Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

Thea Kristine Bergh

ENC-based Collision and Grounding Avoidance System for a Green-Energy Autonomous Surface Vehicle

Master's thesis in Cybernetics and Robotics
Supervisor: Tor Arne Johansen
Co-supervisor: Alberto Dallolio
June 2021

NTNU
Norwegian University of
Science and Technology

Thea Kristine Bergh

ENC-based Collision and Grounding Avoidance System for a Green-Energy Autonomous Surface Vehicle

Master's thesis in Cybernetics and Robotics
Supervisor: Tor Arne Johansen
Co-supervisor: Alberto Dallolio
June 2021

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

Abstract

Autonomous surface vehicles (ASVs) have an important role in the development of ocean observing systems (OOS) because they possess the ability to perform sustainable and continuous open ocean exploration and observation, even in harsh conditions in remote areas. These observations give knowledge about how the ocean and its dynamic processes are changing over time and how they are affected by climate changes. The AutoNaut is a wave-propelled green-energy ASV that is suited for these kinds of missions.

In order for an ASV to be able to operate autonomously on the ocean and along the coast, a robust collision and grounding avoidance system is crucial. The collision avoidance system needs to be compliant with the International Regulations for Preventing Collisions at Sea (COLREGS), and the anti-grounding system should be able to use data from electronic navigational charts (ENCs).

This thesis proposes an ENC-based anti-grounding system and integrates it into the existing simulation-based model predictive control (SB-MPC) collision avoidance system (CAS) developed by Johansen *et al.* [1] and implemented by Hagen [2], [3]. The system is adapted and implemented on the AutoNaut. The collision and grounding avoidance system is COLREGS compliant and can take environmental factors into account when evaluating which action to take in a given situation.

Several simulations are performed in order to test the system in real-life scenarios. The scenarios include static and dynamic obstacles and are done with and without environmental factors included. The results look promising. The anti-grounding system ensures that the AutoNaut avoids going too close to land. When there are multiple obstacles, the system is able to take both static and dynamic obstacles into consideration, at the same time as it accounts for the environmental conditions and chooses the least hazardous action in the situation. Sometimes, this means not complying with COLREGS in order to avoid going too close to land.

Sammendrag

Autonome overflatefartøy (ASV-er) kan spille en viktig rolle i utviklingen av systemer som utforsker og observerer havet på en bærekraftig og kontinuerlig måte, selv i avsidesliggende områder og under tøffe forhold. Disse observasjonene kan gi ny og viktig kunnskap om hvordan havet og dets dynamiske prosesser endrer seg over tid, og hvordan de påvirkes av klimaendringene. AutoNauten er en bølgedrevet ASV som er godt egnet til denne typen oppdrag.

For at en ASV skal kunne operere autonomt ute på havet og langs kysten, er det helt nødvendig at den har et robust system som sørger for at den unngår kollisjoner og grunnstøtinger. Kollisjonsunngåelsessystemet må overholde de internasjonale forskriftene for forebygging av kollisjon på sjøen (COLREGS), og anti-grunnstøtingssystemet skal kunne bruke data fra de elektroniske sjøkartene (ENCs).

Denne masteroppgaven presenterer et ENC-basert anti-grunnstøtingsystem og integrerer det i det eksisterende kollisjonsunngåelsessystemet (CAS) som er et simulasjonsbasert modell-prediktivt reguleringssystem (SB-MPC) utviklet av Johansen *et al.* [1] og implementert av Hagen *et al.* [2], [3]. Systemet er tilpasset til og implementert i AutoNauten. Systemet overholder COLREGS og kan ta hensyn til miljøfaktorer når det vurderer hva som er den beste manøveren for ASVen i en gitt situasjon.

Ulike simuleringer er utført for å teste systemet i virkelighetsnære scenarioer. Scenarioene inkluderer statiske og dynamiske hindringer, og tester er gjort både med og uten å ta hensyn til miljøfaktorer. Resultatene ser lovende ut. Anti-grunnstøtingssystemet forsikrer at AutoNauten unngår å bevege seg for nær land. Når det er flere ulike hindringer, tar systemet hensyn til både de statiske og de dynamiske hindringene, samtidig som den tar i betraktning de nåværende miljøfaktorene, og velger å utføre manøveren som fører med seg minst fare. Noen ganger innebærer dette at systemet ikke overholder COLREGS, men i stedet passer på at AutoNauten ikke går for nær land.

Preface

This thesis concludes my Master of Science degree in Cybernetics and Robotics at the Norwegian University of Science and Technology, NTNU. The thesis has been carried out during the spring of 2021 and has been supervised by Tor Arne Johansen and Alberto Dallolio.

The work presented in this thesis is a continuation of the work performed in the specialization project during the autumn of 2020. The specialization project focuses on validation and study of the existing collision avoidance system, while the master's thesis develops an anti-grounding system that is integrated into this collision avoidance system. As the report of the specialization project has not been published, parts from the report that describe the system have been adapted and used in this report. The sections of the thesis that contain material from the specialization project report are: Section 1.1.1, parts of Chapter 2, Chapter 3 and the description of the existing collision avoidance system in Chapter 5.

I would like to thank my supervisor Tor Arne Johansen for an exciting project and for all guidance and good advice throughout the process. I would also like to thank my co-supervisor Alberto Dallolio for answering all my questions, giving me the necessary insight and knowledge, and for good discussions on the development of the system. Finally, I would like to thank my family and friends for their endless support, and especially my dear Kevin for always being there to motivate and support me.

Thea Kristine Bergh
Trondheim, June 2021

Contents

Figures

Tables

Abbreviations

AIS	Automatic Identification System
ASV	Autonomous Surface Vehicle
CAS	Collision Avoidance System
COLAV	Collision Avoidance
COLREGS	Convention on the International Regulations for Preventing Collisions at Sea
DOF	Degrees of Freedom
ENC	Electronic Navigational Chart
LOS	Line-of-Sight
MPC	Model Predictive Control
OOS	Ocean Observing System
SB-MPC	Simulation-Based Model Predictive Control
WGS84	World Geodetic System 1984

Chapter 1

Introduction

1.1 Motivation

1.1.1 The AutoNaut project

The development of autonomous vehicles has progressed rapidly in the past years due to advancements in research, increasing computer power, cheaper components, and newfound potential applications. An abundance of new and exciting opportunities is opening with this new technology. Fully autonomous cars are reportedly being tested in public in several locations [4], autonomous underwater vehicles (AUVs) can do underwater inspections and maintenance for subsea installations, and autonomous shipping can lead to safer shipping with reduced costs and emissions [5]. These are only some examples of possible applications.

With climate change and global warming, the need to monitor the nature and collect data about changes in the environment is greater than ever. Ocean science and the study of climate change occurring in the ocean are key to understand the evolution of the planet. The ocean plays an important part in understanding, managing, and possibly solving the many challenges related to climate change. Observing the dynamics of the ocean over time is crucial to obtain this knowledge. Global initiatives are working to establish Ocean Observing Systems (OOS) capable of providing sustained observations of the ocean. The Global Ocean Observing System, executed by the Intergovernmental Oceanographic Commission (IOC) of UNESCO, coordinates the work internationally. There are also several national programs around the world, i.e., the Integrated Ocean Observing System (IOOS) operated by the US government [6].

Distant areas with tough weather conditions and challenging accessibility, like the Arctic areas, are often the most interesting and important to observe but also the most vulnerable. Performing human operations in such areas could be demanding and comes with a high risk and a high cost.

The traditional ship-based methods used for ocean surveillance are not optimal. They cannot be performed continuously over a long time period as they depend on an operational crew and researchers, and they lead to significant emis-

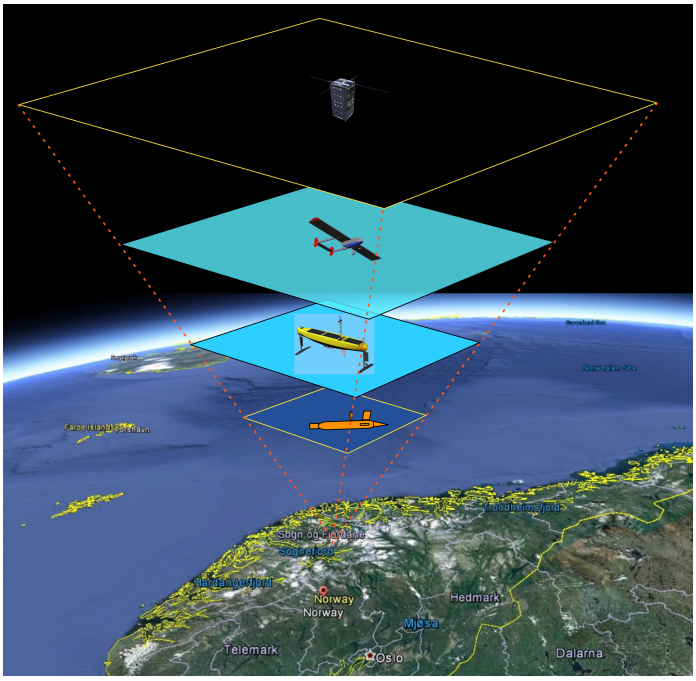


Figure 1.1: Vision for how coordinated observations of the ocean can be achieved [7].

sions of CO_2. Also, these missions disturb the nature and wildlife with motor noise and large vessels, and they are not found to be very cost-effective. Therefore, this current solution is limited in space and time, and is not optimal to perform sustainable observations of the ocean [7].

Autonomous robotic systems have the potential to solve many of these challenges. The latest developments in robotics and artificial intelligence for creating robust autonomous vehicles have made it realistic and achievable for autonomous vessels to be used for long-duration operations in the ocean, also in harsh conditions. They can provide continuous real-time data for scientists over long periods, giving valuable insights into the ocean dynamics. Coordinated observation systems consisting of multiple autonomous robotic systems will provide even more complete data and redundancy in the system.

NTNU has a vision of creating a coordinated observation system consisting of space, aerial, surface, and underwater platforms and vehicles that together can make observations of the ocean over long periods of time, as seen in Figure 1.1. As a part of this project, NTNU acquired the AutoNaut, which is an autonomous surface vehicle (ASV) capable of performing long-duration missions on the open ocean without physical human intervention. It is robust and can handle a tough environment, it is powered solely by green energy resulting in zero emissions, and it is quiet and will not disturb or harm its environment [7]. The AutoNaut is described in greater detail in Chapter 3.

1.1.2 Collision and grounding avoidance systems

One of the biggest challenges for the maritime transport sector is safety, and particularly collisions and groundings. The European Maritime Safety Agency (EMSA) publishes an annual report of marine casualties and incidents for ships flying a flag of the EU Member States, or that occurs within an EU Member States' territorial sea or internal waters. According to the report from 2020 'Annual Overview of Marine Casualties and Incidents 2020' [8], covering the period of 2014-2019, 44% of all casualty events were caused by collision, contact, or grounding/stranding. Half of the casualties occurred in internal waters (as defined by UNCLOS), and 41.5% in a port area. When the underlying factors (accident events) of a casualty or incident are looked at, 54% of the accident events are reported to be in the category of human action.

Accidents at sea can have huge consequences, including human casualties, environmental damage, and destruction of ships and equipment. The development of collision and grounding avoidance systems and the advancement of autonomous vessels can increase safety at sea by eliminating the human factor.

For the AutoNaut, it is important to have a collision and grounding avoidance system implemented to ensure its own and others' safety when operating in the fjord and on the ocean, among islands and other vessels. The AutoNaut's guidance and navigation system consists of three main systems: the path planning system that generates a mission plan with waypoints for the ASV to follow, the path following system that finds the desired course that makes the ASV reach the next waypoint, and the obstacle avoidance system that avoids collisions with dynamic and static obstacles by adding a course offset to the original desired course. The architecture of the navigation system is shown in Figure 3.4.

The path planning system should avoid collisions with known obstacles and avoid crossing known hazardous areas. However, unplanned changes in the AutoNaut's path may occur, caused by maneuvers to avoid dynamic obstacles or other previously unknown obstacles. Therefore, a robust reactive real-time collision and grounding avoidance system is needed.

Since the AutoNaut is a wave-propelled vessel, it cannot directly control its own speed but relies on environmental forces and has limited maneuverability. Therefore, the environmental conditions affect both the navigation and the speed of the vessel, and should be taken into account when deciding the optimal action to take in an obstacle avoidance situation.

This thesis will focus on the reactive obstacle avoidance system, especially the static obstacle avoidance system, including environmental factors. Such a system can ensure that the AutoNaut is able to operate autonomously in the fjord, among islands, and out in the open ocean, without harming itself or its surroundings.

1.2 Problem Description

The objective of this master's thesis is to develop a reactive anti-grounding system for the autonomous surface vehicle, the AutoNaut, using electronic navigational charts (ENCs), and integrate it into the existing simulation-based model predictive control (SB-MPC) collision avoidance system described in Johansen *et al.* [1] and implemented in Hagen *et al.* [2], [3]. The thesis will focus on the following parts:

- Develop an anti-grounding algorithm using electronic navigational charts (ENCs).
- Integrate the anti-grounding system into the existing collision avoidance system by extending the SB-MPC algorithm.
- Include environmental factors in the evaluation of which action the ASV should take in a given scenario.
- Test the anti-grounding system through simulations.

1.3 Relevant Previous Work

1.3.1 Collision avoidance

Collision avoidance (COLAV) has become a very active field of research, as the use of ASVs becomes more and more popular and the applications larger. There has been done a great amount of research on this field, and many different methods and approaches have been proposed to solve the COLAV problem.

In general, the methods can be divided into deliberate (global) and reactive (local) methods. Deliberate methods look at global, stored information about the ASV's surroundings and find a collision-free path for the ASV to follow from start to goal. This information is not necessarily currently available from the vessel's position. The main disadvantage of the deliberate methods is that they require long computational time, which is an issue in real-time applications. Examples of deliberate COLAV methods are the A^* search method [9], [10], Voronoi Diagram [11], and Rapidly-Exploring Random Trees (RRT) [12].

Reactive methods consider only the immediate surroundings of the ASV and are based on currently available sensor data. They have low computational cost and can quickly react to changes. Therefore, they are suited to handle real-time situations in dynamic surroundings. The main issue with reactive methods is that they can get stuck in local minima, meaning that they are not able to find the optimal global solution. Examples of reactive COLAV methods are Velocity Obstacle (VO) [13], Dynamic Window (DW) [14], [15], and Potential Field [16].

A hybrid system containing both a deliberate method and a reactive method is optimal for an ASV, where the high-level path planning is done using a deliberate method, and the real-time obstacle avoidance is done by a reactive method. This is studied in [17] and [18], and is similar to how the AutoNaut system is arranged, as seen in Figure 3.4.

1.3.2 MPC for collision avoidance

The use of Model Predictive Control (MPC) for collision avoidance has been studied extensively and is a powerful method that gives a design flexibility superior to other approaches found in the literature [3].

The collision avoidance concept presented in Johansen *et al.* [1] is a simulation-based model predictive control (SB-MPC) approach, which forms the basis for the collision and grounding avoidance system studied in this thesis. Optimization is done over a finite number of control behaviors in order to help mitigate problems with computational complexity, local minima, and dependability. This method is well known in the literature on robust MPC [19], [20].

The SB-MPC system is further described, implemented, and tested in Hagen *et al.* [2], [3]. In Kufoalor *et al.* [21], a field verification of the system is performed in the North Sea, and it is shown that the MPC approach is capable of providing COLREGS compliant behaviors, ensuring collision avoidance when navigating among vessels equipped with AIS.

The SB-MPC collision avoidance system algorithm will be described in detail in Chapter 5.

1.3.3 Anti-grounding

Only a small part of the literature on obstacle avoidance focuses on anti-grounding methods, also known as static obstacle avoidance (SOA), for ASVs. An example is Tang *et al.* [22] where a general local reactive obstacle avoidance algorithm, including both static and dynamic obstacles, is developed for high-speed USVs. Another example is in Guardeño *et al.* [23], where a new algorithm called the Robust Reactive Static Obstacle Avoidance System (RRSOAS) is developed. An occupancy probability grid is used to model the surroundings of the vessel in this system.

In Blindheim *et al.* [24], the use of MPC for handling emergency situations that normally are taken care of by human operators is investigated, and a dynamic risk-based decision-making algorithm is developed. Simulations are done in a strait with grounding obstacles on each side. The grounding obstacles are modeled as static obstruction circles, which is a simple representation, and it is emphasized that dynamic calculations of the grounding areas should be developed and utilized. The following ad hoc risk cost function is used, which includes a term for grounding risk cost and is interesting to study because it also considers the wind speed and direction.

$$\rho(\boldsymbol{x}_k, \boldsymbol{\theta}_k) = \sum_{j=1}^{J} \left(\mu_1 + \mu_2 \chi_j V_w^2\right) e^{-\frac{1}{\zeta^2}(||\boldsymbol{e}_j - \boldsymbol{p}_k|| - r_j)^2} \tag{1.1}$$

The term $\chi_j = \max(0, \hat{i}_j \cdot \hat{\omega})$, where $\hat{i}_j$ is the unit vector from the ship to each obstacle center, and $\hat{\omega}$ is the unit wind direction vector. The wind speed and direction are included in the cost function to increase the risk of an obstacle to

the west if the wind comes from the east, due to the increased risk of the ship drifting towards this obstacle. If the wind is coming from the west, the risk of the obstacle to the west is not changed. The paper concludes that risk-based MPC is considered an appropriate method for trajectory planning and decision-making algorithm for autonomous ships during emergencies.

In Bakdi *et al.* [25], an algorithm for identifying collision and grounding risk is developed utilizing spatial risk functions. The grounding risk is calculated by looking at the intersection between the vessel's safety domain and the shoreline polygons representing land, which results in multiple grounding risk regions.

In Mazaheri *et al.* [26] a literature review and discussion is done of the available risk models developed for ship grounding risk analysis. More than 90 articles are reviewed, and 13 models are thoroughly assessed.

1.3.4 Electronic navigational charts

The use of Electronic Navigational Charts (ENCs) in the navigation of marine vehicles is a researched topic; however, most of the research focuses on how to use them with deliberate path planning methods. In Kang *et al.* [27] for example, water depth information from the ENCs is used for high-level route planning.

Larson *et al.* [28], [29], utilize ENCs (called DNCs) as one of several sensors to create the world model used in reactive static obstacle avoidance. In Reed and Schmidt [30], the importance and advantages of utilizing ENCs to give the ASV a priori knowledge and a holistic view of its surroundings are emphasized. A behavior-based reactive obstacle avoidance approach is studied, where ENCs are used to identify the obstacles. A method for extracting and converting ENCs from the S-57 format to a database is described. Extraction and transformation of ENC data is also discussed in Mąka and Magaj [31], with a focus on regular and irregular meshes. For the irregular meshes, the resolution is increased in areas around obstacles to obtain a higher level of detail.

In the master's theses of Otterholm [32], Midjås [33], and of Grande [34], ENC-based anti-grounding systems for the existing SB-MPC collision avoidance system are developed.

Otterholm focuses on developing an application that extracts information about mapped hazards from ENCs and translates this information to a suitable input for the CAS. Experimentation on how to parse, extract and handle the information from the ENCs and how to represent the hazards is performed, and a real-time decision-making application that provides cost weights for anti-grounding to the CAS algorithm is developed. The approach is based on representing the static obstacles as polygons and will therefore require a considerable amount of storage space and computational power. This will be further described and discussed in Chapter 4.

