away from the shark true location. To solve this problem, they tried to use the Levy Flight random walk model. Unfortunately, this model would spread the particles too aggressively, which is specially awful when the tag is moving slowly or not moving at all. Finally, they've decided that using an Hybrid Levy and Brownian model was the ideal solution since each model could counter the issues induced by the other. But how does the vehicle decide which model should use? After taking a close look at Figure 4.7, it is possible to realise that ρ is the probability of picking the Brownian model instead of Levy Flight. However there are two ways to compute this variable: (1) fixed and (2) adaptive. The first one means that this probability is unchangeable and can be selected just from knowledge about the shark behaviour. With the other one, ρ keeps being adjusted according

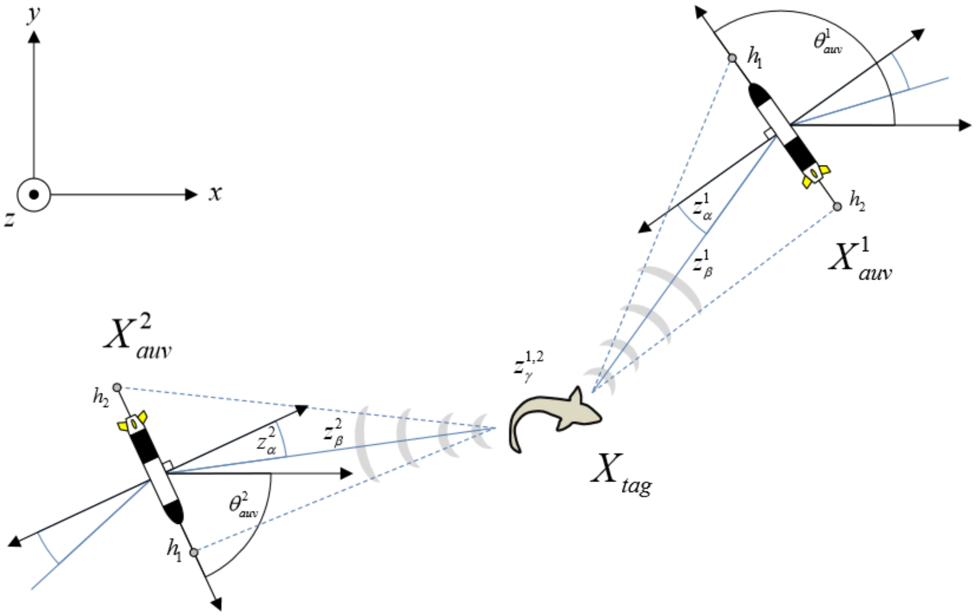


Figure 4.6: State Estimation, taken from [6]

to the likelihood of the measurement. Therefore, the bigger the number of time steps τ, since a valid measurement has been received, the lower the probability of selecting the Brownian motion model and so on.

Algorithm 2 Hybrid Random Walk Motion Model

```
$\theta^p_{tag} \leftarrow \text{UniformDistribution}(0, 2\pi)$
$v^z_{tag} \leftarrow \text{NormalDistribution}(0, \sigma_{vz})$
// $\rho$ is a number between 0 and 1
if $\rho \geq$ UniformDistribution(0, 1) then
    $v^{xy}_{tag} \leftarrow \text{ParetoDistribution}(k, \lambda)$
else
    $v^{xy}_{tag} \leftarrow \text{Abs}(\text{NormalDistribution}(0, \sigma_v))$
end if
$x^p_{tag} \leftarrow x^p_{tag} + |v^{xy}_{tag}| \cos(\theta^p_{tag}) \Delta t$
$y^p_{tag} \leftarrow y^p_{tag} + |v^{xy}_{tag}| \sin(\theta^p_{tag}) \Delta t$
$z^p_{tag} \leftarrow z^p_{tag} + v^z_{tag} \Delta t$
```

Figure 4.7: Motion Model algorithm, taken from [6]

Like [5], this system also circumnavigates only at the surface. This is interesting since in a system of UAVs, as in this thesis, they will also have to track the fish at the surface. Knowing that they were able to achieve good results in both [5] and [6] motivates the use of UAVs for tracking. Note that, one of the theoretical disadvantages of using aerial vehicles was having to pinpoint the target from the surface.

The most recent work was published near the end of 2015. In [12], a system of *REMUS*

SharkCam AUVs provided video records of white sharks during daylight or even throughout the total darkness of the sea deep waters. Therefore, it was possible to witness distinctive behaviours such as feeding and foraging of this sharks which weren't easy to observe until then.

The system was composed by two types of vehicles: the SharkCam-100 and the SharkCam-600. The first one requires daylight and can't go deeper than 100 meters while the other one has *SeaLite Sphere* lights which allows this vehicle to record during night and up to 600 meters below the surface.

According to this article, for short term periods, their algorithm would lead to feasible estimators. It consists in estimating the shark future position, while assuming that the shark will swim in a straight line without significantly varying its velocity, minimising the energy expended. The vehicle runs this algorithm many times a seconds in order to know where the shark will be in the near future. With this information, the vehicle will compute its trajectory and velocity to reach the given position on time.

However, even though they have a system of two types of vehicles, they don't really interact with each other since they are never deployed at the same time. Actually, the second one is only used when the shark goes to depths unreachable for the first AUV. Additionally, when the vehicles are running out of battery, the boat has to fetch it back which can be impractical for a system with several vehicles workings at the same time.

This work was specially important since it explained a verified algorithm for chasing sharks using future position estimation. Linking this algorithm to the particle filter of [6] should be helpful for guiding the software design and the estimation algorithm of the autonomous vehicles.

4.2 Estimation of fish behaviour

Learning more about the shark behaviour, could help us predicting its movement. This is specially important when we deal with sudden signal loss. Unfortunately, there is not much information regarding this subject.

4.3 Multiple target tracking

This thesis work is a particular case of the multiple fish tracking. Even though we are going to design a system to follow one tagged shark at a time, it would also be interesting to learn how multiple fish tracking is currently being done. Unfortunately, no such works have been done yet. For that reason, I will make a brief introduction to the multiple target tracking, presenting some of the most common techniques.

An extensive variety of methodologies rely on the recursive update of tracks with newly made observations. For smaller amounts of targets, the Kalman Filter is an efficient approach for the multitarget tracking [20]. However, when the number of targets increases, identity switches occur more frequently and are not easily corrected due to the recursive nature of this filter.

By exploring multiple hypothesis, Particle Filter techniques can address some of the flaws of the Kalman Filter method. In [21] this technique has accurately tracked people on the ground and planes, at the same time. Other algorithms from the same family like the Probability Hypothesis Density filters, were also used to track multiple objects from noisy observations [22]. However, these methods require careful tuning of several parameters and only look at small time windows since their state space grows exponentially with time.

In order to increase the tracking reliability, some methods use an hybrid approach where observations are initially grouped into small tracks, which are then combined through a higher level method. For example, in [23] they use Kalman filtering to generate basic tracks and then try to merge and split the tracks using the Hungarian algorithm. Similarly, in [24] observations were first turned into trajectory segments using local Principal Component Analysis (PCA) and then, these segments were linked based on their spatial proximity. Even though these techniques are more efficient than the previous ones and contemplate a wider observation window, they do not guarantee convergence to a global optimum. This can lead to wrong identity assignment. In order to make them more robust, they are associated with different optimisation algorithms such as Dynamic Programming [25], Linear Programming [26], among others.

4.4 Verified control architectures

The authors of [14] describe a control architecture for the cooperative implementation of a search algorithm. This work advances the state of the art in the sense that it introduces a formal specification for the implementation of the algorithm and then shows that the design indeed satisfies the specification in sense of a bi-simulation relation. The work is developed in the framework of hybrid automata. The mathematical framework is used to prove the properties of the design under two assumptions: that the way point generation algorithm produces feasible way points and that execution control ensures that feasible way points are reached within the prescribed time windows.

In this work the specification for the system is given as a transition system, Equation 4.1. The team has four states. During each mission, the team switches between two states - *Team Coord* and *Team Motion* - until the mission is completed - *Team Stop*. In the *Team Coord* state the vehicles wait for way points while in the *Team Motion* they are moving according to received way points. The fourth state - *Team Reconfig* - is used when the roles of the vehicles have to be re-allocated.

$$T_{spec} = (Q_{spec}, \rightarrow, I_{spec}, \varnothing, TeamCoord, TeamStop) \tag{4.1}$$

This system follows a master-slave hierarchy. One vehicle is the master and the other vehicles are slaves. Each vehicle has a controller that allows them to switch between modes (eg., stop, motion,...). Each system module is modelled in the framework of hybrid automata represented in Equation 4.2. Therefore, each vehicle is abstracted by a transition system T. The transition systems for the master and the slaves are given by Equations 4.3 and 4.4, respectively.

$$T = (Q, \rightarrow, I, O, Init, Final) \tag{4.2}$$

$$T_M = (Q_M, \rightarrow, I_M, O_M, Init_M, Final_M) \tag{4.3}$$

$$T_{S_1} = \cdots = T_{S_{N-1}} = (Q_S, \rightarrow, I_S, O_S, Init_S, Final_S) \tag{4.4}$$

This architecture encodes the team control logic and the motion control logic for each vehicle. Even though, every vehicle has the same control structure, they can be assigned with different roles, granting different control configurations. In this control architecture the team switches between two modes: (1) way points generation and (2) execution control. The latest is responsible for motion and manoeuvre control.

During the first mode, as the name suggests, a set of way points and coordination times are produced. To ensure that the vehicles can communicate during the communication phase, the coordination times must contemplate the following criteria: (1) the master vehicle must arrive to a certain way point before t_1; (2) the slaves must arrive to their way points and send a message to the master confirming it, before t_2; (3) the communication phase must end before t_3. Note that, each vehicle must receive their new way point before t_3 expires and that, during the communication phase, they should remain close to their designed way point.

The synchronous composition of the control structures of each vehicle gives an automaton that models the behaviour of the team. This automaton (Equation 4.5) satisfies the system specification in the sense of bi-simulation.

$$T_{comp} = T_M || T_{S_1} || \cdots || T_{S_{N-1}} \tag{4.5}$$

A transition system simulates another transition system when a transition taken from the first one can be replicated by a transition taken by the second one. Bi-simulation relationship happens when two systems simulate each other. The authors of [14] showed that the team specification and the composition of the team controllers exhibit a bi-simulation relation, Equation 4.6. As a result, they proved that this control architecture respects the system specification.

$$T_{comp} \sim T_{spec} \tag{4.6}$$

Chapter 5

Approach

In the previous chapter, we have described the evolution in the shark tracking methodologies. Currently, the most effective system is composed by a team of AUVs. However, they are not fast enough to follow some sharks, specially when they go on burst swims. Therefore, using a system of cooperative UAVs in order to track sharks may be a solution for this problem. Tracking acoustic signals at the surface has some inherent limitations. These are: the UAVs cannot dive in the water and the maximum depth at which the shark signal can still be heard is given by the range of the acoustic signal device.

In this chapter we present our approach to the fish tracking problem. Initially, some definitions and the solution to the problems defined in Chapter 3 will be given. While exploring the team control solution, we start with the detailed multilayered architecture of a system of cooperating UAVs and then go into detailed presentation of all components.

Finally, we will present the shark tracks used to validate the system in Chapter 6 and how they were generated.

5.1 Definitions

We need some definitions to help understanding our approach. The definitions build on the terminology introduced in Chapter 3.

Definition 4 (Track Lists)**.** The track list is a set of all the targeted shark observations. Recall that, each observations is characterised by its unique frequency f, the time t when the observation was made, the UAV *(x,y)* location and the distance d to the acoustic tag. Therefore, the track list can be represented as follows:

$$TrackList = \Big\{ observ_1 = (f_1, t_1, (x_1, y_1), d_1), \cdots, observ_N = (f_N, t_N, (x_N, y_N), d_N) \Big\} \tag{5.1}$$

In this work, I am going to use two types of track lists: (1) individual track lists regarding each tag and (2) a global track list that stores the information about every tag. The latest is specially important for updating the data delivered by every vehicle.

Consider the function *Extract_List:* $\{GlobalTL, Reals^+\} \rightarrow IndividualTL$, where *GlobalTL* is the global track list, given by:

$$f \in F, F \subset \mathbb{R}^+, \quad Extract\_List(f, GlobalTL) = GlobalTL(f) \tag{5.2}$$

In Equation 5.2, f is the frequency of the tag which the vehicle must track. Instead of working with the entire track list, this function allows the vehicle to extract the data corresponding to the frequency of its target. Lets denote the output of this function as IndividualTL.

Consider the function

Individual_Fusion: $\{IndividualTL, Reals, Reals, Reals^+\} \rightarrow IndividualTL_{new}$ given by:

$$a \in (\mathbb{R}, \mathbb{R}) \text{ and } t \in \mathbb{R}^+,$$

$$Individual\_Fusion(a, t, IndividualTL) = IndividualTL_{new} \tag{5.3}$$

In Equation 5.3, a are the shark estimated coordinates and t is the current time. This function allows vehicles to record new observations in the fish track list. After this step, the vehicle has to let every other vehicle know about this update. For that reason, the global track list must be updated.

Consider the function *Global_Fusion:* $\{GlobalTL, IndividualTL_{new}\} \rightarrow GlobalTL_{new}$ given by:

$$Global\_Fusion(GlobalTL, IndividualTL_{new}) = GlobalTL_{new} \tag{5.4}$$

Definition 5 (Route)**.** Waypoints(a,b) is a function that, given the coordinates of the vehicle – a – and of its destination – b, returns the list of way points coordinates (x,y) that describe the route of the vehicle. For every pair of coordinates, it also has a binary variable, set to 1 when the vehicle should land on that way point.

Time(a,b,t) is a function that, given the coordinates of the vehicle – a – and of its destination – b, returns the maximum time for arriving at each way point location (x,y) in order for the vehicle to successfully track the fish. It takes into account the current time t.

The route of the vehicle is given by the composition of these two functions. Therefore, consider the function:

Reference_Route: $\{Reals, Reals, Reals, Reals, Reals^+\} \rightarrow \{Reals, Reals, Binary, Reals^+\}$ given by:

$$a \in (\mathbb{R}, \mathbb{R}), \;\; b \in (\mathbb{R}, \mathbb{R}) \text{ and } t \in \mathbb{R}^+,$$

$$Reference\_Route(a,b,t) = [Waypoints(a,b), Time(a,b,t)] \tag{5.5}$$

In Equation 5.5, a are the vehicle current coordinates and b are its destination coordinates. The goal of this function is to derive a list of way points to follow the target. The vehicle should reach each programmed destination within the computed time limit. It is assumed that the vehicle is capable of achieving its mission within the given time.

Definition 6 (Shark's Estimate)**.** Recall the definition of observations in Definition 3.1. Their structure is given by Equation 5.6.

$$observation = (f,t,x,y,d) \tag{5.6}$$

This function requires three simultaneous observations (same recorded time t), from the track list computed in Equation 5.2. Each observation was made from a different UAV. By solving the system of equations 5.7, it is possible to estimate the shark's position at that observation time.

$$\begin{cases} (x-x_1)^2+(y-y_1)^2+z^2 &= d_1^2 \\ (x-x_2)^2+(y-y_2)^2+z^2 &= d_2^2 \\ (x-x_3)^2+(y-y_3)^2+z^2 &= d_3^2 \end{cases} \tag{5.7}$$

In the system of equations 5.7, (x,y,z) are the shark's coordinates at time t and the (x_i,y_i,z_i) with $i \in \{1,2,3\}$ are the coordinates of the UAVs recorded on the observations. After computing the shark's estimated coordinates, all the three observations can be erased from the track list and replaced by a new observation. In this observation the coordinates will now belong to the shark. To differentiate from observations where the recorded coordinates belong to the UAVs, d is replaced by a zero, since the distance is no longer required.

Definition 7 (Future Prediction)**.** This function can only be used by teams of at least three UAVs.

Consider the function *Future_Predict:* $\{Reals, Reals, Reals, Reals, Reals^+\} \rightarrow \{Reals, Reals\}$ given by:

$$a \in (\mathbb{R},\mathbb{R}),\ \ b \in (\mathbb{R},\mathbb{R})\ and\ \ \tau \in \mathbb{R}^+,\ \ Future\_Predict(a,b,\tau) \tag{5.8}$$

In Equation 5.8, a are the target's estimated coordinates and b are the estimated target's velocity components. The goal of this function is to compute the target's predicted coordinates $attime\tau$ from the estimated target's position and velocity.

Definition 8 (Search Prediction)**.** The track list computed in Equation 5.2 is required for this type of prediction. If the track list has at least two temporally close observations, it is possible to predict the shark's future location using the equations from Definition 7. The observations fulfill the temporally close requirement if Equation 5.9 is satisfied.

$$t_1 \in \mathbb{R}^+,\ t_2 \in \mathbb{R}^+\ and\ \ \delta \in \mathbb{R}^+,\ \ |t_2 - t_1| \leq \delta \tag{5.9}$$

In this equation, t_1 and t_2 are the time at which the observations were made and δ the maximum time difference. If this requirement is not verified, the coordinates used are the ones from the most recent observation, in the track list.

The size of the area around the predicted coordinates is given by:

$$t_c \in \mathbb{R}^+,\ t_o \in \mathbb{R}^+\ \alpha > 1\ and\ \ \alpha \in \mathbb{R}^+,\ \ Size(t_c, t_o, \alpha) = \alpha(t_c - t_o) \tag{5.10}$$

In Equation 5.10, t_c is the current time, t_o is the time of the observation used and α is a predefined weight. From this equation, it is possible to see that the bigger the time difference, the bigger the search area around the predicted target's position.

Definition 9 (Landing)**.** Consider the function *Check_Waypoint:* $\{Reals, Reals, Reals, Reals\} \rightarrow Binary$ given by:

$$a \in (\mathbb{R}, \mathbb{R})\ and\ \ b \in (\mathbb{R}, \mathbb{R}),$$

$$Check\_Waypoint(a,b) = \begin{cases} 1, & \text{if the vehicle has reached the way point} \\ 0, & \text{otherwise} \end{cases} \tag{5.11}$$

In Equation 5.11, a are the coordinates of the vehicle while b are the coordinates of the current way point. If that way point is associated with a 1, given by the binary variable, the vehicle should start the landing procedure.

Definition 10 (Take off)**.** TakeOffTime(x,t,τ), is a function that, given the time when the vehicle has reached the way point and the maximum time from Equation 5.5, computes the precise moment in seconds, when the vehicle must take off.

