ABSTRACT

Marine plastic pollution is a large issue these days. Removal of marine plastic pollution is sadly an underrepresented topic in research and development. This report continues work on a simulation and control system intended for a solution that will remove marine plastics. The system consists of an unmanned surface vessel and a negatively-buoyant remotely operated vehicle with a gripper attached. Because the remotely operated vehicle is negatively buoyant, the two vesssels are coupled, meaning that the movements of one inherently affects the movements of the other. This provides interesting control challenges.

In this project, a simulator and control system have continued development from a previously made baseline, and the system has been simulated in various sea states. The results show that the controller is currently not tuned well enough and that the simulation has some work to be done, but the results are nonetheless promising.

Plast i havet er et stort problem i moderne tid, men det er relativt lite forskning når det kommer til fjerning av havplast. Denne rapporten fortsetter arbeid fra et tidligere prosjekt, med fokus på simulering og kontroll av en løsning for å fjerne havbunnsplast. Løsningen, som ble fremmet i det tidligere prosjektet, består av et ubemannet overflatefartøy og et fjernstyr undervannsfartøy. Undervannsfartøyet har negativ oppdrift, som gjør at systemet av overflatefartøyet og undervannsfartøyet blir koblet. Det vil si at krefter som påvirker den ene parten også vil påvirke den andre. Dette fører til interessante problemer for et kontrollsystem.

I dette prosjektet har simulatoren og kontrolleren fra det tidligere prosjektet blitt videreutviklet og testet i mer komplisert sjø. Resultatene av simuleringen viser at kontrolleren trenger mer justering for å bli brukbar og at simuleringen har mangler. Allikevel er resultatene lovende for fremtidig arbeid.

CONTENTS

CHAPTER

ONE

INTRODUCTION

1.1 Marine Plastic Pollution

Marine plastic pollution is a widely documented issue. The exact amount of plastic is not known. The World Economic Forum (2022) estimate there to be betweel 75 and 199 million metric tons of plastic waste currently in the ocean, while Jambeck et al. (2015) estimated that the annual release in 2010 alone was between 4.8 and 12.7 metric tons. According to Isobe and Iwasaki (2022), most plastic types sink, it is therefore reasonable to focus work on subsurface plastics.

There are two major paths for health effects from plastic pollution, larger pieces being broken down into microplastics which then get taken up into the food chain, and material leeching into the surrounding seawater. Once microplastics have entered into the foodchain, leeching becomes a concern within creatures as well.

Leeching refers to the chemical breakdown of materials going into solution, or compounds moving from one carrier to another. For example, bisphenol A (also known as BPA) is a material commonly used in polycarbonate production. Rochester (2013) show in a meta-review that Bisphenol A can affect the endocrine system of humans, especially fetuses and infants. These materials can with certain types of exposure, such as mechanical abbrasion or acidity, leech from their polymerized form into the surrounding water. Other researchers, such as Sørensen et al. (2023)Sørensen et al. (2023) show that some chemicals associated with rubbers and elastomers have a toxic effect on certain marine microalgae. Another paper by Sørhus et al. (2023)**?** shows the negative effects of crude oil on haddock roe and larvae. In short, plastic leeching has been shown to impact the health of both humans and other living things.

Microplastics on the other hand, refer to small particles of plastics, less than 5mm in size along their longest axis. Microplastics might be broken up from larger pieces, for instance by wave motion grinding waste against the seafloor or against each other. Microplastics might also be released from products that include them intentionally, for instance certain exfoliating soaps or polishing compounds use small plastic beads to do the polishing/exfoliating work. The negative health impacts of microplastics both on human and animal life have been shown, for example by Segovia-Mendoza et al. (2020) and Zolotova et al. (2022). Filtering out microplastics is difficult due to their small size clogging filters and other filtering methods quickly. One of the best solutions for dealing with microplastics is to simply avoid them being created and released. The release of microplastics is not something this report will go further into, but removing larger plastics from the seafloor would effectively halt their breakdown and thus halt their release of microplastics.

The above paragraphs demonstrate the health effects of marine plastic pollution, but additionally there is the concern that the physical presence of plastics might harm or inhabit marine life. Ghost fishing for instance, is a large concern from an ecological perspective. Ghost fishing refers to when tools such as fishing nets or pots are lost at sea but continue to drift, allowing them to catch animals that end up harmed or dying without human control. Matsuoka et al. (2005) goes into the issues of ghost fishing in greater detail and explains more fully how and why it is an issue.

Plastics in the ocean are a large issue and this report aims to be a part of alleviating it.

1.2 Previous work

Work on preventing plastics from entering the ocean generally would fall under public works. Public services such as convenient trash collection from private people and businesses, fines for littering or mis-sorting waste and so on, help to prevent waste from reaching the ocean. There are also some public works such as in the australian city of Kwinana where the city has installed drainage nets (fig. 1.1) leading into a nature reserve to prevent contamination

Figure 1.1: Illustration image of a drainage net installed in the city of Kwinana, Australia. The nets installed are used to collect waste before it enters a natural reserve. Image credit of the City of Kwinana

of the reserve from the waterways.

When it comes to collecting waste already released, there are several projects currently underway. One of the most famous is The Ocean Cleanup, illustrated in fig. 1.2. The Ocean Cleanup have shown the efficacy of their solution and are currently working to reduce surface plastic pollution in the oceans and certain rivers. There are also other projects like SeaClear, which primarily focus on removal of waste from shallower and clearer waters where visual identification from the sky is possible.

Figure 1.2: The Ocean Cleanup's ocean solution, consisting of a large net dragged behind two ships. The net captures surface plastics into a hose that can later be collected and dumped on another ship for sorting and disposal. Credit: The Ocean Cleanup

1.3 Remaining work

While the SeaClear project and The Ocean Cleanup are excellent and have done wonderful work, their solution is still very limited with regard to the operational parameters. SeaClear have quite strict limitations regarding water depth and visibility. The Ocean Cleanup on the other hand primarily do their work on the ocean surface. There still remains a metaphorical missing piece of marine waste removal, and that is deeper water semi-autonomous removal. This is what the Plan Sea project which this report is partially based on revolves around.

Through the Plan Sea project, further discussed in Kjørseng (2024), a solution was proposed which includes a surface vessel with an attached remotely operated underwater gripper/remotely operated vehicle(ROV) solution that is able to descend to deep water and pick up litter. This solution has a proof of concept still under development. The project which this report is based on is a part of the Plan Sea project.

Through the semester this master's thesis has been written, a project has also been continued. The project has as a goal to develop a simulation suite and basic control system that allows for development of the control system for the full scale Plan Sea project. The project has been heavily based on previous works as laid out in Kjørseng (2024). The specific goals of this thesis are laid out in section 2.3

1.4 This report

The two main parts of this report are the control system architecture and the continued development of the simulator. In Chapter 2, these two main parts are further described with their own individual problems and hurdles.

In Chapter 3 the design and implementation of the simulation and control system are described. The setup for the simulation test cases to be performed are also laid out here.

Chapter 4 shows the results from the simulations and the control system.

Chapter 5 and contains the discussion of the results and the work done on the control system and simulations. And Chapter 6 is a short conclusion of the project.

Because of the size of the results from the simulations, all the results are included in the appendix A, while only select figures are brought up in chapters 4 and 5.

Additionally, because of its importance to some of the topics in this thesis, the specialization project report is also included in appendix B

CHAPTER

TWO

PROBLEM STATEMENT

As stated in the introduction, this report and the project it is written about are heavily based on Kjørseng (2024). This was intended from the start. The goal of the specialization project was to provide a baseline to work from for this thesis.

Because of their highly interconnected nature, the specialization project is summarized here in some detail to provide appropriate background. As mentioned in the previous chapter, the specialization project is also included in appendix B

2.1 Plan Sea

The Plan Sea Project is a student-driven project with the goal of creating a solution to remove plastic from the seafloor. It is described in detail in the specialization project. In short, the current solution for the Plan Sea project can be broadly divided into three parts: A surface vessel that acts as an operations platform, an underwater gripper that is able to detect and grip plastic on the seafloor, and a basket to collect the plastics for the gripper. A sketch of the Plan Sea project can be seen in fig. 2.1. The collection basket has been postponed to focus on the primary parts of the project.

The Plan Sea project is described in greater detail in Kjørseng (2024) and will thus not be further described here.

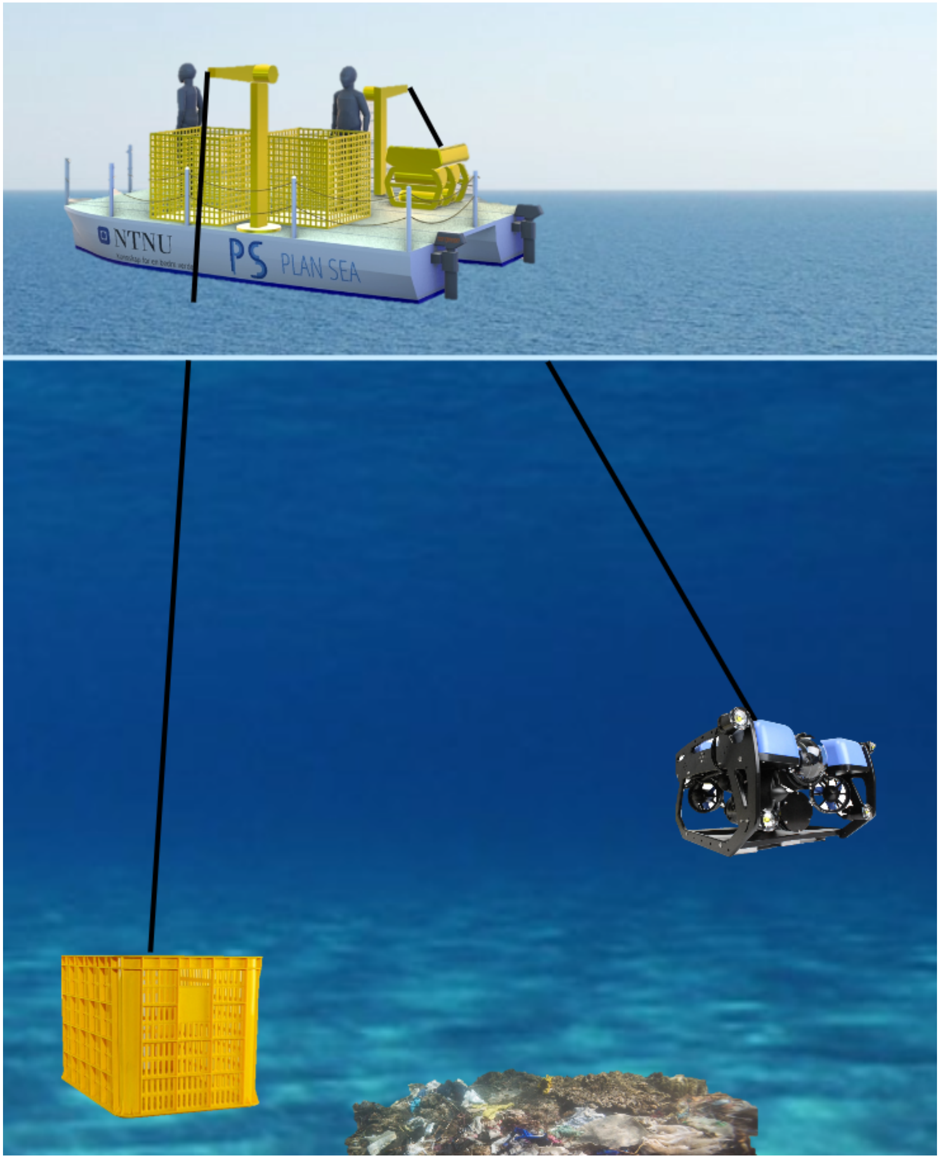


Figure 2.1: A rough overview sketch of the Plan Sea project proposed solution, credit Kjørseng (2024)

2.2 Specialization Project

The specialization project (Kjørseng (2024)) focused on making the framework for this thesis and for further work on the Plan Sea project in general.

Originally, the goals of the specialization project were as follows:

1. Create a simulation to be used in later work
2. Validate the performance of the simulation
3. Create a rudimentary control system for the simulation that can be further developed later
4. Do basic research that can be implemented in this thesis

These goals were achieved with great success. Here is a short summary of each goal and its results.

2.2.1 Creating the simulation

It was clear from the problem statement of the specialization project that a simulation was necessary. A control system without anything to control is not much use to anyone. The simulation would provide a plant for the control system, allowing at least basic development and testing of a seagoing vessel while staying warm and dry on land. With it clear that a simulation is necessary, a question arises: is it better to build a simulation from scratch or to use an existing framework?

When creating a framework from scratch, the developer has full control of everything. This is excellent for experienced developers in well-supported teams. A from-scratch solution could simulate exactly what is needed and in the exact level of detail needed. However, creating a simulation from scratch requires a lot of in-depth knowledge about many topics, including computation, physics, hydrodynamics, program optimization and more. In reality, this is a multi-person job simply due to the knowledge requirements.

Using an existing framework has the benefit of being practically a ready to use solution. It also avoids the issues of the from-scratch solution because these frameworks are generally made by large groups of people with all the in-depth knowledge necessary. The main drawbacks of using ready-made solutions are licensing costs and the learning curve of the simulation framework and its workings.

From the above, it was concluded that a commercial solution would be preferable, especially considering the fact that the simulation in question would require good handling of hydrodynamics and wire simulations, two notoriously difficult simulation topics. Additionally, using a finished commercial simulation solution would make it so the specialization project could focus on implementation. If it was decided that a from-scratch solution was to be made, the specialization project would likely primarily be about the programming of the simulator rather than the development of the control system. Preferably the simulation framework would be usable through code, not only through proprietary software. This requirement would make it easier to interface towards the simulation with the control system once that is implemented. Also, the simulation code should preferably be scriptable with Python due to the candidate's preference for the langauge.

Given all the above, the simulation framework AGX Dynamics made by Algoryx AB was chosen. The framework has modules for wire simulation, including winching movements. It also has a hydrodynamics package for handling buoyancy of the vessels. Additionally, AGX Dynamics is made to interface with ROS2 which would become a bonus for this project. This is elaborated upon in section 3.1.1

The Plan Sea project is based on two vessels. These were implemented into the simulation separately. The surface vessel(USV) was implemented as a simplified version of the designed hull. The ROV was implemented as a simple box with the appropriate dimensions. The basic setup can be seen in fig. 2.2. Due to the demands of the Plan Sea project, the ROV would be negatively buoyant and hanging by a tether from the surface vessel. This was done to increase the lifting capacity of the ROV and is further discussed in Kjørseng (2024).

2.2.2 Validating the simulation

The simulation was validated by comparing the tension measured in the tether to a calculated drag force that would be expected. The surface vessel would be moving at constant speed and the ROV dragged behind. This was done at several speeds, and the tension was measured both on average and as a function of time. The measured tension was then compared with a manually calculated drag force using the drag equation. The simulation was found to perform within $\pm 20\%$ of the calculated drag, which given the assumptions made was concluded to be well within reasonable limits. Further discussion of this can be seen in section 4.2 of Kjørseng (2024)

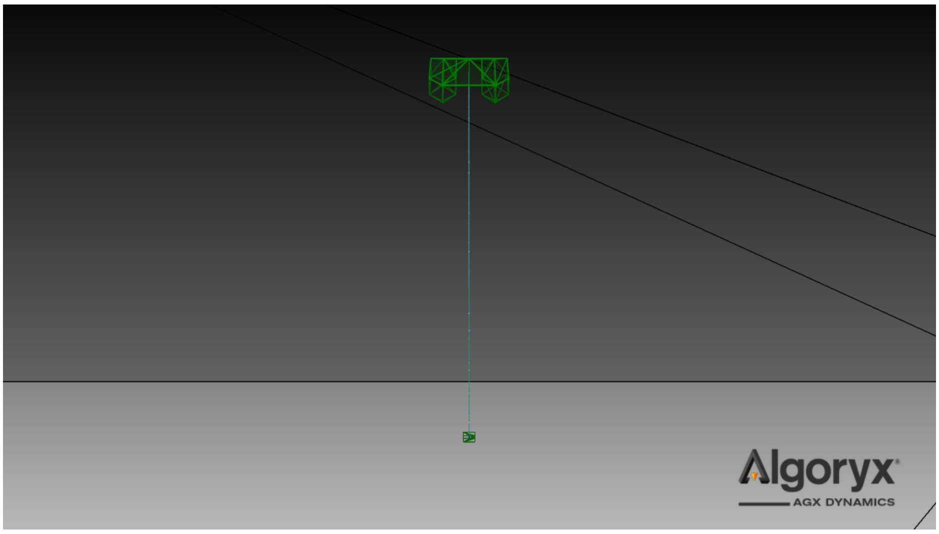


Figure 2.2: The display output from the simulation software with the two vessels implemented. Credit Kjørseng (2024)

2.2.3 Creating a control system

In the specialization project, the control system was implemented directly in the same script as the simulation ran on. This was not ideal, but it did work for a proof of concept. As it was implemented, it could control both the heading of the USV towards its waypoint as well as its total force applied. It was implemented as navigating to a list of points, changing to the next point after it reached some steady state around the current one. The movement of the ROV and crane was not controlled in the control system achieved in the specialization project.

Originally, implementing the control system using ROS2 was a goal of the specialization project. this was removed near the middle of the project due to the complexity of setting up the required framework. ROS2 stands for Robot Operating System 2 and is not really an operating system. ROS2 is a framework for communication between separate nodes of a larger control system. It allows for systems to be easily expanded or moved between different settings and developers. Especially helpful for the Plan Sea project would be the ability to make the simulation and the real-world vessel interchangable. By ensuring that the two communicated through the same channels with the same format of data, the control system outside of the plant would be able to control either of the two. This would allow for the same identical control system, tuned and tested in simulation, to be used almost immediately in the real world.

Due to time constraints and difficulty in setting up a working work-environment for ROS2, it was omitted from the specialization project and remained for the project in this thesis. The implementation of ROS2 is further discussed in section 3.1.1

2.3 New Problem Statement

With the baseline of the specialization project, this project can come with its own problem statement that it attempts to solve for.

- Decouple the control system from the simulation and implement it in ROS2
- Expand the control system to include winch and ROV movement
- Create a human-machine interface/graphical user interface for the control system
- Test that the new control system and simulation work well enough for further development

CHAPTER

THREE

DESIGN

This project can be very broadly divided into two segments, the simulation segment and the control system segment. The goal is that the simulation is to be as true to reality as possible. With a true-to-life simulation, a control system can be built to work in the simulation. Once the control system is created and tested satisfacotrily, it can hypothetically be "transplanted" to the physical test. Because of these two different segments, I will talk about them separately.

3.1 Details on the simulation setup

The exact details of the implementation are more specifically brought up in Kjørseng (2024). Below is a brief elaborate of the simulation setup, further than what was mentioned in section 2.2.1.

The two vessels to be simulated are the surface vessel and the ROV. The ROV was simplified as a cuboid with the proper dimensions according to the manufacturer's specifications. The ROV to be used is a modified BlueROV2 Heavy made by Blue Robotics. The dimensions of which are 575 x 254 x 457 mm (WxHxD). The density of the ROV has not been tested, but is implemented in the simulation as 2000kg/m^3. This is a rough estimation and the real density is likely lower. A high density like this is chosen because the ROV is supposed to be negatively buoyant,

The surface vessel was modelled roughly in CAD and then exported as a

`.obj` 3D-file for AGX to work with. The dimensions and shape of the vessel are roughly corresponding with the real vessel, though not exactly as it is-intended to be exchanged for a more detailed model produced by a different master project at a later date. The density of the surface vessel is implemented at 600kg/m^3 which was arrived at by taking a rough average of the different densities of carbon fiber sandwich plate we had for building the hull and then adding some extra mass to account for batteries, sensorics and other added weights.

As a note on the densities of the vessels: while AGX is able to model non-uniform densities and varying density distributions, this was not implemented in this simulation simply for the constraint of time. It was assumed that the loss of accuracy from assuming uniform density is small enough not to matter. This will be further discussed in section 5.1

Since the ROV sinks it needs a constant force to not sink to the bottom of the simulated ocean. This is achieved by a tether connected to the surface vessel. The tether is modelled as a non-buoyant rope/wire with a radius of 10mm and a Young's modulus of 10^9. These figures are all assumed values and should be corrected when the real values from the physical implementation are known.

The two vessels and the tether are placed in a 200x200x120m pool of water. It is possible in AGX to simulate different sea-states, this was a part of the development work done for this project and is discussed later.

3.1.1 Improvements made since specialization project

In addition, the simulator now is capable of simulating the winching movement of the winch on the ROV. Previously it was implemented as a fixed length wire. This means that the crane now is able to operate. A basic controller for the crane has been created as well.

The simulation has been further developed to allow for more dynamic changing of the seastate. Originally the seastate was implmented as no sea. This is now changable by changing a single variable. The sea is simulated using the wave height equation, eq. (3.1)

$$h = 0.5\sin(0.5x + 0.6t) + 0.25\cos(0.6y + 0.3x + 1.45t) \tag{3.1}$$

Where h is the height above $z = 0$, x and y are the position in the horizontal plane and t is the time since the simulation started.

All of the code for the simulations as well as for the control system is available on Github at `https://github.com/MagnusKjorseng/FordypOgMaster/tree/recovery/Simulator/V4` noa

3.2 Control System

The control system implemented in the specialization project was simply taking in the position of the vessel directly from the simulation, calculating the error from the desired position and then calculating a command based on a PID controller. This worked well enough for a proof of concept, but is difficult to work with when the project increases in scope.

One of the achieved goals of this project has been to decouple the simulation from the control system. This has been done for several reasons relating to the control of the vessels. Most relevant for this project has been a decrease in development and testing time. With the different nodes of a system decoupled, it is possible to work on each node more or less individually, connecting or disconnecting them as needed. Using the node based structure of ROS2 also makes it a lot easier to develop new nodes, as the communication between nodes is clearly defined in the topics and services the nodes communicate over.

The overall desired architecture of the new control system is shown in fig. 3.1. If translated into "ROS terms", the boxes shown are nodes while the arms connecting them are the topics they communicate via.

There are some nodes in fig. 3.1 that are not as obvious at first glance, these are described further here. The "position" and "command" translation scripts exist as a middleware between the simulator and the control system. In order for the control system to be as true-to-life as possible, it has been implemented to accept position data for the USV in the form that a GPS antenna would output, so-called NMEA messages. Message types in ROS2 are strictly defined, and there exist packages which supply these NMEA messages. However, due to limitations of the simulation framework the simulator is not easily able to handle other ROS packages than the ones it has been made with, and the NMEA package is not among them. Therefore, the translation scripts work between the simulator and the control system proper. The simulation outputs position and orientation of the vessels in its own world-space, and the position translation script converts this into a lat-lon format based on a predefined baseline. This then is packaged into the correct format and sent to the guidance system. Similarly, the command

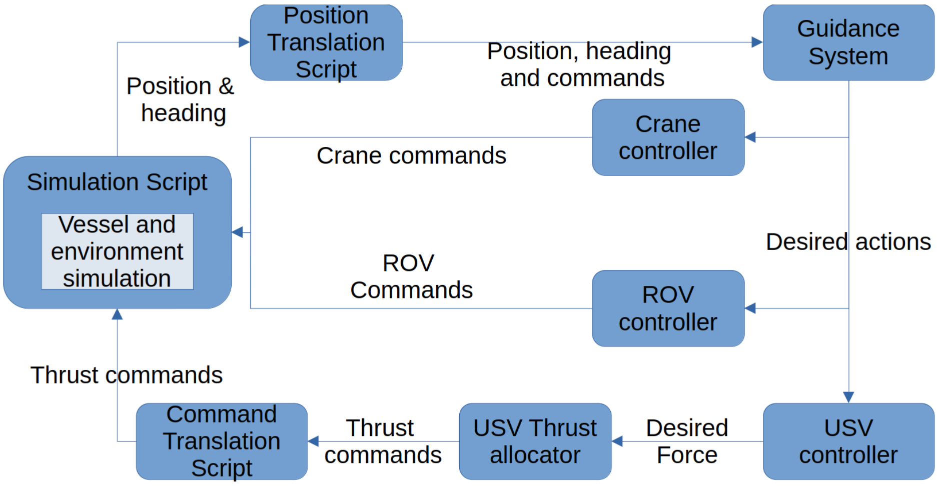


Figure 3.1: The shape of the designed control system

translator receives the commands in a custom format and translates it into a format that the simulator can work with. In a real-world setting, these translation scripts would not be included.

The guidance system is a sort of "central controller", it is what decides the setpoints for the other controllers and instructs them. For instance, on the topic of mastering, the central controller is what would handle that. Mastering, as mentioned in the title, refers to which of the two main vessels, ROV or USV, is "in charge". Because the movements of the vessels are dependent on each other, it would not work to have them try to move in opposite directions from each other. With a more conventional neutrally buoyant ROV this might work, as the tether does not provide as firm of a link between the vessels. This is further discussed in Kjørseng (2024). Ideally, this control system would be able to have either the USV or the ROV be the master, depending on the given situation. If the ROV is undertaking precision work on the seafloor, the USV should work to disturb as little as possible. If the USV needs to move while the ROV is deployed for one reason or another, the ROV should attempt to follow to not put undue strain on the wire. This is what is referred to as "reversible mastering", and the guidance system would be in charge of this. Additionally, the guidance system would handle how the crane and ROV handle vertical movement. Because of the length and elasticity of the wire, it would be a good idea to have only larger movements be taken up by the crane and allow the ROV to use its vertical thrusters to do smaller corrections. This is because due to the large length of the wire, the forces take a long time to transfer all the way down to the ROV. A lot of the movement might also be taken up as stretch in the wire and never properly reach the ROV. Because of these considerations, the crane and ROV should have different height controllers, and this is handled by the guidance system.

The last new part of the system in fig. 3.1 is the USV thrust allocator. The allocator works like a translation layer between the controller and the local controllers of the USV's thrusters. The controller provides a force input on the vessel's center of gravity, but the thrusters need to know how much to thrust and in which direction. The allocator figures this out.

In abstract terms, the allocator finds and applies a transformation matrix T such that

$$Tf = \tau$$

where f is a vector of vectors with the lateral forces for each thruster. For this case with two thrusters, it will look something like

$$f = \begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \end{bmatrix}$$

The transformation matrix can be written explicitly for the USV, since it has a very simple thrust configuration. For larger configurations it might be better to write each thruster's transformation individually and either add or remove them depending on the type of move necessary (larger moves use only larger thrusters etc.).

$$T = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ -l_{y_1} & l_{x_1} & -l_{y_2} & l_{x_2} \end{bmatrix} \tag{3.2}$$

Since τ and T are known, we can find f by performing a pseudoinverse on T leading us to the equation

$$f = T^{\dagger}\tau \tag{3.3}$$

where $T^{\dagger}$ is the pseudoinverse of T.

This can also be written longform as

$$\begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \end{bmatrix} = T^{\dagger} \begin{bmatrix} X \\ Y \\ N \end{bmatrix} \tag{3.4}$$

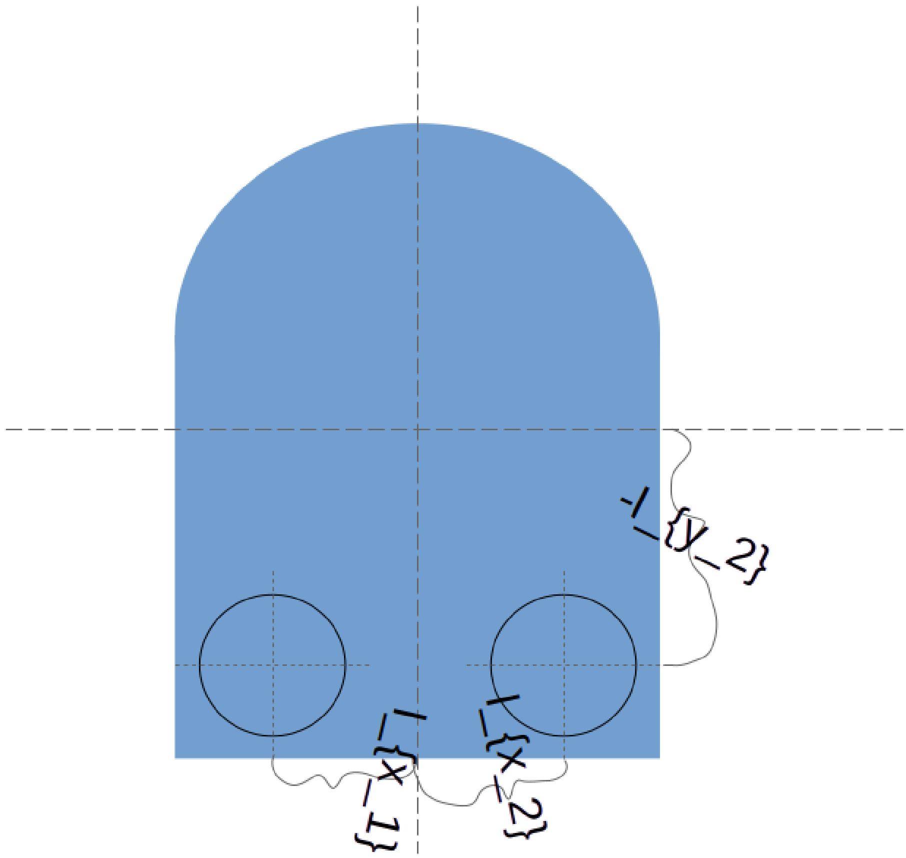


Figure 3.2: Rough sketch of how the thruster position values in eq. (3.2) are found. The thruster configuration will be input to the allocator through a config file.

The full matrix for $T^{\dagger}$ is omitted because the pseudoinverse of a non-square matrix tends to be large and ugly. Because the pseudoinverse will only ever be calculated and used by the allocator, the exact values of it are not important to know or keep track of for humans.

3.3 Case description

The USV controller implemented is quite simple. It has no predictive capability and is completely reactive. This scenario is made to compare the controller's actions as well as the error of the system in varying seastates. The simulation framework does allow for current and wind simulations as well, but these are not implemented and thus will not be simulated. The goal for the surface vessel is to stay stationary, simulating an operation where dynamic positioning is necessary for the ROV to do its work. Additionally the ROV will attempt to follow the movements of the USV so as not to put undue tension on the tether.

Five cases will be compared here, from 0m wave height to 2m wave height, in increments of 0.5m. The different kinds of seas are simulated with the same wavelength. The wave simulation is a simple regular wave in two dimensions with its wave height defined by the equation eq. (3.1). The equation was taken from Algoryx's examples for buoyancy simulation for simplicity and ease of use. The actual value of the equation is irrelevant in the larger scheme of things because the only thing it's doing is providing a local height offset. This offset will then be used to determine how far from the zero-height of the world the vessel is. This equation has been deemed to be "random enough" for the control system to get some challenges. Further discussion around using a regular seastate is done in section 5.1.1

The expected response assuming a well functioning system is that the error should be relatively minor, but there is the danger of the seas being too high. If the seas are two high there are two potential faults, the surface vessel might capsize or the thrusters might exceed their authority. The first fault is obviously disastrous and should be avoided at all costs. The second is unfortunate and might lead to loss of control or scrubbing of the mission. Both of these faults can be used as limits to find the maximum acceptable operational criteria for a mission.

CHAPTER

FOUR

RESULTS

4.1 Case results

The case was tested in which the ROV and USV are initialized at the desired point to hold with a given seastate. There are two variables modified during testing: the length of the wire (and thus the depth of the ROV) and the waveheight. Specifically, the ROV's height was tested at 0m, analogous to being stowed, 50m and 100m. 0m was chosen as a baseline, showing the USV's performance when not affected by the drag of the ROV. From the plans for the project at the time of writing, the planned maximum operating depth is 100m, making this a natural second datapoint. 50m was chosen as a midpoint between the two for additional data collection.

Data was gathered by a data collection ROS2 node which collected and plotted the data. The data was collected over a period of 30 seconds. This time period was chosen because it was difficult to have the simulation be stable for a longer period while the controller was active. While adrift the simulation seemed to be stable indefinitely, but the controller seemed to bring with it instabilities in the wire simulation. It was possible to collect all the data at 30s intervals however, and the data does give enough information to draw conclusions from.