Midjås develops a collision avoidance system for the ReVolt ASV, including an ENC-based anti-grounding system. The algorithm designed by Otterholm is used to extract ENC data, and the anti-grounding system is also inspired by Otterholm.

Grande focuses on the probabilistic version of the SB-MPC (PSB-MPC), as examined in Tengesdal *et al.* [35], [36], and implements an ENC-module where the static obstacles are represented as polygons generated from shapefiles, similar to Otterholm. The grounding cost term that is added to the PSB-MPC algorithm is based on the grounding risk cost presented in Blindheim *et al.* [24], see Equation (1.1).

In the thesis of Lauvås [37], a different method for extraction and representation of ENC data is developed in collaboration with Alberto Dallolio. Here, an SQLite database is created in which adapted ENC data is stored. The main advantage of this method is that it is very fast and in need of little storage space. Therefore, this is the preferred ENC extraction method for the anti-grounding system developed in this thesis. The method and implementation will be described in detail in Chapter 4.

1.4 Outline

The thesis consists of 9 chapters. First, Chapter 2 introduces the theoretical background of the thesis, giving an introduction to terms and concepts used in the other chapters. The AutoNaut is presented in Chapter 3. In Chapter 4, the electronic navigational charts used in the anti-grounding system are described, together with two methods for data extraction and obstacle representation. Chapter 5 describes the collision and grounding avoidance system in detail and how it is implemented in the AutoNaut. The anti-grounding theory is also validated in this chapter. In Chapter 6, the system is tested through various simulations. Chapter 7 contains a discussion of the system and the simulation results. Chapter 8 concludes the thesis, and finally, the proposed future work is presented in Chapter 9.

Chapter 2

Theoretical Background

2.1 Kinematics

Kinematics treat the geometrical relationships between bodies moving in space, while kinetics analyze the forces causing this motion. The coordinate frames used to describe the motion of rigid bodies will be defined in this section. The notation follows the notation of SNAME (1950) [38] for marine craft.

BODY				**NED**
DOF		**Forces and moments**	**Linear and angular velocities**	**Positions and Euler angles**
1	Motions in the x_b-direction (surge)	X	u	x^n
2	Motions in the y_b-direction (sway)	Y	v	y^n
3	Motions in the z_b-direction (heave)	Z	w	z^n
4	Rotation about the x_b-axis (roll)	K	p	ϕ
5	Rotation about the y_b-axis (pitch)	M	q	θ
6	Rotation about the z_b-axis (yaw)	N	r	ψ

Table 2.1: The SNAME (1950) notation for marine craft [39].

2.1.1 Coordinate frames

Different coordinate frames are necessary when analyzing the motion of a marine craft in 6 degrees of freedom (DOFs). The position of the craft is typically expressed in the North-East-Down (NED) frame, while the velocities of the craft

are expressed in the BODY frame, and the orientation expressed in Euler angles describe the orientation of BODY with respect to NED.

There are two main types of coordinate systems, the Earth-centered reference frames and the geographic reference frames. An inertial frame is a nonaccelerating coordinate frame where Newton's laws of motion apply. The coordinate frames presented here are based on chapter 2.1.1 in Fossen (2021) [39].

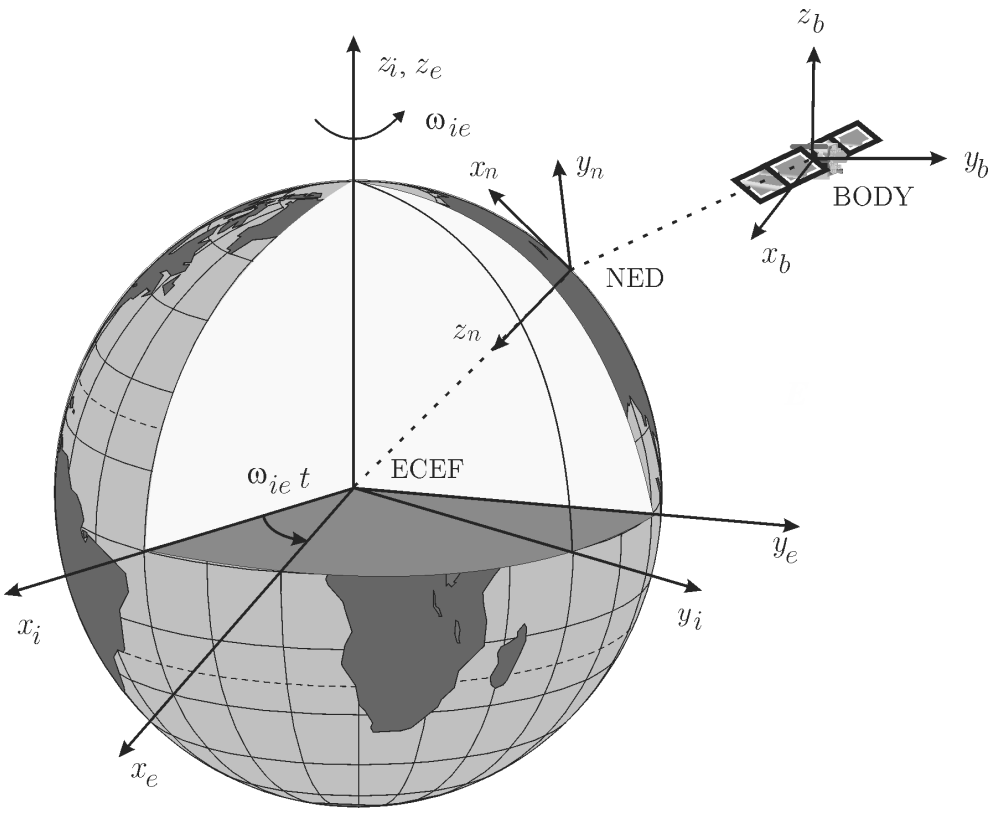


Figure 2.1: Illustration of the coordinate frames [39].

Earth-centered inertial frame (ECI)

The Earth-centered inertial (ECI) frame is denoted by $\{i\} = (x_i, y_i, z_i)$ and has its origin located at the center of the Earth o_i. The ECI frame is fixed in space and is an inertial frame.

Earth-centered Earth-fixed frame (ECEF)

The Earth-centered Earth-fixed (ECEF) frame is denoted by $\{e\} = (x_e, y_e, z_e)$ with origin at o_e at the center of the Earth. Different from ECI, the axes of ECEF rotate with the Earth, relative to ECI.

North-East-Down frame (NED)

The North-East-Down (NED) frame is denoted by $\{n\} = (x_n, y_n, z_n)$ where x_n points towards north, y_n points towards the east and z_n points downwards to the center of the Earth, normal to the Earth's surface. The origin o_n is set on the Earth's surface in the area where navigation is to be done. NED is a geographic reference frame that is tangential to the Earth's surface. It is often approximated as the inertial frame for local navigation.

Body-fixed frame (BODY)

The body-fixed reference frame is denoted by $\{b\} = (x_b, y_b, z_b)$ with origin o_b. This frame is fixed to the body of the craft and moves relative to the inertial reference frame ($\{e\}$ or $\{n\}$). For a marine craft, the body axes are usually set along the principal axes of inertia. Here, x_b is the longitudinal axis from the aft of the craft to the fore, y_b is the transversal axis with direction towards starboard, and z_b is the normal axis with direction from the top to the bottom of the craft. The origin o_b is usually placed midships in the waterline.

2.1.2 Rotation matrices

The rotation matrix R is used to transform vectors between coordinate systems. When transforming a vector from one frame v^{from} to another frame resulting in the vector v^{to}, the rotation matrix $R^{\text{to}}_{\text{from}}$ is used. For a surface vessel in 3 DOFs, the Euler angle rotation matrix $R(\Theta_{nb})$ is defined as $R(\psi)$.

$$R(\Theta_{nb}) = R_{z,\psi} = R(\psi) \tag{2.1}$$

2.1.3 Definitions of heading and course

For a marine surface craft, the relationship between course and heading is important. Figure 2.2 shows the ocean current triangle, which illustrates the relationship between heading and course.

The heading (or yaw) angle ψ is defined as the angle between the x_n axis (north) and the x_b axis of the craft, with positive rotation defined about the z_n axis.

The course angle χ is defined as the angle between the x_n axis (north) and the velocity vector of the craft, with positive rotation defined about the z_n axis. The relationship between the course and the heading is defined as

$$\chi := \psi + \beta_c \tag{2.2}$$

where β_c is the crab angle, which is defined as

$$\beta_c = \tan^{-1}\left(\frac{v}{u}\right) = \sin^{-1}\left(\frac{v}{U}\right) \tag{2.3}$$

The amplitude U of the velocity vector is defined as

$$U = \sqrt{u^2 + v^2} \tag{2.4}$$

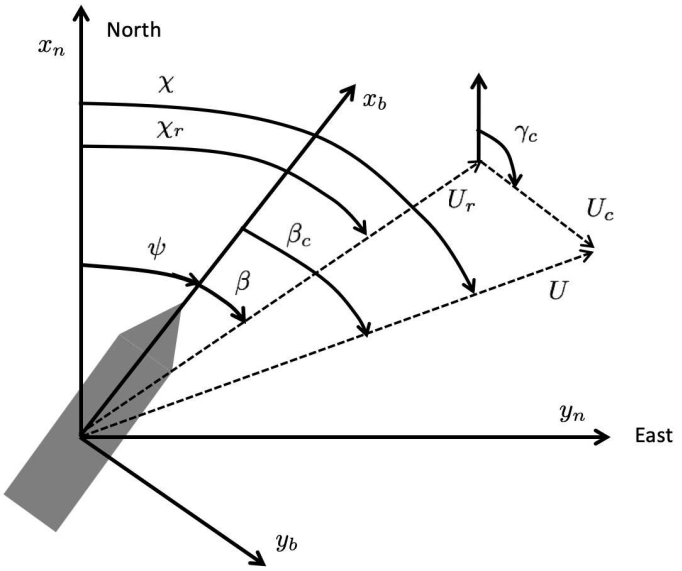


Figure 2.2: Illustration of the ocean current triangle in the horizontal plane [39].

2.2 Rigid-Body Kinetics

The motion of a ship can be described by the rigid-body equations of motion. In this thesis, the equations of motion are used in the prediction of the AutoNaut's trajectory. The AutoNaut is a surface vessel, which means that the motions of the vessel can be described in 3 DOFs (surge, sway and yaw), instead of using the full 6-DOF equations of motion. This is based on the assumption that the dynamics associated with motion in heave z, roll ϕ and pitch θ are small during normal operation of surface vessels and can be neglected. Thus, the 3-DOF (surge, sway and yaw) representation can be used for a simplified motion control model.

The general equations of motion for a surface marine craft in 3 DOFs, can be written on the following form, as seen in Fossen (2021) [39] equations (6.92) and (6.102)

$$\begin{aligned} \dot{\eta} &= R(\psi)\nu \\ M\dot{\nu}_r + C(\nu_r)\nu_r + D\nu_r + D_n(\nu_r)\nu_r &= \tau + \tau_{wind} + \tau_{wave} \end{aligned} \tag{2.5}$$

where $\eta = [x^n, y^n, \psi]^T$ is the generalized position vector expressed in $\{n\}$, $\nu_r = \nu - \nu_c$ is the generalized relative velocity vector expressed in $\{b\}$, with $\nu_c = [u_c, v_c, 0]^T$ being the ocean current velocity vector, and $\nu = [u, v, r]^T$ is the absolute velocity vector. The system inertia matrix M and Coriolis and centripetal

matrix $C(\nu)$ are defined as

$$\begin{aligned} M &= M_A + M_{RB} \\ C(\nu_r) &= C_A(\nu_r) + C_{RB}(\nu_r) \end{aligned} \tag{2.6}$$

where M_{RB} is the rigid-body mass matrix, M_A the added mass matrix, $C_{RB}(\nu)$ is the rigid-body Coriolis and centripetal matrix and $C_A(\nu)$ is the added mass Coriolis and centripetal matrix. Further, D is the linear damping matrix, $D_n(\nu)$ is the non-linear damping matrix and $R(\psi)$ is the rotation matrix between the BODY frame and the NED frame. The environmental forces and moments are disturbances acting on the surface vessel that affect the dynamics of the vessel. The wind and wave forces and moments are represented in τ_{wind} and τ_{wave}, and the current is included in ν_r. The control inputs are added in the vector τ.

2.3 Geodesy

Geodesy is the science of accurately measuring the Earth's geometric shape, orientation in space and gravity field [40]. A geodetic datum, or a geodetic reference frame, is a reference frame for measuring and describing point locations on the Earth. There exist several reference frames. ITRF (International Terrestrial Reference Frame) and WGS84 are the two main global reference frames used today, and the difference between them is only within a few centimeters. Since the Earth is not a perfect ellipsoid, local reference frames can give a more accurate representation of an area than a global frame. EUREF89 is a regional reference frame for Eurasia and is used in "Noregs hovudkartserie i målestokk 1 : 50 000 N50" [41]. OSGB36 is a local reference frame, which is a good approximation for the area of the British Isles.

2.3.1 World Geodetic System 1984 (WGS84)

The World Geodetic System is a global geodetic reference frame given out by NIMA (National Imagery and Mapping Agency), now known as NGA (National Geospatial-Intelligence Agency), which is under the U.S. Department of Defense. The WGS84 is a geocentric reference frame with its origin located in the center of the Earth, its coordinates are given in longitude, latitude and height above the ellipsoid surface, and it is globally consistent within a meter. WGS84 is used by the GNSS-system (Global Navigation Satellite Systems), among them the GPS (Global Positioning System). WGS84 is also used by Kartverket for all Norwegian nautical charts and ENCs.

2.4 The LSTS Toolchain

The software used in this thesis is the open-source LSTS software toolchain, which consists of DUNE onboard software, Neptus command and control software, and

the IMC communications protocol [42]. This software is developed by the Underwater Systems and Technology Laboratory (LSTS) at the University of Porto. The purpose of this software toolchain is to create a networked vehicle system consisting of human operators, autonomous vehicles, and other sensing devices. The network is dynamic and allows for vehicles to come and go, such that the vehicles can work as sensing and communication devices.

2.4.1 Neptus

Neptus is the command and control software that supports the phases in a mission life cycle, which consists of planning, simulation, execution, and review and analysis after the mission. Neptus is used to control the autonomous vehicles from the ground by sending plans, changing settings, and to get an overview of the mission. All vehicles that are connected to the network are shown here. Neptus can communicate to DUNE over IMC.

2.4.2 DUNE

DUNE (DUNE Uniform Navigation Environment) is the onboard software framework used in the embedded systems. It is written in C++ and consists of submodels called tasks that each do a certain logical operation and usually run in distinct threads of execution. The tasks communicate over a message bus by dispatching and consuming IMC messages.

2.4.3 IMC: Inter-Module Communication

IMC (Inter-Module Communication) is the protocol that defines the common control message that is used internally in DUNE and between Neptus and DUNE. These messages can be understood by all vehicles and computers in the network, which is important to allow for communication between the vehicles, sensors, and human operators in the network. The messages can be anything from sensor inputs to control outputs and information about the mission plan.

2.5 COLREGS

The International Regulations for Preventing Collisions at Sea 1972 (COLREGS) are published by the International Maritime Organization (IMO), to secure a common set of navigation rules for ships and other vessels at sea. Autonomous vehicles must also follow the rules of COLREGS, and these rules need to be implemented in the collision avoidance system (CAS) of the AutoNaut to avoid collisions and behave as expected by other vessels.

In this section, the main rules of COLREGS that are relevant for the AutoNaut's CAS are specified as in the convention of 1972 [43].

2.5.1 Rule 8 - Action to avoid collison

(b) *Any alteration of course and/or speed to avoid collision shall, if the circumstances of the case admit, be large enough to be readily apparent to another vessel observing visually or by radar; a succession of small alterations of course and/or speed should be avoided.*

(d) *Action taken to avoid collision with another vessel shall be such as to result in passing at a safe distance. The effectiveness of the action shall be carefully checked until the other vessel is finally past and clear.*

Additionally, if there is sufficient sea-room, alteration of course alone may be the most effective action. Although, if necessary, a vessel should reduce its speed, stop or reverse.

2.5.2 Rule 13 - Overtaking

(a) *Notwithstanding anything contained in the Rules of part B, section I and II, any vessel overtaking any other shall keep out of the way of the vessel being overtaken.*

(b) *A vessel shall be deemed to be overtaking when coming up with another vessel from a direction more than 22.5 degrees abaft of her beam, that is, in such a position with reference to the vessel she is overtaking, that at night she would be able to see only the sternlight of that vessel but neither of her sidelights.*

2.5.3 Rule 14 - Head-on situation

(a) *When two power-driven vessels are meeting on reciprocal or nearly reciprocal courses so as to involve risk of collision, each shall alter her course to starboard so that each shall pass on the port side of the other.*

2.5.4 Rule 15 - Crossing situation

When two power-driven vessels are crossing so as to involve risk of collision, the vessel which has the other on her own starboard side shall keep out of the way and shall, if the circumstances of the case admit, avoid crossing ahead of the other vessel.

2.5.5 Rule 16 - Action by give-way vessel

Every vessel which is directed to keep out of the way of another vessel shall, so far as possible, take early and substantial action to keep well clear.

2.5.6 Rule 17 - Action by stand-on vessel

(a) (i) *Where one of two vessels is to keep out of the way the other shall keep her course and speed.*

(b) *When, from any cause, the vessel required to keep her course and speed finds herself so close that collision cannot be avoided by the action of the give-way vessel alone, she shall take such action as will best aid to avoid collision.*

Chapter 3

The AutoNaut

The AutoNaut is a self-powered autonomous surface vehicle (ASV). It was developed by AutoNaut Ltd (formerly called MOST (Autonomous Vessel) Ltd) in 2013. The ASV is made to remain at sea for several months at the time and therefore it has a robust design. The vessel is stable in rough conditions; it is self-righting in the case of capsize and has, according to the producer, survived 65 kt storm conditions and 10 m waves [44].

The AutoNaut is powered by wave energy, using the patented Wave Foil Technology. This wave propulsion technology converts energy from the pitch and roll motions of the hull in the waves to generate forward propulsion. Spring-loaded foils are mounted to struts on the keel, one at the fore and one at the aft of the vessel. The foils take advantage of the wave-induced vessel motion where the vessel is lifted up on the crest of the wave and drops down into the trough. The vessel can move forward in any direction independently of the wave direction. If the weather conditions are very calm and there is not sufficient energy in the waves to move the AutoNaut, an electrical thruster attached on the stern strut can be used as auxiliary propulsion. For the 5-meter version of the AutoNaut, the speed from wave propulsion is typically 1-3 knots, while the thruster can give a speed of up to 1 knot [45], [46].

All scientific, navigation and communication sensors, the control system and the thruster on the AutoNaut are powered by solar energy. The vessel is equipped with an array of photovoltaic solar panels of 300 W that generate electrical energy, which is stored in batteries.

3.1 NTNUs AutoNaut

The AutoNaut that NTNU has acquired is the 5-meter version of the AutoNaut, which has a max speed of up to 3 knots. The hull, and the propulsion technology and hardware are provided by AutoNaut Ltd., while the rest of the internals, i.e., the control system and sensors, are designed by NTNU. Table 3.1 shows the specifications of the AutoNaut 5. In this thesis, the AutoNaut that is referred to is NTNUs version of the AutoNaut.

