Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Engineering
Department of Marine Technology

Thomas Erling Lone

NAVIGATION TECHNIQUES FOR UNDERWATER VEHICLES IN POLAR REGIONS

Master's thesis in Marine Technology
Supervisor: Asgeir J. Sørensen
June 2020

NTNU
Norwegian University of
Science and Technology

Master thesis

June 2020

Department of Marine Technology

NAVIGATION TECHNIQUES FOR UNDERWATER VEHICLES IN POLAR REGIONS

Supervisor
Prof. Asgeir J. Sørensen
AMOS, NTNU
asgeir.sorensen@ntnu.no

Co-Supervisor
Jens E. Bremnes
PhD Candidate, NTNU
jens.e.bremnes@ntnu.no

Co-Supervisor
Håvard S. Løvås
PhD Candidate, NTNU
haavasl@stud.ntnu.no

Author
Thomas Erling Lone
thomaelo@stud.ntnu.no

NTNU
Norwegian University of
Science and Technology

NTNU Trondheim
Norwegian University of Science and Technology
Department of Marine Technology

MASTER THESIS IN MARINE CYBERNETICS

SPRING 2020

FOR

STUD. TECHN. THOMAS LONE

NAVIGATION TECHNIQUES FOR UNDERWATER VEHICLES IN POLAR REGIONS

Work description (short description)

The main objective of this thesis is to propose methods for autonomous navigation under ice using underwater hyperspectral imaging (UHI) in conjunction with standard navigation sensors such as GPS, DVL, USBL, IMU and magnetic compass. Specifically, the algorithm should be able to evaluate the utility of the vehicle's data measurements, and then control the vehicle's altitude relative to the ice accordingly. Furthermore, emergency strategies such as under ice abort maneuvers and risk management will be discussed. The methods will be tested in a simulation environment using Simulink to model an UHI sensor for altitude estimation along with the other relevant navigation sensors. An observer will also be implemented to aid the navigational system. The vehicle of focus is an autonomous underwater vehicle (AUV). If successful, the methods will be further tested on a hardware setup involving an UHI sensor and two Blueye robots either under lake-ice or simply over a sandy sea bottom. The thesis is a continuation of the previous project thesis conducted the fall of 2019.

Scope of work

1. Review literature within the field of hyperspectral analysis and navigation for autonomous marine platforms with special emphasis on planning and risk analysis.
2. Design and implement models for UHI and other relevant navigation sensors in Simulink.
3. Design and implement a sensor based observer to estimate the AUV's state based on input from the sensor models:
    a. Design for a nonlinear system with 6 degrees of freedom.
    b. Implement an extended Kalman filter (EKF) with real-time linearization of the nonlinear state space model.
4. Propose a method that uses the UHI data to assess the AUV's risk and reward. Furthermore, use this information to find a desired altitude for the AUV.
5. Look into high level planning methods and propose a model predictive controller (MPC) to minimize risk and energy consumption. Other examples are temporal logic, Markov Decision Process (MDP) and contingency handling systems.
6. Perform experiments to test the performance of a UHI aided navigation system under ice for a twin-Blueye setup.

The report shall be written in English and edited as a research report including literature survey, description of mathematical models, description of control algorithms, simulation results, model test results, discussion and a conclusion including a proposal for further work. Source code should be provided. It is supposed that Department of Marine Technology, NTNU, can use the results freely in its research work, unless otherwise agreed upon, by referring to the student's work.

The thesis should be submitted within 15th of July 2020.

Co-Supervisor: Håvard S. Løvås, Jens E. Bremnes

Professor Asgeir J. Sørensen
Supervisor

Abstract

The main focus of this thesis has been implementation and enhancement of current navigation systems for AUVs in polar regions. Recent human caused global issues such as global warming motivates for deeper research in the areas as well as environmental monitoring. AUVs are a popular sensor platform for such missions, but need to operate as robustly and safely as possible while saving costs at the same time. One of the major risks when operation an AUV is losing the vehicle, and this risk increases in polar regions due to worsened operational conditions, the presence of sea ice and reduced navigational capabilities.

This thesis proposes three contributions to the Arctic AUV simulator developed at NTNU by [28].

Firstly, models for all AUV navigation sensors have been implemented with an associated observer. Specifically, heading estimation has been discussed in detail and in particular, an algorithm has been proposed to model and compensate for magnetic compass declination given the geographical location of the AUV mission.

Secondly, a model predictive controller (MPC) has been implemented to improve altitude control under sea ice. This controller is based on the AUV kinematics.

Finally, an algorithm is proposed to simulate the use of an UHI sensor in estimating sea ice altitude. Some basic methods and criteria for mission abort are also included.

The results include simulations validating the performance of the stand alone parts as well as the overall system putting it all together. Both the UHI method and the MPC manage to successfully navigate the AUV along the desired path, but both methods suffer from the lack of information about the terrain ahead.

Further work of interest include creating models that gather information about the surrounding environment to exploit the full potential of the MPC and test the proposed methods on a physical hardware system with two Blueye ROVs.

Sammendrag

Hovedfokuset for denne masteroppgaven har vært å forbedre navigasjonssystemer for AUVer og tilpasse dem polare strøk. Metodene er implementert og testet i simulering. Menneskeskapte utfordringer som global oppvarming motiverer for mer omfattende forskning i slike områder i tillegg til overvåkning av miljøet. AUVer representerer en populær sensorplattform for slike formål, men må evne å operere så robust og sikkert som mulig og samtidig være prisdyktig. En av de største risikoelementene ved bruk av AUV er å miste fartøyet, og denne risikoen er betraktelig høyere i polare strøk grunnet utfordrende driftsforhold, tilstedeværelsen av sjøis og ikke minst nedsatt evne til å navigere nøyaktig.

Denne oppgaven inkluderer tre bidrag til en Arktisk AUV simulator utviklet på NTNU av [28].

Først ble det implementert modeller for navigasjonssensorene til AUVen med en tilhørende observator. Her ble også estimering av kompasskurs diskutert i detalj og mer spesifikt ble det utviklet en algoritme for å kompensere for avvik mellom geografisk og magnetisk Nordpol basert på den geografiske lokasjonen til AUVen.

En model predictive control (MPC) ble så implementert for å forbedre avstandskontrolleren mellom AUVen og sjøisen. Denne kontrolleren er basert på AUVens kinematiske modell. Sist men ikke minst ble det utviklet en algoritme som simulerer bruken av en UHI sensor for å estimere avstanden mellom AUVen og isen. Videre diskuterer oppgaven kort grunnleggende metoder og kriterier for å avbryte operasjonen.

Resultatene inkluderer simuleringer som validerer ytelsen til de separate bidragene så vel som hele systemet satt sammen. Både UHI metoden og MPCen klarte å navigere AUVen med suksess langs referansesignalet, men begge metodene mangler informasjon om det kommende terrenget.

Videre arbeid av interesse omfatter å lage modeller for å bedre informasjonbildet av det omliggende terrenget og dermed ta i bruk det fulle potensialet til MPC samt å teste de foreslåtte metodene på et fysisk system med to Blueye ROVs.

Preface

This master thesis concludes the work carried out in the spring 2020 and marks the end of the five year Master of Science in Marine Technology at the Norwegian University of Science and Technology.

The work has been carried out independently, always with the aim of creating original ideas and innovative implementations. Much of the academic foundation for this work was created as an exchange student to Massachusetts Institute of Technology. As part of both the ocean engineering department and the computer science department I laid a strong platform for further work within marine navigation and controls.

Furthermore, I have conducted several private expeditions to remote and extreme places parallel to my studies such as the summit of Mount Everest and the Seven Summits. This has made me realise how beautiful and vulnerable nature truly is. Hence, I have developed a keen interest in exploring the polar regions, both out of fascination and with a long term goal of contributing to a more sustainable future with better knowledge about climate change and global warming.

Thanks to my supervisor, Professor Asgeir J. Sørensen, I was a visiting student at UNIS, Svalbard in the fall of 2019 which lay the foundations for selecting the topic of this thesis. Through the work of this thesis I have been under supervision of Prof. Sørensen and the two NTNU PhD candidates Jens Bremnes and Håvard Løvås. Their guidance and assistance have been of great help both in terms of collaboration and support.

Originally, the ambition was to build and test a proper hardware setup of the proposed implementations in this thesis, but unfortunately this was not feasible due to the global pandemic of Covid-19.

It has been an honor and a privilege to be under Prof. Sørensens supervision, and I am looking with excitement at what the Centre for Autonomous Marine Operations and Systems (AMOS) and his team will accomplish in the years to come.

Thomas Erling Lone

June 30th, 2020

Acknowledgements

I would like to thank my supervisor Professor Asgeir J. Sørensen for excellent guidance and assistance in providing the necessary resources to complete this thesis to the best of my ability.

Furthermore, Prof. Sørensen has given me generous flexibility and allowed me to pursue my lifelong dream of climbing Mount Everest and The Seven Summits parallel to my studies as well as aim for reaching the poles in the years to come. Additionally, Prof. Sørensen's role in the research community and entrepreneurship has been especially motivating for further work and given me the inspiration to consider pursuing a PhD after graduation.

PhD candidate Jens E. Bremnes has been a wonderful resource from the beginning of this project, providing background material and a simulator for testing my implementations. I appreciate our fruitful discussion on underwater application and navigation, especially in the context of polar research and risk management. Furthermore, I would like to thank Bremnes for patiently reading through the thesis, providing assistance with structure and readability. PhD candidate Håvard S. Løvås and Ecotone AS have both been important resources with respect to the new and promising UHI technology. I am grateful for their input, and I am excited to follow their future development.

Table of contents

List of Figures

List of Tables

Abbreviations

AI	Artificial intelligence
$AMOS$	Centre for Autonomous Marine Operations and Systems
API	Application programming interface
AUV	Autonomous underwater vehicle
CO	Center of origin
CTP	Conductivity temperature pressure
$DGPS$	Differential Global Positioning System
DOF	Degree of freedom
DP	Dynamic Positioning
DVL	Doppler velocity log
EKF	Extended Kalman Filter
FFI	Norwegian Defence Research Establishment (Forsvarets Forskningsinstitutt)
GNC	Guidance, navigation and control
$GNSS$	Global navigation satellite system
GPS	Global positioning system
$HiPAP$	high precision acoustic positioning system
HRI	Human-robot interaction
IMU	Inertial measurement unit
INS	Inertial navigation system
KF	Kalman filter
LBL	Long base line
LoA	Level of autonomy
LOS	Line-of-sight

LP	Linear Programming
LTI	Linear time-invariant
MBE	Multibeam echosounder
MDP	Markov Decision Process
$MEMS$	Microelectromechanical systems
$MIMO$	Multiple input multiple output
MPC	Model Predictive Control
NED	North-East-Down
NLP	Nonlinear programming
$NTNU$	Norwegian University of Science and Technology
OCP	Optimal control problem
OOI	Object of interest
PD	Proportional-derivative
PI	Proportional-integral
PID	Proportional-integral-derivative
PPT	Parts per thousand
RGB	Red green blue
$SLAM$	Simultaneous localization and mapping
SSS	Side Scan Sonars
SUV	Super underwater vehicle
UHI	Underwater hyperspectral imaging
$UNIS$	The University Centre in Svalbard
$USBL$	Ultra-short base line
UUV	Unmanned underwater vehicle
ROV	Remotely operated vehicle

Chapter 1

Introduction

The objective of this thesis is to assess and enhance different underwater navigational techniques and modify them to fit the additional challenges faced in polar conditions. The thesis will emphasise on heading estimation, sensor models with observers, model predictive control (MPC) and the potential use of underwater hyperspectral imaging (UHI) as both a payload and navigational sensor under sea ice.

This chapter will provide relevant background information and motivation for the project contributions.

1.1 Motivation

The polar regions are among the most fragile and vulnerable areas on the planet. The increasing number of human caused global issues require new profitable and sustainable solutions. To find such solutions, it is vital to conduct ocean research, monitor the environment and investigate future technological opportunities in the polar regions. This requires vehicles robust enough to deal with the extreme and challenging environment surrounding the poles, and at the same time operate within a sufficient risk limit with respect to humans, environment, assets and reputation.

Primary production in the oceans is contributing to the majority of worldwide photosynthesis related oxygen production [27]. Recently, scientists discovered microplastics in the Arctic snow and oceans that allegedly have been carried there by winds [22]. These are examples of fields that may be measured and monitored by use of autonomous underwater vehicles (AUVs), and the results may help to find answers on how to tackle the global challenges we face.

Cost reduction, smaller environmental impact and increased safety have for decades motivated using AUVs to conduct such data gathering in the oceans. However, there is always potential for improvement, and today's most advanced AUVs still largely depend on pre-programmed missions and the support of a comprehensive human team.

For the vehicle to maneuver safely and complete the mission objectives it requires a high accuracy navigation system. Thanks to the invention of microelectromechanical systems (MEMS), tiny lightweight sensors are now both affordable and accurate. However, due to small errors in MEMS measurements, the inertial navigation system (INS) will drift unbounded [13]. This motivates using an observer algorithm minimizing the drift as well as aiding the INS using other sensors and methods.

Finally, new and exciting sensors and hardware such as UHI are entering the marked. Such sensors may become a paradigm shift in allowing the underwater vehicle to assess the quality of its measurements real time. Such information may be used by the navigational system to avoid unnecessary risk, but simultaneously allow the vehicle to take more risk when the reward of obtaining high quality data is sufficiently high. This thesis will propose methods for using an UHI sensor to estimate the distance between an unmanned underwater vehicle (UUV) and the sea ice which from now on will be referred to as the *altitude*.

1.2 Underwater navigation in polar regions

The polar regions represent some of the most challenging conditions for research and data gathering on the planet. There exists close to no infrastructure, and the harsh weather and cold temperatures create challenges both to humans and equipment. Most research voyages are vessel based, e.g. an icebreaker, that transports both humans and equipment to the relevant area. Various sensor-platforms may be used to gather data such as a remotely operated vehicle (ROV), unmanned surface vehicle (USV), unmanned aerial vehicle (UAV) or an AUV. The most suitable sensor platform should be chosen based on the mission parameters. An overview of temporal and spatial range for some typical sensor platforms are described in Chapter 9 of [36] and can be seen in Figure 1.1.

In addition to extreme polar conditions, the sensor platforms also have to overcome additional challenges related to navigation that are not as profound closer to the equator. Heading estimation using either magnetic or gyroscopic compassing techniques has a particularly decreasing performance with latitude [14] as described in Section 2.2. Furthermore, scientist often want to gather data close to the sea ice which increases the risk of damaging or losing the vehicle in contact with the ice. If the navigation accuracy becomes too poor, it can also become challenging to recover the vehicle post mission.

The most relevant challenges related to navigation in polar regions are as follows:

- Difficult operational conditions in terms of low temperatures, harsh weather, ice, remoteness and winter darkness.
- Decreased accuracy for both magnetic and gyroscopic compassing.
- Stricter accuracy requirements due to higher operational risk.

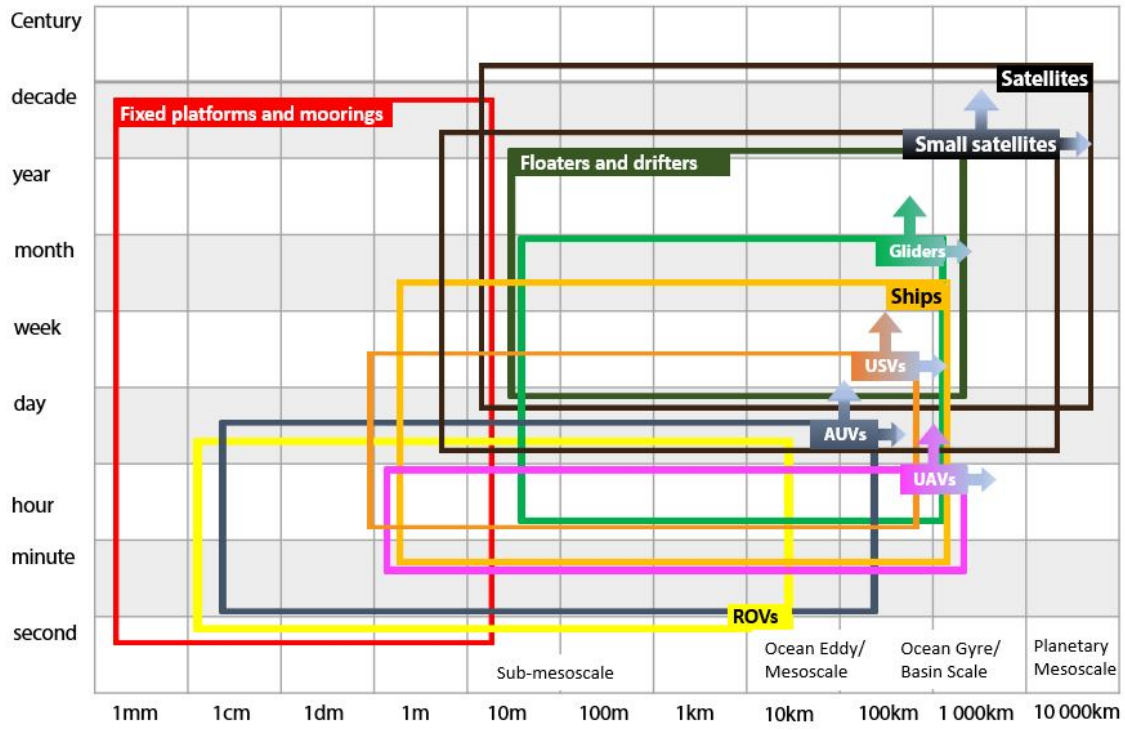


Figure 1.1: Showing the spatial and temporal range of some typical platforms, illustration courtesy of [36].

1.2.1 Unmanned underwater vehicles

An UUV may be used to complete various research missions and marine operations with a varying level of autonomy and human-robot interaction (HRI). They come in different sizes and price ranges with varying attributes such as payload capacity, spatial and temporal range, navigational system and level of autonomy (LoA).

It is normal to separate underwater vehicles into ROVs and AUVs.

Remotely operated vehicle

These vehicles are controlled remotely by an operator, normally sitting on a topside such as a dynamical positioning (DP) vessel. They may carry their own power source, but normally they are connected through an umbilical cable providing them with sufficient power and data communication capabilities. Thus, a major advantage with ROVs is their ability to carry heavy payloads such as tools for subsea operations [8]. However, due to the drag from the umbilical, ROVs have limited spatial range in the horizontal plane.

Autonomous underwater vehicle

An AUV may operate independently of an operator to fully carry out its mission. AUVs may have a varying LoA, but the future aim is that AUVs shall be able to plan a mission and replan real time without the need of human interaction. However, AUVs are usually given specific waypoints and follow them using a guidance and control system with human supervision. The lack of an umbilical cable increases the risk of losing the vehicle, especially during under ice operations. An illustration of a typical AUV with payload and navigation sensors is shown in Figure 1.2.

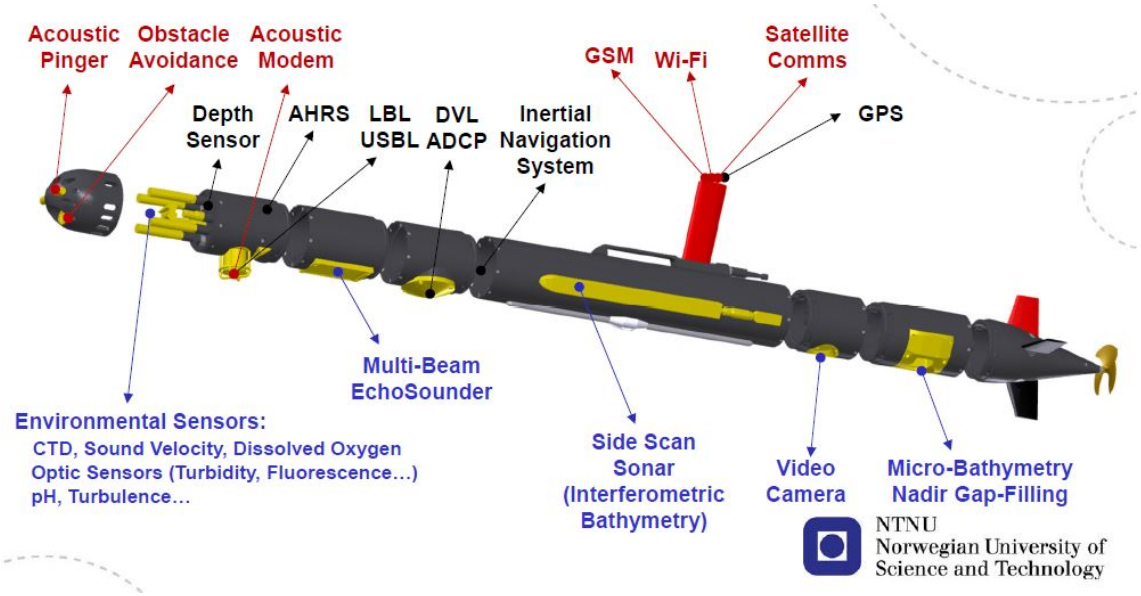


Figure 1.2: An illustration of the typical sensors on board an AUV relevant both for navigation and data sampling (courtesy of AMOS).

AUVs will be the vehicle of focus for the rest of this project.

1.2.2 Definition of autonomy

To categorize different platforms it is useful to define autonomy. The literature represent several definitions of autonomy, mostly based on the level of HRI as well as mission and environmental complexity [18]. As these definitions vary between researchers and represent a series of terms it is more appropriate to define certain states, or classes, of autonomy. The Uninhabited Combat Air Vehicle Program defines four levels of autonomy [7], which later was modified by [39] as follows:

1. Automatic operation (remote control) means that the system operates automatically. The human operator directs and controls all high-level mission planning functions, often preprogrammed.

2. Management by consent (teleoperation) means that the system automatically makes recommendations for mission actions related to specific functions, and the system

prompts the human operator at important points in time for information or decisions. At this level the system may have limited communication bandwidth including time delay, due to i.e. distance. The system can perform many functions independently of human control when delegated to do so.

3. Semi-autonomous (management by exception) means that the system automatically executes mission-related functions when response times are too short for human intervention. The human may override or change parameters and cancel or redirect actions within defined time lines. The operators only concern and focus should be directed at exceptions and extraordinary decisions.
4. Highly autonomous means that the system automatically executes mission related functions in an unstructured environment with ability to plan and replan the mission. The human may be informed about the progress. The system is independent and "intelligent".

1.2.3 Sensor capabilities

Different platforms may require different sensors to operate and collect data. However, regardless of platform type it is common to divide between payload sensors used for data gathering for the research objective, and navigational sensors used to ensure satisfying estimation of the platforms state.

Payload sensors

The objective of an underwater platform is to put the payload sensors in a desired location to perform either remote sensing or direct measurements [35]. If the target is moving, there may also be temporal constraints associated with measuring the data. Some examples of typical payload sensors are:

- Camera (RGB)
- Underwater hyperspectral imaging (UHI)
- Conductivity, temperature and pressure sensors (CTP)
- Magnetometers
- Acoustic Doppler Current Profilers
- Active sonars
- Side Scan Sonars (SSS)

Navigational sensors

In order for the platform to perform its mission, it needs sensors to observe its surrounding environment and conduct state estimation. These sensors measure parameters that are

processed and typically sent through an observer in order to estimate the internal states used by the control system to manoeuvre and guide the platform in a safe and reliable manner. Some examples of typical navigation sensors are:

- Acoustic baseline sensors
  - Long Base Line (LBL)
  - Ultra Short Base Line (USBL)
- Doppler velocity log (DVL)
  - Speed over ground
  - Relative speed in water
  - Altitude
- Global navigation satellite systems (GNSS)
- Camera for geophysical navigation
- Pressure sensors
- Heading sensors (e.g. magnetic compass and gyroscope)
- Inertial sensors

Hybrid sensors

Some sensors might provide data both relevant for completing the mission and aiding the navigational system simultaneously. For example, the pictures taken from a down facing camera on an UUV might be of significant importance for researching a wreck, but simultaneously the images might be used in real time as part of a geophysical navigational algorithm to make the UUV more robust and redundant.

This thesis will try to answer if a similar approach can be used with an UHI sensor facing upwards under sea ice. If the UHI data can be used to estimate the distance (altitude) from the sea ice, it might be an important navigational sensor in polar regions while collecting mission data at the same time. Furthermore, the quality of the hyperspectral data can be used real time to assess the level of collision risk and balance this with the data attractiveness.

This will be further addressed in Section 2.3.

1.2.4 Case study 1: AUV simulation in Simulink

To create, tune and validate the performance of the AUV's sensor models (incuding UHI), observer and MPC a simulator is used. The simulation environment was mainly designed to validate AUV guidance systems for iceberg detection and mapping of icebergs using

simultaneous localization and mapping (SLAM) as part of NTNU postdoc Petter Norgrens PhD thesis [28]. The parameters used in the simulation environment were intended to model the American AUV *Remus* depicted in Figure 1.3. An updated version provided line of sight (LOS) guidance for any arbitrary three dimensional path between given waypoints defined by the user.

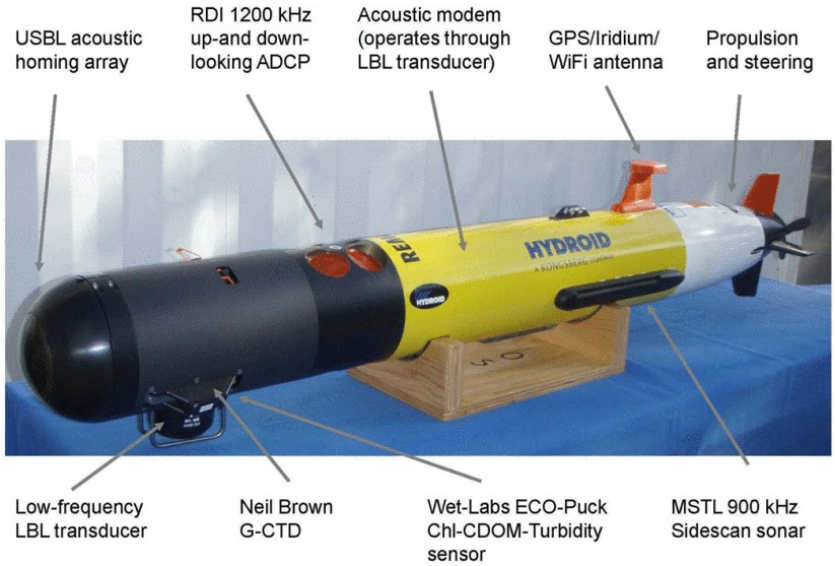


Figure 1.3: The REMUS-100 AUV as delivered from manufacturer. The location of principal systems and sensors is shown as in [23].

The simulation was further modified by NTNU PhD candidate Jens Bremnes [3] to provide an under-ice altitude observer and associated under-ice guidance system allowing the AUV to regulate its distance (altitude) below the ice. The ice samples used in the simulator were gathered on a research mission to Antarctica included in [41].

The majority of work conducted in this thesis has been carried out by use of this simulation environment trying to enhance it and make it even more user friendly for future Marine Cybernetics students at NTNU.

1.2.5 Case study 2: Blueye Robotics

Blueye Robotics is a spin-off company from the research community at NTNU. Their main product is an affordable, lightweight ROV that may be controlled by a smartphone. An example of the product used in the field can be seen in Figure 1.4. Its thrusters and sensors run on a battery package allowing the use of a thin umbilical only transferring data to the operator. It's main commercial use is for professional underwater inspection near the surface, but may also be used by scientists to perform sea bed mapping or look for wildlife. More information may be found through their web page: *https://www.blueye.no/*.

Because the navigational sensors on board the Blueye are comparable to that of an inexpensive AUV, the Blueye might be used as an ideal platform to test AUV navigational, guidance and control systems under ice with low risk of losing the vehicle. There exists a programming interface (API) in Python which makes it possible to attach external sensors

Figure 1.4: An image of a Blueye ROV in Svalbard used by a student at The University Centre in Svalbard (UNIS). Photo: Thomas Lone.

such as an UHI and make them exchange data real time. Phd candidate Håvard Løvås at AMOS has successfully created a twin-Blueye platform where two Blueyes are fixed together with a plate that has a UHI sensor facing down connected directly to the Blueyes. This setup known as as a super underwater vehicle (SUV) has many interesting applications and advantages to traditional platforms. Løvås has in time of writing submitted a paper with more details and information [24].

The platform is a great starting point for testing the UHI sensor as a way of under ice navigation. Simply by turning the UHI sensor upside down (facing upwards) and let the Blueyes run on AUV-like software through the Python API. Thus, despite the Blueyes being ROVs the twin setup with an upward facing UHI sensor could be a inexpensive and safe way to validate the performance of the methods derived in this thesis before conducting actual experiments with AUVs in polar regions.

1.3 Research questions

The main objective of this thesis is to assess and enhance methods for underwater navigation in polar regions, use UHI data to improve payload sensor measurements and MPC to lower risk and improve the control system. More specifically, the aim is to answer the following questions:

1. What specific challenges must be addressed for underwater navigation in polar regions compared to elsewhere?
2. How may different sensor based observer designs enhance the overall navigation system performance?
3. How may UHI data be analysed real time to create an altitude approximation under ice, and how accurate will this be compared to classical methods?
4. Which methods are suitable for controlling the vehicle's distance from the ice with respect to risk/reward and how may an MPC enhance these methods compared to classical control theory?
5. In future work, how may an ROV function as a safer and cheaper platform to test AUV software under sea ice?