Consider the function *TakeOffTime:* $\{Binary, Reals^+, Reals^+\} \rightarrow Binary$

$$x \in \{0,1\},\ \ t \in \mathbb{R}^+\ and\ \ \tau \in \mathbb{R}^+,\ \ TakeOffTime(x,t,\tau) \tag{5.12}$$

In Equation 5.12, x says whether the vehicle is at the current checkpoint or not. On the other hand, t is the time when the vehicle landed and τ is the maximum time, given by Equation 5.5, for the vehicle to arrive at that way point.

Definition 11 (Next way point)**.** Consider the function *Next:* $\{Binary, Reals^+\} \rightarrow Binary$

$$x \in \{0,1\},\ \ t \in \mathbb{R}^+,\ \ Next(x,t,\tau) = \begin{cases} 1, & \text{when the vehicle must take off} \\ 0, & \text{otherwise} \end{cases} \tag{5.13}$$

In Equation 5.13, x says whether the vehicle is at the current checkpoint or not. On the other hand, t is the take off time given by Equation 5.12. When the output is 1, the vehicle must start tracking the following way point.

5.2 Motion Control

The solution to the motion control problem was designing an interface that could isolate the low level control from the high level coordination control. This design element allows the system to abstract the low level control by one or more manoeuvres, sent as commands by the high level control. Which actuators should be activated to change the UAV's orientation, perform take off and landing manoeuvres or increase/decrease the UAV velocity, does not concern the vehicle controller but the designed interface.

We found that we only required one manoeuvre: the manoeuvre "go to way point". In this manoeuvre, the vehicle has to follow a set of way points at a fixed velocity. The generation of this way points is described in Definition 5. Every time the vehicle reaches a way point, it uses the function described in Definition 9 to figure out if it should land on that way point or not. If positive, after landing, the UAV will have to wait on that location until the timer described in Definition 10 expires. In both cases, the UAV uses the function in Definition 11 to know when to move to the next way point. Note that the UAV only has to wait for the timer to expire and subsequently take off if it had to land on that way point.

5.3 Estimation

In our system, the UAVs measure the distance to their target with nothing more than the intensity of the target's acoustic signal. After receiving a ping, the UAVs virtually draw a sphere surface with radius equal to the measured distance. This provides them with a set of all the possible target's position. Therefore, after combining data from multiple vehicles, we can reduce the number of possible points.

Multiple estimation scenarios will be described in this section. Before moving on, there are two important things that you should remember while reading the rest of the section: (1) if the vehicles are estimating their target's position then at least one ping has been received by the system because estimation and, (2) all the internal angles, made by a squad triangular formation, must be greater than 35° to avoid singularities when triangulating the target.

5.3.1 Individual tracking

In order to estimate the position of the target with only one vehicle we need to take advantage of the speed difference between the shark and the UAV. The strategy here is to perform two small movements (eg., 20 meters) after a ping has been received by the vehicle. Hopefully, this will lead to three temporally close pings. After computing the intersection of these pings, we will be able to estimate the fish position. Then, the UAV should move to the estimated fish position and repeat the process.

$$\begin{cases} x_1 = x_0 \vee x_1 = x_t \\ x_2 = x_1 + \gamma \\ x_3 = x_1 + \frac{x_2 - x_1}{2} \end{cases} \tag{5.14}$$

$$\begin{cases} y_1 = y_0 \vee y_1 = y_t \\ y_2 = y_1 \\ y_3 = \sqrt{(x_2 - x_1)^2 - (\frac{x_2 - x_1}{2})^2} \end{cases} \tag{5.15}$$

In Equations 5.14 and 5.15 the (x_i,y_i,z_i) with $i \in \{1,2,3\}$ are the coordinates of the three vertices of the triangular movement. (x_0, y_0), (x_t, y_t) and γ denote, respectively, the initial position of the UAV, the estimated target position and the length of the base of the triangle. The initial position is only used at the start of the individual tracking. The estimated target position is computed as described in Definition 6. Instead of being three simultaneous observations as described there, it uses the three temporally close observations recorded in a complete triangular movement. In the system of Equations 5.7 the (x_i,y_i,z_i) with $i \in \{1,2,3\}$ are the three vertices coordinates that describe the complete triangular movement.

Certainly, since we are not using simultaneously recorded observations to estimate the target's position, the position obtained is affected by a measurement error that cannot be ignored. Additionally, with only one ping, the vehicle can't differentiate a situation where the shark is moving away in terms of its horizontal position - (x,y) - from a situation where the shark is very close to the UAV's (x,y) coordinate but diving deeper. Therefore, the vehicle has to keep hopping around, making this method high battery consuming. However, with this strategy, a UAV can follow a shark alone with nothing more than the intensity of the acoustic signal.

5.3.2 Cooperative tracking

In cooperative tracking, a team of UAVs is pursuing the same target. We have decided that a team has a maximum number of four elements. For this reason, there are four different situations when a team is estimating the position, velocity and direction of a target. This situations are directly associated to the number of vehicles that have received a ping from their target, at the same time. However, the number of cases can be reduced to three, since the procedure is the same whether there are three or four pings.

5.3.2.1 One ping

The goal of every strategy is to increase the number of simultaneous pings received by the team, since it helps improving the accuracy of the estimation. When we are discussing cooperative tracking, tracking with only one ping is not an option. For this reason, I have developed a team switching formation that will hopefully help increasing the number of pings received by the team.

Basically, the UAV that has received the ping should remain still, while the others move around the circular area, drawn from the intensity of the received acoustic signal.

In order to increase efficiency, the switching formation changes according to the number of vehicles. Figures 5.1, 5.2 and 5.3 represent the strategy for a team of two, three and four vehicles, respectively. The x denotes the position of the vehicle that has received the ping, the small circles are the way points for its teammates and the circumference is the intersection of the ping (sphere cap) with the water surface. The team way points coordinates are given by the Equations 5.16 and 5.17.

$$x = \begin{cases} x_1 + \beta \times r & \text{east way point} \\ x_1 - \beta \times r & \text{west way point} \end{cases} \tag{5.16}$$

$$y = \begin{cases} y_1 + \beta \times r & \text{north way point} \\ y_1 - \beta \times r & \text{south way point} \end{cases} \tag{5.17}$$

In these equations, (x,y), (x_1,y_1),r and β denote, respectively, the way point coordinates for a UAV in the team, the way point coordinates of the UAV that received the signal, the radius of the sphere cap (distance to the ping) and a tuning variable. By changing β, we can decide how close to the circumference should the way points be.

After reaching their new places, if the team didn't manage to receive more than one ping, they should switch to the next formation. However, the radius - r - of the circular area must still be updated every time.

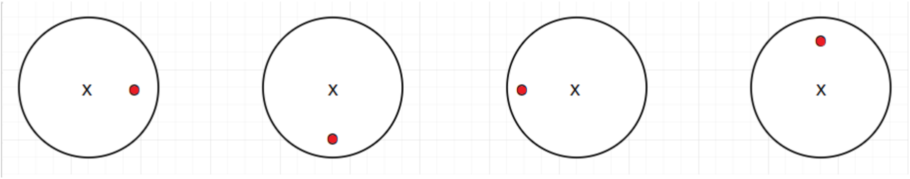


Figure 5.1: Switching formation with two vehicles

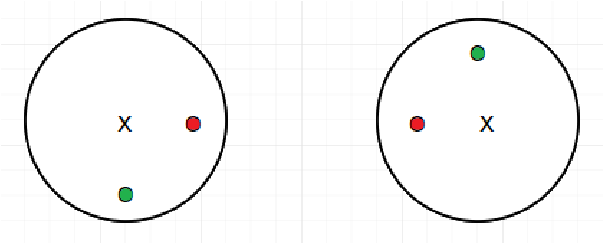


Figure 5.2: Switching formation with three vehicles

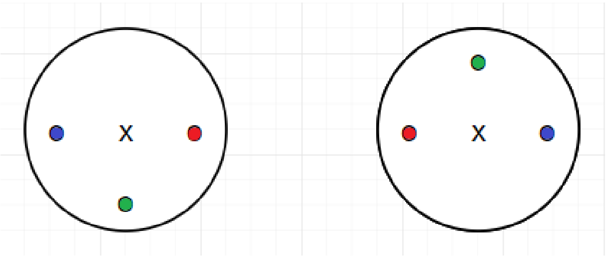


Figure 5.3: Switching formation with four vehicles

5.3.2.2 Two pings

The intersection of the two sphere surfaces, made by the pings, is a closed curve that crosses the sea surface in two points. Note that, this is only true if the distance of both vehicles to the target is not the same while the shark is at the zero depth. Assuming that the shark is not above the water we can ignore the top half curve of the intersection. Unfortunately, with only these two pings we are not able to calculate the depth of the shark. Actually, the only thing we know for sure, is that the fish x and y coordinates are within the line drawn between the two surface points. The two surface points can be computed by solving the system of equations 5.18. In these equations the (x_i,y_i) and d_i with $i \in \{1,2\}$ are the coordinates and the distance to the tag recorded on the two simultaneous observations. The line formed by the two surface points is given by an affine function like Equation 5.19.

$$\begin{cases} (x-x_1)^2+(y-y_1)^2 = d_1^2 \\ (x-x_2)^2+(y-y_2)^2 = d_2^2 \end{cases} \tag{5.18}$$

$$y = mx + b \tag{5.19}$$

Firstly, we slightly move one of the vehicles in the direction of one of those surface points. Since the UAVs are much faster than the shark, normally at least ten times faster, the shark won't be moving much since the last ping. Therefore, after receiving two new simultaneous pings, we can compute two new surface points. By intersecting the line, made by these two new points and the line from the previous intersection, we obtain an estimate of the fish x and y coordinates, Equation 5.20. The first equation of this system of equations is the line formed by the first two surface points and the second equation is the line formed from the two new surface points, computed after moving one of the vehicles.

$$\begin{cases} y = m_1x + b_1 \\ y = m_2x + b_2 \end{cases} \tag{5.20}$$

Then, move one of the vehicles in the direction of the estimated shark's position. Compute the new line, intersect it with the previous one and obtain a new estimate. Repeat this procedure in order to follow the shark.

Bare in mind that if the lines are almost parallel, the intersection point can be very far from the fish position, tricking the vehicle when it moves there, since it probably won't receive the signal there. In order to counter this algorithm's flaw, we assume that the shark can't move faster than eight meters per second, eliminating unrealistic points. In this case, instead of moving in the direction of the point, just move a small distance (eg., 20 meters) in any direction and continue the process.

Additionally, don't forget that the two pings situation can occur in a team of two, three or even four vehicles. For teams with more than two elements, one vehicle should remain while the others move according to the described algorithm. The team should move in a triangle formation if it has three UAVs or in a square formation if it has four. The goal here is to maximise the probability of receiving at least three simultaneous pings.

5.3.2.3 Three pings

Unlike the previous cases, in this one we are dealing with three simultaneously pings. This means that unless the shark is at the surface, we will always get two points from the intersection - Definition 6. One of those points is the real shark position and depth, while the other is the mirrored point in respect to the surface of the sea.

After two temporally close triple pings, we can compute the fish velocity and direction - Equations 5.21 and 5.22. This can be used to predict the future position of the shark (Definition 7), allowing the vehicles to move there once the signal starts to die out. Every time the team receives new triple pings, the shark position and velocity should be updated.

$$V_x = \frac{X_{EST_t} - X_{EST_{t-1}}}{\Delta t} \tag{5.21}$$

$$V_y = \frac{Y_{EST_t} - Y_{EST_{t-1}}}{\Delta t} \tag{5.22}$$

In this strategy, the team only has to move once the signal is lost or when the signal's intensity from the last UAV in the team, who still receives it, drops bellow a given threshold. For this reason, this strategy is the best regarding battery consumption.

5.4 Predictions

In our system, there are two types of prediction. The first one is when the UAV is in the presence of the acoustic ping. In this case, the UAV can compute the velocity and direction of the shark, V_x and V_y, from the data recorded on the observations. With this we can predict the near future shark's position (few seconds or minutes ahead). For this, we also have to assume that the shark will maintain a straight movement during this temporal difference. The calculation of the predicted coordinates can be seen in Definition 7. This type of prediction is only used for teams of three or four UAVs, since smaller teams or stand-alone vehicles use active estimation to follow the shark and not a waiting manoeuvre with a prediction algorithm.

The predicted coordinates of the target at time $t+\tau$ are given by Equations 5.23 and 5.24, where X_t and Y_t are the estimated coordinates of the shark and V_{x_t} and V_{y_t} are the shark's velocity components, at time t. The velocity components are computed in Equations 5.21 and 5.22.

$$X_{t+\tau} = X_t + V_{x_t} \times \tau \tag{5.23}$$

$$Y_{t+\tau} = Y_t + V_{y_t} \times \tau \tag{5.24}$$

The second type of prediction is done when the UAVs are locating the shark. Basically, past information recorded on their track list is used to predict the target's current position. If only one observation was made, we can't predict the direction of the shark's movement so they just search around that area. If there are more than two temporally close observations (in the order of minutes), it is possible to predict the direction and velocity of the shark, leading to possible future area where the shark might be. The searching area is proportional to the time difference between the current and the observation time. This algorithm is described in Definition 8.

5.5 Team Control Structure

5.5.1 Overall architecture

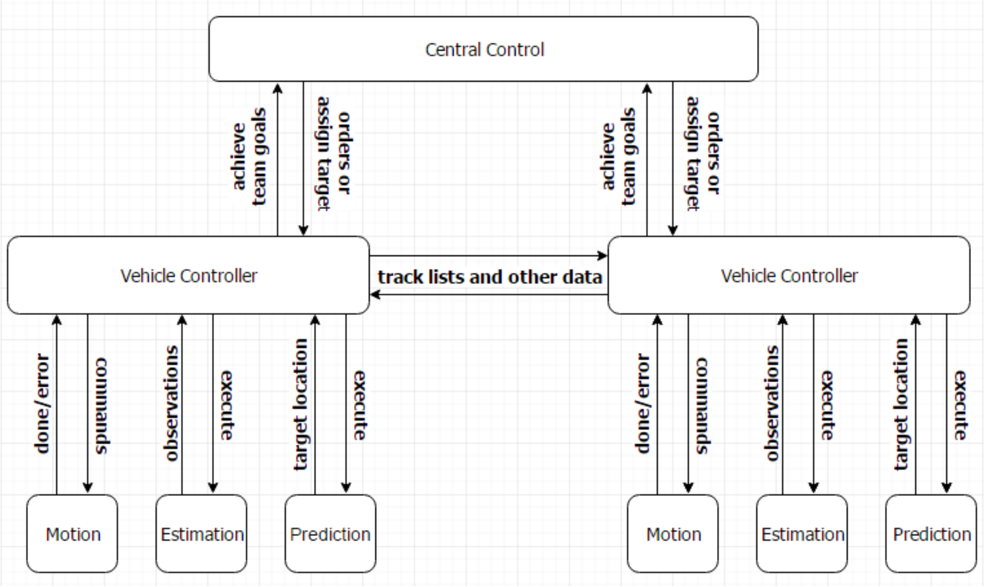


Figure 5.4: Control Architecture for a system of cooperative UAVs

Our system is composed by several UAVs, equipped with acoustic receivers, that can land on the sea in order to listen for acoustic pings, sent by the tags placed on the targeted sharks. These vehicles can work either alone or in teams. We proposed the control architecture depicted in Figure 5.4 to handle the coordination and control of the UAV system. Additionally, a central control station will also be used to coordinate teams of vehicles and assign targets to them.

The estimation of the target's position is done by combining the data from temporally close pings. The most accurate position estimate requires at least three simultaneous pings from different vehicles. From two, temporally close, position estimates, the system can compute the target's velocity and predict its future location, allowing the vehicles to move and wait there for it.

Regarding communications, the vehicles only communicate when they are not flying (Assumption 4). A vehicle radar is incorporated in every UAV, allowing them to know which vehicles are within range and if they are moving or not. Additionally, UAVs must share their track lists with any vehicle they encounter. The communications among vehicles is represented in Figure 5.4 by the arrows between the vehicle controllers. Moreover, if the UAVs are working in teams, they must wait for orders and answer to the central control requests.

5.5.2 Information structures

The information structures are design elements that are part of the set of fundamental ideas of the approach. They contain all the data required for coordinating the system (eg., recorded observations, team info, ...). Since sharing information is what differentiates individual from cooperative systems, the information structures are a crucial element in our cooperative tracking approach.

In this work, track lists were used in order to improve the system quality and cooperative shared information. As it was already said, they will be used to store all the data regarding each tag, recall Definition 4. Basically, every time a UAV receives information from a certain acoustic device, it should update the track list by introducing a new data row for that device frequency.

However, when should this information be shared? Well, there are two different situations. Every time a vehicle comes across another one, in other words, is within communication range, their global track list should be shared between each other. This will allow a constant flow of information between the team members, even if they are not currently tracking the same target. How can this be useful? Imagine that a vehicle was following target A but encountered another vehicle chasing target B. After a while, the first vehicle aborted its current mission to start tracking target B. If the vehicles hadn't shared their global track lists (Definition 4), the only thing the vehicle would know about its new target would be its initial position. However, the target could have moved far away from that position, making it impossible for the vehicle to find it. With the new data delivered by the other team member, the vehicle could have a more precise temporal information, increasing the probability of success of its new mission.

In the second case, the vehicles should share the data regarding their current target with the team control, when they are preforming a team mission. This will allow the control to adapt its tracking strategies to the information delivered by the team of vehicles.

Note that, there are other important individual details that should be transmitted between every vehicle in a team. This data should contemplate the following information : (1) the current state of the vehicle, (2) the number of times a vehicle has failed to locate the target, (3) if the vehicle has heard the tag signal while it was listening for it and, (4) if the signal is close - intensity higher than a predefined value. This information is critical for the team and individual vehicle controllers since it defines their state and procedure.

5.5.3 Vehicle controllers

Each UAV has two main movements: (1) search for their target until they receive the target's ping and, (2) estimate and follow the target in the presence of the acoustic signal. The searching algorithm is a very simple strategy, represented by Figure 5.5. The vehicle defines four way points around the target's last known position and travels to each one. Every time it reaches one of those way points, it waits for a certain amount of time, trying to find the acoustic signal. If it fails to locate it must move to the next way point. Finally, if it reaches the last way point and fails to spot the target, the vehicle will abort its current mission and wait for a new target to be assigned. This algorithm can be adapted for cooperative search (Figure 5.6). In this case, the searching area increases with the number of vehicles. Each vehicle is only assigned with a part of this area. Additionally, it is the control station that decides the way points for the team of vehicles.