Figure 3.1: The AutoNaut ASV in the Trondheimsfjord [7].

The term ASV (Autonomous Surface Vehicle) will be used for the AutoNaut in place of USV (Unmanned Surface Vehicle) since the AutoNaut can operate autonomously without human intervention, which is the definition of an ASV. USVs can also include vehicles that are operated remotely by humans, making ASV a more exact term for the AutoNaut.

Dimensions	
Length	5.0 m
Beam	0.8 m
Displacement	230 kg
Draft	0.7 m
Mast height	1.5 m
Speed	1-3 knots

Table 3.1: Vessel specifications for NTNUs AutoNaut [44], [46].

3.2 System Architecture

The system architecture of the AutoNaut is designed to ensure robustness to mission failure and a high degree of redundancy. The system is divided into three layers, as shown in the simplified diagram in Figure 3.3. This layered subdivision

provides the necessary robustness and redundancy [47].

Level 1 is the low-level component that takes care of system monitoring and works as a fallback autopilot in situations where the main navigation system is unable to control the ASV's rudder. This component is set up as a state machine, with a normal mode, fallback mode and a manual mode. It is able to detect anomalies in the system and will intend to find a solution and inform the operators about the problem. The complexity of this component is desired to be as low as possible.

Level 2 includes the navigation system with a course-keeping autopilot, AIS-based collision avoidance and ENC-based anti-grounding. The architecture of the navigation system is shown in Figure 3.4. The computational unit used in this level is the single-board computer BeagleBone Black (BBB). It runs GLUED (GNU/Linux Uniform Environment Distribution), and DUNE and the collision and grounding avoidance system is executed on this computer. Level 2 receives data from GPS, magnetometer, IMU and Weather Station and uses this data to determine the current state of the vessel and the state of the sea. A path following line-of-sight (LOS) guidance law is used to compute the desired course that is given as reference to the control system, which makes the vessel reach the desired waypoint. The collision and grounding avoidance system implemented in Level 2 will be described in detail in Chapter 5.

Level 3 is used for the scientific payload and does not include functions for guidance or navigation purposes. Therefore, level 3 will not be studied further in this thesis.

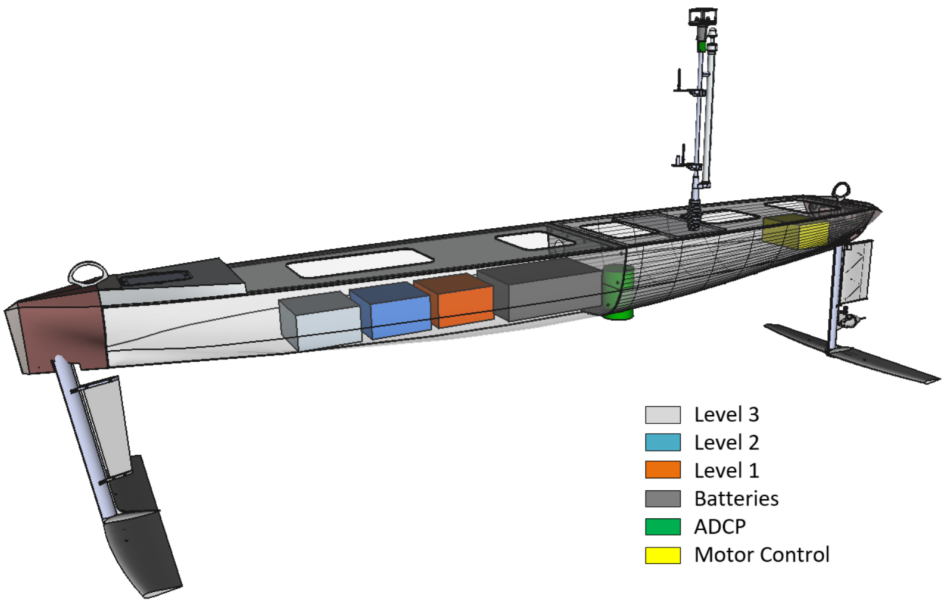


Figure 3.2: 3D model of the AutoNaut where the placement of the levels is shown [47].

3.3 Sensors

3.3.1 Navigation sensors

The navigation sensors are used for level 2 and include GPS, AIS, Weather Station, IMU, and a digital compass. GPS is also used in level 1. A full overview of the sensors used for each level is shown in Figure 3.3. The AIS is important for the collision avoidance system. The Weather Station is used both for navigational purposes and environmental analysis. An overview of the navigation sensors is given in Table 3.2.

Sensor name	**Description**
Vector™V104 GPS Smart Antenna	Provides accurate heading and position.
Raymarine AIS650	Equipped with a GPS antenna and a VHF antenna. Receives continually AIS data from surrounding vessels.
Airmar 120WX Weather Station	Provides wind speed and direction, air temperature and barometric pressure readings [48].
ADIS16485 IMU	Inertial measurement unit (IMU) used for control and navigation.
Honeywell HMR3000 Digital Compass	Provides heading, pitch and roll.
SenTiBoard	A sensor timing board that accurately records when sensor messages are validated [49].

Table 3.2: Navigation sensors, information from [46].

3.3.2 Communication sensors

The communication sensors are used for level 1 and level 2. An overview of the communication sensors is shown in Table 3.3 and in the high-level structure diagram in Figure 3.3.

3.3.3 Scientific sensors

The scientific sensors are implemented on level 3 and include sensors for measuring current, waves, ice, water conditions, chlorophyll and tracking of fish. An overview of these sensors is shown in Table 3.4.

Sensor name	Description
RockBLOCK+ Iridium	Provides satellite communication with the Iridium satellites.
Owl VHF	Radio transceiver.
MikroTik 3G/4G Modem	Cellular modem that supports 2G, 3G and 4G (LTE) connectivity.

Table 3.3: Communication sensors, information from [46].

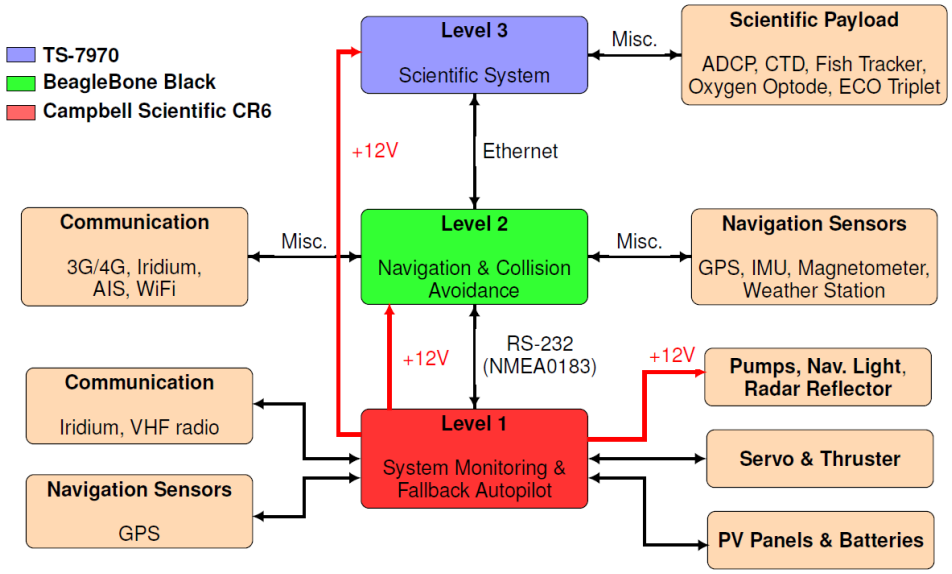


Figure 3.3: System architecture for the AutoNaut [47].

Sensor name	**Description**
Nortek Signature500 ADCP	Measures current profiles and turbulence, wave height, direction and ice tracking.
Seabird CTD SBE 49	A CTD sensor that measures conductivity, temperature and pressure of seawater.
ThelmaBiotel TBLive	A hydrophone that allows for a live data feed for active tracking of fish.
Aanderaa Oxygen Optode 4835	Measures absolute oxygen concentration and % saturation in the sea.
WET Labs ECO Puck Triplet	Is configured to do biogeochemical measurements of chlorophyll and FDOM fluorescence, remote sensing and particle dynamics measurements, blue, green and red backscattering.

Table 3.4: Scientific sensors, information from [46].

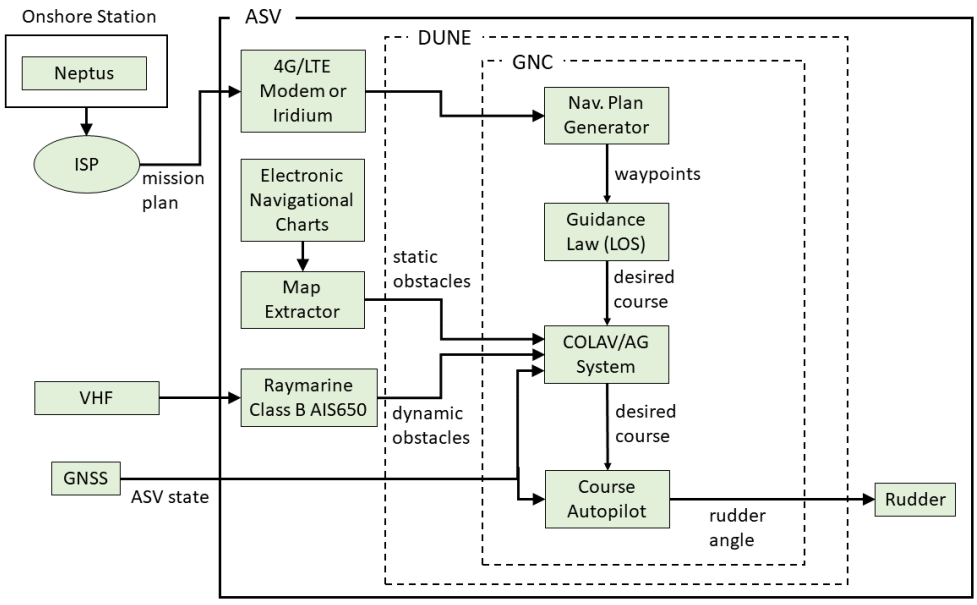


Figure 3.4: Navigational system architecture.

Chapter 4

Electronic Navigational Charts for Anti-Grounding Systems

A human mariner uses nautical charts when navigating, especially in unfamiliar areas. Similarly, an ASV should be able to read and utilize the information found in a nautical chart by using electronic navigational charts (ENCs). Understanding ENCs would give a great advantage to the ASV, increase the safety of navigation and its autonomy. It is also important that ASVs have access to the same standardized information as the human mariners when navigating in the same waters.

From the ENCs, the ASV can obtain knowledge about its surroundings and improve both its a priori situational awareness and the reactive obstacle avoidance system. Thus, it will be aware of the surrounding grounding obstacles when it has to make unplanned maneuvers to avoid dynamic obstacles, and the system can choose the actions that minimize hazard considering all obstacles.

The anti-grounding system developed in this thesis is based on Electronic Navigational Charts (ENCs). ENCs are vector-based electronic maps that contain all information necessary to conduct safe navigation at sea. The database containing the ENCs is made by national hydrographic offices for the International Hydrographic Organization (IHO). In Norway, it is Kartverket that is responsible for producing and updating the Norwegian ENCs. The maps follow the IHO standard S-57 for transfer of digital hydrographic data [50].

Different methods can be used to extract and represent data from the ENCs. In this chapter, two methods for data extraction and for obstacle representation are studied. In the method used by Otterholm [32], Midjås [33] and Grande [34], shapefiles are created and static obstacles are represented as polygons. In the method developed in Lauvås [37], the static obstacles are represented as point clouds and stored in a database.

4.1 The S-57 Standard

The S-57 Standard is designed to describe real-world entities that are relevant to hydrography. The model used by the standard presents the data as objects containing spatial and descriptive characteristics. Figure 4.1 illustrates the model used by the S-57 standard [51]. The spatial object describes the location and geometry of the feature object and can be of type vector, raster, or matrix. The vector representation can be implemented as points, lines, or areas corresponding to zero, one, or two dimensions. The feature objects are categorized into four types of objects: Meta, Cartographic, Geo, and Collection. These are defined in the IHO Object Catalogue found in [52].

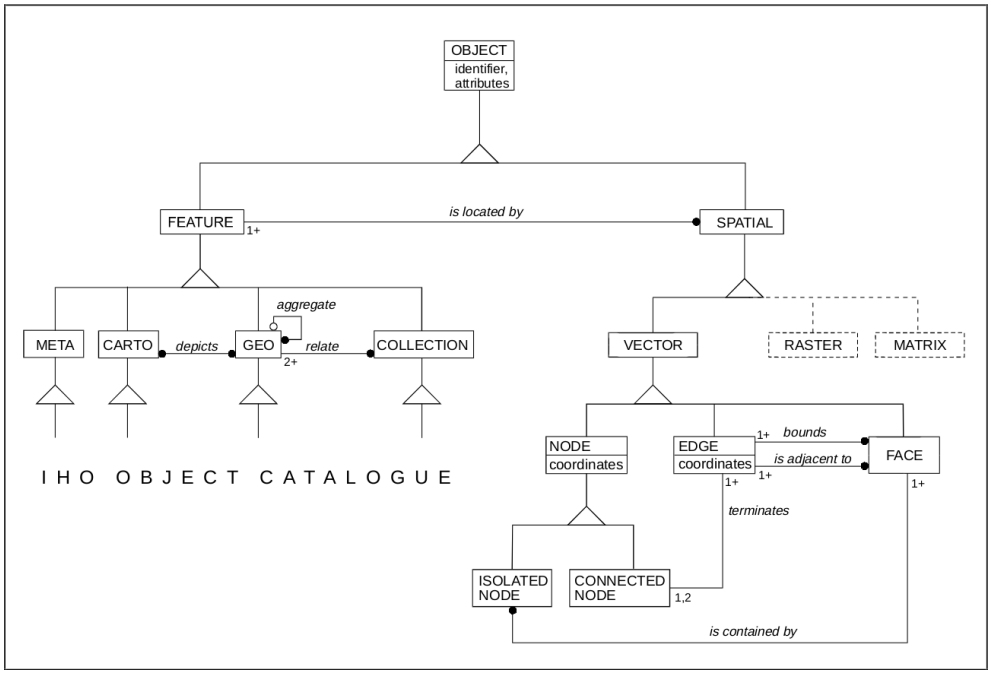


Figure 4.1: S-57 theoretical data model [51].

The IHO Object Catalogue for the S-57 standard [52] contains descriptions and classifications of the physical entities that exist in the real world, like buoys, beacons, etc. The entities are categorized into a finite number of types, called feature object classes. Each object can further be described by attributes and attribute values.

Objects that are most relevant to the AutoNaut are shown in Table 4.1. The complete list of objects that can be encountered in an ENC is found in [52]. The object DEPARE (Depth Area) defines water areas where the depth is within a certain range of values and will be used to obtain depth information for the anti-grounding system presented in this thesis. In DEPARE, the depth range is defined by the attributes DRVAL1 and DRVAL2. DRVAL1 is the shallowest depth, and DRVAL2 is the deepest depth in the range.

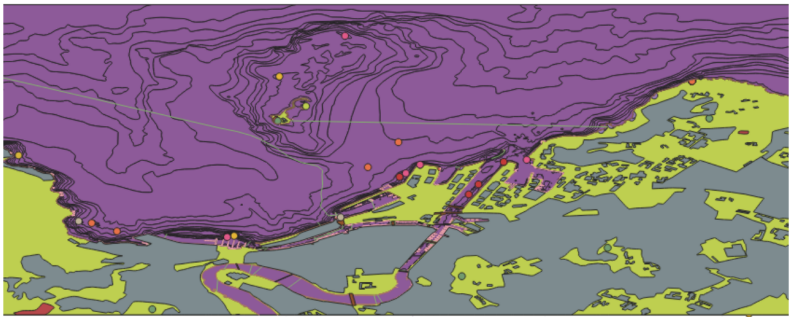
4.2 Shapefiles and Polygon Representation

In Otterholm [32], an application that extracts ENC data, saves it in a shapefile as polygon representations, and gives it as input for the CAS, is developed. Midjås [33] uses the algorithm developed by Otterholm. The ENC-module created by Grande [34] uses the same approach of generating shapefiles that represent the ENC data as polygons.

The approach is based on creating geometrical polygons that represent no-go zones where there is land or static obstacles. The algorithm does this by extracting the desired S-57 features from the ENCs and returns the extracted information in a file of type ESRI Shapefile. A shapefile is a format for storing nontopological geometry and attribute information for the spatial features in a data set. A set of vector coordinates is used to describe the geometry of a feature and can represent them as points, lines, or polygons [53]. In the file, the different features are stacked in separate layers, and the algorithm merges these layers into one layer containing all the information represented as polygons. The resulting polygons are the outline of the hazardous areas defined as no-go zones for the vessel. Figure 4.2a shows the complete ENC data, and Figure 4.2b shows the result of the algorithm, which is extracted data represented as polygons in one layer.

During operation, the predicted path of the ASV is checked against the no-go zones in the map to see if the ASV is in danger of colliding with any static obstacle in the near future. Action to avoid collision is then taken if necessary. The system is implemented using ROS as the framework and GDAL/OGR to access data from the ENC.

This approach is seen to be quite computationally heavy and in need of considerable storage space because the shapefiles are large and, as pointed out by Midjås, checking for intersections between the no-go zones and the predicted path of the ASV at every time step could cause computational problems in real-time. The AutoNaut is a small vessel that should be able to operate using only satellite communication, and it needs a compact and fast solution for usage of ENCs. Therefore, the method presented here is not optimal for the anti-grounding system of the AutoNaut.



(a) The original ENC data containing all features [33].

(b) Reduced shapefile containing only desired features, showing the no-go zones in purple [33].

Figure 4.2: The ENC extraction and representation method used by Otterholm, Midjås and Grande, [33].

4.3 Database and Point Cloud Representation

In the anti-grounding system presented in this thesis, the method used for extraction and representation of ENC data is the method developed in the thesis of Lauvås [37], in collaboration with Alberto Dallolio. This method is very fast and needs little storage space, which is a great advantage for the AutoNaut, and is therefore the preferred solution. In short, the method extracts data of interest from the ENCs, creates two-dimensional grids from the data, and stores the data coordinates and attribute values in a database. The ENC information is then easily accessed from the anti-grounding algorithm.

According to Lauvås, Kartverket recommends using the FME software suite when working with the data from S-57 ENCs. FME (Feature Manipulation Engine) is a spatial ETL (Extract, Transform and Load) application, which focuses on translation of geographic data. The FME Desktop Workbench is suited for extraction and transformation of data; it supports both S-57 ENCs and SQLite3, and is therefore used by Lauvås.

The FME workbench shown in Figure 4.4 is used to create two-dimensional grids of points from the original polygon representation of the S-57 DEPARE object. First, a square grid is created around the ASV in a given resolution, and each point is given a DEPARE attribute value. Then, a filter removes all the areas that

are not described by DEPARE. The coordinates of the remaining points are stored in the database as Lat and Lon, together with their corresponding DRVAL1 and DRVAL2 values.

A local SQLite3 database is stored on the ASV. The SQLite3 database is chosen because it is lightweight, it does not need to be connected to a server, it is portable since it stores all data in a single file, and it is already integrated in the LSTS toolchain and utilized by both DUNE and Neptus.