1.4 Main contributions

This master thesis has contributed to improving an existing AUV simulator by proposing sensor models, a sensor based observer, and an MPC for altitude. Furthermore, a literature study has been conducted with special focus on heading estimation and polar related challenges as well as a detailed discussion on different approaches to successfully abort under ice operations without loss of vehicle. A specific overview of the project contribution follows:

- Design and validation of sensor models with special focus on polar relevant error models.
- Design and validation of a sensor based observer formulated as an extended Kalman filter.
- Implementation of a MPC for altitude.
- A simplified sensor model for UHI working as an altimeter.
- An algorithm that links the UHI data to the MPC for optimized control allocation.

1.5 Outline

Chapter 2 provides general approaches and methods for underwater navigation with emphasis on heading estimation. The UHI sensor is described in detail with focus on its relevance for under ice operations. The basic principles of MPC are presented along with similar topics of relevance. Furthermore, a mathematical model and control system for a six degrees of freedom (DOF) AUV is included.

Chapter 3 describes the proposed methods and implementations for the simulator and the hardware platform (presented in Section 1.2.5), namely the sensor models, the nonlinear observer, the UHI algorithm and the MPC.

Chapter 4 discusses simulation results illustrating the performance of the sensor models, the sensor based observer, the MPC and the overall system.

Chapter 5 concludes the project in context of the scope of work and presents recommendations for future work.

Appendix A gives numerical values used for the mathematical models of the Remus AUV in the simulator gathered from [3].

Appendix B provides the waypoints creating the lawn mower trajectory used in simulations.

Appendix C presents the controller and guidance gains used for all the simulation runs described in Chapter 4.

Chapter 2

Theory on underwater navigation and control

The intention of this chapter is to provide the necessary theoretical principles for the research contributions in Chapter 3. After a brief overview of approaches to navigation and acoustic technology, heading estimation in polar regions will be discussed in depth along with UHI on its ability to function as a navigational sensor.

The last part of the chapter provides the basic principles of state estimation and observer theory before presenting the mathematical AUV model and the guidance and control system used in the simulator.

2.1 Underwater navigation techniques

The main aim of a navigational system is to determine the systems degrees of freedom with a sufficient accuracy in order to carry out a mission. Underwater navigation introduces additional challenges, mostly because measurements from the global positioning system (GPS) are unavailable. Furthermore, the presence of sea ice, ice ridges and the possibility of icebergs makes the navigation in the polar regions even more challenging [33].

2.1.1 General approaches to underwater navigation

Given the lack of GPS signals and the limitations of acoustic technology, it is harder to perform accurate navigation underwater than for surface or aerial based vehicles.

Three typical approaches of underwater navigation are:

- Dead reckoning
- Acoustic positioning

- Geophysical navigation

Dead reckoning

This approach refers to navigation without communication with any reference point, topside or satellite. Thus, the system must navigate solely based on its internal sensors to consecutively update its states. With the lack of a bounded error in state estimation, the error will grow unbounded in time implying that the system can only navigate successfully for a limited time in order to fulfil the mission requirements.

Typical sensors used for underwater dead reckoning are gyroscopes, accelerometers, magnetic compasses and pressure sensors. The combination of these sensors with associated software to estimate the position in dead reckoning is often referred to as the inertial navigation system (INS).

Acoustic positioning

An underwater navigation system may send an acoustic ping to another device and use the time delay and phase shift to estimate both distance and angle between the system and the device. Such a device is often referred to as an acoustic responder, and may be attached to a topside such as DP vessel, another underwater vehicle or simply a fixed submerged installation (e.g. on the seabed or the underside of sea ice). As long as the transponder has a known location relative to earth, the position of the navigation system can also be determined. A common acoustic positioning system for underwater vehicles is USBL.

Furthermore, it is possible to send small data packages with acoustic waves of different frequency between the system and the transponder. This way the transponder may transmit its own position relative to Earth so that the navigation system can get frequent position updates thus bounding the position error. The range of such acoustic signals varies depending on the water conditions and the properties of the signals themselves, but for most applications they can be several km long.

Geophysical navigation

The principle of this approach is to use reference points in the terrain to navigate and compensate for dead reckoning drift and decrease the dependence on periodic GPS measurements [32]. The geophysical reference points may be localized using either optic or acoustic techniques. Examples on how to obtain such reference points are to image the sea bed, use sonar scan or hyperspectral sensors.

A continuation of this method is called simultaneous localization and mapping (SLAM) which allows the system to operate even more independently as it maps the surrounding environment and keeps track of its relative position to reference points on this map simultaneously.

2.1.2 Acoustics

Acoustic waves are vibrations propagating through a medium. The principle, commonly referred to as sound when waves propagate in air, also applies to water. When using acoustics for underwater navigation it is important to consider the imperfections of ocean water. It is in general inhomogeneous, especially over large distances. Salt is present, and its concentration may vary greatly. In addition the water column often consists of different temperature layers, and other particles than salt might disturb the path of acoustic waves. Furthermore, the water will absorb some of the wave energy by means of heat and suspended solid particles decreasing the spatial range of any acoustic system.

Speed of sound

Speed of sound of the acoustic medium is an important parameter for any acoustic system. In sea water it will vary with the density (ρ) and elastic modulus (E) of the respective water mass as shown in Equation 2.1.

$$c = \sqrt{\frac{E}{\rho}} \tag{2.1}$$

In seawater this speed might be approximated based on temperature (T), salinity level (S) and depth (z) using the empirical model of [26] as shown in Equation (2.2) which is valid down to z=1000 [m].

$$c_{sw} = 1449.2 + 4.6T + 0.055T^2 + 0.00029T^3 + (1.34 - 0.01T)(S - 35) + 0.016z \ [\frac{m}{s}] \tag{2.2}$$

Here, the temperature is given in degrees Celsius, the salinity level in parts per thousand (ppt) and the depth z given in meters. Typical values representable for most ocean states are between 1450 and 1500 m/s.

By knowing the velocity of the acoustic waves through the medium it is possible to determine the distance from an object echoing back the signal given that the duration of the signal is measured.

Multiple paths

Despite having a good indication of the wave speed trough the medium, multiple paths are a common challenge for underwater acoustic systems. It occurs when acoustic waves bounce off of multiple objects and propagate in several direction so that the system receives several echoes at different times. An illustration of the phenomenom is provided in Figure 2.1.

By considering only the shortest distance (direct path) of sufficient amplitude between the transmitter and the object of interest (OOI) this effect is normally avoided.

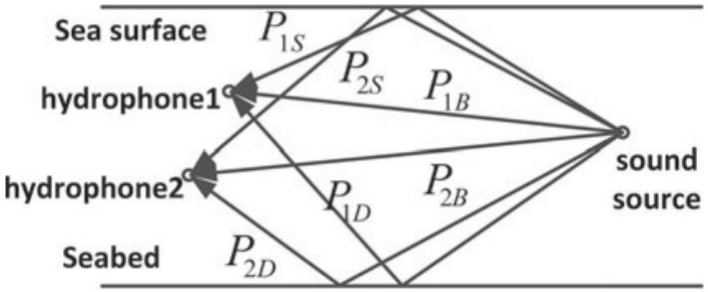


Figure 2.1: Illustration showing how a ping signal may find several paths towards the target. Only the shortest (P_{1B} and P_{2B}) is of interest when calculating the distance [42].

Doppler effect

The OOI may have a velocity relative to the signal origin, resulting in an apparent shift in the wave frequency referred to as the Doppler effect. While complicating the signal processing, it can actually be of great use as the phase shift correlates to the relative velocity of the object and transmitter.

A typical sensor for measuring this effect is a Doppler velocity log (DVL). Even the velocity profile of an entire water column may be found using this principle.

2.1.3 Control methods to account for ice induced risk

In polar regions sea ice offers additional challenges that must be addressed to minimize operational risk. Unfortunately, there is always a risk of losing the vehicle whenever deploying an UUV under ice, or damaging the vehicle due to a collision with the ice. The navigational sensors should try to map the surrounding geophysics of the ice in as much detail as possible.

The collision risk can always be lowered simply by increasing the altitude between the ice and the AUV. However, in most cases this will compromise on the data quality obtained by the payload sensors. In other words, an appropriate algorithm needs input as to what is high quality data, and use this to carefully balance risk and reward during the mission.

Due to the scope of this thesis, this section will focus on methods that may help balancing the risk of collision with ice against the reward of going as close as possible to get high quality data.

Autonomous alteration of control parameters

One method to minimize risk under sea ice is to implement an algorithm that autonomously alters the control parameters based on the state of the vehicle and the surrounding ice.

If the measurement conditions are especially favorable, it might be beneficial to move even

closer to the ice thus increasing the risk. Normally the control system of the UUV will use smooth reference signals, often subject to a low pass filter or similar. Additionally, the guidance system (e.g. LOS, see Section 2.6.5) normally makes the UUV approach softly toward the path of interest to minimize actuator force and wear and tear. Such measures are mainly added to minimize energy consumption and wear and tear of the AUV's actuators.

However, if the risk for collision is sufficiently high such parameters might be switched over in a more aggressive/direct mode to give more precise maneuvering capabilities. This will increase energy consumption but might give the AUV similar data quality at a lower risk.

Robust control

The aim of using robust control is to ensure stability of a system given certain bounded modelling errors. Hence, this will provide increased safety during under ice operations because the actual AUV will behave much the same as in simulations despite small deviations in the modelling.

H-infinity and Sliding mode control are both famous robust control techniques.

Model predictive control

The motivation for using MPC is to find an optimal control allocation to minimize energy consumption and state error. In stead of a typical PID controller which may give a lot of wear and tear on the actuators, and overshoot the reference signal, this method will ensure optimal control action based on the predefined cost function (objective function) $\Phi(\mathbf{w})$. Here $\mathbf{w}$, the decision vector, is a series of control actions for the total time horizon defined by the user. The solution to this control problem may also be subject to constraints such as saturated actuator force and physical obstacles (i.e. state constraints) such as the ice. This section is based on the book *Model Predictive Control* by Eduardo F. Chamacho [4].

To understand how an MPC works it is essential to know the basic principles of a nonlinear programming problem (NLP). The aim is to minimize a predefined cost function $\Phi(\mathbf{w})$ (objective function) by choosing an optimal decision variable $\mathbf{w}$. Furthermore the objective function may be subject to constraints which are categorize as either equality or inequality constraints. The mathematical formulation of the NLP problem can be written as:

$$\begin{aligned} &\min_{\pi} \Phi(\mathbf{w}) \\ \mathbf{s.t.} &\mathbf{g}_1(\mathbf{w}) \leq 0 \\ &\mathbf{g}_2(\mathbf{w}) = 0 \end{aligned} \tag{2.3}$$

Here $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$ are inequality and equality constraints respectively. In the case of an MPC they may provide physical limitations to where the AUV may go, saturate the control force or add obstacles to the environment. Note that if the constraints $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$ are affine, i.e.

linear combinations of $\mathbf{w}$, the NLP is reduced to a linear programming problem (LP). By applying a nonlinear solver, the solution of the NLP problem will be a series of decision variables $\mathbf{w}^*$ such that:

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \Phi(\mathbf{w}) \tag{2.4}$$

Now consider a discrete nonlinear multiple input multiple output (MIMO) system:

$$\boldsymbol{x}(k+1) = \boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k)) \tag{2.5}$$

$\boldsymbol{x}$ denotes the state vector, $\boldsymbol{u}$ the input vector and k the current timestep. Based on this model, the next state may be estimated given the input at the current time step. The cost function of an MPC uses this by summing up an error between the desired and current values both for the states and the inputs. Furthermore, the different states and inputs can be weighed differently, allowing the user to define what is important. Hence, the cost function J is comparable to that of and LQR controller, and will be a sum over the time horizon:

$$J_N(\mathbf{x}, \mathbf{u}) = \sum_{k=0}^{N} \ell\left(\mathbf{x}_{\mathbf{u}}(k), \mathbf{u}(k)\right) \tag{2.6}$$

For simplicity, the contribution to the cost function for a single timestep has been denoted ℓ which is a function the state and input for that timestep. It is normally formulated similar to an LQR where $\mathbf{x}^r$ and $\mathbf{u}^r$ are the reference state and reference input vector (normally zero):

$$\ell(\mathbf{x}, \mathbf{u}) = \|\mathbf{x}_{\mathrm{a}} - \mathbf{x}^r\|^2_{\mathrm{Q}} + \|\mathbf{u} - \mathbf{u}^r\|^2_{\mathrm{R}} \tag{2.7}$$

In the above equations J is the cost function, N is the number of time steps used in the MPC, $\boldsymbol{x}$ and $\boldsymbol{u}$ denotes the states and the input vector and Q and R are the weighting matrices for the states and inputs respectively.

The overall goal is to choose an optimal sequence of inputs that minimizes this cost function. The user defines a time step and the number of time steps to optimize over. The algorithm will for all these time steps calculate an optimal control action as shown in Figure 4.25.

With this framework, the MPC is now an an optimal control problem (OCP) and can be

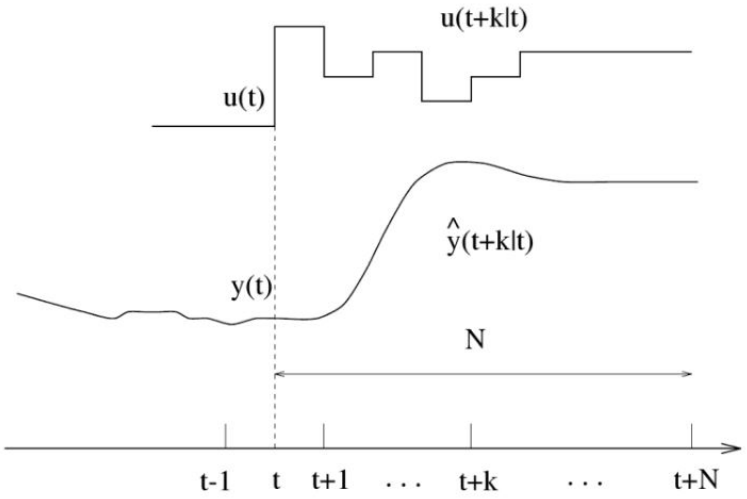


Figure 2.2: Graph illustrating the basic principle of MPC. For a given time horizon, the algorithm will optimize a series of control steps. Gathered from [4].

formulated as an NLP in the following manner:

$$\begin{aligned} \underset{\mathbf{u}}{\text{minimize}} \quad & J_N\left(\mathbf{x}_0, \mathbf{u}\right) = \sum_{k=0}^{N-1} \ell\left(\mathbf{x}_\mathbf{u}(k), \mathbf{u}(k)\right) \\ \text{subject to} \ : \quad & \mathbf{x}_\mathbf{u}(k+1) = \mathbf{f}\left(\mathbf{x}_\mathbf{u}(k), \mathbf{u}(k)\right) \\ & \mathbf{x}_\mathbf{u}(0) = \mathbf{x}_0 \\ & \mathbf{u}(k) \in U, \quad \forall k \in [0, N-1] \\ & \mathbf{x}_\mathbf{u}(k) \in X, \forall k \in [0, N] \end{aligned} \tag{2.8}$$

If this problem has a feasible solution any numerical NLP solver will be able to return the desired sequence of inputs for the control system.

The three basic iterative steps of an MPC algorithm are:

- Prediction.
- Online optimization.
- Receding horizon implementation.

In other words, after optimising the series of control actions for the total time horizon, only the first control action is executed. Iteratively it repeats the whole optimization real time at the next time step ensuring that even though the system did not behave as expected, the current control action will always be optimal from the current state. This is what differentiates MPC from other optimal control methods such as linear quadratic regulator (LQR).

The user can define as many time steps as a desired time horizon for the algorithm to optimize over. Naturally, by selecting a smaller time step and increasing the overall time horizon it will be more computationally heavy, but it will also give a better result given

the cost function. It is also an advantage that most optimizers will iterate over and over again either until a threshold is reached for the solution or the time has run out. In other words, the algorithm will always have some solution in memory ready for use. It might not be the optimal solution, but at least it will know what to do when the next time step demands a control action.

It is not always the case that you can know far into the future what the desired reference signal should be. For example, using an UHI sensor as an altitude estimator, the angle of which the UHI is rigged to the sensor platform will decide how far in the horizontal plane the AUV may sense the ice and therefore know what altitude will be desired. Hence, it is still important to have an algorithm to handle this and choose the best possible desired altitude based on the ice data from the navigational sensors.

Some disadvantages using MPC is that it is potentially computationally heavy and needs to be done in real time to ensure updated results. Even with sufficient computational power, the optimizer tries to solve a potentially complex nonlinear problem and might not converge at all in certain situations. Hence it is always a good idea to add redundancy by adding a backup controller such as a PID.

Markov Decision process

MDP is a great mathematical framework to use in situations where the outcome of a certain action is partly under the control of the decision maker and partly random. The probability of reaching a new state s' given an action a only depends on the current state s. The following tuple may be used to mathematically describe the MDP framework $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R})$ where,

- $\mathcal{S}$ is a set of states $\{s_1, \ldots, s_n\}$
- $\mathcal{A}$ is a set of actions $\{a_1, \ldots, a_n\}$
- $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is a transition function that represents conditional transition probabilities between states s and s' when executing action $a-$ that is, $\mathcal{T}(s, a, s') = P(s'|a, s)$
- $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is a reward function.

The solution to this problem is $\pi(s)$ which means the desired action given the current state s. Because the state space is inherently stochastic, the optimal policy $\pi*(s)$ is the one that maximises the expected utility. By propagating this forward in time we get the total discounted utility to become:

$$U^{\pi}(s) = E\left[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(S_t)\right] \tag{2.9}$$

Here s_t marks the random state the agent reaches for each time step applying the policy π. In other words the optimal policy may be mathematically expressed as:

$$\pi^*(s) = \text{argmax}_{a \in \mathcal{A}(s)} \sum_s P\left(s'|a, s\right) U\left(s'\right) \tag{2.10}$$

The value iteration algorithm and policy iteration algorithm may be used for iteratively calculating an optimal policy [3].

Fuzzy logic

Fuzzy logic is logic based on a deciding variable between two numbers such as 0 and 1. By contrast, boolean logic is based on values that are either true or false.

Fuzzy logic has been used on altitude control of an airship [20], and this application may be very similar to that of an underwater vehicle's altitude under ice.

A fuzzy-based approach was also used on an AUV to minimize the chance of collision and vehicle loss during under ice operations in Antarctica [37]. The principle used here was to assess the risk using fuzzy logic, and thus give the different operations a risk rating. Afterwards this risk rating would be used for decision making during the operation to make it as safe as possible.

Temporal constraints and emergency strategies under ice

In addition of loosing or damaging the vehicle due to a collision, other challenges must be faced when operating under ice. In general it is harder to call an abort, because the vehicle cannot simply float to the surface, beep a GPS location and get picked up by the vessel or topside. Furthermore, even if the AUV decides to abort and return to the starting point, it must be able to navigate back precisely as the topside might not be able to move a lot due to the surrounding ice conditions. In some cases, the deployment phase may be conducted from the surface of the ice only facilitating the deployment by a hole in the ice. In such a case, an AUV must be able to call an abort and independently return back to this hole in order to avoid loss of equipment.

Stricter temporal constrains might also be imposed by the challenging conditions at the poles. Normal lithium ion batteries gets drained in cold temperatures and there should be given a larger safety margin with respect to return back to the home point. Additionally, the topside might have challenging conditions at the surface, and the AUV should always be within a certain temporal threshold were the topside know they may hold the location depending on the current and developing weather situation.

Should anything happen during the mission, both the topside and the AUV should be able to call an abort. The AUV may have different conditions for choosing abort such as low battery level, collision, temporal threshold, or other unexpected situations. Furthermore,

it is important that the AUV has specified algorithms to deal with a mission abort in the best way possible. It might depend on the situation, and there are a lot of different ways to solve this. If an AUV is following a specific pattern (e.g. a lawn mower maneuver) to collect data, it must choose either to follow the same tracks back to the starting point, or perhaps choose the shortest route possible. The best option depends on a lot of factors, and should be chosen to minimize risk.

2.2 Heading estimation

Kenneth Gade at Norwegian Defence Research Establishment has worked with navigation for several years. In 2016 he published [14] which has been a basis for the methods shown in this project thesis.

The main purpose of a navigation system is to estimate the six degrees of freedom of the vehicle. Roll and pitch may easily be estimated by knowing the gravitational vector relative to the body using an accelerometer. Depth is often estimated through a pressure sensor. The horizontal position may be more challenging to find, but is often estimated through global navigation satellite systems (GNSS), an acoustic topside with known global position, dead reckoning or a combination of these. The heading is often found with a gyroscope that may or may not be accurate enough for gyrocompasing. An overview broken down on these parameters from [14] is shown in Figure 2.3.

Green/italic = Often satisfactory Red/underlined = Challenging		GNSS (or similar) available	
		Yes	**No**
Gyros with sufficient accuracy for gyro-compassing	**Yes**	Category A1: *Heading* *Horizontal position* Typical cases: Large/expensive vehicles (not submerged), e.g. airplanes, ships, helicopters	Category A2: *Heading* Horizontal position Typical cases: Underwater navigation of large/expensive vehicles, e.g. submarines, autonomous underwater vehicles (AUVs)
	No	Category B1: Heading *Horizontal position* Typical cases: Light/small/cheap applications in air, land or at sea, e.g. unmanned aerial vehicles (UAVs), boats, robots, cameras, personnel	Category B2: Heading Horizontal position Typical cases: GNSS denied light/small/cheap applications, e.g. indoor navigation, applications under GNSS jamming, low-cost underwater navigation

Figure 2.3: Illustrating the different categories of navigation systems segmented on gyro accuracy and GNSS availability.

Kenneth Gade from Norwegian Defence Research Establishment (FFI) has recognized 7 practical methods for estimating heading of a vehicle. Each method corresponds to a vector that has a known orientation relative to the ellipsoid of earth. This vector must have a known orientation relative to Earth $\boldsymbol{x}^E$, and with respect to the body of the vehicle $\boldsymbol{x}^B$.

$$\boldsymbol{x}^E = \boldsymbol{R}_{\boldsymbol{EB}}\boldsymbol{x}^B \tag{2.11}$$

Some methods will be more suited for certain applications. However, the objective of this section is to undertake all of them and discuss how they may be relevant for a marine underwater operation in the polar regions.

Method 1: The magnetic vector field of the earth

The first method has been used by humans for centuries, simply by locating the vector pointing towards north and using that as a reference relative to the vehicle to determine heading. There are electric currents of molding iron towards the Earths kernel, thus creating a massive magnetic field that stretches far into space. Because of varying state of iron currents and general inhomogeneities within Earth, the magnetic field is neither constant in time or symmetric around the rotational axis of earth. Thus, the poles of the magnetic field will not perfectly correspond to the geographical poles of earth. This will cause a constant angle of error between the true north and the magnetic north depending on where and when you are measuring with a magnetic compass, called the *declination* angle. Furthermore, the magnetic field is relatively weak compared to other magnetic disturbances that can be caused by human made objects in urban areas, natural occurring magnetic materials in the ground, changes in solar winds, geomagnetic storms and ferrous metals or electronically induced distortions from the vehicle itself.

Despite these challenges, magnetic heading estimation may be attractive for AUVs as it is affordable, self-contained and can be used either as a stand-alone solution or in a sensor fusion with for example a gyroscope. For under ice operations the most relevant disturbances are declination and inclination angles that are high in magnitude near the poles. The declination in the relevant area may be modeled as a constant bias and compensated for by use of for example [9].

As shown by [17] it is possible to bound the magnetic compass error if the AUV is given frequent chances to get a Differential Global Positioning System (DGPS) reading near the surface. However, one of the main challenges of conducting AUV operations in polar regions is the presence of sea ice and the possibility of ice bergs or other obstacles. Even though a DGPS fix is hard to accommodate, an interesting approach could be to attach GPS-beacons in the ice that may communicate with the AUV using an acoustic transponder.

Method 2 The angular velocity of the Earth

This method exploits that the earth angular rotation with respect to inertial space ($\overrightarrow{\omega}_{IE}$) by definition points in the same direction as the geographic north pole (see Figure 2.4).

Long term bias drift along with temperature dependant bias change are among common MEMS gyrocompassing challenges [21]. Such challenges require a high focus on error elimination. Taking the Allen variance and applying an extended Kalman filter (EKF) is effective to compensate for the slowly varying bias of the gyroscope [19]. A carouselling scheme suggested by [31] may reduce the bias dramatically, and make even MEMS gyroscopes accurate enough for gyrocompassing applications.

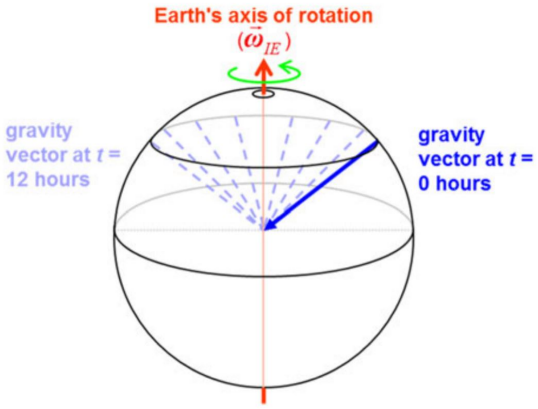


Figure 2.4: The gravitational vector rotates relatively to the inertial space (figure assumes low/zero velocity relative to Earth). Courtesy of Gade [13].

An AUV may manoeuvre in this specific patters to obtain higher gyroscope accuracy. However, a gyroscope will always have an unbounded error as it will navigate on a dead reckoning basis. Furthermore, the angular velocity of the earth has less impact on the sensor at higher latitudes. Thus, the uncertainty using this method will be proportional to $1/cos(latitude)$ [14] which is especially unsuited for polar applications.

Method 3 External objects

If the global position of two objects (O_1 and O_2) is known, the vector between them ($\overrightarrow{P}_{O_1O_2}$) is known in the frame with respect to Earth. If the orientation of the vehicle is known with respect to $\overrightarrow{P}_{O_1O_2}$, the heading may be calculated.

Cameras are often used for practical implementations of this method, using fixed landmarks such as human made distinctive buildings or objects. For underwater application cameras have poor range, so other sensors such as sonars may be used instead. An example is to fix long-baseline acoustic transponders to certain reference points on the bottom of the seabed to guarantee state feedback in a limited area complementary to dead reckoning based on the inertial measurement unit (IMU).

Because most level and pack ice (no ridges) in the Arctic Ocean rarely gets thicker than 7 meters even during winter season [2], it might be possible to attache acoustic transponders within the ice. This would be of special benefit for research operations close to the ice but with great water depth (3500 meter at the North Pole) not allowing sea bed mounted transponders.

Method 4 Objective relative to vehicle

Similarly to method 3, a vector from the vehicle itself to a known objective may be defined, $\overrightarrow{P}_{BO}$. However, for this method to work, the position of the vehicle must be known with a high degree of accuracy, and relies heavily on the availability of frequent GPS readings

as shown in Figure 2.3 [14]. Thus, it might not be the optimal solution for under ice applications.

Method 5 Body fixed vector

Another solution is to define a vector between two objects on the vehicle itself with known position relative to the body frame. To use this vector for heading calculation one must know the position of these objects relative to the earth frame as well, and the horizontal component of the vector must be sufficiently large. For an AUV it would be optimal to mount one such device in the tail and one in the head of the vehicle. These devices could for instance be acoustic transponders, which would require at least three fixed probes close by to triangulate an accurate position.

Method 6 Vehicle velocity vector

If the vehicle has a velocity vector with a horizontal component, the heading can be determined based on this vector. An attractive way to estimate the velocity vector for an AUV is by the use of a Doppler velocity log based on the principles explained in Section 2.1.2. If the current speed is known, the velocity relative to earth may also be calculated measuring the velocity relative to the surrounding water masses. This relative velocity might for example be calculated using a hydrodynamic model of the AUV [14]. For vehicles that are not equipped with gyroscopes with high enough accuracy to sufficiently estimate heading, method 6 is used [15].

A clear disadvantage of this method is its inherent inaccuracy during low speeds or high pitch angle resulting in a small horizontal velocity component.

Method 7 Vehicle acceleration vector

Method 6 requires a direct measurement of velocity which is not always available. However, tiny and energy inexpensive MEMS accelerometers are integrated in most INS also for more affordable vehicles [14]. After subtracting gravitational acceleration and Coriolis effect the signal is ready to be processed.

The clear disadvantage which is especially relevant for AUVs is that this method requires sufficient horizontal accelerations to estimate heading. An AUV may operate for long sections at a time at a constant velocity.

2.3 Underwater Hyperspectral Imaging

Patented in 2015, an UHI is a relatively new sensor combining a push-broom hyperspectral imager with an external light source [38]. These components as a whole may be used

on several platforms (e.g. an UUV) and connect with electrical cables and optical fibers. Halogens lights are often used as the external light source as they have close to a uniformally distributed energy spectrum across the frequencies of visible light. The imaging sensor itself can record measurements of reflected light from 380 to 800 nm and thus creating a energy spectrum of the environment. This in turn might be used to create an unique spectrum library for different OOIs. It is shown that the use of UHI in stead of traditional imaging techniques using the 3-color RGB-based pictures and video, brings the technology more suitable for automatic identification of OOI [40].