There were five different wave heights that were used for simulations, 0m to 2m in 0.5m intervals. Looking at the simulation and the results, only the three lowest of these seem reasonable to compare. The vessel tumbled a lot for the higher wave heights and the simulation started being very unstable. When the vessel tumbled, sometimes it also got "wrapped up" in the wire, causing further instability. This wrapping can be seen in fig. 4.1

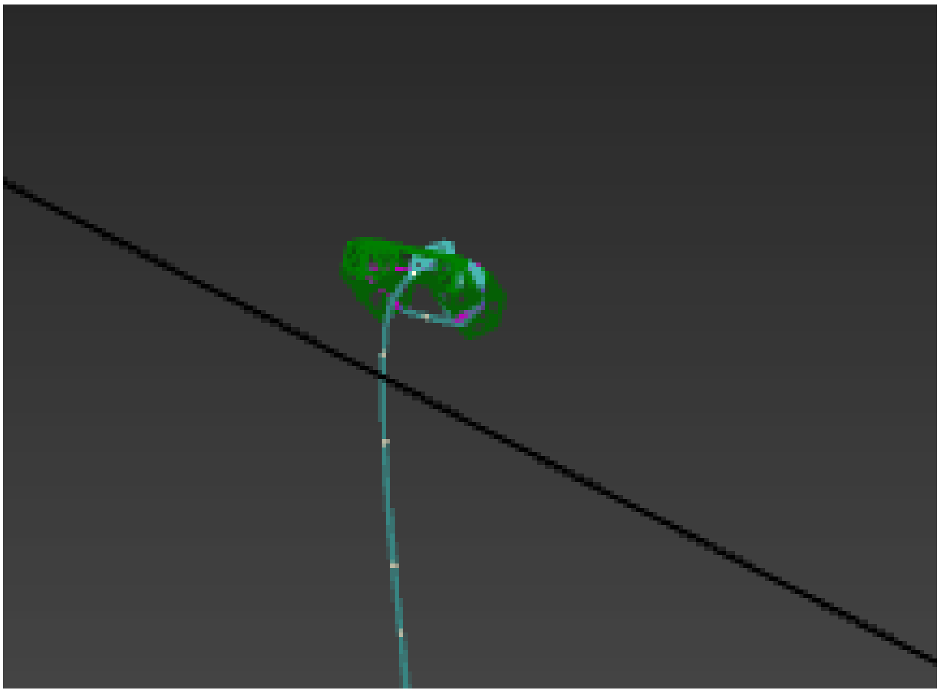


Figure 4.1: Figure showing the USV entwined in the ROV wire due to tumbling in the waves. The vessel is the green wireframe while the tether is the teal and pink line.

As mentioned in section 3.1.1, the equation used to simulate the waves is eq. (3.1), reiterated here.

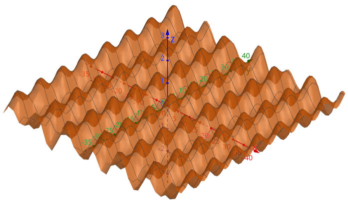
$$h = 0.5\sin(0.5x + 0.6t) + 0.25\cos(0.6y + 0.3x + 1.45t)$$

Where h is the height above $z = 0$, x and y are the position in the horizontal plane and t is the time since the simulation started. The shape of this equation can be seen plotted in fig. 4.2

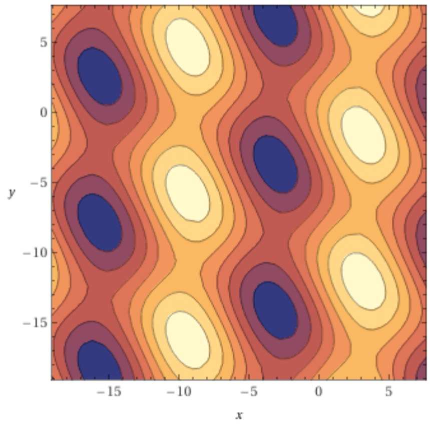
A lot of data was collected. Only the data which will be directly discussed in the next chapter will be brought up here. All results can be seen plotted in appendix A.

4.1.1 Uncontrolled behaviour

As a baseline, the USV was placed in the simulation with no control system applied. The results with no waves are seen in fig. 4.3, while with higher



(**a**) 3D representation. Note that the Z-axis is not scaled the same as the X- and Y-axes



(**b**) Contour plot

Figure 4.2: The wave-shape for the simulated wave given at $t = 0$.

waves can be seen in fig. 4.4 and fig. 4.5. These three figures show the USV's position in X- and Y- direction. The starting position is represented by a dot and the desired position is represented by an X in the plot. The line shows the position the USV follows as time progresses. The results in those three figures are very similar for the same wave height, regardless of ROV depth. Therefore, for brevity, only one ROV depth is shown for each wave height.

For the 0m wave case there is minimal movement, on the order of millimeters. This is likely due to noise and chaos in the simulation: small initial conditions randomized with time which cause calculations to be slightly different. It is probably safe to ignore the movement in the 0m case. There are similarities between the 0.5m wave height and the 1m wave height cases, as well as the higher wave heights. There is a movement south-west first before it moves south-east. What likely happens here is that the vessel finds some local minimum in the wave shape function (fig. 4.2) and is carried along. Sadly, the shape of the waves is not shown in the simulator viewport, so this is not visually confirmed. Because of the minimal movement with no wave interference, the 0m wave case will not be further plotted here. It is however plotted fully in appendix chapter A.

The error for the position can be seen in fig. 4.6. The various position errors have a similar shape but vary in magnitude. Comparing the error at 0.5m waves (fig. 4.6) with the position at 0.5m waves (fig. 4.4), it is clear that these results are related. The Y-direction error increases early to about 5m, then is stable. The X-direction error goes first in one direction and then in the

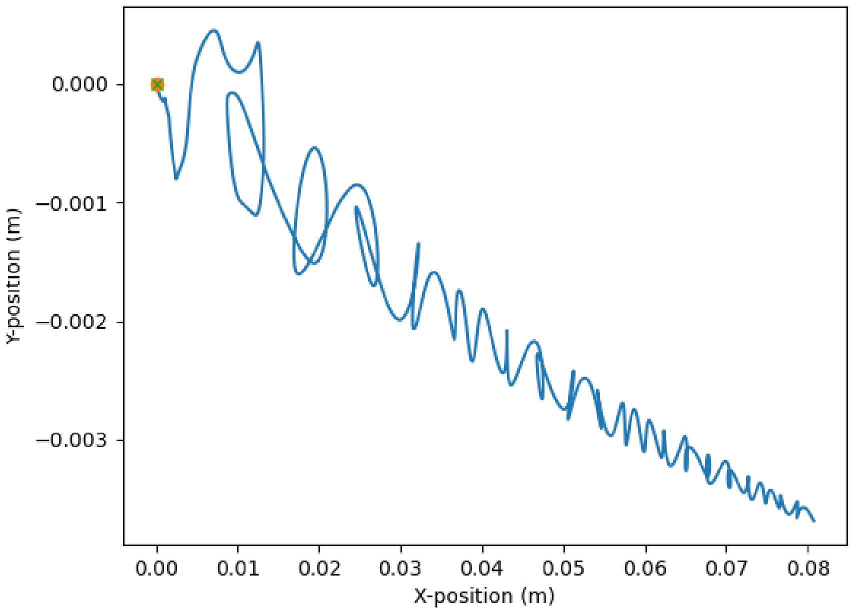


Figure 4.3: The USV's position uncontrolled with 0m waves with the ROV retracted

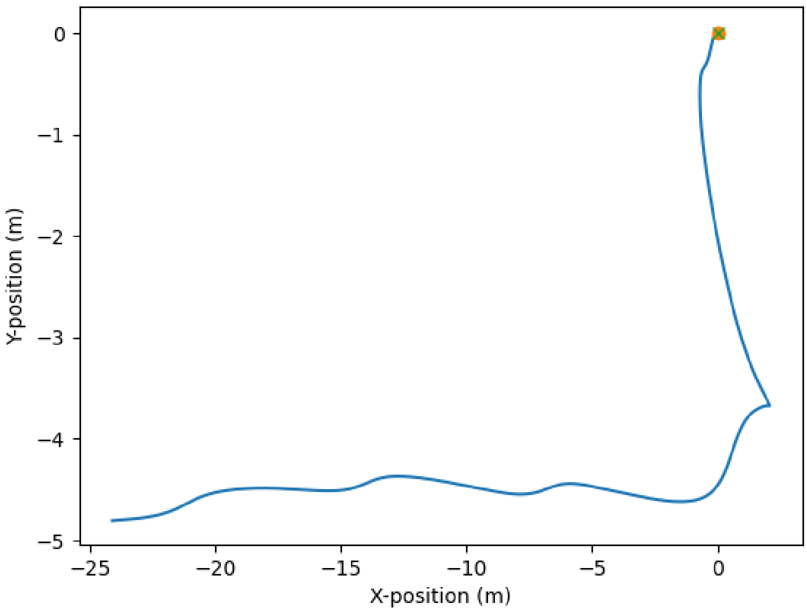


Figure 4.4: The USV's movement uncontrolled with 0.5m waves with the ROV retracted

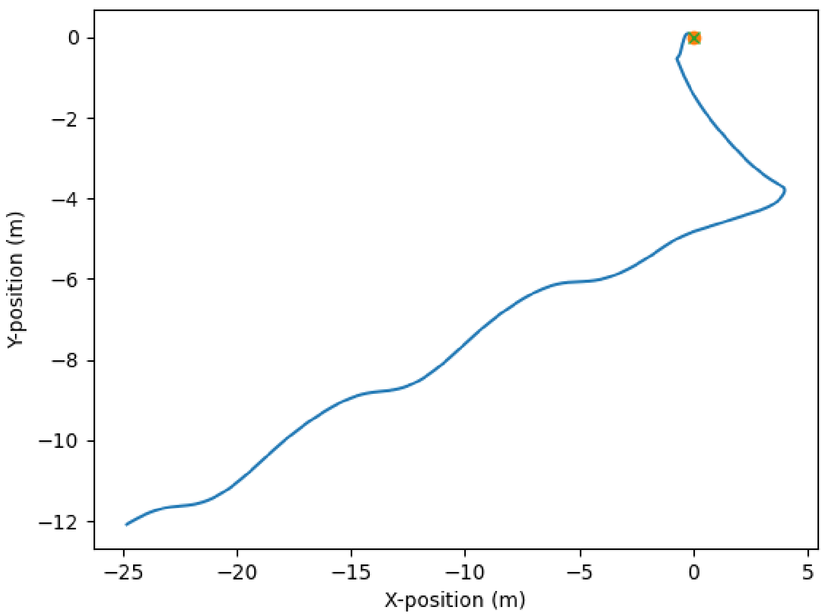


Figure 4.5: The USV's movement uncontrolled with 1m waves with the ROV retracted

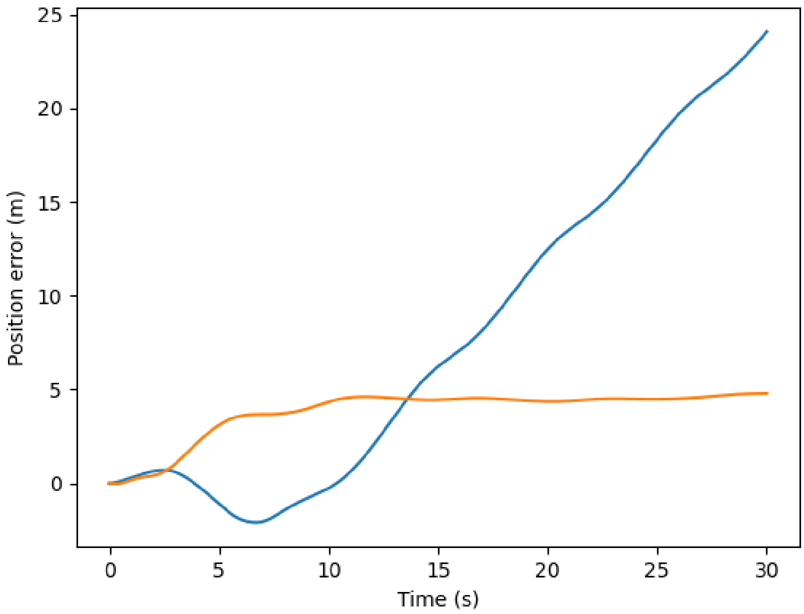


Figure 4.6: The USV's position error, uncontrolled with 0.5m waves with the ROV retracted. The X-direction error is in blue while the Y-direction in yellow.

other until the simulation ends. Comparing this with the position movement, we see a quick rise to $y = -5$ followed by a constant move westwards until $x \approx -25$

The heading of the USV is also charted. This can be seen in fig. 4.7. Once again, the shapes of the different wave heights and ROV-states are very similar, and so for brevity, only one example will be brought up here. The desired heading here is the starting heading, i.e. that the vessel stays facing the way that it started. This heading is 180°, or due south. In the 0.5m and 1m waves it does almost an about face, turning to face almost 30°. All of the uncontrolled heading results show similar results. It is therefore reasonable to assume that the wave trough that carries the USV southwest also turns it to face northeast. This is most likely caused by the hydrodynamic effects of the vessel and it finding a stable state with regards to the seastate. Why this stable state is going in reverse is not known, but further analysis of the hull shape and stability can probably illucidate this.

In this simulation, the ROV is also uncontrolled. With regards to mastering,

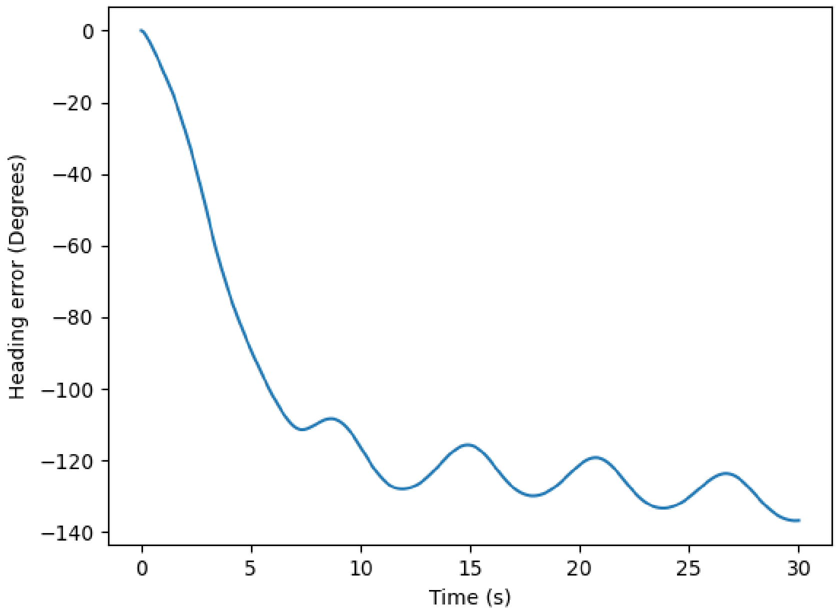


Figure 4.7: The USV's heading uncontrolled with 0.5m waves with the ROV retracted

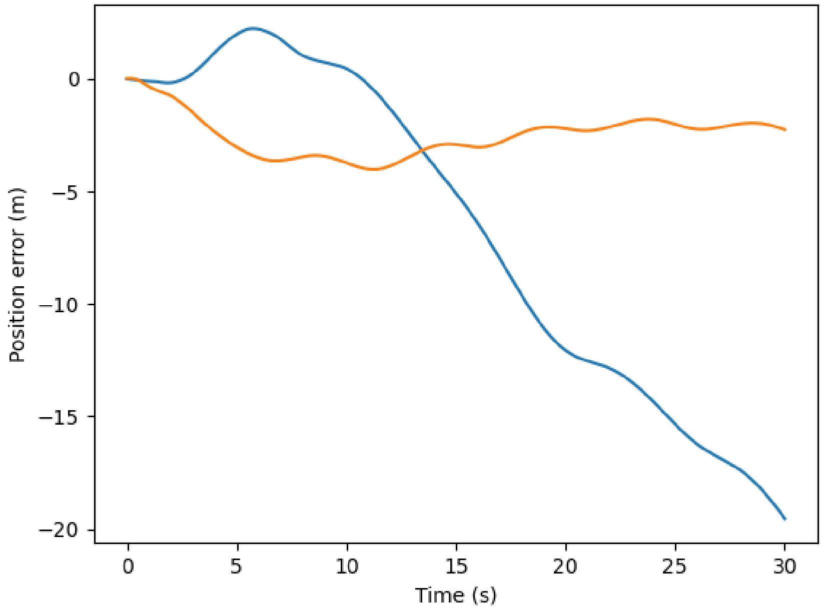


Figure 4.8: ROV horizontal error at 50m depth with 0.5m waves

the USV is defined as the master for this scenario. This means that the ROV's "target position" is directly below the USV. Because of this, the ROV error has also been measured without the control system enabled to use as a control. The position in the horizontal plane can be seen in fig. 4.8 and the depth error can be seen in fig. 4.9. As above, the shapes of the curves are very similar for all cases except the control case, therefore only one example is used. Of note: for some reason the "depth error" plots got some slightly jumbled data in, and therefore plot the "error" as relative to 100m deep. This means that for the ROV at 0m, a value of -100 is desired, for the ROV at 50m, a value of -50 is desired and for the ROV at 100m a value of 0 is desired. This error was spotted too late to remake the plots and thus will be abn error in the depiction.

4.1.2 Controlled response

The same simulations as above were performed but this time with the control system enabled. For this scenario, the starting position and target position are the same at $\begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$. In other words, this simulates the vessel being

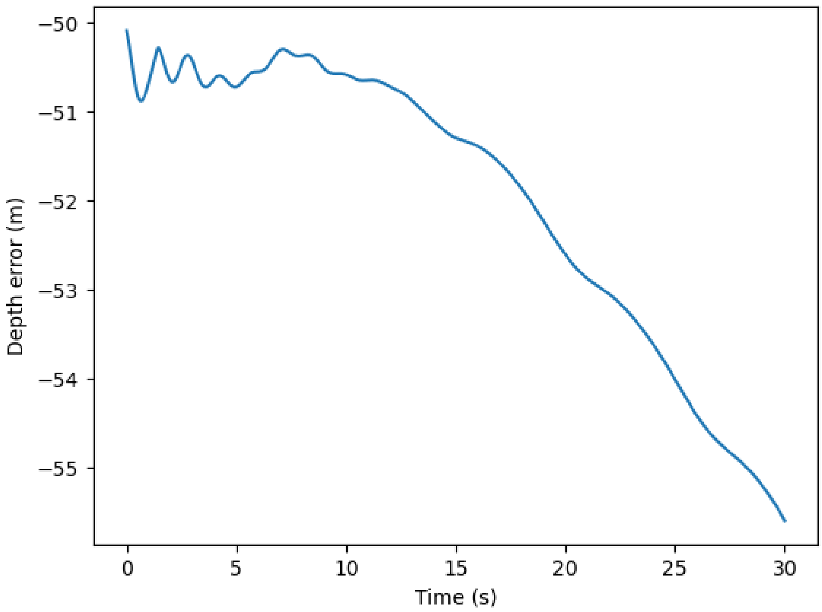


Figure 4.9: ROV depth error at 50m with 0.5m waves. Note the incorrect Y-axis as described in the text.

in a stable position and trying to maintain it. In fig. 4.10 the movement of the USV can be seen for the ROV at 50m depth. Again, the shape of the results are very similar for the varying ROV depths. The shape of the 1m case (fig. 4.21c) is suspiciously close to the uncontrolled case (fig. 4.5). It is not clear whether this is an error in measuring or if 50m is a strange harmonic place. Looking at the results for other wave heights it could be that the simulator simply overpowered the control system in this run, as the results are similar but different across different runs.

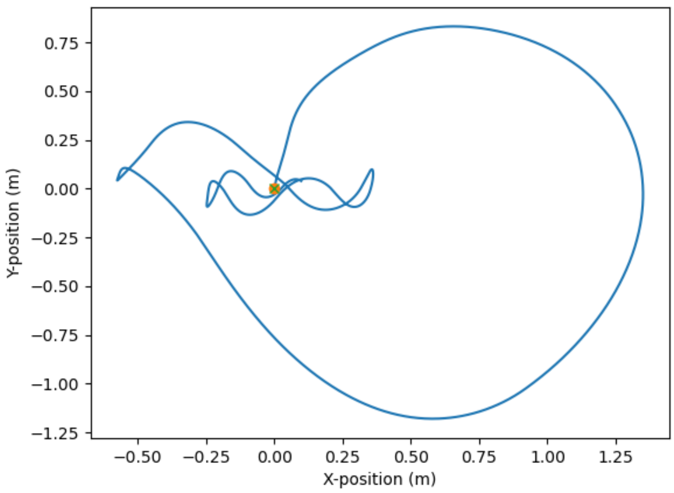
The controlled response for the USV position at higher waves demonstrates where a lot of the issues with the simulation come in. This can be seen in fig. 4.11 which shows how the control system attempts and struggles to move around.

Similarly, the position error both demonstrates that the control system is working in smaller waves (fig. 4.12), but also that it is not working well enough in higher waves, (fig. 4.13).

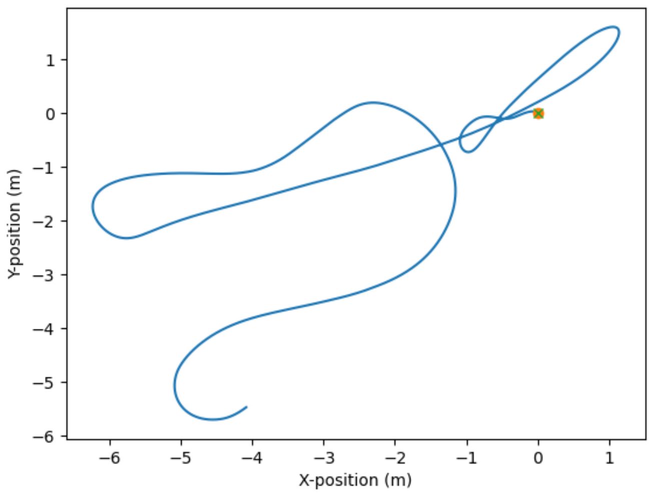
As for the heading error, the control system seems to be working fairly well. It needs more tuning, but these are promising results. The heading with the ROV deployed to 50m but at different wave heights can be seen in fig. 4.16. The results with the ROV "retracted" had some strange harmonics that can be seen in fig. 4.15. This is likely due to the swinging mass that is the ROV hanging from underneath the vessel, fig. 4.14. Realistically, the ROV would have some sort of launch-and-recovery system that locks it into a more stationary position when retracted and stowed. This would all but eliminate this swinging mass problem, but has not been simulated here.

The desired heading for this setting is 0°, or due north. The simulator is initialized heading due south, so the vessel needs to turn 180° around. Due to the nature of the data being constrained between 0 and -360 (as it is an error), the plotting is in this range as well, this causes some strange shapes to appear as the vessel oscillates about 0. This was attempted fixed but could not be done in time. As such, the strange shape of fig. 4.16a is actually nearly perfect, only having some small oscillations. The heading controller has not been tuned very well, and so with more attention paid to tuning it, these results could potentially be improved a lot.

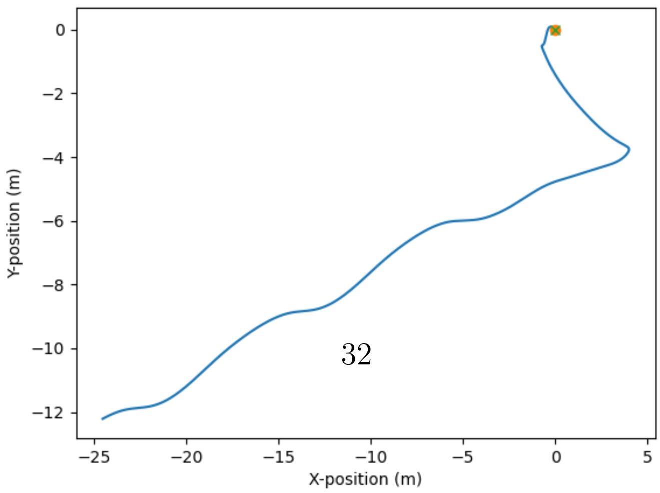
In addition to the heading and position data from the uncontrolled cases, in the controlled cases the forces applied by the controller were also recorded. The applied forces are nearly identical for the different ROV depth configurations. This means that the depth of the ROV does not affect the USV in a large way. This is expected as the wire takes up a lot of the slack, and the



(a) 0m waves

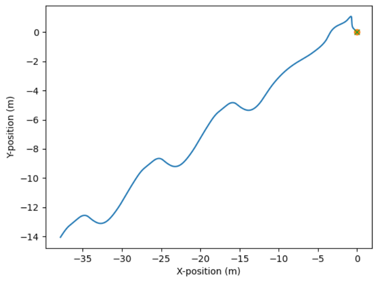


(b) 0.5m waves

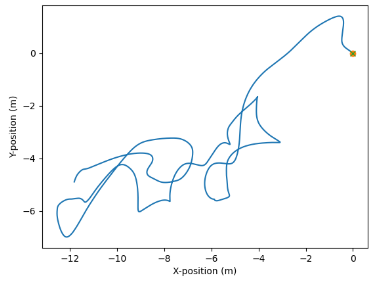


(c) 1m waves

Figure 4.10: The USV's movement controlled with varying

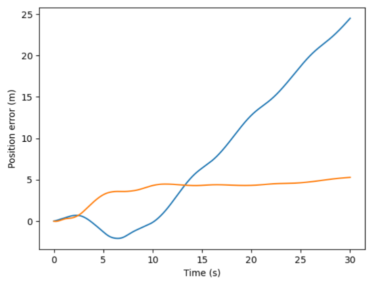


(a) Without control

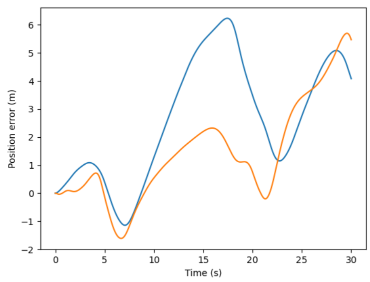


(b) With control

Figure 4.11: USV position with 2m waves with and without the control system enabled. Notice how "squiggly" the position is in the controlled case

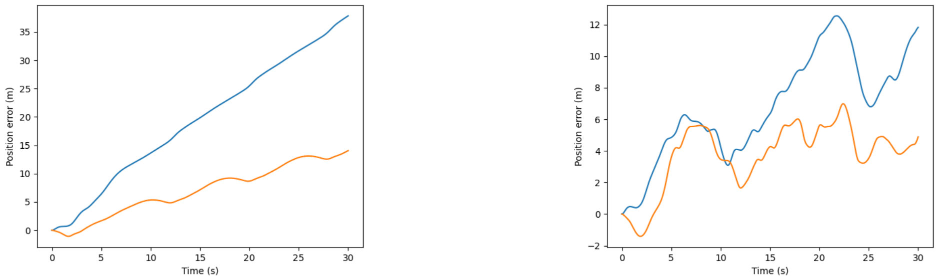


(a) Without control



(b) With control

Figure 4.12: USV position error with 0.5m waves with and without the control system enabled. The amplitude of the error is much lower with control, showing that the control system actually works.



(a) Without control

(b) With control

Figure 4.13: USV position error with 2m waves with and without the control system enabled. The error of the USV is much lower with controls than without

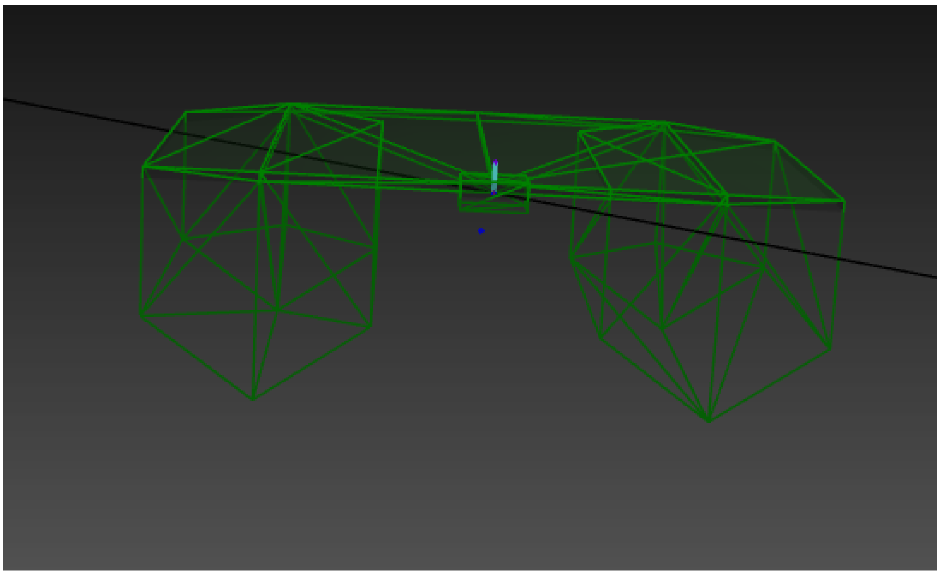


Figure 4.14: Simulator viewport showing the ROV tucked in at 0m wire length

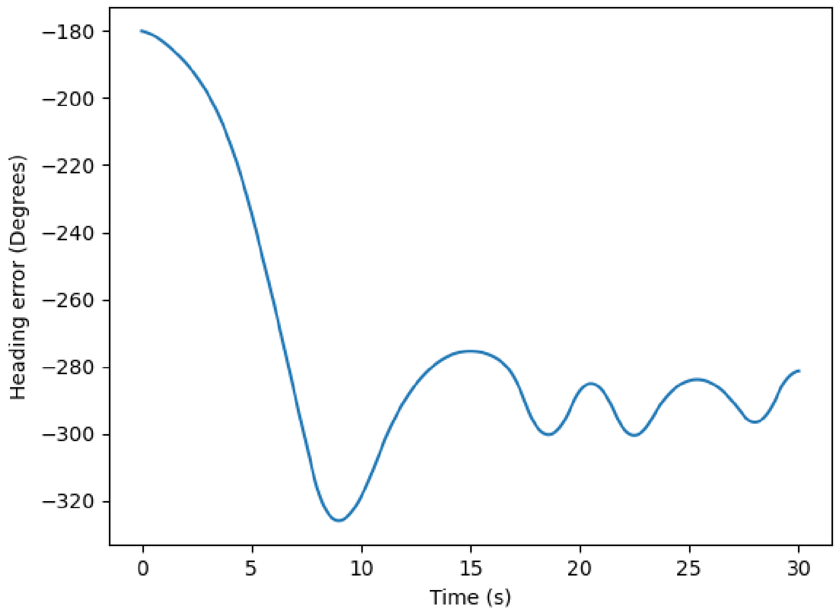
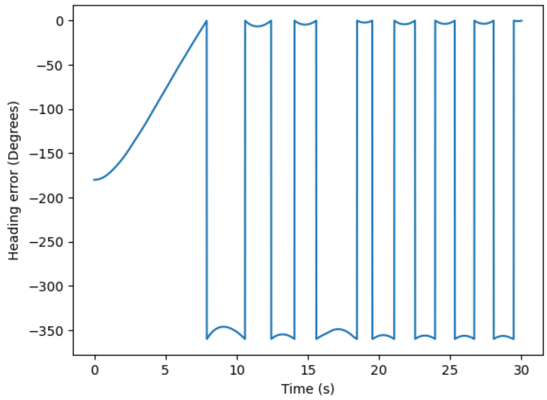
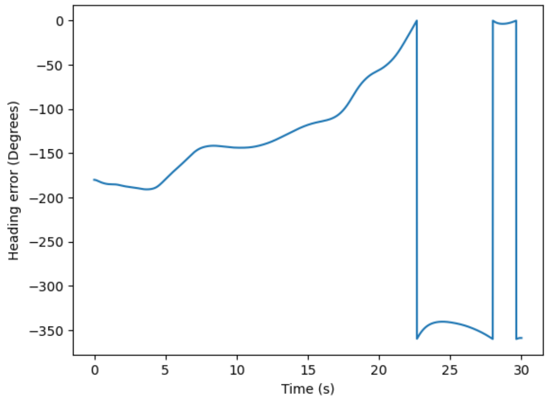


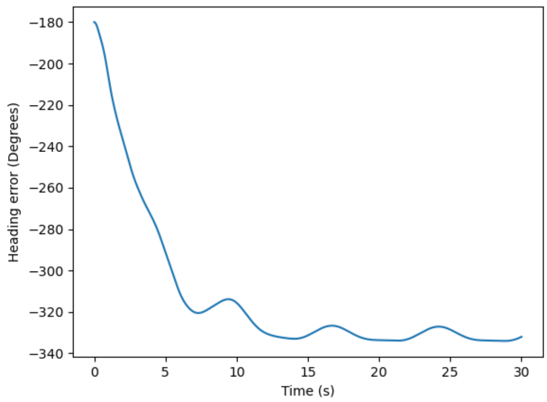
Figure 4.15: Heading error for the USV with the ROV retracted with 0m waves. Notice that the controller is not able to reach the desired set point of 0.



(a) 0m waves



(b) 0.5m waves



(c) 1m waves

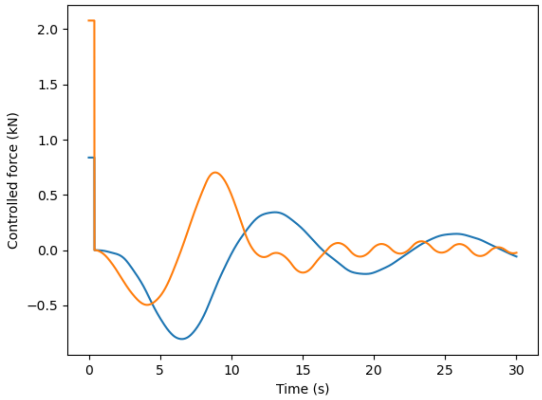
Figure 4.16: The USV's heading error with control and varying wave heights, ROV at 50m depth

ROV is approximately 1% of the mass of the USV. Some examples of the lateral forces can be seen in fig. 4.17 while torque can be seen in fig. 4.19. Of note: both the lateral and momentum forces are capped by the controller. This is to simulate the maximum allowable authority or allowed force in that direction. This cap does not seem to be reached for the lateral forces, but the plot makes it very clear that the torque forces are very underpowered as the system is saturated most of the time.