In the anti-grounding algorithm, the data can be easily retrieved from the database in the form of WGS84 location points through a simple SQL query. The SQL statement is as follows: "SELECT * FROM DEPARE WHERE Lat BETWEEN a and c AND Lon BETWEEN d and b;", where a, b, c, and d are as defined in Figure 4.3. The statement can be specified further by setting a specific value for DRVAL1 and/or DRVAL2, and thus reducing the amount of handled data. In the implementation of the anti-grounding algorithm, a minimum safe depth value can be set, which will decide the value of the deepest depth in the depth range, DRVAL2. The SQL statement will then be "SELECT * FROM DEPARE WHERE DRVAL2=value and Lat BETWEEN a and c AND Lon BETWEEN d and b;".

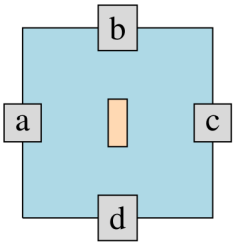


Figure 4.3: The square limits of the area around the vessel where grounding data is retrieved in the SQL query [37].

In Figure 4.5a, the original DEPARE ENC object is shown. The lines represent the depth contours. Figure 4.5b shows the result of the query above and the method presented in this section. The point cloud represents the depth contour corresponding to DRVAL2 = 10.0 m in a square area around the AutoNaut. A result of the query can also be seen in Figure 5.7a.

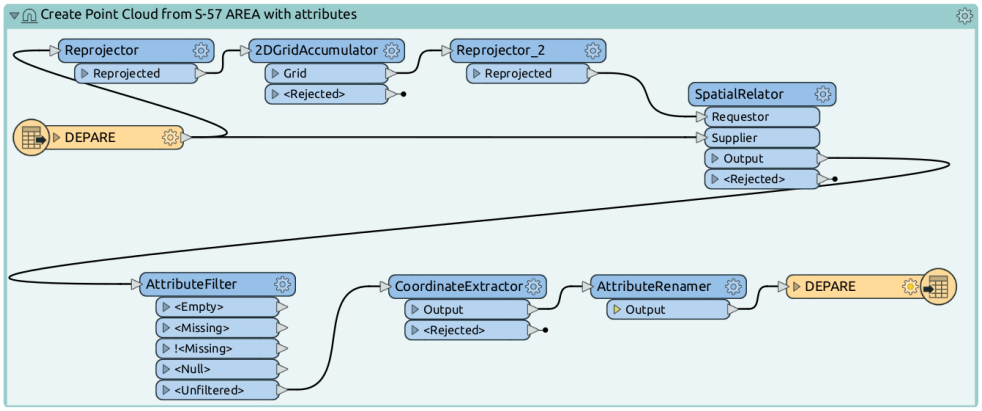
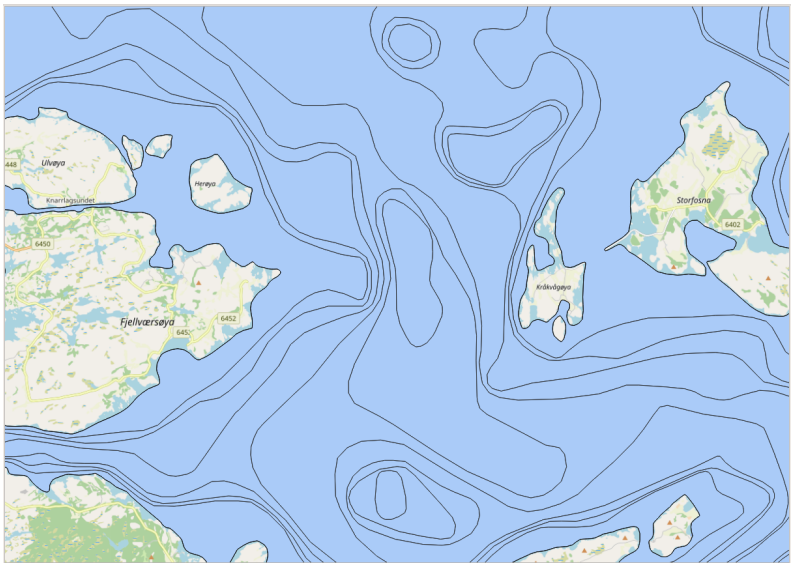
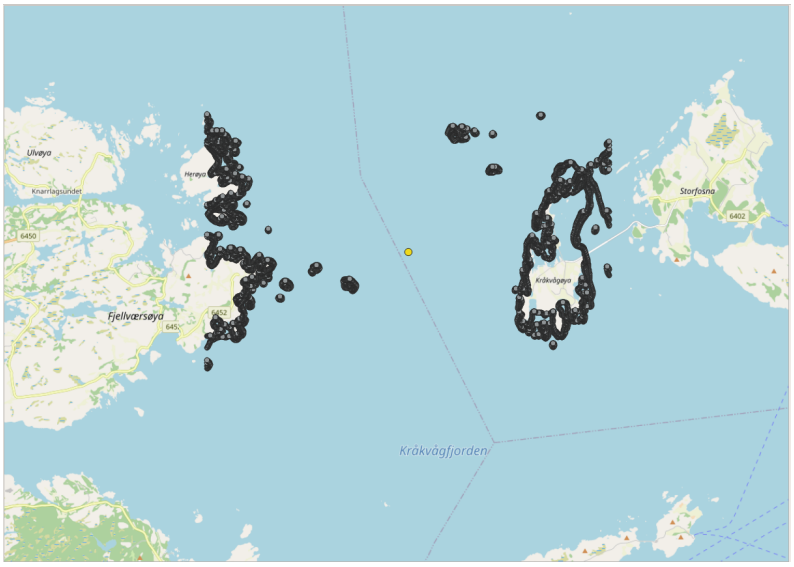


Figure 4.4: The FME workbench used to create point clouds of the DEPARE ENC data, from [37].



(a) The original DEPARE ENC object where the lines represent depth contours.



(b) The resulting point cloud retrieved from the database. The depth contour corresponding to DRVAL2=10.0 m in a square area around the AutoNaut.

Figure 4.5: The ENC extraction and representation method developed by Lauvås and used in this thesis.

Acronym	**Object class**	**Definition**
BCNISD	Beacon, isolated danger	A beacon erected on an isolated danger of limited extent, which has navigable water all around it.
BCNLAT	Beacon, lateral	A lateral beacon is used to indicate the port or starboard hand side of the route to be followed.
BOYCAR	Buoy, cardinal	A cardinal buoy is used in conjunction with the compass to indicate where the mariner may find the best navigable water.
BOYINB	Buoy, installation	An installation buoy is used for loading tankers with gas or oil.
BOYISD	Buoy, isolated danger	An isolated danger buoy is a buoy moored on or above an isolated danger of limited extent, which has navigable water all around it.
BOYLAT	Buoy, lateral	A lateral buoy is used to indicate the port or starboard hand side of the route to be followed.
BOYSAW	Buoy, safe water	A safe water buoy is used to indicate that there is navigable water around the mark.
BOYSPP	Buoy, special purpose/general	A special purpose buoy is primarily used to indicate an area or feature, the nature of which is apparent from reference to a chart, Sailing Directions or Notices to Mariners.
COALNE	Coastline	The line where shore and water meet.
DEPARE	Depth area	A water area whose depth is within a defined range of values.
DEPCNT	Depth contour	A line connecting points of equal water depth.
UWTROC	Underwater/awash rock	A concreted mass of stony material or coral which dries, is awash or is below the water surface.
WRECKS	Wreck	The ruined remains of a stranded or sunken vessel which has been rendered useless.

Table 4.1: A selection of relevant S-57 objects for the AutoNaut.

Chapter 5

Collision and Grounding Avoidance System

In this section, the collision avoidance system (CAS) and the integrated anti-grounding system, including environmental factors, will be presented. The CAS used in the AutoNaut is based on the simulation-based model predictive control (SB-MPC) system presented in Johansen *et al.* [1] and further implemented in Hagen [2], [3].

The anti-grounding system is developed through the work of this thesis. It is based on ENCs and the same SB-MPC concept as the collision avoidance system. It is a reactive anti-grounding system meant to make sure that the AutoNaut is aware of and avoids any static obstacles when unexpected course changes occur, and the originally planned path no longer is viable, for example, when avoiding dynamic obstacles.

5.1 MPC

The concept in Johansen *et al.* [1] is based on model predictive control (MPC). MPC is a control method where a finite-horizon open loop optimal control problem is solved at each time step [54]. This optimization will give an optimal control sequence where the first control action is implemented. The MPC can take the predicted future state of the system into account when deciding which control actions to take in the present. The principle of MPC is shown in Figure 5.1.

In a CAS, the MPC is advantageous as it can compute an optimal trajectory for the vessel to follow, considering all the complex constraints present in the situation. These constraints include the motion of other obstacles, environmental forces affecting the maneuverability of the vessel, and uncertainties that need to be accounted for. In order to formulate an optimization problem and assessing the system's performance, a cost function representing the hazard is used. This cost function will be minimized to find an optimal path.

The method used to implement MPC for the CAS, which exploits the benefits

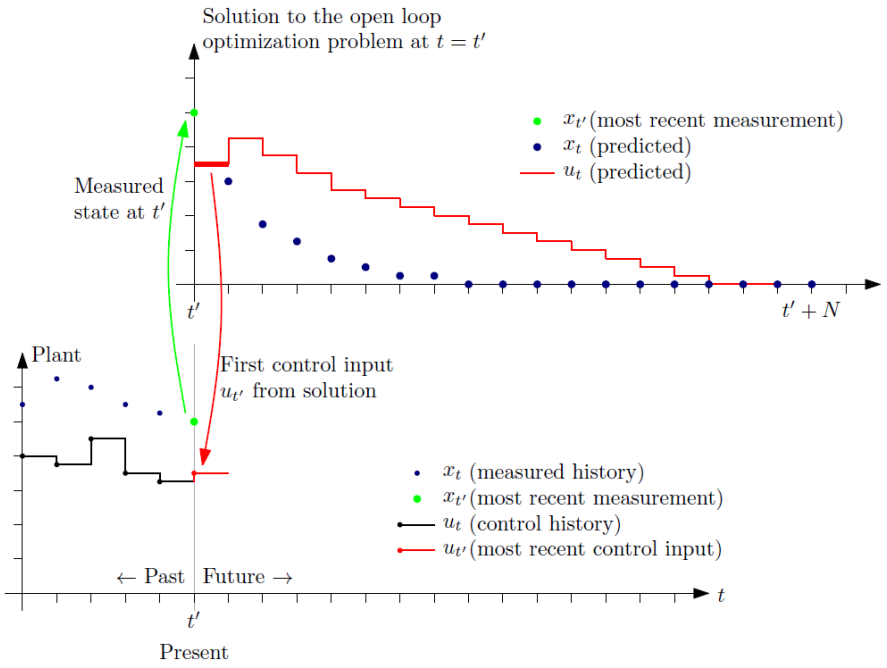


Figure 5.1: The MPC principle [54].

and reduces the issues of MPC, has a small number of control behaviors that can be selected as the optimal control action. These control behaviors can consist of a set of course offsets and propulsion commands. According to Johansen *et al.*, this approach is very effective, giving high performance and low implementation complexity.

The CAS and the integrated anti-grounding system are implemented as a hazard minimizing problem that is finite horizon, with finite scenarios over a finite number of control behaviors. This optimization problem is solved based on the updated information at regular intervals. A simulation model of the vessel is used to predict the hazard of choosing a given control behavior, considering the ship dynamics and other factors affecting the resulting behavior of the vessel.

5.2 The Collision and Grounding Avoidance Algorithm

In this section, the concept and theory of the CAS algorithm from Johansen *et al.* [1] and of the integrated anti-grounding algorithm, including environmental factors, developed in this thesis are described. As shown in the block diagram in Figure 5.2, the collision and grounding avoidance system receives waypoints and planned speed (if available) from the mission planner and information about the obstacles from the AIS sensor and the ENCs. Then, the system evaluates the

hazard for the given situation, sends modified course and propulsion commands to the autopilot, and initiates the alarms if necessary.

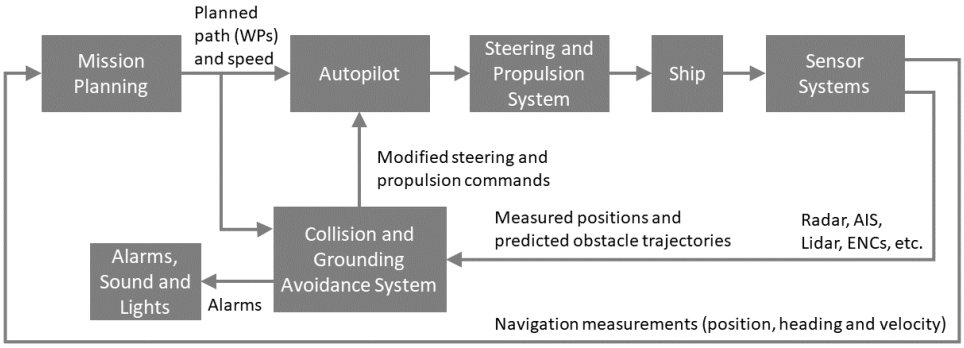


Figure 5.2: Block diagram illustrating the information flow between the main modules in the system. Modified version of Fig. 1 in [1].

Figure 5.3 shows the architecture of the collision avoidance control algorithm as described in [1]. The trajectories of the obstacles are predicted based on the data from the sensors at the same time as the ship's own trajectory is simulated. Then the hazard is evaluated for each scenario, and the control behavior that results in the minimum worst-case hazard is selected.

5.2.1 Control behaviors

As long as the hazard is low, the vessel will keep to the nominal scenario, which is to follow the desired path using the guidance algorithm. If the hazard increases above a given level, the control behavior that results in the least hazardous behavior is selected. This set of control behaviors should include as many alternatives as the limit of computational power allows. In [1] the following minimum set of alternatives is proposed:

- Course offset at -90, -75, -60, -45, -30, -15, 0, 15, 30, 45, 60, 75, 90 degrees
- Keep speed (nominal propulsion), slow forward, stop and full reverse propulsion commands.

For the AutoNaut, the speed is not used as an alternative control behavior, only the course offsets. The AutoNaut has limited control over its speed. Since it is wave-propelled, it depends on the state of the ocean, and it has yet no implementation to predict how the speed will be in the future. So, for the AutoNaut, the current speed is used when the future state is predicted for the different control behaviors, and only the optimal course offset is computed in the CAS algorithm. As a result, the possible control behaviors consist of the course offsets included in χ_{ca}:

$$\chi_{ca} = [-90^\circ, -75^\circ, -60^\circ, -45^\circ, -30^\circ, -15^\circ, 0^\circ, 15^\circ, 30^\circ, 45^\circ, 60^\circ, 75^\circ, 90^\circ]$$

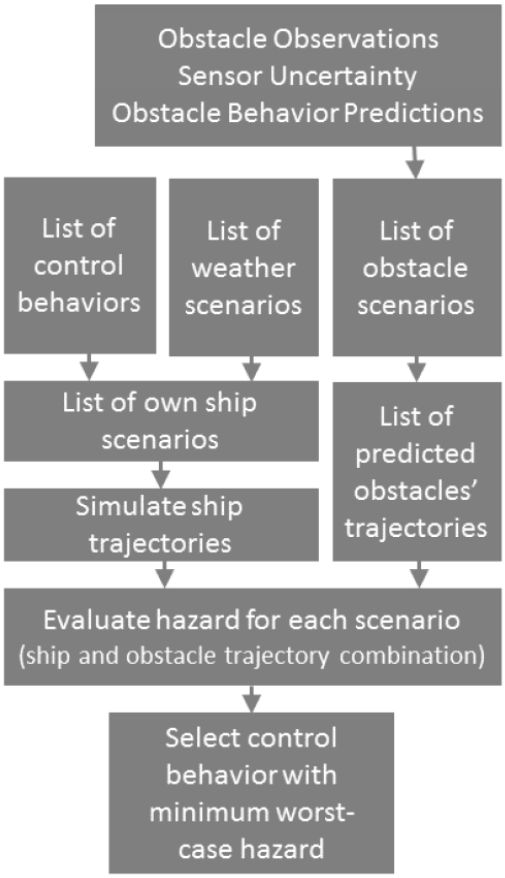


Figure 5.3: Summary of the collision avoidance control algorithm [1].

The CAS finds the optimal control behavior and gives a course angle offset χ_{ca}^k to the autopilot of the AutoNaut. The total commanded course angle is then $\chi_c = \chi_{LOS} + \chi_{ca}^k$. If there is no collision risk present in the situation and the hazard is zero, the course offset will be set to zero, and the LOS path following will continue as normal. The PI course controller then computes the commanded rudder angle with controller gains K_p and K_i.

$$\delta = K_p(\chi_c - \chi) + K_i \int_0^t (\chi_c - \chi) dt \tag{5.1}$$

5.2.2 Prediction of the own ship trajectory

The trajectory of the own ship needs to be predicted to see how it will respond to the control behaviors. In [1] the following 3-DOF ship model for surface vessels

is used, where roll, pitch and heave motions are neglected.

$$\begin{aligned} \dot{\eta} &= R(\psi)\nu + \nu_c \\ M\dot{\nu} + C(\nu)\nu + D(\nu)\nu &= \tau + R(\psi)^T \tau_w \end{aligned} \tag{5.2}$$

where $\eta = [x, y, \psi]^T$ is position and heading angle in the earth-fixed frame and $\nu = [v_x, v_y, r]^T$ is surge, sway and yaw in the body-fixed frame. The general equations of motion for a surface vessel are described in Section 2.2, where a full description of the matrices and vectors used in these equations can be found.

In [2] a simpler model, where environmental forces are neglected, is proposed to use for the AutoNaut:

$$\begin{aligned} \dot{\eta} &= R(\psi)\nu \\ M\dot{\nu} + C(\nu)\nu + D(\nu)\nu &= \tau \end{aligned} \tag{5.3}$$

A straight-line trajectory prediction is also proposed in [2] because it is less computationally heavy and improves the run-time of the algorithm. The computational complexity must not be too high in order for the system to keep its capability of acting in real-time. Using the straight-line trajectory prediction is assumed to be sufficient for the AutoNaut as it is a relatively small vessel with fast dynamics. Therefore, in the implementation of the CAS algorithm for the AutoNaut, this straight-line trajectory prediction is used.

5.2.3 Prediction of the dynamic obstacle trajectory

The prediction of the dynamic obstacles' future trajectories is done using straight line trajectories. This is a simple short-term prediction.

$$\begin{aligned} \bar{\eta}_i^{lat}(t) &= \hat{\eta}_i^{lat} + k_{lat}\hat{v}_i^N(t - \tau_i) \\ \bar{\eta}_i^{long}(t) &= \hat{\eta}_i^{long} + k_{long}\hat{v}_i^E(t - \tau_i) \end{aligned} \tag{5.4}$$

Here the constants k_{lat} and k_{long} are constants that convert from meters to degrees. The future point in time is t and the time of the last observation is τ_i.

For the AutoNaut, a straight-line prediction of the dynamic obstacles' trajectories is a good approximation because it is intended to operate far out at the sea where quick course changes are rare, and because it gets updated position, heading, and velocity from the obstacles every 5th second. Thus, it is capable of reacting to changes in the obstacles' courses.