The technology has the potential of analysing the underside of sea ice and qualitatively measure light, water and ice conditions as well as the organic life.

Currently, Econtone AS in Norway has the patent for production of UHI sensors, and an illustration of their product mounted on an underwater vehicle may be seen in Figure 2.5 [38].

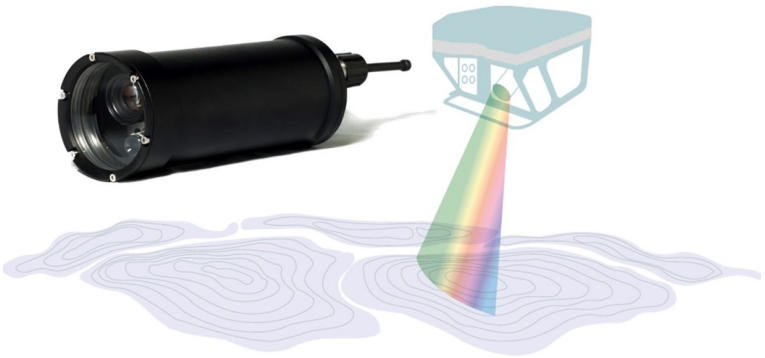


Figure 2.5: Illustration showing the UHI hardware to the left and how it may scan the sea bed mounted to an UUV. Courtesy of Ecotone AS.

2.3.1 Applications

Hyperspectral Imaging sensors may be deployed on numerous platforms such as satellites, suborbital aircrafts, boats or on underwater platforms to map anything from sea bed to primary production levels in the oceans to biomass on the African savanna. This section will focus on UHI sensors and their applications for underwater mapping of seabed and sea ice. The section is mainly based on the book "Underwater hyperspectral imagery to create biogeochemical maps of seafloor properties" by NTNU professor Geir Johnsen [40].

A general challenge in exploring and mapping the seafloor is its complexity and dynamic behavior, exhibiting large variability in biogeochemical composition over space and time. The traditional methods often have spatial limitations and are qualitative meaning that extensive human interpretation is required to obtain results. Such methods include *in situ* diver surveys, ship-based acoustics (echosounders), benthic box core (i.e. 'grab' samples), epibenthic sledge and beam trawl samples, underwater photography, and video towed from boat to mention some. In general, traditional methods rely on sunlight which does not always have enough intensity to sample the seafloor which is typically optically deep.

One of the distinct advantages of UHI is its use of an external light source providing

enough radians to rely on the spectral signature of the unique features of the benthos. Thus, different OOIs from minerals to plankton absorb and reflect different intensities for different frequencies in the visible spectrum giving them a unique spectrum i.e. 'optical fingerprint'. By mounting the UHI sensor on a moving platform such as an ROV or AUV these optical fingerprints can be used to qualitatively and quantitatively develop maps of minerals, benthic habitats, substrates, and organisms.

In general, the dominating factor limiting the use of an UHI sensor is the visibility in the water. It may vary a lot from place to place from 40 m in the Bahamas to merely 0.5 m in the fjord of Trondheim. Plankton, which often is of interest to measure can to a large extend limit the use of an UHI. The radiant, i.e. the flux of photons through the hyperspectral sensor, depends on both the light emitted from the sun and the external light source and is important for high quality measurements. It is also possible to turn off the external light while under ice in order to measure the natural light conditions which lays the foundation for primary production in that habitat. This has already been done by several scientist close to the ice edge where the nutrients trapped in the ice give rise to plankton and ice algaes. This process is highly season dependent. UHI sensors have already been used in successfully mapping biomass of algaes on sea ice [6].

2.3.2 Technological principles

An UHI sensor weighs about 5 kg and may dive to depths of 6000 meters with a titanium housing. It consists of both a regular RBG camera and the main unit which is a hyperspectral imager. Additionally, most UHIs also have an associated altimeter to calculate the distance from the measured objective.

The hyperspectral imager measures light of different frequencies, also outside the frequency domain of visible light to humans. How much light the UHI receives is measured in radiance which is a measurement unit for electromagnetic energy denoted $L\left[Wm^{-2}sr^{-1}nm^{-1}\right]$. Thus, the radiance is the most important parameter to evaluate the utility of the data. It will depend on many factors such as the visibility of the water, the intensity of sunlight and the external light source and the material and distance from the OOI.

The radiance is affected by two major principles when light is sent out from the external light source:

- Attenuation.
- Backscattering.

Attenuation is the principle that the light is being sent out from the light source so that the intensity will decrease as the distance from the light source increases. This relation is exponential, and under the assumption of only one external light source present it will be given by:

$$L(z,\hat{\xi}) = L(0;\hat{\xi})e^{-c(\lambda)z} \tag{2.12}$$

Here z and $\hat{\xi}$ denote the distance and the direction from the light source respectively, and

$c(\lambda)$ is an attenuation coefficient that depends on the wave length of the emitted signal.

Backscattering is an additive effect meaning that it will add unwanted radiance as the omitted light reflect off of particles in the water and environment around the sensor. Due to this effect, the quality of measurements will decay with distance and particles in the water [25]. It is not straight forward to filter this radiance out of the measured signal, but in clear waters and modest distances it is possible to get high quality results despite backscattering.

The main principle of the UHI sensor is that it sends out light in a push-broom i.e. fan-like pattern and measures the received light spectrum from the environment. A simplified model often assumes a flat surface of the sea bed or ice. Thus, the spatial range will be proportional to the altitude thus allowing the mission to map a lot more of the surface by increasing the altitude. However, this will strongly compromise on the utility of the data which can be assumed proportional to the radiance. A sketch of the simplified UHI model may be seen in Figure 2.6.

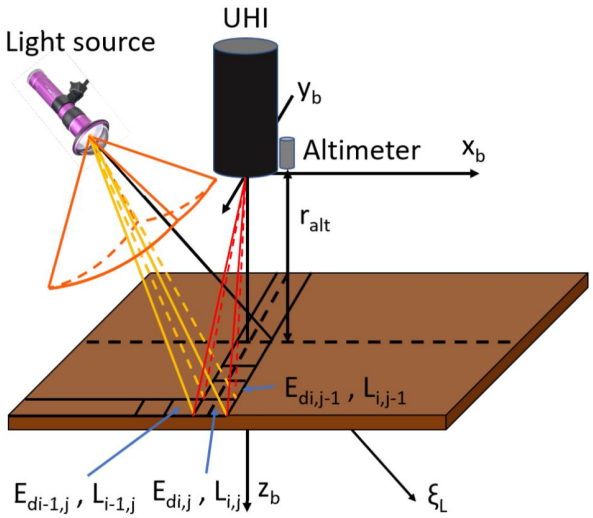


Figure 2.6: Illustration showing the simplified UHI model assuming a flat surface. Courtesy of PhD candidate Håvard Løvås.

2.3.3 UHI as a navigational sensor

A relatively new field which is one of the main contributions of this master thesis is to look into the possibility of using an UHI sensor for navigation. Much of the technology both in terms of sensors, but also software such as machine learning (ML) and artificial intelligence (AI) is to a large extent based on replicating and improving human behavior. If a submarine was diving under sea ice for mapping primary production levels there would probably be a person assigned to evaluate the quality of the measurements and use his visual to ask the captain to go either closer or further away from the ice based on the utility of the measurements. In other words, the main motivation for using UHI as a navigational sensor is when it is already a part of the platform as a payload sensor. Note that in the submarine example you might also have a person responsible for risk and another one

responsible for the overall mission objectives. They might disagree on the next move for the submarine, and this must be solved in the best way possible. Similarly for an AUV with an UHI sensor a complex algorithm must be made to manage risk in light of the utility of the UHI measurements.

2.4 Observer theory

This section will describe the principles of state estimation using an observer to improve the navigation sensor measurement before entering the control system. It is based on the work of Fossen [10]. There are several well tested state estimation techniques and observer methods in the literature. However, this section will focus on the following three theories:

- Luenberger observer.
- Kalman filter.
- Passive nonlinear filter.

Before describing in in more detail, it is relevant to describe the concept of observability. A system is observable if for any arbitrary sequence of actuator inputs and states, the current state can be determined in final time using only the outputs (sensor measurements). Consider a linear time-invariant (LTI) system represented by the following state space model:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \\ \boldsymbol{y} &= \boldsymbol{H}\boldsymbol{x} \end{aligned} \tag{2.13}$$

Here, $\boldsymbol{A}$ is called the state or system matrix, $\boldsymbol{B}$ is the input matrix and $\boldsymbol{H}$ is the output matrix. $\boldsymbol{x}$ denotes the state vector, and $\boldsymbol{y}$ is the output vector of the state space model. It follows from the definition above that for the system to be observable, $\boldsymbol{A}$ and $\boldsymbol{H}$ must satisfy that the observability matrix in Equation 2.14 is invertible [16].

$$\mathcal{O} = \left[\boldsymbol{H}^\top \left| \boldsymbol{A}^\top \boldsymbol{H}^\top \right| \cdots \left| \left(\boldsymbol{A}^\top \right)^{n-1} \boldsymbol{H}^\top \right. \right] \tag{2.14}$$

2.4.1 Luenberger observer

The Luenberger observer is a simple design that makes the state estimation error converge towards zero. Consider the observable LTI system:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{E}\boldsymbol{w} \\ \boldsymbol{y} &= \boldsymbol{H}\boldsymbol{x} + \boldsymbol{v} \end{aligned} \tag{2.15}$$

$\boldsymbol{w}$ and $\boldsymbol{v}$ represent the model and sensor error vectors respectively, here assumed to be zero-mean white noise. $\boldsymbol{E}$ is the error matrix. By assuming that $\boldsymbol{w}$ and $\boldsymbol{v}$ are zero, and that the system dynamics can be accurately copied by an observer, the following system is obtained:

$$\begin{aligned} \dot{\hat{\boldsymbol{x}}} &= \boldsymbol{A}\hat{\boldsymbol{x}} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{\gamma}(\boldsymbol{y}, \hat{\boldsymbol{y}}) \\ \hat{\boldsymbol{y}} &= \boldsymbol{H}\hat{\boldsymbol{x}} \end{aligned} \tag{2.16}$$

The hats symbolize estimated values. The term $\boldsymbol{\gamma(y, \hat{y})}$ is designed to make the error dynamics stable, i.e. $\hat{\boldsymbol{x}} \to \boldsymbol{x}$ as $t \to \infty$. The error dynamics are shown in the following equations:

$$\boldsymbol{\gamma}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \boldsymbol{K}\boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} = \boldsymbol{y} - \hat{\boldsymbol{y}} = \boldsymbol{H}\tilde{\boldsymbol{x}} \tag{2.17}$$

Here, $\tilde{\boldsymbol{x}} := \boldsymbol{x} - \hat{\boldsymbol{x}}$, and its dynamics are governed by:

$$\dot{\tilde{\boldsymbol{x}}} = \boldsymbol{A}\tilde{\boldsymbol{x}} - \gamma(\boldsymbol{y}, \hat{\boldsymbol{y}}) \tag{2.18}$$

By substituting from Equation 2.17 the final state error dynamics for the Luenberger observer is:

$$\dot{\tilde{\boldsymbol{x}}} = (\boldsymbol{A} - \boldsymbol{K}\boldsymbol{H})\tilde{\boldsymbol{x}} \tag{2.19}$$

Asymptotical convergence may be shown given the above assumptions for a suitable constant design matrix $\boldsymbol{K}$.

2.4.2 Kalman filter

The Kalman filter is a recursive minimum variance filter that estimates the system states from a series of noisy measurements. The algorithm may also be used for a nonlinear system by linearizing the system around certain operational states. This is referred to as an extended Kalman filter, and the linearization may be done prior to the mission or real time. Kalman filters can be applied both to continuous and discrete systems, the latter will be further addressed in this section. Consider the following discrete state space model:

$$\begin{aligned} \boldsymbol{x}(k+1) &= \boldsymbol{\Phi}\boldsymbol{x}(k) + \boldsymbol{\Delta}\boldsymbol{u}(k) + \boldsymbol{\Gamma}\boldsymbol{w}(k) \\ \boldsymbol{y}(k) &= \boldsymbol{H}\boldsymbol{x}(k) + \boldsymbol{v}(k) \end{aligned} \tag{2.20}$$

A complete overview of the discrete Kalman filter algorithm may be found in Table 2.1. The relevant parameters of the algorithm is explained here:

- $\bar{\boldsymbol{P}}$ represent the predicted state estimate covariance based on the previous state and input.

Design matrices	$\boldsymbol{Q}(k) = \boldsymbol{Q}^\top(k) > 0, \boldsymbol{R}(k) = \boldsymbol{R}^\top(k) > 0$
Initial conditions	$\overline{\boldsymbol{x}}(0) = \boldsymbol{x}_0$
	$\overline{\boldsymbol{P}}(0) = E\left[(\boldsymbol{x}(0) - \hat{\boldsymbol{x}}(0))(\boldsymbol{x}(0) - \hat{\boldsymbol{x}}(0))^\top\right] = \boldsymbol{P}_0$
Kalman gain matrix	$\boldsymbol{K}(k) = \overline{\boldsymbol{P}}(k)\boldsymbol{H}^\top(k)\left[\boldsymbol{H}(k)\overline{\boldsymbol{P}}(k)\boldsymbol{H}^\top(k) + \boldsymbol{R}(k)\right]^{-1}$
State estimate update	$\hat{\boldsymbol{x}}(k) = \overline{\boldsymbol{x}}(k) + \boldsymbol{K}(k)[\boldsymbol{y}(k) - \boldsymbol{H}(k)\overline{\boldsymbol{x}}(k)]$
Error covariance update	$\hat{\boldsymbol{P}}(k) = [\boldsymbol{I} - \boldsymbol{K}(k)\boldsymbol{H}(k)]\overline{\boldsymbol{P}}(k)[\boldsymbol{I} - \boldsymbol{K}(k)\boldsymbol{H}(k)]^\top + \boldsymbol{K}(k)\boldsymbol{R}(k)\boldsymbol{K}^\top(k), \quad \hat{\boldsymbol{P}}(k) = \hat{\boldsymbol{P}}(k)^\top > 0$
State estimate propagation	$\overline{\boldsymbol{x}}(k+1) = \boldsymbol{\Phi}(k)\hat{\boldsymbol{x}}(k) + \boldsymbol{\Delta}(k)\boldsymbol{u}(k)$
Error covariance propagation	$\overline{\boldsymbol{P}}(k+1) = \boldsymbol{\Phi}(k)\hat{\boldsymbol{P}}(k)\boldsymbol{\Phi}^\top(k) + \boldsymbol{\Gamma}(k)\boldsymbol{Q}(k)\boldsymbol{\Gamma}^\top(k)$

Table 2.1: Overview of the discrete Kalman filter algorithm from [10].

- $\hat{\boldsymbol{P}}$ represent the a posteriori state estimate covariance.
- $\bar{\boldsymbol{x}}$ represent the predicted state estimate based on the previous state and input.
- $\hat{\boldsymbol{x}}$ represent the a posteriori state estimate
- $\boldsymbol{Q}$ is the process covariance matrix, i.e how much do you trust your model?
- $\boldsymbol{R}$ is the measurement covariance matrix, i.e how much do you trust your sensors?
- $\boldsymbol{K}$ is the Kalman gain.

2.4.3 Passive nonlinear filter

One drawback using a Kalman filter for state estimation is the needed tuning process that may be both challenging and time consuming. Even back-stepping methods proposed in [12] need some information about the system which can be time consuming and expensive to gather. A passive nonlinear filter drastically eases this tuning process. A typical passive nonlinear filter will make the state estimation error globally exponentially stable which can be proven by Lyapunov analysis, which is not the case for extended Kalman filters. For further details see [11].

2.5 Mathematical models of AUVs

This section briefly describes the mathematical model of an AUV based on [10].

2.5.1 Kinematics

Kinematics explains geometrical aspects of the system. An AUV has six degrees of freedom usually referred to as *surge, sway, heave, roll, pitch and yaw*, denoted:

$$\boldsymbol{\eta} = [N \ E \ D \ \phi \ \theta \ \psi]^T \tag{2.21}$$

Reference frames

To analyse and describe the motions of an AUV it is practical to define several reference frames, both with respect to the earth but also with respect to geographical locations, sensors and the body itself. Figure 2.7 shows an AUV with different reference frames.

- *The Earth fixed frame* {e} origin corresponds to the earth center, and rotates at the earth angular rate $\omega_e = 7.2921\ x\ 10^{-5}$ rad/s. However, for relatively slow moving vehicles such as an AUV, this frame can be assumed to be inertial thus neglecting the angular rate.
- *North-East-Down* {n} (NED) is defined with an origin relative to the Earth's reference ellipsoid with one axis pointing towards true north, one towards true east and the last on towards the earth center. Most AUVs have such limited spatial range that flat-earth navigation using NED-frame is sufficient.
- *BODY-frame* {b} is fixed to the body and its attitude is described relative to the inertial frame fixed to the body.
- *Sensor-frame* {s} might be useful for AUVs both for navigational and payload purposes. Normally a sensor is fixed to the body, but its frame may have an attitude and origin that differs from the body frame.

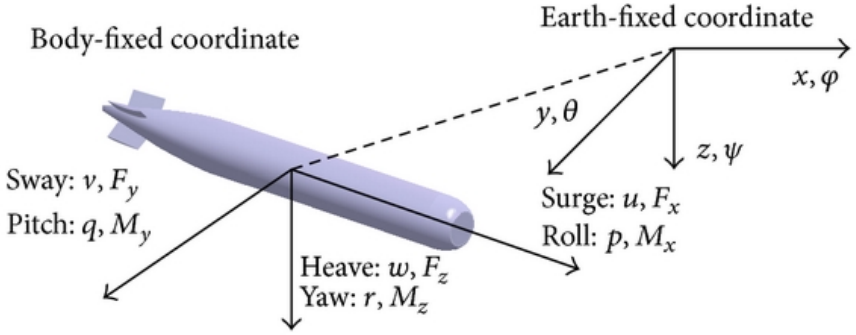


Figure 2.7: Illustration showing the body frame and earth-fixed frame relative to the AUV in addition to the names of the different DOFs [30].

Transformations matrices

To express coordinates and vectors in another reference frame a transformation matrix may be used. $\mathbf{R}_b^a$ is defined as the matrix that relates reference frame $\boldsymbol{a}$ to $\boldsymbol{b}$ in the following manner:

$$\boldsymbol{\nu}^a = \mathbf{R}_b^a \boldsymbol{\nu}^b \qquad \forall\ \boldsymbol{\nu} \in \mathbb{R}^3 \ \mid\ \boldsymbol{\nu} \neq \overrightarrow{0} \tag{2.22}$$

$\boldsymbol{\nu} = [u\ v\ w\ p\ q\ r]^T$ represent the time derivative of position and attitude of the AUV. Such a matrix from a uni-axial rotation as expressed in [10] are shown below.

$$\boldsymbol{R}_{x,\phi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \mathrm{c}\phi & -\mathrm{s}\phi \\ 0 & \mathrm{s}\phi & \mathrm{c}\phi \end{bmatrix}, \quad \boldsymbol{R}_{y,\theta} = \begin{bmatrix} \mathrm{c}\theta & 0 & \mathrm{s}\theta \\ 0 & 1 & 0 \\ -\mathrm{s}\theta & 0 & \mathrm{c}\theta \end{bmatrix}, \quad \boldsymbol{R}_{z,\psi} = \begin{bmatrix} \mathrm{c}\psi & -\mathrm{s}\psi & 0 \\ \mathrm{s}\psi & \mathrm{c}\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.23}$$

Here, $s\cdot = sin(\cdot)$ and $c\cdot = cos(\cdot)$. Furthermore, the rotational matrix $\boldsymbol{R}_b^n(\boldsymbol{\Theta}_{nb})$ inverse is its own transpose such that:

$$\boldsymbol{R}^{-1} = \boldsymbol{R}^\top \quad \Longrightarrow \quad \boldsymbol{R}\boldsymbol{R}^\top = \boldsymbol{R}^\top\boldsymbol{R} = \boldsymbol{I} \tag{2.24}$$

Linear velocity transformation
A complete rotational matrix as a function of the relative attitude (roll, pitch and yaw angles) between two reference frames are obtained by multiplying the matrices from (2.23) together. Let $\boldsymbol{\Theta}_{nb} = [\phi\ \theta\ \psi]^T$.

$$\boldsymbol{R}_b^n(\boldsymbol{\Theta}_{nb}) := \boldsymbol{R}_{z,\psi}\boldsymbol{R}_{y,\theta}\boldsymbol{R}_{x,\phi} \tag{2.25}$$

Equation 2.25 yields:

$$\boldsymbol{R}_b^n(\boldsymbol{\Theta}_{nb}) = \begin{bmatrix} \mathrm{c}\psi\mathrm{c}\theta & -\mathrm{s}\psi\mathrm{c}\phi + \mathrm{c}\psi\mathrm{s}\theta\mathrm{s}\phi & \mathrm{s}\psi\mathrm{s}\theta + \mathrm{c}\psi\mathrm{c}\phi\mathrm{s}\theta \\ \mathrm{s}\psi\mathrm{c}\theta & \mathrm{c}\psi\mathrm{c}\phi + \mathrm{s}\phi\mathrm{s}\theta\mathrm{s}\psi & -\mathrm{c}\psi\mathrm{s}\phi + \mathrm{s}\theta\mathrm{s}\psi\mathrm{c}\phi \\ -\mathrm{s}\theta & \mathrm{c}\theta\mathrm{s}\phi & \mathrm{c}\theta\mathrm{c}\phi \end{bmatrix} \tag{2.26}$$

So that the time derivative of the position expressed in {n} can be related to the linear velocity in {b} as follows:

$$\dot{\boldsymbol{p}}_{b/n}^n = \boldsymbol{R}_b^n(\boldsymbol{\Theta}_{nb})\,\boldsymbol{v}_{b/n}^b \tag{2.27}$$

Angular velocity transformation
A similar approach may be used to transform between the time derivative of the attitude and the body-fixed angular velocity vector which is defined as $\omega_{b/n}^b = [p, q, r]^\top$.

$$\dot{\boldsymbol{\Theta}}_{\boldsymbol{nb}} = \boldsymbol{T}_{\boldsymbol{\Theta}}(\boldsymbol{\Theta}_{\boldsymbol{nb}})\,\boldsymbol{\omega}_{\boldsymbol{b/n}}^{\boldsymbol{b}} \tag{2.28}$$

Where,

$$\boldsymbol{T}_\Theta(\boldsymbol{\Theta}_{nb}) = \begin{bmatrix} 1 & \mathrm{s}\phi\mathrm{t}\theta & \mathrm{c}\phi\mathrm{t}\theta \\ 0 & \mathrm{c}\phi & -\mathrm{s}\phi \\ 0 & \mathrm{s}\phi/\mathrm{c}\theta & \mathrm{c}\phi/\mathrm{c}\theta \end{bmatrix} \tag{2.29}$$

Notice that this rotational matrix has a mathematical singularity for pitch angles of $\theta = \pm 90^\circ$ which might be avoided in numerical calculations using quaternions. However, this is an unusual state for an AUV and will not be further discussed in this project. Finally, the full 6 DOF kinematic equations for an AUV may be expressed as:

$$\dot{\boldsymbol{\eta}} = \mathbf{J}_\Theta(\boldsymbol{\eta})\boldsymbol{\nu} \tag{2.30}$$

$$\begin{bmatrix} \dot{\boldsymbol{p}}^n_{b/n} \\ \dot{\boldsymbol{\Theta}}_{nb} \end{bmatrix} = \begin{bmatrix} \boldsymbol{R}^n_b\left(\boldsymbol{\Theta}_{nb}\right) & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \boldsymbol{T}_\Theta\left(\boldsymbol{\Theta}_{nb}\right) \end{bmatrix} \begin{bmatrix} \boldsymbol{v}^b_{b/n} \\ \boldsymbol{\omega}^b_{b/n} \end{bmatrix} \tag{2.31}$$

2.5.2 Kinetics

Kinetics is the analysis of the forces and moments causing motion to the system. General kinematic equations for a 6 DOF system may be expressed with Fossen's robot-like vectorial marine craft model [10]:

$$\begin{aligned} \dot{\boldsymbol{\eta}} &= \mathbf{J}_\Theta(\boldsymbol{\eta})\boldsymbol{\nu} \\ \mathbf{M}\dot{\boldsymbol{\nu}}_r + \mathbf{C}\left(\boldsymbol{\nu}_r\right)\boldsymbol{\nu}_r + \mathbf{D}\left(\boldsymbol{\nu}_r\right)\boldsymbol{\nu}_r + \mathbf{g}(\boldsymbol{\eta}) &= \boldsymbol{\tau} \end{aligned} \tag{2.32}$$

Here, $\mathbf{M} = \mathbf{M}_{RB} + \mathbf{M}_A$ represent the rigid-body inertial and added mass matrixes. Furthermore, $\mathbf{C}\left(\boldsymbol{\nu}_r\right) = \mathbf{C}_{RB} + \mathbf{C}_A$ is the Coriolis and centripetal matrix due to {b}'s rotation around the assumed inertial frame {n}. Damping due to hydrodynamic effects are included in $\mathbf{D}\left(\boldsymbol{\nu}_r\right)$ and finally, $\boldsymbol{g}(\boldsymbol{\eta})$ represent restoring forces and moments. A typical AUV has actuators such as propeller (p), rudder fins (r) and stern fins (s). These create forces and moment represented by $\boldsymbol{\tau}$.

Unless the AUV has an even number of propellers rotating in opposite direction, the propulsion system will create a torque in roll in addition to the desired surge force. This relationship has been derived by [5].

$$X_p = K_T \rho D^4_{prop} n|n| \tag{2.33}$$

$$K_p = K_Q \rho D^5_{prop} n|n| \tag{2.34}$$

K_T and K_Q are system specific thrust and torque coefficients respectively, n is the propellers angular speed and D_{prop} is the propeller diameter. The remaining forces and moments caused by ordered angles of the stern or rudder fins, δ_s and δ_r are derived by Prestero, [29].

$$Y_r = \frac{1}{2}\rho C_{L,\alpha} S_{fin}\left[u^2\delta_r - uv - x_{fin}(ur)\right] \tag{2.35}$$

$$Z_s = -\frac{1}{2}\rho C_{L,\alpha} S_{fin} \left[u^2\delta_s + uw - x_{fin}(uq)\right] \tag{2.36}$$

$$M_s = \frac{1}{2}\rho C_{L,\alpha} S_{fin} x_{fin} \left[u^2\delta_s + uw - x_{fin}(uq)\right] \tag{2.37}$$

$$N_r = \frac{1}{2}\rho C_{L,\alpha} S_{fin} x_{fin} \left[u^2\delta_r - uv - x_{fin}(ur)\right] \tag{2.38}$$

ρ denotes the water density, $C_{L,\alpha}$ represent the lift coefficient of the relevant control surface. The latter may be approximated numerically as a function of angle of attach α, the fin area S_{fin} and the distance x_{fin} with respect to the design center of origin (CO) of the vehicle.

The resulting actuator forces used for low-level control are:

$$\boldsymbol{\xi} = \begin{bmatrix} n & \delta_s & \delta_r \end{bmatrix}^T \tag{2.39}$$

2.6 AUV control systems

The main objective of a control system is to safely maneuver the vehicle to complete the mission objectives which may be subject to temporal, spatial or other constrains such as fuel consumption. A control system may be divided in three subsystems, namely *guidance, navigation* and *control* (GNC) [10] as shown in Figure 2.8.

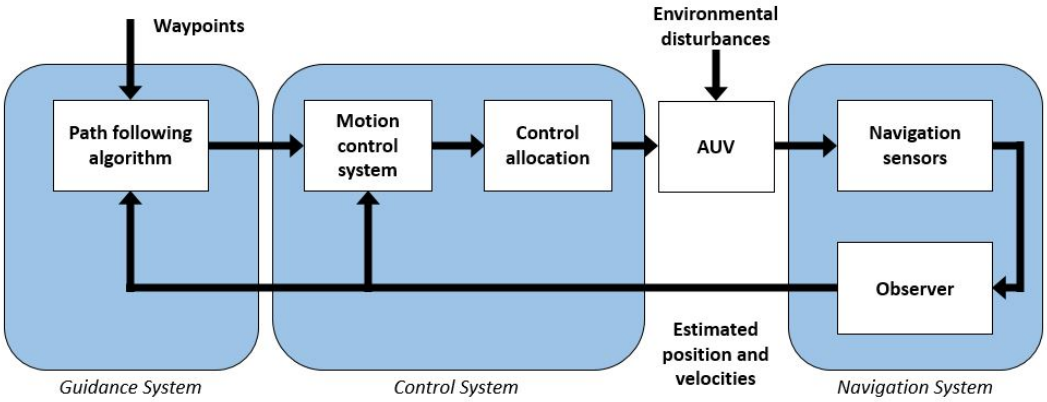


Figure 2.8: A block diagram for a typical motion control system, inspired by [10].

For the purpose of this project the main focus will be on the navigation system including state estimation.

2.6.1 Architecture of autonomy

Motion control will involve different levels of autonomy, from high level mission planning to low level actuator control. [35] defines three different levels using a "bottom-up" approach.