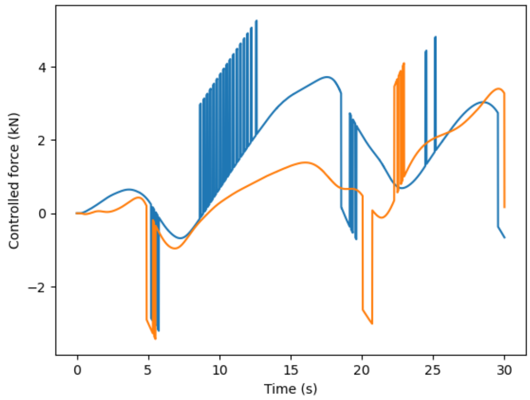
Several of the lateral force plots are strangely discontinuous, compare for instance the two plots in fig. 4.18. These are both for the 1m case but with the ROV at different heights. One possibility for the appearance of these discontinuities is the fact that the controllers are currently not implemented to take into account the time between steps. This means that if the data collector would wait two ticks before plotting rather than one, the amplitude of the plotted point would be doubled. With natural computation delays in the simulation, this could occur quite frequently.

The ROV's controller is also tested and shows similar results to above. It is able to keep up when there is no movement or when the waves are low, but when the waves reach and exceed 1m in height the system struggles to keep up. The case for the ROV at 100m at different wave heights can be seen in fig. 4.20

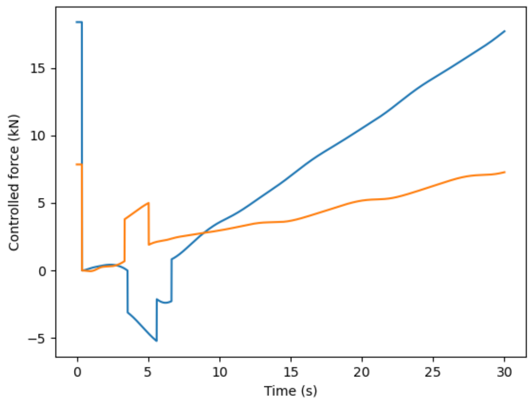
The depth control needs definite work, but is also capable of holding position. The case at 100m is shown in fig. 4.21. A comparison between a controlled depth and uncontrolled is shown in fig. 4.22.



(a) 0m waves

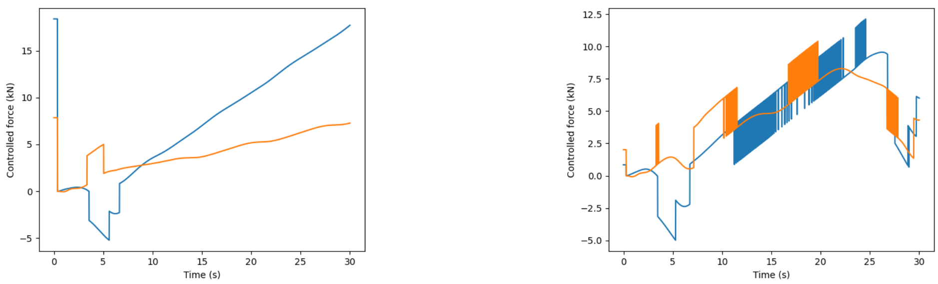


(b) 0.5m waves



(c) 1m waves

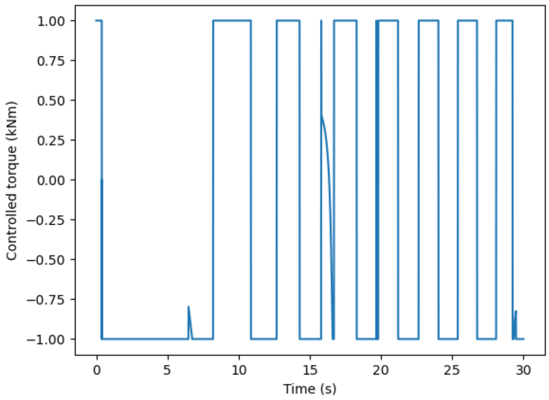
Figure 4.17: The USV's lateral forces applied during varying wave heights, ROV at 50m depth



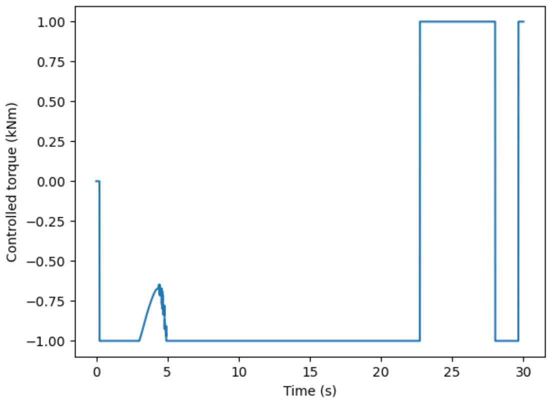
(a) ROV at 50m

(b) ROV at 100m

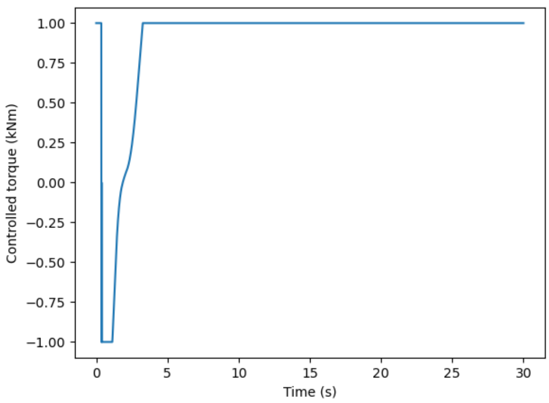
Figure 4.18: The lateral forces output by the controller at 1m wave height. The overall shape of the forces are similar, but the 100m case has many discontinuities while the 50m case does not.



(a) 0m waves

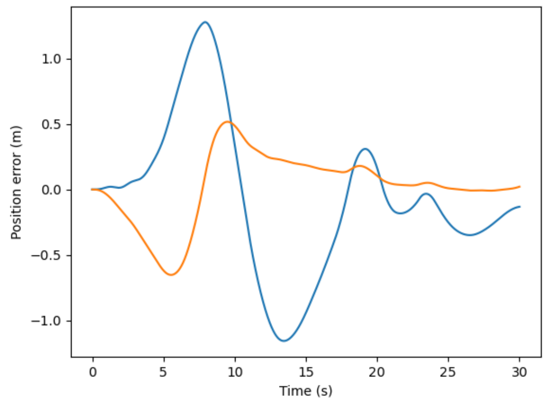


(b) 0.5m waves

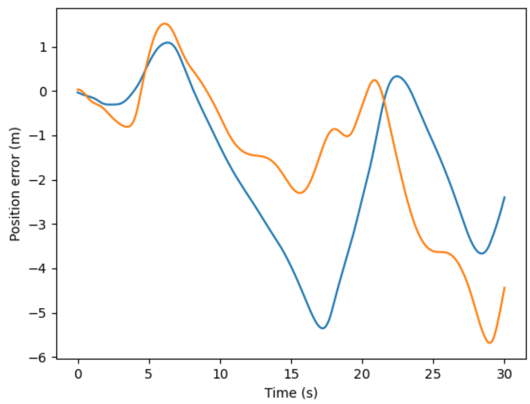


(c) 1m waves

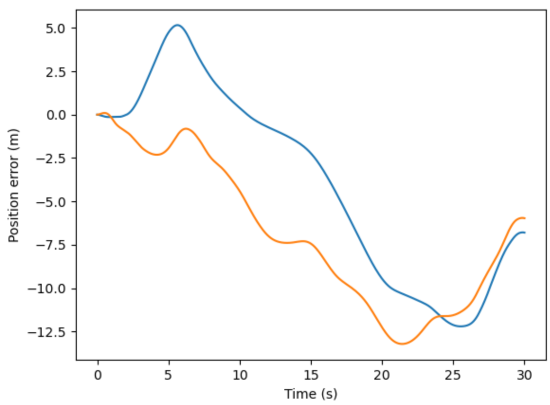
Figure 4.19: The USV's torque applied during varying wave heights, ROV at 50m depth. The system is capped at 1kNm of torque and is saturated at this level for most of the time



(a) 0m waves

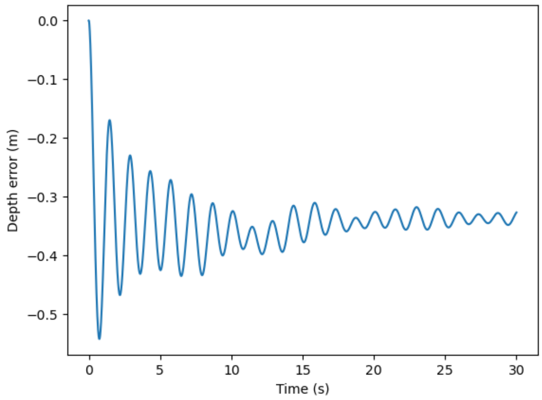


(b) 0.5m waves

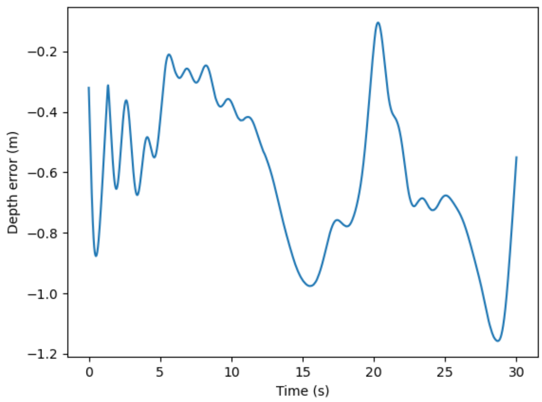


(c) 1m waves

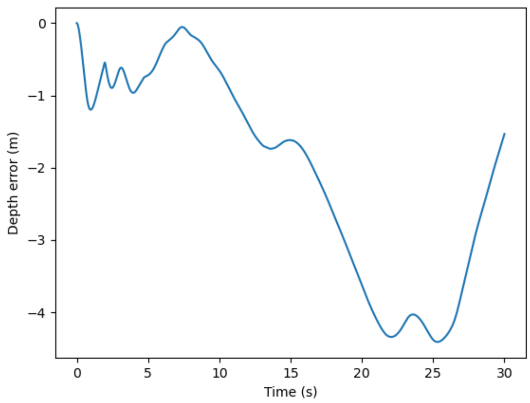
Figure 4.20: The ROV's positional error in the XY plane at 100m depth with different wave heights.



(a) 0m waves

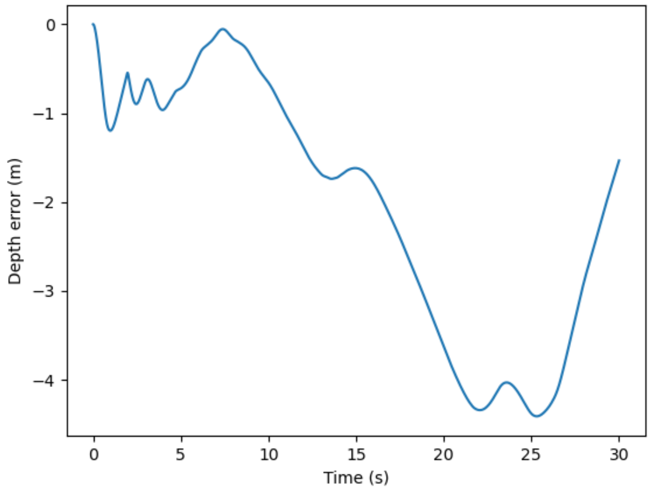


(b) 0.5m waves

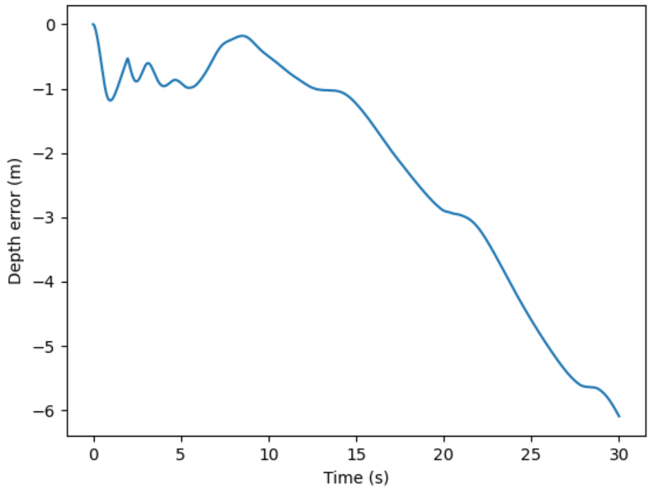


(c) 1m waves

Figure 4.21: The ROV's depth error at 100m depth with different wave heights.



(a) With control



(b) Without control

Figure 4.22: Comparison of ROV depth error at 100m depth and 1m wave height between a controlled and uncontrolled system. Note that the magnitude of error is almost twice as high for the uncontrolled system.

4.2 Simulator results

4.2.1 Validation

Validation of the simulator was primarily performed in the specialization project. Kjørseng (2024), section 4.2. There it was done as both an imperical measurement as well as a more general measurement. The specialization project found that the simulator performs as expected.

As a summary of the validation, the tension experienced by the tether between the ROV and the USV was both calculated using manual methods and simulated. The calculations and simulations were performed at 6 different speeds, between 0m/s and 5m/s in 1m/s increments. The ROV was observed in the simulations to drag behind the USV, since the USV is the powered part in this validation and the ROV is only hanging behind as a passive part. As the angle of the tether changes, the cross-sectional area of the ROV as well as the drag coefficient changes. To account for this, the calculated drag was calculated at two separate drag coefficients, one for a flat-facing cuboid and one for an edge-facing cuboid, respectively chosen from tables as 2.05 and 1.05. The forward facing area was assumed in calculations to be constant, however this is an obvious simplification and source of error.

The tension calculated on the wire was compared to the tension provided by the simulator, and was found to be within an acceptable range of error. The deviation between calculation and simulation was found to be between 3% and 55%, which considering the severe simplifications assumed makes the results of the simulation seem valid. Additionally, as could be expected, the deviation between calculation and simulation changes as the speed, and thus the drag coefficient and forward facing area, changes. At low speeds, the lower drag coefficient provides more accurate results, while at higher speeds the higher drag coefficient provides more accurate results. This is likely an effect of the forward facing area changing significantly, but is consistent with the expectation looking at the simulation, that the drag increases with speed. Accounting for this change in drag with speed, the deviation can be said to be between 3% and 20%, which is definitely within acceptable ranges for accuracy.

All these elements are discussed further and in greater detail in section 4 of the specialization project.

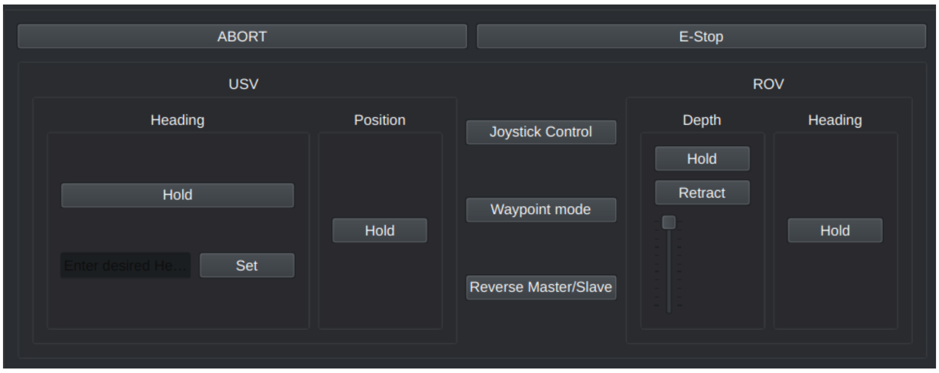


Figure 4.23: A mockup in code for a graphical user interface for the system

4.2.2 Developments of the simulation

A few developments have been made to the simulation itself. Primarily when it comes to ease of use. The largest change is the move from the controller being embedded in the simulator script to the simulator and controller being two different scripts. This has also allowed the previously monolithic "controller", which handled both position estimation, force calculation, the control loop itself and feedback, to be split into several smaller components that are easier to trouble-shoot, develop and exchange if necessary.

The new controller architecture has already been discussed somewhat and shown in fig. 3.1. Almost all the elements that are shown in the figure have been developed, but the allocator has been disabled due to issues with trouble-shooting it. The guidance system has also not been created but is implemented in a "distributed" way. What this means is that all the functions of the guidance system are in place, but they are implemented in duplicate across other nodes. This is a further development for future work.

Another development is a further refinement of the wave simulation. Previously it was a choice of waves on or off, but now the waves can be assigned a specific wave height. This allows for the kind of simulations that have been described above.

A graphical user interface (GUI) has also been mocked up using the framework QT. As it stands, it is not connected to the simulation, but implementing this should be fairly simple. The GUI-mockup is shown in fig. 4.23

CHAPTER

FIVE

DISCUSSION

There are three elements that will be discussed here: results and development of the simulator, results and development of the control system

5.1 Simulated cases

The goal of the simulated scenario is to show whether the simulator and the control system are suitable for the purpose of controlling the vessel. The results shown so far are promising but indicate that they are not.

As mentioned early in section 4.1 several more cases were simulated but their data has been discarded from this discussion. Originally the goal for simulation was to simulate only 0m, 1m and 2m waves, but this was expanded to include also 0.5m and 1.5m waves. This turned out to be a good thing, as the simulation does not do well with waves above 1m in height. There are two criteria that the system does not meet in higher waves. The first is vessel stability, the second is controllability.

When it comes to vessel stability, the vessel capsizes very easily at higher wave heights. No direct data has been gathered on this, but it is visible in the simulator viewport. See fig. 5.1 for an example of this. It's not unreasonable that the vessel struggles with large waves as it is fairly small. The length of the vessel implemented here is approximately 2.8m long and 3m wide. Comparing this size to a 2m wave, it makes intuitive sense that the vessel is unstable in those situations.

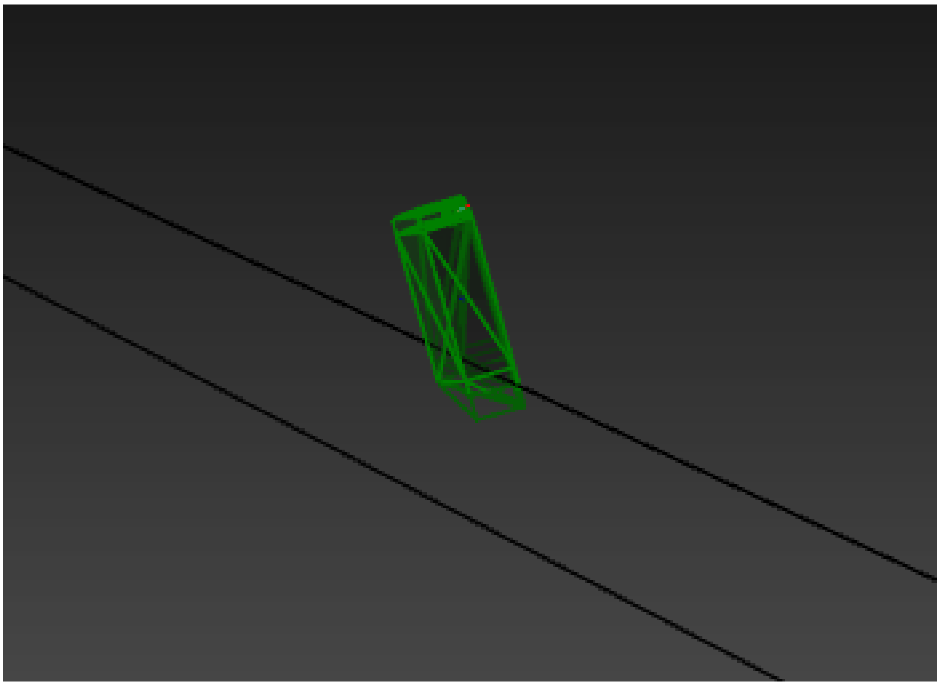


Figure 5.1: The vessel experiencing a 2m wave almost capsizing

Another possible contributor to this easy capsizing is that the vessel is currently simulated uniformly dense. This is unrealistic and a limitation of the current simulation. A real vessel is almost always heavier near the keel than near the deck, as this provides greater stability. For this vessel in particular, the hull and deck are made from lightweight carbon fiber boards, while heavy batteries and ballast will be placed low down in the hulls. This would make it so the large flat deck is nowhere near as dense as the lower hulls, and would hopefully shift the center of gravity down towards the keels and help with stability. It is possible to implement density gradients and non-uniformly dense materials in AGX, and this should be considered as a future addition.

With regards to controllability, the results (especially fig. 4.11) demonstrate how the vessel is not capable of controlling itself in the simulated scenario. It looks from the results like the power allowed for the vessel is too low, meaning that it is not able to withstand the external forces placed upon it. The vessel drifts even in relatively low waves, see fig. 4.21c. Another example of low authority is the torque curves, fig. 4.19. These reach saturation very quickly in most situations, meaning that they no longer are able to control the vessel any more than they already are. The issues with command authority stem

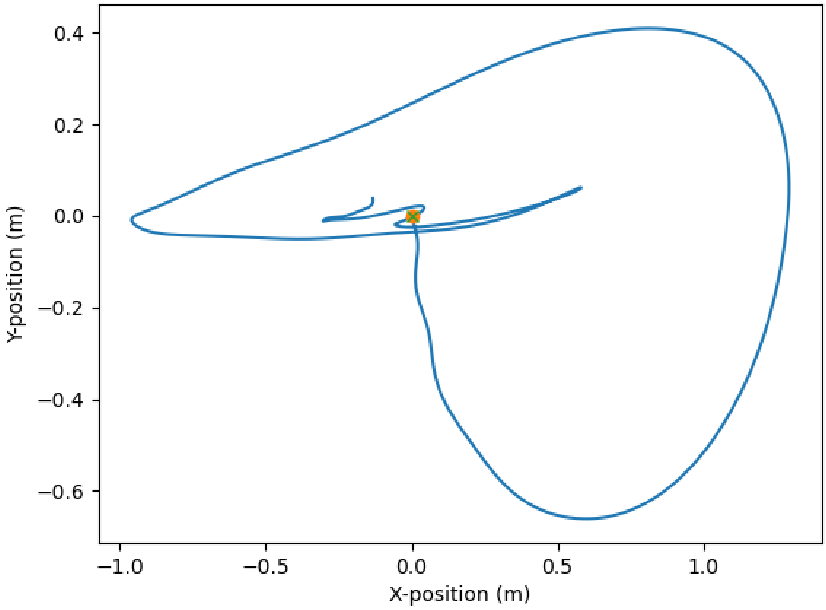


Figure 5.2: The position of the USV in no waves with ROV at 100m

from improper assumptions made about the vessel's capability, as well as the quite violent situations that the vessel is placed in.

Even discounting insufficient command authority, it is clear that the controller is still not functioning as desired. This can be seen in the wave-less cases, such as below in fig. 5.2. The vessel is moving around in these large circles. The same is seen in the 0.5m wave cases, though less clearly due to the drift of the vessel. This shows that the system is improperly tuned. The same tuning issues can be seen in the heading control results, where there is a constant oscillation about or near the setpoint. Luckily, these issues are easily solved with higher command authority and controller tuning.

5.1.1 On the subject of seastate

This simulation uses regular waves, meaning that the wave height at any point in space and time is defined by the wave height equation discussed in previous chapters. This is not realistic compared to a real-world seastate. Real-world seastates are random and complex. This makes them harder to simulate than regular waves, and also changes how they affect the vessel.

There is an argument for using a regular seastate in this sort of simulation. As the shape of the uncontrolled results shows, (fig. 4.5), the effect is predictable. The vessel in this current simulation setup will always move south-west when left adrift with wave forces. This is helpful for this kind of "stupid" (i.e. not a "learning") control system. Random effects won't be as prominent, and two different scenarios or control systems can be compared more easily. On the other hand, if the control system had machine learning implemented in some way, this regular seastate would be detrimental to the results. This is because the system would likely overfit to this specific kind of seastate and not be generalizable to other seas.

5.1.2 On the subject of mastering

One of the goals of this project has been to implement a reversibly mastered system, i.e. that both the surface vessel and the ROV can be the one "in charge" of deciding where to go. If the ROV is mastered, the USV attempts to follow, and the other way around. This has been implemented in the controllers for the USV and the ROV, but has not been tested in simulation.

The way the mastering works as it is implemented now is that if the vessel is mastered it recieves its target position from the operator, while if it's slaved it simply tries to match the position of the master vessel. In short: the system is currently mastered and reversibly mastered, but since there is no human-machine interface right now, this mastering has to be changed manually in code.

5.2 Development

One of the goals of moving from a bespoke or monolithic system architecture to one based on ROS2 was ease of development. The node based structure of a ROS2 system allows nodes to be swapped in or out, as well as to be developed and debugged in near isolation. From experience of developing this system now as well, it is possible to say that the node-based architecture allows for the ease of development that was expected. The decoupling and "node-ification" of the system has greatly improved its ease of further development. As a concrete example to this, the data collection node is a separate node from the rest of the control system but collects information from all of the other nodes. Implementing this might be complicated with a different system architecture, but was simply a matter of subscribing to the right topics here and that was that.

In the current implementation, the system provides the same information as a physical system would. The position and orientation of the USV could be given by a combination of dead-reckoning, GNSS systems and land-based location systems like RTK. For the ROV it could be through a sonically based positioning system, such as an ultra-short-baseline positioning system, dead-reckoning or by using seafloor landmarks for positioning. In this way, the simulator and a real-world system would be almost completely analogous.

The tool `rqt_graph` is able to plot the ROS graph for a system and outputs the following graphs where the rectangular boxes represent topics and the rounded bubbles are nodes. The graph system and without the data collection node is shown in fig. 5.3

5.2.1 Future developments

There is further work necessary on the development front. A functioning human-machine interface is high on the priority list. As shown in fig. 4.23, a non-functioning graphical user interface has been mocked up using the QT framework. Future work would include connecting the GUI to a chart plotter to allow an operator to easily input positions to the system.

While the results show that the control system is working, it is not working very well. The goal of this project has not been to fully develop the control system, but to create the development space to do further development. As it stands, further tuning of the controller parameters is necessary. It is also necessary to find more information on the physical vessel and its capabilities to set the control authority more appropriately. It would also be nice to implement more advanced control functions, such as predictive control based on historical data.

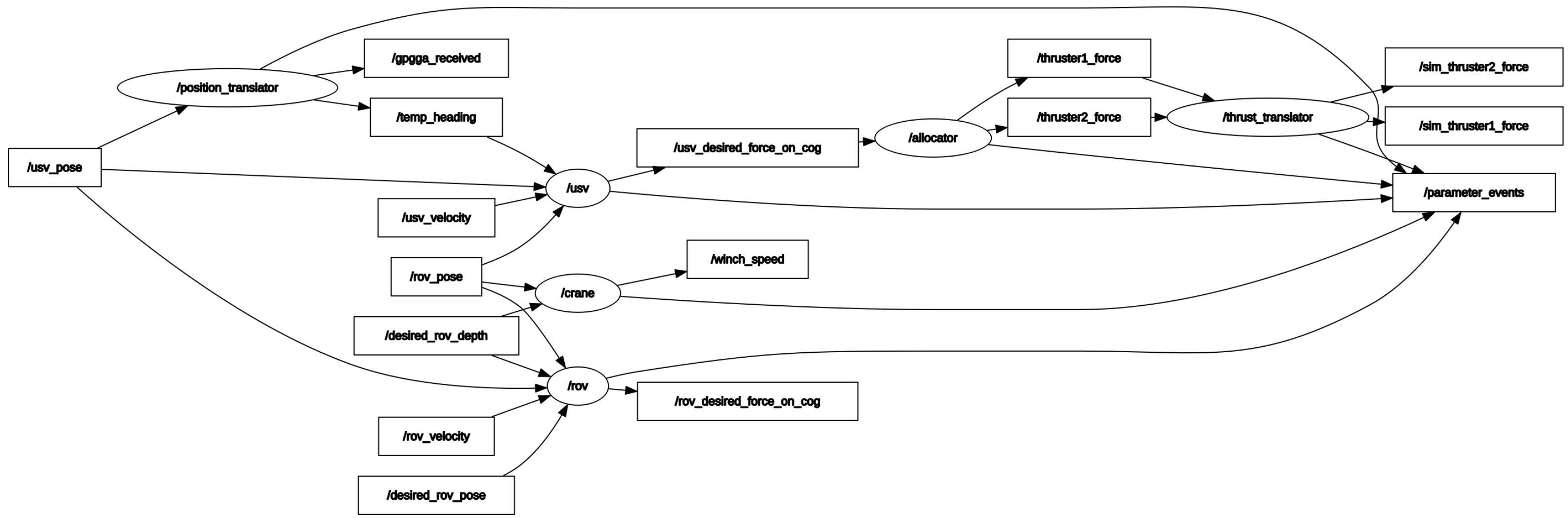


Figure 5.3: Graph of the ROS node network without the data collector

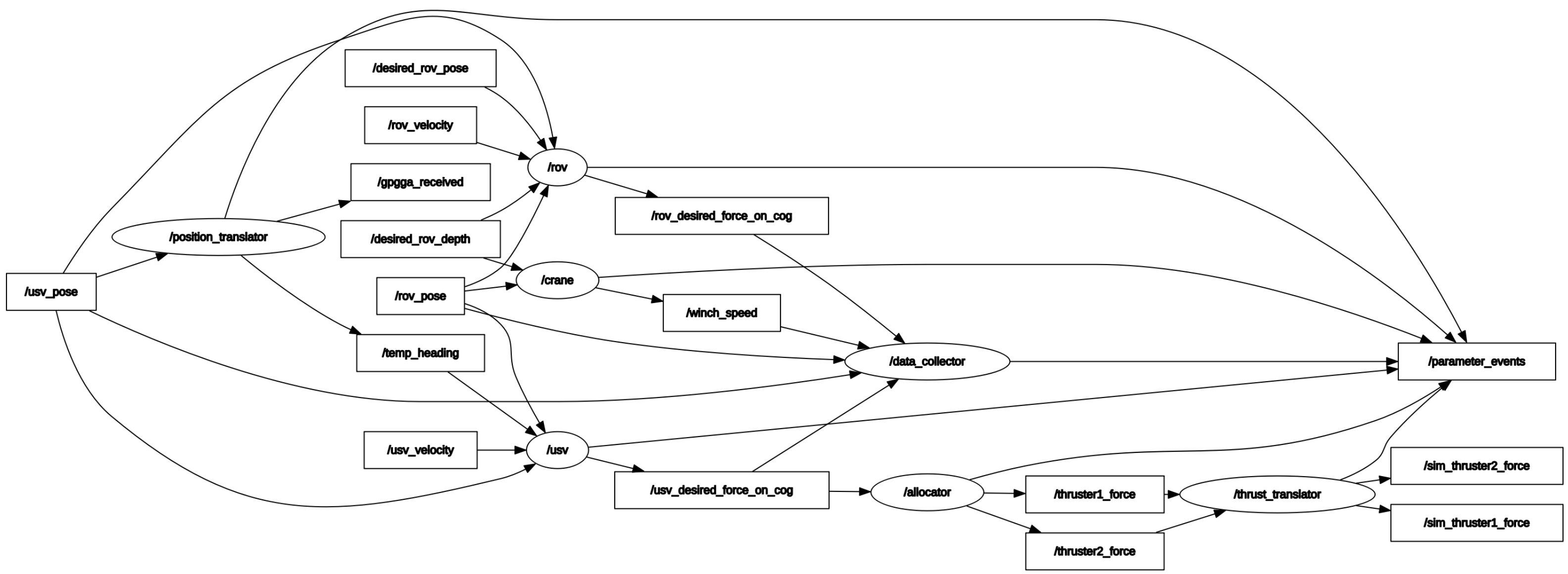


Figure 5.4: Graph of the ROS node network with data collection active

CHAPTER

SIX

CONCLUSIONS

Removing plastic from the ocean is a large and complex issue. The project undertaken for this thesis, as well as by Kjørseng (2024), will hopefully help somewhat with this issue. The goal of this project has been to continue the work on a coupled system consisting of an unmanned surface vessel and a non-buoyant remotely operated underwater vehicle. The specific work has been relating to further development of a control system and simulation for this coupled system.

This thesis has had four stated goals:

1. To reimplement the controller from the specialization project in the framework ROS2
2. To extend the controller to include winch and ROV control
3. To create a rough graphical user interface
4. To test the simulator in more situations than previously

The controller has been decoupled from the simulation framework, as it was in the beginning of this project. It is now implemented in a node-based structure using ROS 2. The node structure allows for quicker development and easier trouble-shooting of new modules. It also makes it so the control system is more or less agnostic as to whether it's connected to a simulator or a real-world vessel. Using this node based structure, the controller has also been extended to allow for control of the ROV's winch and the ROV itself.