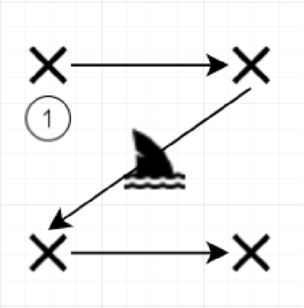


Figure 5.5: Search Strategy

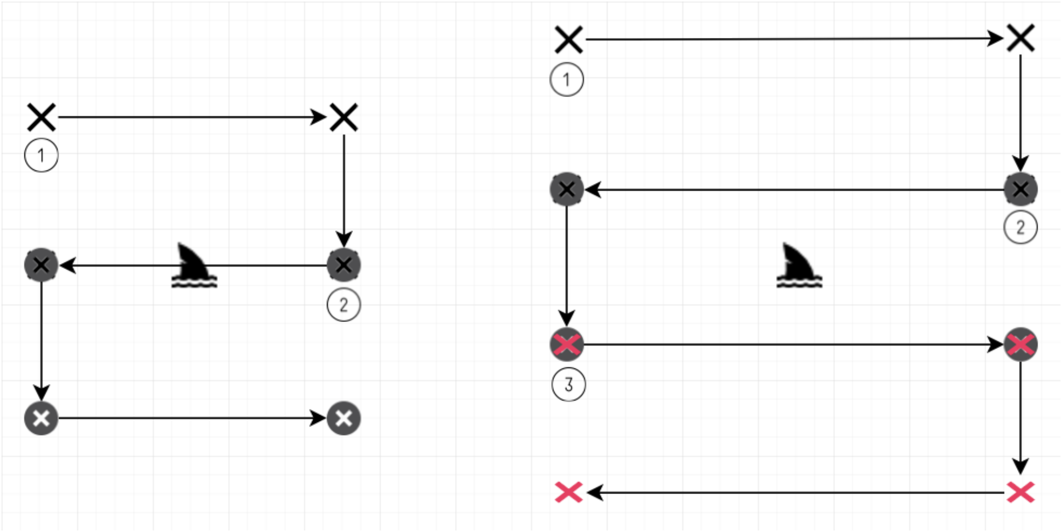


Figure 5.6: Search Strategy for 2 and 3 UAVs

Table 5.1 represents the automaton of the vehicle. The table has five entries: the current state q, the transition event s, the next state Φ, the variables that should be updated after the transition and the output event ϑ. Let $Count$, t, t_1, n and $prev_n$ denote, respectively, the number of times a UAV failed to find its target when searching for it, the current time, the listening timer, the number of vehicles with the same target shown in the vehicle radar, the number of vehicles in the team after leaving the listen state. From the point of view of each UAV, they can be either working alone or in teams. Therefore, they have states to differentiate solo from cooperative missions. Some input events such as "coopSearchWP" and "coopTrackWP", used for team coordination are provided by the central control.

The actions or low level functions a vehicle has to preform are defined by their current state. During the active estimation state, the vehicle controller queries the estimation function for observations, required for defining way points for the individual tracking. In a similar way, this also happens during the team estimation. However, in the latest the central station gives clear orders to the vehicle controller provide it with observations.

During the check state, vehicle controllers are responsible for communicating with nearby vehicle controllers in order to retrieve data. This helps them knowing which vehicles also have the same target, forming a team.

The prediction function can be executed during the search states, to maximise the probability of finding the target. It can also be used for near future prediction when defining way points during the actual tracking.

Finally, if the UAV has received/computed way points, take off commands must be sent to the motion control. The vehicle controller will then wait for confirmation that the manoeuvre was performed successfully or if an error occurred.

5.5.4 Composition of controllers

The composition of controllers happens when multiple UAVs have to work together to achieve a common goal. This requires synchronisation between the vehicle controllers. We have decided that there would be a maximum number of 5 teams and each team could have up to 4 members. Each team is characterised by its elements and by its state. The possible team states are the following: (1) listen, (2) wait, (3) cooperative search and (4) fly state.

On the first state, the vehicles are at the sea surface, listening for acoustic pings and updating their track lists with new shark position estimates. Once their listening timer is over, each element of the team must inform the control station that they are capable of moving on. Note that, the listening timer is computed by the control station, at the same time it decides the team's next way points. By taking into account the current position of each team vehicle, the control station can estimate the arrival time for each vehicle if they travel with a fixed velocity. Adding some seconds (eg. 20), for possible delays, to the maximum arrival time gives the listening timer. For this reason, all the vehicles will leave the listen state at the same time, avoiding deadlocks. On the wait state, since pings were retrieved by the team, they are waiting for new way points delivered by the station. While they wait, they keep listening for pings. On the other hand, at the cooperative

Table 5.1: Vehicle Automaton

$q \in Q$	$s \in \Sigma$	$\Phi(q,s)$	**reset**	$\vartheta \in \Theta$
Initial	assignedTarget	Listen	Count:=0 t:=0	-
Listen	abortMission	Initial	-	-
Listen	$t > t_1$	Check	prev_n:=n	send personal data share track lists
Check	$n \geq 1$ AND $Count = \delta$ AND $ping = 0\ \forall v \in vehicles$	Initial	-	-
Check	$n > 1$ AND $Count < \delta$ AND $ping = 0\ \forall v \in vehicles$	Coop Search	Count:=Count+1	send target TL send team data
Check	$n = 1$ AND $Count < \delta$ AND $ping = 0$	Search	Count:=Count+1	searchWP
Check	$n = 1$ AND $ping = 1$	Active Estimation	Count:=0	trackWP
Check	$n > 1$ AND $ping = 1\ \forall v \in vehicles$	Team Estimation	Count:=0	send target TL send team data
Coop Search	*coopSearchWP_i*	Fly	-	Take off
Search	searchWP	Fly	-	Take off
Active Estimation	trackWP	Fly	-	Take off
Team Estimation	coopTrackWP_i	Fly	-	Take off
Fly	reachedWP AND n=prev_n	Listen	t:=0	Land
Fly	n!=prev_n	Check	prev_n:=n	Land

search state the control provides the team with a sequence of way points. Since these way points are all delivered at the same time, this state is only used to update their current destinations. The flying state is reached when all the vehicles, at search or wait state, inform the station that they will take off. Finally, the team goes back to the first state once every vehicle has landed and informed the station.

5.5.5 Central Control

Operators in the control station have some other tasks that should be mentioned. They are the ones that decide where the UAVs should be deployed, what target should be assigned to each vehicle, when should this assignment occur and if a vehicle should abort its current mission. Certainly, there are some rules for some of these tasks. The deployment of more than two vehicles in the same area (within communication range) must be done in triangular or squared formations. Secondly, due to the maximum capacity of a team, the same target can’t be assigned to more than four UAVs. Assigning can be done any time after deploying. Finally, even though the UAVs can

Table 5.2: Control Station Automaton

$q \in Q$	$s \in \Sigma$	$\Phi(q,s)$	**reset**	$\vartheta \in \Theta$
$Elements_i = \{\}$ $State_i = \varnothing$	at least two vehicles with the same target within communication range	$Elements_i \neq \{\}$ $State_i = listen$	-	-
$Elements_i \neq \{\}$ $State_i = listen$	$size(Elements_i \leq 1$	$Elements_i = \{\}$ $State_i = \varnothing$	-	-
$Elements_i \neq \{\}$ $State_i = listen$	$signalClose = 0$ $\forall v \in team_i$	$Elements_i \neq \{\}$ $State_i = search$	-	$coopSearchWP_i$
$Elements_i \neq \{\}$ $State_i = listen$	$signalClose = 1$ for at least one $v \in team_i$	$Elements_i \neq \{\}$ $State_i = wait$	-	-
$Elements_i \neq \{\}$ $State_i = wait$	Take Off $\forall v \in team_i$	$Elements_i \neq \{\}$ $State_i = fly$	-	-
$Elements_i \neq \{\}$ $State_i = wait$	$signalClose = 0$ $\forall v \in team_i$	$Elements_i \neq \{\}$ $State_i = wait$	-	$coopTrackWP_i$
$Elements_i \neq \{\}$ $State_i = search$	Take Off $\forall v \in team_i$	$Elements_i \neq \{\}$ $State_i = fly$	-	-
$Elements_i \neq \{\}$ $State_i = fly$	Land $\forall v \in team_i$	$Elements_i \neq \{\}$ $State_i = listen$	-	-

be forced to abort their mission if requested, they must only stop their current mission once they land.

5.6 Generation of fish tracks from sparse observations

In order to test our approach, we need to have some fine-grained shark tracks. In fact, the simulation should be made as close to reality as possible since it will be used in future practical experiments by LSTS. For that reason, the shark tracks should be generated from real shark data. Fortunately, Dr. Nuno Queiroz provided us with two types of data: (1) shark's position in terms of latitude and longitude and (2) shark's depth variation. This data will be used to simulate the shark swimming behaviour according to latitude, longitude and depth. The LSTS toolbox was used to convert the GPS coordinates into rectangular x and y coordinates.

Firstly, it is important to understand the definition of shark tracks. These tracks are a set of way points that represent the course of a tagged shark. Many shark tracks were recorded in [9] and will be used to validate our approach. However, since they used GPS satellite tags which only transmitted data every 12 hours, there is no information, regarding the shark position, between transmissions. Due to this lack of temporal data, we had to make Assumption 5. With this assumption, it is possible to represent the complete shark track.

Assumption 5. The shark maintains a constant velocity and moves in a straight line, between every two consecutive GPS coordinates.

From the first set of data we were only able to generate two dimensional shark tracks. Therefore, a depth value must be added to every point of each shark track in order to have three dimensional shark tracks. The shark's depth variation data was also recorded on real sharks and provides the minimum and maximum depth of the shark between two consecutive data measurements. Note that, since sharks exhibit less steep depth transitions at night than at day, this data is divided according to their recorded hour. These depth boundaries should be randomly associated to every position of the two dimensional shark tracks. Then, the real depth of the shark is given by a random value between the two depth boundaries.

5.6.1 Fish models

As a result of the algorithm explained in the previous section, several tracks from blue sharks crossing the Atlantic Ocean were retrieved. There are a total of 24 different shark tracks.

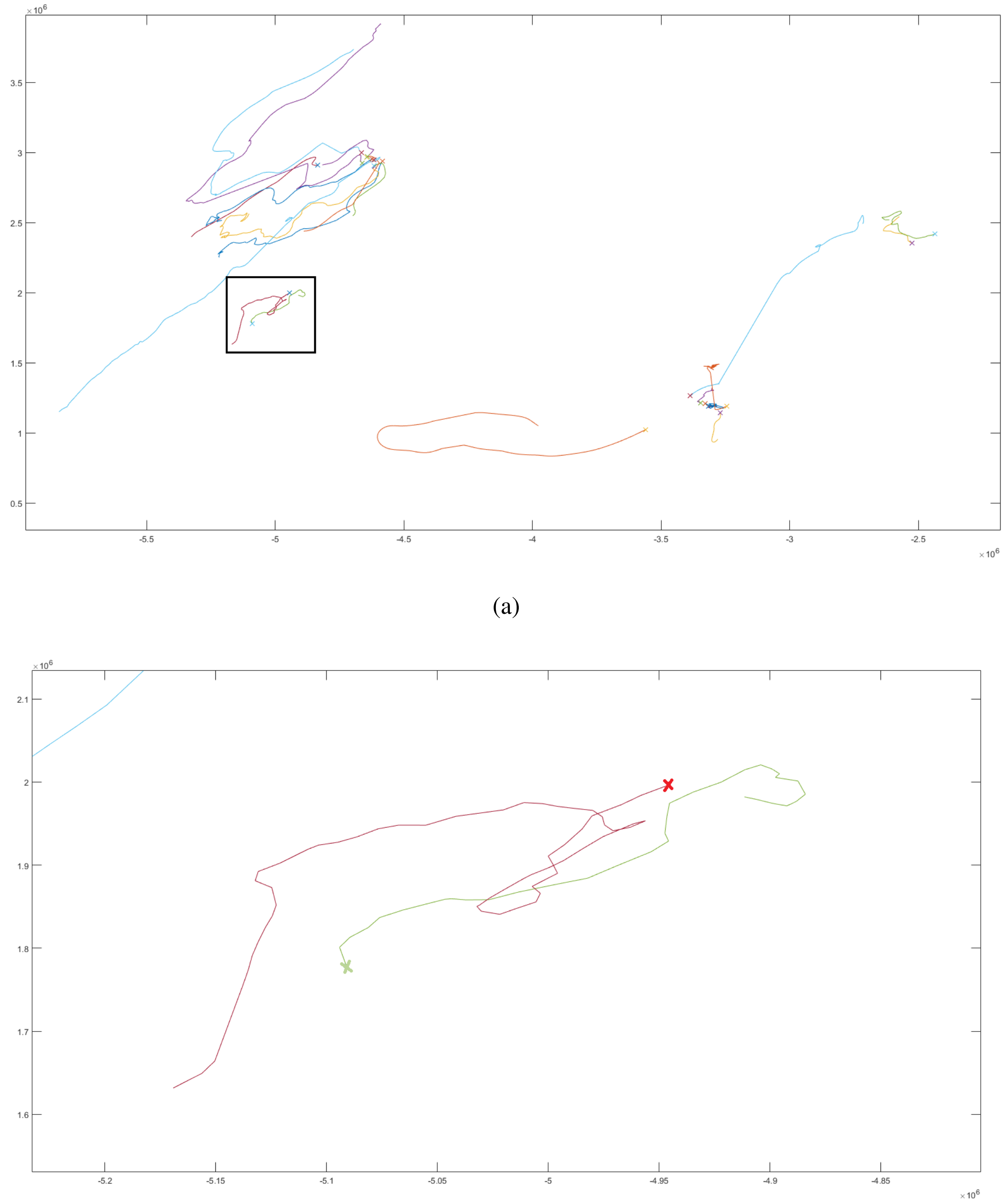
Since the shark tracks generated until now only have x and y coordinates, This can be seen in Figure 5.7a. It is important to notice that while some of these tracks correspond to, for example, a twenty-day run, others have a shorter or even longer duration.

Lets take a closer look of Figure 5.7a. In Figure 5.7b, we can clearly see two different shark tracks. The 'X' marks the initial position of the shark. Even though, both sharks started around the same area, their route was completely different. While the yellow shark exhibited a pretty straight movement, the red one, specially during the first days, kept changing its direction. There might be several reasons to explain this disparity but, probably, the latest shark was hungrier than the second and found that area to be abundant in prey. Also, these data might not have been recorded exactly at the same time (eg., during different seasons), leading to different animal behaviour.

Nevertheless, it would be interesting to obtain better shark models since, even though, sharks tend to move approximately in a straight line (eg., Levy motion model), they also exhibit random search or bigger route deviations. However, as Dr. Nuno Queiroz pointed out, the goal of this thesis is not designing good shark models. For that reason, using real recorded data to validate the tracking system should be good enough. Additionally, since the shark in this model never stops moving and maintains its velocity between every twelve hours, we are already forcing the vehicles to deal with an odd situation, where remaining still for more than two minutes can mean losing the target. Note that, the estimation and tracking algorithms designed do not take into account the straight movement of the shark. In other words, the vehicles do not know that the shark will maintain its direction and speed between every twelve hours.

And yet, these shark tracks are not completed until they have depth values associated to it. A maximum and minimum value is randomised every time the model shark changes its direction (every twelve hours). Figures 5.8 and 5.9 represent two possible examples of the shark's depth boundaries. The instantaneous depth value of the shark can be any value between these boundaries. It is possible to see that the shark can reach depths of 600 meters, making it impossible for the vehicles to track it since the maximum range of the acoustic signal is 500 meters (Assumption 1).

To sum up, even though the fine-grained tracks generated by this algorithm are not as complex as those occurring in nature, they can still make the tracking quite challenging.



(a)

(b)

Figure 5.7: (a) Recorded tagged shark tracks. (b) Closer look of two shark tracks.

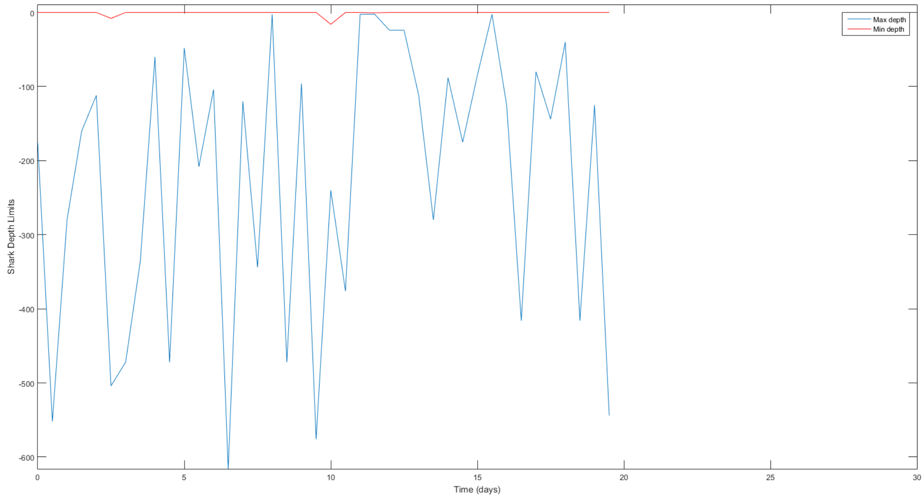


Figure 5.8: Shark's depth boundaries for a mission of 20 days

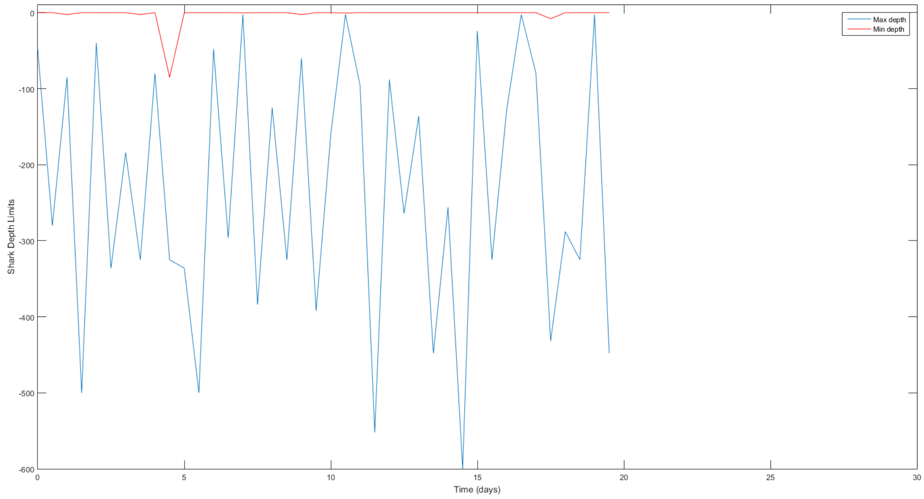


Figure 5.9: Shark's depth boundaries for a mission of 20 days

Chapter 6

Simulation studies

This chapter has the purpose of describing how we have tested our theoretical approach through several simulation runs. Firstly, I will provide the reader with a list of all the tests that will be used to validate the system. Secondly, I will briefly explain how the simulation works. Then, I will be discussing the test runs, exploring the initial conditions of the system, the vehicle behaviour and the system performance.