5.2.4 COLREGS compliance

The CAS computes the expected collision hazard and degree of COLREGS compliance for a future point in time, for each control behavior scenario, by using information about its own and the obstacles' future trajectories. This information is illustrated in Figure 5.4. Definitions that help evaluate the COLREGS compliance of the vessel are as follows:

- **CLOSE**: An obstacle is close to the own vessel if $d_{0,i}^k(t) \leq d_i^{cl}$ where $d_{0,i}^k(t)$ is the distance between own vessel and obstacle i for scenario k. This distance d_i^{cl} is the smallest distance where the COLREGS rules will apply.

- **OVERTAKEN**: The own vessel is overtaken by the obstacle if the obstacle has higher speed, is close to own vessel and

$$\vec{v}_0^k(t) \cdot \vec{v}_i(t) > \cos(68.5°)|\vec{v}_0^k(t)||\vec{v}_i(t)| \tag{5.5}$$

  where $\vec{v}_0^k$ is the predicted velocity of own ship and $\vec{v}_i$ is the predicted velocity of the obstacle with index i in scenario k.

- **STARBOARD**: The obstacle is starboard of own vessel if the bearing angle of $\vec{L}_i^k(t)$ is larger than the heading angle of the vessel. $\vec{L}_i^k$ is a unit vector in the LOS direction from own ship to the obstacle with index i in scenario k.

- **HEAD-ON**: The obstacle is head-on if it is close to own vessel, and the obstacle speed $|\vec{v}_i(t)|$ is not close to zero and

$$\begin{aligned} \vec{v}_0^k(t) \cdot \vec{v}_i(t) &< -\cos(22.5°)|\vec{v}_0^k(t)||\vec{v}_i(t)| \\ \vec{v}_0^k(t) \cdot \vec{L}_i^k(t) &> \cos(\phi_{ahead})|\vec{v}_0^k(t)| \end{aligned} \tag{5.6}$$

- **CROSSED**: The obstacle is crossed if it is close to own vessel and

$$\vec{v}_0^k(t) \cdot \vec{v}_i(t) < \cos(68.5°)|\vec{v}_0^k(t)||\vec{v}_i(t)| \tag{5.7}$$

  The value of the angle ϕ_{ahead} is selected to define where an obstacle is said to be ahead of the own ship.

5.2.5 Risk factor and cost function for collision

Risk factor

The risk factor for collision with obstacle i is defined as

$$\mathcal{R}_i^k(t) = \begin{cases} \frac{1}{|t-t_0|^p}\left(\frac{d_i^{safe}}{d_{0,i}^k(t)}\right)^q, & \text{if } d_{0,i}^k(t) \leq d_i^{safe} \\ 0, & \text{otherwise} \end{cases} \tag{5.8}$$

where t_0 is the current time and $t > t_0$ is the time of prediction. The index k denotes a scenario associated with a single course offset belonging to $\chi_{ca} = [-90°, +90°]$. The COLREGS rule 16 states the importance of keeping a safe distance to other vessels. The distance d_i^{safe} and the exponent $q \geq 1$ must be selected large enough to follow these rules. In order to prioritize avoiding collisions that are close in time over those that are further into the future, the exponent $p \geq 1/2$ gives an inverse proportionality with time until occurrence of the event. Thus, the collision risk factor is higher for events close in time than for events in the more distant future.

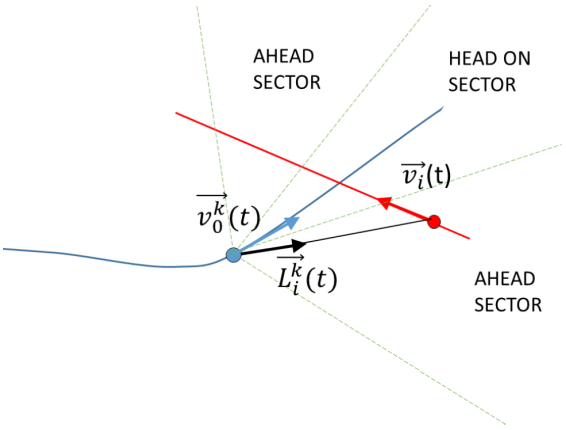


Figure 5.4: The main information used for hazard evaluation at a given future time t in scenario k. The blue dot represents the predicted position of the own ship at future time t and the red dot represents the predicted position of the obstacle vessel [1].

Cost function

The cost associated with collision with an obstacle i is chosen to be

$$C_i^k(t) = K_i^{coll} |\vec{v}_0^k(t) - \vec{v}_i^k(t)|^2 \tag{5.9}$$

where K_i^{coll} is the cost of collision parameter. The kinetic energy of the obstacle is included, in order to minimize the consequences of a collision if a complicated situation would occur, where a collision is unavoidable.

5.2.6 Risk factor and cost function for grounding

Risk factor

The risk factor for grounding associated with each course offset k is defined as

$$\mathcal{R}_G^k(t) = \begin{cases} \frac{1}{|t-t_0|^{Pg}} \left(r_c^k + r_+^k + r_-^k\right)^q, & \text{if } d_c^k(t) \leqslant d_k^{safe_G} \\ & \text{or } d_+^k(t) \leqslant d_k^{safe_G} \\ & \text{or } d_-^k(t) \leqslant d_k^{safe_G} \\ 0, & \text{otherwise} \end{cases} \tag{5.10}$$

where

$$\begin{aligned} r_c^k &= \frac{d^{safe_G}}{d_c^k(t)} \\ r_+^k &= \frac{d^{safe_G}}{d_+^k(t)} \\ r_-^k &= \frac{d^{safe_G}}{d_-^k(t)} \end{aligned} \tag{5.11}$$

The distance d_c^k is the distance between the own vessel and the grounding obstacle in the center direction $\chi_{LOS} + \chi_{ca}^k$, which is in the direction of the course offset that is currently being evaluated in scenario k. The distances d_-^k and d_+^k are the distances from land found in the $-15°$ and $+15°$ direction, $\chi_{LOS} + \chi_{ca}^k \pm 15°$, and to the center path, as defined in Figure 5.5. This is done to ensure that the ASV keeps enough distance to land on the side as well as forwards. Here, $-15°$ and $+15°$ are used, since this grounding data already is available for these courses. However, this can be extended to include a wider range of directions to ensure that the distance to land in the range -90°to +90°is being kept larger than the safe distance to land. Further, t_0 is the current time, and $t > t_0$ is the time of prediction; d^{safe_G} is the minimum safe distance to land.

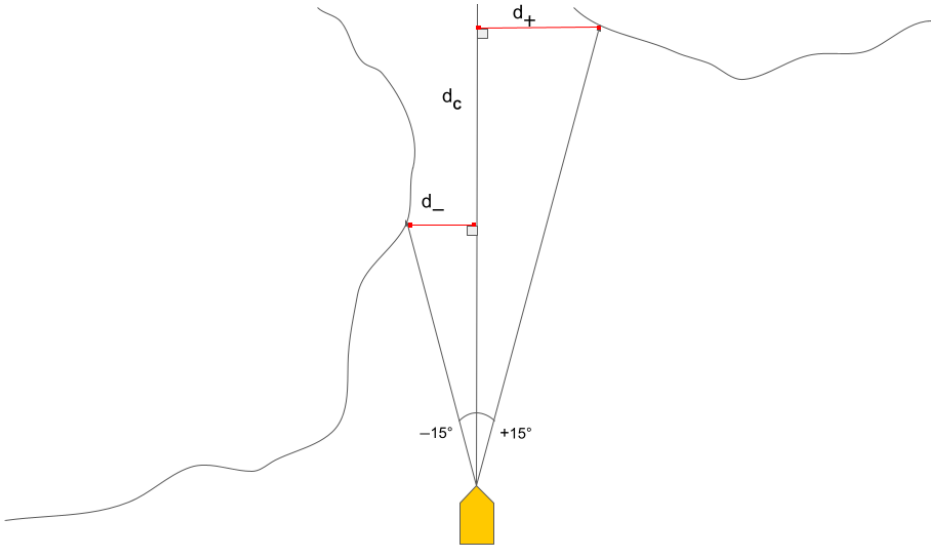


Figure 5.5: Definitions of the distances used in the anti-grounding risk function.

For collision avoidance, the exponent factor p is used to prioritize events that are close in time over those that are further in the future and because there is an uncertainty of where the dynamic obstacle will be located in the future. For anti-grounding, the obstacles and their locations are static, meaning the last argument is no longer valid. Therefore, the exponent p_g should have a smaller value for anti-grounding than for collision avoidance.

Cost function

The cost associated with grounding is chosen to be

$$C_G^k(t) = K_{ground} + K_{env}E^k \tag{5.12}$$

where K_{ground} is the cost of grounding parameter, $K_{env} = [k_1, k_2, k_3, k_4, k_5]$ is a vector containing the weights for each environmental factor, and

$$E^k = [B, Z, WH, W(\chi_{ca}^k), C_O(\chi_{ca}^k)]^T$$

is the vector containing the environmental factors. The velocities of the obstacle and of the AutoNaut are not included, because the grounding obstacles are static and the velocity of the AutoNaut is low.

5.2.7 Environmental Factors for Anti-Grounding

Many of the situations that the AutoNaut will encounter in the fjords and along the coast consist of multiple obstacles, both dynamic and static, and occur during varying conditions. In order to safely maneuver these situations, the AutoNaut needs to make informed and well-balanced decisions and find a solution that minimizes the hazard and, thus, potential damage to the AutoNaut and its surroundings.

The optimal solution in such complex scenarios is often dependent on the environmental conditions. If the sea is calm, there is no wind, waves, or current, and the seabed in the ASV's surrounding coasts mainly consists of sandbanks, then going closer to land should not lead to a dangerous scenario, i.e., with associated high cost and hazard. Therefore, in this situation, the main concerns would be to avoid colliding with dynamic obstacles with a clear margin and comply with the rules of COLREGS.

However, if the sea is rough, the wind and currents strong, the waves high, and there are mainly rocky islands around, going close to land will be more dangerous and the consequences of a grounding higher. Then, avoiding ground should probably be a higher priority than complying with COLREGS, and in some cases, even higher than avoiding collisions with other vessels. This is especially the case if the obstacle vessels are small and keep a very low speed or are moored (e.g., fishing vessels).

In order to differentiate between these kinds of situations, environmental factors should be taken into account when evaluating a scenario and deciding the optimal action for the AutoNaut to take.

The environmental factors considered in this work are:

- Bathymetry B
- Heave displacement Z
- Wave height WH
- Wind W
- Ocean Currents C_O

The value of each environmental factor is not crucial on its own; it is the total picture of the state of the environment that is important. If the conditions are harsh, all factors are normally affected and have a higher value than if the conditions are calm. The environmental factors are intended to reflect the degree of danger caused by the environmental conditions in the situation.

Bathymetry

The bathymetry factor represents the amount of damage that is expected if the ASV hits ground. The bathymetric features might differ greatly between fjords

and coastal areas of Norway. In this work, it is decided that a steep rocky shelf will get a higher cost than a flat sandbank because the ASV would suffer greater damage if it collides with the former.

Information about the bathymetry and the seabed can be extracted from the ENCs. The ENCs contain the object Seabed Area (SBDARE), which contains the attribute Nature of Surface (NATSUR). NATSUR describes the surface material of the seabed and categorizes it into one or several categories, including mud, silt, sand, stone, rock, coral, and boulder. Although this information can be quite sparse, it gives an indication of the seabed conditions.

In areas with little information about the bathymetry, it is possible to use ENCs to compute the slope of the coastal shelf based on depth information and horizontal distance between the depth queries. The ENCs contain a Sounding object (SOUNDG), which are depth measurements that can be used to calculate the steepness of the underwater terrain slope towards the coastline. Despite the lack of accurate information, it is possible to assume that steep shelves are likely to be made of rocks and stony materials.

The full bathymetric information will give an idea of what the underwater terrain looks like and how dangerous it would be for the AutoNaut to crash into it.

Heave displacement

Heave is the linear motion in z_b-direction as described in Section 2.1. A large displacement in the heave of the AutoNaut will give a higher environmental cost because it indicates a rough sea, which increases the danger of grounding. The heave displacement is measured directly in the AutoNaut by the GPS. Alternatively, the ASV's pitch angle could be used.

Wave Height

High waves will give a higher environmental cost because it will make the water shallower in the troughs than indicated in the ENCs, and give added force to a grounding. The wave height is measured on the AutoNaut by the GPS and is found in weather forecasts.

Wind

The wind factor is a function of the course offset χ_{ca}^k:

$$W(\chi_{ca}^k) = V_w \max\left(0, \cos(\chi + \chi_{ca}^k - \beta_{V_w})\right)^2 \tag{5.13}$$

where V_w is the absolute wind speed and β_{V_w} is the wind direction as defined in Figure 5.6.

The function is inspired by the ad hoc risk cost function from Blindheim *et al.* [24], see Equation (1.1), and is designed to give a higher cost to the course

offsets that coincide with the wind direction. This is due to the increased risk of the AutoNaut drifting in the wind direction towards a grounding obstacle. Keeping a larger safety margin than normal to ground is therefore necessary in these cases. Thus, the cost is highest when the wind blows in the same direction as $\chi + \chi_{ca}^{k}$. The exponent is added in order to achieve a higher cost differentiation between the course offsets, so the difference between the offsets is large enough for the system to choose one over the other.

The wind speed and direction are measured in real-time on the AutoNaut by the Weather Station and can also be found in weather forecasts.

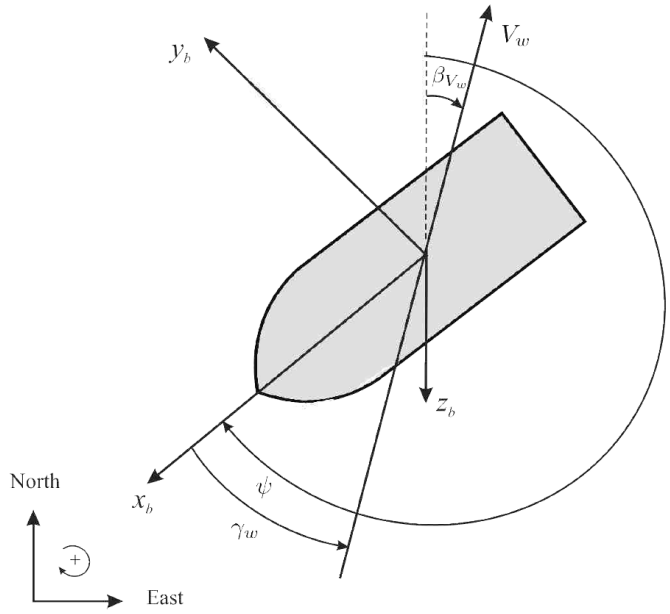


Figure 5.6: Definition of the wind speed V_w, wind direction β_{V_w} and wind angle of attack γ_{rw} relative to the bow [39].

Ocean Current

The ocean current factor C_O is defined similarly as the wind factor W and is also a function of the course offset χ_{ca}^{k}.

$$C_O(\chi_{ca}^{k}) = V_c \max\left(0, \cos(\chi + \chi_{ca}^{k} - \beta_{V_c})\right)^2 \tag{5.14}$$

where V_c is absolute current speed and β_{V_c} is the horizontal current direction.

The AutoNaut carries an acoustic Doppler current profiler (ADCP) that can measure current speed and direction up to 60 m depth. The currents can also be found in the weather forecast.

Total environmental cost

The total environmental cost for a course offset χ_{ca}^k is:

$$K_{env}E^k = k_1B^2 + k_2Z^2 + k_3WH + k_4W(\chi_{ca}^k) + k_5C_O(\chi_{ca}^k) \tag{5.15}$$

The weights in K_{env} can be adjusted to make some factors have a greater impact on the total cost than others, depending on the system and the situation. The impact of large values of the bathymetry factor and the heave displacement factor is emphasized by adding an exponent.

The bathymetry, heave displacement, and wave height factor will give a constant cost for each course offset and will therefore only affect the decision between collision or grounding. The wind and current factor, on the other hand, are functions of the course offset and will give different costs for different course offsets, depending on the direction of the wind and the current. Therefore, these factors will also affect which course offset is optimal in pure anti-grounding.

The total environmental cost is added to the grounding cost function, Equation (5.12), and will only have an impact on the total hazard $\mathcal{H}^k(t_0)$, in Equation (5.19), if the grounding risk factor $\mathcal{R}_G^k(t)$, Equation (5.10), for the given course offset is nonzero, i.e., there are grounding obstacles close by in the direction of the course offset.

5.2.8 Cost of deviating from nominal course

The term $f(\chi_{ca}^k)$ is added to the hazard function $\mathcal{H}^k(t_0)$ to ensure predictability of the vessel's motion and ensure compliance with several COLREGS rules.

$$f(\chi_{ca}^k) = k_\chi(\chi_{ca}^k - \chi_{ca,last}) + \Delta_\chi(\chi_{ca}^k - \chi_{ca,last}) \tag{5.16}$$

where k_χ and Δ_χ are functions of the difference between the current course offset and the last course offset $\delta_{\chi_{ca}} = \chi_{ca}^k - \chi_{ca,last}$.

$$k_\chi\left(\delta_{\chi_{ca}}\right) = \begin{cases} k_{\chi_P}\chi_{ca}^2, & \text{if } \delta_{\chi_{ca}} < 0, \text{ turn to port} \\ k_{\chi_{SB}}\chi_{ca}^2, & \text{if } \delta_{\chi_{ca}} > 0, \text{ turn to starboard} \end{cases} \tag{5.17}$$

$$\Delta_\chi\left(\delta_{\chi_{ca}}\right) = \begin{cases} k_{\Delta\chi_P}\delta_{\chi_{ca}}^2, & \text{if } \delta_{\chi_{ca}} < 0, \text{ turn to port} \\ k_{\Delta\chi_{SB}}\delta_{\chi_{ca}}^2, & \text{if } \delta_{\chi_{ca}} > 0, \text{ turn to starboard} \end{cases} \tag{5.18}$$

where k_{χ_P}, $k_{\chi_{SB}}$, $_{\Delta\chi_P}$ and $_{\Delta\chi_{SB}}$ are tuning parameters defined in Table 6.1.

The $f(\chi_{ca}^k)$ term will favor a straight-line trajectory, as stated by COLREGS rule 17, and affects the priority of staying on the nominal path with the nominal course. In addition, it differentiates between course offsets to starboard and to port, and can thus penalize course offsets to port harder than to starboard, which is in compliance with COLREGS rules 14, 15, and 17. In total, this will make the vessel's actions more predictable for other vessels, which is important in order to avoid potential collision scenarios.

5.2.9 Hazard evaluation criterion

The hazard associated with scenario k, at time t_0, is

$$\begin{aligned}\mathcal{H}^k(t_0) = \max_i \max_{t \in D(t_0)} \left(C_i^k(t)\mathcal{R}_i^k(t) + C_G^k(t)\mathcal{R}_G^k(t) + \kappa_i \mu_i^k(t) \right) \\ + f\left(\chi_{ca}^k\right)\end{aligned} \tag{5.19}$$

It consists of the collision hazard, the grounding hazard, the cost for violating COLREGS, and the cost of deviating from the nominal course. The term for grounding hazard $C_G^k(t)\mathcal{R}_G^k(t)$ is replacing the $g(\cdot)$ term used in Johansen *et al.* [1].

The term $\kappa_i \mu_i^k(t)$ is the cost of not complying with COLREGS, where κ_i is the tuning parameter.