A rough user interface has been mocked up, but is not currently functioning.

The reason why the simulator is desired in the first place is as a testing platform for the control system, as well as to help determine operational criteria for a potential mission. The simulator and controller together have been tested in one scenario. The scenario involves the surface vessel being placed in water with the ROV at varying depths and with simulated waves at different heights. This has been tested both with and without the control system enabled, to show the natural response of the simulator and simulated vessel, as well as the controller's effect.

It is clear from the simulated results that the controller is not well enough tuned. There are uncomfortable oscillations in the system. Additionally, assumptions made about the vessels are shown to be either wrong or impractical: the vessel is not able to maintain position in even the smallest simulated waves. Further work needs to be done to establish the capabilities of a real-world vessel and this needs to be implemented into the simulator before it can be used predictively.

Furthermore, the simulated results show two major limitations with the simulation as it is. Firstly, the simulator can be unstable if the wire is mishandled, leading to crashes. This is a limitation of the simulation framework and is not possible to avoid without changing the framework out. Secondly, the surface vessel is currently implemented as a uniformly dense hull, this has led to some instabilities in wavy seas. Further work should look into implementing non-uniform density distribution for the hull to more realistically simulate its response.

Some work has also been done to make the controller "reversibly mastered", what this means is that either of the coupled vessels can be the leading one and the other will dynamically follow. This has been roughly implemented, but would require further work with regards to how the mastering is decided and changed.

Other further work includes the graphical user interface and better controller tuning,

Overall, the project this thesis is based on has reached its goals satisfactorily. Further work is however necessary before the simulator here is usable as a proper analogue of a real-world vessel.

BIBLIOGRAPHY

FordypOgMaster/Simulator at recovery · MagnusKjorseng/FordypOgMaster. URL `https://github.com/MagnusKjorseng/FordypOgMaster/tree/recovery/Simulator`.

Atsuhiko Isobe and Shinsuke Iwasaki. The fate of missing ocean plastics: Are they just a marine environmental problem? *Science of The Total Environment*, 825:153935, June 2022. ISSN 0048-9697. doi: 10.1016/j.scitotenv.2022.153935. URL `https://www.sciencedirect.com/science/article/pii/S0048969722010270`.

Jenna R. Jambeck, Roland Geyer, Chris Wilcox, Theodore R. Siegler, Miriam Perryman, Anthony Andrady, Ramani Narayan, and Kara Lavender Law. Plastic waste inputs from land into the ocean. *Science*, 347(6223):768–771, February 2015. doi: 10.1126/science.1260352. URL `https://www.science.org/doi/10.1126/science.1260352`. Publisher: American Association for the Advancement of Science.

Magnus Kjørseng. Simulation of a coupled system con- sisting of a non-buoyant remotely op- erated vehicle and a surface vessel. Technical report, December 2024.

Tatsuro Matsuoka, Toshiko Nakashima, and Naoki Nagasawa. A review of ghost fishing: scientific approaches to evaluation and solutions. *Fisheries Science*, 71(4):691–702, July 2005. ISSN 1444-2906. doi: 10.1111/j.1444-2906.2005.01019.x. URL `https://doi.org/10.1111/j.1444-2906.2005.01019.x`.

Johanna R. Rochester. Bisphenol A and human health: A review of the literature. *Reproductive Toxicology*, 42:132–155, December 2013. ISSN 0890-6238. doi: 10.1016/j.reprotox.2013.08.008. URL `https://www.sciencedirect.com/science/article/pii/S0890623813003456`.

Mariana Segovia-Mendoza, Karen E. Nava-Castro, Margarita I. Palacios-Arreola, Claudia Garay-Canales, and Jorge Morales-Montor. How microplastic components influence the immune system and impact on children health: Focus on cancer. *Birth Defects Research*, 112(17):1341–1361, 2020. ISSN 2472-1727. doi: 10.1002/bdr2.1779. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/bdr2.1779. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/bdr2.1779.

Lisbet Sørensen, Tânia Gomes, Amaia Igartua, Inger Larsen Lyngstad, Ana Catarina Almeida, Martin Wagner, and Andy M. Booth. Organic chemicals associated with rubber are more toxic to marine algae and bacteria than those of thermoplastics. *Journal of Hazardous Materials*, 458:131810, September 2023. ISSN 0304-3894. doi: 10.1016/j.jhazmat.2023.131810. URL https://www.sciencedirect.com/science/article/pii/S0304389423010932.

Elin Sørhus, Lisbet Sørensen, Bjørn Einar Grøsvik, Jérémie Le Goff, John P. Incardona, Tiffany L. Linbo, David H. Baldwin, Ørjan Karlsen, Trond Nordtug, Bjørn Henrik Hansen, Anders Thorsen, Carey E. Donald, Terje van der Meeren, William Robson, Steven J. Rowland, Josef D. Rasinger, Frode B. Vikebø, and Sonnich Meier. Crude oil exposure of early life stages of Atlantic haddock suggests threshold levels for developmental toxicity as low as 0.1 mcg total polyaromatic hydrocarbon (TPAH)/L. *Marine Pollution Bulletin*, 190:114843, May 2023. ISSN 0025-326X. doi: 10.1016/j.marpolbul.2023.114843. URL https://www.sciencedirect.com/science/article/pii/S0025326X23002746.

World Economic Forum. Top 25 recycling facts and statistics for 2022, June 2022. URL https://www.weforum.org/stories/2022/06/recycling-global-statistics-facts-plastic-paper/.

Natalia Zolotova, Anna Kosyreva, Dzhuliia Dzhalilova, Nikolai Fokichev, and Olga Makarova. Harmful effects of the microplastic pollution on animal health: a literature review. *PeerJ*, 10:e13503, June 2022. ISSN 2167-8359. doi: 10.7717/peerj.13503. URL https://peerj.com/articles/13503. Publisher: PeerJ Inc.

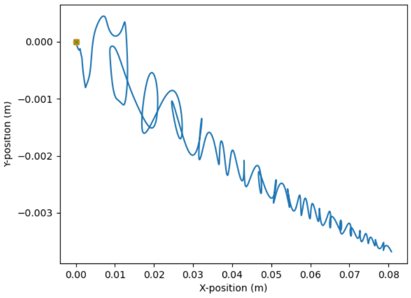
APPENDIX

ONE

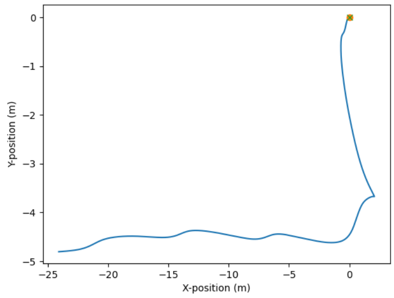
ALL SIMULATION RESULTS

A.1 USV position

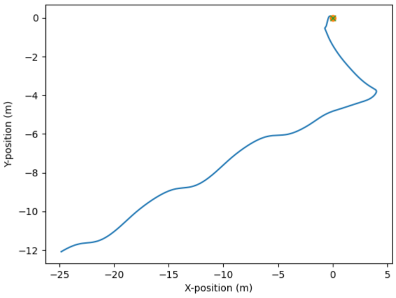
A.1.1 Uncontrolled



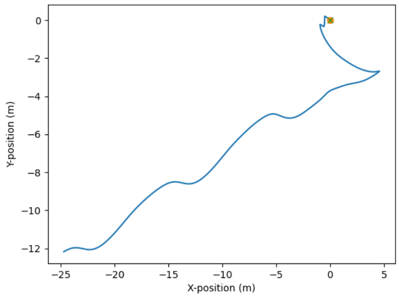
(a) 0m



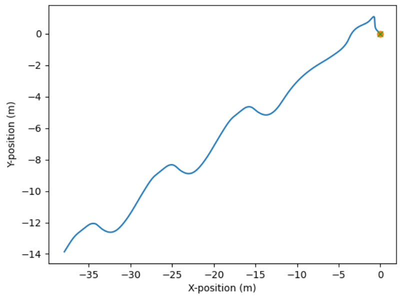
(b) 0.5m



(c) 1m

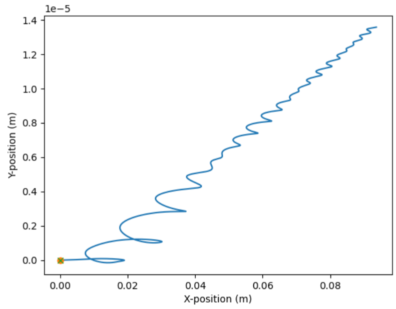


(d) 1.5m

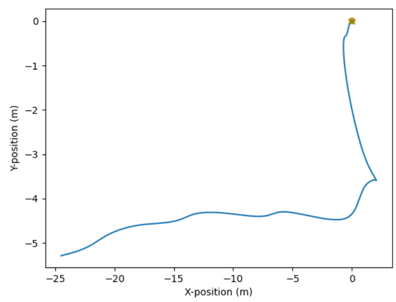


(e) 2m

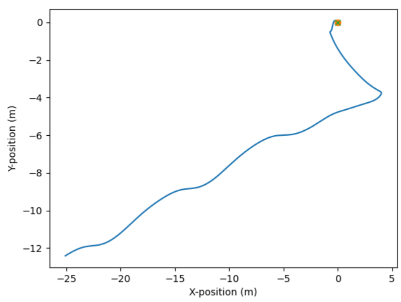
Figure A.1: USV position without control at different wave heights, ROV at 0m



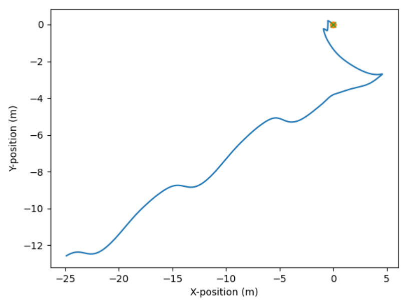
(a) 0m



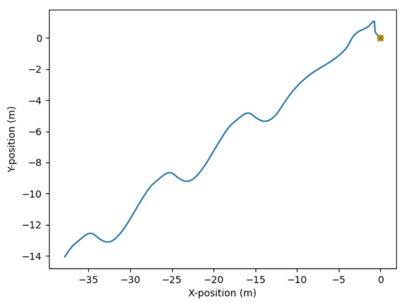
(b) 0.5m



(c) 1m

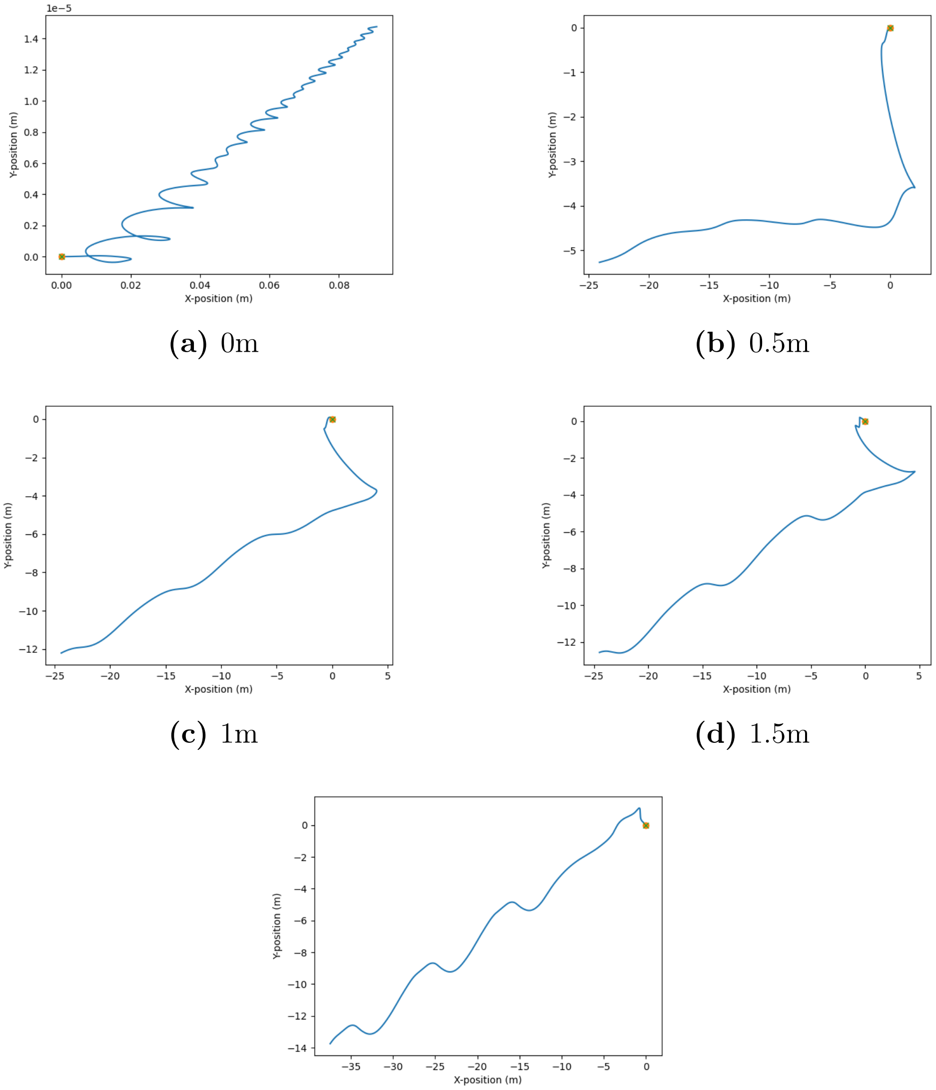


(d) 1.5m



(e) 2m

Figure A.2: USV position without control at different wave heights, ROV at 50m



(a) 0m

(b) 0.5m

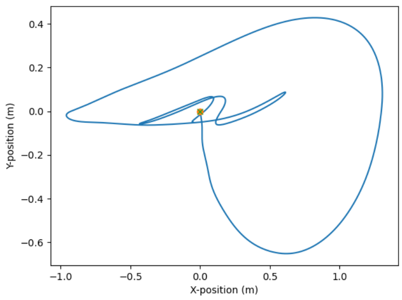
(c) 1m

(d) 1.5m

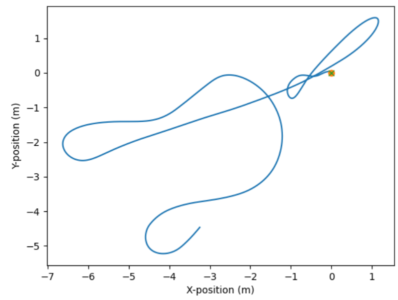
(e) 2m

Figure A.3: USV position without control at different wave heights, ROV at 100m

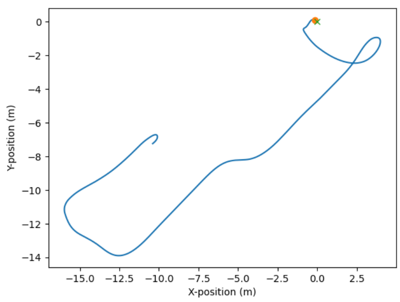
A.1.2 Controlled



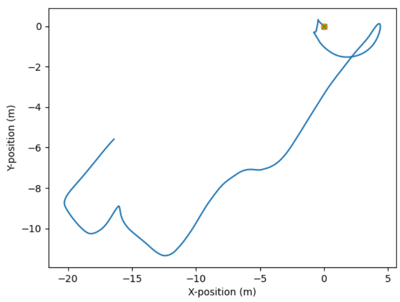
(a) 0m



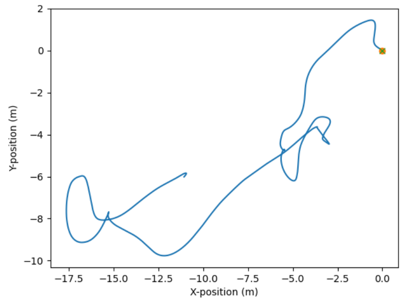
(b) 0.5m



(c) 1m

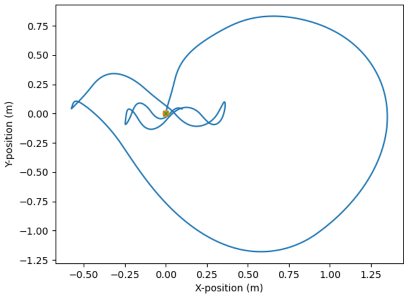


(d) 1.5m

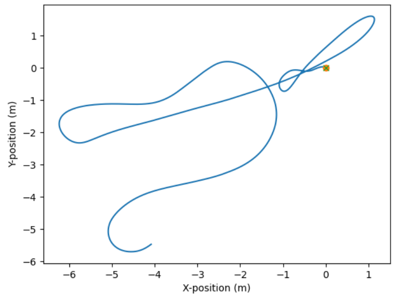


(e) 2m

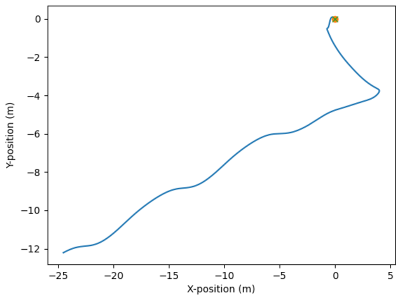
Figure A.4: USV position with control at different wave heights, ROV at 0m



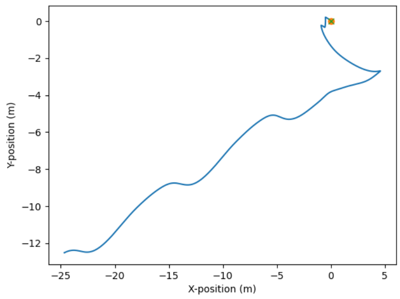
(a) 0m



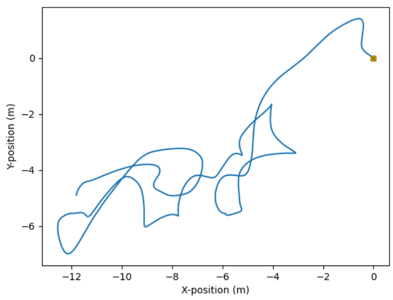
(b) 0.5m



(c) 1m

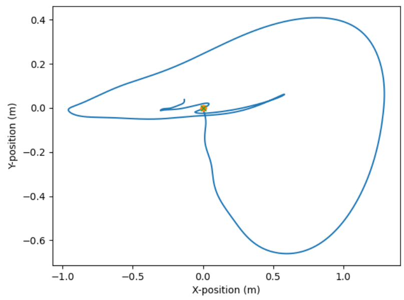


(d) 1.5m

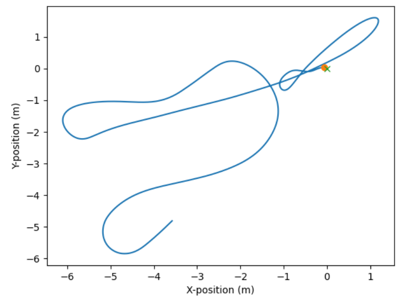


(e) 2m

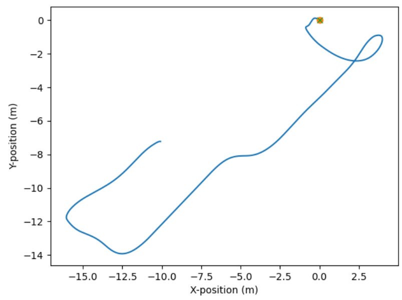
Figure A.5: USV position with control at different wave heights, ROV at 50m



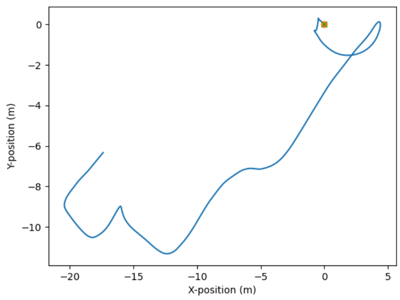
(a) 0m



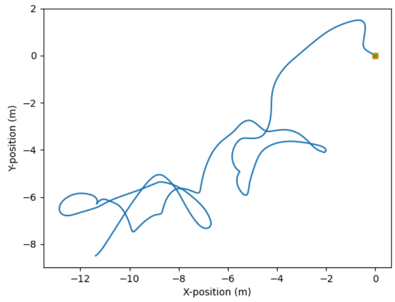
(b) 0.5m



(c) 1m



(d) 1.5m



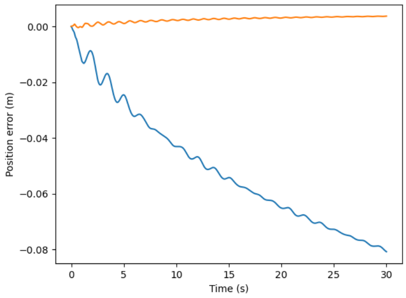
(e) 2m

Figure A.6: USV position with control at different wave heights, ROV at 100m

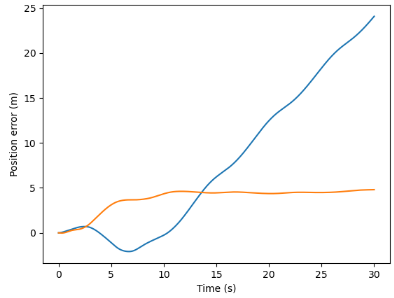
A.2 USV position errors over time

For the following figures the X- and Y- components of the movement of the USV are shown as blue and yellow respectively.

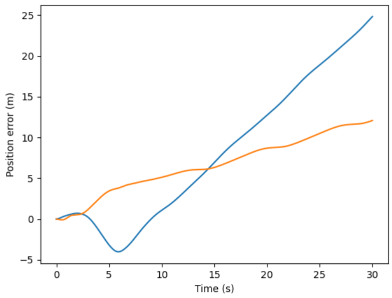
A.2.1 Uncontrolled



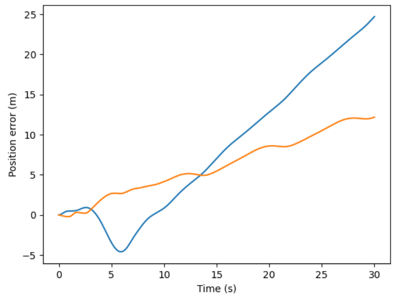
(a) 0m



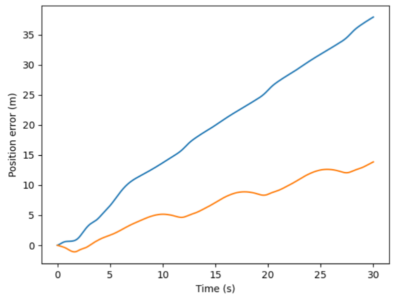
(b) 0.5m



(c) 1m

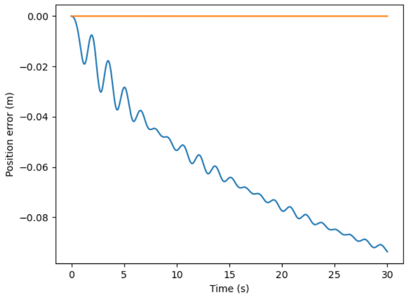


(d) 1.5m

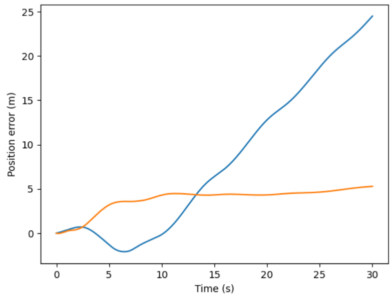


(e) 2m

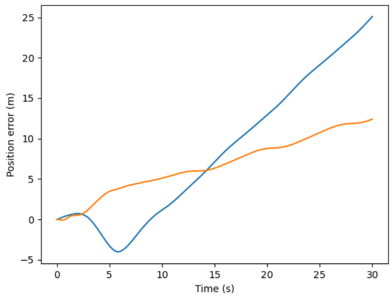
Figure A.7: USV position error without control at different wave heights, X-direction as blue, Y-direction as yellow. ROV at 0m



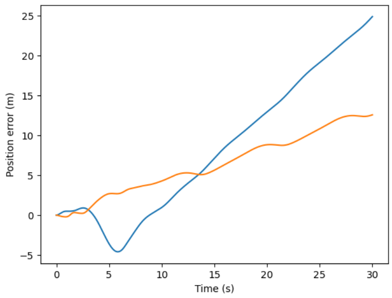
(a) 0m



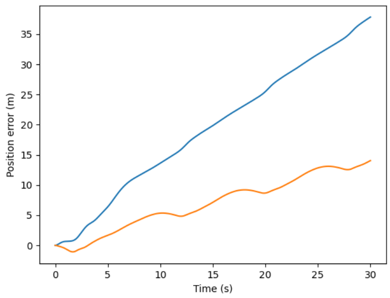
(b) 0.5m



(c) 1m

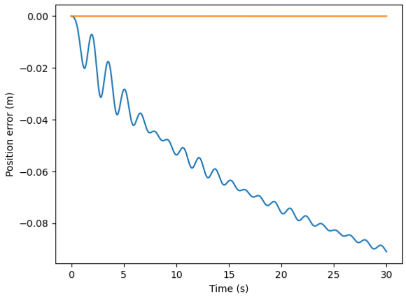


(d) 1.5m

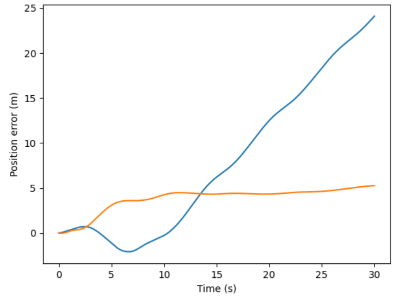


(e) 2m

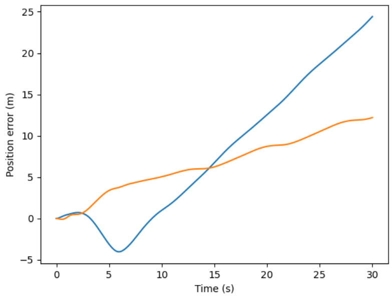
Figure A.8: USV position error without control at different wave heights, X-direction as blue, Y-direction as yellow. ROV at 50m



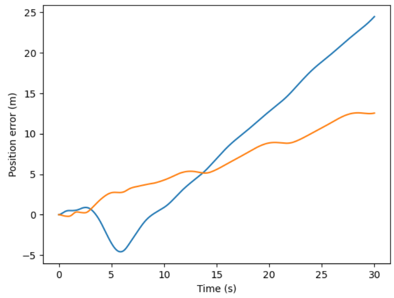
(a) 0m



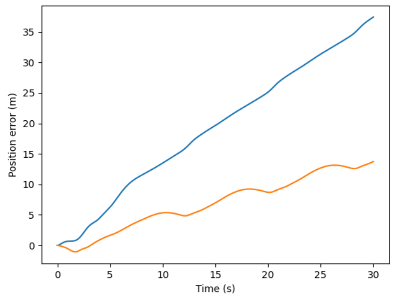
(b) 0.5m



(c) 1m



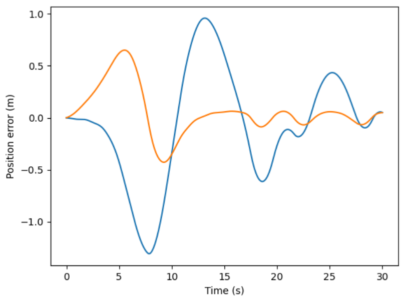
(d) 1.5m



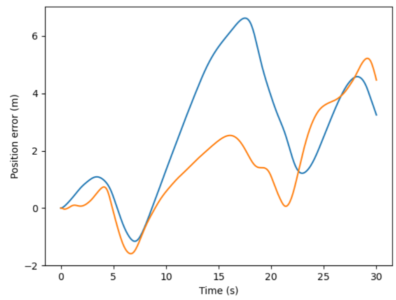
(e) 2m

Figure A.9: USV position error without control at different wave heights, X-direction as blue, Y-direction as yellow. ROV at 100m

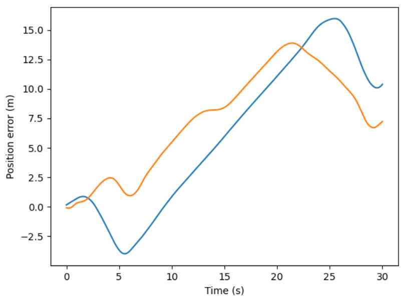
A.2.2 Controlled



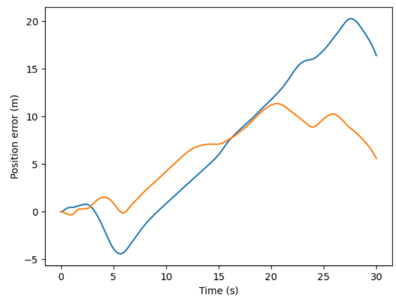
(a) 0m



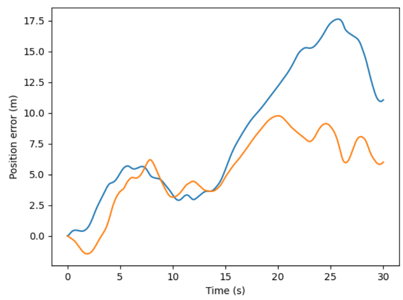
(b) 0.5m



(c) 1m

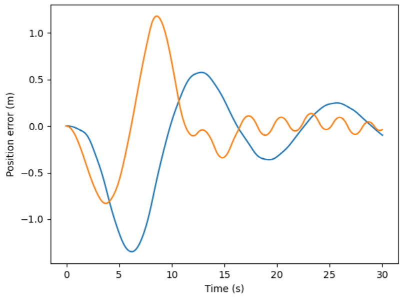


(d) 1.5m

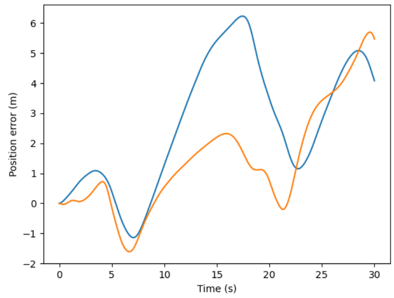


(e) 2m

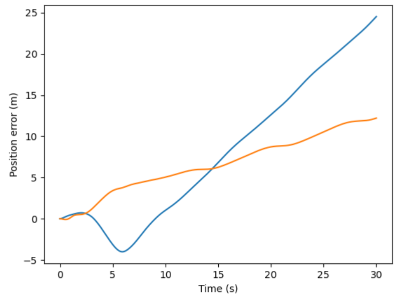
Figure A.10: USV position error with control at different wave heights, X-direction as blue, Y-direction as yellow. ROV at 0m



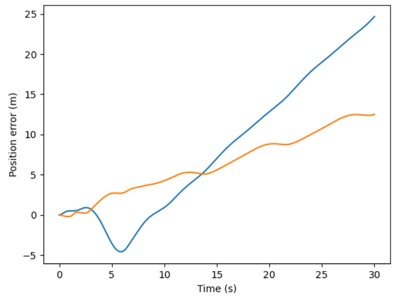
(a) 0m



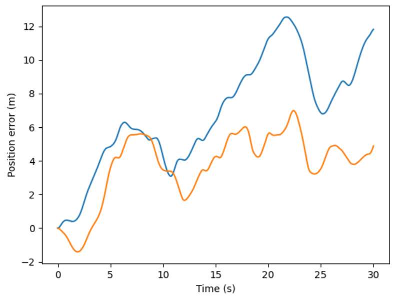
(b) 0.5m



(c) 1m

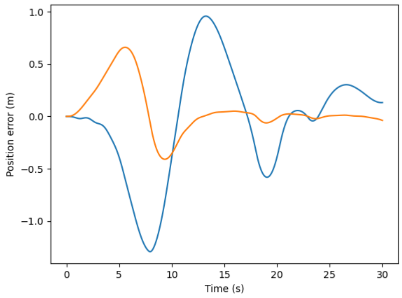


(d) 1.5m

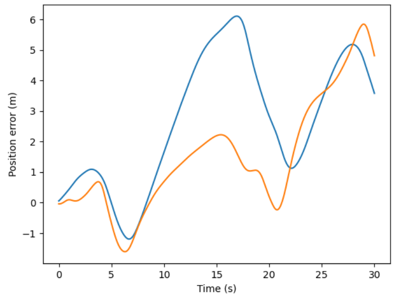


(e) 2m

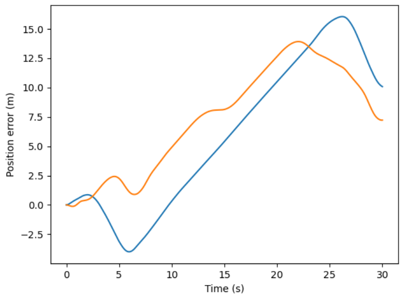
Figure A.11: USV position error with control at different wave heights, X-direction as blue, Y-direction as yellow. ROV at 50m