6.1 Test plan

A test plan was created in order to evaluate the performance and quality of the system. The measures of performance are: the mean error in the estimator (computed by the difference between the target's estimated and real position), how smoothly does the system track targets (qualitative measurement) and how frequently does it have to move to new positions. Smoothly here means that the movement of the UAVs always follows the same pattern, not random Instead of making an extensive list, I will be covering the most relevant system properties such as, the estimation strategies for different number of UAVs, simultaneous multiple team tracking, long mission runs, among others.

This plan is represented in the table below. For each main characteristic of the system (eg., estimation and searching strategies, multiple teams tracking multiple targets, ...) , I have provided the procedure which will be used to evaluate it. The procedure has a variable number of steps, being the first step (step 1) the initial conditions. The initial conditions have to be changed between runs in order to contemplate a greater number of different situations. Only the different situations will be presented and discussed.

Table 6.1: Test plan

ID	Test Description	Procedure
1	Individual tracking	1 - Deploy a vehicle near the initial position of a target 2 - Assign that target to the vehicle 3 - Wait until either the shark is lost or simulation time reaches 400s
2	Individual search	1 - Deploy a vehicle far from the initial position of a target 2 - Assign that target to the vehicle 3 - Stop once the vehicle has given up searching or it has found the target
3	Target estimation with 2 UAVs	1 - Place two vehicles near the initial shark position 2 - Assign that target to both vehicles 3 - Wait until either the shark is lost or simulation time reaches 700s
4	Target estimation with 3/4 UAVs	1 - Place three/four vehicles near the initial shark position 2 - Assign that target to all vehicles 3 - Wait until either the shark is lost or simulation time reaches 800s 4 - Confirm that the control has managed to compute the position and velocity of the target
5	Cooperative Search	1 - Deploy a group of vehicles far from the initial position of a target 2 - Assign that target to that group of vehicles 3 - Stop once the team has given up searching or it has found the target
6	UAVs joining other UAVs at the middle of the mission	1 - Make one vehicle or team of vehicles track a target 2 - Introduce a new vehicle at the middle of their mission 3 - Confirm they have changed their strategy
7	UAVs leaving their team during the mission	1 - Form a team for a given target 2 - Wait until they start tracking or searching 3 - Abort the mission for one of those vehicles 4 - Confirm they have changed their strategy
8	Multiple teams tracking fish at the same time	1 - Form two teams 2 - Wait until they have retrieved their tracking strategies 3 - Confirm that they are both working at exactly the same time
9	Long-run (Attempt to follow fish for at least 3 days)	1 - Assign a target to a vehicle or group of vehicles 2 - Wait until the simulation time hits 3-4 days 3 - Confirm they have either lost their target or successfully track it until then

6.2 Description of the simulation environment

The code developed follows a object-oriented programming language. The software used for this programming purpose was MATLAB. Since the system is composed by the UAVs, the tagged sharks and the central control, a class object was created to represent the functions and variables that describe each one of them. More than 2000 lines of code were written to simulate our system. The UAV Class has 410 lines of code covering functions such as search and individual tracking way points generation (triangular movement), update track lists and team data (information structures), among others. The Shark Class only has 90 lines of code since the shark does not have much actions. Actually, all it does is follow a three dimensional track generated by another script (which has 60 lines of code) by updating its position and depth. The Class that required more tuning and attention was the Central Control Class. This Class has more than 950 lines of code since coordinating multiple teams of vehicles is not an easy task. The rest of the code belongs to the main where all the Classes are called.

During the simulation runs, it is possible to see a live representation of the system behaviour - locating and/or tracking moving sharks. Only the UAVs and the sharks are drawn in the live simulation plot. This plot is a two dimensional representation of the simulation world. In order to make this simulation as close to reality as possible, a measure of time is associated to each iteration.

The simulation evolves in two steps: (1) an initialisation and (2) an Infinite loop. During the first one, we must decide: the number of sharks, the number of UAVs and where should the vehicles be deployed. On the second one, the system components are updated according to their actions. Initially, the sharks are associated to one of the shark tracks designed on the previous chapter. Throughout the entire simulation run, they just follow along the three dimensional shark track. In the live plot they are represented by an asterisk.

On the other hand, the vehicles are represented by the letters "x" and "o" or by squares and diamonds. They follow the automaton controller described in the previous chapter. Moreover, the received acoustic pings are also plotted. Since it is a two dimensional plot, we can only see the intersection of the sphere cap with the water surface. For that reason, they will only coincide with the shark's real position if the shark is at the sea surface.

The simulation time is updated by Equation 6.1. The variable step can be changed according to the user specifications. It allows the simulation to implement a fast forward mechanism. For example, if the step is 1 then it will take 86400 iterations to simulate one day, while if we make the step 50 it will only have to run 1728 iterations. This makes long-runs less complex. In terms of real time, this can be the difference between taking some minutes to some hours for simulating one day of tracking. Unfortunately, this is not accepted for all the tracking strategies.

$$time = time + step; \tag{6.1}$$

The operator only has two interactions with the simulation. Every time an UAV has no target, a pop-up message will be printed asking if he wants to assign a target. By pressing the number "0" he can ignore this message. If the UAVs already have a target, a message will appear from x on x seconds, asking if the operator wants to abort that vehicle's mission. The answer to this question must be "Y" for yes and "N" for no.

6.3 Discussion of the test runs

During this section, we will be discussing the test runs. For each test, several cases will be presented in a modular way: initial conditions of the system, description of the system's behaviour and the results. Using different initial conditions allows the observation of different system behaviours. For example, if we deploy the UAVs very close to the initial position of the shark we will maximise the probability of having more UAVs in the presence of the acoustic signal (as long as the shark depth allows that). This is important if we want to test a specific system characteristic. The example given is used to evaluate the performance of the estimation strategies.

The sharks in this simulation were associated to one of the shark tracks retrieved in the previous chapter. Their depth was computed with the algorithm described in the Data Processing section

6.3.1 Individual Tracking

Case 1.

Initial conditions:

- Shark's last known location (x,y) = (0,0)
- Shark's depth = 137 meters
- Vehicle's (x,y) coordinates = (0,50)

In Figure 6.1, a UAV is tracking a shark at 137 meters depth all by itself. By intersecting the data from three temporally close pings, it estimates the shark's position. The mean error of this estimate, computed from the difference between the real shark's and the estimated target's position, was of 21.4742 meters. Even with a small positional error, the vehicle is able to follow the target.

A simple assumption made calculations quite simple. Here we assume that the pings are received at the same time. This results in a tracking bias visible in all these graphs.

Case 2.

Initial conditions:

- Shark's last known location (x,y) = (0,0)
- Shark's depth = 271 meters
- Vehicle's (x,y) coordinates = (0,50)

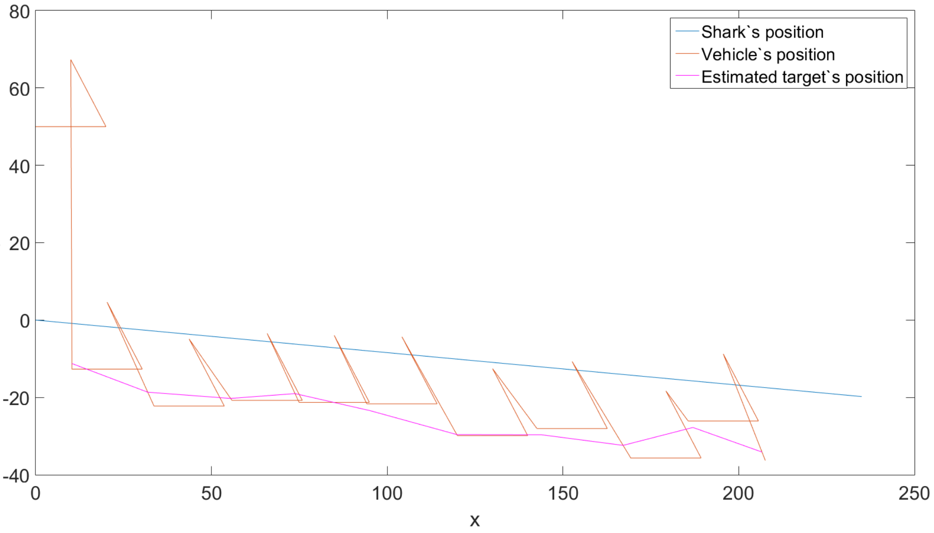


Figure 6.1: Tracking shark moving at 137 meters depth

In the Figure 6.2, a UAV is tracking a shark at 271 meters depth all by itself. The only difference from the previous simulation run is the shark's depth. Note that the performance of the UAV is identical to the first one. As a result, the error is also equal to 21.4742 meters. This run shows how the depth is irrelevant for the single vehicle estimation technique. In this technique, one of the aspects that influence the estimation error is the frequency at which the vehicle estimates the target's position (takes the three measurements). Since the frequency of the acoustic signal (>5 seconds; 10 seconds for this target) is bigger than the flying time of the UAV (<5 seconds), while collecting measurements, we can conclude that, in this strategy, the error is proportional to the ping's frequency.

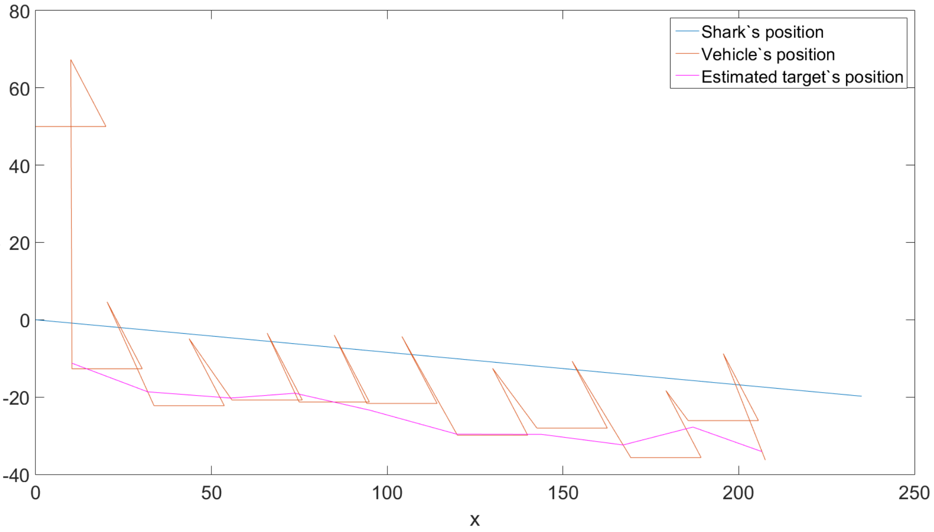


Figure 6.2: Tracking shark moving at 271 meters depth

Case 3.

Initial conditions:

- Shark's last known location (x,y) = (0,0)
- Shark's depth = 175 meters
- Vehicle's (x,y) coordinates = (-50,50)

In the Figure 6.3, the UAV has a different initial position. In this case, the mean positional error was of 24.7392 meters. Clearly, as we have seen from the previous runs, it takes a few target estimates before the error stabilises. However, after this one, we can see that the farther the UAV is from the shark, when it starts estimating, the bigger the initial measurement error. This will make the mean error slightly bigger.

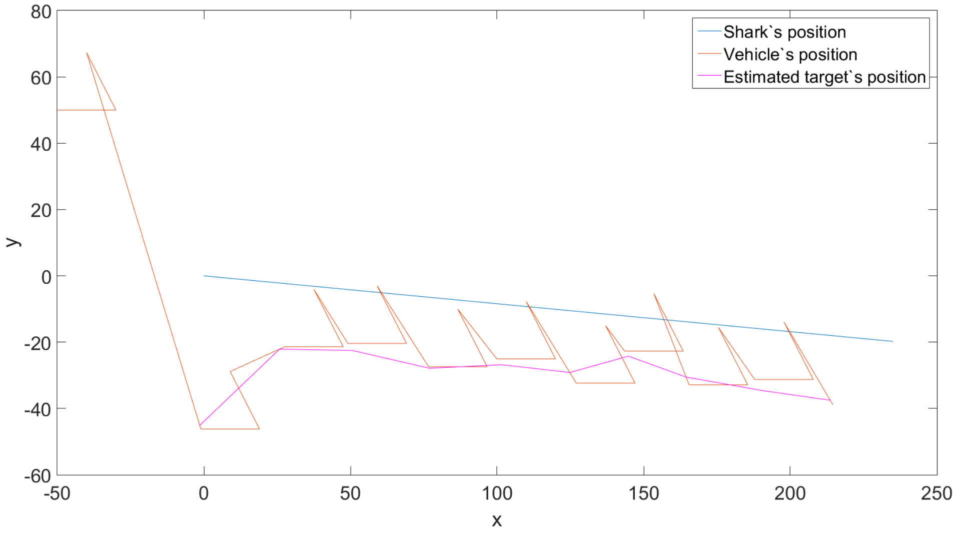


Figure 6.3: Tracking shark moving at 175 meters depth

Case 4.

Initial conditions:

- Shark's last known location (x,y) = (0,0)
- Shark's depth = 20 meters
- Vehicle's (x,y) coordinates = (-70,-60)

In the Figure 6.4, the UAV was deployed at a farther location, the farthest from all the runs until now. The mean positional error of 27.66 meters confirms that the initial position affects the initial shark estimation and therefore, the mean error.

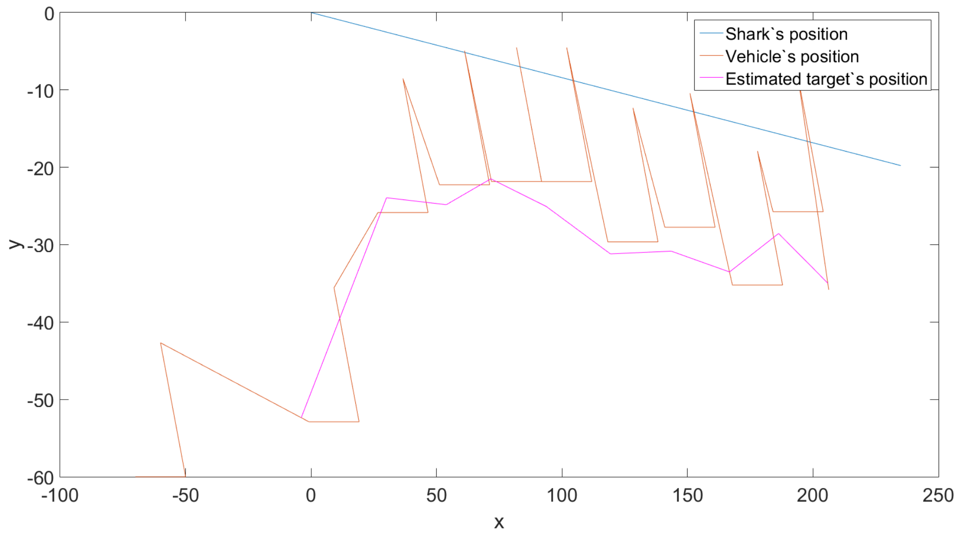


Figure 6.4: Tracking shark moving at 20 meters depth

Case 5.
Initial conditions:

- Shark’s last known location (x,y) = (-2435.5 kms,2389.1 kms)
- Shark’s depth = 169 meters
- Vehicle’s (x,y) coordinates = (-2435.5 kms,2389.2 kms)

In the Figure 6.5, a different shark is being tracked by a UAV. It is also possible to confirm that, the vehicle has successfully achieve its mission. The mean positional error was of 24.44 meters.

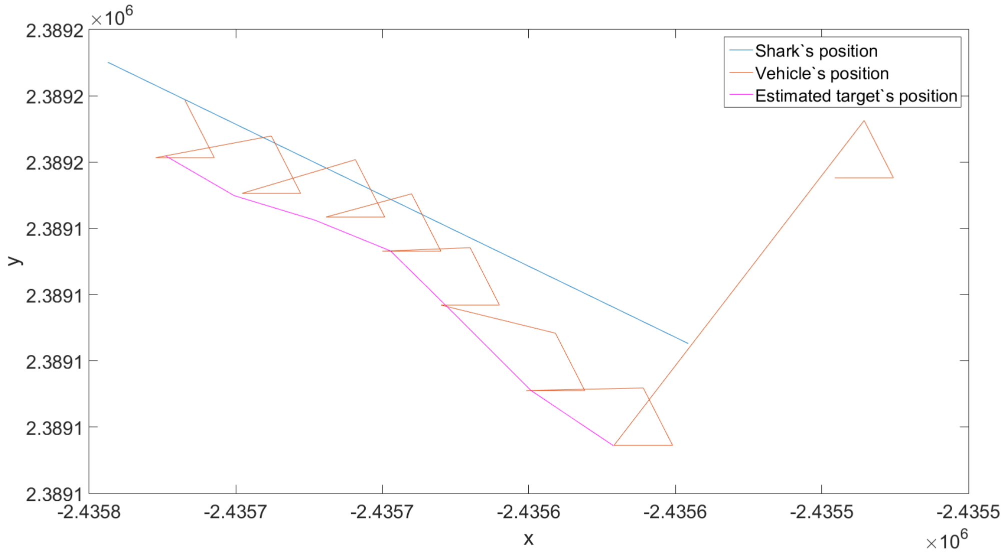


Figure 6.5: Tracking shark moving at 169 meters depth

6.3.2 Individual search

Case 1.

Initial conditions:

- Shark's last known location (x,y) = (0,0)
- Shark's depth = 400 meters
- Vehicle's (x,y) coordinates = (-900,-800)

In the Figure 6.6, the UAV is deployed approximately 1.2 kms away from the shark's last known location. Since the range of the acoustic signal is 500 meters, from this location it is impossible to receive pings so it must move closer. Note that, the first stop it made, it didn't receive any ping so it had to move to the second way point (last position of the UAV on the figure). Here we have successfully received a ping.

Recall that the red circle indicates the set of possible positions of the shark, at the surface of the sea, in respect to the UAV. Since the plot is in 2D, it doesn't exactly match any position of the shark since the shark is at 400 meters depth and not at the surface.