- Mission planner level: Here the mission objective is defined and the mission is planned. Subject to contingency handling, any input from payload sensor data analysis and any other input from the autonomy layer, the mission may be replanned.
- Guidance and optimization level: Handles waypoints and creates references commands for the controller to follow.
- Control execution level: At this level the plant control and actuator control takes place [34].

A simplified illustration of this typical architecture can be seen in Figure 2.9.

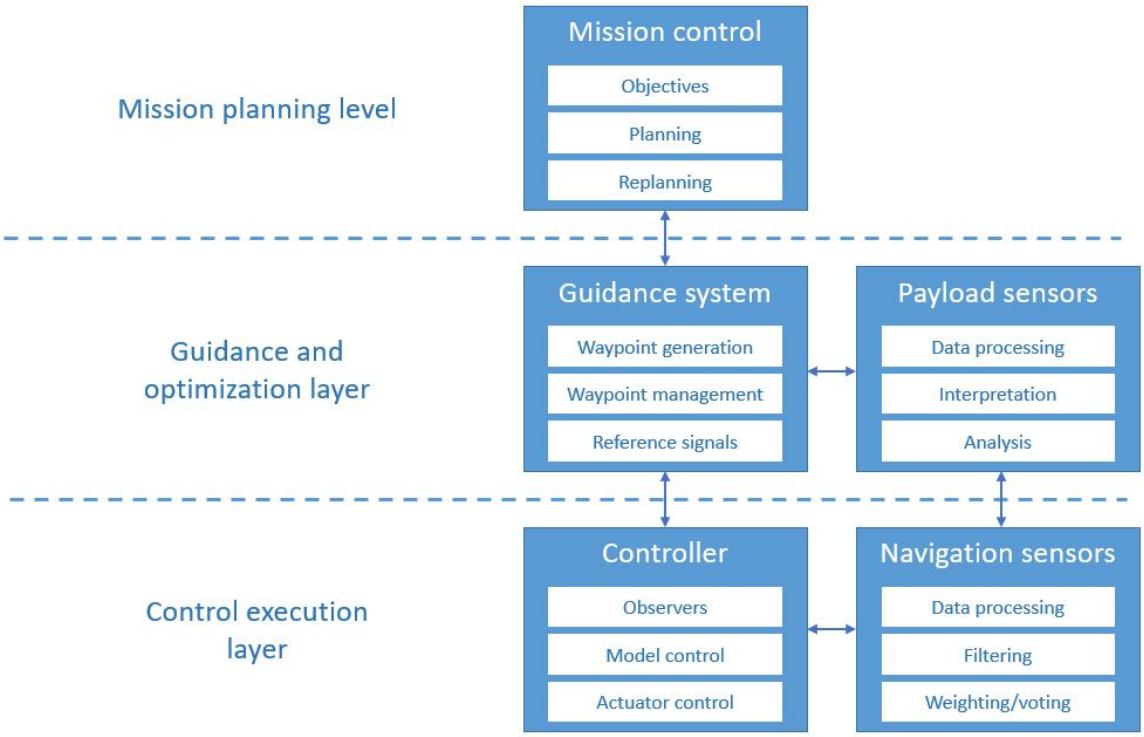


Figure 2.9: A simplified illustration of different control levels, adopted from [35]

The following three sections will briefly describe the control systems used in the AUV simulation environment. The simplified AUV model assumes that the subsystems for speed, heading and altitude are independent of each other, thus it is possible to design a separate control system for each of them.

2.6.2 Speed controller

The simulator's speed controller is a proportional-integral (PI) controller on the speed error, where the speed is defined as $U = \sqrt{u^2 + v^2 + w^2}$. It is further assumed that the speed may be regulated by using the shaft speed n as actuator input according to the

following control law:

$$n = k_{p,n}\left(U - U_d\right) + k_{i,n}\int_0^t \left(U - U_d\right) dt \tag{2.40}$$

Here $k_{p,n}$ and $k_{i,n}$ are the proportional and integral control gain respectively. U is the estimated speed by the observer as defined above and U_d is the desired speed.

2.6.3 Heading controller

The heading is controlled by commanding specific rudder fin angles, δ_r, governed by the following proportional-integral-derivative (PID) control law:

$$\delta_r = k_{p,\delta_r}\left(\psi - \psi_d\right) + k_{i,\delta_r}\int_0^t \left(\psi - \psi_d\right) dt + k_{d,\delta_r} r \tag{2.41}$$

Here, ψ is the estimated vehicle heading defined in {b} and ψ_d is the desired heading. k_{p,δ_r}, k_{i,δ_r} and k_{d,δ_r} are the proportional, integral and derivative control gain respectively.

2.6.4 Altitude controller

In the altitude controller, the desired altitude is achieved by controlling the pitch of the AUV. Two separate feedback loops are used in order to achieve this. The first loop is a PI controller for the altitude error and yields a desired pitch angle. The second loop is a PID controller that orders the desired pitch by use of the AUV stern fins. The two loops are governed by the following equations:

$$\theta_d = k_{p,\theta}\left(a - a_d\right) + k_{i,\theta}\int_0^t \left(a - a_d\right) \tag{2.42}$$

$$\delta_s = k_{p,\delta,s}\left(\theta - \theta_d\right) + k_{i,\delta_*}\int_0^t \left(\theta - \theta_d\right) dt + k_{d,\delta,q} \tag{2.43}$$

Here, θ_d and δ_s denote the desired pitch angle and the desired stern fin angle respectively. $a - a_d$ and $\theta - \theta_d$ is the estimated error in altitude and pitch respectively. The different k's are the controller gains systematically named by their subscript.

Note that altitude and depth are two different sides to the same story as they are always governed by the following equation:

$$z = a + t_i \tag{2.44}$$

Where z is the depth, a is the altitude (i.e. distance between the vehicle and the ice) and t_i is the ice thickness. In order words, by simply replacing a with z in the above control laws, its will work as a depth controller.

2.6.5 Guidance system

The simulation guidance system is based on a LOS lookahead approach as described in [10]. This steering law involves path-following along straight path segments between predefined waypoints in the horizontal plane, $WP_k = [x_k y_k]^T$. The desired altitude under the ice is hold constant.

Based on the estimated position from the navigation system, a crosstrack error e is calculated relative to the desired path. The desired course angle is calculated as follows:

$$\chi_d = \chi_p + \chi_r(e) \tag{2.45}$$

Here, χ_p is parallel to the vector between the previous and the next waypoint, WP_k and WP_{k-1} respectively. $+\chi_r(e)$ is the term that tried to push the AUV back towards the path. The higher error, the more aggressively the desired course is back towards the path such that:

$$\chi_r(e) := \arctan\left(\frac{-e}{\Delta}\right) \tag{2.46}$$

Δ is the lookahead distance, and may be found using various techniques. The desired heading is found by the following equation:

$$\psi_d = \chi_d - \beta \tag{2.47}$$

Here, the crab angle b takes into account the current in the ocean i.e. the sideways drift of the vehicle, and can be expressed as $\beta = \arcsin\left(\frac{v}{U}\right)$ where v is the velocity in sway and U is the total velocity of the vehicle.

For further details, see [10].

Chapter 3

Methods for AUV navigation and control

This chapter will present and describe the key contributions developed and implemented in this work. Most of the methods and implementations are original and based on the principles described in Chapter 2.

The first section will present the overall simulation environment (Arctic AUV Simulator) developed at NTNU by Norgren (PhD) in Simulink [28].

The following sections will focus on the implemented sensor models proposed in this thesis with the associated observer (EKF) before describing the MPC for altitude of the AUV in detail. Finally the algorithm that connects the UHI data with the MPC or traditional controller will be presented.

3.1 Arctic AUV Simulator

The simulation environment used in this master thesis was created in MATLAB/Simulink and C++. It was mainly created to test AUV guidance systems for iceberg detection and the mapping of icebergs or sea ice using simultaneous localization and mapping (SLAM). An AUV research mission to the near-coastal regions of the Weddel, Bellingshausen and Wilkes Land sectors of Antarctica by Williams [41] mapped several sections of sea ice which has been introduced to the simulator by Bremnes [3]. It was further modified in order to support LOS guidance of an arbitrary piece-wise linear 3D path, and the implemented altitude observer created by [3] has been replaced to fit the implemented UHI sensor model.

The vehicle model in the simulator is based on the American AUV Remus depicted in Figure 1.3. This particular AUV has one propeller in the aft in conjunction with to set of fins in the stern and rudder respectively. The numerical values representing the physics of Remus are included in Appendix A. The control system consists of three parts for separately regulating speed, heading and altitude as described in Chapter 2. Note that the altitude controller also contains a subcontroller regulating the pitch angle of the AUV.

The gains for these controllers along with the relevant parameters for the guidance law may be found in Appendix C.

The simulator was originally simplified in the way that it did not include any state estimators or disturbances in the sensor models. In other words, perfect knowledge about states such as positions, orientations and linear and angular velocities, were assumed at all times. In reality this can be compared to a scenario where the AUV operates in an area with a pre-installed high precision acoustic positioning system (HiPAP) infrastructure, such that it does not need to navigate with dead reckoning. Thus, it was a natural aim during the work of this thesis to add this functionality to the simulator making its behavior more realistic. This implementation will be further discussed in Section 3.2.

Based on the experience of PhD candidate Bremnes, the 7th ice map from Antarctica was chosen for all simulations in this thesis as it is the most complete sample and yield reliable results. By trial and error Bremnes found an appropriate initial state for this map which has been used for most simulations in this thesis. All simulation runs have been started from 225 meters north and 50 meters east in the horizontal plane. The initial depth has been set to 15 meters below the water surface when testing sensor models and observer, and only 10 meters when testing the UHI as a navigational sensor (with or without the MPC). Additionally the initial heading angle was set to 90° and the initial speed in surge was set to 1.3 m/s for all simulation runs. The remaining states were initiated at zero.

From this initial state, the guidance system is ordered to follow a list of waypoints in the horizontal plane which are included in Appendix B.

3.2 Sensor models and observer

This chapter will describe the implementation of sensor models (including UHI) and the sensor based observer in the Arctic AUV simulator described in Section 3.1. First, a brief introduction to various mathematical error models will be presented as these are used as building blocks in the design of the sensor models. Then, a more detailed description of the models for the different sensors will follow including the algorithm that compensates for magnetic declination based on the geographical location of the AUV operation. Finally, the observer implementation (EKF) will be presented in detail.

3.2.1 Sensor error models

To design realistic sensor models it is necessary to model the measurement error caused by all processes present during a mission. Examples can be electronic disturbances in MEMS, magnetic declination in compasses, calibration bias in a pressure sensor and measurement noise. Such errors may be represented by different mathematical models, the three described below will be the basis for the sensor models of this section and are inspired by [13].

Sensor	Standard deviation [σ]
Pressure gauge	0.001 [m]
Magnetic compass	0.062 [rad]
DVL	0.003 [m/s]
Gyroscope	0.0012 [rad/s]
Accelerometer	0.007 [m/s^2]

Table 3.1: Typical standard deviations for AUV navigation sensors from [8].

Misalignment error A bias such as a calibration error or a constant disturbance on the sensor may cause a constant bias error. However, assuming that the sensors are perfectly calibrated, most natural disturbances will vary over time, and will thus be modeled by either a colored noise or a white noise process.

Colored noise (Markov process of first order) Colored noise will have a power spectrum representing a certain bandwidth. It may be used to simulate a slowly varying process such as a bias changing with the environment. In this project, the colored noise will be represented using a Markov model of first order. The first order differential equation describing this process is as follows:

$$\dot{v} + \mu v = w \tag{3.1}$$

The block diagram for a first order Markov process may be seen in Figure 3.1. v represents the slow varying output signal, w is a Gaussian noise and μ is the time constant.

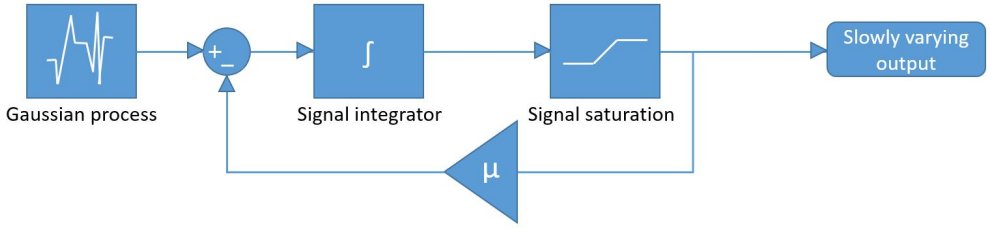


Figure 3.1: Block diagram of a first order Markov process as implemented in this thesis.

White noise (Gaussian noise) White noise generates a stochastic unpredictable signal suitable for modelling fluctuations made by electronic disturbances in the sensors. Such disturbances may often be approximated by a Gaussian distributed signal meaning that the signal is zero-mean and normally distributed with standard deviation [σ]. The standard deviation depends on the sensor and is often provided by the manufacturer. An overview of standard deviations for typical AUV navigation sensors are included in Table 3.1 from [8].

Sensor	$\boldsymbol{\sigma_G}$	$\boldsymbol{\sigma_M}$	$\boldsymbol{\mu_M}$
Pressure gauge	0.02 $[m]$	0.2 $[m]$	0.1 $[-]$
Magnetic compass	0.062 $[rad]$	0.062 $[rad]$	0.1 $[-]$
DVL	0.005 $[m/s]$	0.3 $[m/s]$	0.1 $[-]$
Gyroscope	0.0012 $[rad/s]$	0.035 $[rad/s]$	0.1 $[-]$
Accelerometer	0.007 $[m/s^2]$	0.05 $[m/s^2]$	0.1 $[-]$

Table 3.2: Parameters used when modelling the sensor error for all navigation sensors. $\boldsymbol{\sigma_G}$ and $\boldsymbol{\sigma_M}$ are the standard deviations used for the Gaussian noise and the Markov process respectively. $\boldsymbol{\mu_M}$ is the time constant used in the Markov process.

3.2.2 Navigational sensor models

To design a sensor model, the various error models from Section 3.2.1 are added to the perfect state signals in the simulator. However, the error models must be tuned to realistically represent the properties of the relevant sensors. A description of the different sensor model designs implemented in the simulator follows in the next sections.

Basic sensor building block

All the sensors are modelled to contain a slowly varying Markov process and a white noise as components to the final sensor measurement signal. For the sensors with multiple components such as the DVL or the accelerometer, all three components have been given different processes to create truly random sensor noise. The final measured signal from sensor i can be expressed as in the following equation:

$$\gamma^i_{measured} = \gamma^i_{true} + \varepsilon^i_{Markov} + \varepsilon^i_{Gaussian} \tag{3.2}$$

Here, $\varepsilon^i_{Gaussian}$ will have a corresponding standard deviation σ_G. ε^i_{Markov} will have a corresponding w with standard deviation σ_M and time constant μ_M as described in Section 3.2.1.

The parameter σ_G is mostly gathered from Table 3.1. However, some values have been increased in order to further test the robustness of the system. The parameters for the Markov process are adjusted accordingly to obtain appropriate results. A complete list of the implemented values can be found in Table 3.2.

Note that the UHI sensor is modelled in a more complex manner, and is separately presented in Section 3.2.3.

Compensation for magnetic compass declination

When modelling the magnetic compass sensor, another feature was implemented to compensate for the geographical declination angle discussed in Section 2.2. Taking latitude

and longitude of the mission's start point as input, a function has been implemented to calculate the declination angle and use this to compensate for the difference between the geographical and the true north pole. The calculation is based on the international database in [9].

The input used in this thesis is $78°N$ and $15.5°E$ which are the approximate coordinates of UNIS at Svalbard, Norway. Note that this part of the implementation shows the effects of the declination angle given a certain geographical point, and thus the coordinates of UNIS may be used even though the ice section used in the simulations is from Antarctica.

3.2.3 UHI model

An UHI is a complex sensor with many features and parameters. However, as the focus of the UHI in this thesis is on the application as a navigational sensor, a simplified model providing the necessary features will be sufficient. To process the full extent of data of which an UHI sensor measure as a payload sensor is a field of its own and outside the scope of this thesis.

The UHI block in Simulink will use depth and ice thickness as inputs, and give two outputs that are crucial for the purpose of navigation:

- radiance i.e. utility of measured data
- altitude between the sensor and sea ice

Radiance

The radiance is a measure of the light intensity that reaches the UHI sensor and is a good approximation of the utility of the payload data. As shown in Section 2.3.2 the radiance dissipates with increasing altitude. This is mainly driven by attenuation which is given by Equation 2.12. For the implementation in this thesis, the ice and water are both assumed to be homogeneous media, and are thus given one attenuation constant each. Note that these two constants are further assumed to be independent of the wavelength measured by the UHI sensor:

$$L(z, \hat{\xi}_{UHI}) = L_{sun}(e^{-c_i(t+t_s)} + e^{-c_w a}) \tag{3.3}$$

Here, $\hat{\xi}_{UHI}$ is the direction of the UHI sensor, which is assumed to point straight upward towards the ice. L_{sun} is the radiance from the sunlight in the direction of the UHI sensor, i.e. downwards. c_i and c_w is the attenuation constants for the ice and water respectively, and t and a are the ice thickness and altitude so that the following equation is always true.

$$z = a + t \tag{3.4}$$

Note that this assumes that the ice is perfectly flat at the surface and level with the water level. This assumption is not physical as ice has a lower density than water and in fact always rises a little higher than the surrounding ocean level (about 10% of the ice thickness). To account for this, the term t_s is added in Equation 3.5 and represents the thickness of snow, ice and slush that rises above the sea level. However, the same c_i is assumed for this part of the ice.

It is also important to note that two different light sources get measured by the UHI, namely the sunlight L_{sun} and the external light from the halogen lamp L_{ext}. The sunlight must travel through the ice a distance of t and the water a distance of a whereas the light from the external light source must travel through water to the ice and back to the AUV i.e. $2a$.

Finally, the backscatter effect β is added, which for simplicity is assumed to be constant as long as the AUV is fully submerged. This yields a total radiance as shown in the equation below:

$$L(z, \hat{\xi}_{UHI}) = L_{sun}(e^{-c_i(t+t_s)} + e^{-c_w a}) + L_{ext}e^{-2c_w a} + \beta \tag{3.5}$$

The radiance will be the sum of two terms. One will be based on the intensity of the sun light and the other on the intensity of the reflected light from the ice made by the external light source on the AUV. Both contributions will include a constant backscatter term and an attenuation which decreases exponentially with altitude. For simplicity, a constant attenuation coefficient will be assumed i.e. independent of wave length.

Altitude

The altitude will be estimated by an UHI sensor as described in Section 2.3.2 and in time of writing no published papers describe how this method works or performs in the field. However, there is no reason to believe that it is not possible to estimate the altitude at least as well as a traditional altimeter using this method. For the purpose of this thesis, the radiance will be the most important UHI contribution for selecting a desired altitude for the AUV, and therefore it is sufficient to create a fluctuating altitude estimate based on the same sensor model principles presented in the previous sections.

The UHI altitude will be a sum of the actual altitude and an error term:

$$a_{UHI} = a + a_e \tag{3.6}$$

Where a_{UHI} is the modelled altitude output of the UHI sensor, a is the true altitude and a_e is an error term modelling the offset caused by using an UHI sensor to estimate altitude. a_e is modelled as a slowly varying Markov process as described in Section 3.2.1.

3.2.4 Extended Kalman filter implementation

An extended Kalman filter (EKF) was designed taking the sensor model signals as input and performing state estimation sent further into the controller. The main motivation during this project has been to create a sensor based observer only using the kinematic equation of the AUV. For this to work, the linear acceleration measurements were used as an input for the prediction step in Kalman filter algorithm. Similarly, the angular rates measured by the gyroscope were used as an input to predict the attitude. Thus, the dynamics governing the EKF were as follows:

$$\begin{bmatrix} \dot{\boldsymbol{p}} \\ \dot{\boldsymbol{\Theta}} \\ \dot{\boldsymbol{v}} \end{bmatrix} = \begin{bmatrix} 0_{3x3} & 0_{3x3} & \boldsymbol{R(\Theta)} \\ 0_{3x3} & 0_{3x3} & 0_{3x3} \\ 0_{3x3} & 0_{3x3} & 0_{3x3} \end{bmatrix} \begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{\Theta} \\ \boldsymbol{v} \end{bmatrix} + \begin{bmatrix} 0_{3x3} & 0_{3x3} \\ \boldsymbol{T(\Theta)} & 0_{3x3} \\ 0_{3x3} & \boldsymbol{I}_{3x3} \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{a} \end{bmatrix} \tag{3.7}$$

Here, the state is comprised of $\boldsymbol{p} = [N\ E\ D]^T$, $\boldsymbol{\Theta} = [\phi\ \theta\ \psi]^T$ and $\boldsymbol{v} = [u\ v\ w]^T$. The system input are the measurements from the gyroscope and the accelerometer, $\boldsymbol{\omega} = [p\ q\ r]^T$ and $\boldsymbol{a} = [a_x\ a_y\ a_z]^T$ respectively. The kinematic matrices $\boldsymbol{R(\Theta)}$ and $\boldsymbol{T(\Theta)}$ are described in Section 2.5.

The relation between the states and the measurements are given through the measurement matrix $\boldsymbol{H}$.

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} \tag{3.8}$$

The sensors provide 8 measurements relevant for the state vector $\boldsymbol{x}$: 1 from the pressure gauge, 1 from the magnetic compass, 3 from the DVL and 3 from gyroscope. The resulting measurement relation is governed by the following:

$$\begin{bmatrix} y_{PG} \\ y_{MC} \\ y^u_{DVL} \\ y^v_{DVL} \\ y^w_{DVL} \\ y^\phi_{Gyro} \\ y^\theta_{Gyro} \\ y^\psi_{Gyro} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{\Theta} \\ \boldsymbol{v} \end{bmatrix} \tag{3.9}$$

Initialisation

For the Kalman filter algorithm described in Table 2.1 to work, some initial values must be defined.

As this sensor based approach to state estimation did not involve any aid from position

measuring sensors, the initial state $\boldsymbol{x_o}$ was set to waypoint 1 seen in Table B.1 in Appendix B.

The initial state covariance matrix $\boldsymbol{P}$ will converge as the algorithm process, and was set to $\boldsymbol{I_{9x9}}$.

$\boldsymbol{R}$ represents the measurement covariance matrix and was thus set to be a diagonal matrix of the sensor noise standard deviation from Table 3.1 to the second power.

$\boldsymbol{Q}$ represents the process covariance matrix and was set to $\boldsymbol{I_{9x9}} \times 10^{-3}$.

Linearization

Because the kinematics of the AUV navigation sensors are nonlinear, the system must be linearized around a specific set point in order to use the Kalman filter algorithm. However, Equation 2.31 is only nonlinear in attitude which can be exploited to lower the computational time of calculating the Jacobian matrix for all 9 states.

To lower computational time even further, the linearized systems may be precalculated offline for a number of operating points, e.g. every combination of attitudes given as integers in degrees. This is often done for industrial purposes as it is reliable and robust. However, this would effectively yield $360^3 = 46\ 656\ 000$ combinations which could be computationally heavy. Another disadvantage using this approach is that no matter how close the chosen setpoint is to the current attitude, it will never be completely accurate.

For this project the system was linearized real time around the exact state of interest which gives an even more accurate result. A Matlab function in the EKF simply takes in the attitude of the previous state estimation and insert it into Equation 3.9, evaluating the matrices $\boldsymbol{R(\Theta)}$ and $\boldsymbol{T\Theta)}$ which results in an LTI system as the matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ now are constant and independent of the states. The LTI system is then discretized yielding the matrices $\boldsymbol{\Phi}$, $\boldsymbol{\Delta}$ and $\boldsymbol{\Gamma}$ which are further used in the Kalman filter algorithm as described in Table 2.1.

Signal syntheses

Finally, the angular velocities measured by the gyro are added together with the 9 states from Equation 3.9 to create the final state estimate from the EKF. These twelve estimated states are further sent into the controller as described by the equations in Section 2.5.

3.3 Method for choosing desired altitude based on UHI data

Given the sensor models, observer and UHI model developed in the former sections, this section will describe the key algorithm that combines these implementations into a method for using UHI as a navigational sensor. The UHI sensor yields two outputs, namely the

measured altitude and radiance. These two parameters will through an algorithm result in the final desired altitude that is sent to the altitude controller or MPC.

The algorithm should also include basic abort strategies in case of emergency such as vehicle loss (e.g. loss of contact), collision, hardware failure and low battery capacity. Finally, the algorithm should ensure that the measured radiance is properly registered and that a log is kept for the user, especially when the radiance is to low for sufficient post processing of the data.

This algorithm must be able to use three main parameters to choose an appropriate desired altitude.

- Collision risk
- Spatial coverage
- Data utility

These three terms must be taken into account and put up against each other in order to find a desired altitude. In general, it will be challenging to evaluate the three aspects, and it is still normal that a human operator chooses the evaluation of these parameters.

3.3.1 Collision risk

For this algorithm, the risk of collision will increase with decreasing altitude. Thus, a term will be added that increases the desired altitude if the altitude goes beyond a certain threshold. Furthermore, another threshold that the desired altitude cannot surpass (i.e. a minimum desired altitude) is introduced.

3.3.2 Spatial coverage

The spatial coverage is challenging as the further away from the ice the AUV is, the more spatial area the UHI sensor will be able to monitor and measure. Simultaneously, every increase in altitude will be at the expense of the radiance due to the attenuation effect described in Section 2.3.2.

3.3.3 Data utility

Finally, the utility of the UHI data must be taken into account. As described earlier, this is highly correlated with the radiance measured by the UHI sensor. However, in reality the measurement quality will depend on variables such as the properties of the ice, algaes, particles in the water, salinity etc. However, there is no point in having a bigger radiance than needed, so if this value is sufficient, it will be a goal of the algorithm to back away from the ice as much as possible to maximize spatial range and minimize the risk of collision.

3.3.4 Control mode

There are numerous ways to combine these parameters in order to find an appropriate desired altitude. However, there is one important principle when deciding what distance the AUV should be from the ice. The data utility should simply be over a certain threshold at all times. If the UHI data is to poor to be post processed and analyzed, the entire objective of the mission is gone. This is either the case due to equipment failure, bad ice and/or water conditions or simply that the AUV should go closer to the ice.

On the other hand, if the AUV can move unconstrained towards the ice in order to maximize the data utility, the risk of collision may increase dramatically. Hence, a constraint that the desired altitude may not surpass should be enforced.

The two former thresholds might get in conflict with each other resulting in an infeasible solution, i.e. the altitude constraint must either be broken in order to get close enough to the ice for proper data gathering or vice versa. However, safety must be prioritised over completing the mission goals and at least the AUV can log locations and time when the data quality was not usable due to bad conditions. It is always possible to return later in better conditions than taking the risk of losing expensive equipment in the vulnerable nature the polar regions represent.

This thesis proposes a simple, yet powerful mechanism to deal with this situation. Two different control modes are defined, based on the two thresholds presented above. It is either attractive to approach or retreat the ice based on the radiance measured by the UHI sensor, and both the threshold constraints below must be satisfied at all times:

- L_{min}, Threshold on radiance i.e. data utility
- a_{des}^{min}, Threshold on minimum desired altitude

If it is attractive to get closer to or further away from the ice depends on the radiance threshold, L_{min}. The idea behind this approach is that for a certain radiance, L_{min}, the collected UHI data would be worthless. In other words, if the radiance is lower than this value i.e. $L(a) < L_{min}$ it is desirable to go closer to the ice. If $L(a) > L_{min}$ the goal is actually the opposite, as more altitude will increase both safety and spatial range. To change the desired altitude in one of the two direction an alternation term is defined as $\Delta a = |L - L_{min}|\ r$ where r is a reaction coefficient.

$$\mathrm{a_d}(L) = \begin{cases} \hat{a} + |L - L_{min}|\ r & L(a) > L_{min} \\ \hat{a} - |L - L_{min}|\ r & L(a) \leq L_{min} \end{cases} \tag{3.10}$$

$\hat{a}$ denotes the estimated altitude from the UHI sensor model. When this desired altitude is calculated, it is passed through a saturation block where it at least is given the value of a_{des}^{min}. This ensures safety, but might result in data that have a radiance too low to be post processed and used for the purpose of the mission.

3.4 Model Predictive Control

The MPC was implemented to create a robust framework for further enhancement of using the UHI as a navigational sensor. It also represents an optimal control method which is a new contribution to the Arctic AUV Simulator. The proposed MPC is based on the kinematic model of the AUV, in this case regulating the pitch of the vehicle in order to obtain the desired altitude.

The kinematic equation with the states of interest is:

$$\begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{a}} \\ \dot{\boldsymbol{\theta}} \end{bmatrix} = \begin{bmatrix} cos(\theta) & 0 \\ -sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{q} \end{bmatrix} \tag{3.11}$$

In other words, this is formulated as a two dimensional problem where the surge, altitude and pitch are the relevant states and the control inputs are speed and pitch rate. However, as the simulation guidance system used the path following approach, the reference speed for the AUV is originally set constant at 1.3 m/s. Thus, the AUV has no desired states in the horizontal plane. It is therefore reasonable to set u as a free variable, and use the current speed as an input for the MPC. This assumes that u stays approximately constant for the duration of the MPC time horizon. In other words, the only output of the MPC will be q i.e. the pitch rate.