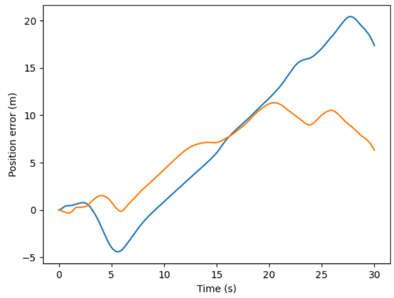
(a) 0m



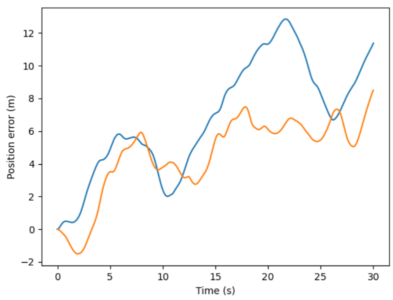
(b) 0.5m



(c) 1m



(d) 1.5m

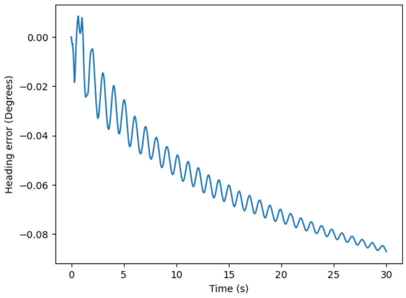


(e) 2m

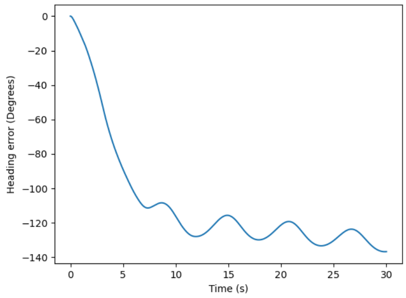
Figure A.12: USV position error with control at different wave heights, X-direction as blue, Y-direction as yellow. ROV at 100m

A.3 USV heading error

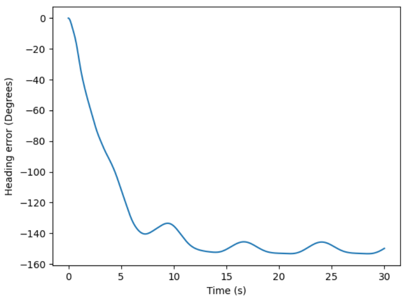
A.3.1 Uncontrolled



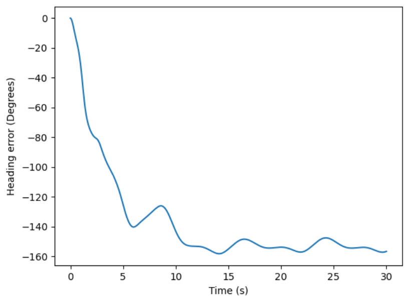
(a) 0m



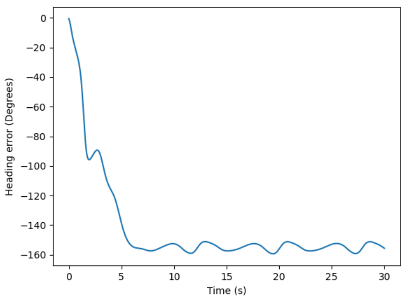
(b) 0.5m



(c) 1m

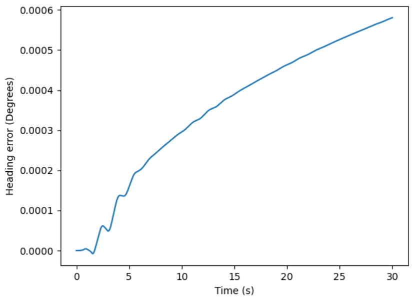


(d) 1.5m

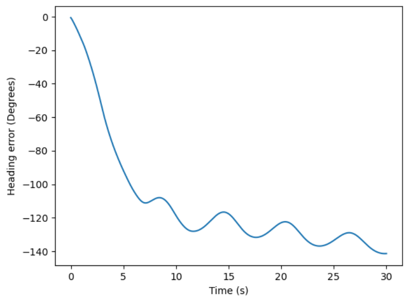


(e) 2m

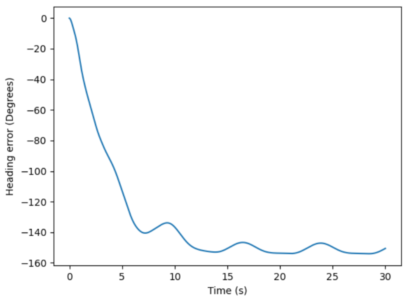
Figure A.13: USV heading error at different wave heights, ROV at 0m, without control



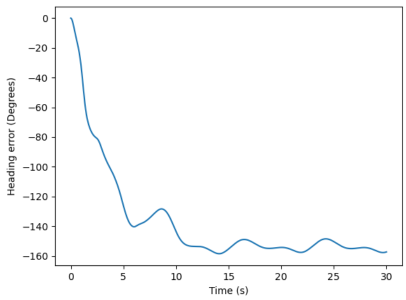
(a) 0m



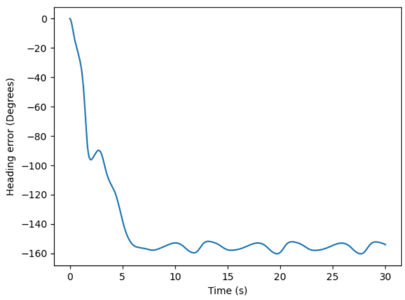
(b) 0.5m



(c) 1m

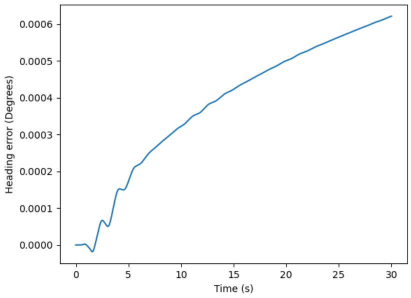


(d) 1.5m

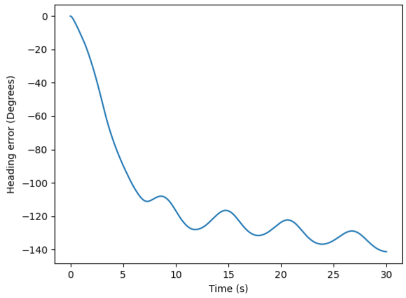


(e) 2m

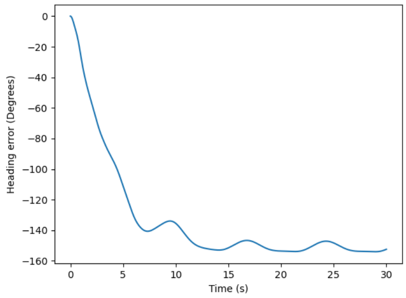
Figure A.14: USV heading error at different wave heights, ROV at 50m, without control



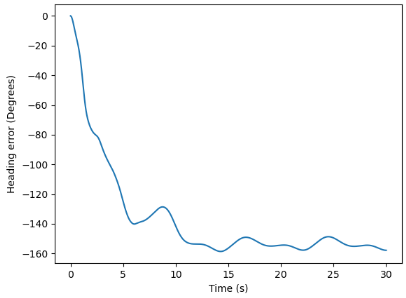
(a) 0m



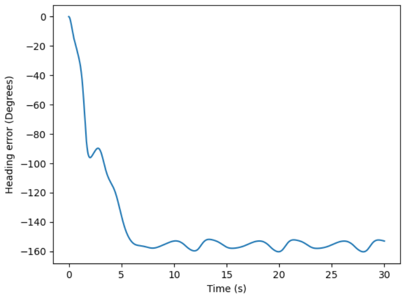
(b) 0.5m



(c) 1m



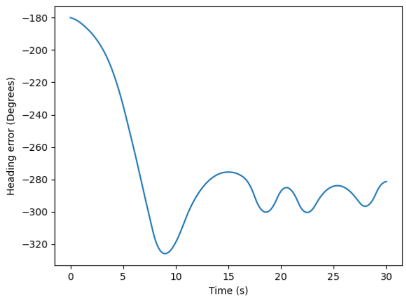
(d) 1.5m



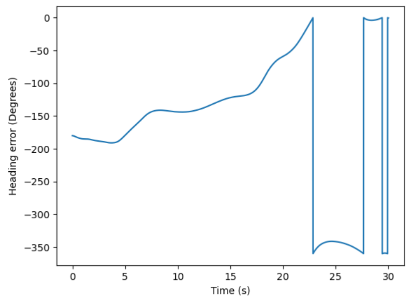
(e) 2m

Figure A.15: USV heading error at different wave heights, ROV at 100m, without control

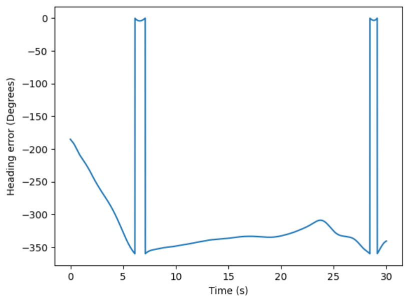
A.3.2 Controlled



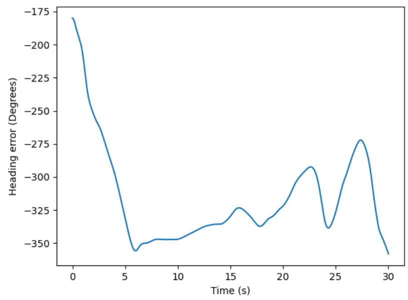
(a) 0m



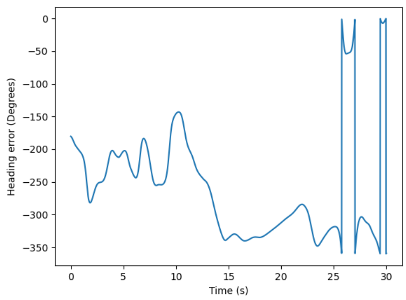
(b) 0.5m



(c) 1m

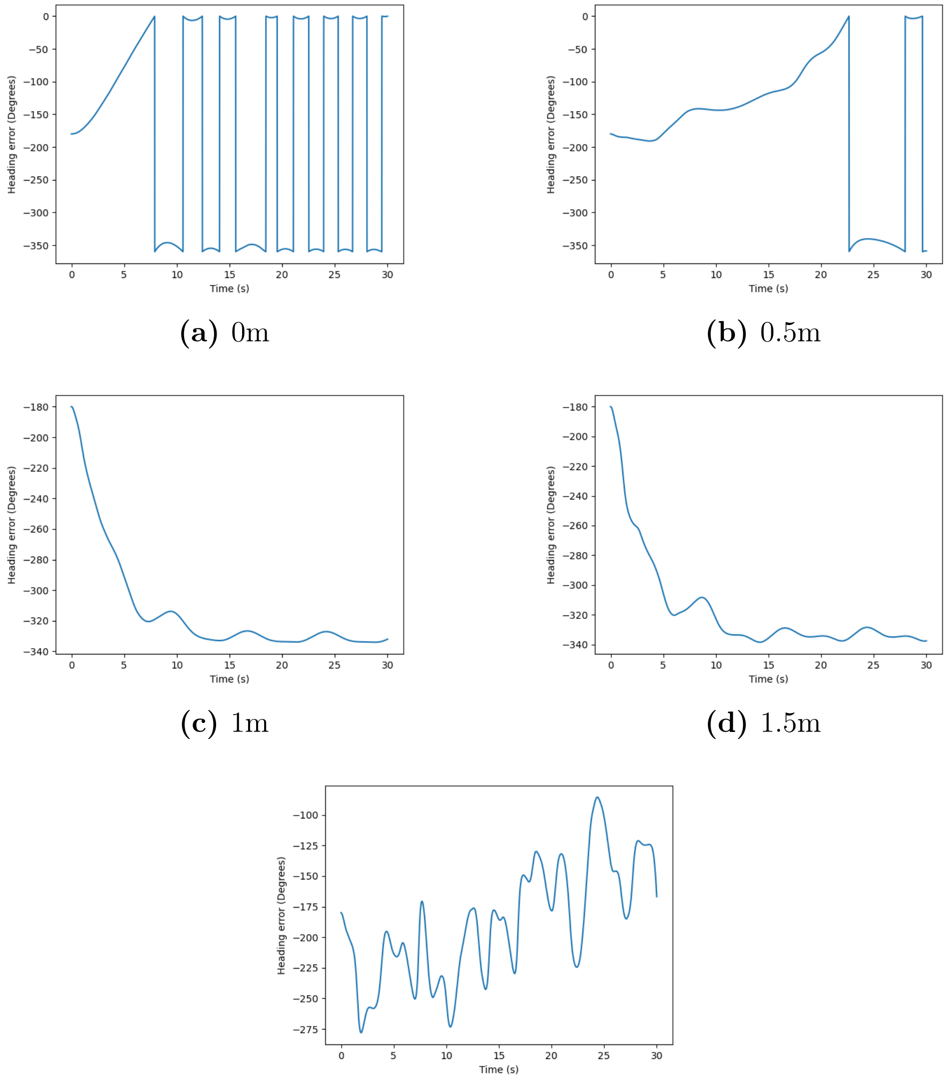


(d) 1.5m



(e) 2m

Figure A.16: USV heading error at different wave heights, ROV at 0m, with control



(a) 0m

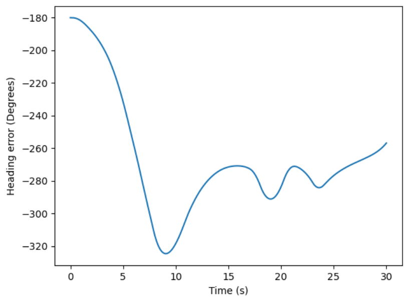
(b) 0.5m

(c) 1m

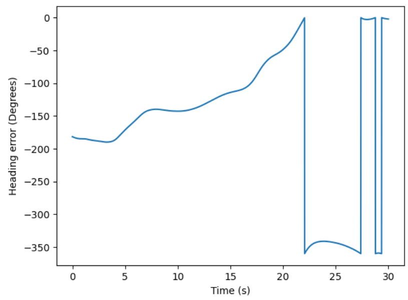
(d) 1.5m

(e) 2m

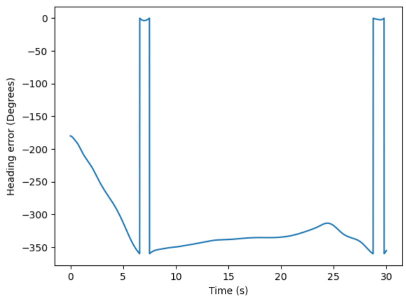
Figure A.17: USV heading error at different wave heights, ROV at 50m, with control



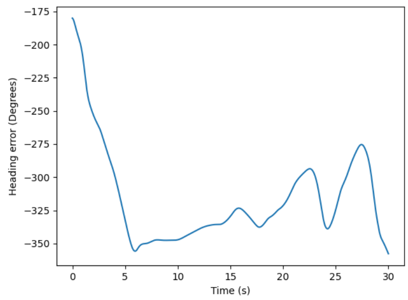
(a) 0m



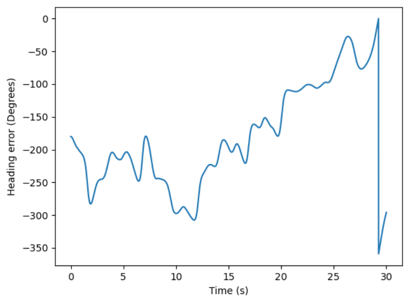
(b) 0.5m



(c) 1m



(d) 1.5m

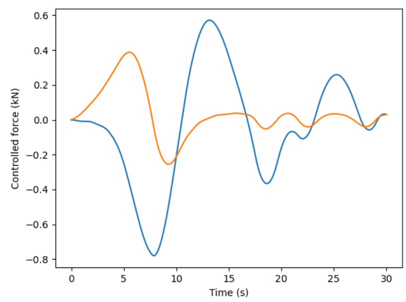


(e) 2m

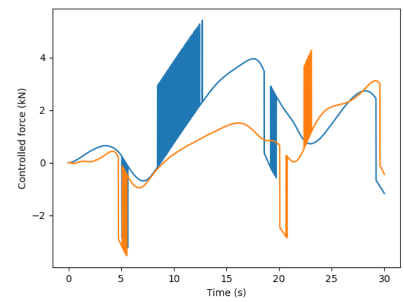
Figure A.18: USV heading error at different wave heights, ROV at 100m, with control

A.4 USV applied forces

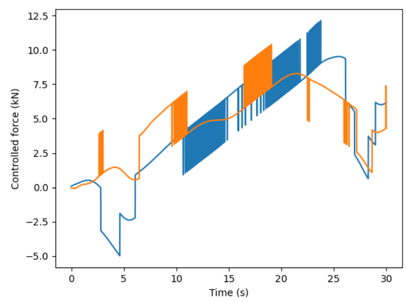
A.4.1 Lateral forces



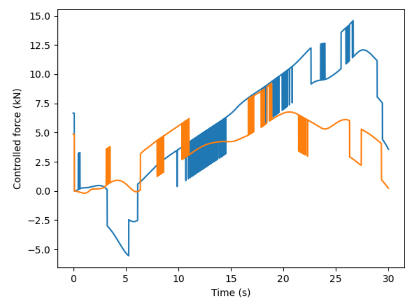
(a) 0m



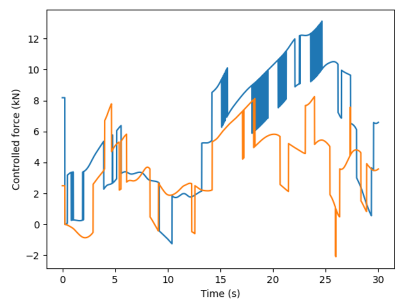
(b) 0.5m



(c) 1m

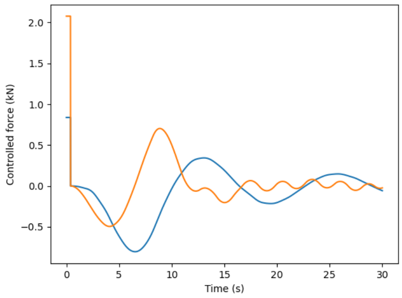


(d) 1.5m

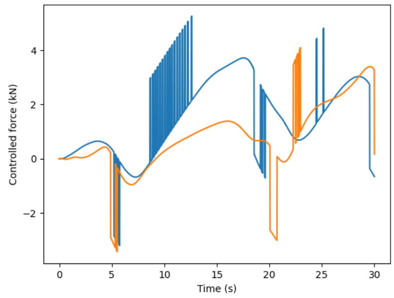


(e) 2m

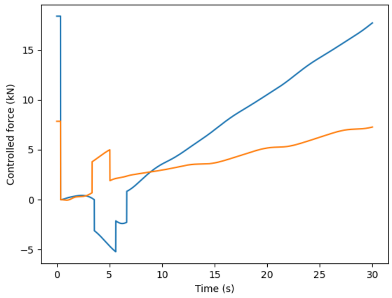
Figure A.19: USV lateral forces applied by controller at different wave heights, ROV at 0m



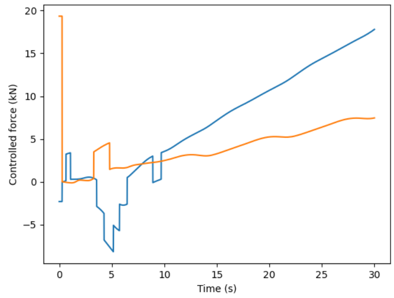
(a) 0m



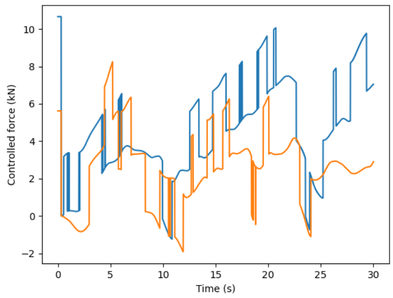
(b) 0.5m



(c) 1m

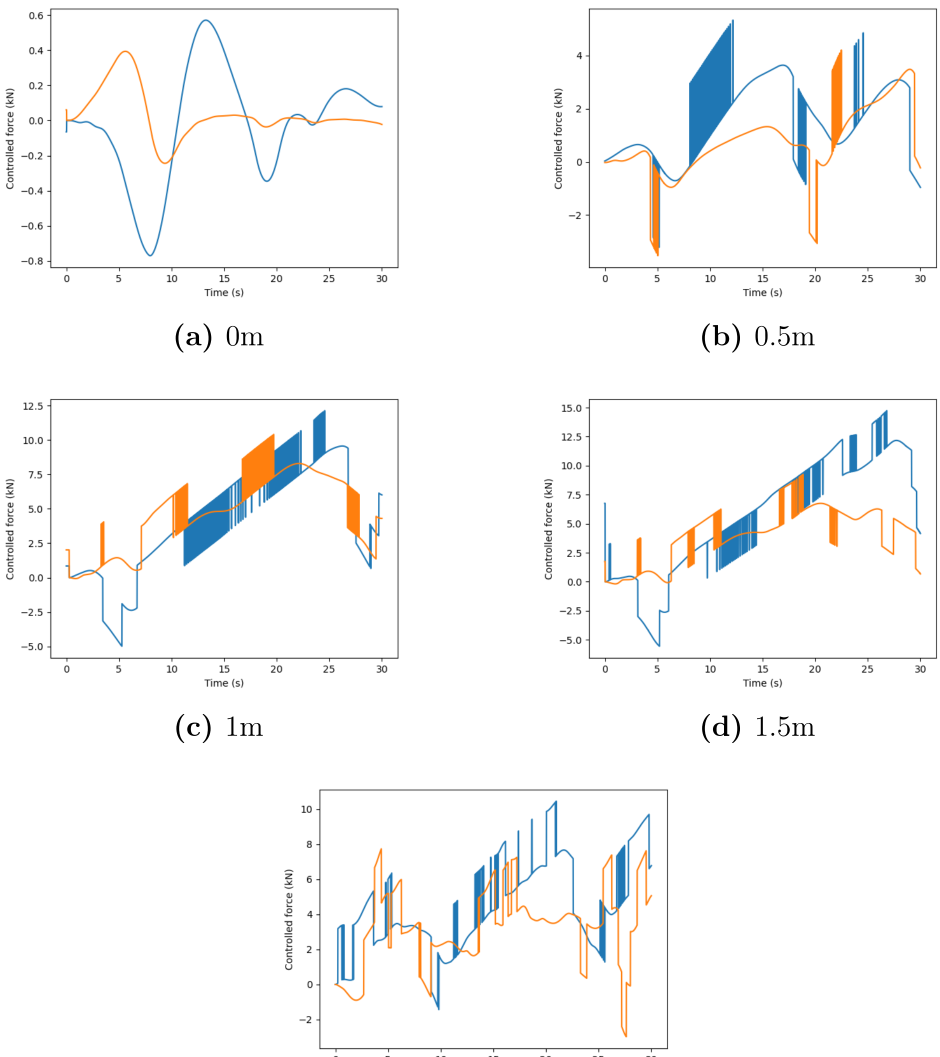


(d) 1.5m



(e) 2m

Figure A.20: USV lateral forces applied by controller at different wave heights, ROV at 50m



(a) 0m

(b) 0.5m

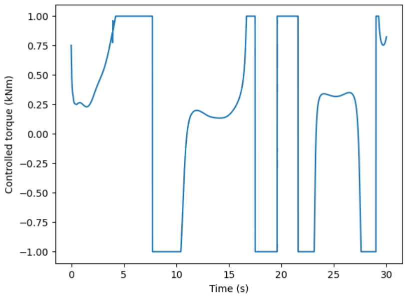
(c) 1m

(d) 1.5m

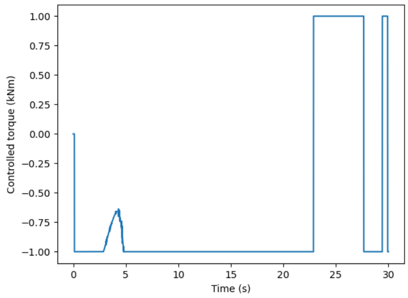
(e) 2m

Figure A.21: USV lateral forces applied by controller at different wave heights, ROV at 100m

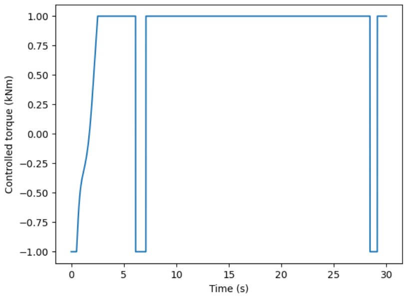
A.4.2 Torque



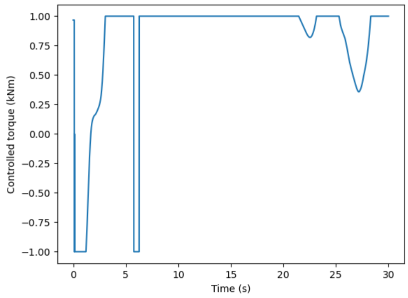
(a) 0m



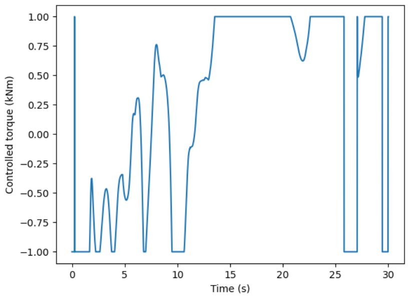
(b) 0.5m



(c) 1m

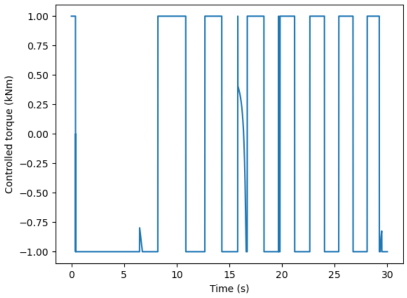


(d) 1.5m

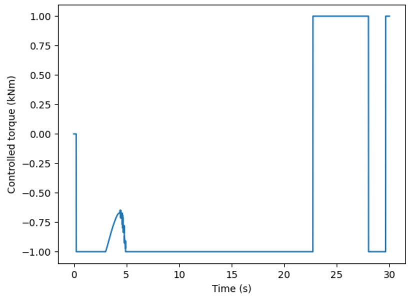


(e) 2m

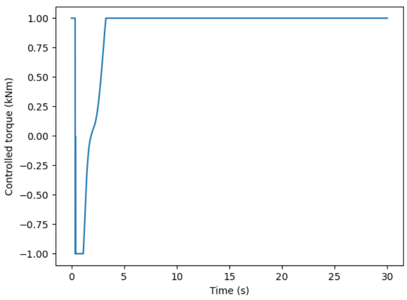
Figure A.22: USV torque applied by controller at different wave heights, ROV at 0m



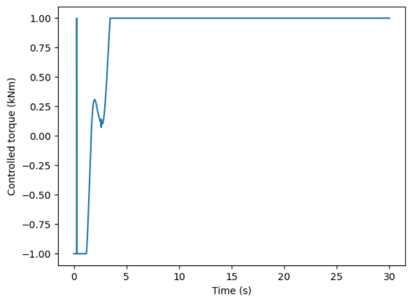
(a) 0m



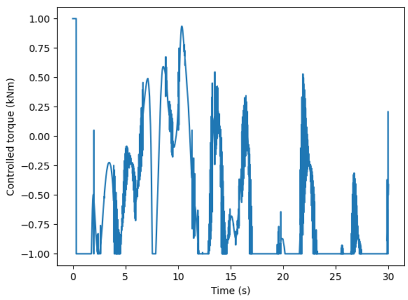
(b) 0.5m



(c) 1m

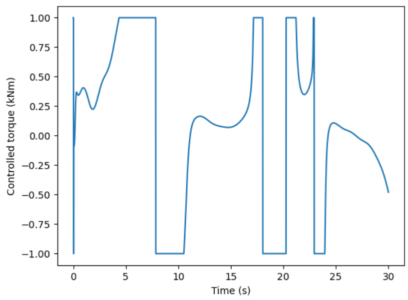


(d) 1.5m

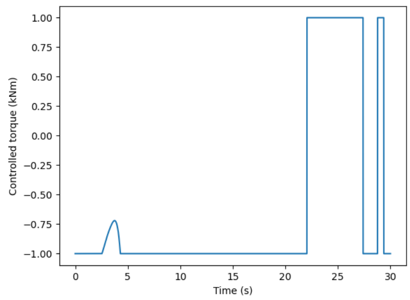


(e) 2m

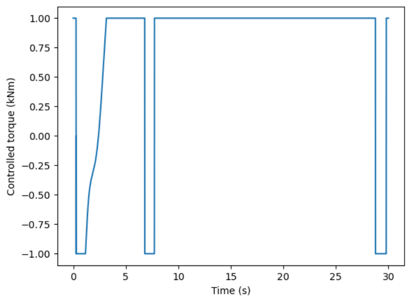
Figure A.23: USV torque applied by controller at different wave heights, ROV at 50m



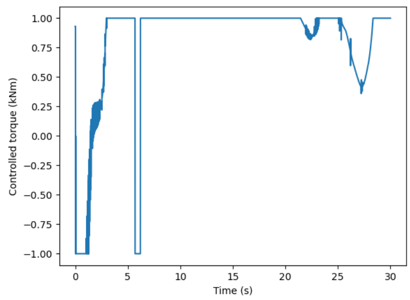
(a) 0m



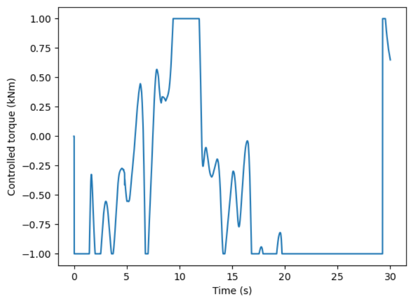
(b) 0.5m



(c) 1m



(d) 1.5m

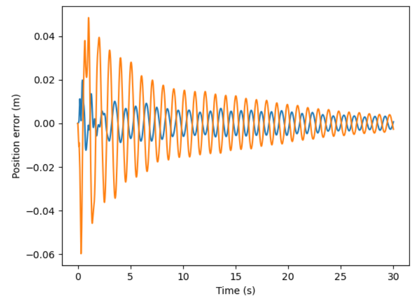


(e) 2m

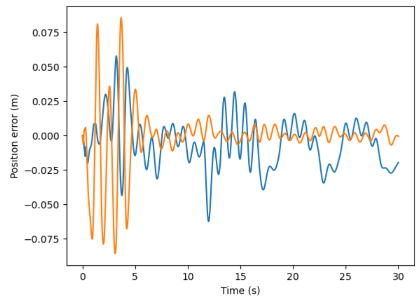
Figure A.24: USV torque applied by controller at different wave heights, ROV at 100m

A.5 ROV position error over time

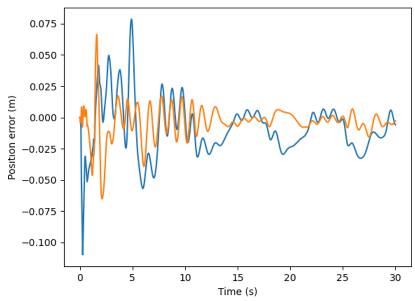
A.5.1 Uncontrolled



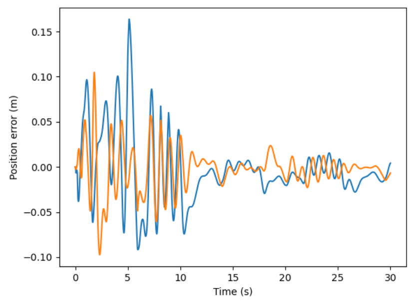
(a) 0m



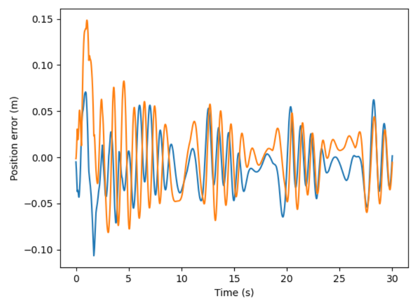
(b) 0.5m



(c) 1m

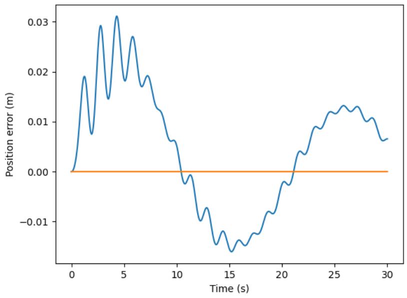


(d) 1.5m

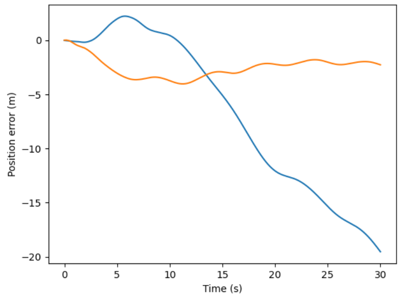


(e) 2m

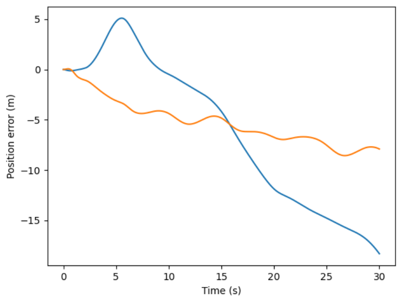
Figure A.25: ROV position error without control at different wave heights, ROV at 0m