5.2.10 Control decision

The selected control behavior at time t_0 among the scenarios $k \in 1, 2, ..., N$ is the one with minimal hazard $\mathcal{H}^k(t_0)$

$$k^*(t_0) = \arg\min_k \mathcal{H}^k(t_0) \tag{5.20}$$

This minimization is done at regular intervals, e.g., every 5 seconds, to consider new information from the sensors about the obstacles. In [1] it is emphasized that this optimization is deterministic and guarantees that the global minimum is found after a pre-defined number of cost function evaluations.

5.3 Implementation

The implementation of the system is described in this section. The code developed in this thesis can be found in the following repository: `https://github.com/adallolio/dune/tree/AutoNaut-CAS-Thea`. The repository contains the full DUNE software for the AutoNaut, including the original collision avoidance system and the new anti-grounding system.

The collision and grounding avoidance system is implemented in DUNE as a separate task. In the configuration file of the AutoNaut (ntnu-autonaut.ini), the parameters needed for the different tasks are defined and sent to the tasks as IMC messages. The parameters needed for the collision and grounding avoidance system and their values are shown in Table 6.1. The system is divided into two subsystems of anti-grounding and collision avoidance, and is continuously monitoring the surroundings of the AutoNaut. If an obstacle is detected, the corresponding dynamic or static obstacle flag is set to true, and the SB-MPC algorithm will run with the information obtained in the task. The environmental factors are added as a part of the anti-grounding system.

In this section, the implementation of the anti-grounding system will be described in greater detail than the implementation of the collision avoidance system, as the latter is similar to the description in [1] and [2], while the anti-grounding system has been developed through this master's thesis.

5.3.1 Collision avoidance system

The collision avoidance system receives AIS information about surrounding vessels as an IMC message from the AIS sensor. This information includes position and static and voyage-related data. The distance between the AutoNaut and the obstacle is calculated, and the dynamic obstacle flag is set to true if an obstacle vessel is inside the Maximum Obstacle Surveillance Range (`obst_range`). Information about the obstacle vessels is sent to the SB-MPC algorithm as an argument in the `getBestControlOffset` function. In the SB-MPC algorithm, the concept described in Section 5.2, Johansen *et al.* [1], and Hagen [2], [3] is followed.

5.3.2 Anti-grounding system

The anti-grounding system is checking for static obstacles every $t_g = 60$ s. There is no need for higher frequencies given the low ASV speed. In this implementation of the anti-grounding system, the S-57 object DEPARE (Depth area) is the only object considered. This means that only grounding obstacles are considered, but the system can easily be extended to include other static obstacle types.

When the system checks the surroundings for grounding obstacles, a query is made to the database containing the ENC data point clouds, as described in Section 4.3. It asks for all depth area data points that are located closer than the 5000 m (Contours Distance) square area around the AutoNaut, and that have its deepest depth (DRVAL2) equal to the Minimum Safe Depth Value parameter. These points are stored in a depth vector. An example of retrieved points is shown in Figure 5.7a, where the grey points are depth area data points, the AutoNaut is shown in yellow, and its next waypoint is in green.

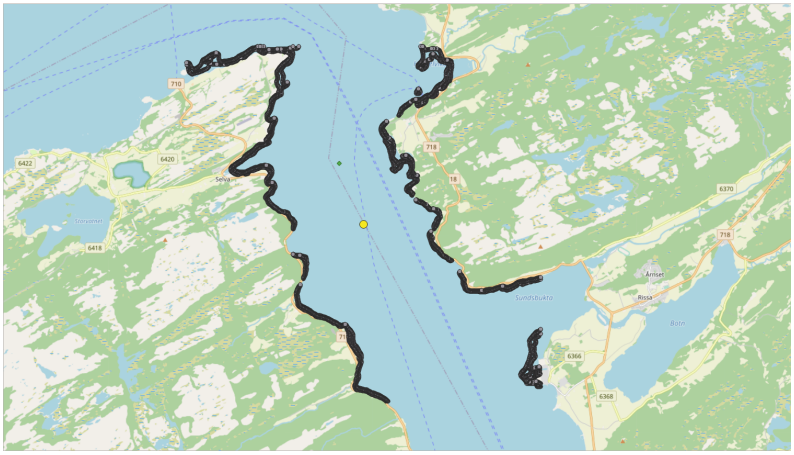
Then, the depth vector is iterated through in order to find the point that is the closest to the AutoNaut for each of the directions corresponding to the course offsets. A margin of $\pm 5°$ is added to the course offsets to ensure that a data point is found and that the system is robust. This results in a new vector containing the closest data point in each course offset direction, shown as red points in Figure 5.7b.

If a data point is closer to the AutoNaut than the SB-MPC surveillance range, here set to 2000.0 m, the point is defined as a static obstacle, and the flag for static obstacles is set to true. The grounding cost, including environmental cost, is then calculated. When the static obstacle flag is true, the SB-MPC algorithm with the function `getBestControlOffset` is run, sending the information about the static obstacles as argument.

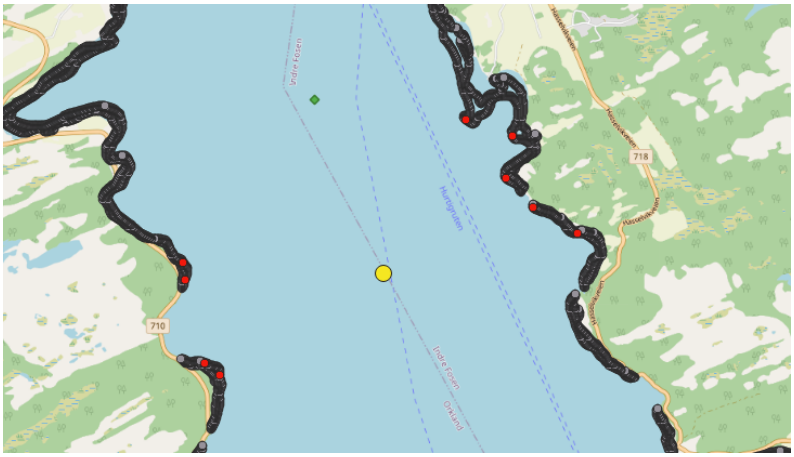
In the SB-MPC algorithm, calculations of the risk and cost of grounding are done for each course offset as described in Section 5.2.6. The total hazard for collision and grounding is then calculated, see Equation (5.19), and the optimal course offset in the given situation is found and returned to the task. This optimal course offset is added to the course angle from the LOS guidance law, and the final desired course angle is sent to the autopilot that will control the AutoNaut to follow the course.

5.3.3 Environmental factors

The total environmental cost is added to the grounding cost, see Equation (5.12). In this implementation, constant simulation values are given to the factors. In a real-life situation, the environmental factors would come directly from sensors on the AutoNaut, or from a weather forecast. Due to a lack of access to ENC data, the bathymetry factor has not been implemented properly yet but has instead been given a constant value of 1.



(a) ENC data points retrieved from the database in a square area round the AutoNaut.



(b) The red points show the ENC data points closest to the AutoNaut in each course offset direction.

Figure 5.7: ENC depth area (DEPARE) data points retrieved from the database.

5.4 Grounding Hazard Plots

The anti-grounding theory presented in this chapter is validated in the following section. Several plots showing the evolution of the grounding hazard have been made. The grounding hazard is extracted from Equation (5.19) and plotted as follows

$$\mathcal{H}_G(t_0) = \max_{t \in D(t_0)} (\mathcal{C}_G(t)\mathcal{R}_G(t)) \tag{5.21}$$

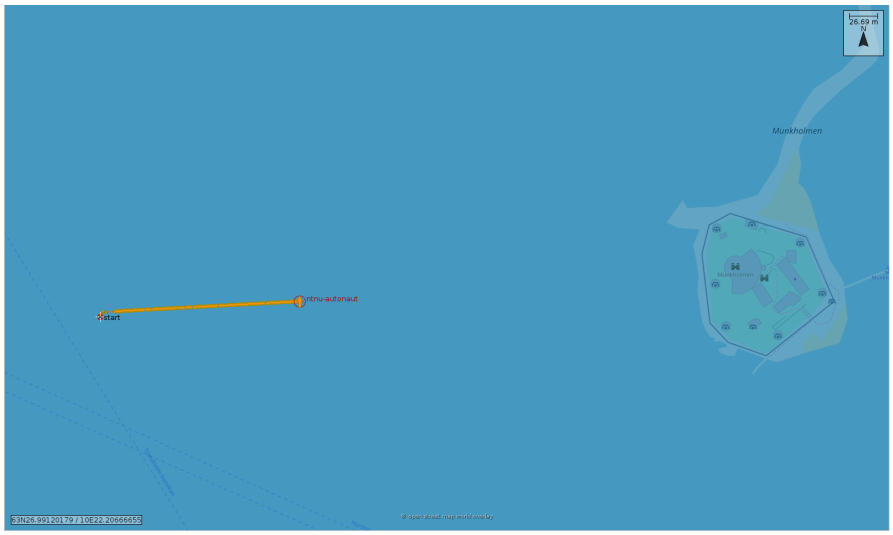
where C_G is the grounding cost function (Equation (5.12)) and R_G is the risk of grounding (Equation (5.10)). The environmental factors are included in the grounding cost function. In the plots, shown in Figure 5.8, Figure 5.9 and Figure 5.10, the evolution of the grounding hazard is plotted together with the distance to land. The cost of grounding K_{ground} is set to 100.0 for all simulations done here. The anti-grounding system updates every $t_g = 60s$, while the data is saved with an interval of 20 s, which is the reason for the steps in the plot.

In the first scenario, shown in Figure 5.8a, the AutoNaut moves straight towards land at Munkholmen. The plot in Figure 5.8b, which does not include environmental factors, shows how the grounding hazard increases when the distance to land gets smaller and smaller as the AutoNaut approaches land. Note that the magnitude of the hazard reaches up to 7×10^4 in this case.

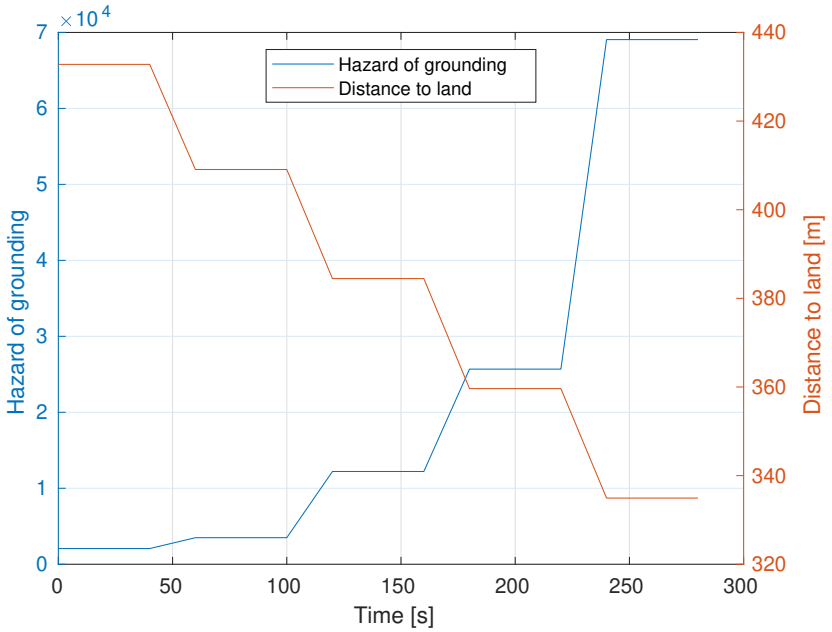
In Figure 5.9, the same scenario is tested with environmental factors added. Different values for wind and current speed and direction are tested, while the remaining values are set to constant values of bathymetry $B = 1.0$, heave $Z = 2.0$ m, and wave height $WH = 2.0$ m. The given wind and current directions are the absolute directions, and the course of the AutoNaut is kept at 87° in all simulations. The plots in Figure 5.9a and Figure 5.9b show that when the wind and current directions are aligned with the course of the AutoNaut (a), the hazard is higher than when the directions are opposite of the AutoNaut's course (b). In fact, the cost of the wind and current is zero in the latter case. Therefore, the hazard is significantly higher in (a) than in (b). In Figure 5.9c, the wind and current speeds are lower than in (b), but still, the hazard is higher in (c) because the directions align with the course of the AutoNaut. In Figure 5.9d, the wind and current come from the front sides of the AutoNaut, which results in a smaller hazard than in (a), but a similar hazard as in (c), and higher than in (b). Note that the hazard has a magnitude in the size of 10^5 in all of the four plots shown here, which is considerably higher than in the plot without environmental cost, see Figure 5.8b. This shows that the environmental factors can have a large impact on the grounding hazard.

The second scenario is shown in Figure 5.10a, where the vessel moves towards land before it turns about 160° and moves away from land. Here, environmental factors are not added, but the grounding cost is still at 100.0. The plot in Figure 5.10b shows how the grounding hazard increases when the distance to land decreases, as before. Then, when the AutoNaut turns around and heads away from Munkholmen, the grounding hazard immediately drops to zero because the distance to land in this direction is significantly larger.

The behavior shown through these plots validates that the anti-grounding system functions as desired and that the theory presented is correct. When the AutoNaut approaches land, the grounding hazard increases significantly, and when it turns away from land, the grounding hazard is reduced to zero. Adding the environmental factors results in a considerable increase in the grounding hazard, depending on the speed and direction. A higher speed gives a larger hazard, and the more the directions coincide with the course of the AutoNaut, the larger is the hazard.

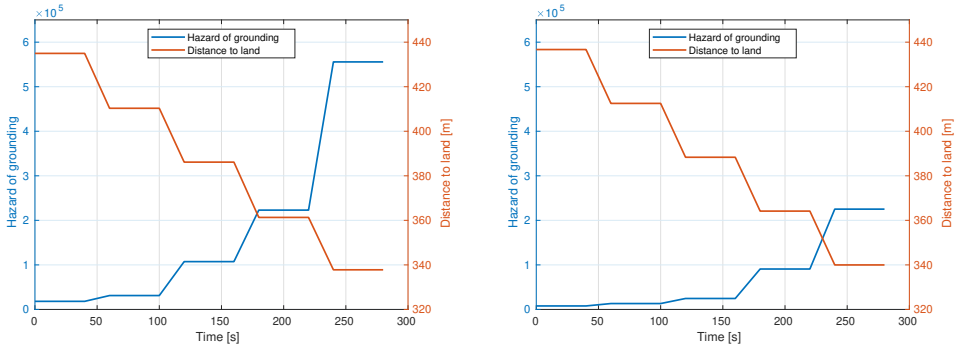


(a) The AutoNaut moves straight towards land at Munkholmen.



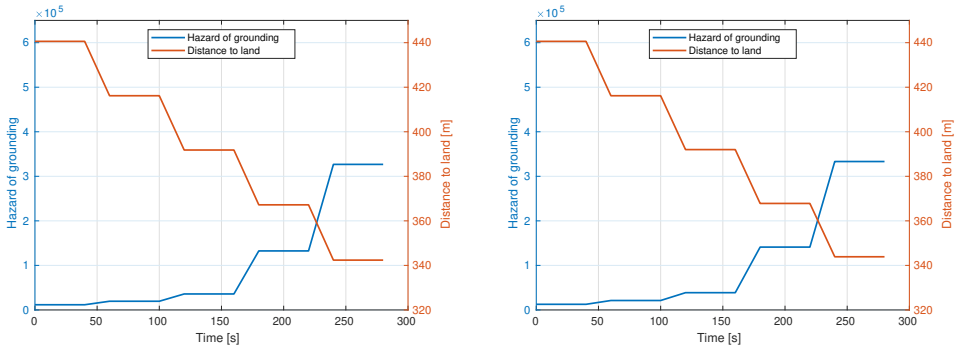
(b) Plot of grounding hazard and distance to land. The grounding cost is 100 and no environmental factors have been added.

Figure 5.8: Plot showing the evolution of the grounding hazard when the AutoNaut moves towards land.



(a) $V_W = 10.0$ m/s, $\beta_{V_W} = 90°$, $V_C = 0.3$ m/s $\beta_{V_C} = 90°$

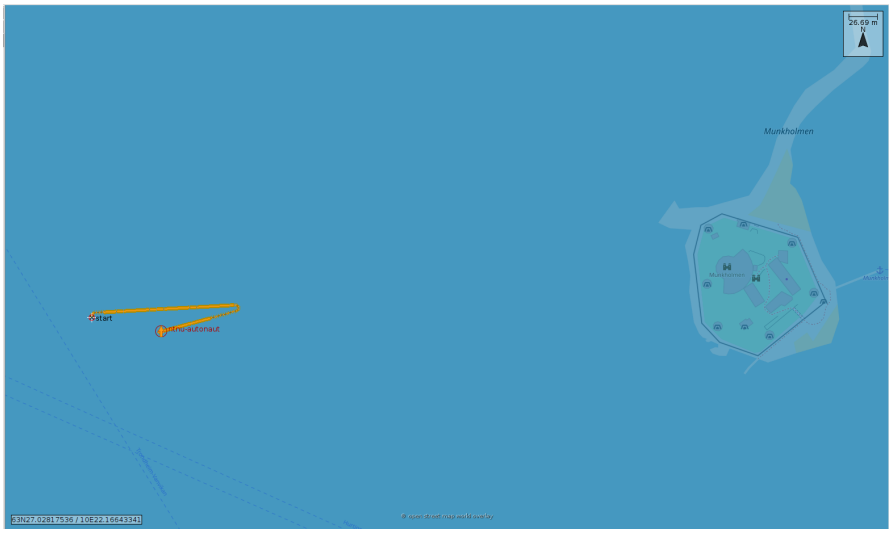
(b) $V_W = 10.0$ m/s, $\beta_{V_W} = 270°$, $V_C = 0.3$ m/s $\beta_{V_C} = 270°$



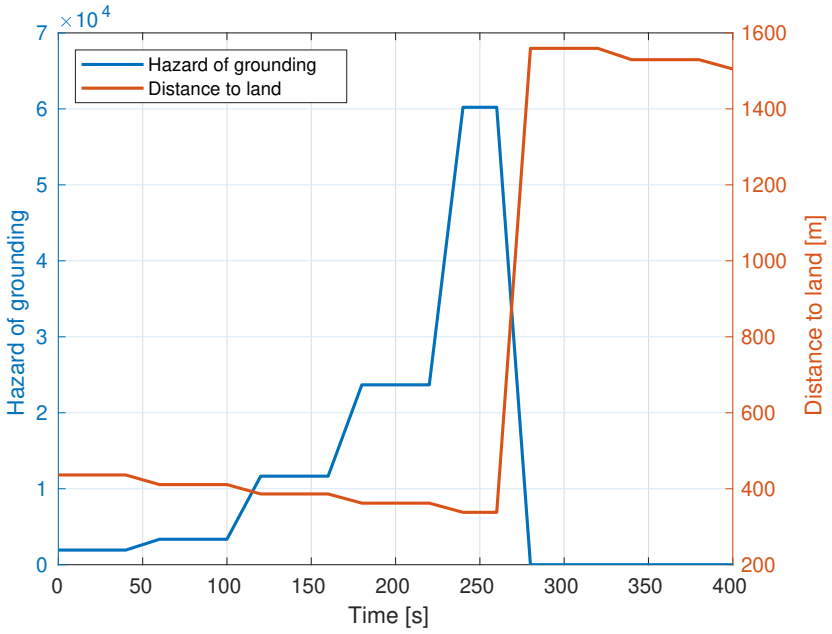
(c) $V_W = 5.0$ m/s, $\beta_{V_W} = 90°$, $V_C = 0.1$ m/s $\beta_{V_C} = 90°$

(d) $V_W = 10.0$ m/s, $\beta_{V_W} = 60°$, $V_C = 0.3$ m/s $\beta_{V_C} = 120°$

Figure 5.9: Plots of the grounding hazard and distance to land with environmental factors added, when the AutoNaut moves towards land. Different wind and current speeds and directions are tested.