The MPC was implemented in Simulink using a Matlab system block. The required inputs for solving the NLP optimisation problem were the relevant states (in this case x, a and θ), the reference signal, i.e. the desired altitude and finally the measured current speed u as mentioned above. The output of the MPC block is the pitch rate q.

Because the MPC runs real time computational time is essential and the large number of variables and parameters must be taken into account. The proposed solution is to use the Matlab system block with the property *interpreted execution* in order to declare predefined symbolic parameters and governing equation in the source code. This is just done once as an initialising step drastically reducing the run time.

This first part of the MPC source code defines symbols for the relevant parameters, and adds their governing equations. Here, Equation 3.11 is formulated along with the cost function and associated kinematic constraints (see Equation 2.8). All these symbolic expressions are created by iterating over the N time steps. The cost function is defined as shown in Equation 2.7. Further, there is assumed no couplet terms between the states, so that both Q and R are diagonal matrices. Q will be of three dimensions and R only one as the only control input to be decided is q.

$$Q = \begin{bmatrix} Q_x & & \\ & Q_a & \\ & & Q_\theta \end{bmatrix} \quad , \quad R = R_q \tag{3.12}$$

The second and final part of the source code introduces the actual input values to the MPC, namely the surge, altitude, pitch, speed and reference altitude a_d. These values are then associated with their respective symbolic expressions, before the NLP can be formulated. Casadi (a third party open source NLP solver) is imported into Matlab. This toolbox then attempts to find an optimal solution, and if successful the new and updated pitch rate q is send as an output of the system.

When the optimal pitch rate q is determined, the next block in Simulink then ensures that the reference pitch is calculated and sent to the pitch controller. This is done by kinematic integration, where Q is inserted into Equation 3.11 to obtain the time derived states. These are then numerically integrated, and the pitch, which now is the desired pitch angle is further sent into the pitch controller to obtain the optimal stern fin angle. Finally, the AUV dynamics will ensure that the desired altitude is reached.

3.5 Hardware platform architecture

Another contribution in this work is an architecture for the hardware of the super underwater vehicle (SUV) described in Section 1.2.5 consisting of two Blueyes mounted together. Originally, this work was intended to result in a full hardware test using the developed UHI navigational system on the SUV, but unfortunately this was not feasible due to the global pandemic of Covid-19. However, the work still contributes towards creating such a setup in the future, and is therefore relevant to present the architecture in this section.

An image of the SUV is seen in Figure 3.2.

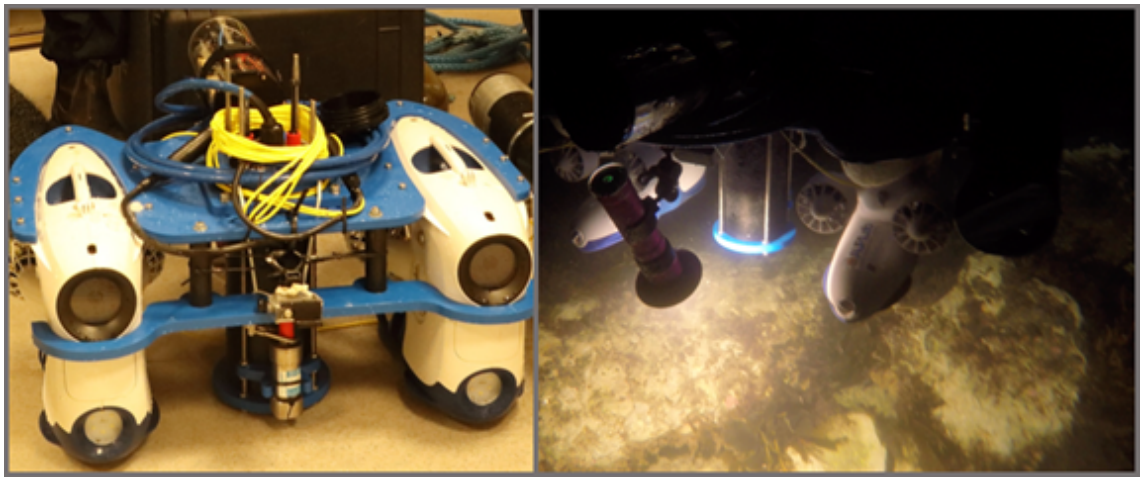


Figure 3.2: The SUV Dual-Vehicle Configuration of the ROVs. The black cylinder in the middle shows the UHI sensor. The left hand frame shows the vehicle on land and the right hand frame shows the vehicle during an operation in Kongsfjorden, Svalbard. Courtesy of [24].

The SUV has a greater payload capacity than the separate ROVs, which allows it to carry a relatively big payload sensor such as an UHI. However, when the UHI is to be used as a navigational sensor, the different components need to communicate real time, and it is reasonable to propose an architecture for the setup as a whole. The remaining part of this section will describe the architecture proposed in Figure 3.3.

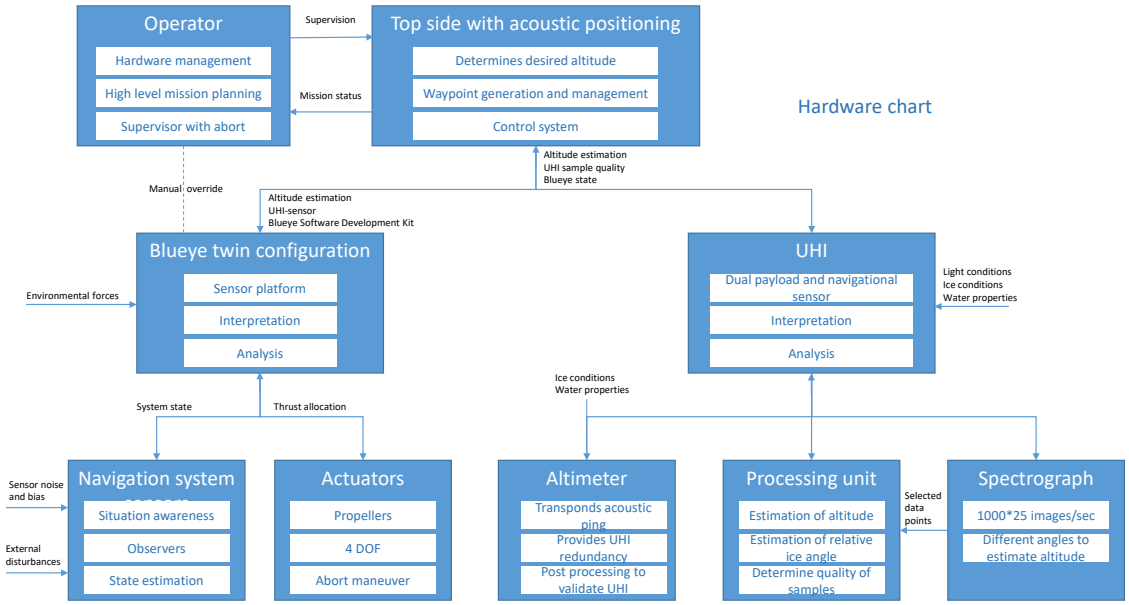


Figure 3.3: A chart showing the proposed architecture of the various components and the flow of data for the twin-Blueye SUV platform and the associated UHI sensor.

The top layer consist of the operator and the top side i.e. the surface point to which the SUV is connected. They communicate between one another through a user interface so that the operator can supervise and even manually overtake the mission. The topside receives necessary data from the remaining system to perform control allocation, find the desired altitude based on UHI data and manage waypoints and guidance.

The second layer consist of the SUV and the UHI. They do not directly communicate with each other, but most of the data essential for completing the mission is sent through the topside.

The SUV has an on board navigational system and a set of actuators. The actuators and control system as a whole will be distinctly different than for a single Blueye ROV and must be altered accordingly [24]. Similar to a single Blueye ROV the whole SUV operates with only 4 DOFs namely surge, sway, heave and yaw. The actuators might also have predefined abort maneuvers. The SUV setup also has a navigational system performing state estimation to account for external disturbances.

The UHI sensor consists of three different hardware units, namely an altimeter, a processing unit and the spectrograph. The latter captures as many as 1000 images (one measurement) 25 times a second. In time of writing, this is too much data to be processed real time, but a small fraction of it is selected based on the nature of the mission. This is sent real time to the processing unit where it can be used to estimate altitude. The altimeter is mostly part of the setup to provide redundancy for the processing unit, and to allow the operator to validate the quality of the altitude estimates made by the UHI sensor after the mission.

Chapter 4

Results and discussion

This chapter will present and discuss the most important results of this master thesis. Unless otherwise specified, the theory, methods and parameters are implemented in the way described in Chapter 3 for all the sections of this chapter.

The various simulation cases will be presented in a way so that it is possible to reproduce the results using Matlab/Simulink and the files and folders attached as part of the submission of this thesis.

4.1 Navigation sensor models for the Remus AUV

The sensor models created a realistic measurement output based on the perfect state reading of the simulator. The mathematical models used for the different sensors are further described in Section 3.2.2. Plots of the sensor measurement error (i.e. $\gamma_\varepsilon = \gamma^i_{measured} - \gamma^i_{true}$) follow in Figures 4.1-4.5. The high frequent measurement noise can be seen slowly varying due to the underlying Markov process as expected.

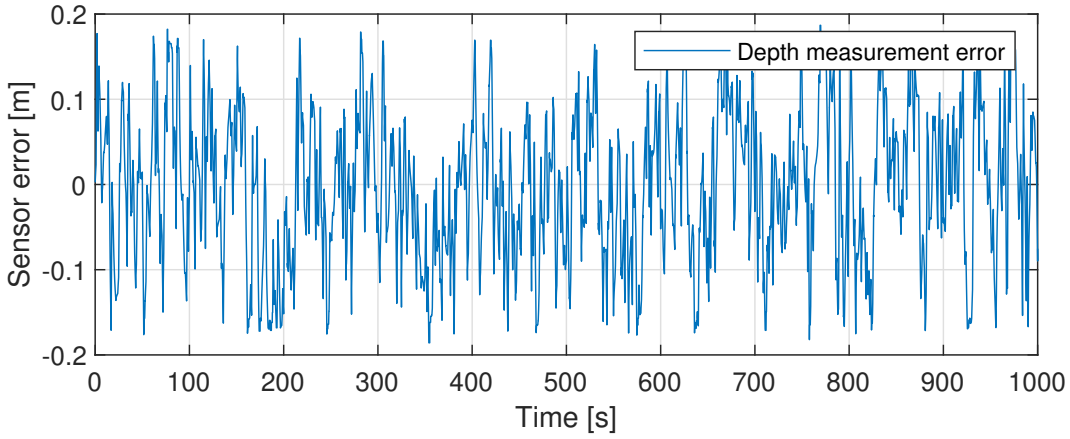


Figure 4.1: Error in the depth measurement based on the pressure sensor.

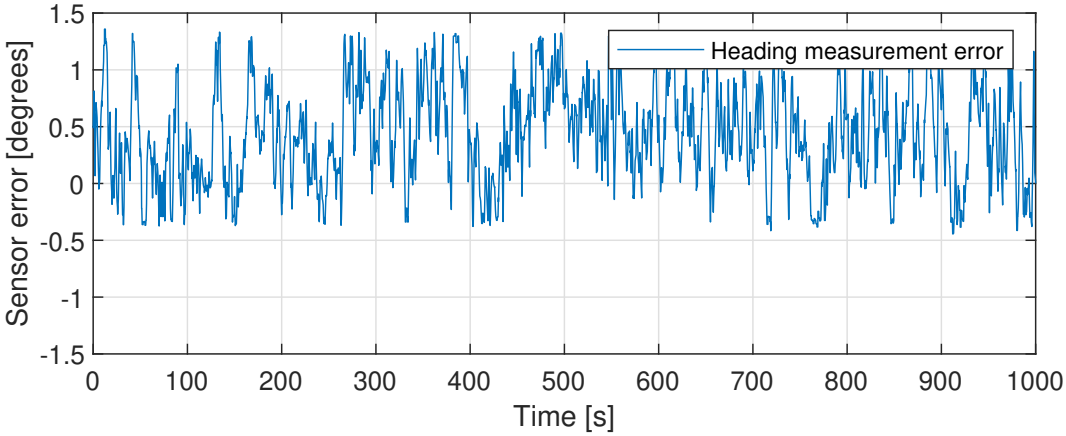


Figure 4.2: Error in the heading measurement based on the magnetic compass. Note the constant bias due to the declination angle.

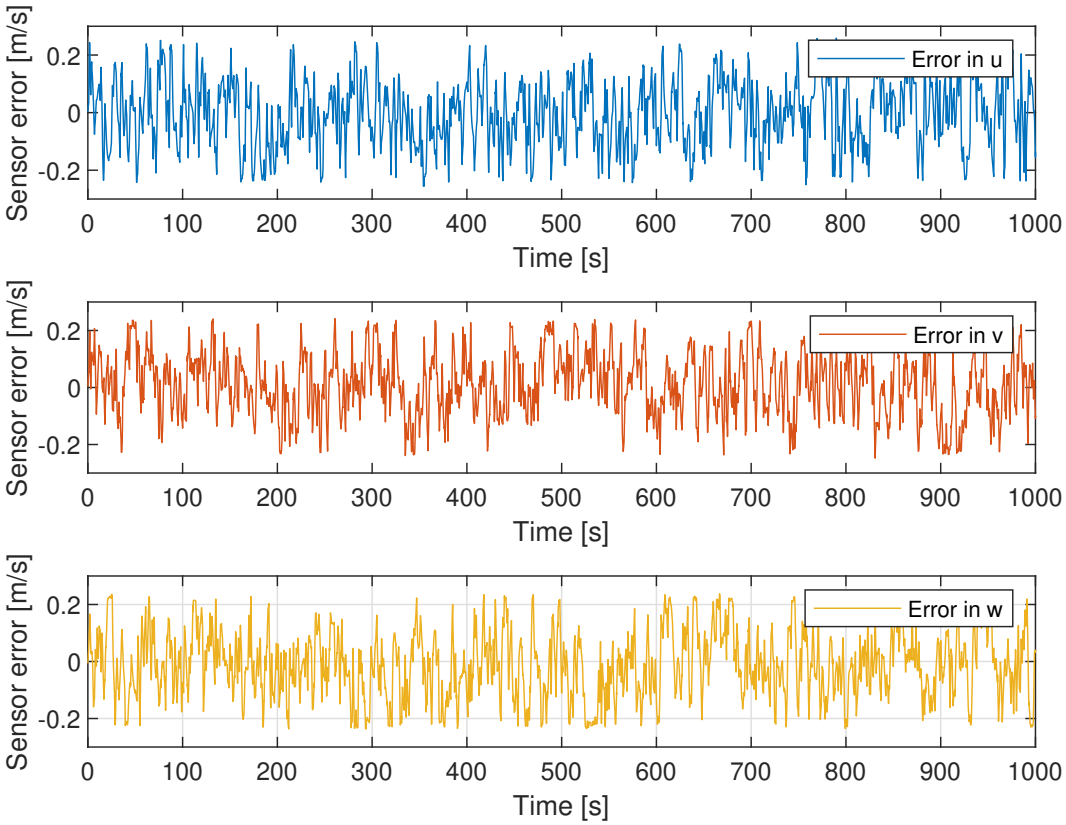


Figure 4.3: Error in the linear velocity measurement based on the DVL sensor.

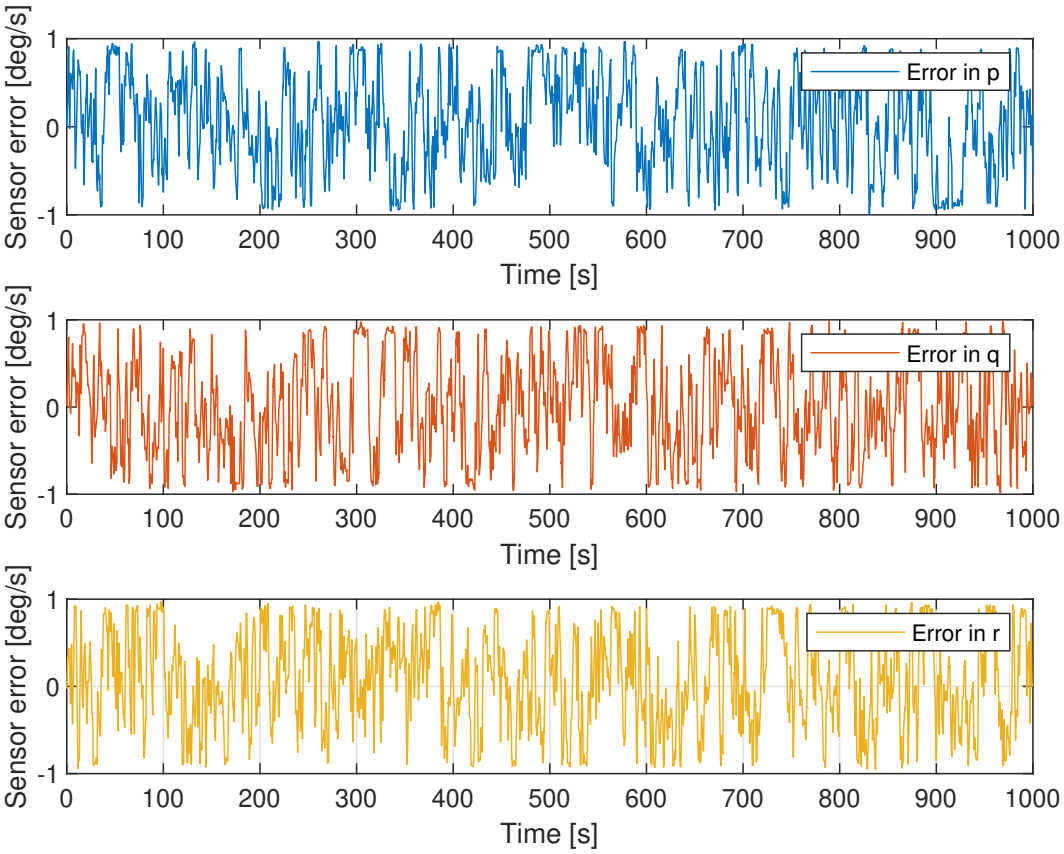


Figure 4.4: Error in the angular rate measurement based on the gyroscope.

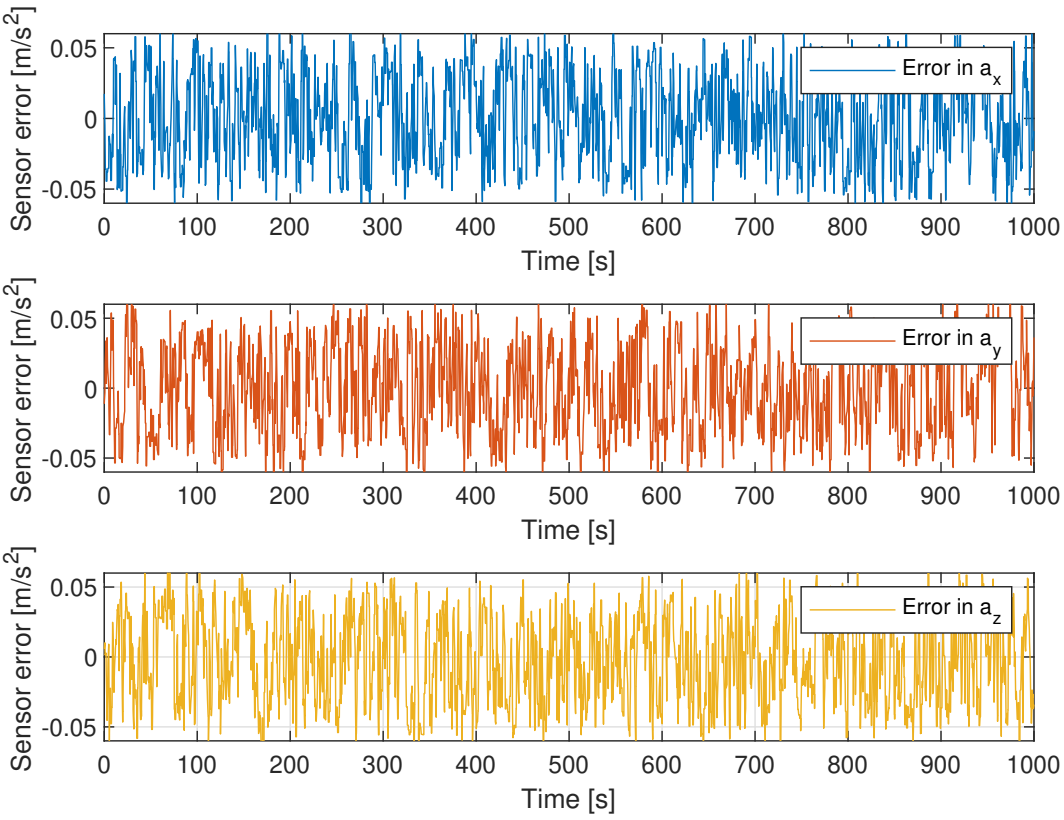


Figure 4.5: Error in the linear acceleration measurement based on the accelerometer.

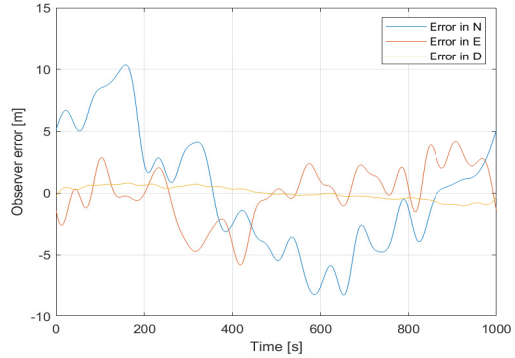
As seen in Figure 4.2 the magnetic compass measurement has a constant bias offset due to the declination angle which makes its mean value different from zero.

Even though these plots might seem alike it is important to look at the magnitude on the y-axis and note that they reflect the error parameters for the respective sensors listed in Table 3.2. A close inspection reveals that the sensors differ quite a lot with respect to the density of noise and the dominance of either the slowly varying Markov process or the white noise.

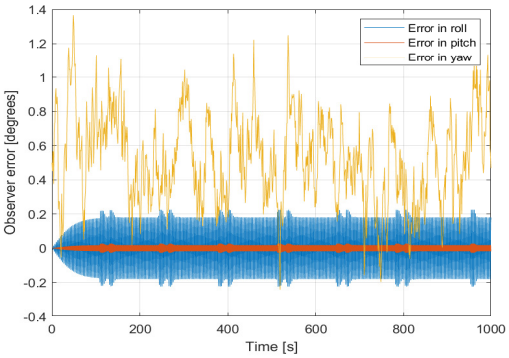
4.2 Observer performance

To validate the observer performance the state estimation values from the EKF were compared with the true state of the simulator. Figure 4.6 shows the state estimation error, i.e. $\tilde{\boldsymbol{x}} = \hat{\boldsymbol{x}}_{\boldsymbol{EKF}} - \boldsymbol{x}$.

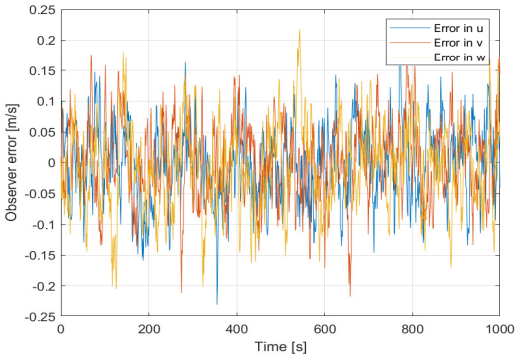
Figure 4.6a shows that the NED errors drift slowly in time. The horizontal states (i.e. surge and sway) drifts the most with a magnitude of around 10 meters. The error in depth is far lower because this state is directly measured by the pressure sensor. For the errors of the attitude estimation (Figure 4.6b), the roll and pitch oscillates through the entire time series with two distinct spikes in amplitude every time the AUV turns. The yaw however, has a noisy slow varying error. The EKF error for the linear and angular



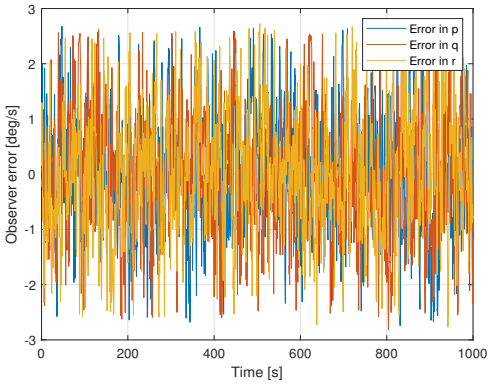
(a) State estimation error in NED.



(b) State estimation error in attitude.



(c) State estimation error in linear velocities.



(d) State estimation error in angular rates.

Figure 4.6: A complete overview that shows the extended Kalman filter performance based on the sensor noise given in Section 3.2.2.

velocities shown in Figure 4.6c and Figure 4.6d respectively are both dominated by a combination of colored and white noise. The angular velocities have a more dense noise spectrum which reflects how these states are measured by a sensor with more uncertainty than the linear velocities. Note that the noisy gyroscope measurements are sent directly into the controller and are never filtered by the observer. Ideally, this should be filtered out beforehand.

4.3 Overall performance of sensor models and observer

The overall system performance incorporating both the simulation models and the EKF were evaluated by comparing results from the original version of the simulator as described in Chapter 3 with the newly implemented sensor models and observer. As the former version of the simulator assumed perfect state estimation, it was expected that the sensor models would introduce noise and uncertainty and that the overall performance would decrease. However, this is an important improvement to the simulator as it is more realistic and tests the system performance more robustly.

As expected, the AUV drifts as its navigation system must rely fully on internal sensors i.e. using the dead reckoning approach. However, the EKF appears to filter out most of the Gaussian distributed sensor noise and thus making an important improvement to

the inherent disturbance of the sensors. Despite that the Markov process adds a bias that makes the AUV drift, it may seem from the results, that because this slowly varying process alternates around zero it in time compensates for its own drift making the AUV move back towards the desired path. The latter is especially prominent in Figure 4.6a.

Figure 4.7 and Figure 4.8 clearly illustrate the difference between simulating using perfect state feedback and sensor models respectively. Despite the AUV drifting in Figure 4.8, it still manages to get within the radius of acceptance for most of the waypoints. However, it actually misses the second waypoint from the top right, but because its navigational system believes it is within the radiance of acceptance it is automatically reassigned to the next waypoint. A full three dimensional illustration of the AUV navigating in a lawn mower pattern beneath the ice may be seen in Figure 4.9.

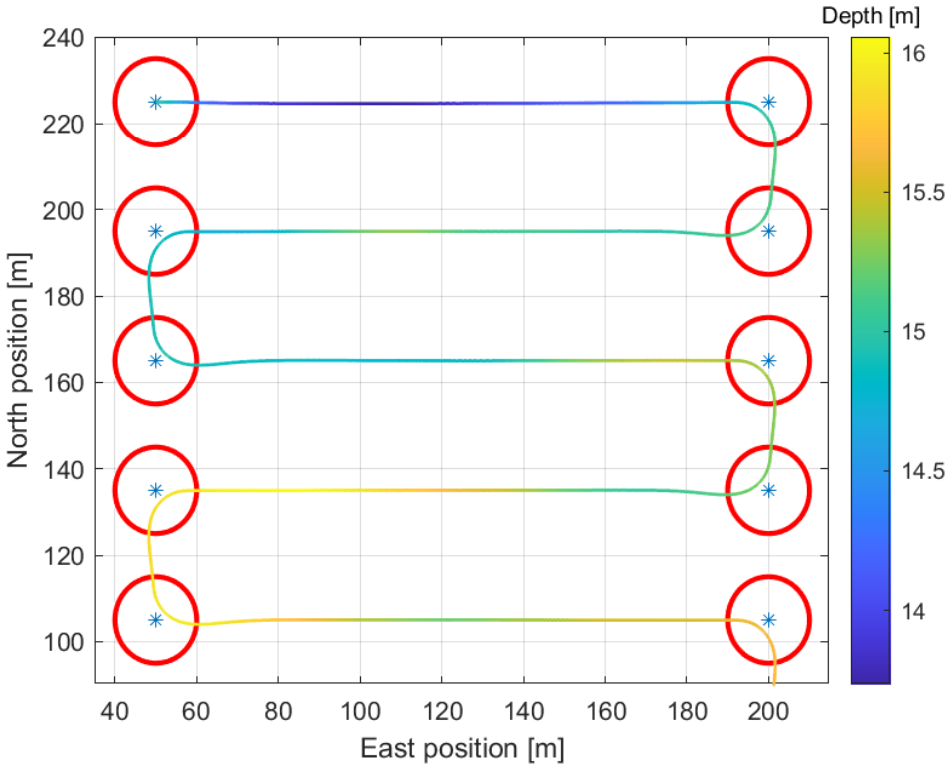


Figure 4.7: A plot of the AUV path during a 1000s simulation with perfect state feedback. The red circles indicate the radius of acceptance for the waypoints. The color represents the depth.