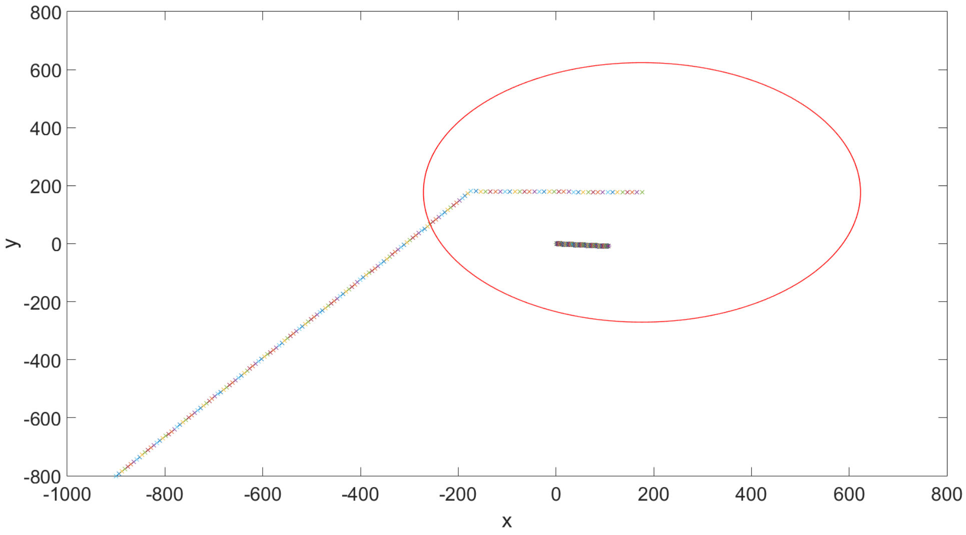


Figure 6.6: Searching for shark moving at 400 meters depth

Case 2.

Initial conditions:

- Shark's last known location (x,y) = (0,0)
- Shark's depth = 478 meters
- Vehicle's (x,y) coordinates = (-1000,1000)

In the Figure 6.7, the UAV is deployed at 1.4 kms from the shark's last known location. Observe how it failed to pin point its target, through the entire search strategy. The problem here is not the initial position of the vehicle but the depth of the shark. At 478 meters depth, the UAV only has a radius of $\approx$ 140 meters, from the shark's current (x,y) position, where it can land and listen to the acoustic signal. Since the vehicle's route is defined in order to cover a larger area around the shark's last known location, the way points can be placed outside the required radius.

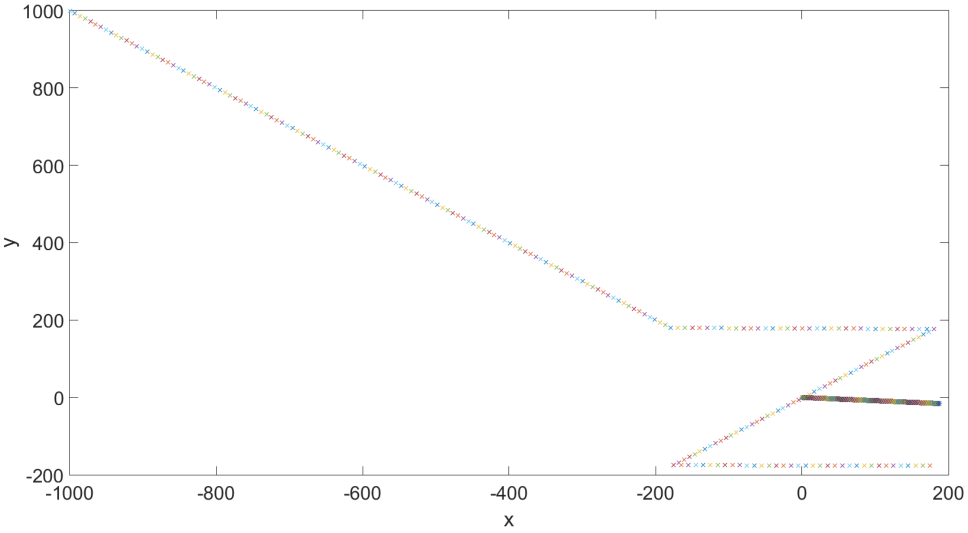


Figure 6.7: Searching for shark moving at 478 meters depth

Case 3.
Initial conditions:

- Shark's last known location (x,y) = (-2435.5 kms,2389.1 kms)
- Shark's depth = 0 meters
- Vehicle's (x,y) coordinates = (-2434.5 kms,2390.1 kms)

In the Figure 6.8, the strategy was tested for another shark. Here you can see that, since the shark is at the surface, the red circle made by the vehicle matches the real shark's position.

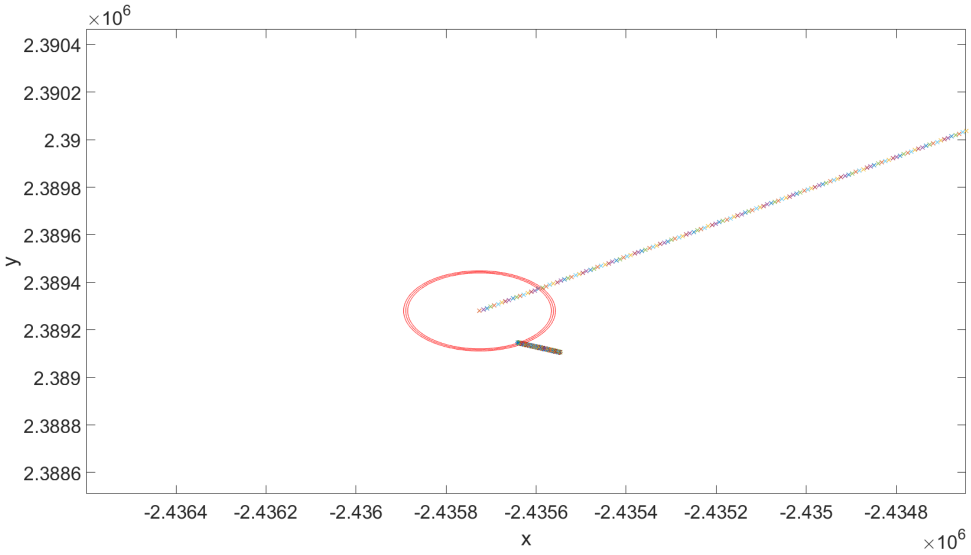


Figure 6.8: Searching for shark moving at 0 meters depth

6.3.3 Estimation with 2 UAVs

Case 1.

Initial conditions:

- Shark's last known location (x,y) = (0,0)
- Shark's depth = 356 meters
- Vehicle 1 (x,y) coordinates = (0,50)
- Vehicle 2 (x,y) coordinates = (-50,50)

In Figure 6.9, a system of two UAV is tracking a shark at 356 meters depth. Both of them start in the presence of the target's acoustic pings. Therefore, the two ping estimation strategy is used. At the very start, the yellow vehicle moves to the right, ignoring the new target's estimated value, because the shark's velocity was bigger than the limit of 8 m/s. Even though, this strategy could use a better filter in order to reduce the spikes in the estimated target's position (Figure 6.10), the system is still able to follow the shark's route. In this simulation run, the mean error was of 15.85 meters.

Additionally, as only one vehicle moves at a time, the other vehicle can recharge its battery while listening for the shark's signal. As a result, this strategy is less battery consuming than the individual tracking.

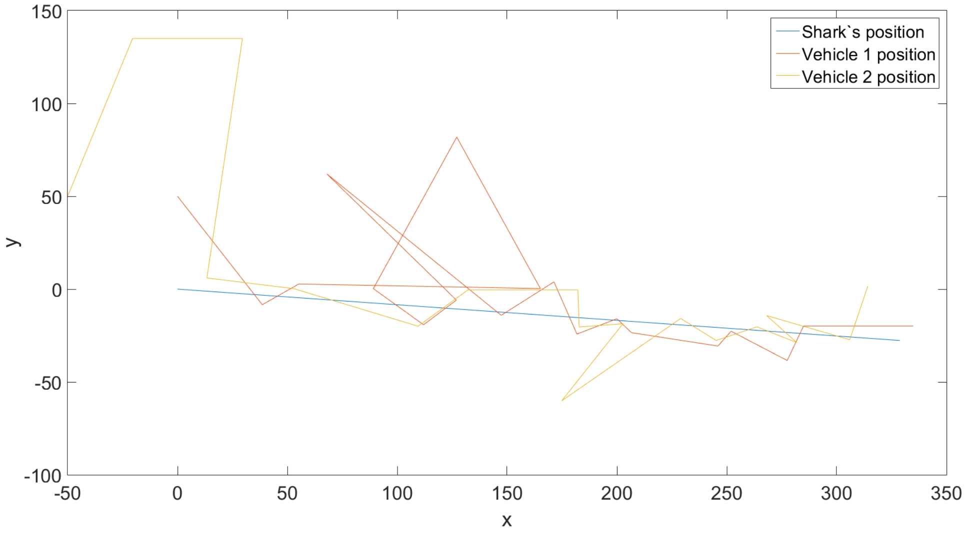


Figure 6.9: Tracking shark moving at 356 meters depth

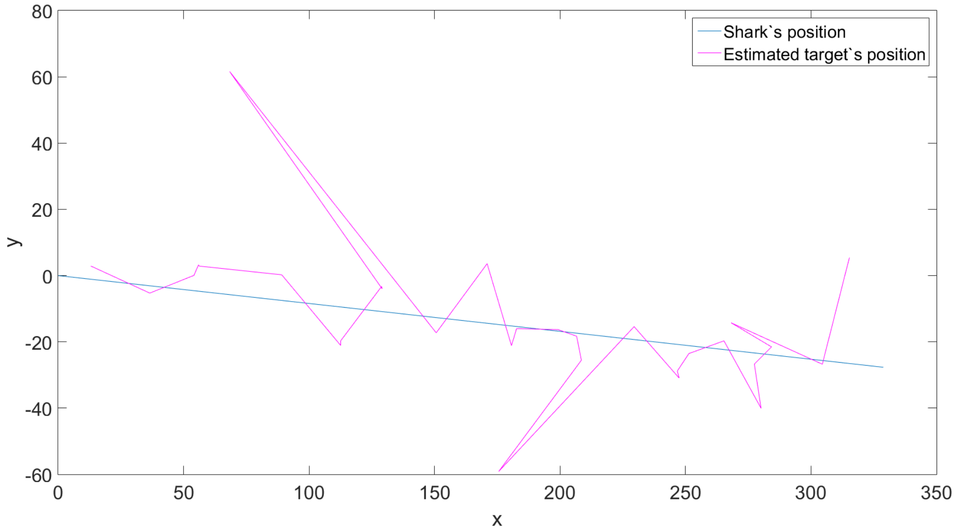


Figure 6.10: Comparison between estimated and real shark's position with 600 simulation seconds

Case 2.

Initial conditions:

- Shark's last known location (x,y) = (-2435.5 kms,2389.1 kms)
- Shark's depth = 163 meters
- Vehicle 1 (x,y) coordinates = (-2435.5 kms,2389.15 kms)
- Vehicle 2 (x,y) coordinates = (-2435.55 kms,2389.67 kms)

In Figure 6.11, another team of two UAV is tracking a different shark. In this case, only the orange UAV starts within signal range. Hence, cooperative one ping estimation strategy is used until both vehicles receive the signal. Luckily, after its first stop, the yellow UAV is able to hear the signal, starting the two pings estimation strategy. In this situation, the mean error was of 14.1 meters.

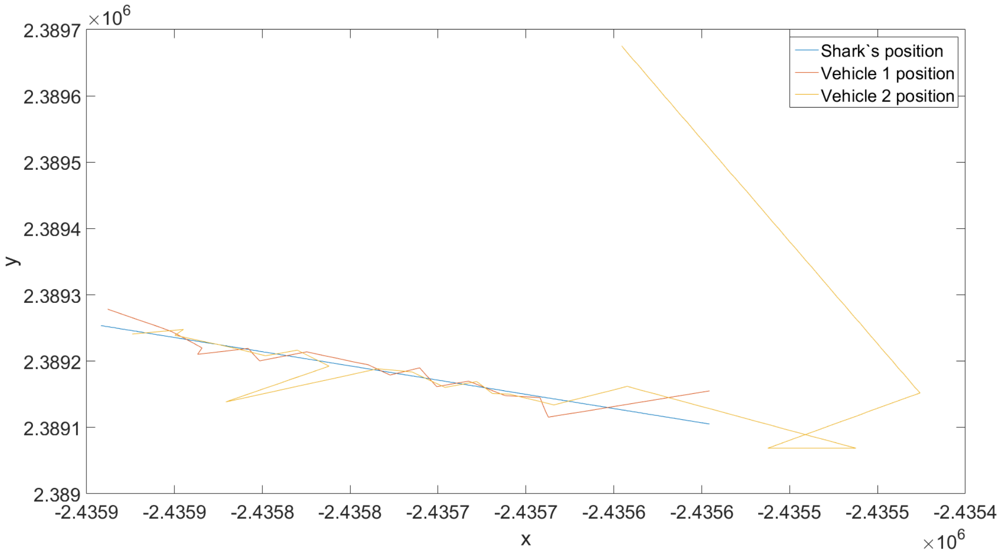


Figure 6.11: Searching for shark moving at 0 meters depth

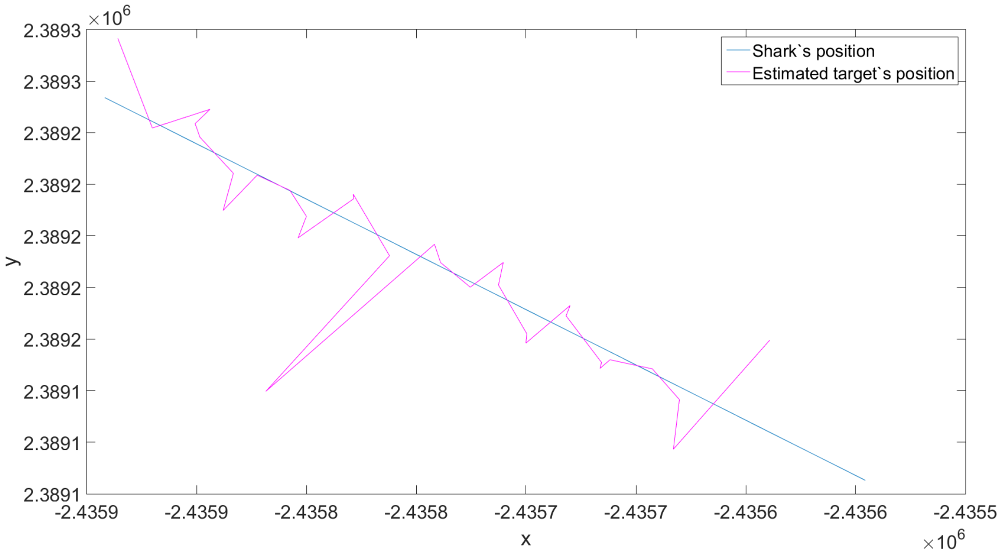


Figure 6.12: Comparison between estimated and real shark’s position with 600 simulation seconds

6.3.4 Estimation with 3/4 UAVs

Case 1. Initial conditions:

- Shark's last known location (x,y) = (0,0)
- Shark's depth = 402 meters
- Vehicle 1 (x,y) coordinates = (0,50)
- Vehicle 2 (x,y) coordinates = (400,50)
- Vehicle 3 (x,y) coordinates = (200,396)

Figure 6.13 illustrates the tracking of a shark by a team of three UAVs. Initially, only the orange UAV is close enough to hear the signal. Therefore, the team starts with a one ping estimation for three vehicles. After their first landing, the other two UAVs have managed to receive the target's signal. Recall that, the listening timer is computed in such a way that, if the UAVs land within the signal range, they will be able to, at least, hear the signal twice. By sharing their new information with the control station, the position and velocity of their target is determined. This allows the team to accurately follow the shark.

In this strategy, since the team only moves once the signal fades for every element in the team, the UAVs consume less energy. This is also affected by the depth of the shark. If the shark is closer to the surface, the signal will take longer to disappear than if it is at high depths, allowing the vehicles to wait longer before moving.

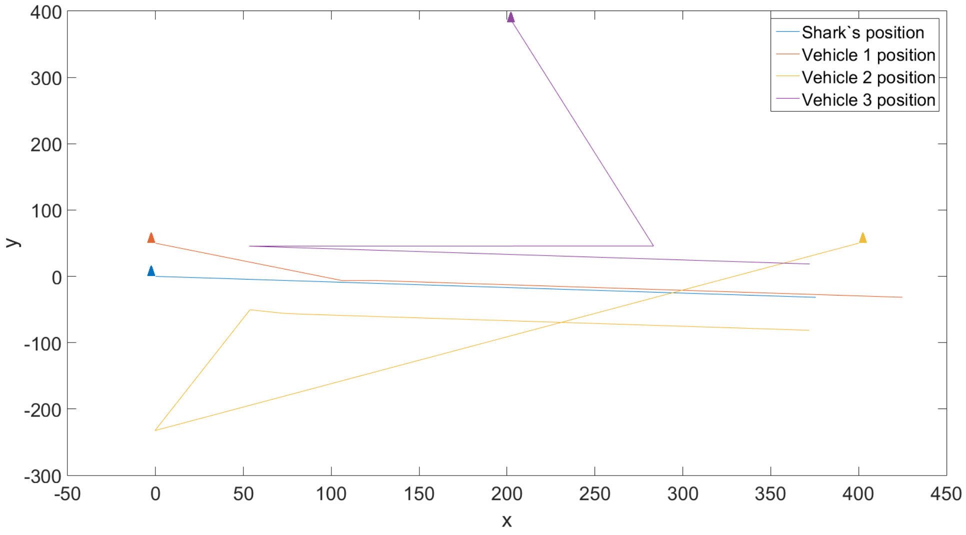


Figure 6.13: Tracking shark moving at 402 meters depth

Moreover, as it was already expected, the error in the estimate is almost null - ≈ 0.5 meters - which is only due to numerical calculations, Figure 6.14.