(a) The AutoNaut moves straight towards land at Munkholmen, then turns away from land.



(b) Grounding hazard and distance to land. The cost of grounding is set to 100, and no environmental factors have been added.

Figure 5.10: Plot showing the evolution of the grounding hazard when the AutoNaut moves towards and then away from land.

Chapter 6

Simulation Study

In order to evaluate the performance of the collision and grounding avoidance system presented in Chapter 5, simulations have been performed in Neptus for different scenarios. The parameters used for the simulations are shown in Table 6.1. If the value of a parameter is changed between scenarios, they are specified in each scenario description. The set of possible control behaviors is as presented in Section 5.2.1. The speed of the obstacle vessel is the same in all scenarios: 6.2 m/s, while the AutoNaut normally keeps a speed between 0.20 m/s and 0.70 m/s. The prediction of the future trajectory of the AutoNaut is dependent on its speed. Since a speed prediction model is not included in the system yet and the real-time speed of the AutoNaut oscillates between 0.20 m/s and 0.70 m/s, the mean speed value of 0.45 m/s is here used for prediction.

The results from the simulations are shown as snapshots of the scenario, taken in the mission review and analysis tool in Neptus (MRA) after the simulation is finished. The AutoNaut is shown as an orange arrow, and its path is an orange line. The obstacle vessel is shown as a red arrow with a red line as its path. The arrows do not represent the course of the vessels but will always point to the north. In these scenarios, the obstacle follows a straight-line path to its waypoint. It does not have any collision avoidance system implemented and will not comply with COLREGS or take any action to avoid collisions. Thus, the AutoNaut has to handle the situation on its own. The depth area contours relevant for each scenario are shown on a map using the software QGIS.

Collision avoidance simulations have been done in Johansen *et al.* [1], Hagen [2], [3], Midjås [33] and Sæter [55], but either with different vessels or with different parameters and control behavior alternatives.

Simulations for an integrated anti-grounding system have been done in Otterholm [32], Midjås [33] and in Grande [34], but utilizing other methods for ENC-data extraction and representation, as is reviewed in Section 1.3 and Chapter 4.

Parameter	**Value**	**Description**
`obst_range`	5000 m	Maximum obstacle surveillance range
T	600 s	Prediction horizon - Simulation time
dt	5 s	Horizon time step
t_g	60 s	Timer for anti-grounding check
P	0.5	Weight on time to evaluation instant
P_G	0.05	Weight on time to evaluation instant for grounding
Q	4.0	Weight on distance at evaluation instant
d_{close}	1000.0 m	COLREGS distance
d_{init}	2000.0 m	SB-MPC surveillance range
d_{safe}	300.0 m	Minimum safe distance to vessels
d_{safe}^G		Minimum safe distance to land
$drval_2$		Minimum safe depth value
K_{coll}	1.0	Cost of collisions
K_{ground}		Cost of grounding
K_{env}		Weights on environmental factors.
κ	3.0	Cost of not complying with COLREGS
κ_{TC}	100.0	Cost of changing COLREGS
ϕ_{AH}	68.5°	Angle within an obstacle is said to be ahead
ϕ_{OT}	68.5°	Angle outside of which an obstacle will be said to be overtaking, if the speed of the obstacle is larger than the ship's own speed
ϕ_{HO}	22.5°	Angle within which an obstacle is said to be head on
ϕ_{CR}	68.5°	Angle outside of which an obstacle is said to be crossing, if it is on the starboard side, heading towards the ship and not overtaking the ship
k_χ	1.5	Cost of deviating from nominal course
$k_{\Delta\chi_{SB}}$	0.5	Cost of changing the course offset to starboard
$k_{\Delta\chi_P}$	0.9	Cost of changing the course offset to port
$k_{\chi_{SB}}$	1.5	Cost of selecting turn to starboard
k_{χ_P}	10.5	Cost of selecting turn to port
	5000.0 m	Contours Distance

Table 6.1: Parameters for simulation

6.1 Pure Anti-Grounding

A new anti-grounding system is developed in this thesis, and simulations will therefore first be done with only static obstacles present, in order to verify that the performance of the system is as desired. Normally, it can be assumed that well-known static obstacles such as land will be avoided in the mission planning phase, by the deliberate (global) obstacle avoidance system, and that waypoints will not be placed on the other side of an island or peninsula as is done here. The following pure anti-grounding simulations are therefore intended as a proof of concept to show that the anti-grounding algorithm makes the AutoNaut keep a safe distance to land.

6.1.1 Scenario 1 - Selbekken

In this scenario, shown in Figure 6.1, the waypoint is set on the other side of a peninsula close to Selbekken. The minimum safe distance to land is set to 100 m, because the waypoint is located close to land on many sides, and the AutoNaut should be able to reach it without going closer to land than the minimum safe distance indicates. The minimum safe depth is set to 10.0 m, meaning that the AutoNaut considers this depth as grounding and wants to keep the minimum safe distance to this depth contour. The 10.0 m depth contour is the outer contour shown in the map in Figure 6.1c.

When anti-grounding is not activated, the AutoNaut takes the shortest path to the waypoint, which is a straight line across land, as seen in Figure 6.1a. When anti-grounding is activated, the AutoNaut instead chooses to add course offsets to starboard and takes a path around the peninsula. It keeps a safe distance to land the whole time, as seen in Figure 6.1b. This scenario shows that the anti-grounding system performs as desired.

It should be emphasized that the anti-grounding algorithm is a part of the reactive obstacle avoidance system and is not intended for use in the path planning system. Therefore, it does not aim at finding the optimal path to a waypoint. Even though the anti-grounding system manages to find a good solution in this situation, it can not be guaranteed that this will always be the case. Because of its relatively short time horizon, it can easily get stuck in local minima and not be able to find the optimal path for a whole mission.

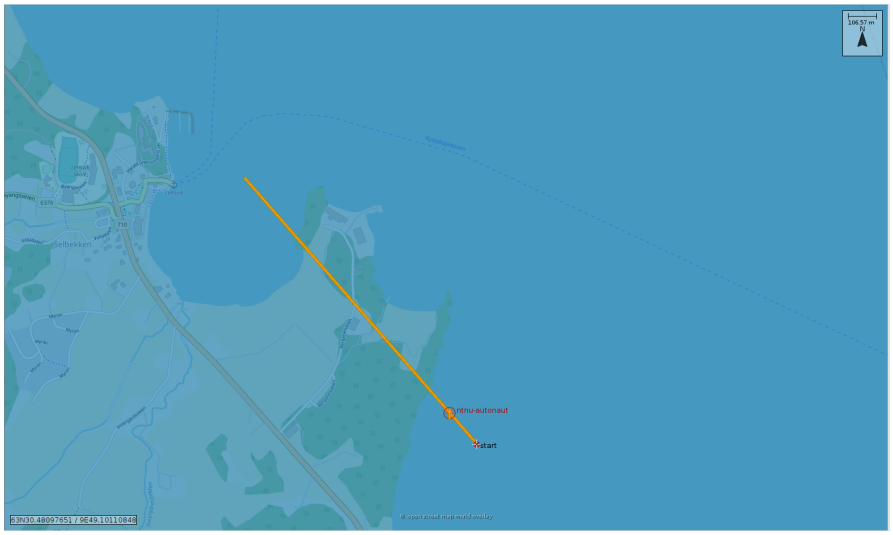
6.1.2 Scenario 2 - Munkholmen

In this scenario, shown in Figure 6.2, the waypoint is set on the other side of some shallow waters close to Munkholmen. The minimum safe distance to land is set to 300 m, as there is no land very close by the waypoint. The minimum safe depth is set to 20.0 m, and the depth contour corresponding to 20.0 m can be seen in Figure 6.2c as the outer contour.

When the anti-grounding system is not activated, the AutoNaut chooses the shortest path straight forward, crossing the area with shallow waters, as can be

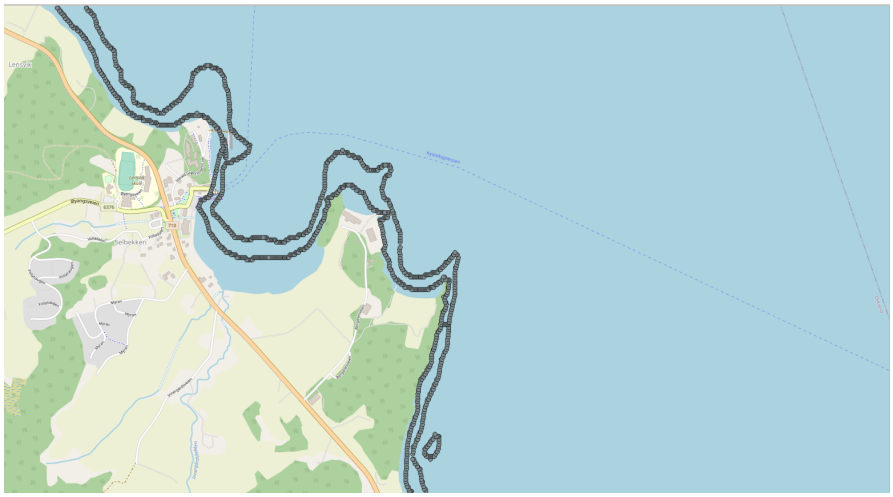
seen in Figure 6.2a.

When the anti-grounding system is activated, the AutoNaut instead adds course offsets to starboard, and goes around the whole shallow area while keeping a safe distance to this area, as can be seen in Figure 6.2b. This shows that the anti-grounding system works as intended. As discussed in Section 6.1.1, the anti-grounding algorithm itself is not expected to find the optimal path to the waypoint, but to avoid land in reactive obstacle avoidance situations. From the scenarios looked at in this section, the system has proven that it is able to avoid ground in these kinds of short-horizon situations.



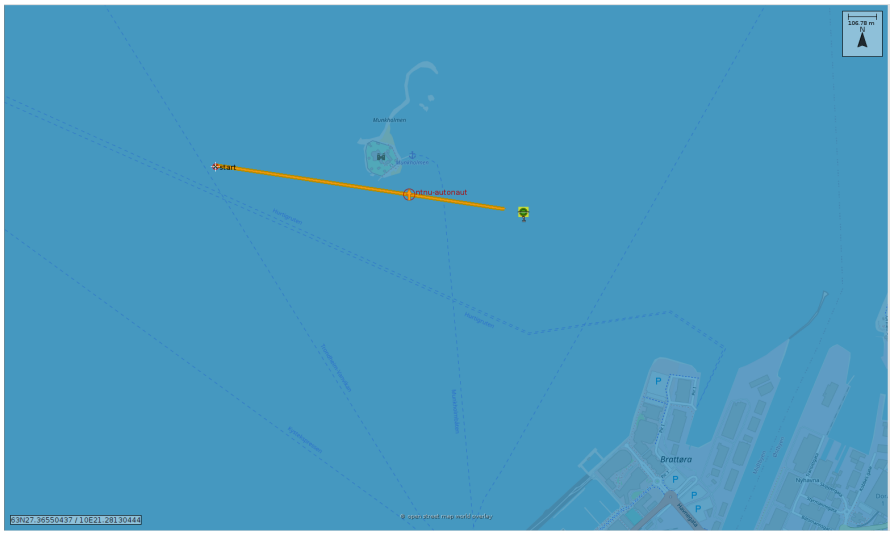
(a) Scenario without anti-grounding activated.

(b) Scenario with anti-grounding activated.

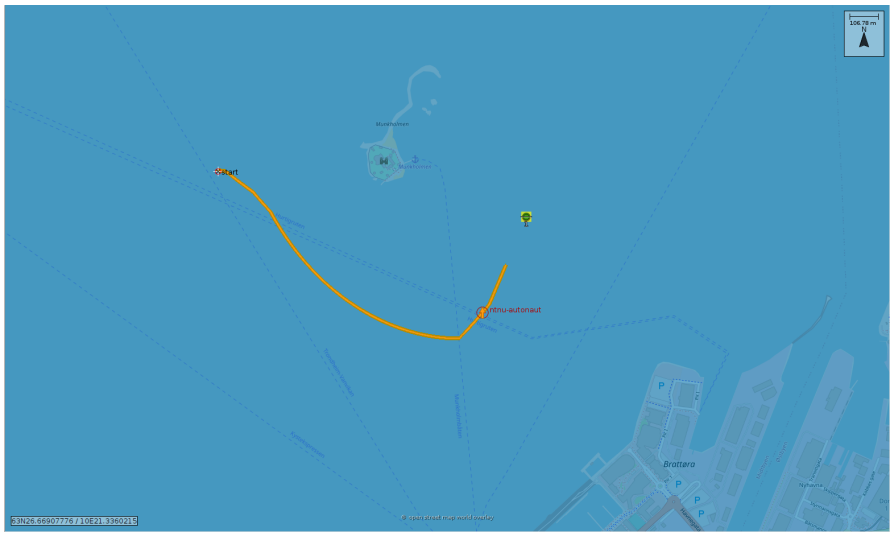


(c) Depth contours where the outer contour is 10.0 m deep.

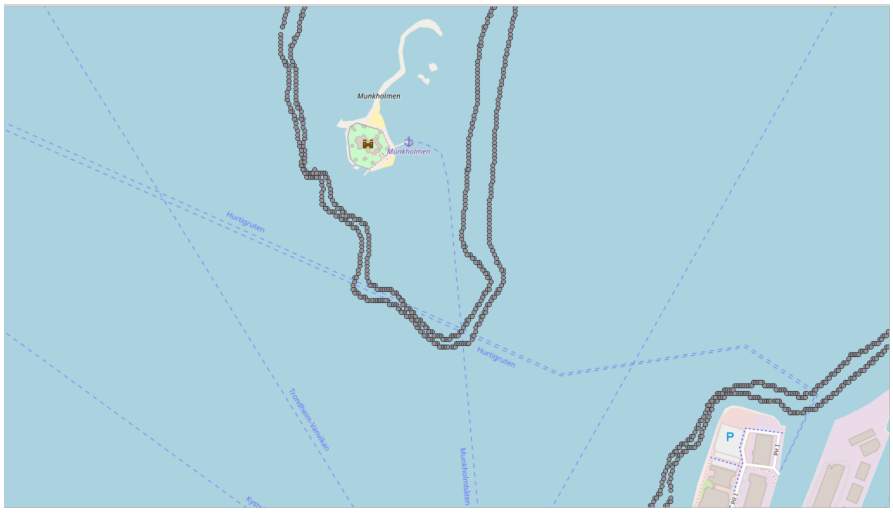
Figure 6.1: Pure anti-grounding scenario 1.



(a) Scenario without anti-grounding activated.



(b) Scenario with anti-grounding activated.



(c) Depth contours where the outer contour is 20.0 m deep.

Figure 6.2: Pure anti-grounding scenario 2.

6.2 Anti-Grounding and Collision Avoidance

In these scenarios, there are both static and dynamic obstacles present, and the anti-grounding system will be tested together with the collision avoidance system to see how they function together. Here, the full system is expected to evaluate the two hazards and find a solution where the AutoNaut is able to keep a safe distance to both obstacles. Two head-on scenarios and one scenario where the obstacle vessel crosses from starboard will be tested.

6.2.1 Head-on scenario

In a head-on scenario, the COLREGS rule 14 says that each vessel should change its course to starboard in order to pass the other on the port side. For these scenarios, the minimum safe distances to vessels and to land are both set to 300.0 m, and the minimum safe depth value is 10.0 m. The cost of grounding is set to 100.0.

Scenario 1

In this head-on scenario, which is shown in Figure 6.3, the obstacle and the AutoNaut follow almost the exact same path towards each other and will meet with very few meters distance if no action is taken. When the anti-grounding system is deactivated, the AutoNaut chooses to go 90° starboard, since this is the preferred COLREGS direction as stated in rule 14, and thus it complies with COLREGS. The shortest distance between the two vessels is measured to be 142 m, and the shortest distance to land is about 65 m.

The distance between the AutoNaut and the obstacle vessel is shorter than desired because the SB-MPC surveillance range d_{init} is set to 2000 m in these simulations, which means that the AutoNaut starts evaluating the need to take action when an obstacle vessel is closer than 2000 m. In this head-on scenario, the obstacle vessel keeps a high speed and approaches the AutoNaut straight ahead, which means that the AutoNaut has too little time to react, and can not manage to keep a 300 m distance, although it turns 90° starboard. However, the fact that the AutoNaut clearly shows that it takes action to avoid collision and has turned 90° when the obstacle vessel passes, mitigates the problem with the short distance. The d_{init} parameter can be set to a higher value if considered necessary.

When the anti-grounding system is enabled, the AutoNaut turns to port instead, to avoid getting closer to land, at the same time as it avoids collision with the obstacle to the same degree as before. The shortest distance between the two vessels is 202 m, and the shortest distance to land is about 240 m, which explains why the AutoNaut does not choose to go starboard towards land and further decrease this distance. This behavior is not COLREGS-compliant behavior, but in this situation, the priority of avoiding grounding is considered more important than complying with COLREGS.

Scenario 2

In this head-on scenario, shown in Figure 6.4, the AutoNaut's path is a bit further starboard to the path of the obstacle (approx. 21 m), seen from the AutoNaut. The distance to land in 30°- 45° offset direction is also larger than in Scenario 1.

When the anti-grounding system is not activated, the AutoNaut turns 90° starboard like in Scenario 1. This is in compliance with COLREGS and results in a shortest distance between the two vessels of 168 m, and a shortest distance to land of not more than 50 m. This scenario is shown in Figure 6.4a.

When the anti-grounding system is activated, the AutoNaut considers that the safest action is to turn 30° and 45° starboard, instead of 90° starboard as when without anti-grounding, or 90° to port as in Scenario 1. In the chosen direction of 30° and 45° to starboard, land is further away than in Scenario 1, making this choice the perfect balance between avoiding collision and avoiding grounding. As the obstacle vessel comes closer (around 360 m), the risk of collision increases so much that the AutoNaut has to turn 90° starboard to obtain a larger distance to the obstacle vessel. Then the risk of collision is considered to be larger than the risk of grounding. Still, the distance to land has been kept significantly larger than when anti-grounding was deactivated. The shortest distance between the two vessels is 122 m, and the shortest distance to land is around 120 m. In total, the AutoNaut avoids getting too close to land while still complying with COLREGS and avoiding collision with the obstacle vessel. This scenario is shown in Figure 6.4b.