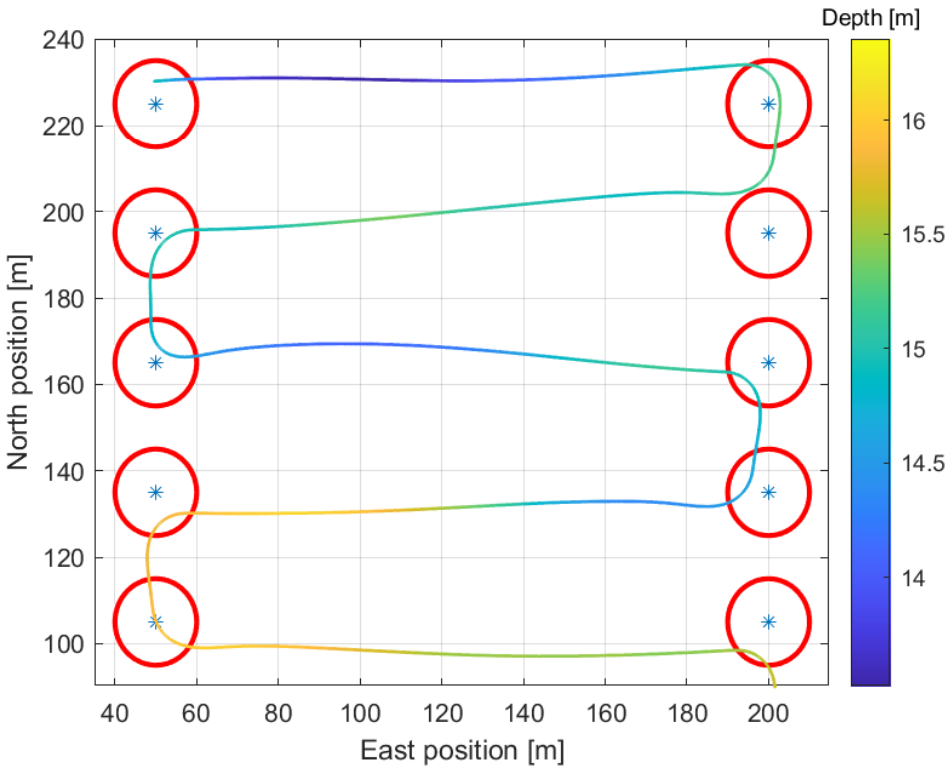


Figure 4.8: A plot of the AUV path during a 1000s simulation with sensor models and state estimation feedback. The red circles indicate the radius of acceptance for the waypoints. The color represents the depth.

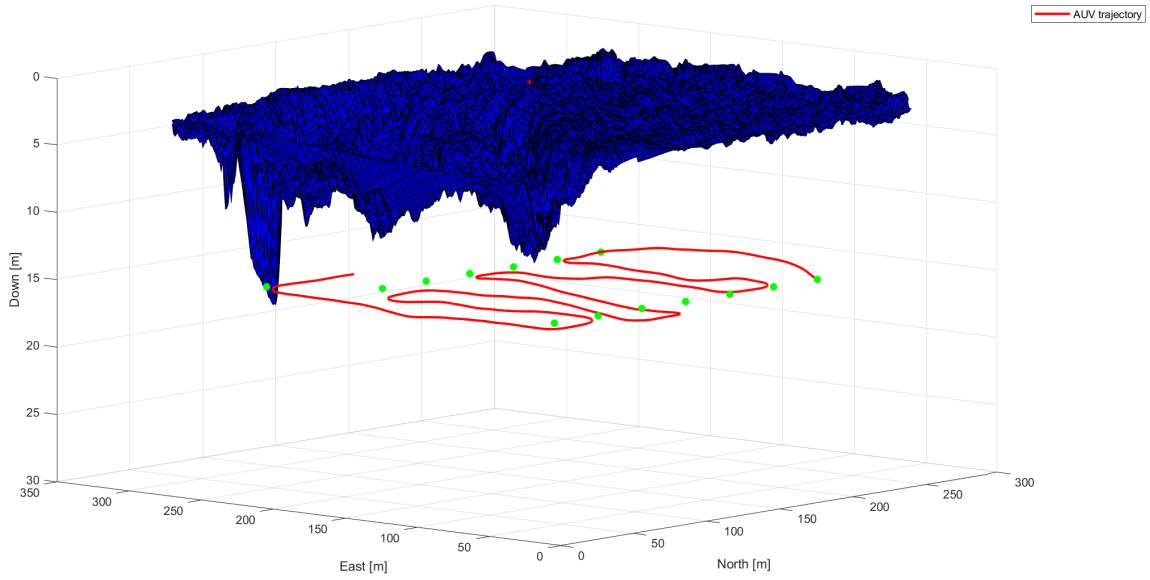


Figure 4.9: 3D view of the AUV navigating successfully under the ice sheet sampled in Antarctica. The red line illustrates the AUV's trajectory.

Variable	Symbol	Value
Ice attenuation coefficient	c_i	$1\ [\frac{1}{m}]$
Sea water attenuation coefficient	c_w	$0.2\ [\frac{1}{m}]$
Snow thickness	t_s	$0.1\ [m]$
Backscattering	β	$5\ \left[Wm^{-2}sr^{-1}nm^{-1}\right]$
Radiance from sunlight	L_{sun}	$300\ \left[Wm^{-2}sr^{-1}nm^{-1}\right]$
Radiance from external light source	L_{ext}	$2000\ \left[Wm^{-2}sr^{-1}nm^{-1}\right]$
Total reflected radiance threshold	L_{min}	$35\ \left[Wm^{-2}sr^{-1}nm^{-1}\right]$
Reaction coefficient	r	$\frac{1}{35}\ \left[W^{-1}m^3srnm\right]$
Minimum desired altitude	a_{des}^{min}	$4\ [m]$

Table 4.1: The values used for the general simulation case of the algorithm using UHI as a navigational sensor.

4.4 Altitude control using UHI as a navigational sensor

This section will test the overall performance of the UHI system, both the UHI sensor model from Section 3.2.3 and the control algorithm for altitude using UHI described in Section 3.3. A general simulation case will first be presented in order to validate the performance of the system before two special cases are presented. These two cases will represent the spectrum of different simulation settings as they will represent a risky and a conservative approach.

4.4.1 General case

The intention of the general case is to validate that the method works as expected and to illustrate its potential based on the previous sections. This general case is meant to use normal parameters for the simulation so that the system can behave in a stable and reliable way. The parameters used here are all described in detail in Section 3.3 and Equation 3.10 and their numerical values for the general case can be found in Table 4.1.

First of all, note that c_i is about five times as big as c_w. This is because the ice has more impurities and absorbs more energy than water, thus the attenuation is greater. The snow thickness t_s is set to 0.1 m, which is relatively small. However, the sea ice was almost snow free during summertime in Antarctica when the relevant ice section was explored. Unless this particular section has any ridges or other structures on the surface, it is fair to assume that the surface is flat. Regarding the backscattering, this is a term that will depend on altitude because it also has an attenuation effect in the water. However, for the distances in question i.e. less than 20 m, it is reasonable to assume that the backscattering is constant. Regarding the radiance from the two light sources, i.e. the sun and the external source, the latter is set to be much more powerful than the other. In the polar regions, the sun is low in the sky and does not produce much radiance. The external light sources, which are often halogen or LED based, can produce impressive amounts of radiance which is necessary since their omitted light has to travel round trip to the ice and back to the UHI sensor.

Finally, the radiance threshold is set to 35 which is a normal value for decent UHI data.

The reaction coefficient is set to its inverse, which implies that if the radiance is zero, the desired altitude will be 1 meter shorter than the current altitude (see Equation 3.10). For safety, the minimum desired altitude is set to 4 meters, which should be sufficient to avoid collision with the ice.

Figure 4.10 shows an overview of the simulation as the red path of the AUV stretches under the blue sheet of ice scanned in Antarctica. It is clear that the AUV successfully navigates by the instructed lawn mower pattern, and that it changes altitude due to the change in ice thickness.

A bird's-eye view of the same situation may be seen in Figure 4.11. This clearly shows that the AUV successfully navigates in the horizontal plane thanks to the LOS guidance system. From the two lowest stretches from East to West, it appears that the AUV is diving a little deeper.

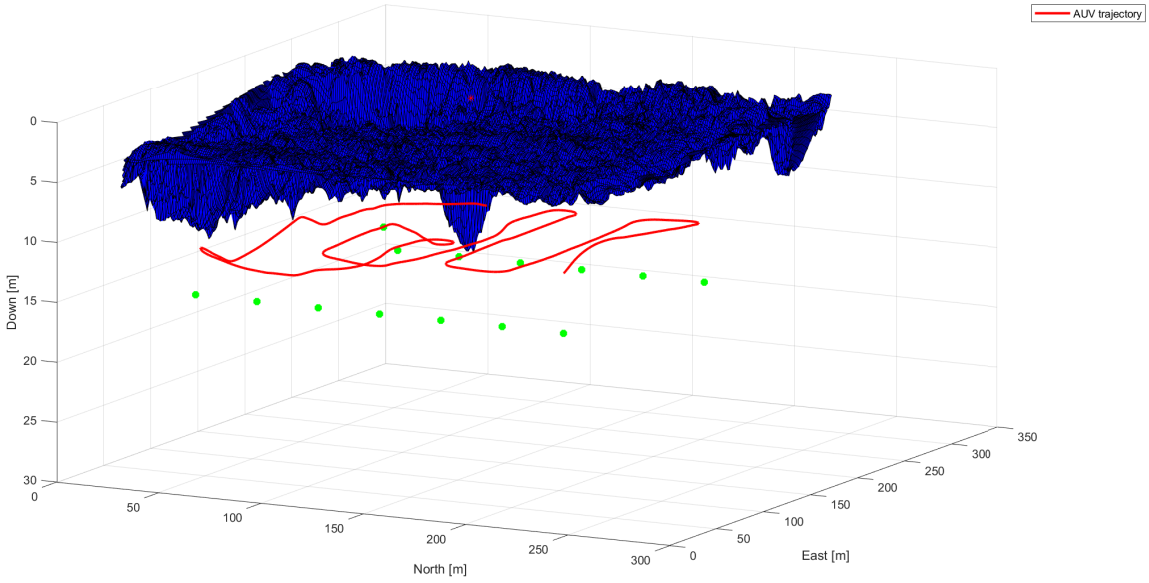


Figure 4.10: Showing the AUV successfully navigating under the blue ice sheet from Antarctica. The green dots symbolise the waypoints inserted at a depth of 15 m.

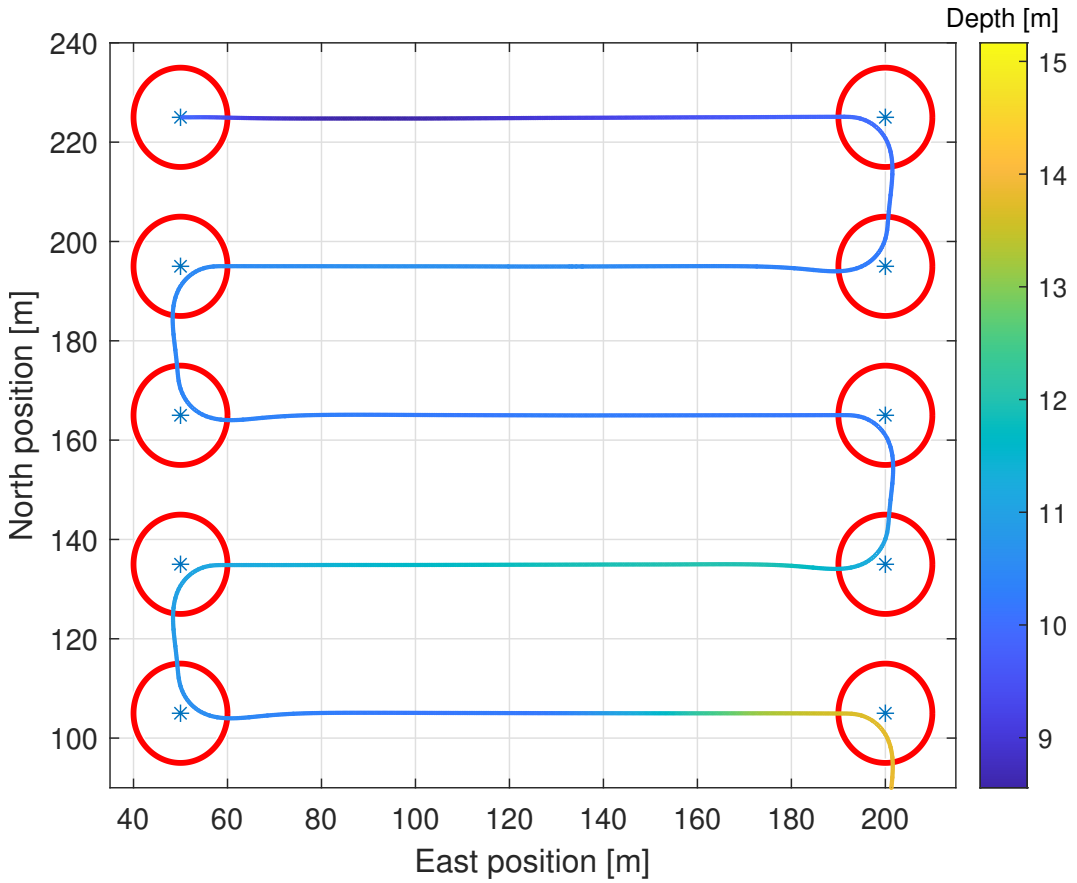


Figure 4.11: Graph showing a bird's-eye view of the AUV navigating successfully through the lawn mower pattern. The red circles illustrate the radius of acceptance of which the AUV must pass through before going to the next wayoint.

The two following plots show the two-dimensional version of the lawn mower pattern.

By looking carefully at Figure 4.12 one can observe the lag between the actual depth and the desired depth. The latter is simply the sum of the desired altitude and the ice thickness, i.e. $a_d + i_t$, and is caused by the time it takes between the initialisation of the trust allocation until the AUVs pitch and altitude actually change.

It is also clear that the ice has several sections that abruptly point far down into the water, and the ice thickness increases rapidly e.g. at 400, 600 and 800 seconds. This abrupt changes in ice thickness changes the radiance and thus the desired altitude. However, it takes time for the dynamical system to follow, and despite implementing $a_{min} = 4m$ as a safety margin, it appears that sufficiently high ice gradients such as the one at 800 seconds may result in a collision.

From Figure 4.13 it is clear that the UHI algorithm performs as desired. It shows a negative correlation between desired altitude and the ice thickness. This is expected, as thicker ice lowers the radiance and makes it attractive for the AUV to approach the ice and vise versa. Note that it is not a linear negative correlation, and that the desired altitude has a lower variance than the ice thickness due to the exponential nature of the radiance.

Furthermore, Figure 4.13 shows that whenever the ice gradient is sufficiently high (i.e. the thickness changes rapidly) so is the desired altitude gradient. The AUV actuators are not designed to follow such big changes in reference signal, and this results in an increased state error for the AUV in these locations.

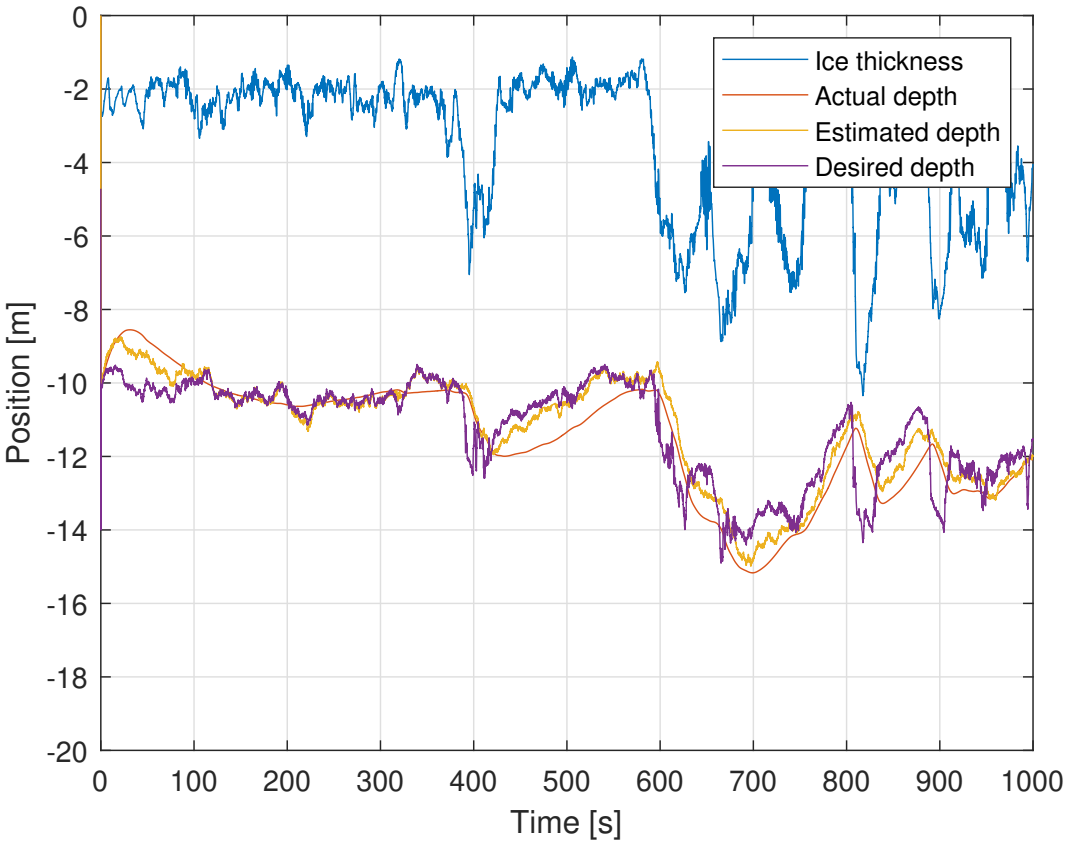


Figure 4.12: This graph illustrates how the AUV dives under the ice seen from the side. The difference between the actual depth and the estimated depth is the slowly varying Markov process added in Section 3.2.3.

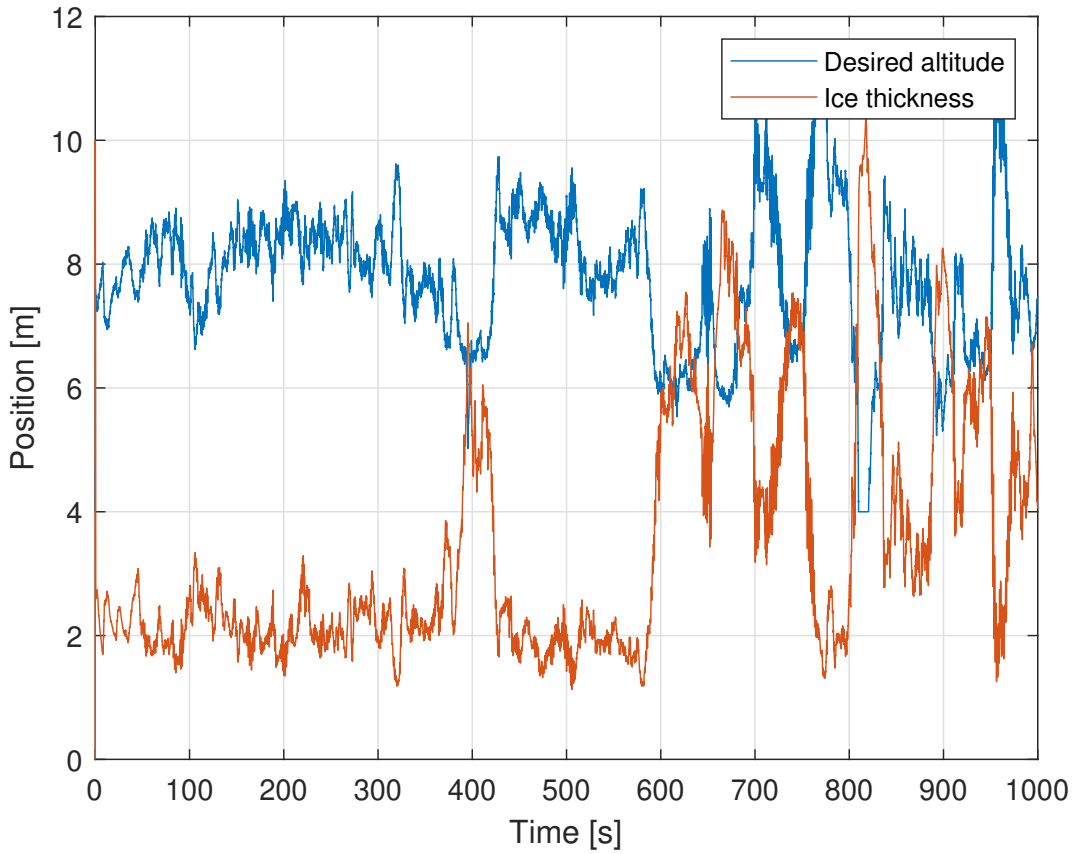


Figure 4.13: Shows the negative correlation between desired altitude and ice thickness. As expected, the thick ice makes the radiance decrease causing the UHI algorithm to push the AUV closer to the ice.

A plot of the attenuation broken down over the different light sources is seen in Figure 4.14. Here, the values represent how much of the original radiance reaches the UHI sensor after travelling through ice and water. From around 100-400 seconds the total radiance is stable around an equilibrium of L_{min} proving that the UHI algorithm produces desired reference signals for the altitude controller. However, at approximately 400, 600, 800 and 900 seconds, the radiation spikes which is expected due to the rapid changes in ice thickness (and thus altitude) at these points.

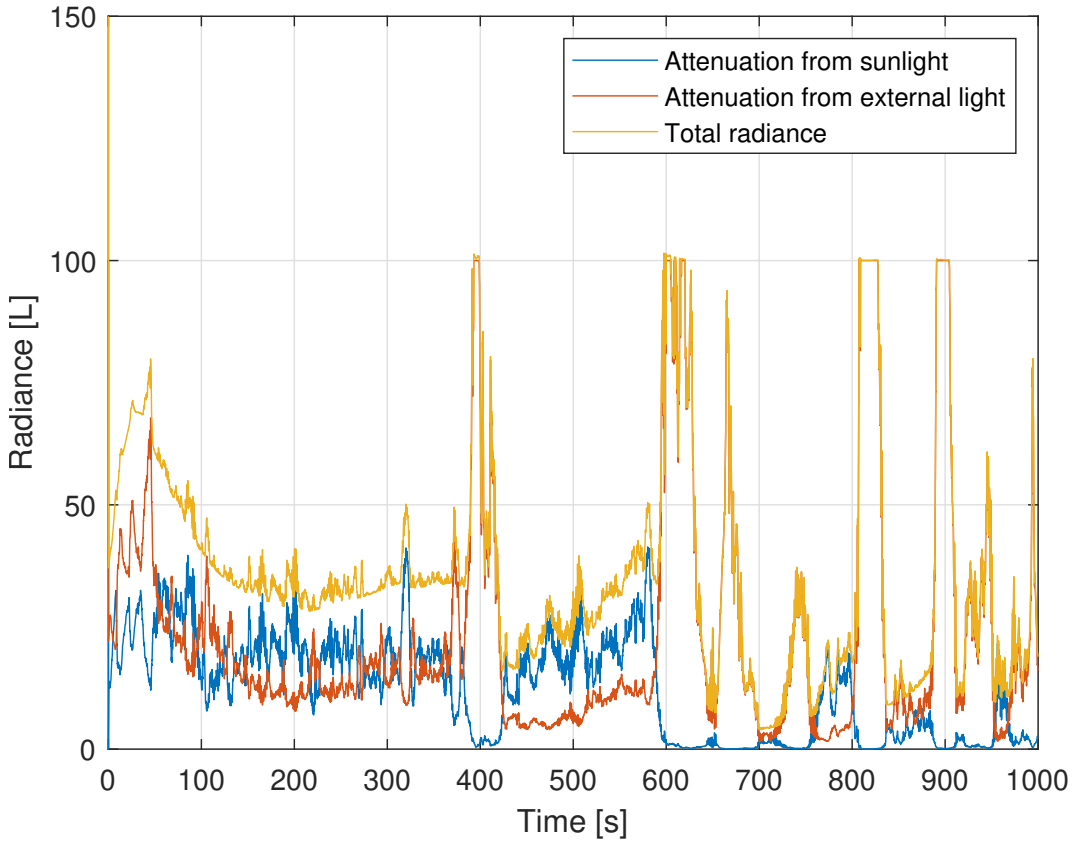


Figure 4.14: Break down of radiance as a sum of light from the sun and the external light source. Whenever the ice is stable, the radiance approaches an equilibrium around L_{min}.

Even though the simulator is provided with actuator saturation limits, it is always a good idea to check the trust allocation to see that everything looks normal and physically feasible. Figure 4.15 and Figure 4.16 show propeller shaft speed and angles of the stern and rudder fins respectively. All the values and their rate of change look reasonable.

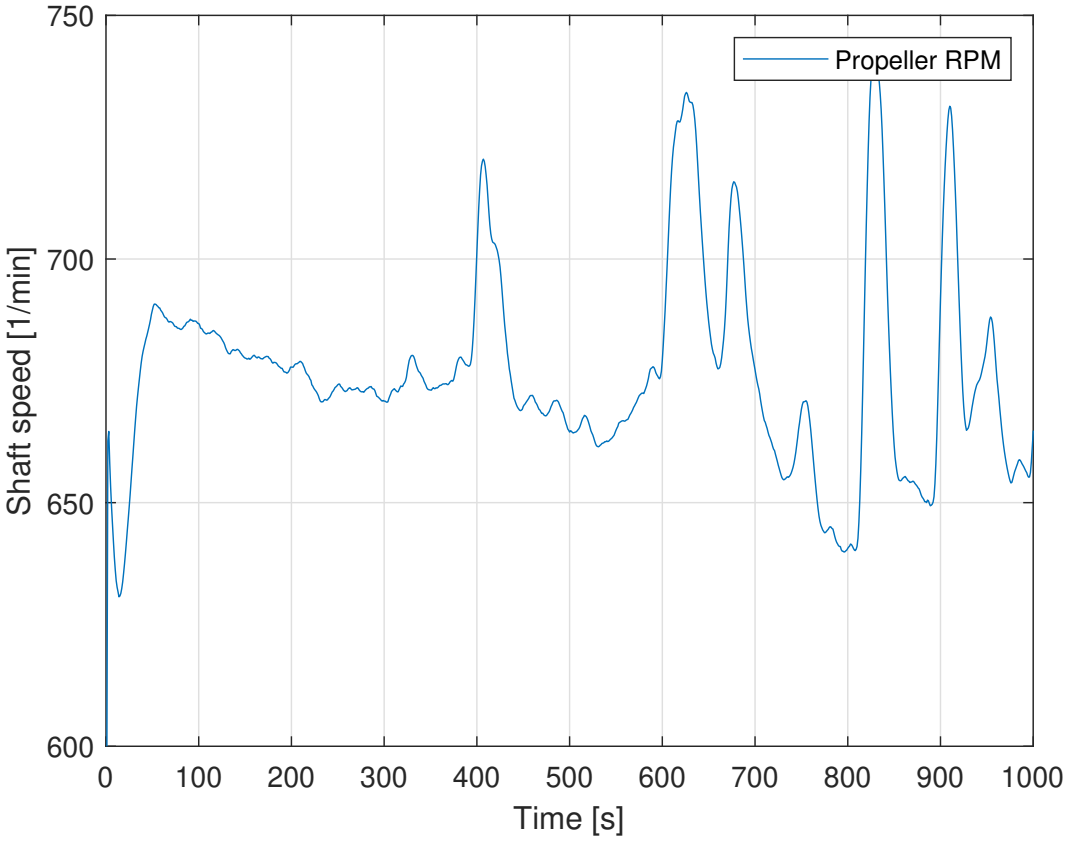


Figure 4.15: Graph showing the shaft speed of the AUV propeller. It does not alternate much, as the AUV has a constant reference speed of 1.3 m/s. The magnitude of the signal looks reasonable, about ten turns per second.