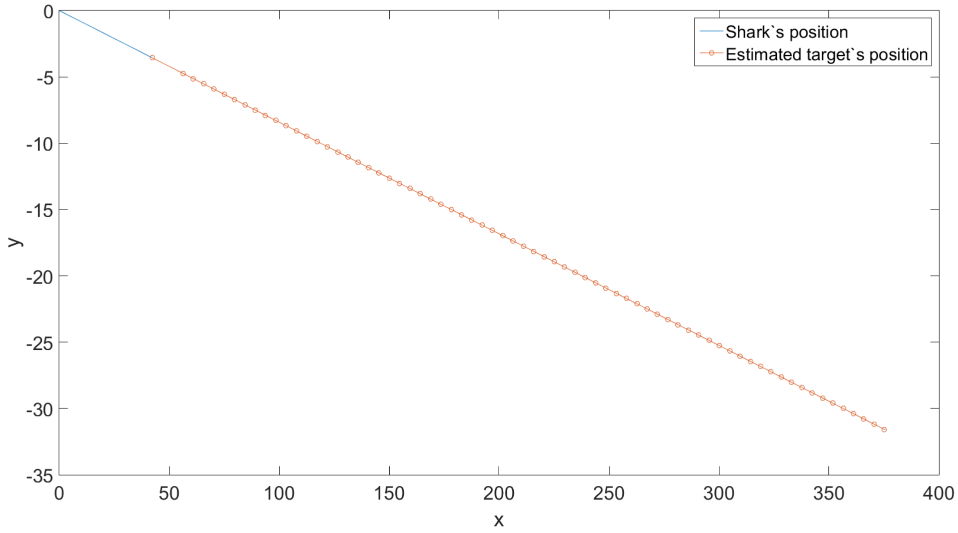


Figure 6.14: Comparison between estimated and real shark's position with 3 vehicles

Case 2. Initial conditions:

- Shark's last known location (x,y) = (-2435.5 kms,2389.1 kms)
- Shark's depth = 288 meters
- Vehicle 1 (x,y) coordinates = (-2435.5 kms,2389.15 kms)
- Vehicle 2 (x,y) coordinates = (-2435.55 kms,2389.15 kms)
- Vehicle 3 (x,y) coordinates = (-2435.55 kms,2389.8 kms)

The situation represented in Figure 6.15 is similar to the previous one. In this case, vehicle 1 and 2 start in the presence of the signal. For that reason, the team starts with a two ping estimation strategy. In this strategy, the UAV who has received the signal with highest intensity - vehicle 1 - stays still. Since the team has three elements, the other two have to move to a possible shark position, computed by the control station. Quickly after landing, they have changed their strategy to the three pings estimation strategy. Since the error is identical to the previous case, it will not be plotted.

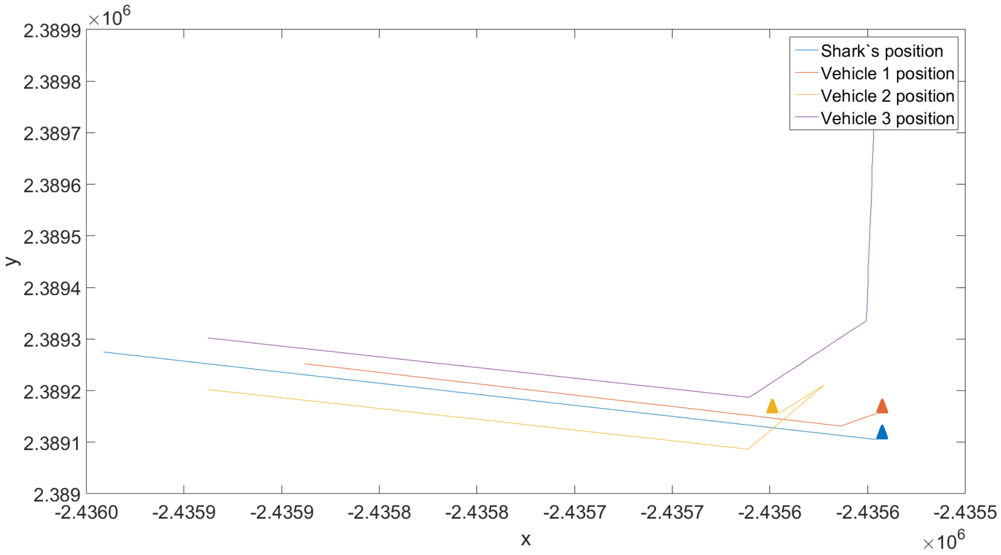


Figure 6.15: Tracking shark moving at 288 meters depth

Case 3. Initial conditions:

- Shark's last known location (x,y) = (-2435.5 kms,2389.1 kms)
- Shark's depth = 305 meters
- Vehicle 1 (x,y) coordinates = (-2435.5 kms,2389.15 kms)
- Vehicle 2 (x,y) coordinates = (-2434.9 kms,2389.15 kms)
- Vehicle 3 (x,y) coordinates = (-2435.5 kms,2389.3 kms)
- Vehicle 4 (x,y) coordinates = (-2434.9 kms,2389.3 kms)

In Figure 6.16, a team of four UAVs is tracking a shark at 305 meters depth. As in the first case of this section, only one UAV starts in the presence of the signal. Therefore, the other UAVs move closer to the first vehicle, landing in the first formation of Figure 5.3. After picking up the signal, the team forms a diamond around the estimated shark's position and waits for new way points and take off orders.

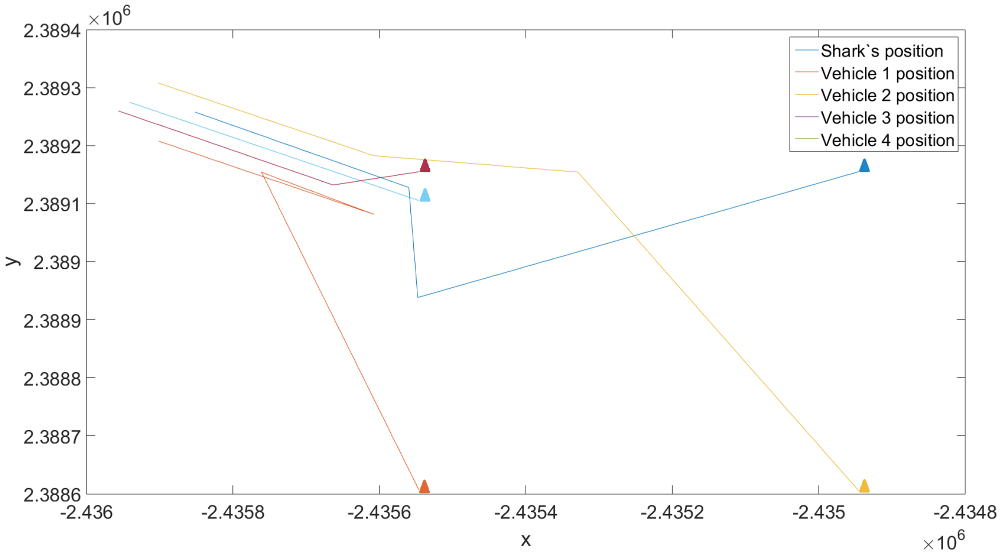


Figure 6.16: Tracking shark moving at 305 meters depth

6.3.5 Cooperative Search

Case 1.

Initial conditions:

- Shark's last known location (x,y) = (0,0)
- Shark's depth = 516 meters
- Vehicle 1 (x,y) coordinates = (1 km,1 kms)
- Vehicle 2 (x,y) coordinates = (0.8 km,1 kms)
- Vehicle 3 (x,y) coordinates = (0.9 km,1.73 kms)

In the Figure 6.17, a team of three UAVs is trying to find their target. The reason for failing here was clearly the shark's depth. With an acoustic range of 500 meters, it is impossible to receive pings from a device at depths bigger than this value.

Case 2.

Initial conditions:

- Shark's last known location (x,y) = (0,0)
- Shark's depth = 13 meters
- Vehicle 1 (x,y) coordinates = (1 km,1 kms)
- Vehicle 2 (x,y) coordinates = (0.8 km,1 kms)
- Vehicle 3 (x,y) coordinates = (0.9 km,1.73 kms)

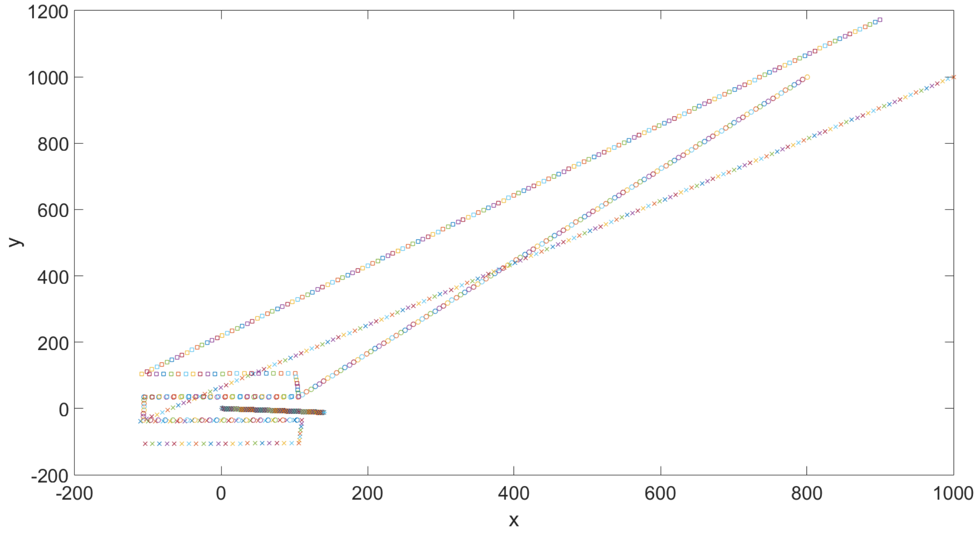


Figure 6.17: Searching for shark moving at 516 meters depth

In the Figure 6.18, the same team of three UAVs is trying to find their target. This time, the shark was very close to the surface, 13 meters depth, making it possible for the team to spot it right after they land for the first time. Note that the intersection of the surface pings almost leads to the read shark's position. This happens because the depth of the shark is very low.

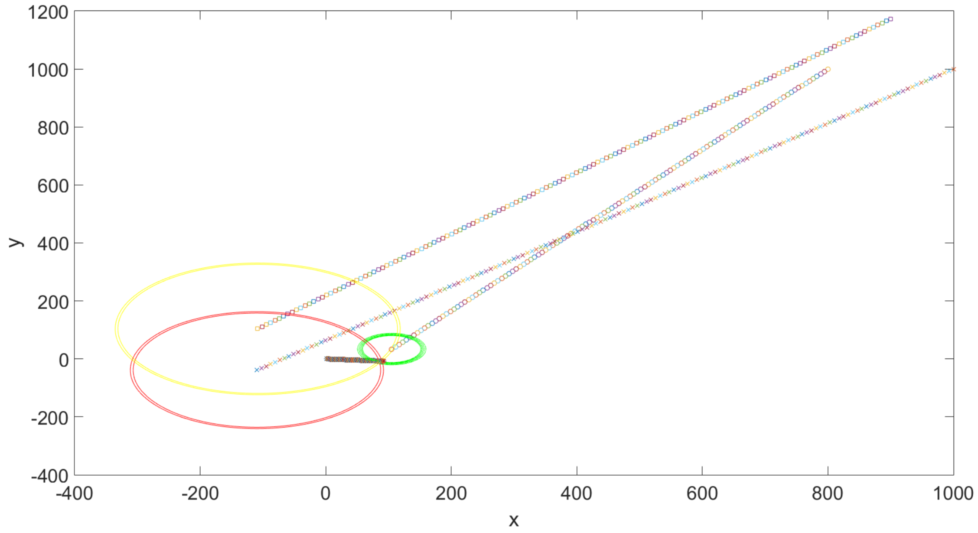


Figure 6.18: Searching for shark moving at 13 meters depth

Case 3.

Initial conditions:

- Shark's last known location (x,y) = (0,0)

- Shark's depth = 353 meters
- Vehicle 1 (x,y) coordinates = (-1 km,1 km)
- Vehicle 2 (x,y) coordinates = (-0.8 kms,1 km)

Finally, figure 6.19 represents a team of two UAVs searching for a target at 353 meters depth. They have both managed to received pings once they landed, for their first time, around the shark's last known location.

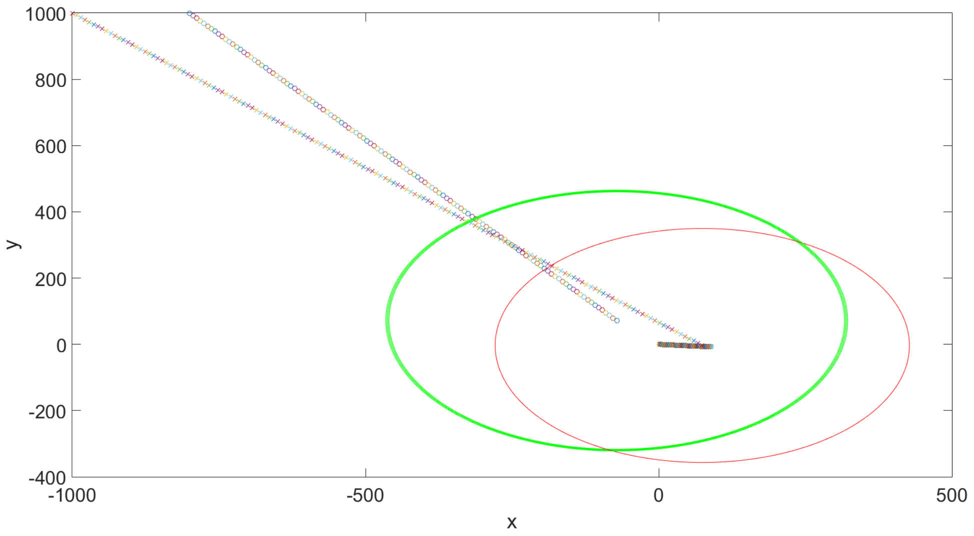


Figure 6.19: Searching for shark moving at 353 meters depth

6.3.6 Joining other UAV(s)

Case 1.

Initial conditions:

- Shark's last known location (x,y) = (-2435.5 kms,2389.1 kms)
- Shark's depth = 580 meters
- Vehicle 1 (x,y) coordinates = (-2435.5 kms,2389.15 kms)
- Vehicle 2 (x,y) coordinates = (-2435.3 kms,2389.15 kms)
- Vehicle 3 (x,y) coordinates = (-2435.4 kms,2389.323 kms)

Lets observe, how the system reacts to the entrance of new members. In Figure 6.20, two UAVs - purple and orange - start by searching for their target. At time 70, when they were at their second way point (represented by circles on their routes), the yellow UAV is assigned with the same target as theirs. Since the new vehicle is within the communication range of the team, it joins them on their mission. Then, the team increases its search area and the search algorithm restarts. Unfortunately, they still failed to pin point the target. Note that, since the shark was at 580 meters depth, they were doomed to fail.

The simulation was stopped once UAV 1 and 3 have failed to locate their target for the fourth time, forcing them to abort their mission.

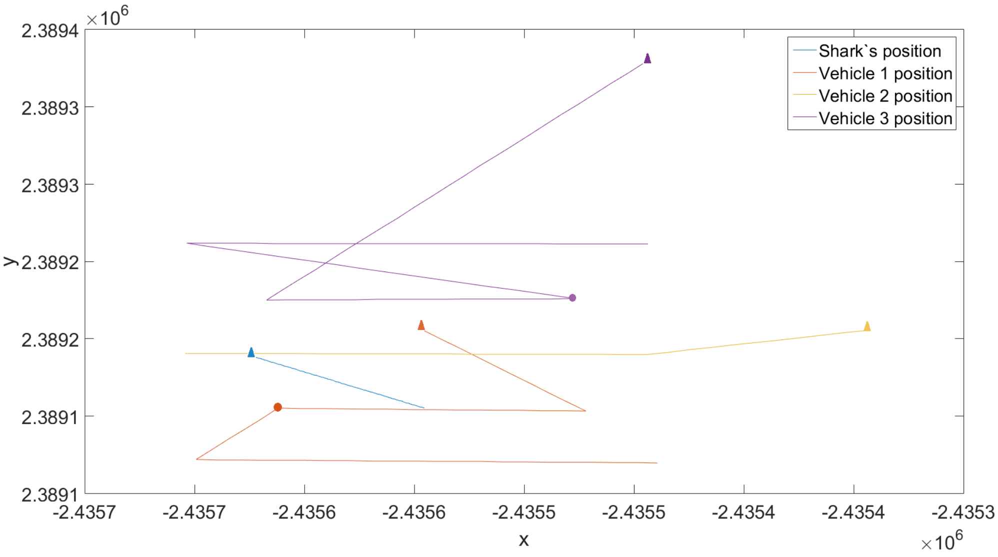


Figure 6.20: Searching for shark moving at 580 meters depth

Case 2.

Initial conditions:

- Shark's last known location (x,y) = (-2435.5 kms,2389.1 kms)
- Shark's depth = 50 meters
- Vehicle 1 (x,y) coordinates = (-2435.5 kms,2389.15 kms)
- Vehicle 2 (x,y) coordinates = (-2435.3 kms,2389.15 kms)
- Vehicle 3 (x,y) coordinates = (-2435.4 kms,2389.9 kms)

In Figure 6.21, a system of two UAVs is tracking a shark at 50 meters depth. It is possible to see that the team was using the two pings strategy to estimate the target's position. At time 220, another vehicle is introduced into the team. By forming a triangle around the estimated target's position, all the UAVs have managed to receive the ping. Then, the team adopted a three pings estimation strategy to proceed the mission.

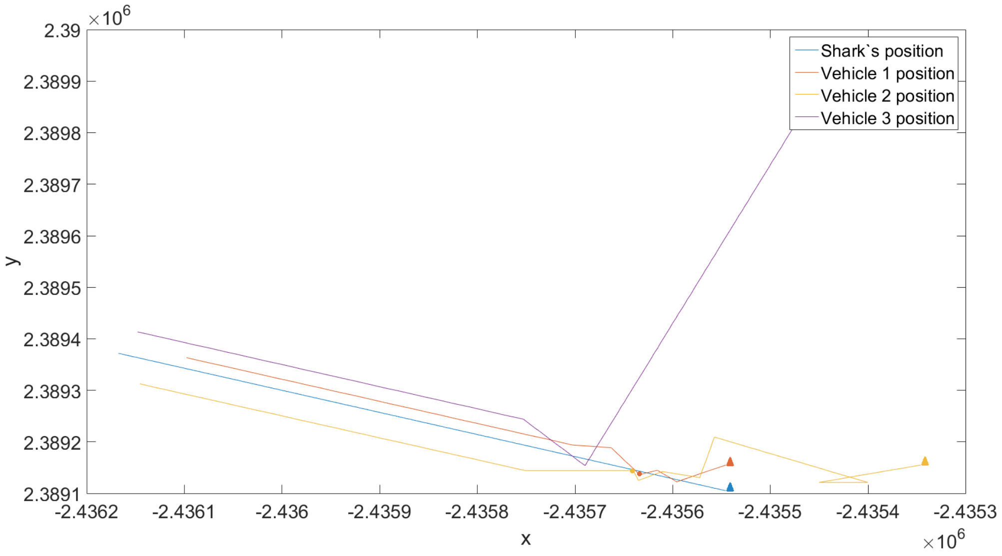


Figure 6.21: Tracking shark moving at 50 meters depth

6.3.7 Leaving team

Case 1.
Initial conditions:

- Shark's last known location (x,y) = (0,0)
- Shark's depth = 273 meters
- Vehicle 1 (x,y) coordinates = (0,50)
- Vehicle 2 (x,y) coordinates = (200,50)
- Vehicle 3 (x,y) coordinates = (100,223)

Initially, a team of UAVs is using the three pings estimation strategy to follow their targeted shark (Figure 6.22). At time 870, the operator decided that the purple UAV should abort its mission, forcing it to leave the team. With one less member, the team had to change their strategy to a two pings estimation strategy. It is easy to spot when did this happen since the UAVs changed from a straight line to a zigzagging movement.