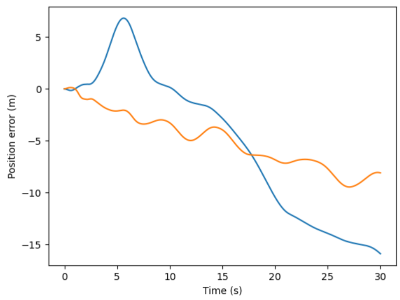
(a) 0m



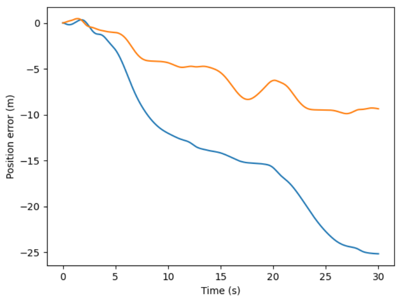
(b) 0.5m



(c) 1m

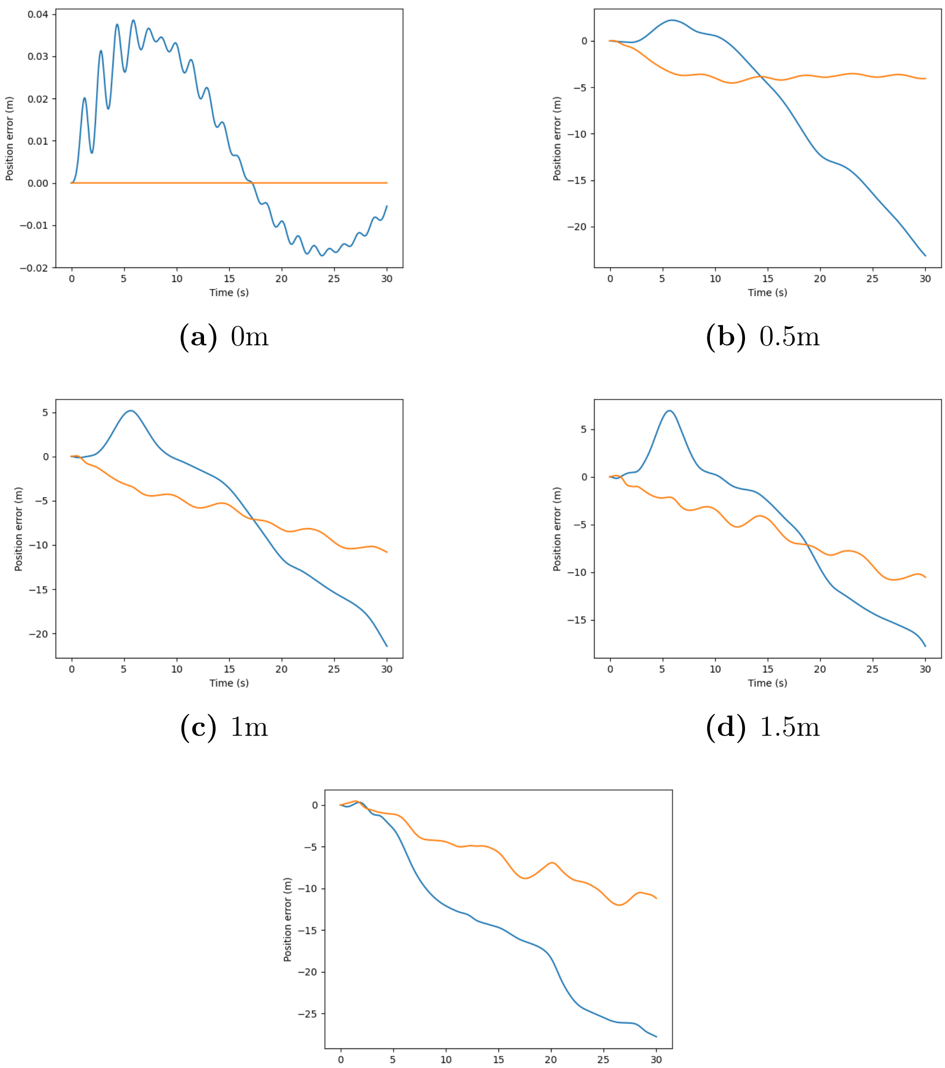


(d) 1.5m



(e) 2m

Figure A.26: ROV position error without control at different wave heights, ROV at 50m



(a) 0m

(b) 0.5m

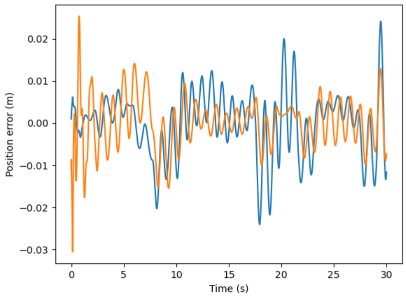
(c) 1m

(d) 1.5m

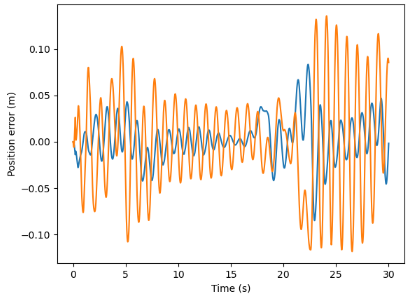
(e) 2m

Figure A.27: ROV position error without control at different wave heights, ROV at 100m

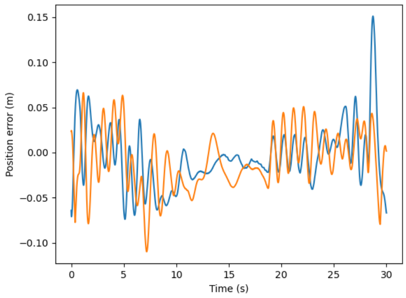
A.5.2 Controlled



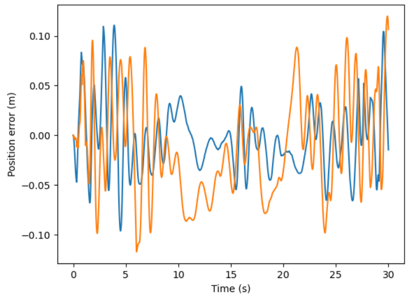
(a) 0m



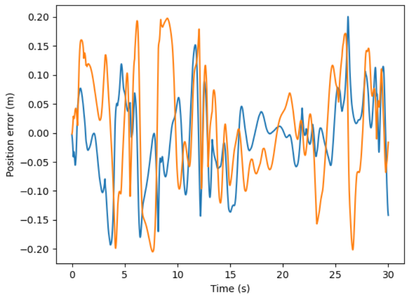
(b) 0.5m



(c) 1m

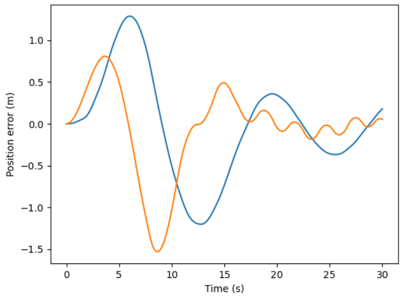


(d) 1.5m

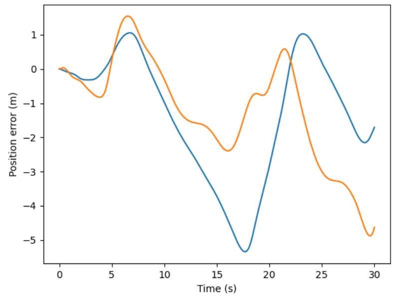


(e) 2m

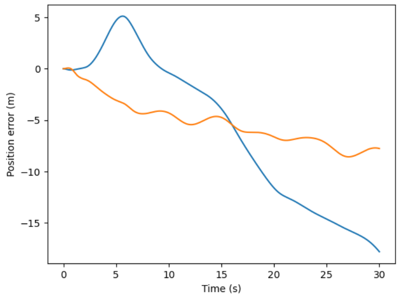
Figure A.28: ROV position error with control at different wave heights, ROV at 0m



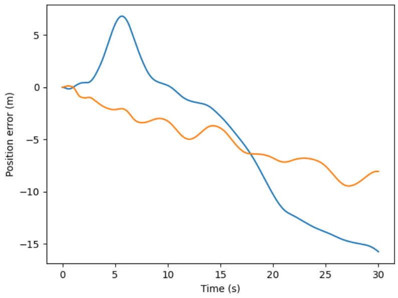
(a) 0m



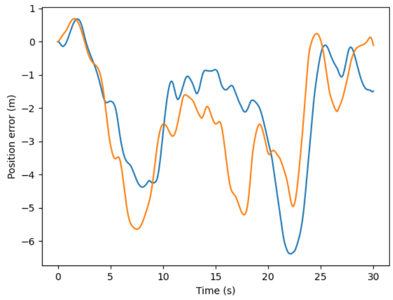
(b) 0.5m



(c) 1m

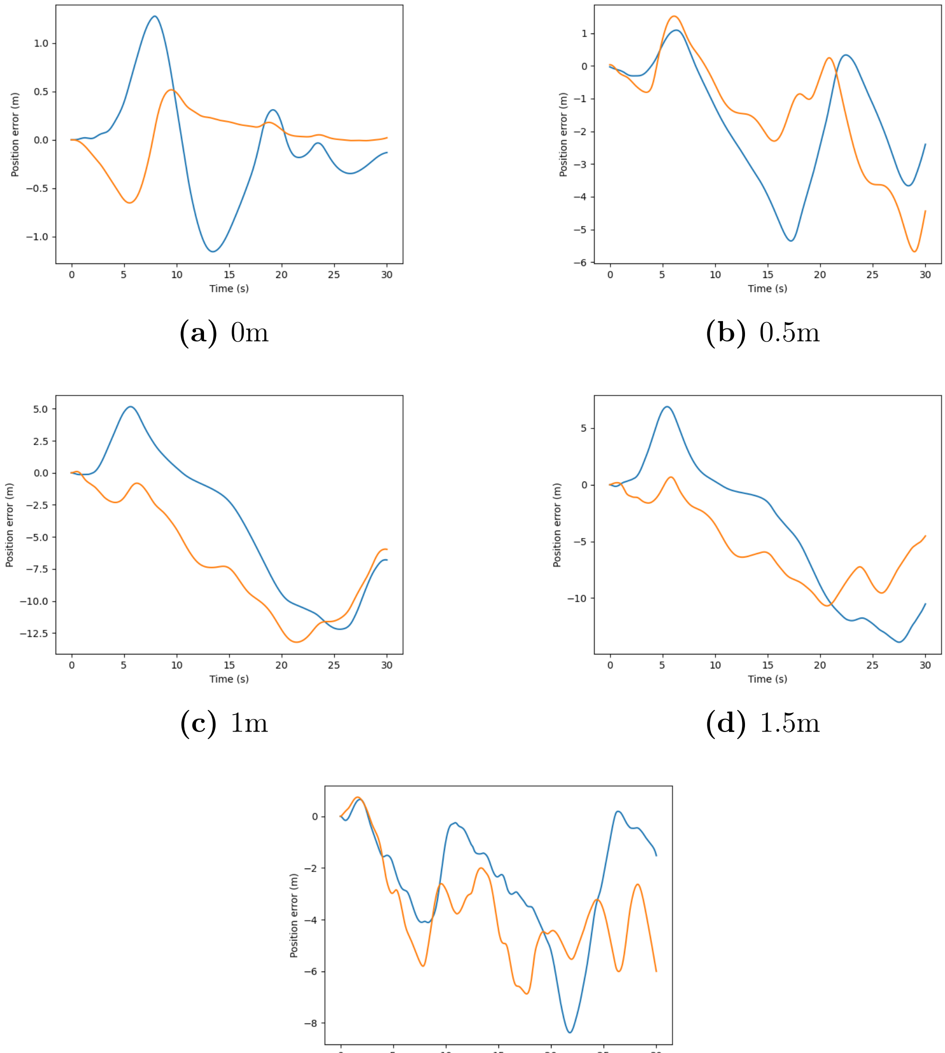


(d) 1.5m



(e) 2m

Figure A.29: ROV position error with control at different wave heights, ROV at 50m



(a) 0m

(b) 0.5m

(c) 1m

(d) 1.5m

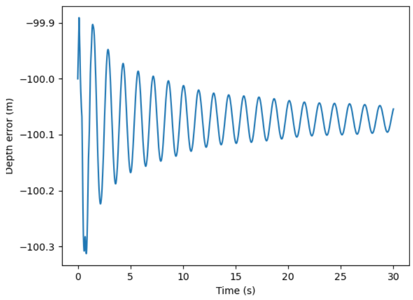
(e) 2m

Figure A.30: ROV position error with control at different wave heights, ROV at 100m

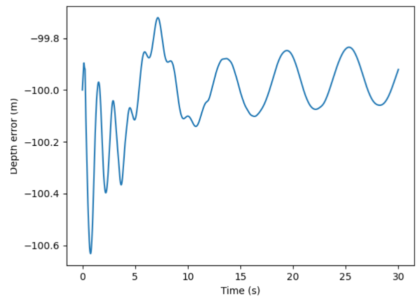
A.6 ROV depth error over time

Note that it seems the depth calculations for the plotting have gone wrong. They ended up being relative to 100m depth. It was not possible to fix this, so please simply imagine that the depths are correct. For the retracted ROV, the goal height is -100m, while for the fully extended one it is 0m.

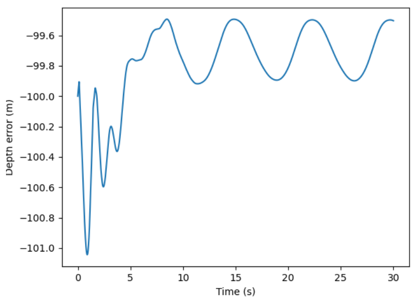
A.6.1 Uncontrolled



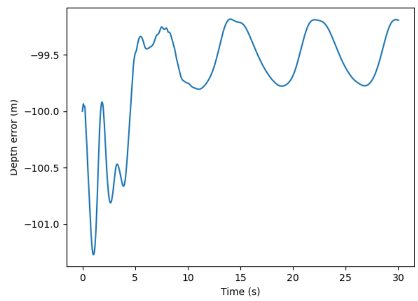
(a) 0m



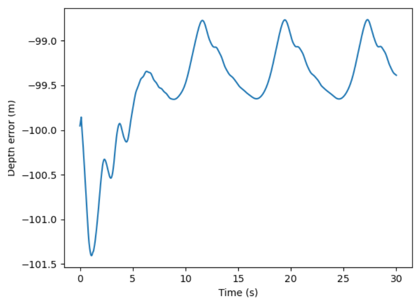
(b) 0.5m



(c) 1m

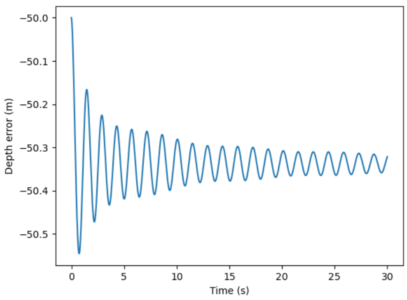


(d) 1.5m

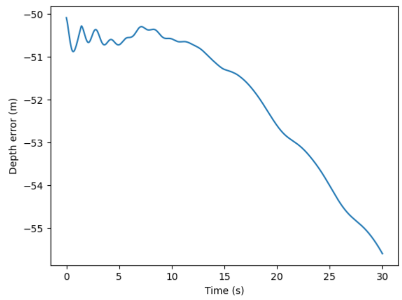


(e) 2m

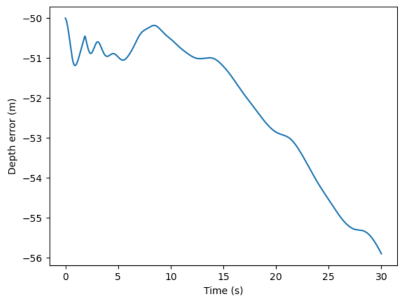
Figure A.31: ROV depth error without control at different wave heights, ROV at 0m



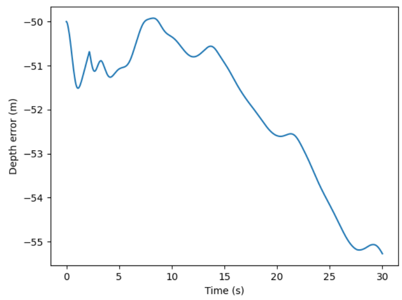
(a) 0m



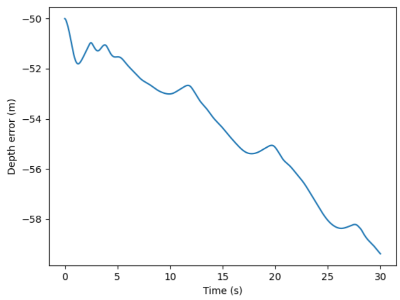
(b) 0.5m



(c) 1m

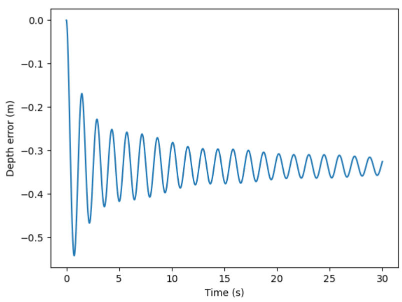


(d) 1.5m

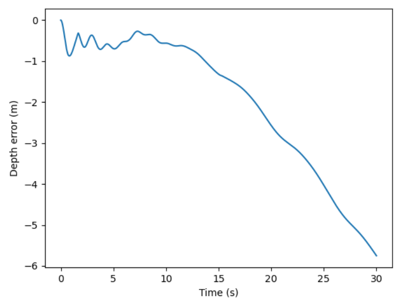


(e) 2m

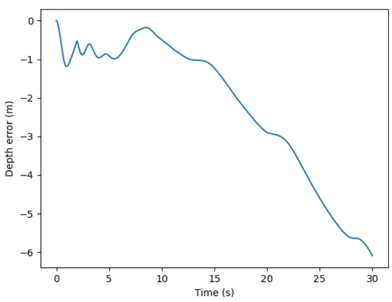
Figure A.32: ROV depth error without control at different wave heights, ROV at 50m



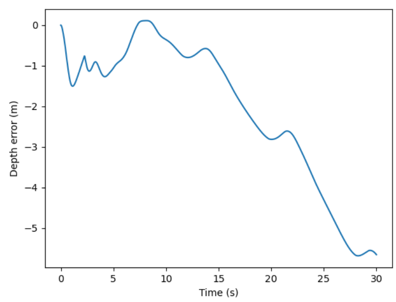
(a) 0m



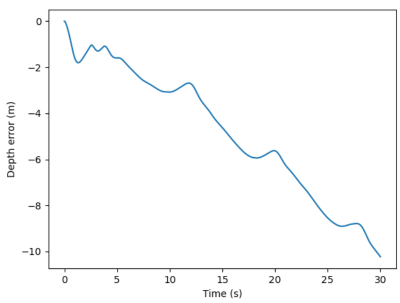
(b) 0.5m



(c) 1m



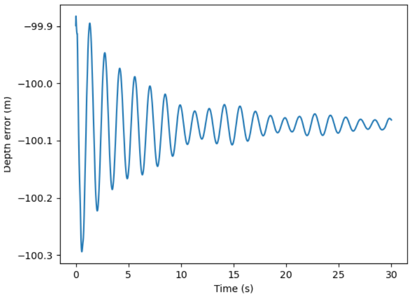
(d) 1.5m



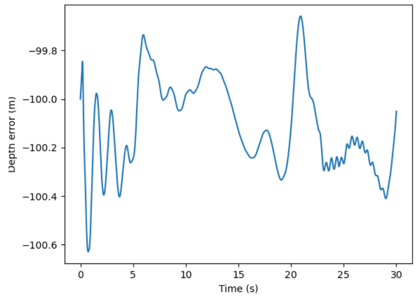
(e) 2m

Figure A.33: ROV depth error without control at different wave heights, ROV at 100m

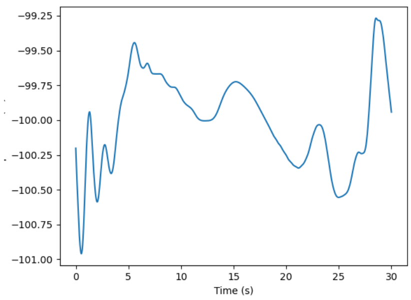
A.6.2 Controlled



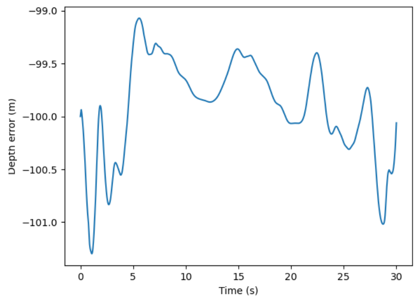
(a) 0m



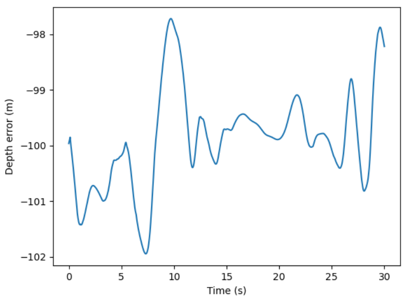
(b) 0.5m



(c) 1m

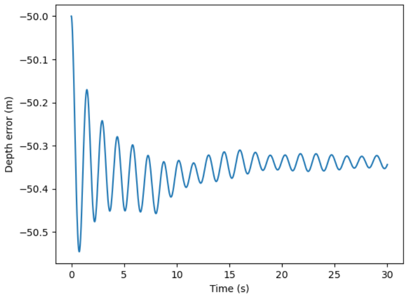


(d) 1.5m

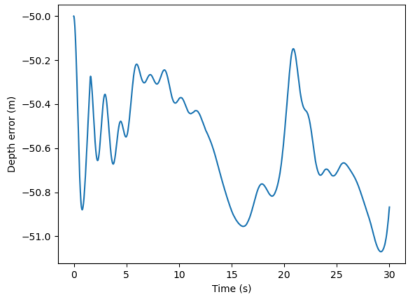


(e) 2m

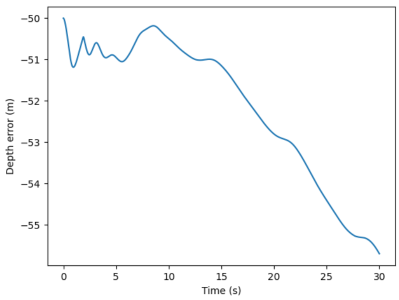
Figure A.34: ROV depth error with control at different wave heights, ROV at 0m



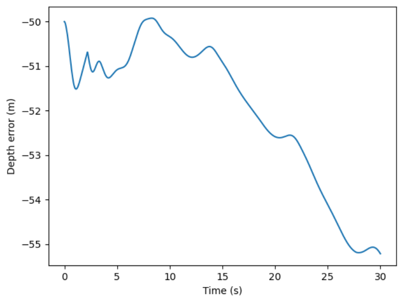
(a) 0m



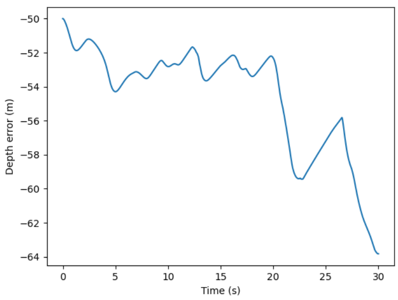
(b) 0.5m



(c) 1m

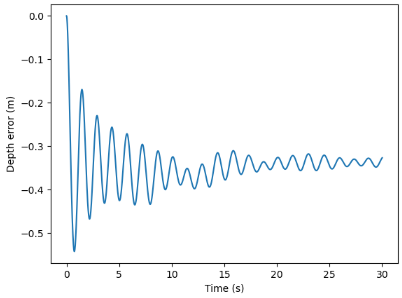


(d) 1.5m

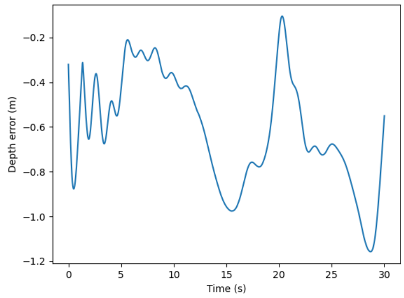


(e) 2m

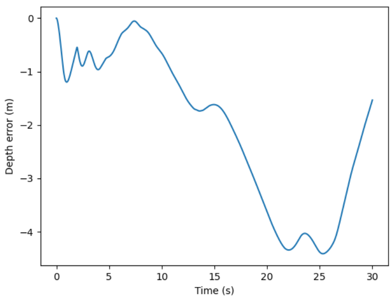
Figure A.35: ROV depth error with control at different wave heights, ROV at 50m



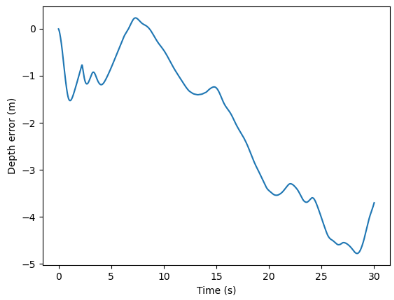
(a) 0m



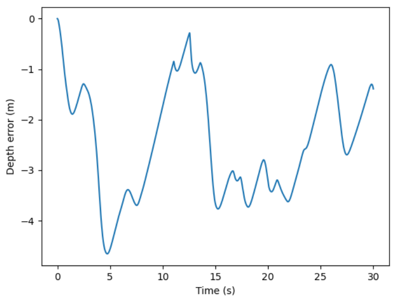
(b) 0.5m



(c) 1m



(d) 1.5m



(e) 2m

Figure A.36: ROV depth error with control at different wave heights, ROV at 100m

APPENDIX

TWO

SPECIALIZATION REPORT

Candidate number 10016

Simulation of a coupled system consisting of a non-buoyant remotely operated vehicle and a surface vessel

Specialization Project Report, IP502122
December 2024

Norwegian University of Science and Technology
Faculty of Engineering
Department of Ocean Operations and Civil Engineering

NTNU

ABSTRACT

This project purports to simulate a non-buoyant Remotely Operated Vehicle (ROV) to be used for underwater litter collection. This is done as a part of a student-driven project at NTNU in Ålesund which aims to build a functioning solution to find and pick plastic litter on the seabed.

An ROV can be used to find and grab underwater plastic litter, however lifting heavier litter is difficult. This is because heavier litter will require more force to lift. If this force is to be provided by only the ROV's thrusters, very large thrusters are required. This project simulates an ROV which is negatively buoyant, that is it sinks in water, and tethered to a winch on a floating surface vessel. The project uses Algoryx' AGX as a framework for simulating the resulting coupled system consisting of a surface vessel, an ROV and a flexible tether. The simulation includes hydrodynamic effects and is configurable to include weather effects like wind, waves and currents on both vessels.

The results of this project show that AGX is a usable framework for simulating this type of problem. Additionally the project has created a simulation which is usable for further work with this configuration of a non-buoyant ROV. Using the simulation, a simple proportional-derivative controller has been implemented and tested.

The report concludes that the simulator created is useful in further exploration on the topic of non-buoyant ROVs. The simulator has shown itself to be valid in the use-cases tested. Applications for this simulator include control system development, stability studies of the complete system, coupled system problems and as an engineering tool to properly dimension a physical vessel.

CONTENTS

ABBREVIATIONS

Abbreviations used in the report listed in alphabetical order.

- **CFD** Computational Fluid Dynamics
- **CoG** Center of Gravity
- **NTNU** Norwegian University of Science and Technology (Norsk Teknisk og Naturvitenskapelig Universitetet)
- **PD** Proportional-Derivative [controller]
- **ROV** Remotely Operated Vehicle

CHAPTER

ONE

INTRODUCTION

1.1 About this report

This report is a summary of work done in the course IP502122 - Specialization Project, undertaken in the autumn of 2024 at NTNU in Ålesund as a part of my master's degree in Product and Systems Engineering. The specialization project is intended as an introduction and "head start" on my master's thesis next semester. As such, the work done here is used in large part as a jumping off point for later.

1.2 Marine plastic pollution

Plastic pollution in the oceans has been widely documented, however the amount of plastic currently in the ocean is uncertain. Jambeck et al.[1] estimates that in 2010, somewhere between 4.8 and 12.7 million metric tons(MT) of plastic ended up in the ocean. According to the World Economic Forum[2], there is between 75 and 199 million MT of plastic waste currently in the ocean. Around two thirds of all plastics that end up in the ocean are heavier than seawater [3] meaning that they sink and either drift in the pelagic zone or end up on the seafloor as litter. Removing litter and plastic pollution on a large scale is difficult, removing it under many meters of ocean makes it much more difficult.

It is undesirable to have plastic waste in the oceans. This is because of the health effects the plastics have on marine and terrestrial life. Two points are especially of note: microplastics and leeching. Microplastics are plastic particles smaller than 5mm. Leeching on the other hand, is the plastics' chemical interaction with the seawater surrounding them, leeching harmful chemicals into the water.[4][5][6][7] For both of these issues, the best solution is to remove the litter. This is because plastics' general longevity. For example, Oluwoye et al. [8] found that polyethylene, commonly used as a coating for subsea structures, would take about 800 years to degrade on the ocean floor. Polyethylene is also used in many consumer- and industrially facing applications, for instance in plastic cannisters for liquids, boxes and crates for fishing or other industrial practices, or as plastic bags.

1.3 The Plan Sea Project

The desire to deal with sub-surface marine plastic waste, i.e. litter both in the pelagic zone and on the seabed, was what sparked the Plan Sea project. Plan Sea is a student driven project at NTNU in Ålesund with the goal of finding, developing and testing a potential solution for removing sub-surface plastics. The project is at time of writing still in its early phases and ongoing. At time of writing, a hull has been constructed from carbon fiber sandwich boards. Thrusters have been mounted to the hull and work on controlling them has started. Additionally, an ROV has been acquired for the project.

Since this is not a report focused on the Plan Sea project, the proposed solution arrived at in Plan Sea will not be discussed in detail. However, because of the relationship between this project and Plan Sea, it is necessary to describe the solution at a surface level.

1.3.1 The proposed solution

The solution which the Plan Sea project is aiming for is an ROV-based solution with an unmanned tender-vessel on the surface. The ROV has a gripper attached and will navigate to find litter, grab it and pick it up. The surface vessel exists to provide the ROV with a greater lifting capacity. If the ROV was to lift purely under the force of its own thrusters, as is traditional for ROVs, the total amount of lifting force available would be limited by the vertical thrust available. This would mean that either the ROV would have to have a very large amount of vertical thrust available relative to its size, or that the total lifting capacity would be very small, neither of which are desirable. By connecting the ROV to the surface vessel with a winch and a lifting cable, it is possible to use the ROV to do fine-navigation to find and attach to litter, and then use the lifting force of a winch and the total buoyancy of the surface vessel for lifting heavier objects. Ideally an ROV which originally only has a total lifting force of about 15kg would be able to lift heavy cables or car tires. Additionally, the solution is supposed to have an undersea basket for collection to avoid having to lower the ROV to the seabed and then lift it up to the surface for each piece of waste. Instead the ROV can be lowered down and can pick litter it finds and place it in the basket. Once the basket is full it can then be lifted up and either emptied on deck, exchanged for another basket or taken back to shore for further sorting there. A sketch of the solution can be seen in fig. 1.1.

Using this solution allows for completely ignoring the buoyancy of the ROV, unlike traditional ROVs. Traditional ROVs are generally designed to be neutrally buoyant, meaning that they neither sink nor float, but keep their vertical position in water once placed there. Since the Plan Sea ROV will be attached to a cable to the surface vessel at all times, it can instead hang from the cable. This means that it's possible to attach larger grippers, more battery capacity, more detection/lighting/navigation equipment, and otherwise allows for any desired modifications to be done to the ROV. Additionally, since the ROV doesn't need to provide vertical thrust, it is much easier to not disturb the seabed which will provide a clearer view for detection equipment based on visible or near-visible light. However, having a non-buoyant ROV does come with some drawbacks.

One drawback of this solution is that it will switch between two operating modes, searching/grabbing and lifting. In the search/grab mode the ROV will be near-neutrally buoyant, or somewhat negative. When lifting the ROV might be severely negatively buoyant. These two wildly different operating modes increase the needed complexity of the sys-

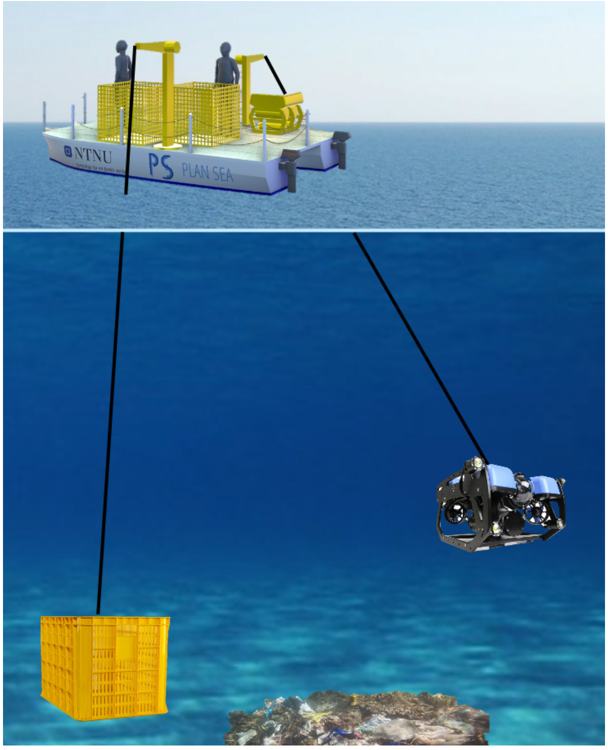


Figure 1.1: A sketch of the proposed solution for Plan Sea showing a surface vessel, a tethered ROV and a collection basket

tem. Another drawback is that as the ROV is hanging from the cable, it creates a coupled system consisting of the surface vessel and the ROV, and necessitates the two moving together as one unit. The forces the surface vessel experiences, such as waves or wind, will impact the ROV, likewise currents or snags the ROV experiences will affect the surface vessel.