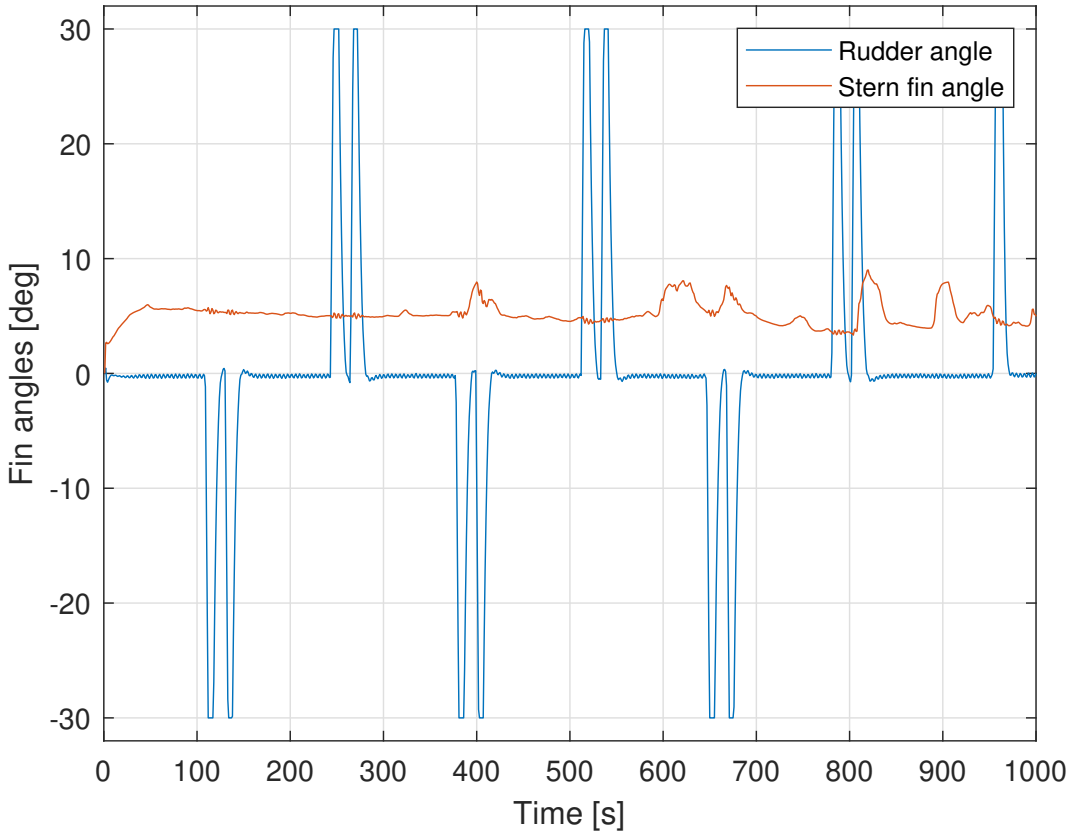


Figure 4.16: Graph showing the angles of the rudder and stern fins respectively. The rudder fin reaches its maximum design angle about every 200 seconds as part of the lawn mower pattern turning. The stern fin adjusts the pitch rate and thus the altitude.

Lastly, it is reasonable to take a look at the state errors. This says a lot about the performance of the low level control system, such as the speed, altitude and pitch controls. Figure 4.17 shows the altitude and speed error. The pitch error can be seen in Figure 4.18. Non of the values are of alarming magnitude, and it is clear that the pitch PID control is properly tuned and performs well. However, it is interesting to note how the errors of both plots increase at the same time spots as when the ice gradient is high in Figure 4.12. This is natural as the reference states change rapidly, and the dynamics cannot follow fast enough. However, it is yet another motivation of having on board sensors or other platforms such as aerial drones or satellites that may provide information about the upcoming terrain. The system should apply such knowledge of the route ahead and try to plan for such rapid changes in ice thickness.

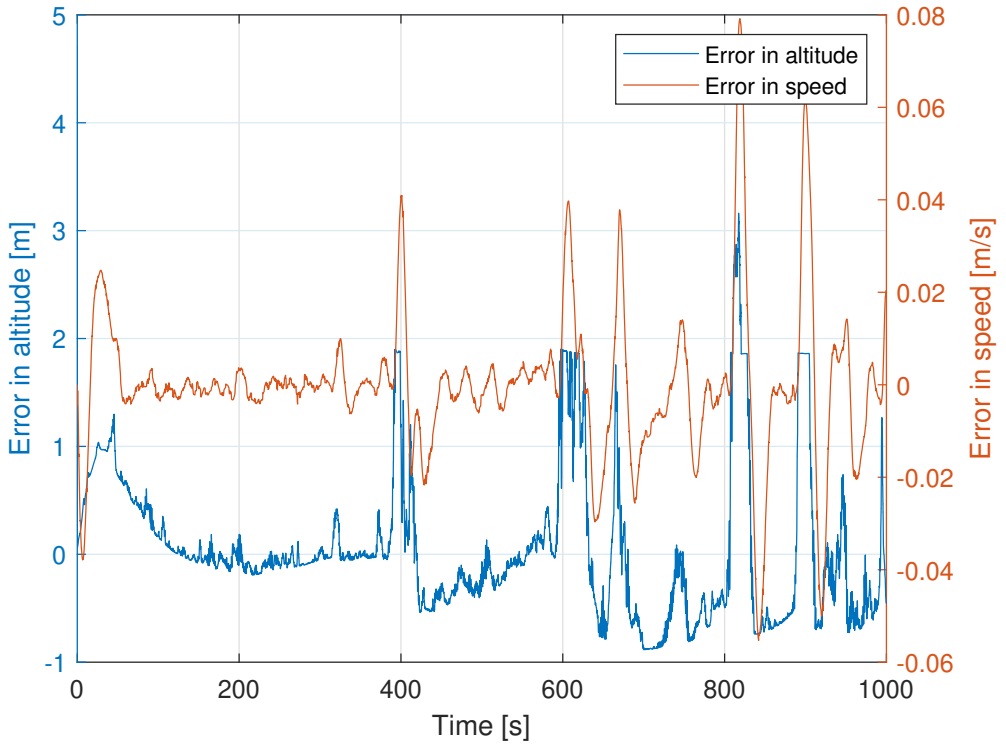


Figure 4.17: Graph showing the state error in altitude and speed.

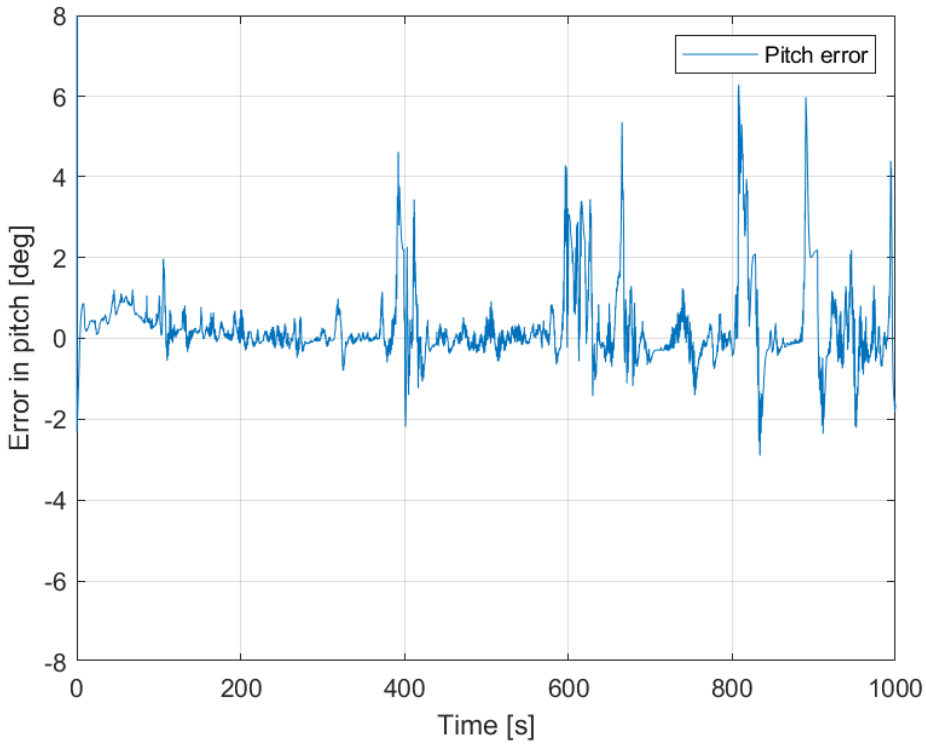


Figure 4.18: Graph showing the state error in pitch.

Variable	Changed	Symbol	Conservative	Risky
Ice attenuation coeff.	No	c_i	1	1
Sea water attenuation coeff.	No	c_w	0.2	0.2
Snow thickness	No	t_s	0.1	0.1
Backscattering	No	β	5	5
Radiance from sunlight	No	L_{sun}	300	300
Radiance from ext. light	No	L_{ext}	2000	2000
Minimum radiance threshold	Yes	L_{min}	25	50
Reaction coefficient	Yes	r	$\frac{1}{70}$	$\frac{1}{17.5}$
Minimum desired altitude	Yes	a_{des}^{min}	5	3

Table 4.2: The values used for the conservative and risk-willing case respectively, using UHI as a navigational sensor. Units as in Table 4.1.

4.4.2 Conservative vs risk-willing case

The aim of this section is to illustrate how the system may perform when being set up for an either highly conservative or a more risk-willing behaviour. From the general case, only three parameters are changed as shown in Table 4.2.

The remaining part of this section will compare some selected results from the two simulations and discuss advantages and challenges with the two different approaches.

Depth plots

Comparing the dive path of the risk-willing and conservative AUV the most obvious difference is that the conservative approach keeps its distance from the ice. It is satisfied with less radiance and does not wish to risk going closer to the ice.

Another important difference between the two is that the conservative AUV has a low reaction coefficient unlike the risk-willing one. This makes the reference altitude smoother for the conservative AUV.

Lastly, the risk-willing AUV only has a safety margin of $a_{des}^{min} = 3m$. It appears that this is not sufficient as it actually collides with the ice around 800 seconds into the simulation. Despite having a high rate of change on the reference signal, the dynamics of the AUV are to slow, and for such demanding terrain it is necessary to have either faster dynamics or more safety margin.

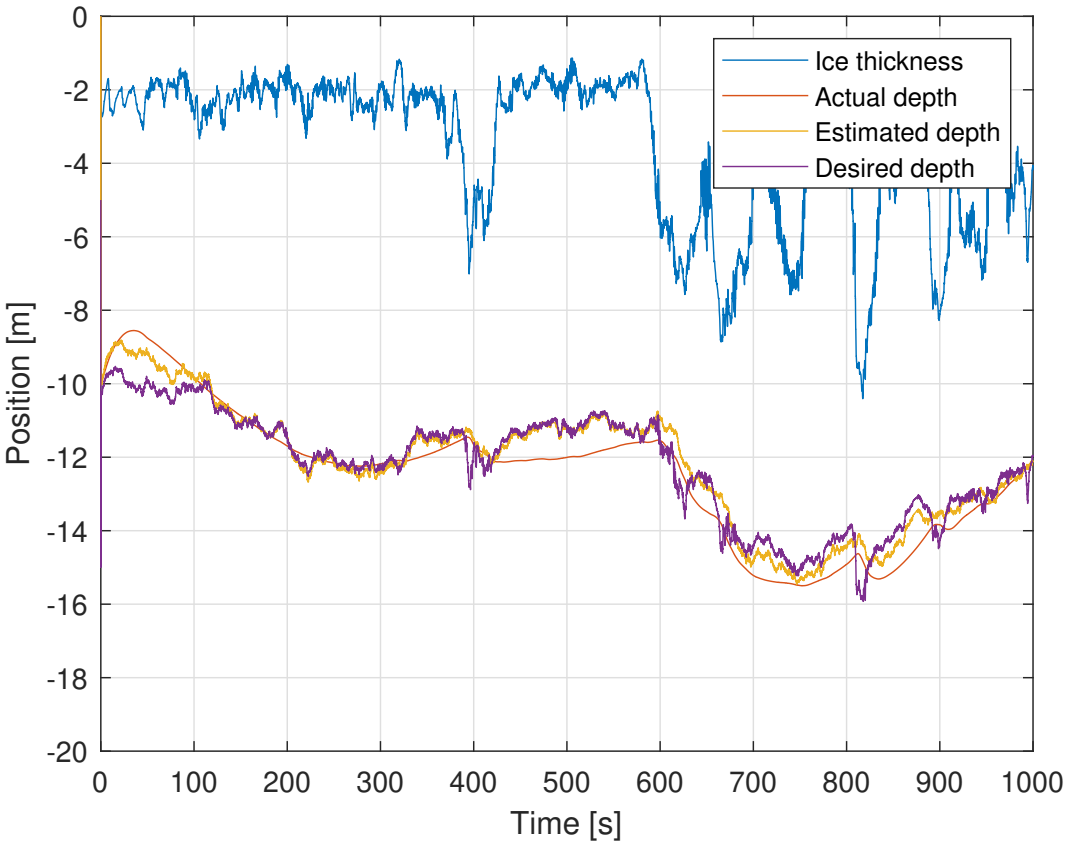


Figure 4.19: Showing the conservative AUV keeping a larger distance from the ice.

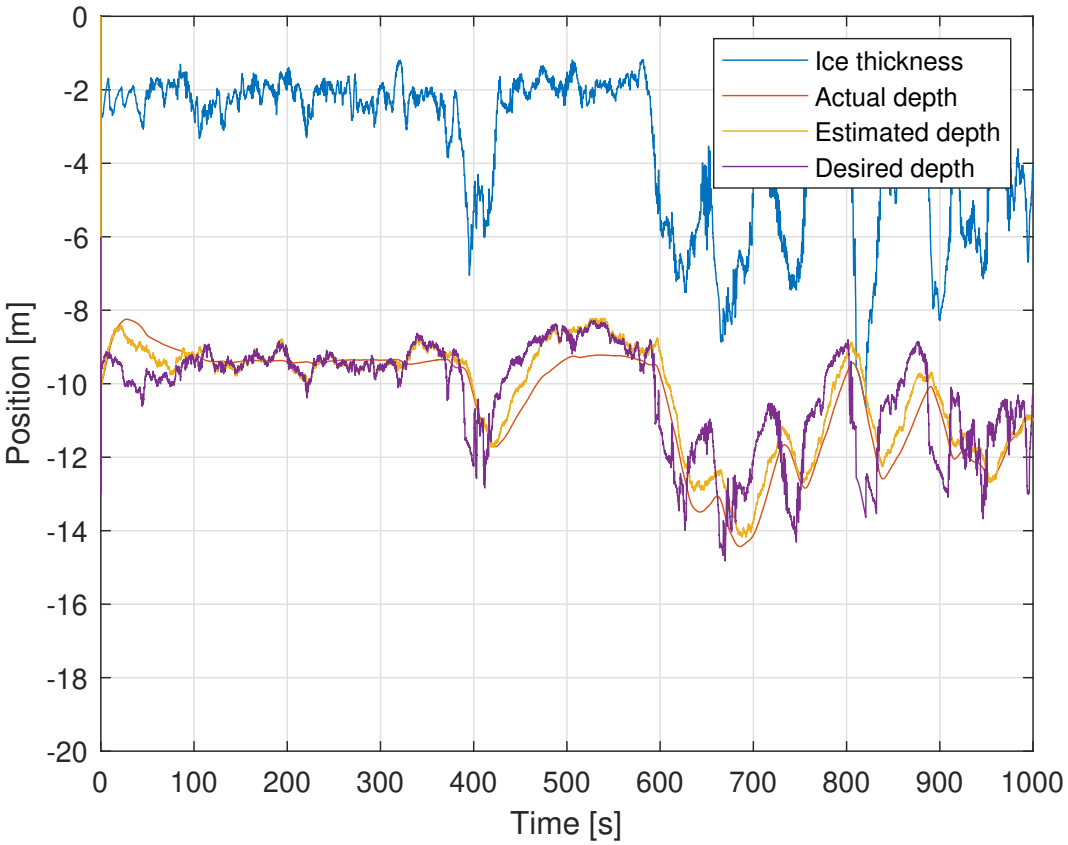


Figure 4.20: Showing the risk-willing AUV attempting to go dangerously close to the ice and increase the radiance as the thickness increases. It collides at approximately 800 seconds.

Radiance comparison

Similar to the general case, both the conservative and risk-willing AUV reach a stable radiance around L_{min} as long as the ice thickness is fairly constant e.g. from 100-350 seconds of the simulation (Figure 4.21 and Figure 4.22).

For the purpose of the UHI algorithm, the radiance has a maximum constraint at 100, which is the reason why the risk-willing AUV does not get the highest values as it is very close to the ice. However, being close to the ice the risk-willing AUV shows more radiance spikes during the simulation, and definitely got the best overall data by taking extra risk.

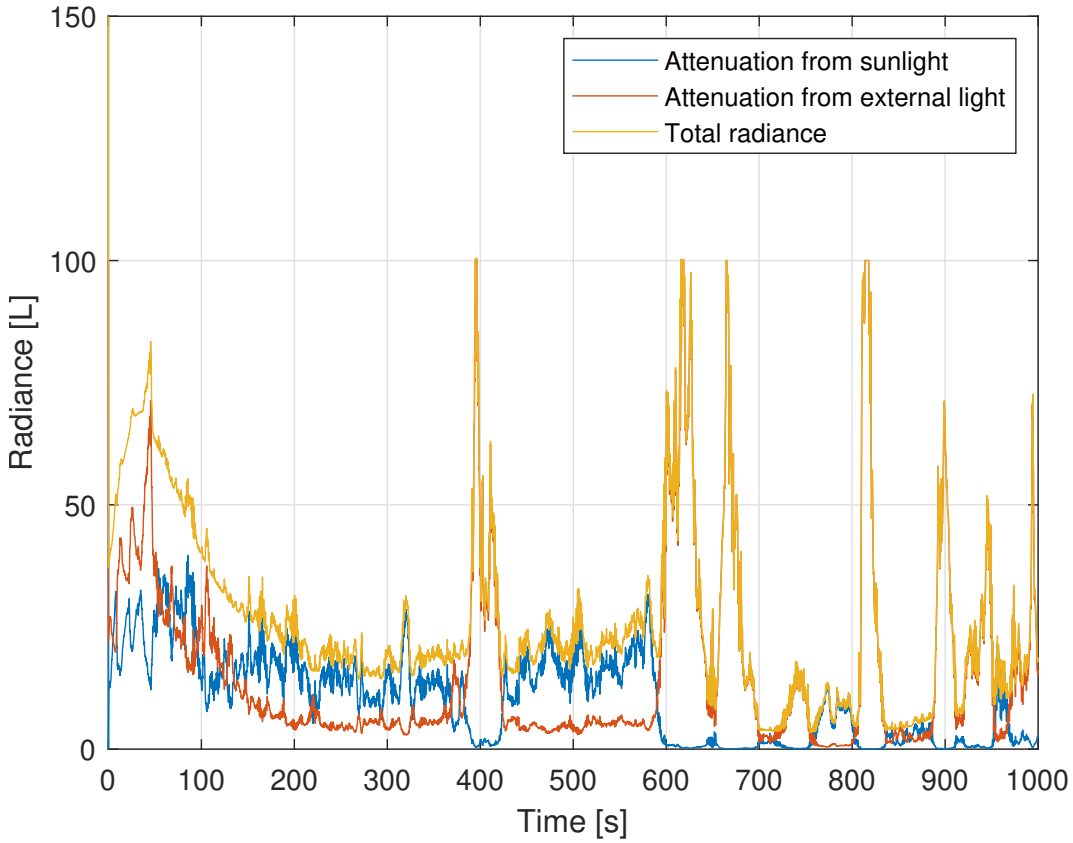


Figure 4.21: Showing the measured radiance of the conservative AUV

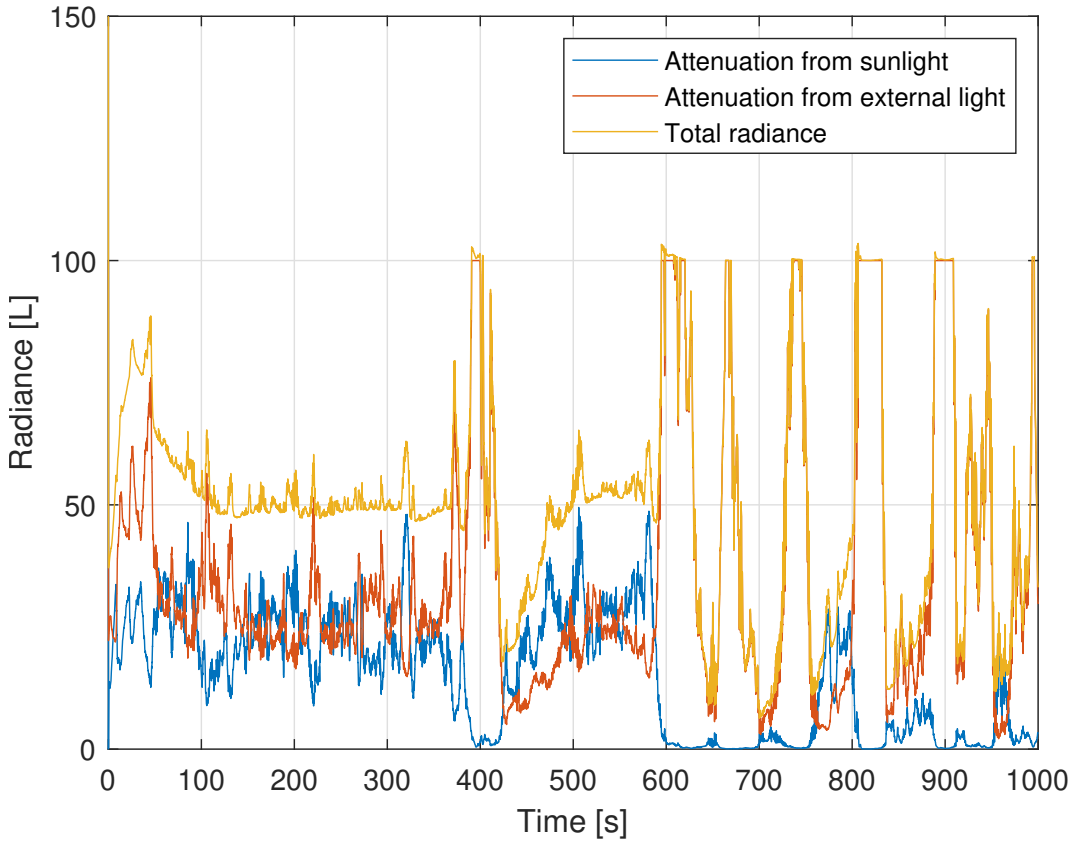


Figure 4.22: Showing the measured radiance of the risk-willing AUV

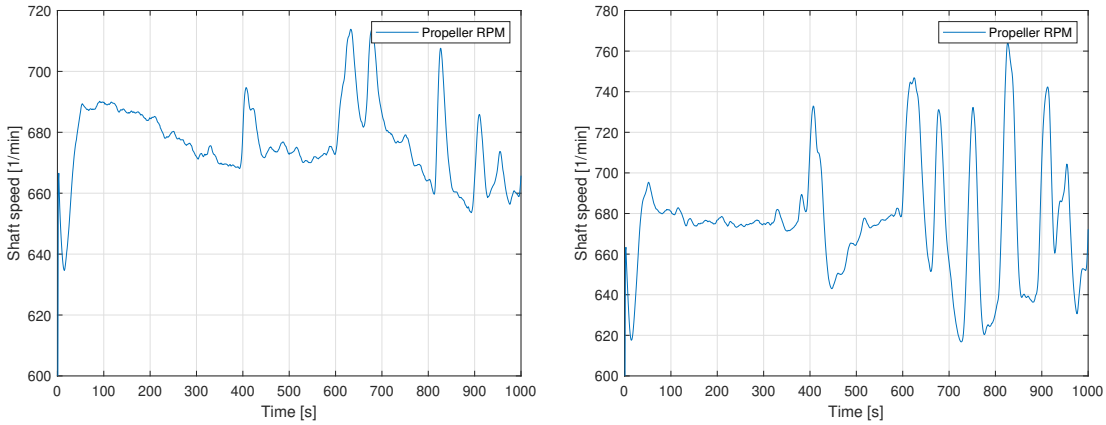
Actuator force

Another comparison that is interesting to look at is the difference in actuator force. As seen in Figure 4.23 it is clear that the more risk-willing AUV alternates the shaft speed more frequently and with a greater order of magnitude.

The same tendency is seen for the stern fins in Figure 4.24. These fins regulate the pitch and in turn the altitude, and must naturally be used more to account for the highly varying reference altitude of the risk-willing AUV.

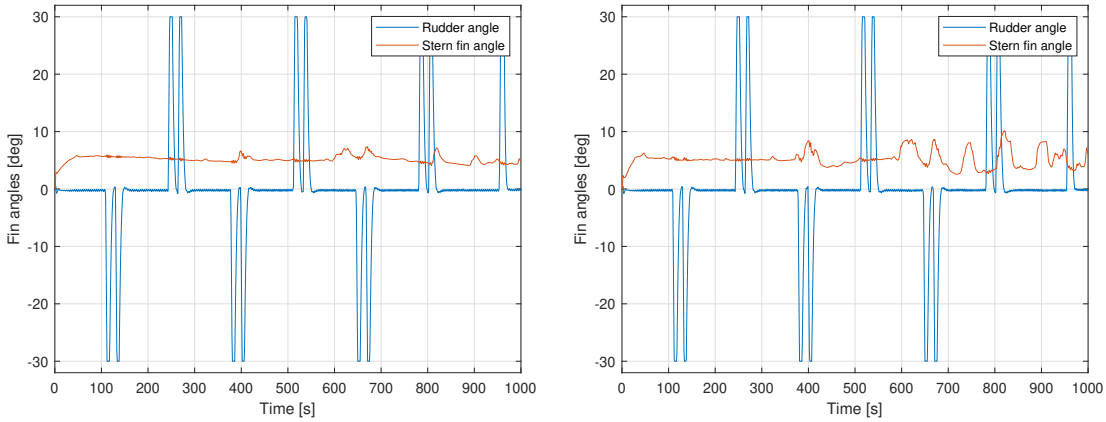
However, the rudder angle use seems to be identical in both cases. This can be explained by the imposed maximum rudder fin angle of 30 degrees which is reached in every turn of the lawn mower pattern in order to reach the next waypoint as fast as possible. In between the turn, the rudder angle is oscillating with a tiny amplitude, probably to even out some instability in the horizontal plane.

It is clear that the risk-willing AUV uses the actuators more and harder than the conservative one due to its higher reaction coefficient. Such behavior will lead to wear and tear on the actuators and is also penalising the energy system on board. This is yet another cost of wanting a more high performance system.



(a) Showing the shaft speed of the conservative AUV. **(b)** Showing the shaft speed of the risk-willing AUV.

Figure 4.23: Showing the shaft speed of the two cases. Clearly, the risk-willing AUV to the right uses the propeller more actively.



(a) Fin angles for the conservative AUV.

(b) Fin angles for the risk-willing AUV.

Figure 4.24: Showing the stern and rudder fin angles for the conservative AUV (left) and risk-willing AUV (right).

4.4.3 Discussions on UHI as a navigational sensor

To obtain the results of the last two sections, several assumptions have been made. This section will illuminate and discuss the method and arguments for these assumptions.

First of all the performance of the UHI system is dependent on many factors and parameters. The attenuation coefficients of both ice and water may vary greatly based on salinity levels, temperature gradient and other particles. Furthermore, the thickness of snow and surface terrain might greatly impact the radiance reached from the sun, and the spectrum of these fotons.

Additionally, the nature of the UHI sensor depends on the energy spectrum it receives from the surroundings. The attenuation coefficients depend on wave lengths as well as the backscatter. Some wavelengths will be more important than others depending on the OOI. Furthermore, the radiance of the omitted light from the sun and the external sensor might change. Especially the sun is suject to changes such as time of day and year, weather and other factors.

These are all factors that determine how close to the ice an UHI sensor needs to be in order to provide high quality data. Situations may occur when it is simply not worth the risk of getting these data (e.g. during an algae bloom in the polar oceans), and it is a better option to wait for better conditions.

Because of this, L_{min}, which is set by the operator, is a very important parameter. The challenge is that by setting this value to high the risk of collision will increase and the spatial coverage will decrease. However, by setting it to low the data gathered by the AUV might be worthless, not contributing to the mission objectives. A possible solution may be to use the UHI data real time in order to know when the OOI contributes to the mission goals or not. In this case, L_{min} may be increased only if the OOI is of sufficient importance. This will contribute to maximizing the utility and minimizing the risk of the operation.

Another solution might be to have an abort strategy during the mission. If the L_{min} commands the AUV to go closer to the ice than a_{des}^{min} for a longer time, it may trigger an abort that prevents the AUV from taking unnecessary risk. Such a system could also incorporate battery life, emergency collision handling and similar scenarios.

4.4.4 MPC

All the results from the former sections used a PI controller to stabilize the altitude dynamics and a PID controller to stabilize the pitch dynamics as described in Section 2.6.4. In this section however, the goal is to create a reference signal for pitch by using an MPC in conjunction with the UHI sensor. The desired altitude is yet calculated in the exact same way as for the general UHI case described in Section 4.4.1, with parameters as those in Table 4.1. The values for the MPC implementation as described in Section 3.4 may be found in Table 4.3. The time step T is set to 0.2 seconds with $N = 10$ iterations meaning that the MPC has a total time horizon of 2 seconds. In addition to the values

MPC Variable	Symbol	Value
Time step	T	0.2 $[s]$
Number of iterations	N	10 $[-]$
Pitch rate limit	q_{lim}	15 $[deg/s]$
Lower surge limit	x^l_{lim}	$-\infty$ $[m]$
Upper surge limit	x^u_{lim}	∞ $[m]$
Lower depth limit	y^l_{lim}	0 $[m]$
Upper depth limit	y^u_{lim}	15 $[m]$
Lower pitch limit	θ^l_{lim}	-15 $[deg]$
Upper pitch limit	θ^u_{lim}	15 $[deg]$
Surge penalty	Q_x	0 $[-]$
Altitude penalty	Q_a	5 $[-]$
Pitch penalty	Q_θ	0 $[-]$
Pitch rate penalty	R_q	1 $[-]$

Table 4.3: The values used for the MPC in conjunction with the UHI as a navigational sensor.

listed in Table 4.3, there are some specific parameters for the settings of the NLP solver in Casadi. These may be found in the attached files of this thesis.