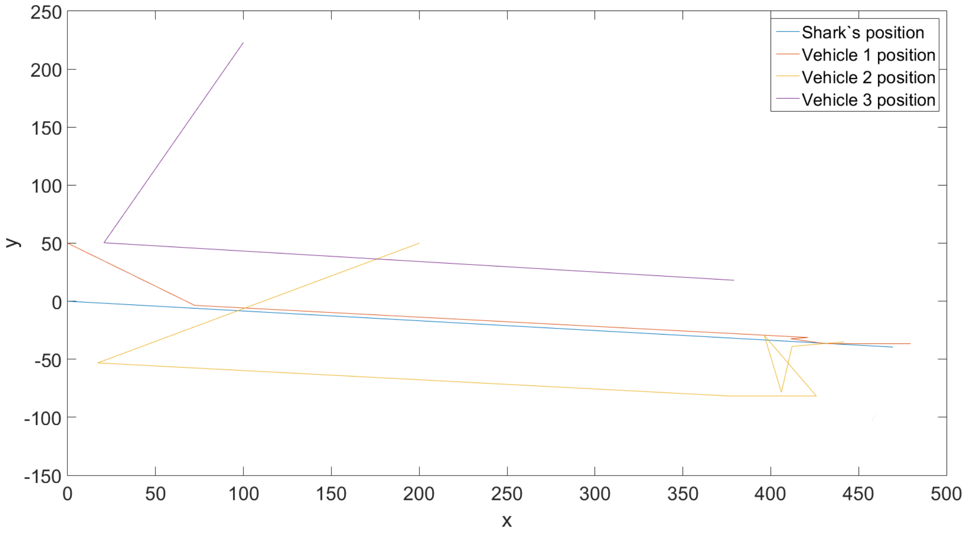


Figure 6.22: Tracking shark moving at 273 meters depth

Case 2.
Initial conditions:

- Shark's last known location (x,y) = (0,0)
- Shark's depth = 87 meters
- Vehicle 1 (x,y) coordinates = (0,50)

- Vehicle 2 (x,y) coordinates = (200,50)

What happens when an UAV leaves a team, composed by only two elements? This situation is represented in Figure 6.23, where a team of two UAVs is tracking a shark at 87 meters depth. At time 230, the operator aborted the mission of the orange vehicle. When the yellow UAV finds out that the team was disbanded, it quickly changes its strategy to be able to track its target all by itself - triangular movement.

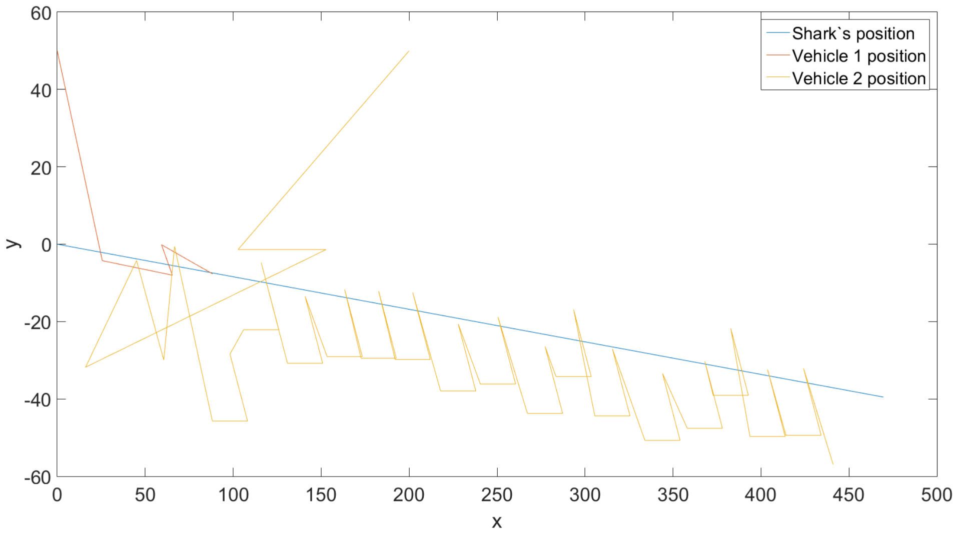


Figure 6.23: Tracking shark moving at 87 meters depth

6.3.8 Teams and multiple targets

This test was done in order to prove that multiple teams can track different targets at the exact same time. Since all the strategies have already been covered, there is no point in showing more than one example.

In this example, we have two teams tracking two different sharks. On the simulation they start within the range of the acoustic signal so we will only observe their estimation strategy.

Team 1 - Initial conditions:

- Shark’s last known location (x,y) = (0,0)
- Shark’s depth = 211 meters
- Vehicle 1 (x,y) coordinates = (50,50)
- Vehicle 2 (x,y) coordinates = (-50,50)
- Vehicle 3 (x,y) coordinates = (0,50)
- Vehicle 4 (x,y) coordinates = (0,-50)

Team 2 - Initial conditions:

- Shark's last known location (x,y) = (-2435.5 kms,2389.1 kms)
- Shark's depth = 13 meters
- Vehicle 1 (x,y) coordinates = (-2435.5 kms kms,2389.2 kms)
- Vehicle 2 (x,y) coordinates = (-2435.6 kms kms,2389.2 kms)
- Vehicle 3 (x,y) coordinates = (-2435.55 kms kms,2389.7 kms)
- Vehicle 4 (x,y) coordinates = (-2435.55 kms kms,2388.8 kms)

In Figure 6.24, we can see two dots: one on the northwest corner and the other near the southeast corner. They represent the activity of the two teams. Since the sharks are too far from each other, we can't figure out what is happening with each team without zooming in.

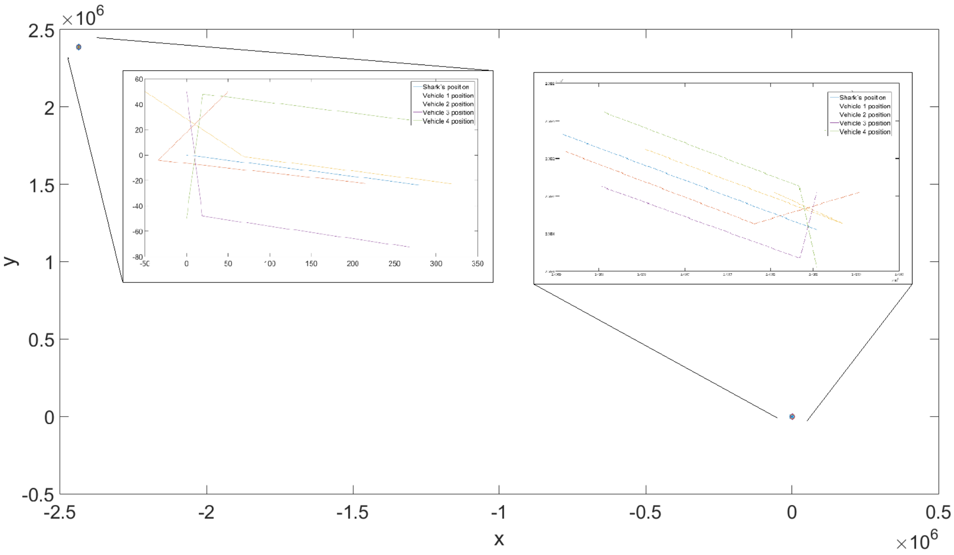


Figure 6.24: Simultaneous tracking

Figures 6.25 and 6.26 show, respectively, team 1 and 2 tracking their targets with the three pings estimation strategy. Recall that, even though they are four, the fourth measurement is not required to estimate the shark's position. Basically, the fourth element is there just to provide larger surveillance area and eliminate blind spots.

From this test run, we can see that even though they have an asset in common - the central control - the teams are still able to work simultaneously. Note that, the real system would probably require a collision avoidance protocol for the communications between the UAVs and the central control.

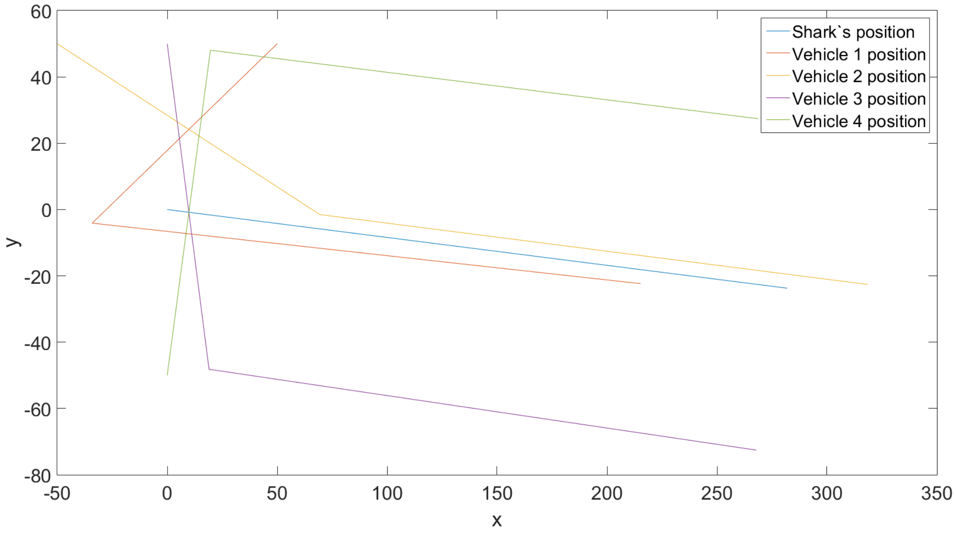


Figure 6.25: Team 1 tracking shark moving at 211 meters depth

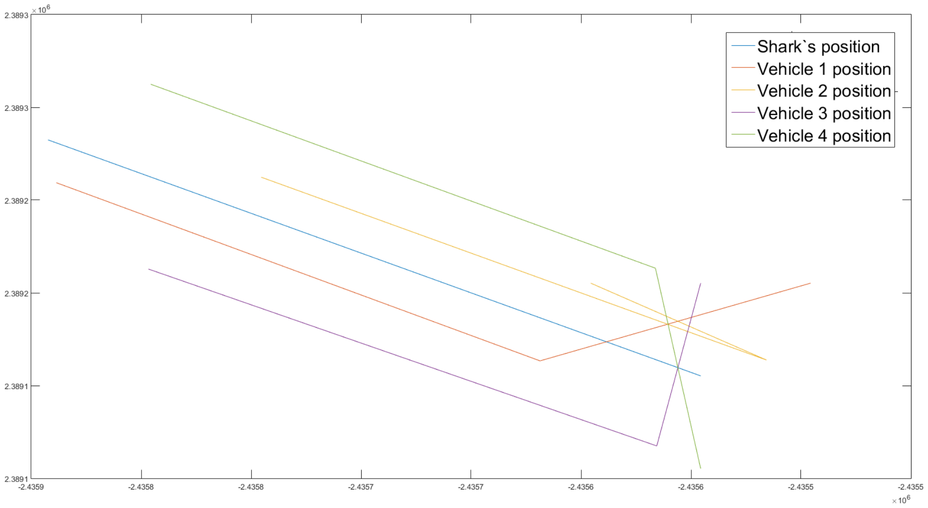


Figure 6.26: Team 2 tracking shark at 13 meters depth

6.3.9 Long-run

It would be interesting to see how the system behaves when tracking the same target for several days in a row. In missions like this, the system has to deal with multiple adversities such as sudden changes in the shark's direction, velocity and depth, reaching places outside of the central control's communication range, UAVs running low on battery, among others. However, in this thesis we will only be covering the changes in the shark's behaviour.

In Figure 6.27, a stand-alone UAV attempted to track a shark for several days. Unfortunately, after 48h of continuous tracking the signal disappeared. This situation is represented in Figure 6.29, where the UAV tried the search strategy to find its target. Unsuccessfully, once it reached the searching timeout, it aborted its mission.

For a closer look of the UAV tracking strategy check Figure 6.28.

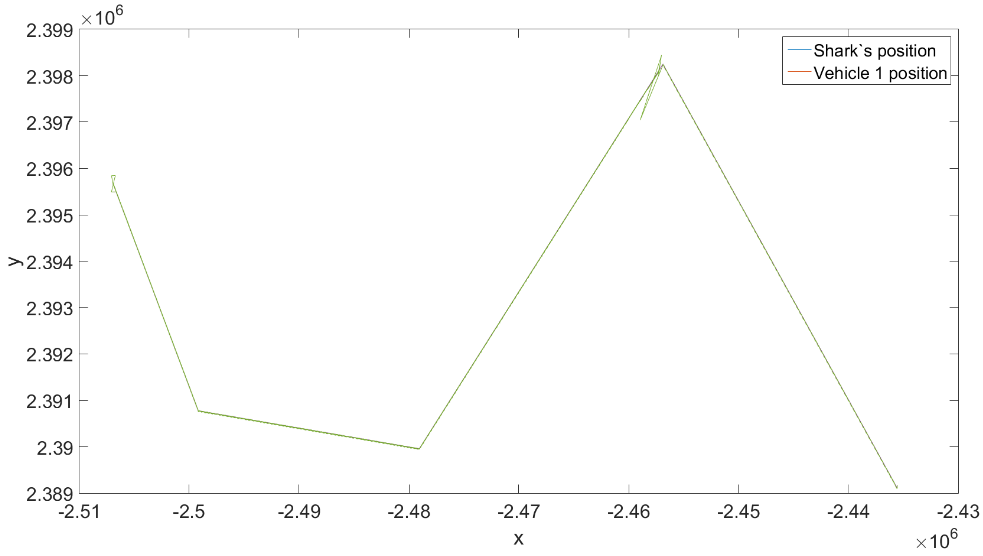


Figure 6.27: Stand-alone UAV tracking shark for 48h

Another interesting aspect is how the UAV reacted to the first change on the shark's direction and depth. Actually, it lost the target for a few seconds. Probably, this happened because it mixed data from the two different shark movements. However, since the UAV is faster than the shark, once it found out it had lost the signal, it started searching and picked up the signal right away.

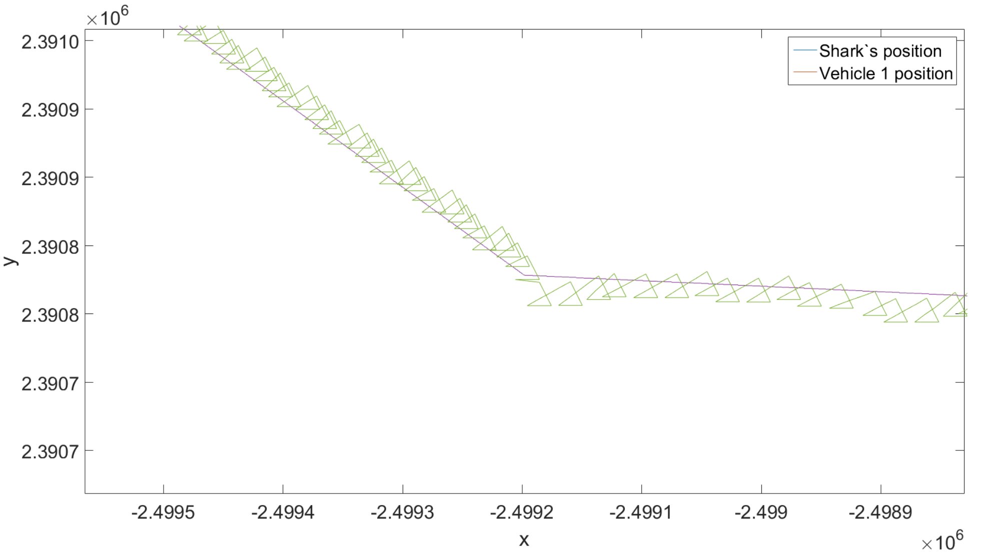


Figure 6.28: UAV tracking strategy

Note that, in a real life situation, the UAV wouldn't be able to work for 48h straight. Eventually, it would run out of battery. However, it is still interesting to see that, from the point of view of the tracking and searching strategies, the vehicle can achieve outstanding results.

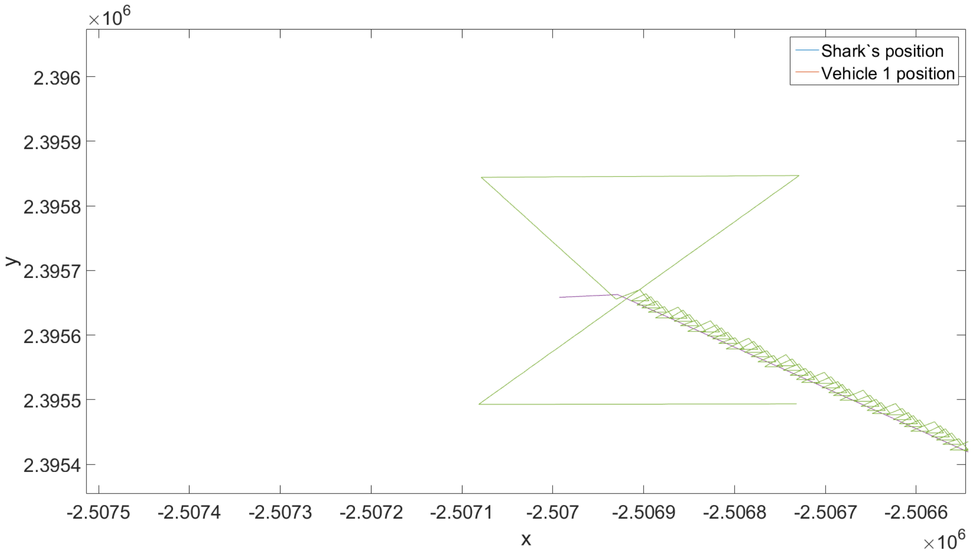


Figure 6.29: UAV searching for shark

Due to the computational effort of this type of simulations, I will not be able to show how a long-run for just two vehicles would look like. Recall that, a system composed of one or two UAVs has to actively track the target. Since their search and tracking strategies were made to respect small time steps (order of seconds), increasing the simulation step in order to reduce the number of iterations, won't allow them to preform as they should. However, these strategies for a stand-alone vehicle are less complex than for a system of two UAVs. Note that, for the first case, there are no communications among UAVs or with the central control.

For teams with more than three elements, since they only move when the signal fades for all the elements, it is possible to increase the time step without changing the behaviour of the system. In Figure 6.30, a team is tracking a shark for several days. It is possible to see that the team reacts quickly to steep changes in the shark's movement (Figure 6.31).