1.4 Control systems

A control system commands and regulates the behaviour of other systems automatically. For this project in particular, the control system will be in charge of maintaining and changing positions of the two vessels. A simplified function block diagram of the total system can be seen in fig. 1.2. The goal of the simulator is to function as a drop-in replacement for the vessel, local controllers and environmental impact shown in the figure. This makes it so that the development can happen digitally to then be quickly deployed in the real-world.

1.4.1 Considerations because of a coupled system

In the marine sector, dynamic positioning (DP) is commonly used. DP allows for a vessel to maintain a position or a course automatically despite external effects. This is used for example for offshore supply vessels which need to stay stationary relative to an anchored platform to allow loading and offloading of supplies. DP is also used for applications such as laying subsea fiberoptic cables, where maintaining correct speed and course is important to avoid damaging the cables. For the Plan Sea project too, a DP system is

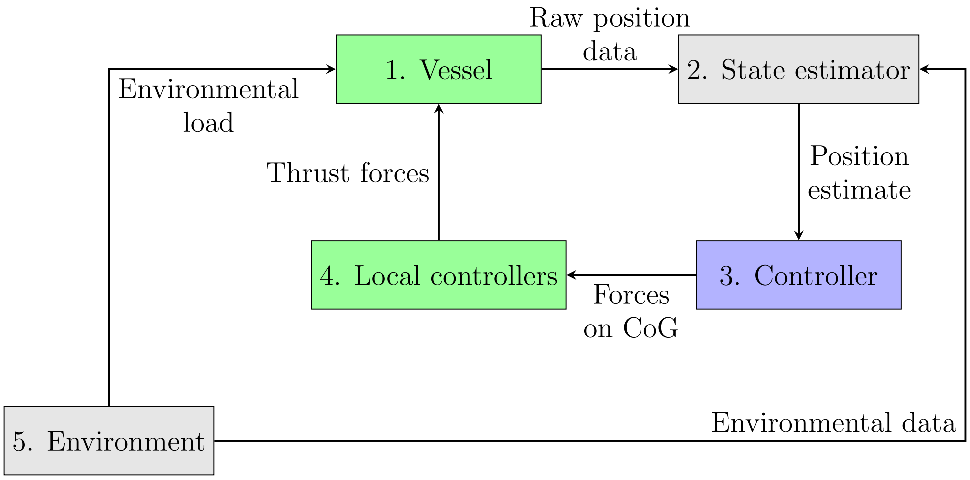


Figure 1.2: A simplified function block diagram of the total system. Grey blocks are not currently implemented, green blocks are simulated and blue is for the controller which is system-agnostic

necessary because it consists of vessels that need to maintain specific positions at sea with wind, wave and current forces affecting the vessels

Normally a DP system only considers one vessel, however for the Plan Sea project it has to be more comprehensive than that because the two vessels are coupled. This is all further discussed in section 2.2

1.4.2 The need for rapid prototyping

Rapid prototyping is becoming increasingly popular with time. The goal of rapid prototyping is to create some simulated environment in which you can test and iterate on a solution until it is acceptable. Then, once you have a solution that works within a certain level of acceptability you can start to put materials and resources into building and implementing the solution in the real world.

For my purposes, rapid prototyping will allow me to experiment with control system tuning and variables without having to deploy the full-scale vessel every time. Ideally, the solution arrived at in the prototyping stage will be directly applicable to the full-scale version, which allows for rapid deployment. The hope is that any issues will be detected and solved while testing digitally, meaning that we hopefully avoid large surprises during deployment.

1.5 Problem description

A simplified sketch from fig. 1.1 can be seen in fig. 1.3. It shows the three main components of this system: The surface vessel, the ROV, and the tether between them. It also shows the forces in the tether. As the tether holds the ROV up, the tether pulls the surface vessel down. This follows from Newton's second law of motion. The figure also shows the coupled nature of the system, since the tether will be taut at all times, both vessels will experience this force from the other at all times.

In fig. 1.4 two scenarios are shown overlaid, one where the ROV is hanging straight below

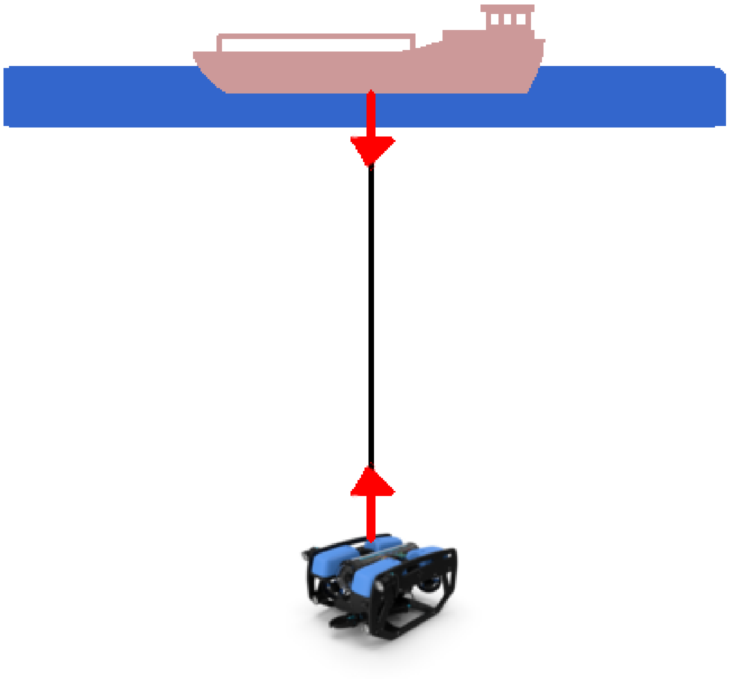


Figure 1.3: A simplified sketch of the problem, forces experienced by each vessel are shown in red. Forces are not to scale.

the surface vessel and another where it is at an angle. The result is that the ROV is not at a constant height. If we imagine a desired elevation above the seafloor is constant, then either the ROV needs to have more tether payed out and provide lift through its own thrusters, or the surface vessel needs to move to allow the ROV to hang perpendicular to the surface of the sea. If more tether was to be payed out and the system then stabilizes, the ROV will fall to the lowest possible point. It is possible then that the ROV might collide with the seabed or other objects. The ideal solution then becomes that the surface vessel follows the ROV, or the ROV only operates within a given area of operations directly underneath of the surface vessel.

Further, this non-perpendicular arrangement will lead to tangential forces, shown in fig. 1.5. The forces will of course be equal and opposite on the vessel's end, though this has been omitted from the figure for clarity. The horizontal force the tether imparts on the ROV will act as a restoring force, trying to move it back to be perpendicular to the surface vessel. The surface vessel likewise will experience a pull towards the ROV.

Because these two vessels are connected, and therefore dependent on each other's positions, the control system needs to take this into account. Probably, the simplest solution will be to have one large control system handling both, or alternately having one of the vessels take a leading part and the other attempt to follow. This will be touched on later in this report, though not discussed in detail.

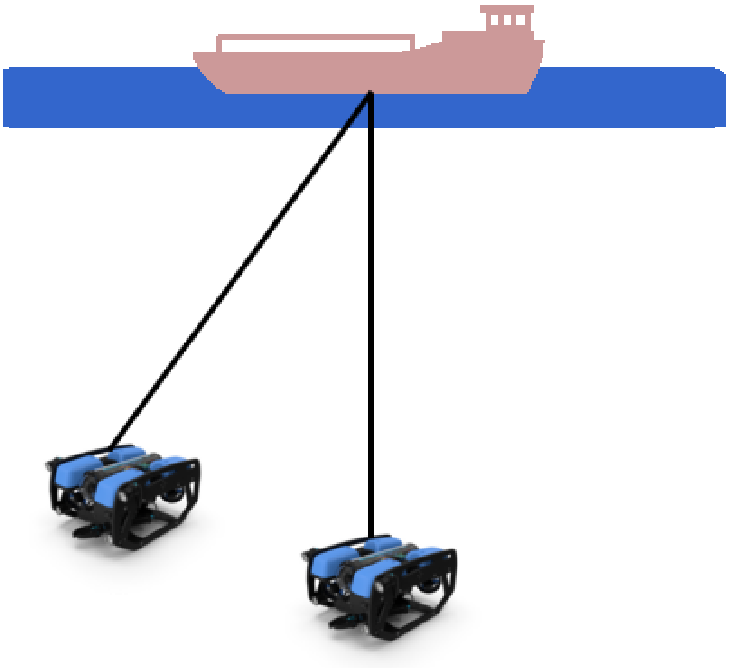


Figure 1.4: The ROV is hanging from the same point on the surface vessel with an equal tether. Note how the height of the ROV has changed because the tether has stayed the same

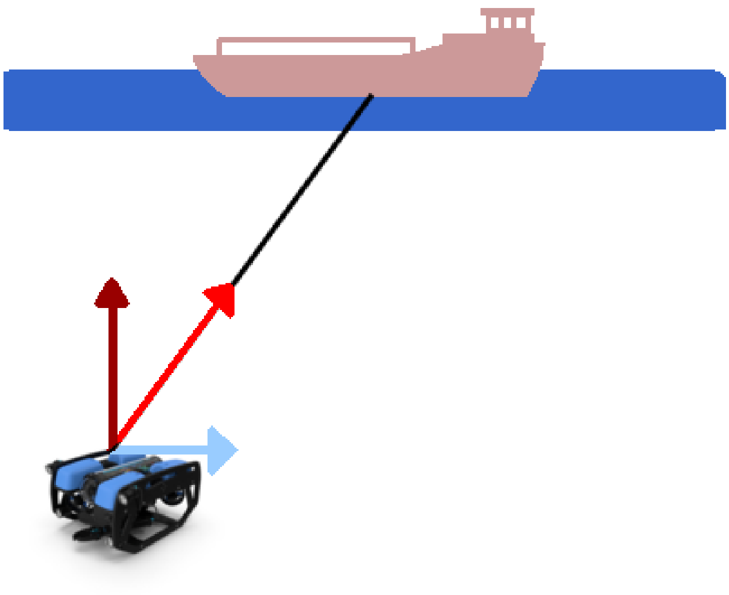


Figure 1.5: The component forces on the ROV resulting from an angled lift, decomposed

1.6 Statement of intent

For this project I want to create a simulation which is able to quantify the effects that the parts of the system have on each other. I want to be able to measure the tension in the wire, the force exerted by the vessels, how well the vessels follow each other or orders given and influence from the environment. The goal of this simulation is to be used as a future design tool for finalizing design of the Plan Sea vessel, its control systems, as well as defining operational criteria.

CHAPTER

TWO

MODELLING AND CONTROL DESIGN

2.1 State of the art

There is little literature specifically on the topic of coupled powered systems with elements both above and below the surface of the sea. I have however found bits and pieces that are useful. Primarily the offshore oil and gas industry, as well as general ROV operations are able to help a bit. The main issue with this project is that it is fairly unique. Usually when dealing with ROV operations, the surface vessel is intended to stay stationary for longer periods of time, or the ROV is neutrally buoyant and able to do its work below regardless of what the surface vessel does on the surface. This project deals with a powered non-buoyant ROV which needs to be able to do both large motions to sweep larger areas of seabed, as well as precision work to move in on and pick up litter. This is a unique problem from the literature I have found.

There are several papers discussing for instance the effect of currents on deep-submergence suspended tooling or ROVs used for oil and gas installations that can help. For example, Lian and Sortland 1996 [9]. They also performed simulations, however their paper is focused on non-powered remotely operated tooling as opposed to a remotely operated vehicle. Their results can be useful as a kind of sanity check for the results of the simulations of this project, though their working area is down to 1500m below the surface while it's improbable that the Plan Sea project will ever operate below 200m.

The tether is also a consideration for this project, Chen et al., 2021 [10] discuss the hydrodynamic effects on ROV tethers under complex sea conditions. Whether or not this is useful for this implementation is uncertain, as the tether simulations are all done by the simulation software. Additionally the tethers used for this iteration are likely so small in diameter that their effects in the sea are likely small. It might be possible to expand the tether simulations to include Chen et al.'s findings in future.

Enevoldsen et al., 2018 [11] provide one of the more useful documents on the topic. They discuss a simplified modelling strategy for ROVs to allow for both greater control capabilities and simulated efforts. The state of this project has not used this information, but it will be very helpful in later iterations to expand the accuracy of the simulation for the ROV, both for control purposes and for a more accurate simulation. For instance a model of the vessel can be implemented into the controller so that it's able to act predictively and not just reactively like a PID controller would. Anderlini, Parker and Thomas, 2018 [12] discuss control of an ROV carrying an object. They discuss the sudden added mass of the vessel and how to compensate for it from a control perspective, though their paper focuses more on autonomous underwater maintenance vehicles. This paper as well will be very helpful for further implementation work, but has not been used in any great extent here. The Anderlini paper also primarily focuses on self-propelled underwater vehicles, not the externally lifted version that this project considers. Thingstad and Hveding, 1982 [13] have a conference paper on non-buoyant ROVs for performing subsea work. On the surface this sounds perfect, but looking into the paper it is more focused on the physical construction of the ROV rather than the control of it. This makes it less helpful for me. Additionally, Thingstad and Hveding's paper is more than 40 years old at time of writing, and applications of control theory, as well as microelectronics and actuators have evolved a lot since then, making what little control they do discuss less useful.

2.2 Mathematical basis

The root problem can be decomposed into equations 2.1 and 2.2. They show the forces that impact the momentums of both the surface vessel and the ROV. The forces that have an impact are hydrodynamic forces, such as buoyancy, righting moment etc. The propulsive forces that the vessels' thrusters provide. Environmental forces coming from waves, winds and currents. And finally the force the coupling acts with on each vessel. Do note that though the form of the equations is the same for the surface vessel and the ROV, the values both in total and in each individual element are not necessarily equal. Hydrodynamic, propulsive and environmental forces will be entirely individual for each element because of their physical shape and capabilities, and while the coupling forces are linked, their relationship is not necessarily linear. See section 2.2.1 for elaboration.

$$M_{v_{\mathrm{surf}}} = \sum f = f_{hydro} + f_{prop} + f_{env} + f_{coupling} \tag{2.1}$$

$$M_{v_{\mathrm{ROV}}} = \sum g = g_{hydro} + g_{prop} + g_{env} + g_{coupling} \tag{2.2}$$

The equations demonstrate the need for simulation compared to analytical examination of the problem. Hydrodynamic forces and environmental forces contribute to a highly dynamic system for which it is difficult or impossible to find a closed form expression. This means that calculating an expected state for the dynamic system manually is labour and time intensive. Using a simulation instead of analytical methods hides these problems away. The simulation will take care of the complex interactions which allows me to focus on extracting interesting data. I will further discuss simulation options in chapter 3

Using a simulation does introduce a new requirement: that of validating the models' accuracy. There are many ways of doing this, both intuitively and mathematically. I will get talk more about the validation methods I've chosen in section 4.2.

The simulation allowing for "setting and forgetting" whatever parts of the force equations are desired allows the user to focus on whatever specific field they are interested in. For

Figure 2.1: Illustration image of a chain forming a catenary between two posts under the force of gravity.
Used under Creative Commons, credit: `https://en.wikipedia.org/wiki/File:Kette_Kettenkurve_Catenary_2008_PD.JPG`

example a user might examine the propulsive force required given a certain seastate, or how hull shape and hydrodynamics affects the stability of the total system. Using simulation gives the user a greater degree of freedom in finding exactly the variables they're interested in.

2.2.1 Catenaries

When a rope or chain is suspended from two points and affected by forces not in-line with the two points, the chain forms a catenary. Catenaries are relevant to this project because the lifting tether will form a catenary whenever the ROV is not directly underneath the surface vessel and also not experiencing any external forces. If there is a current or movement affecting the tether it will form a catenary.

The fact that the lifting tether will form a catenary is helpful to know, as one might intuitively assume that the lifting tether would be straight between the surface vessel and the ROV. If the tether were smooth, finding the position of the ROV could be done by trigonometry, knowing the length of the tether, its angle at the lifting point relative to gravity, and knowing the tether's azimuth, the ROV could be located using Pythagoras' theorem. However, since this simplification can't necessarily be done, a slight complication has to be added. The angle at the lifting point is not necessarily directly linked to the angle of a straight line to the ROV. This can be seen demonstrated in fig. 2.2

In 2 dimensions, a catenary is well defined as

$$y = a \cosh \left(\frac{x}{a} \right)$$

Where a defines the width of the catenary and can be found in relation to the relevant

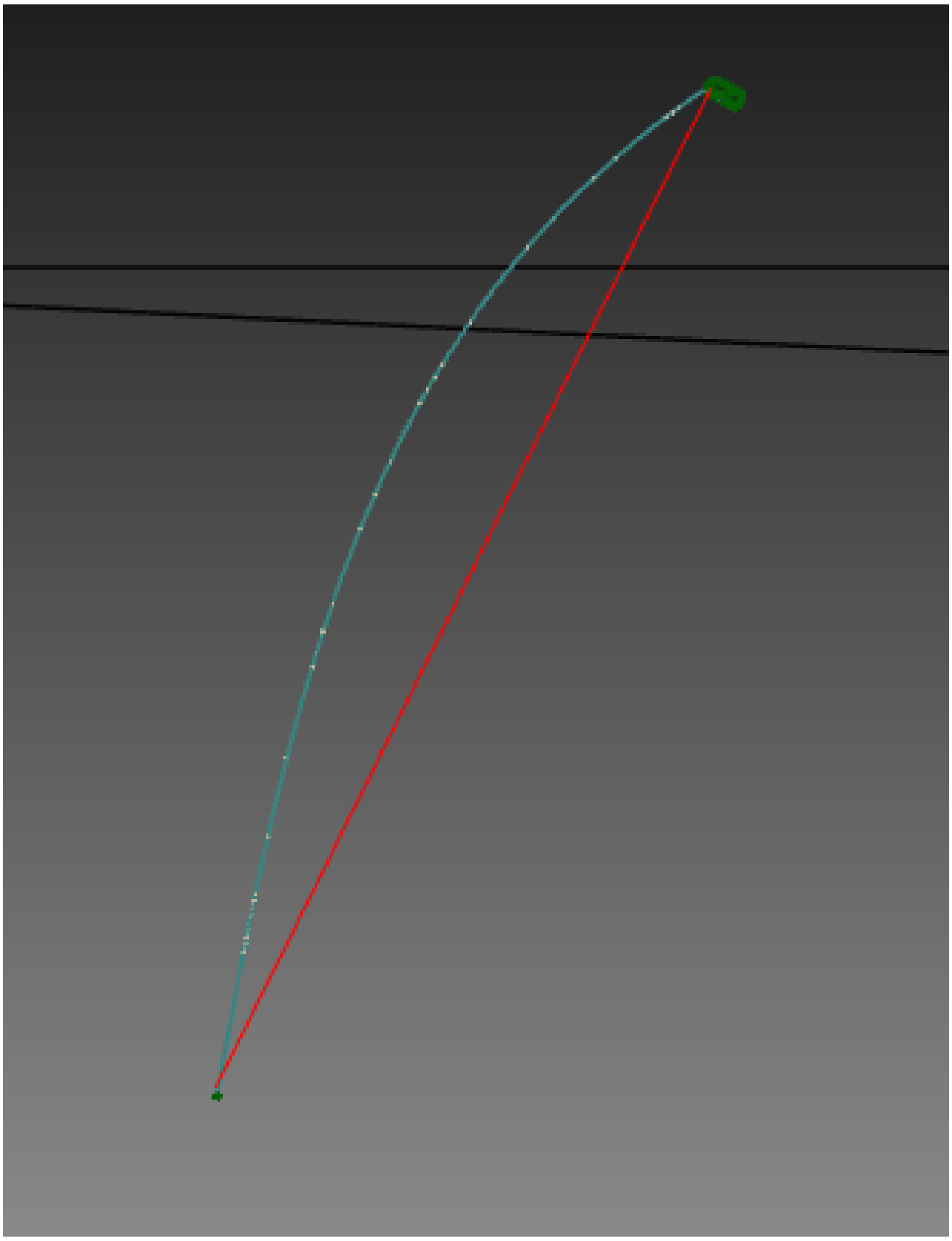


Figure 2.2: The simulation showing the towing line (light blue) and a straight line between the surface vessel and the ROV (red). Note how the towing line is not straight

forces acting on the rope. For this project's applications however, the shape of the tether can't necessarily be simplified to 2D. I have been unable to find a simple, closed form of the catenary equation in 3D for a rope (as opposed to a plane), though it may well exist. This further solidifies the necessity of a simulation over using analytical methods. Additionally, the simulator has shown that the tether does not necessarily form a catenary. This is also shown in fig. 2.2 where the shape of the tether is more bowed near the surface than near the bottom.

This simulator can be used to find an error between assuming a straight line between the ROV and the surface vessel, compared to the real situation. This relationship or error can be included as a part of the internal model of the vessel's state estimator and be used as a part of positioning the ROV under water.

CHAPTER

THREE

SIMULATION FRAMEWORK

The goal of this specialization project is to build a simulator which is applicable to the case of the Plan Sea project. There are four elements that need to be simulated for this to be considered a success with a fifth "nice-to-have" element. The elements are:

- Surface vessel
- Subsurface vessel
- Connecting wire
- Control system
- Weather impacts

There are two main paths to take with regards to making the simulation. I could make everything from first principles or I could use an already existing simulation framework and build something on top of that. Both solutions have positives and negatives.

Creating a system from scratch would be an interesting challenge. It would also give me exactly the results I'd want with very little overhead, assuming that my own programming skills and understanding of the relevant physics are up to the task. The bespoke, home-made simulator also wouldn't have any associated licensing costs. On the other hand, to create a simulator which includes buoyancy, fluid dynamics, wire physics, handles a coupled system and also has some form of graphical interface/readout for the user is a large task to undertake, especially in the roughly 4 months available to this project.

Using a commercially available system also has a fair few benefits. I would have a mostly ready-made framework within which I can just configure the simulation as I need it. Changing out parameters and variables would also be very simple, as it's essentially the same operation as configuring the simulation from the beginning. I believe also that the results from a commercial solution would be more reliable than my own attempt. It is reasonable to assume that an industrially used, commercially available simulation solution made by a team of physicists, computer engineers and other specialists is going to be more accurate with regards to its results than me. Commercially available solutions are not all perfect however. There are licensing costs associated with many simulation frameworks. Some having enormous costs for the scale of a student project. There is also likely more computational overhead with a commercially available solution due to them being as wide in their application as possible. This can make the commercially available solution slower or less responsive in use.

After considering the points above for both commercially available and personally crafted simulation frameworks, I decided that using a ready-made solution would be better. This was especially decided because of the time constraints of this project. My goal is to have a working simulation that can be used to provide information, the goal is not to make a simulator. If I was to make it myself, the project would quickly turn into "make a simulator" rather than "make a simulation", simply because of the scale of the undertaking.

3.1 AGX

After considering multiple simulation options and on the advice of a professor, I landed on using Algoryx' AGX as a simulation framework. AGX has a solid wire simulation package, a hydrodynamic simulation package, and allows for scripting and setup using both Python and C++. It has interfaces towards both Unreal Engine and the Unity engine for further graphical display of the results or graphical user interfaces. AGX allows for multi-body simulations, and in addition it has interfaces for ROS2 which I will get into in section 3.3.

My implementation of AGX is based on Python as that's the language I'm most familiar with. I am aware of Python's inferiority to a C++ based approach in terms of speed and efficiency with regards to computational resources. The time investment required for me to get to an acceptable level of C++ proficiency was not worth it for this project, and so I stick with Python.

3.1.1 Limitations of AGX

By using a ready-made simulation platform I am able to quickly implement a simulation without needing to worry about the mathematical models that exist behind the simulation. This allows me to focus on achieving results, although it also necessitates a degree of trust in the simulation. I am able to verify whether the simulator acts as expected or not, but changing the governing equations is not necessarily something I am able to do.

Another limitation of AGX is that it's a licensed software. This means that in order to apply and use the findings of this report, the software needs to be acquired. This is a limitation for further research, and ideally the findings should be based on open and available software or arrived at from first principles.

Algoryx lists some known limitations of the relevant packages in their user manual[14] pp. 95 and 169, as well. For the hydrodynamic module for example, wakes and object-generated waves are not possible, semi-submerged convex objects will also not be water-tight. There is also the possibility of unwanted behaviour when using multiple linked models. I believe this is in reference to multiple rigidly linked models, not loosely linked ones like I'm using. A limitation of the wire module is that wires can't handle torsion forces. This is likely unnecessary to model for the lifting tether, as it will be sensitive to torsion like some cables or wires might be.

3.2 Description of the simulation setup

The simulation starts with defining a water volume. This is done using AGX built-in functions. Currently the volume is 200m along the sides and 100m deep, though this is arbitrary. I've implemented one "water controller" which is the object that handles wave, current and wind forces. Currently this controller is completely still and there is no force inputs, but it is implemented so that adding weather forces is simple for later iterations. In AGX, each element in a simulation has to be individually added to that simulation. Once both the water volume and the water controller have been added to the active simulation, the vessels are made.

The surface vessel and the ROV are both implemented as children of a general vessel parent class. I've done this for later ease of expansion. The ROV is just a simple box with a given density and size. The ROV's size I've taken from BlueROV's websites[15], as an example of the size of ROV we will be working with in this project, though as mentioned it's just implemented as a box. The University does currently have a BlueROV that is intended to be used for the initial steps of the Plan Sea project, as well as for my master's thesis which will be based on this work, that is why I've chosen to use their dimensions. The exact dimensions are $0.45m \times 0.575m \times 0.254m$ for a total volume of $0.0654m^3$. If a more accurate shape for the ROV is desired, it is possible to implement it similarly to the surface vessel, but since this project is mostly a first-order approximation of the problem, I believe a simple box is sufficient. The ROV's density I have wildly exaggerated and arbitrarily chosen to be $2000\frac{kg}{m^3}$. This is an absurd exaggeration and should be replaced with more accurate data later. As a proof of concept it will work fine. This gives a total mass of the ROV of 131kg.

The surface vessel's shape is defined by a wireframe stored in an `.obj` file I've made that approximates the shape of the hull, while its density I've approximated from the density of the carbon-fiber sandwich board used to $600\frac{kg}{m^3}$. The simulation software has provided a mass calculation for the vessel to approximately 2300kg. This gives the ratio of mass between the two vessels of roughly 5%, which is far more than an assumed real-world estimation of less than 1%. This will skew the results to exaggerate the impacts of the ROV's motion.

In addition to a class which handles the shape and properties of the surface vessel, I have made a class that handles controlling the vessels. The controller is just a simple implementation of a PD controller. The controller checks the position of the vessel on each timestep and calculates an error. The error is then controlled using the PD controller and a response is found. The response is clamped within a given authority limit that simulates the vessel's control authority limited by the physical capabilities of its thrusters. I've estimated the authority for right now, but it should be changed later when the real command authority is found.

Currently, the way the controller is acting on the vessel is by simply applying a force in a given direction on the CoG of the vessel. For the surface vessel, any force and error given in Z-direction is zeroed as the surface vessel will not be able to cancel its own heave in waves. This can be changed later and the Z-direction error/control can be implemented as heave-compensation for the ROV. This isn't implemented yet though. I've also implemented a simple proportional controller for heading control which works by adding a torque to the body it's controlling. In a future iteration, both the simplistic

Figure 3.1: Illustration image of the BlueROV2 used as a basis for the ROV in this project. Credit: BlueRobotics

control and heading control could be implemented by splitting the force inputs from one single input at CoG to one input at the position of each thruster. Then yawing motion can be applied with differential thrust and a heading-term can be added to the controller. Whether the complications of simulating force inputs at the thruster locations are necessary is not clear yet. If we consider the ideal future implementation in a physical vessel, the controller designed now is only intended to give the commands to the vessel as force acting on CoG. Thrust allocation and local control will be designed elsewhere and simply be a node in the ROS2 system that will be the final vessel. Either way, the option of making the simulation more true to life exists if it should be desirable.

Finally, the wire which connects the ROV and the surface vessel is created. AGX has two similar wire-like simulation objects: Wires and Cables. The documentation gives a comparison of wires and cables. Because torsion of the wire is irrelevant, we want to be able to winch the wire in and out and we are working on long wires, the Wire module should be used. The way wires work in AGX is as a series of links connected to or passed through a series of nodes. For this implementation only two nodes are necessary, one connecting the wire to the surface vessel and one connecting it to the ROV. More nodes could be used for example if the wire was to be pulled around a corner, or through certain points for routing. The Wire module of AGX allows for winches to be simulated as well, with given speeds, gearings, torques etc.. I have not implemented this in the current simulator.

In addition to all the "necessary" elements, the simulation also has a manual controller for the vessels. It is possible to use the keyboard to give manual force inputs on the vessels. This uses the same framework for adding force as the controller does. I've done this as a way to debug and test the system a bit.

The final simulator can be seen on Github[16]. The URL is `https://github.com/MagnusKjorseng/FordypOgMaster/tree/main/Simulator/V3`

3.3 ROS2

ROS2, short for Robot Operating System 2, is an easily expandable and configurable operating system used primarily for hobbyists and research in control and robotics. The main selling point of ROS2 is its node model where different parts of a control system can be placed in separate, segregated nodes with certain interfaces. Those nodes then either publish data to or subscribe to data from what ROS2 calls topics. This system grants a developer or team of developers flexibility with regards to changing out certain nodes while still allowing the larger system to work. I.e. experimenting and changing out single nodes, so long as the same topics are still used, is extremely simple.

I was not able to implement ROS2 during this project. I believe this is because ROS2 is not designed to be used on Windows and requires a lot of workarounds which I was not able to figure out. This section is in the report more as a reference and a reminder for future iterations based on this report that ROS2 would make the project easily scalable and iterable and should be implemented later.

AGX has the ability to work with ROS2, having built in methods for both publishing and subscribing to topics. Applied to this project it would in theory allow for the control system to be built in ROS2 and connected to the simulator. The control system could then be tested and tuned in different simulated environments until satisfactory results are achieved. When the control system works as intended, it can then be disconnected from the simulator and connected to a physical system and be tested in the real world.

The end result of the process above would be two equivalent systems, one digital and one physical, which would allow for rapid prototyping in the digital space before quick deployment in physical space.

CHAPTER

FOUR

RESULTS

4.1 Simulation setup

The finalized simulation setup consists as described in **??** of two vessels connected by a tether. The vessels exist in a $200m \times 200m \times 100m$ large water volume. The submerged vessel is denser than the surrounding water and the floating vessel is less dense. For the testing I've done I have set the tether length, and therefore ROV depth, to be 20m. This was done arbitrarily, but chosen because it allows for the effects of the system to propogate quickly, meaning stabilization time is lower. The two vessels are connected by a tether which is 10cm in diameter. This diameter was chosen mostly so it would be visible in the readout as otherwise it would be so thin as to disappear. Of course, choosing a larger tether makes the tether's impact on the results larger. This may be a source of error in the later results. The wire is defined with a Young's modulus of 10^9, this value was chosen as an estimate based on other somewhat similar tethers I found commercially available.

The simulation works as would be intuitively expected. An example of the simulation graphical interface can be seen in fig. 4.1. The figure shows a surface vessel and the ROV under water in green, and the teal tether connecting them. The simulated water is shown as a grey volume, but it's not easy to distinguish it in fig. 4.1 because it takes up the entire screen. When interacting with the interface the water volume becomes more obvious.

4.2 Model Validation

There are two ways I will attempt to validate the simulator results here. One is intuitively and the other is using analytical methods and a simple case. The intuitive demonstration is difficult to convey in text-form in a report, but I have placed some animations of the results on GitHub[16] alongside the simulator. The intuitive demonstrations consist of starting the simulation, pulling objects around and seeing if they "act right". Human minds are excellent pattern recognition machines, and I will use this to my advantage here, as if something is "off", it would be noticable.

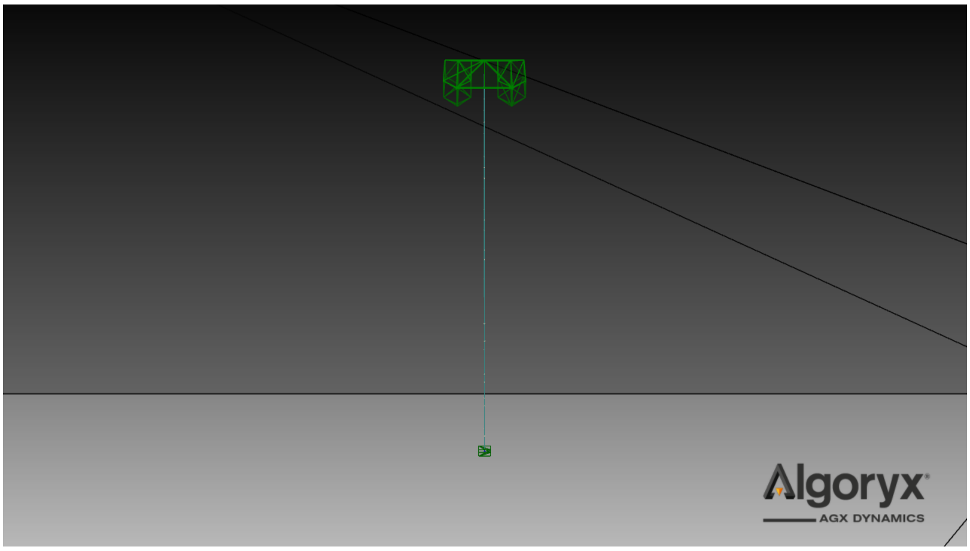


Figure 4.1: The graphical user interface of AGX with the simulation running

For the analytical validation I will use analytical methods and simple physics to estimate an expected result, and then simulate the result to compare the values for both.