The simulation results are shown in Figure 4.25. One of the major differences when using MPC in stead of a traditional PI and PID controller, is that the depth of the AUV seems more smooth. As the MPC plans ahead using a constant desired altitude, it wants to minimize actuation usage and approach the reference signal slowly. A challenge arise in this case because the system does not have sufficient information about the ice thickness ahead, and therefore the desired altitude changes faster than the MPC reference pitch allows. This effect can be minimized by increasing the altitude penalty Q_a, but will not be solved as long as the MPC does not have information to plan ahead.

Another interesting aspect of Figure 4.25 is that the otherwise smooth depth signal shows distinct jumps around 50, 400, 600, 800 and 900 seconds into the simulation. These are probably caused by the UHI algorithm changing mode between going closer to or further away from the ice. Figure 4.26 clearly shows a strong correlation between the desired altitude and the desired pitch angle.

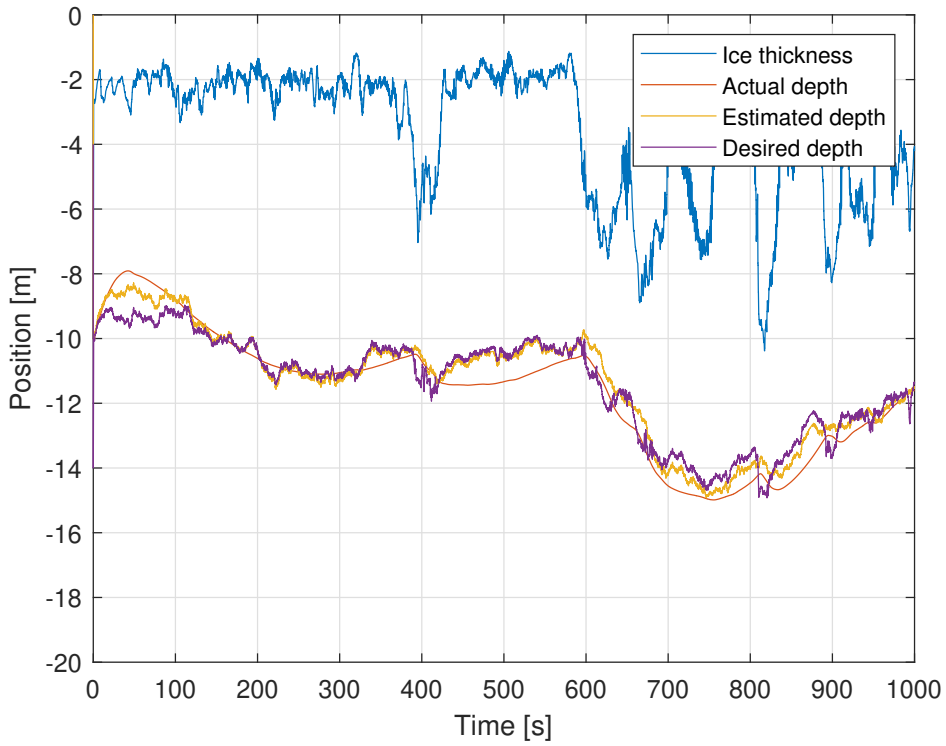


Figure 4.25

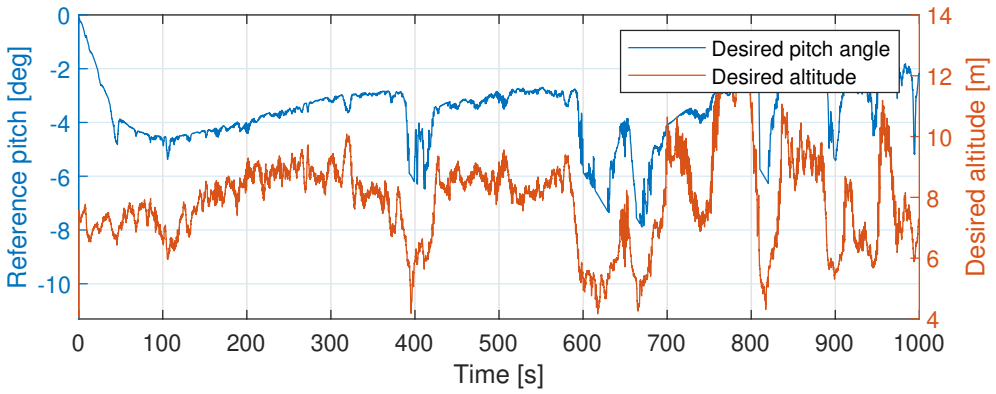


Figure 4.26

In this case the MPC converted, and obtained a feasible and optimal solution for all time steps. Note that this may not always be the case, as the NLP can be unsolvable in certain situations. Additionally, the formulation of the NLP problem is based on estimated states, and in this case the speed was assumed to be constant. As this does not reflect the true state space, there is no guarantee that the found solutions are truly optimal.

It is clear that the MPC results are not more satisfying than the traditional control methods in this case. However, the results demonstrate the power of MPC as a framework with a great potential when introducing new sensors and technology that may provide information about the AUV surroundings.

4.5 Final remarks

There are some key elements from this chapter that are worth pointing out in regard of further work and improvement suggestions.

First of all, for the system in general it is a great challenge to overcome the unbounded drift in state estimation due to the dead reckoning navigation approach. Because of the nature of the noise, which more or less averages around zero, the system is still able to perform the mission as planned with minor deviations as it will occasionally drift back towards the desired path. The simulations in this thesis had a duration of 1000 seconds (approximately 16 minutes) in which the AUV covered an area of about 250 square meters. This is a fairly long duration without navigational aids from sensors with known position such as a topside or acoustic receivers to remove the dead reckoning drift. Therefore, it is satisfying to see that the AUV is able to dive for that long without drifting outside the acceptance zone of its waypoints. However, in the present of a current, or a larger bias inherent in the sensor measurements, the whole trajectory would be shifted accordingly with no compensation from the control system. This will always be an issue with unaided dead reckoning, and must be taken into account when choosing hardware for the AUV.

It will always be of interest to aid the AUV as much as possible, especially during under ice operations that are in general more challenging than ordinary missions. If there is a strong ocean current or ice drift present at time of deployment it is possible to measure this speed and use it as an input into the AUV's navigational system so it may compensate for it despite using dead reckoning.

The former motivates the implementation of an error state (indirect) Kalman filter that may be aided by other sensor inputs such as acoustic transponders or GPS. An illustration of the error state Kalman filter design proposed by [13] is shown in Figure 4.27.

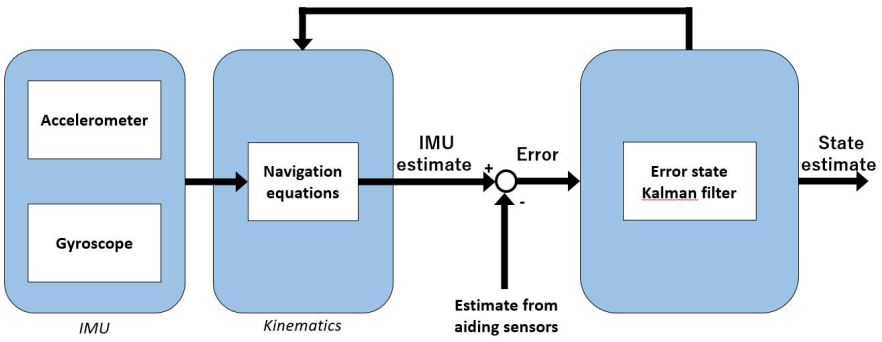


Figure 4.27: The IMU data is processed and aided by other sensors for redundancy and drift compensation.

Further, some states vary, or even oscillate to fast for the observer to converge in time. From Figure 4.6b we see that the EKF has an oscillating error in roll of about 0.2 degrees because it never truly converges before the state changes again. This might be solved by further tuning the EKF, or improving the AUV control system so that the roll oscillations are minimized.

With respect to the UHI navigation method there are some highlevel takeaways from the

results in this chapter. Firstly, as demonstrated by the ice sheet from Antarctica used in these simulations, the sea ice in polar regions is challenging terrain. The ice gradients rate of change might be high and giant submerged ice formations may appear suddenly without warning. As most AUVs depend on continuous speed in the water to maneuver, they are vulnerable to sudden changes, especially when they are unprepared. Secondly, when the ice gradient suddenly increases it takes time for the AUV to dive deeper as the pitch dynamics must stabilize before the altitude can do the same. Most AUVs have a sensor pointing straight forward, but even this might alert the emergency system to late to avoid a collision. A solution that might contribute to increased safety is to take the pitch angle into account when deciding the safety margin. In stead of simply having an a_{min}, it would add a term for the pitch angle. The higher up towards the ice the nose of the AUV is pointing, the stricter the a_{min} constraint would be.

To summarize, it is essential that the AUV has as much knowledge about the terrain ahead as possible to avoid unforeseen events. An example of this can be to have a tilted DVL sensor, or even experiment by tilting the UHI forward. More information about the terrain ahead would be an essential improvement as it can be used in autonomous planning and control methods such as MPC. Furthermore, the UHI should be equipped with a contingency handling system to deal with unforeseen events and be able to abort the mission and return to the home destination. These points motivate for a more in depth research into risk theory, optimal control and planning which is an important part of the further work of this thesis.

Chapter 5

Conclusion and further work

The work performed for this master thesis has lead to three main contributions to the Arctic AUV simulator.

- Navigation sensor models with associated observer
- Model predictive controller for altitude regulation under sea ice
- Altitude reference algorithm based on UHI as a navigational sensor

All these implementations have contributed to making the simulation environment more realistic and up to date. Hopefully, this updated version may be fruitful for future students pursuing similar research topics. An especially exciting field of research is the use of new high tech sensors such as UHI for data gathering and navigation.

This chapter will illuminate some of the most essential findings and contributions of this work and add insight into what further work might be most pressing and which direction we believe this field might take in the years to come.

5.1 Conclusion

The planned contributions in this work were implemented in the Artic AUV Simulator successfully, and it has been a rewarding process to obtain satisfying results. Despite the main ambition of hardware tests on a Blueye based hardware setup was cancelled due to the global pandemic of Covid-19, the overall progress has been highly rewarding.

The sensor models were an important contribution to the simulation environment because they incorporate such an inherent part of physical sensors and hardware. There will always be some sort of disturbance that the control system must show robustness against. The associated observer, in this case an EKF improves state estimation and adds another layer of realistic software that will be found on most up to date AUVs. Together these two modules represent a contribution by making the Arctic AUV simulator more realistic.

By providing sensor models and observer to the Arctic AUV simulator by [28], it is now possible to simulate missions and operations in polar regions more accurately. As expected, these contributions to the simulator made the performance worse compared to the original environment assuming perfect state estimation. However, the overall system with sensor models and observer managed to complete the mission and pass through the acceptance circle of its waypoints.

The model predictive control has created a solid framework for optimal control under ice, and we believe this will be an important aspect of under ice operations going forward. Despite not utilising the full potential of planning with an optimal controller in this thesis, it illustrated its potential and looks as a promising framework in conjunction with sensors that can estimate the environment ahead.

Similar to the newly designed UHI sensor it is likely that the future will introduce several other high tech sensors that might far outperform UHI both as a navigational and payload sensor. The sensor platform will also improve going forward. One of the pressing challenges is most AUVs depend on forward speed to maneuver successfully, and that makes them vulnerable to collision with the ice.

5.2 Further work

Acquiring new knowledge and completing the major objectives of this work only creates a larger perspective of what it possible to accomplish and what can be done further. The most pressing next step is to test the models and navigation methods on an actual hardware system which unfortunately was not possible due to Covid-19. What we believe is the best place to pick low hanging fruit and continue the work of this thesis is listed below.

1. Add new or modified sensors to the platform that may improve the information about the route ahead.
2. Combine look-ahead information with planning algorithms and optimal control to manage risk.
3. Successfully modify a Blueye Robotics ROV to an AUV in order to more safely test the discussed navigation techniques on real hardware.

As we have seen in Chapter 4, one important and reoccurring challenge is the lack of information about the surrounding environment which makes planning difficult. This may be accomplished by testing out new sensors or simply creating a new configuration of the existing hardware such as pointing a DVL or UHI sensor slightly forward. Similarly, it is interesting to look into new concepts such as fixed installations in the ice or multiple agents under the ice. This will be a vital step towards improved planning algorithms and help the high level mission management.

Furthermore, risk management and optimal control will be an important part of using information about the surroundings as much as possible. In challenging conditions, a

robot similar to a human being has less margin for error and should behave as optimally as possible to ensure mission success. We also believe risk management, emergency strategies and high level planning will be the pressing aspects of making under ice operations cheaper and more autonomous.

Finally, a natural next step is to test the methods and implementations presented in this thesis on the hardware setup discussed in Section 1.2.5. To our knowledge, no one has ever used an UHI sensor to aid the navigation system, not to mention under ice. It will be exciting to follow AMOS and PhD candidates Løvås and Bremnes in what they may discover and explore in the years to come.

Bibliography

[1] B. Allen, W. S. Vorus, and T. Prestero. "Propulsion system performance enhancements on REMUS AUVs". In: *Modeling, Identification and Control* 3 (2000), pp. 1869–1873.

[2] R. H. Bourke and R. P. Garrett. "Sea ice thickness distribution in the Arctic Ocean". In: *Cold Regions Science and Technology* 13.3 (1987), pp. 259–280.

[3] J. E. Bremnes. "Towards robust autonomy of underwater vehicles in Arctic operations". MA thesis. Norwegian University of Science and Technology, July 2019.

[4] E. F. Camacho and C. Bordons. *Model Predictive Control*. Springer-Verlag London Limited, 2007.

[5] J. Carlton. *Marine Propellers and Propulsion*. 2nd ed. Elsevier Science, 2007.

[6] E. Cimoli, A. Lucieer, K. M. Meiners, L. C. Lund-Hansen, F. Kennedy, A. Martin, A. McMinn, and V. Lucieer. "Towards improved estimates of sea-ice algal biomass: experimental assessment of hyperspectral imaging cameras for under-ice studies". In: *Annals of Glaciology* 58 (2017), pp. 68–77.

[7] National Research Council. *Autonomous Vehicles in Support of Naval Operations*. Washington, DC: The National Academies Press, 2005.

[8] F. Dukan. "ROV motion control systems". PhD thesis. Norwegian University of Science and Technology, June 2014.

[9] C. C. Finlay, S. Maus, C. D. Beggan, T. N. Bondar, A. Chambodut, T. A. Chernova, A. Chulliat, V. P. Golovkov, B. Hamilton, M. Hamoudi, R. Holme, G. Hulot, W. Kuang, B. Langlais, V. Lesur, F. J. Lowes, H. Lühr, S. Macmillan, M. Mandea, S. McLean, C. Manoj, M. Menvielle, I. Michaelis, N. Olsen, J. Rauberg, M. Rother, T. J. Sabaka, A. Tangborn, L. Tøffner-Clausen, E. Thébault, A. W. P. Thomson, I. Wardinski, Z. Wei, and T. I. Zvereva. "International Geomagnetic Reference Field: the eleventh generation". In: *Geophysical Journal International*. Vol. 183. 3. Dec. 2010, pp. 1216–1230.

[10] T. I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, 2011.

[11] T. I. Fossen. "Nonlinear Passive Control and Observer Design for Ships". In: *Modeling, Identification and Control* 21 (2000), pp. 129–184.

[12] T. I. Fossen and Å. Grøvlen. "Nonlinear Output Feedback Control of Dynamically Positioned Ships Using Vectorial Observer Backstepping". In: *IEEE Transactions on Control Systems Technology* 6 (1998), pp. 121–128.

[13] K. Gade. *Introduction to Inertial Navigation and Kalman Filtering, Tutorial for IAIN World Congress, Stockholm, Sweden.* 2009. URL: https://www.navlab.net/Publications/Introduction_to_Inertial_Navigation_and_Kalman_Filtering.pdf (visited on 05/30/2020).

[14] K. Gade. "The Seven Ways to Find Heading". In: *Journal of Navigation* 69.5 (Apr. 2016), pp. 955–970.

[15] K. Gade and B. Jalving. "An aided navigation post processing filter for detailed seabed mapping UUVs". In: *Proceedings of the 1998 Workshop on Autonomous Underwater Vehicles (Cat. No.98CH36290).* Aug. 1998, pp. 19–25.

[16] A. Gelb, J. F. Kasper Jr., R. A. Nash Jr., C. F. Price, and A. A. Sutherland Jr. "Applied Optimal Estimation". In: *MIT Press* (1988).

[17] A. J. Healey, E. P. An, and D. B. Marco. "Online compensation of heading sensor bias for low cost AUVs". In: *Proceedings of the 1998 Workshop on Autonomous Underwater Vehicles (Cat. No.98CH36290).* Aug. 1998, pp. 35–42.

[18] H. M. Huang. *Autonomy Levels For Unmanned Systems.* 2016. URL: https://www.nist.gov/el/intelligent-systems-division-73500/cognition-and-collaboration-systems/autonomy-levels-unmanned (visited on 05/30/2020).

[19] L. I. Iozan, M. Kirkko-Jaakkola, J. Collin, J. Takala, and C. Rusu. "North finding system using a MEMS gyroscope". In: *Proceedings of European Navigation Conference on Global Navigation Satellite Systems, Braunschweig, Germany.* Oct. 2010, pp. 19–21.

[20] J. G. Guo and J. Zhou. "Altitude Control System of Autonomous Airship Based on Fuzzy Logic". In: *International Symposium on Systems and Control in Aerospace and Astronautics.* Vol. 2. 2008, pp. 1–5.

[21] B. R. Johnson, E. Cabuz, H. B. French, and R. Supino. "Development of a MEMS gyroscope for northfinding applications". In: *IEEE/ION Position, Location and Navigation Symposium.* May 2010, pp. 168–170.

[22] C. Katz. *Tiny pieces of plastic found in Arctic snow.* 2019. URL: https://www.nationalgeographic.com/environment/2019/08/microplastics-found-in-arctic-snow/ (visited on 02/19/2020).

[23] A. Kukulya, A. Plueddemann, T. Austin, R. Stokey, M. Purcell, B. Allen, R. Littlefield, L. Freitag, P. Koski, E. Gallimore, J. Kemp, K. Newhall, and J. Pietro. "Under-ice operations with a REMUS-100 AUV in the Arctic". In: *IEEE/OES Autonomous Underwater Vehicles.* 2010, pp. 1–8.

[24] H. Løvås, M. Ludvigsen, and A. J. Sørensen. "Framework for Combining Multiple Light Underwater Vehicles into Super Underwater Vehicle". Submitted, awaiting publication. 2020.

[25] D. R. Lyzenga, N. P. Malinas, and F. J. Tanis. "Multispectral bathymetry using a simple physically based algorithm". In: *IEEE Transactions on Geoscience and Remote Sensing* 44.8 (2006), pp. 2251–2259.

[26] A. Medwin and C. S. Clay. *Fundamentals of Acoustical Oceanography.* Academic Press, 1998.

[27] K. Morsink. *With Every Breath You Take, Thank the Ocean.* 2017. URL: https://ocean.si.edu/ocean-life/plankton/every-breath-you-take-thank-ocean (visited on 04/30/2020).

[28] P. Norgren. “Autonomous underwater vehicles in Arctic marine operations : Arctic marine research and ice monitoring”. PhD thesis. Norwegian University of Science and Technology, 2018.

[29] T. Prestero. “Development of a six-degree of freedom simulation model for the REMUS autonomous underwater vehicle”. In: *MTS/IEEE Oceans 2001. An Ocean Odyssey. Conference Proceedings (IEEE Cat. No.01CH37295)* 1 (2001), pp. 450–455.

[30] D. Qi, J. Feng, and J. Yang. “Longitudinal Motion Control of AUV Based on Fuzzy Sliding Mode Method”. In: *Journal of Control Science and Engineering* 2016 (Feb. 2016), pp. 1–7.

[31] B.M. Renkoski. “The Effect of Carouseling on MEMS IMU Performance for Gyrocompassing Applications”. MA thesis. Massachusetts Institute of Technology, Jan. 2008.

[32] H. Rice, S. Kelmenson, and L. Mendelsohn. “Geophysical navigation technologies and applications”. In: *PLANS 2004. Position Location and Navigation Symposium (IEEE Cat. No.04CH37556)*. 2004, pp. 618–624.

[33] H. Singh, T. Maksym, J. Wilkinson, and G. Williams. “Inexpensive, small AUVs for studying ice-covered polar environments”. In: *Science Robotics* (2017).

[34] A. J. Sørensen. “Structural issues in the design and operation of marine control systems”. In: *Annual Reviews in Control* 1 (2005), pp. 125–149.

[35] A. J. Sørensen and M. Ludvigsen. “Towards Integrated Autonomous Underwater Operations”. In: *IFAC-PapersOnLine* 48.2 (2015), pp. 107–118.

[36] A. J. Sørensen, M. Ludvigsen, P. Norgren, Ø. Ødegård, and F. Cottier. “Sensor-Carrying Platforms”. In: *POLAR NIGHT Marine Ecology*. Ed. by J. Berge, G. Johnsen, and J. H. Cohen. Springer, Germany, 2020. Chap. 9, pp. 241–275.

[37] T. Y. Loh, M. P. Brito, N. Bose, J. Xu, and K. Tenekedjiev. “A Fuzzy-Based Risk Assessment Framework for Autonomous Underwater Vehicle Under-Ice Missions”. In: *Risk Analysis, and international journal*. Vol. 39. 12. 2019, pp. 2744–2765.

[38] J. Tegdan, S. Ekehaug, I. M. Hansen, L. M. S. Aas, K. J. Steen, P. Pettersen, F. Beuchel, and L. Camus. “Underwater hyperspectral imaging for environmental mapping and monitoring of seabed habitats”. In: *OCEANS 2015 - Genova*. 2015, pp. 1–6.

[39] I. B. Utne, A. J. Sørensen, and I. Schjølberg. “Risk Management of Autonomous Marine Systems and Operations”. In: *International Conference on Offshore Mechanics and Arctic Engineering*. Vol. 3B: Structures, Safety and Reliability. June 2017.

[40] Z. Volent, H. Dierssen, R. Pettersen, M. Ardelan, F. Søreide, P. Fearns, M. Ludvigsen, M. Moline, and G. Johnsen. “Underwater hyperspectral imagery to create biogeochemical maps of seafloor properties”. In: *Subsea Optics and Imaging* (Dec. 2013), pp. 508–535.

[41] G. Williams, T. Maksym, J. Wilkinson, C. Kunz, C. Murphy, P. Kimball, and H. Singh. “Thick and deformed Antarctic sea ice mapped with autonomous underwater vehicles.” In: *Nature Geoscience* 8 (2015), pp. 61–67.

[42] T. Zhang, Z. Wang, Y. Li, and T. Jinwu. “A Passive Acoustic Positioning Algorithm Based on Virtual Long Baseline Matrix Window”. In: *Journal of Navigation* 72 (Aug. 2018), pp. 1–14.

Appendices

Appendix A

Parameters of the AUV model

This section describes the simulator parameters with courtesy of [3]. The values for the 6 DOF equation of motion (2.32) implemented in the simulator are taken from [29]. The rigid-body, added mass and damping matrices for the REMUS 100 AUV are as follows:

$$\boldsymbol{C}_{RB} = \begin{bmatrix} 30.479 & 0 & 0 & 0 & 0.597 & 0 \\ 0 & 30.479 & 0 & -0.597 & 0 & 0 \\ 0 & 0 & 30.479 & 0 & 0 & 0 \\ 0 & -0.597 & 0 & 0.189 & 0 & 0 \\ 0.597 & 0 & 0 & 0 & 3.462 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3.45 \end{bmatrix} \tag{A.1}$$

$$\boldsymbol{M}_A = \begin{bmatrix} 0.93 & 0 & 0 & 0 & 0 & 0 \\ 0 & 35.5 & 0 & 0 & 0 & -1.93 \\ 0 & 0 & 35.5 & 0 & 1.93 & 0 \\ 0 & 0 & 0 & 0.07 & 0 & 0 \\ 0 & 0 & 1.93 & 0 & 4.88 & 0 \\ 0 & -1.93 & 0 & 0 & 0 & 4.88 \end{bmatrix} \tag{A.2}$$

$$\boldsymbol{D}(\boldsymbol{\nu}_r) = \begin{bmatrix} 1.62|u_r| & 0 & 0 & 0 & 0 & 0 \\ 0 & 1310|v_r| & 0 & 0 & 0 & -0.632|r| \\ 0 & 0 & 1310|w_r| & 0 & 0.632|q| & 0 \\ 0 & 0 & 0 & 0.13|p| & 0 & 0 \\ 0 & 0 & -3.18|w_r| & 0 & 188|q| & 0 \\ 0 & 3.18|v_r| & 0 & 0 & 0 & 188|r| \end{bmatrix} \tag{A.3}$$

The rigid-body and added mass centripetal and Coriolis matrices, denoted as $\boldsymbol{C}(\boldsymbol{\nu}_r) = \boldsymbol{C}_{RB}(\boldsymbol{\nu}) + \boldsymbol{C}_A(\boldsymbol{\nu}_r)$ are given by:

$$\boldsymbol{C}_{RB} = \begin{bmatrix} \boldsymbol{0}_{3\times 3} & -m\boldsymbol{S}(\boldsymbol{v}^b_{auv}) - m\boldsymbol{S}(\boldsymbol{\omega}^b_{auv})S(\boldsymbol{r}^b_g) \\ -m\boldsymbol{S}(\boldsymbol{v}^b_{auv}) + m\boldsymbol{S}(\boldsymbol{r}^b_g)S(\boldsymbol{\omega}^b_{auv}) & -\boldsymbol{S}(\boldsymbol{I}_b\boldsymbol{\omega}^b_{auv}) \end{bmatrix} \tag{A.4}$$

$$\boldsymbol{C}_A = \begin{bmatrix} \mathbf{0}_{3\times3} & -\boldsymbol{S}(\boldsymbol{A}_{11}v_{auv}^2 + \boldsymbol{A}_{12}\omega_{auv}^2) \\ -\boldsymbol{S}(_{11}v_{auv}^2 + \boldsymbol{A}_{12}\omega_{auv}^2) & -\boldsymbol{S}(\boldsymbol{A}_{21}v_{auv}^2 + \boldsymbol{A}_{22}\omega_{auv}^2) \end{bmatrix} \tag{A.5}$$

where $\boldsymbol{S}(\boldsymbol{r}_g^b) = -\boldsymbol{S}^T(\boldsymbol{r}_g^b) \in \mathbb{R}^{3\times3}$ is a skew-symmetric matrix, $\boldsymbol{r}_g^b$ represents the location of the AUV's center of gravity with respect fo the CO, I_b represents the inertia tensor and $\boldsymbol{A}_{ij} \in \mathrm{R}^{3\times3}$ given by

$$\boldsymbol{M}_A = \begin{bmatrix} \boldsymbol{A}_{11} & \boldsymbol{A}_{12} \\ \boldsymbol{A}_{21} & \boldsymbol{A}_{22} \end{bmatrix} \tag{A.6}$$

The propeller coefficients are taken from [1]. The remaining parameters for the REMUS 100 AUV are given by Table A.1.

Table A.1: REMUS 100 AUV parameters

Description	**Symbol**	**Value**
CG lever arm w.r.t. CO	r_g^b	$[0, 0, 0.0196]$ $[m]$
CB lever arm w.r.t. CO	r_b^b	$[0, 0, 0]$ $[m]$
Weight of vehicle	W	$299[N]$
Vehicle buoyancy	B	$306[N]$
Density of sea water	ρ	$1025[kg/m^3]$
Fin lift coefficient	$c_{L,\alpha}$	$3.12[-]$
Fin surface area	S_{fin}	$6.65 \times 10^{-3}[m^2]$
Fin placement w.r.t. CO	x_{fin}	$-0.638[m]$
Thrust coefficient	K_T	$2.5075[-]$
Torque coefficient	K_Q	$0.3203[-]$
Propeller diameter	D_{prop}	$0.1397[m]$

Appendix B

Waypoints for AUV guidance

Index	$\boldsymbol{N}$ $[m]$	$\boldsymbol{E}$ $[m]$
1	225	250
2	195	250
3	195	40
4	165	40
5	165	250
6	135	250
7	135	40
8	105	40
9	105	250
10	75	250
11	75	40
12	45	40
13	45	250

Table B.1: Waypoint coordinates used for the results shown in Chapter 4.

Appendix C

Control and guidance gains for the Arctic AUV Simulator

Table C.1 shows the control gains used for the Remus AUV in the Arctic AUV Simulator. Note that the pitch controller is a subcontroller of the altitude controller.

Controller	**P gain**	**I gain**	**D gain**
Altitude controller	0.05	0.001	–
Pitch controller	1.5	0.15	1.9
Speed controller	300	100	1
Heading controller	1.6	0.2	1.5

Table C.1: Shows the PID controller gains used in Chapter 4.

Table C.2 shows the parameters used for the LOS guidance system.

Parameter	**Symbol**	**Value**
Look-ahead distance	Δ	$20[m]$
Radiance of acceptance	R_{acc}	$10[m]$

Table C.2: Shows the parameters used for the guidance law described in Section 2.6.5.

Thomas Erling Lone

NTNU
Norwegian University of
Science and Technology