The simulation was stopped at the start of the eight day. For the entire simulation run, the team was able to keep up with its target.

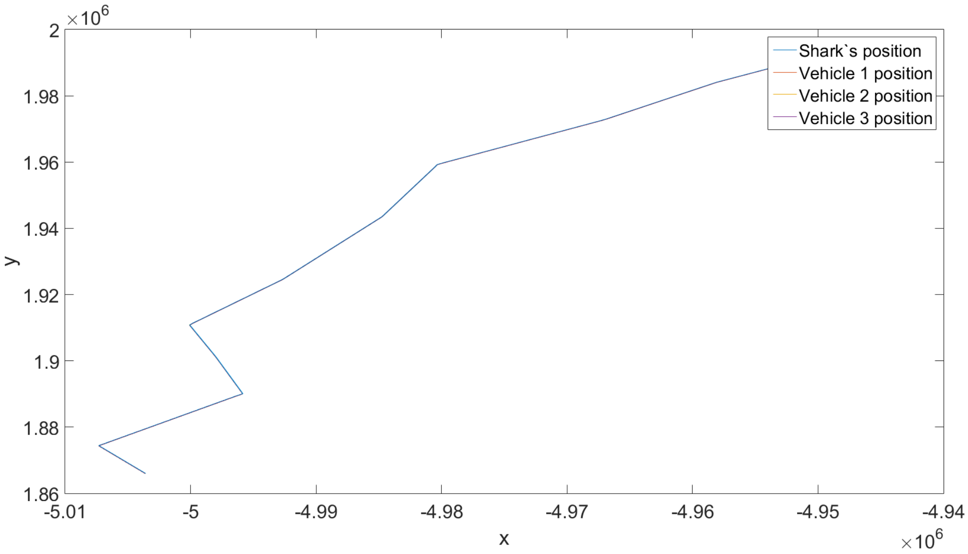


Figure 6.30: Team tracking shark for seven days

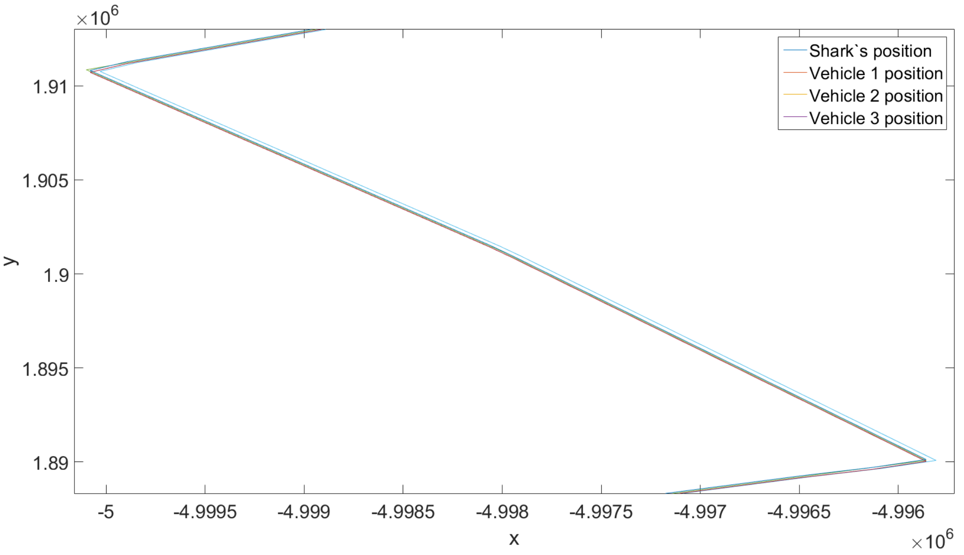


Figure 6.31: Team reacting quickly to big changes in the shark’s direction

6.4 Conclusions

The system can run with stand-alone UAVs or teams of cooperative UAVs. From the test plan, the 1^{st} and 2^{nd} test only respect the first type of systems while the 3^{rd}, 4^{th}, 5^{th}, 7^{th}, 8^{th} only consider cooperative systems. The rest of the tests evaluate the performance of both system types.

We will start by presenting the conclusions regarding stand-alone missions and then regarding cooperative missions. There are two types of strategies explored in these simulation runs: estimation and tracking strategies and searching/prediction strategies. Due to the diversity of the first type of strategies most of the conclusions concern the estimation and tracking strategies. Therefore, we will first present the conclusions for each estimation strategy and then we move to more global conclusions.

Stand-alone missions

- Continuous movement when tracking or estimating;
- Mean error of $\approx$ 22 meters;
- The mean error is not affected by the shark's depth but by the frequency of the acoustic signal;
- The farthest the vehicle is from the target when it starts estimating, the larger the error will be;
- Even if a wrong target's position estimate is made, due to changes in the depth, direction or speed of the shark, the UAV is fast enough to pick back the signal with the search algorithm.

Estimating and tracking with only two pings

- When tracking or estimating, only one UAV moves at a time, allowing the other UAV to recharge its battery while waiting;
- Mean error of $\approx$ 16 meters;
- Bigger spikes (punctual error) in the shark's estimated position;
- Due to these spikes, the movement of the UAVs tends to be less smoothly.

Estimating and tracking with three or four pings

- All UAVs move at the same time;
- Vehicles only move when the signal starts fading for the last vehicle in the team;
- More time to recharge batteries since they stop for longer periods of time;
- The closer the shark is to the surface the longer it will take for the signal to fade;
- Error is almost null and only exists due to numerical calculations.

Global

- All the algorithms fail when the shark is not within the acoustic device range (shark dived deeper than the maximum range of the device). Additionally, the search algorithms can also fail if the shark has moved to far from the area created around the shark's last known location;
- Increasing the number of team elements up to 3, improves the estimation (more accurate target's position) and reduces the number of times the vehicles have to move during the same time period;
- Teams respond quickly to changes in the number of vehicles;
- Multiple teams can track multiple targets at the same time, while sharing common assets;
- For long missions, teams of three UAVs have the best performance since they can outlast (in terms of battery) the other types of systems and have the most accurate target's estimate.

Chapter 7

Conclusions and Future Work

In this thesis, a control architecture for cooperative tracking has been developed. The goal was to persistently track ocean sharks with UAVs. With nothing more than the intensity of the acoustic signal, systems composed by teams and/or stand-alone UAVs estimate and track tagged targets for long periods of time. Even though the selected acoustic tags do not transmit any information regarding the shark's depth, the estimation and tracking strategies designed allow the system to track a tagged shark at any depth. We should also point out, that unlike systems composed by AUVs, our system of aerial vehicles can track fast fish capable of reaching 8m/s, the top speed of a diving shark.

Although the strategies developed are simple, the global performance (in simulation) shows potential for future field operations. However, there is room for improvement. Note that UAVs can't dive in the water so if the shark dives deeper than the maximum range of the acoustic device, they are bound to lose the signal. For this reason, the success rate of this type of systems increases with the increment of the maximum range of the acoustic signal.

At the beginning of this thesis, we have presented the research questions that should be addressed by this work. Therefore, we will continue this conclusion by answering these questions.

Can we estimate the position of the fish based on real observations? Certainly we can. The distance to the target recorded on these observations is used to define the set of all the possible target positions. By combining data from multiple UAVs, the number of possible positions can be reduced. In fact, with three UAVs deployed in the triangular squad formation, described in Chapter 5, there will always be only one feasible target position.

Can we improve the quality of this estimation with multiple vehicles? As it was possible to see in the previous chapter, the mean error of the target's estimate in respect to the real target's position is inversely proportional to the number of vehicles. The mean error of the estimate was $\approx$ 22 meters, $\approx$ 16 meters and 0.5 meters for systems with stand-alone UAVs, teams of two and teams with at least three vehicles, respectively. Teams with more than three UAVs are a waste of resources since, with the defined triangular formation, only three simultaneous pings are required to precisely estimate the position of the target.

Can we improve the tracking strategy by using multiple UAVs? Yes. Tracking with three vehicles was proven to be the best. Since they don't have to actively estimate the position of their target, they can stay longer on the same position, reducing their battery consumption. Recall that, while they wait for orders they continue to record observations. By sharing the new data with the central control when the signal starts to fade, the central control updates the information in order to accurately compute the velocity and direction of the target, making the team respond faster to quick changes in the shark's route.

Even though a fourth UAV in a team is not required for estimating and tracking tagged sharks, it can still be useful when searching for it, since the search area increases with the number of UAVs. Therefore, it might not be a bad idea to use teams of four elements to locate the target and, once it is found, reduce the number of elements to three.

From all the works published until now discussed in Chapter 4, this is the first time UAVs are used to tackle the fish tracking problem. Actually, this innovative system addresses some of the limitations of the systems composed by AUVs. These aerial vehicles are much faster than sharks so they can keep up with it even if the sharks go on burst swims. Additionally, unlike related works, our teams of three UAVs don't have to continuously move in order to successfully track the target, making it possible to withstand longer missions. The key limitation of this system is the maximum depth a shark can reach without leaving the area defined by the maximum range of the acoustic device. Once the shark leaves that area the UAVs will not be able to receive the acoustic signal. Unlike the AUVs that can dive in the water, making the maximum shark's depth adjustable, the UAVs have to stay at the sea surface.

Having demonstrated the feasibility of using cooperative UAVs to track tagged sharks in the ocean, the next step must be actual ocean testing of this code. This can be done first with AUV carrying tags to mimic shark behaviour. A similar experiment was already conducted by LSTS and NTNU, in Norway. After doing this, it will be possible to move to real sharks.

Nevertheless, this work was focused on designing a sound architecture within which controllers and estimators are deployed. The focus was on modularity and soundness modularity to allow substitution by other controllers and estimators. Sounded to ensured that the whole system is sound thus accommodating improvements and extensions. These can be the subject of future work. Additionally, the same techniques from the verified control architectures, described in Chapter 4, should be used to prove some of the properties of the system.

Regarding the developed estimation strategies, the two pings strategy is by far the one that requires more attention. The movement of the vehicles is not as smoothly as it should due to the spikes in the target's estimate. This could be improved by considering the propagation time of the acoustic signal to reduce the number of possible target locations. Additionally, implementing a particle filter can also help avoiding false measurements.

Another limitation of the systems developed until now is the maximum duration of the field operations the vehicles can withstand without having to stop because of the battery level. Our approach does not solve this problem. Actually the control and estimation design should be improved by using knowledge on the battery level of each UAV in order to optimise the battery consumption of the UAVs. For example, in the two pings estimation strategy, instead of being the UAV farthest from the target to actively estimate the target's position, it could be the UAV which has a higher battery level.

A further area of investigation would be using different types of autonomous vehicles to track fish. Probably a mixed system of ASVs, AUVs and UAVs could cover some of the limitations expressed above. For example, it could increase the maximum depth at which the shark can reach without losing track of it.

Our developments can be used to track not only several ocean species but human made vessels such as AUVs and submarines, making it useful for scientific and military purposes.

References

[1] João Hespanha. Hybrid and switched systems. URL: https://web.fe.up.pt/~jtasso/HS10.htm [last accessed 10.06.2017].

[2] N. QUEIROZ et al. CIBIO – Universidade do Porto. Understanding shark movements and behaviour in relation to environment from satellite telemetry: Where and why?

[3] E. Farris M. Moline M.A. Lowe C.G. Clark C.M. Forney, C. Manii. Tracking of a tagged leopard shark with an auv: Sensor calibration and state estimation. 2012.

[4] Tom Austin Mark Dennett Robin Littlefield Gregory Packard Mike Purcell Roger Stokey Gwyneth E. Packard, Amy Kukulya. Continuous autonomous tracking and imaging of white sharks and basking sharks using a remus-100 auv. 2012.

[5] D. Shinzaki, C. Gage, S. Tang, M. Moline, B. Wolfe, C. G. Lowe, and C. Clark. A multi-auv system for cooperative tracking and following of leopard sharks. In *Proceedings - IEEE International Conference on Robotics and Automation*, pages 4153–4158, 2013.

[6] Taylor Peterson Connor White Christopher G. Lowe Yukun Lin, Hannah Kastein and Christopher M. Clark. A multi-auv state estimator for determining the 3d position of tagged fish. 2014.

[7] National Oceanic and Atmospheric Administration. Ocean exploration. URL: http://oceanexplorer.noaa.gov/facts/unexplored.html [last accessed 09.07.2017].

[8] LSTS. Sunfish tracking. URL: http://sunfish.lsts.pt/en [last accessed 09.07.2017].

[9] N. Queiroz, N. E. Humphries, G. Mucientes, N. Hammerschlag, F. P. Lima, K. L. Scales, P. I. Miller, L. L. Sousa, R. Seabra, and D. W. Sims. Ocean-wide tracking of pelagic sharks reveals extent of overlap with longline fishing hotspots. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 113, pages 1582–1587, 2016.

[10] M. R. Heithaus, L. M. Dill, G. J. Marshall, and B. Buhleier. Habitat use and foraging behavior of tiger sharks (galeocerdo cuvier) in a seagrass ecosystem. *Marine Biology*, 140(2):237–248, 2002.

[11] C. G. Meyer, T. B. Clark, Y. P. Papastamatiou, N. M. Whitney, and K. N. Holland. Long-term movement patterns of tiger sharks galeocerdo cuvier in hawaii. *Marine Ecology Progress Series*, 381:223–235, 2009.

[12] A. L. Kukulya, R. Stokey, C. Fiester, E. M. H. Padilla, and G. Skomal. Multi-vehicle autonomous tracking and filming of white sharks carcharodon carcharias. In *Autonomous Underwater Vehicles 2016, AUV 2016*, pages 423–430.

[13] J. Merril Valerdi, R. and P. Maloney. Cost metrics for unmanned aerial vehicles. In *16th Ligther-than-air Systems Technology Conference and Robotic Balloon Systems Conference. American Institute of Aeronautics and Astronautics*, 2005.

[14] J. B. De Sousa, K. H. Johansson, J. Silva, and A. Speranzon. A verified hierarchical control architecture for co-ordinated multi-vehicle operations. *International Journal of Adaptive Control and Signal Processing*, 21(2-3):159–188, 2007.

[15] J. Z. Sasiadek e P. Hartana. Sensor data fusion using kalman filter. In *Proceedings of the Third International Conference on Information Fusion*, 2000.

[16] e Gerhard Lakemeyer Alexander Ferrein, Lutz Hermanns. Comparing sensor fusion techniques for ball position estimation. In *Robot Soccer World Cup*, pages 154–165, 2005.

[17] Aamir Ahmad and Pedro U. Lima. Multi-robot cooperative object tracking based on particle filters. In *Robotics and Autonomous Systems*, pages 37–42, 2011.

[18] Dr. Nuno Miguel Cabral Queiroz. Personal page. URL: `https://cibio.up.pt/people/details/ncqueiro` [last accessed 11.10.2016].

[19] Robin Littlefield Frederic Jaffre Dr. Edgar Mauricio Hoyos Padilla Dr. Gregory Skomal Amy L. Kukulya, Roger Stokey. 3d real-time tracking, following and imaging of white sharks with an autonomous underwater vehicle. 2013.

[20] J. Orwell M. Xu and G. Jones. Tracking football players with multiple cameras. In *Proc. Int'l Conf. Image Processing*, pages 2909–2912, 2004.

[21] W. Du and J. Piater. Multi-camera people tracking by collaborative particle filters and principal axis-based integration. In *Proc. Asian Conf. Computer Vision*, pages 365–374, 2007.

[22] M. Taj E. Maggio and A. Cavallaro. Efficient multi-target visual tracking using random finite sets. In *IEEE Trans. Circuits and Systems for Video Technology*, pages 1016–1027, 2008.

[23] A. Hoogs G. Brooksby A. Perera, C. Srinivas and H. Wensheng. Multi-object tracking through simultaneous long occlusions and split-merge conditions. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2006.

[24] B. Frühstück C. Beleznai and H. Bischof. Multiple object tracking using local pca. In *Proc. Int'l Conf. Image Processing*, 2006.

[25] R. Lengagne F. Fleuret, J. Berclaz and P. Fua. Multi-camera people tracking with a probabilistic occupancy map. In *IEEE Trans. Pattern Analysis and Machine Intelligence*, pages 267–282, 2008.

[26] P.P.A. Storms and F.C.R. Spieksma. An lp-based algorithm for the data association problem in multitarget tracking. In *Computers and Operations Research*, pages 1067–1085, 2003.

[27] Hot topic - tracking great white sharks. URL: `http://www.dosits.org/hottopic/hotacoustictrackinggwsharks/` [last accessed 05.03.2017].

[28] Kim Fulton-Bennett. New mbari scientist uses sound to study ocean predators and their prey, 2016. URL: `goo.gl/S1kcJ5` [last accessed 05.03.2017].

[29] Ron Instrella. Midar -active multispectral imaging, 2016. URL: https://www.nasa.gov/ames/las/midar-active-multispectral-imaging/ [last accessed 03.03.2017].

[30] JASON KURTIS. Tracking great white sharks off cape cod by land, by air, by sea, 2016. URL: http://abcnews.go.com/US/tracking-great-white-sharks-off-cape-cod-land/story?id=41483731 [last accessed 03.03.2017].

[31] LSTS. Toolchain. URL: http://www.lsts.pt/toolchain [last accessed 25.02.2017].

[32] MATLAB. Documentation. URL: http://nl.mathworks.com/help/ [last accessed 05.06.2017].

[33] Aurora Flight Sciences. Aurora's centaur locates great white sharks off nantucket coast, 2016. URL: goo.gl/74QWhA [last accessed 05.03.2017].

[34] L. L. Sousa, F. López-Castejón, J. Gilabert, P. Relvas, A. Couto, N. Queiroz, R. Caldas, P. S. Dias, H. Dias, M. Faria, F. Ferreira, A. S. Ferreira, J. Fortuna, R. J. Gomes, B. Loureiro, R. Martins, L. Madureira, J. Neiva, M. Oliveira, J. Pereira, J. Pinto, F. Py, H. Queirós, D. Silva, P. B. Sujit, A. Zolich, T. A. Johansen, J. B. De Sousa, and K. Rajan. Integrated monitoring of mola mola behaviour in space and time. *PLoS ONE*, 11(8), 2016.

[35] Gilbert Strang. *Introduction to Linear Algebra*. Fifth edition, 2016.

[36] Edward A. Lee Varaiya and Pravin. *Structure and Interpretation of Signals and Systems*. Second edition, 2011. URL: LeeVaraiya.org.

[37] Arthur Gelb. *Applied Optimal Estimation*. illustrated, reprint edition, 1974.

[38] Norwegian University of Science Technology. Home page. URL: http://www.ntnu.edu/ [last accessed 11.10.2016].