4.2.1 Intuitive validation

I have done two tests here, one where I pull the surface vessel and look at how the ROV follows, and the other where I pull the ROV and see how the surface vessel follows. The shape of the tether is also relevant here. Looking at the results, shown in the `Results` directory of the simulator on the Github repo, there is nothing that pops out as obviously wrong. Of note is that the surface vessel flips in the ROV pulled test. This is because I pulled the ROV down and given the situation the surface vessel was in it was more stable flipped upside down. This is not a realistic scenario for the real-life project because the amount of force exerted pulling the surface vessel down will never be that large. Looking at the tension in the tether at the ROV it reaches upwards of 20kN at points, which corresponds to roughly 20 tonnes lifted.

In addition to the two tests above I've also tried swinging the ROV as a pendulum underneath the surface vessel. This test also confirms the intuitive assumption that when the surface vessel is far more massive than the ROV, it will be less affected when pulled like this, although it will still be affected.

I will note that in the two first animations the ROV is shown as larger than in the last. This was due to an implementation error on my part. The result of this is that the ROV is far more massive than it's supposed to be, and also larger which gives it greater drag in the water. Neither of these affect the intuitive approach as the physics still "act right".

4.2.2 Analytical validation

For the analytical part of the validation I will use a simple case in which the surface vessel moves forward, towing the ROV behind it. This will be roughly analogous to how the ROV and tether will respond in currents, though not entirely interchangable. The current speed will impart parallel forces on the tether and ROV while the towing case will impart a tension force on the tether. Still, this is a simple case for rough validation of the model and can be further worked or reworked for later use.

It is possible to analytically find an expected tension in the tether and then compare this with the simulated results. I will ignore the effects of the tether in the analytical calculations for simplicity. This will be a source of error on the final result as the tether will have an effect in the simulation.

The tension on the tether in the towing case will be dependent on the resistance of the water around the tether and ROV as well as the effect of gravity. This gives the equation

$$F = F_g - F_b + F_D$$

Where F is the total force pulling on the wire, F_g is the force due to gravity, F_b is the buoyant force and F_D is the drag force.

For the force of gravity I will assume that gravity is in-line with the tether. This is not the case in reality as the ROV will lag behind a bit as can be seen in fig. 5.1. This causes effectively a cosine error. This angle can be quite large, however since the only thing keeping the ROV from sinking is the tether, the gravitational force is still transmitted through the tether. The force of gravity becomes

$$F_g = mg \approx 131kg \times 9.8\frac{m}{s^2} \approx 1300N$$

Buoyant force is given by the volume the ROV displaces and is given by

$$F_b = \rho V$$

Where ρ is density of the fluid, V is volume displaced and g is the acceleration due to gravity. The seawater used in the simulation is defined with $\rho = 1025\frac{kg}{m^3}$ and the volume of the ROV is previously found to be $V = 0.0654m^3$. Using these values, we can find that the buoyant force is

$$F_b = \rho V g = 1025\frac{kg}{m^3} \times 0.0654m^3 \times 9.8\frac{m}{s^2} = 657N$$

When it comes to drag force, the ROV will be the largest influence due to its large size compared to the tether. The resistance for an object in a fluid (drag) is given by the equation

$$F_D = \frac{1}{2}\rho v^2 C_D A$$

Where v is velocity, C_D is the coefficient of drag and A is the cross-sectional area. For the cross-sectional area I will assume that the ROV doesn't rotate around any axis as it is being dragged. In reality it definitely will, which will change both A and C_D. Including the effect of these rotations will be very complex, and so I will ignore them. I will assume that the cross-sectional area is the front facing area, defined as

$$A = 0.45m \times 0.254m = 0.114m^3$$

Coefficient of drag is a value found experimentally and normally referenced in tables. The drag coefficient of a cube is according to tables 1.05, while the drag coefficient of a

Velocity (m/s)	Calculated drag (N) $C_D = 1.05$	$C_D = 2.05$	Calculated tension(N) $C_D = 1.05$	$C_D = 2.05$	Simulated tension (N)
0	0	0	643	643	628
1	61	119	704	763	626
2	245	479	888	1122	747
3	552	1077	1195	1721	1318
4	981	1916	1625	2559	2388
5	1533	2994	2177	3627	3901

Table 4.1: Calculated drag and tensions in the tether between the surface vessel and the hanging ROV. Results are from both calculation and simulation. Simulated tensions are an average over a 60s period.

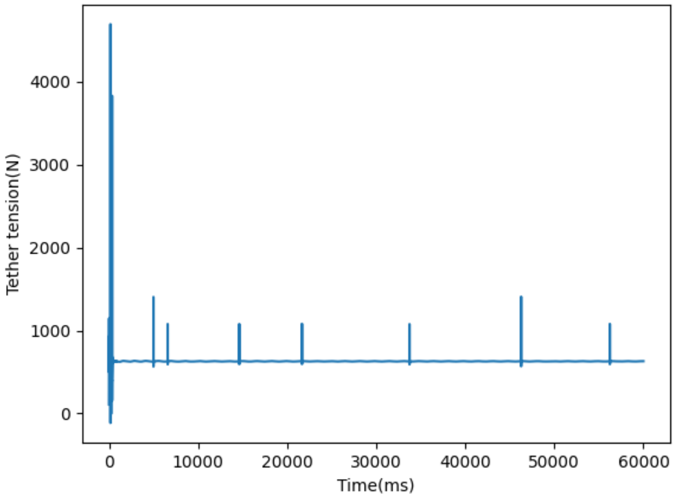
Velocity (m/s)	Deviation($C_D = 1.05$)	Deviation($C_D = 2.05$)
0	-2.4%	-2.4%
1	-12.5%	-21.9%
2	-18.9%	-50%
3	10.7%	-30%
4	32.0%	-7.2%
5	55.8%	7.1%

Table 4.2: Discrepancies between the calculated tension and the simulated tension from table 4.1. Negative numbers indicate that simulated tension is lower than calculated tension

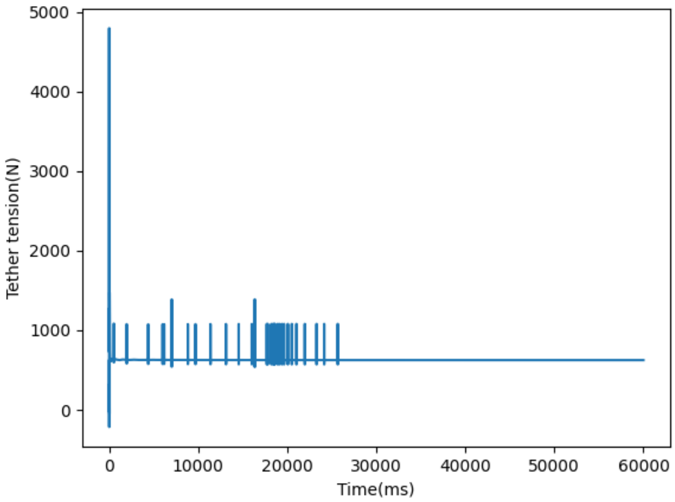
square prism perpendicular to flow is 2.05. The ROV is simulated as a simple box which is roughly half of a cube, divided horizontally. I believe the actual coefficient of drag on the simulated ROV will be somewhere between these and I will use both values to calculate an upper and lower bounds. For best results, either CFD analysis or physical experiments on an equivalent shape could be done.

The dragging of the ROV behind will cause it to no longer be directly beneath the surface vessel. This means that the force of gravity and the force of drag will not be acting in-line with the tether. This is no matter, as the total force will still have to be carried by the tether, so off-axis forces will not impact this. The ROV lagging behind will however lead to it being easier to tumble, changing its forward facing area which would affect the drag calculations.

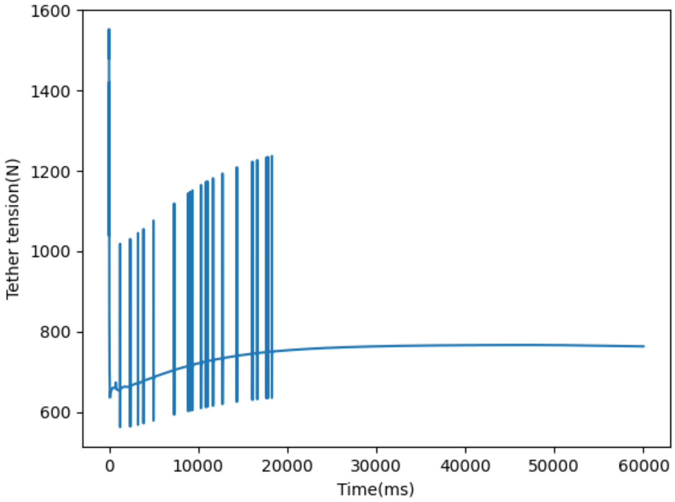
That only leaves the velocity as a variable. I will do simulations in increments of 1m/s from 0 to 5m/s, roughly 10 knots. The simulated tensions will be the average of tensions over a 60s period. This period is chosen so that the system is allowed to stabilize. The calculated and simulated tensions can be seen in table 4.1. Graphs of tensions can be seen in fig. 4.2.



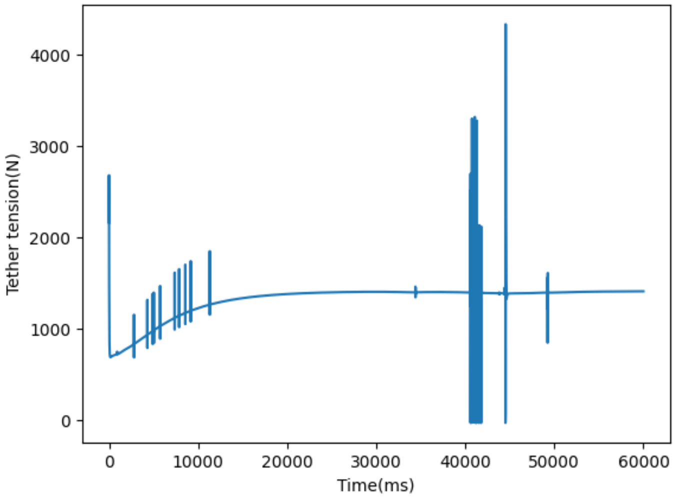
(a) $v = 0$



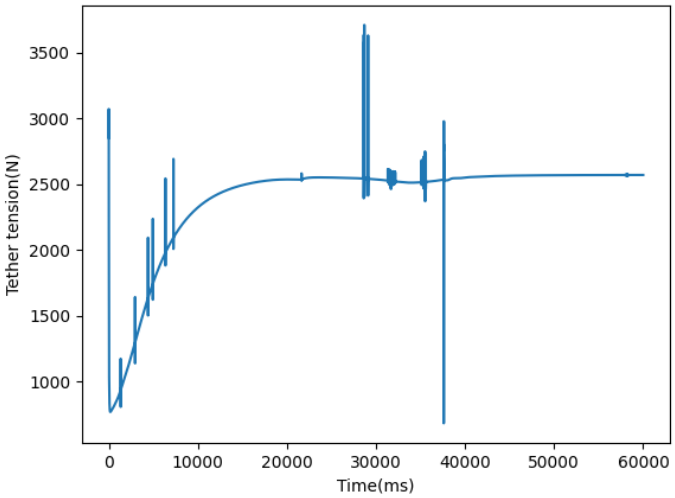
(b) $v = 1\frac{m}{s}$



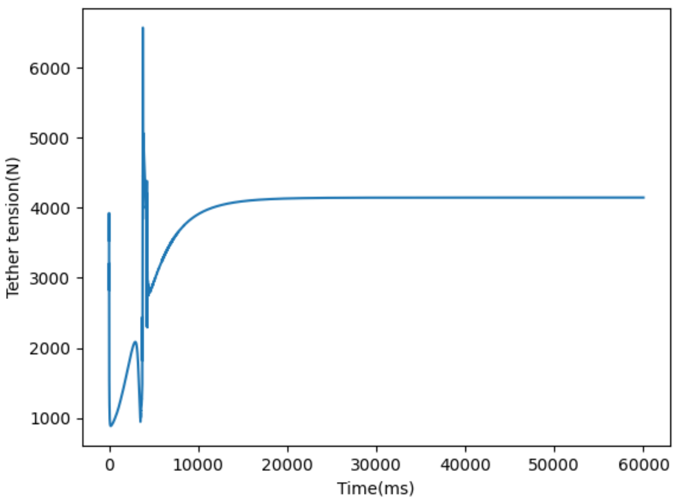
(c) $v = 2\frac{m}{s}$



(d) $v = 3\frac{m}{s}$



(e) $v = 4\frac{m}{s}$



(f) $v = 5\frac{m}{s}$

Figure 4.2: Graphs of tension in the towing tether at different towing speeds. The graphs are spiky at points, this is further discussed in section 5.1.1

4.3 Control system results

The control system has been implemented as a simple PD controller. The weights for the controller are set to

$$k_p = m_{\text{surf}}\omega^2$$
$$k_d = 2m_{\text{surf}}\zeta\omega$$

Where m_{surf} is the mass of the surface vessel, ω is a damping frequency and ζ is the damping ratio. Through experimentation, I found that $\omega = 0.45$ and $\zeta = 1.2$ gave good results for this vessel.

The way the controller is implemented it is able to take in a list of several targets and will treat them as individual targets sequentially. Once the vessel has reached one within a given acceptable error (currently set to 0.1m), the next target in the list is selected and the vessel moves towards it. The target selection finds not only if the current step has achieved the goal but also checks the errors for the last 50 timesteps. This is done so that the vessel couldn't hypothetically run straight through the waypoints, and instead has to actually come to a stop (or close to it) at the target positions. The number 50 was chosen arbitrarily. While there is a simple heading controller implemented, heading error is not taken into account with whether the vessel has reached the target or not.

One simulation with the control system was done with 4 waypoints. The vessel moved from the starting point at (0,0) to (10,15) to (50, -30) to (-20,10) and back to (0,0). The route has been illustrated in fig. 4.3. These points were chosen arbitrarily and were chosen because they are a fair distance away from each other. In total, the theoretical shortest distance to travel would be roughly 180m. The time taken for the simulated vessel to move was 208s, or approximately 3.5 minutes. The total error can be seen in fig. 4.4.

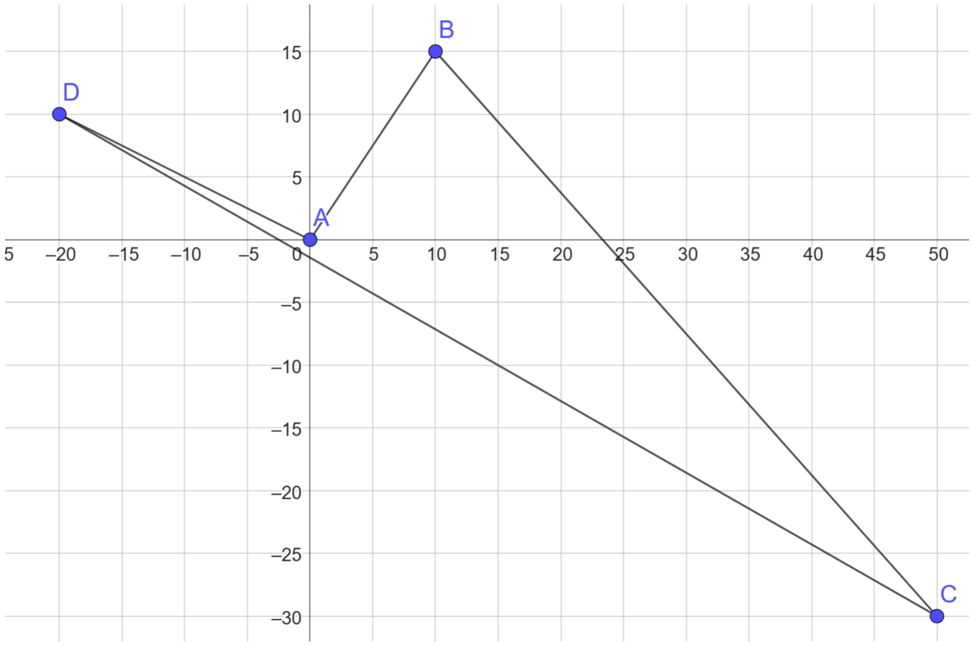


Figure 4.3: The route the vessel was controlled through. Starting at A it moved through B, C and D before going back to A. Returning to start is not a requirement of the controller but was chosen as a waypoint

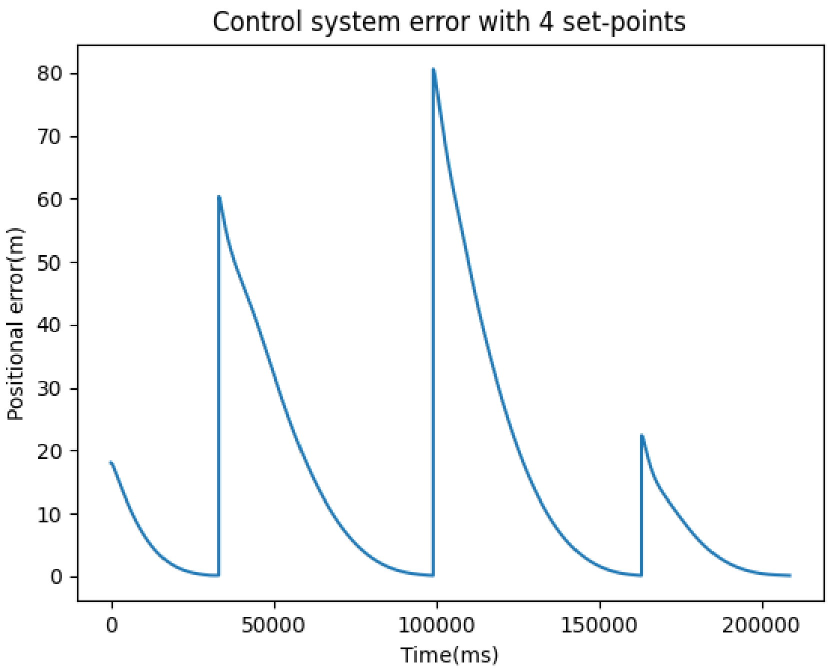


Figure 4.4: Total error for the vessel between the four waypoints

CHAPTER

FIVE

DISCUSSION

5.1 Validation of the simulation

In section 4.2 I presented two methods of validating the simulator. The intuitive and analytical methods. The intuitive validation I have already explained there. The results of the intuitive validation I did I believe show that the simulator at least "acts right". Any flaws that it may have are not so obvious that they are visible without analysis. I will here try to do some further analysis here below of the tension data.

5.1.1 Tension data

Looking at the results from table 4.1 and fig. 4.2, I believe the simulator can be called valid for this use-case. In all cases, except for the fastest one, the simulated tension falls between or below the assumed drag values. The deviations from the calculated values are shown in table 4.2. The cause of the deviations are likely because of the way I've done the calculations; ignoring tumbling and using a "best guess" value for the drag coefficient. Looking at the actual simulation for the highest speed I can confirm that my assumptions will be completely off, as the ROV in that case is practically surfaced as it is dragged behind the surface vessel, see fig. 5.1. The cross-sectional area used in the calculations will be far off from the real values here, as it's almost the "top" surface of the ROV that's facing the direction of travel. Another point to note is that at these high speeds the surface vessel tips so far forward it gets caught up by the hydrodynamic forces at the water's surface and flips over. I believe this happens because the tether is not perfectly centred on the body of the surface vessel, but it is also somewhat a bad sign for such high speeds with this vessel. It indicates that stability analysis at different speeds should be performed. When it comes to the vessel's tipping at higher speeds and validity of the results: the velocity with which the tether is dragged is set and independent of the surface vessel. The vessel is being given a speed with infinite acceleration and is simply along for the ride.

One comment about the data from fig. 4.2 is that the data is spiky at points. I'm not sure what has caused this, but I believe it to be instabilities in the simulation. I have

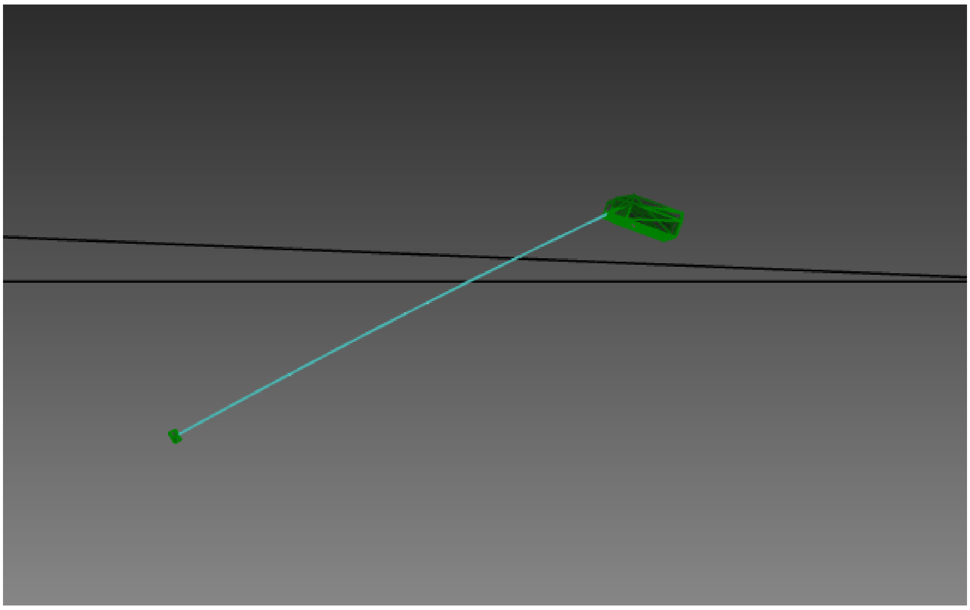


Figure 5.1: The ROV shown dragged behind the surface vessel at $v = 5\frac{m}{s}$

not studied the individual data points, but I am fairly certain that the spikes only last for one or two timesteps. Using a windowed smoothing filter with a fairly large window, maybe 100 samples, will most likely remove the spikes without affecting the data much.

What this dragging simulation tells us is that at this depth, the ROV is not usable. This can be a useful tool to gauge what speeds the surface vessel will be able to move without adversely affecting the ROV underwater. For example, in fig. 2.2 the surface vessel is also moving at $v = 5\frac{m}{s}$, but the ROV is not as affected because the tether damps more when it is longer. The results also show that the analytical method provides an overestimate for the force that the tether will be required to endure under low-speed conditions. This can be used as a tool to find suitable tether materials.

Further, looking at the two first simulation cases they show a very close relationship with the analytical method. At $v = 0$, the simulator has 2.4% less tension than the analytical method expects. This can be because the tether has an impact. If the tether is slightly non-buoyant, the ROV would experience a smaller tension than the surface vessel. This is because the surface vessel would carry the weight of both the tether and the ROV, while the ROV will not "notice" the tether in its connection.

Looking at $1\frac{m}{s} \leq v \leq 4\frac{m}{s}$ the tension is lower than calculated. Looking at the graphical output of the simulation, the suspicion mentioned in section 4.2.2 is confirmed and the ROV is indeed tumbling a bit, presenting a sharp angle towards the direction of travel instead of the blunt front. This would lower the coefficient of drag significantly. For instance, an angled cube, presenting a corner towards the stream rather than a face, has $C_D = 0.80$ compared to a face-on cube with $C_D = 1.05$.

All in all, I believe the differences in results between the analytical method and the simulated method can be seen as a combination of tumbling and angling. By angling I mean presenting a sharper angle towards the stream, and by tumbling I mean presenting the larger top face instead of the smaller front face. Angling would decrease the coefficient of drag while tumbling would increase the forward facing area. I believe these two effects explain the discrepancies seen. Further, I believe that the simulator provides more accurate data than the analytical method could as I have used it. I believe this because the simulator is able to account for angling, tumbling and other effects which are much harder to implement in the closed-form equations I've used.

5.2 Use as a rapid prototyping tool

The simulator shows a lot of promise as a rapid prototyping tool. As it stands at time of writing, it is fairly easy to configure. Changing for example initial conditions, depths of water/length of cable or the type and tuning of controller is very simple. Configuring the seastate is somewhat more complex as it's necessary to describe an elevation function for the sea surface. Currents and waves are not currently implemented, but waves should be fairly simple to implement given an elevation field, and judging by the rest of the framework I doubt currents are much more difficult. AGX allows for currents using the `WaterFlowGenerator` interface.

5.3 Future work

Currently only the surface vessel has a control system implemented and this control system is a waypoint style system. The ROV is only a hanging weight as implemented right now. In the final implementation of the Plan Sea project, it's likely that one of the vessels will be actively following the other to allow for the desired range of motion or lifting strength. Thus further development on the control side of the simulator is necessary.

The tether connecting the surface vessel and the ROV is currently a fixed length. AGX does have methods of simulating winches, including their internal forces, required torques and other elements. This implementation should be considered a high priority to make the simulation closer to reality and allow for more dynamic states to be simulated.

Further, there are no environmental effects as the simulator stands today. Implementing especially currents would be essential to getting a model that can be instructive about the real-life applicability of the controllers designed. Waves would also be important to implement, along with a heave-compensation system to allow the ROV to maintain altitude even as the surface vessel is moving vertically.

The simulator right now requires a knowledge and understanding of both Python and AGX to configure and run. It would be nice if in the future it is possible to configure basic functions of the simulator through a graphical user interface as opposed to through code. I believe this should be a lower priority to implement, as currently the simulator is a development tool, but it would be nice to have.

Another nice-to-have is a better model of the surface vessel and the ROV. More detailed models would allow for a more accurate hydrodynamic simulation to be run. Higher detail would also increase the complexity of the model though, so it should only be implemented to a certain point. I believe that better models should also be considered a low priority, as the approximations taken right now are likely accurate enough, within one order of magnitude or so.

For my Master's thesis, I would like to work further on this simulator to create a well functioning control system and then apply it to the real-life vessel and ROV. I would implement some of these changes, as well as deal with the physical modifications of the real-life vessel and ROV.

CHAPTER

SIX

CONCLUSIONS

The stated goal of the project has been to make a simulation of the Plan Sea project's proposed implementation. The project consists of a small surface vessel and a non-buoyant remotely operated vehicle attached to each other by a lifting wire. The simulation has been made to allow for control system design, as well as to act as an engineering tool for this specific project, allowing different sizes and types of wires to be used, allowing for changing sea-state and currents, or the type and size of vessels.

In this report I've documented why the simulation would be a helpful tool for prototyping. I've also described some of the reasons why I believe simulation is more helpful for this case than finding analytical solutions to the problems posed by the project.

I have created the simulation and run some validation tests on it to see whether it acts close to as should be expected from a realistic simulation. I have also created a simple PD-control system which is able to position the surface vessel at desired points in the world-space. The results found show that the simulation is likely more accurate than the simple analytical methods used to estimate the forces that would act on the system.

I have proposed a list of topics for future iterations of the simulator, as well as stating my goal of continuing work on the simulator to use it for further development of the Plan Sea project.

In all, I would count this project a success. The simulator has been created and validated, and is now ready for further processing and use in the Plan Sea project.

BIBLIOGRAPHY

[1] Jenna R. Jambeck et al. "Plastic waste inputs from land into the ocean". In: *Science* 347.6223 (Feb. 2015). Publisher: American Association for the Advancement of Science, pp. 768–771. DOI: 10.1126/science.1260352. URL: https://www.science.org/doi/10.1126/science.1260352 (visited on 06/12/2024).

[2] World Economic Forum. *Top 25 recycling facts and statistics for 2022*. en. June 2022. URL: https://www.weforum.org/stories/2022/06/recycling-global-statistics-facts-plastic-paper/ (visited on 06/12/2024).

[3] Atsuhiko Isobe and Shinsuke Iwasaki. "The fate of missing ocean plastics: Are they just a marine environmental problem?" In: *Science of The Total Environment* 825 (June 2022), p. 153935. ISSN: 0048-9697. DOI: 10.1016/j.scitotenv.2022.153935. URL: https://www.sciencedirect.com/science/article/pii/S0048969722010270 (visited on 06/12/2024).

[4] Joana Correia Prata et al. "Environmental exposure to microplastics: An overview on possible human health effects". In: *Science of The Total Environment* 702 (Feb. 2020), p. 134455. ISSN: 0048-9697. DOI: 10.1016/j.scitotenv.2019.134455. URL: https://www.sciencedirect.com/science/article/pii/S0048969719344468 (visited on 15/12/2024).

[5] Natalia Zolotova et al. "Harmful effects of the microplastic pollution on animal health: a literature review". en. In: *PeerJ* 10 (June 2022). Publisher: PeerJ Inc., e13503. ISSN: 2167-8359. DOI: 10.7717/peerj.13503. URL: https://peerj.com/articles/13503 (visited on 15/12/2024).

[6] Mariana Segovia-Mendoza et al. "How microplastic components influence the immune system and impact on children health: Focus on cancer". en. In: *Birth Defects Research* 112.17 (2020). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/bdr2.1779, pp. 1341–1361. ISSN: 2472-1727. DOI: 10.1002/bdr2.1779. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/bdr2.1779 (visited on 15/12/2024).

[7] Gloria Ukalina Obuzor and Uchechi Bliss Onyedikachi. "Chemical Leaching into Food and the Environment Poses Health Hazards". en. In: *Modernity in Health and Disease Diagnosis: The Account from STEM Women*. Ed. by Eucharia Oluchi Nwaichi. Cham: Springer Nature Switzerland, 2023, pp. 129–148. ISBN: 978-3-031-34963-8. DOI: 10.1007/978-3-031-34963-8_14. URL: https://doi.org/10.1007/978-3-031-34963-8_14 (visited on 15/12/2024).

[8] Ibukun Oluwoye et al. "Degradation and lifetime prediction of plastics in subsea and offshore infrastructures". In: *Science of The Total Environment* 904 (Dec. 2023), p. 166719. ISSN: 0048-9697. DOI: 10.1016/j.scitotenv.2023.166719. URL:

https://www.sciencedirect.com/science/article/pii/S0048969723053445 (visited on 06/12/2024).

[9] Walter Lian and Bjem Sortland. "Manoeuvering of Bodies Suspended at Extreme Water Depth". en. In: Los Angeles, May 1996. (Visited on 10/12/2024).

[10] Peng Chen et al. "Dynamic characteristics of deep-sea ROV umbilical cables under complex sea conditions". In: *Ocean Engineering* 239 (Nov. 2021), p. 109854. ISSN: 0029-8018. DOI: 10.1016/j.oceaneng.2021.109854. URL: https://www.sciencedirect.com/science/article/pii/S002980182101204X (visited on 19/09/2024).

[11] Thomas Thuesen Enevoldsen et al. "Simplified Modelling and Identification of an Inspection ROV". en. In: *IFAC-PapersOnLine* 51.8 (2018), pp. 257–262. ISSN: 24058963. DOI: 10.1016/j.ifacol.2018.06.386. URL: https://linkinghub.elsevier.com/retrieve/pii/S240589631830716X (visited on 19/09/2024).

[12] Enrico Anderlini, Gordon G. Parker and Giles Thomas. "Control of a ROV carrying an object". en. In: *Ocean Engineering* 165 (Oct. 2018), pp. 307–318. ISSN: 00298018. DOI: 10.1016/j.oceaneng.2018.07.022. URL: https://linkinghub.elsevier.com/retrieve/pii/S0029801818312654 (visited on 19/09/2024).

[13] Per J. Thingstad and Haakon D. Hveding. "A Nonbuoyant ROV for Performing Heavy Subsea Work". en. In: *Offshore Technology Conference*. Houston, Texas: Offshore Technology Conference, 1982. DOI: 10.4043/4182-MS. URL: http://www.onepetro.org/doi/10.4043/4182-MS (visited on 19/09/2024).

[14] Algoryx. *AGX Dynamix User Manual*. URL: https://www.algoryx.se/download/installers/AGX_Dynamics_UserManual.pdf (visited on 15/12/2024).

[15] *BlueROV2 Heavy Configuration Retrofit Kit*. en-US. URL: https://bluerobotics.com/store/rov/bluerov2-upgrade-kits/brov2-heavy-retrofit/ (visited on 14/12/2024).

[16] *FordypOgMaster/Simulator at main · MagnusKjorseng/FordypOgMaster*. en. URL: https://github.com/MagnusKjorseng/FordypOgMaster/tree/main/Simulator (visited on 14/12/2024).