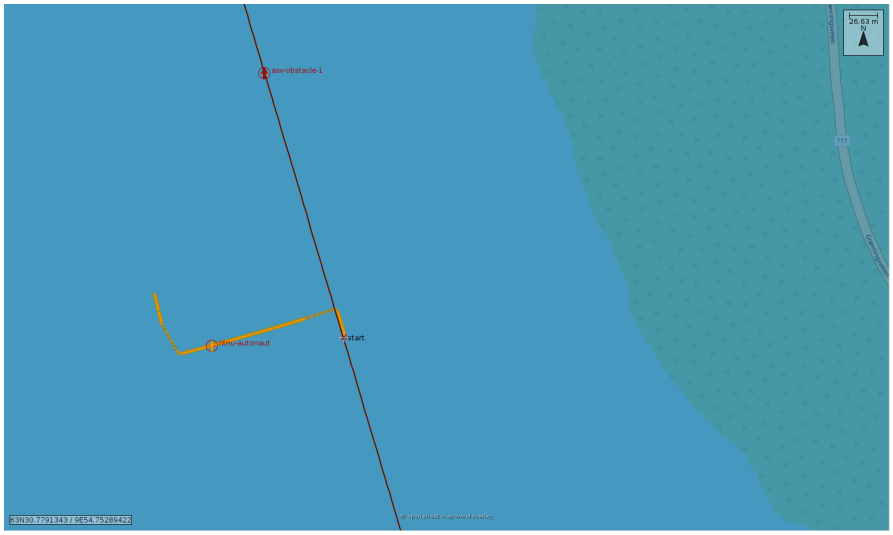
6.2.2 Obstacle vessel crossing from starboard

In a scenario where the obstacle vessel crosses from starboard, the AutoNaut is the give-way vessel, according to COLREGS rule 15, because it has the obstacle vessel on its starboard side. Thus, the AutoNaut has to keep out of the way and, if possible, avoid crossing ahead of the obstacle vessel. For this scenario, the minimum safe distances to vessels and to land are both set to 300.0 m, and the minimum safe depth value is set to 10.0 m. The cost of grounding is set to 50.0.

The result of the simulations done for the crossing from starboard scenario is shown in Figure 6.5. When the anti-grounding system is not activated, the AutoNaut turns 45° starboard, which is in compliance with COLREGS. The AutoNaut takes early action and makes it clear for the obstacle that it intends to pass behind it, which is in compliance with COLREGS rule 16. The shortest distance between the vessels is 318 m, and the shortest distance to land is around 110 m.

With the anti-grounding system activated, the AutoNaut still turns starboard as in compliance with COLREGS, but it chooses a course offset of only 15° and 30° to starboard, instead of between 45° and 75° as before. This means that the AutoNaut passes the obstacle vessel at a closer distance than before, at 289 m, but the distance to land is increased to around 170 m, and land is kept at a much safer distance than before.

(a) Head-on without anti-grounding activated.



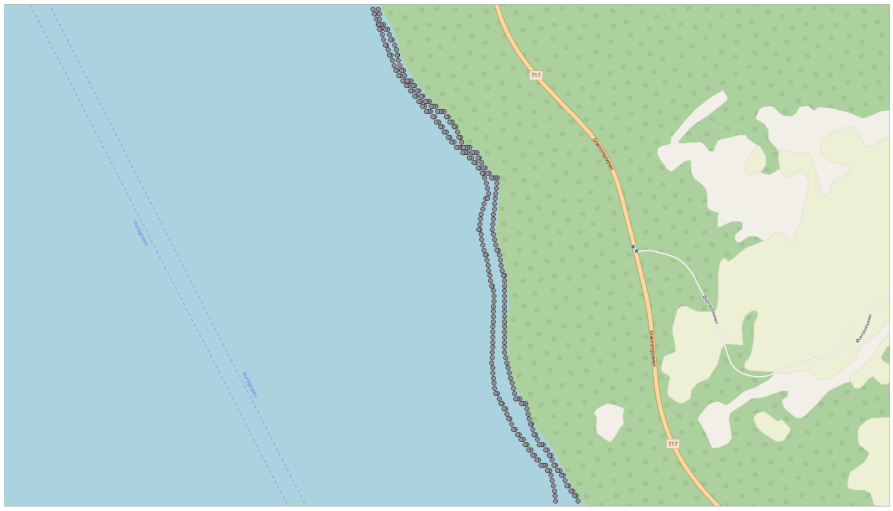
(b) Head-on with anti-grounding activated.

(c) Depth contours where the outer contour is 10.0 m deep.

Figure 6.3: Head-on scenario 1, with static and dynamic obstacles.

(a) Head-on without anti-grounding activated.

(b) Head-on with anti-grounding activated.

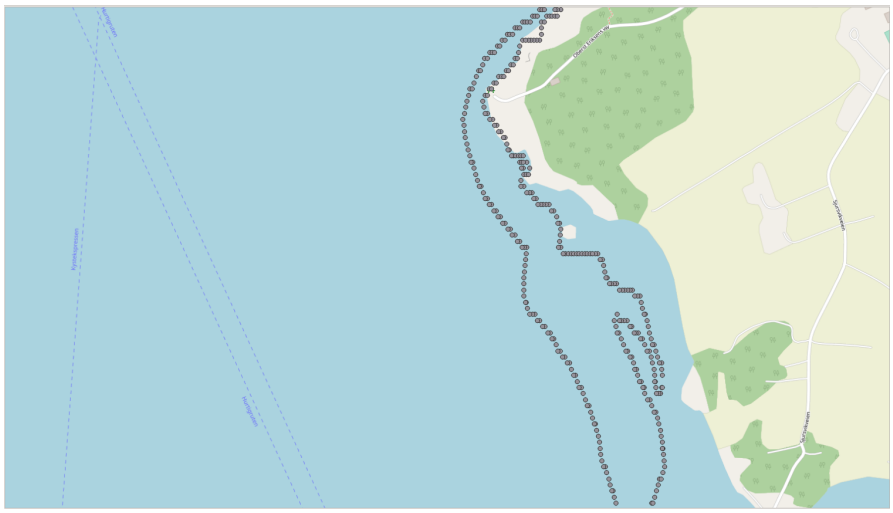


(c) Depth contours where the outer contour is 10.0 m deep.

Figure 6.4: Head-on scenario 2, with static and dynamic obstacles.

(a) Crossing from starboard without anti-grounding activated.

(b) Crossing from starboard with anti-grounding activated.



(c) Depth contours where the outer contour is 10.0 m deep.

Figure 6.5: Crossing from starboard scenario, with static and dynamic obstacles.

6.3 Anti-Grounding and Collision Avoidance With Environmental Factors

In this section, the environmental factors and their additional cost $K_{env}E^k$ are added to the system, as described in Section 5.2.7. The scenarios that will be looked at here are the scenarios where the AutoNaut originally chose to turn towards land to avoid collision with another vessel, and thus increased the risk of grounding. A tough environmental state will be added, and then it will be tested if the environmental factors have an impact and, most importantly, the desired impact on the decisions of the AutoNaut. In rough conditions, it is expected that the AutoNaut prioritizes avoiding grounding obstacles more than before and is careful about going close to land. Only scenarios with high environmental cost are looked at here because a low environmental cost leads to the same behavior as without the added cost. A change in behavior demands quite a high additional cost, which is desired because complying with COLREGS is important to ensure safe navigation at sea and should be the standard behavior as long as it is not considered unsafe. The total environmental cost can be tuned by adjusting the weights in K_{env}. In the simulations shown in Figure 6.6a and Figure 6.7a, a black arrow indicating the absolute direction of the wind and current is included, together with the wind and current speeds.

6.3.1 Head-on

This is the same head-on scenario as studied in Section 6.2.1 scenario 2, shown in Figure 6.4, only here environmental cost has been added. The results for this head-on scenario with environmental cost can be seen in Figure 6.6. The values of the environmental factors, the grounding cost, and the weights used in this scenario are shown in Table 6.2. The values of the factors are set to create tough environmental conditions, and although it is not a very probable scenario to have 2 meter high waves in the fjord, the values are chosen to test the system in these rough conditions. The plot in Figure 6.6b shows the cost of each environmental factor together with the resulting total environmental cost $K_{env}E^k$ (in red) for each course offset. The cost is highest for the course offsets that coincide with the wind and current direction, as intended by design. The costs from the ocean current and the bathymetry are quite low compared to the cost from wind and heave. This can be changed by adjusting the weights in K_{env}.

The result of the simulation is shown in Figure 6.6a. It can be seen that the AutoNaut turns 90° to port, instead of turning 30° and 45° to starboard where it was considered safe to go in Section 6.2.1 scenario 2 when the environmental factors were not added, see Figure 6.4. Because there is a quite strong wind and a current moving in a 30° direction, in addition to high waves leading to heave displacement, the total environmental cost is so high that keeping a safe distance to land is prioritized over choosing the COLREGS compliant behavior. Still, the AutoNaut manages to avoid collision with the obstacle vessel at the same time

as a safe distance is kept to land on starboard side. The shortest distance to the obstacle vessel is 118 m, which is only slightly shorter than before in Section 6.2.1 scenario 2, while the shortest distance to land has been increased to more than 200 m. This result shows that in rough environmental conditions, the system will prioritize avoiding grounding obstacles, and that the environmental factors can have an impact on the decisions of the AutoNaut.

Parameter	**Value**	**Description**
B	1.0	Bathymetry
Z	2.0 m	Heave Displacement
WH	2.0 m	Wave Height
V_W	10.0 m/s	Absolute Wind Speed
β_{V_W}	30.0°	Absolute Wind Direction
V_C	0.3 m/s	Absolute Current Speed
β_{V_C}	30.0°	Absolute Current Direction
K_{env}	[10.0, 50.0, 50.0, 50.0, 100.0]	Environmental Factor Weights
K_{ground}	100.0	Cost of Grounding

Table 6.2: Environmental factor values and parameters used in the head-on scenario.

6.3.2 Obstacle crossing from starboard

This is the same crossing from starboard scenario as in Section 6.2.2, which is shown in Figure 6.5. In this section, the environmental cost has been added. The results for this scenario can be seen in Figure 6.7. The values of the environmental factors, the grounding cost, and the weights used in this scenario are shown in Table 6.3 and the plot of the environmental costs is shown in Figure 6.7b. The environmental conditions are slightly better than in the head-on scenario in Section 6.3.1, and the weights are set lower. As a result, the environmental costs are smaller. Still, the cost is highest for the course offsets that coincide with the wind and current direction. The cost of bathymetry, heave, and wave height are all 10.0; therefore, only the line representing the cost of the wave height factor is visible in the plot.

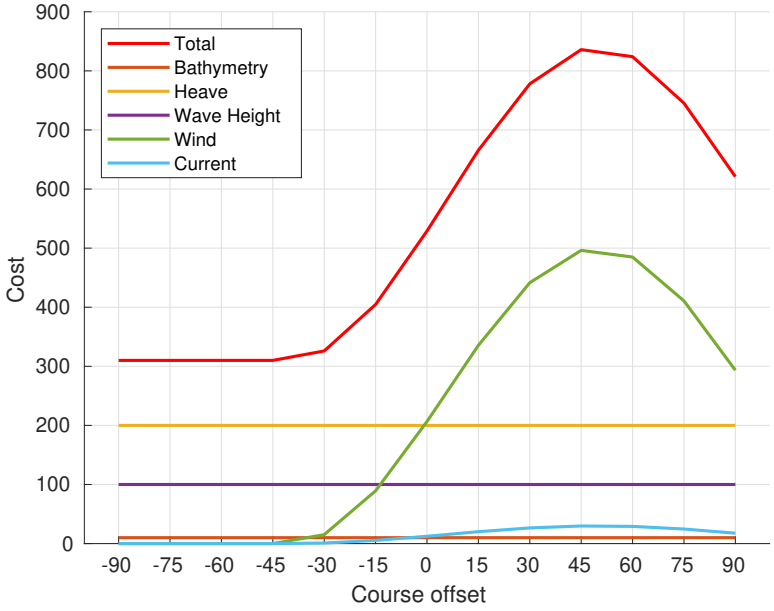
The simulation result is shown in Figure 6.7a. Here it can be seen that the AutoNaut turns 15° starboard and keeps the course offset at 15° until the obstacle vessel has passed, which is in compliance with COLREGS rule 15. Since the direction of the wind and the current coincides with a 45° course offset, all the course

offsets close to 45° receive a high environmental cost. In the 15° course offset direction, however, there is no ground close to the AutoNaut, the risk of grounding is therefore 0, and thus this becomes the optimal course offset choice. The shortest distance between the two vessels is 317 m, and the shortest distance to land is around 185 m. This result shows that in less rough environmental conditions, the system manages to comply with COLREGS, at the same time as it keeps a safe distance to other vessels and to ground.

Parameter	**Value**	**Description**
B	1.0	Bathymetry
Z	1.0 m	Heave Displacement
WH	1.0 m	Wave Height
V_W	10.0 m/s	Absolute Wind Speed
β_{V_W}	20.0°	Absolute Wind Direction
V_C	0.3 m/s	Absolute Current Speed
β_{V_C}	20.0°	Absolute Current Direction
K_{env}	[10.0, 10.0, 10.0, 10.0, 100.0]	Environmental Factor Weights
K_{ground}	50.0	Cost of Grounding

Table 6.3: Environmental factor values and parameters used in the crossing from starboard scenario.

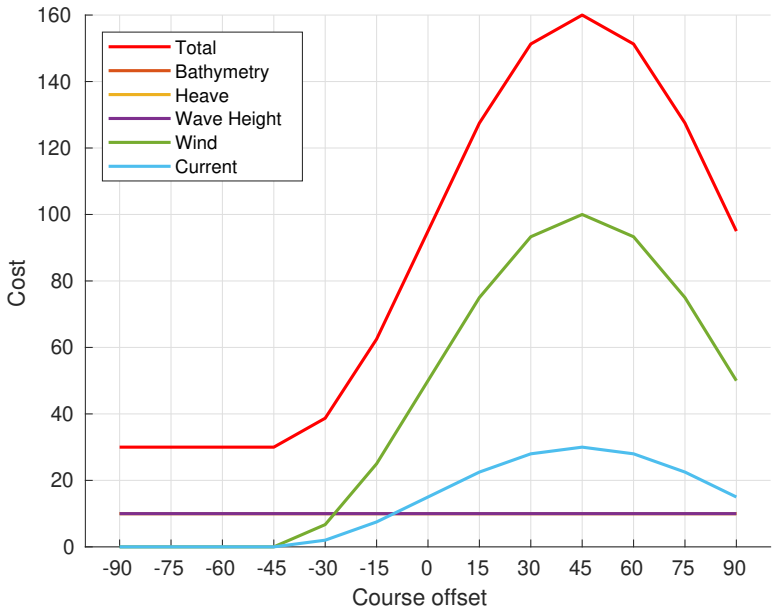
(a) Head-on scenario with environmental cost included.



(b) Environmental cost for each course offset.

Figure 6.6: Head-on scenario, with static and dynamic obstacles, and with environmental factors.

(a) Crossing from starboard scenario with static and dynamic obstacles and with environmental factors.



(b) Environmental cost for each course offset.

Figure 6.7: Crossing from starboard scenario, with static and dynamic obstacles, and with environmental factors.

Chapter 7

Discussion

The anti-grounding system developed in this thesis is a reactive static obstacle avoidance system intended to make the AutoNaut be aware of and avoid the surrounding static obstacles in situations where it has to diverge from its originally planned path. These situations will typically occur when the AutoNaut has to change its course to avoid collision with dynamic obstacles. Therefore, it is important that the existing collision avoidance system and the new grounding avoidance system work well together. The simulations done in Chapter 6 show that the performance of the collision and grounding avoidance system is overall good and as desired.

7.1 Pure Anti-Grounding

In the scenarios with pure anti-grounding, described in Section 6.1, where there are only static obstacles present, the anti-grounding system proves its ability to avoid grounding and keep a safe distance to land. As mentioned in Section 6.1, this anti-grounding system is not designed or expected to find the optimal path from start point to mission goal; that is the responsibility of the mission planning system. Therefore, the results obtained from the simulations prove that the system is robust for reactive static obstacle avoidance scenarios.

Initially, the anti-grounding system only evaluated the distance to land in the center direction, where the predicted speed direction of the AutoNaut would be, for each possible course offset. The result was that a course offset that made the AutoNaut tangent a hazardous area could be regarded as the optimal choice. Then, the AutoNaut could risk ending up very close to land on port or starboard side, or be lead into a local minimum, like a bay or a narrow strait. In Otterholm [32], this problem is pointed out in the discussion and future work sections. Therefore, the anti-grounding system regards the distance to land in a 15° direction to starboard and to port, in addition to the center direction, for each course offset, as described in Section 5.2.6 and illustrated in Figure 5.5. Here, only ±15° is evaluated, but this can be extended further by including additional directions and thus obtaining better control of the hazardous areas on the sides of the AutoNaut. In Figure 6.1b

it can be seen that the vessel is quite close to the first small promontory on the port side, which is due to the fact that there is no ground in the $-15°$ direction or the center direction of the chosen optimal course offset. An extension of this feature should therefore be considered implemented.

7.2 Anti-Grounding and Collision Avoidance

When the anti-grounding and collision avoidance systems are tested together in Section 6.2, the AutoNaut shows that it is able to avoid both the static and dynamic obstacles simultaneously. In the second head-on scenario (Figure 6.4) and in the crossing from starboard scenario (Figure 6.5), the anti-grounding system makes the AutoNaut choose a course offset that leads it further from land and closer to the obstacle vessel than without anti-grounding activated. In both cases, a course offset to starboard is chosen, which is in compliance with COLREGS. In the first head-on scenario (Figure 6.3), land is closer on starboard side than in the other two scenarios, and the AutoNaut turns to port to avoid moving closer to land, which is not in compliance with COLREGS. However, the system considers that the hazard of turning towards land is higher than the potential hazard of not complying with COLREGS. The limit between when complying with COLREGS should be prioritized and when avoiding ground should be prioritized can be changed by tuning the parameters of the system, found in Table 6.1. The parameter κ represents the cost of not complying with COLREGS, and the parameters $k_{\Delta\chi_{SB}}$, $k_{\Delta\chi_P}$, $k_{\chi_{SB}}$ and k_{χ_P} can be tuned to penalize course changes to port higher than course changes to starboard, which is in compliance with COLREGS. This is described in Section 5.2.8. The grounding cost K_{ground} can be tuned to make it more or less important to avoid grounding.

Generally, the system can be tuned to achieve the desired behavior for a given vehicle and situation. The SB-MPC algorithm, where both the collision and grounding avoidance algorithms are implemented, finds the optimal control behavior by minimizing the hazard and selecting the related course offset. In practice, this means that the size of the hazard and thus the size of the risk and cost decides which course offset is considered optimal. Therefore, the desired system behavior can be achieved by tuning the parameters and the different costs, depending on the vessel and the type of mission. In the case of the AutoNaut, which is a small, slow-moving, wave-propelled ASV unable to control its own speed directly and consequently has limited maneuverability, it is important to keep a relatively large distance to land. In other cases, for vessels that keep a higher speed and that are able to control and change their speed and course easily, it could be more important to comply with COLREGS and not as dangerous to move closer to land.

7.3 Environmental Factors

In the scenarios presented in Section 6.3, the environmental factors are added to the grounding cost function, making the choice of optimal course offset dependent on the current environmental conditions. The result is that the AutoNaut prioritizes to keep a larger distance to land when the wind and/or current direction aligns with the course of the AutoNaut. This behavior is as desired. In the head-on scenario in Figure 6.6a, the wind and current move in a 30° direction, which coincide with a course offset of about +45°, resulting in a higher environmental cost for the course offsets around +45°, as shown in Figure 6.6b. As a consequence, the AutoNaut turns to port, which is not in compliance with COLREGS.

In the crossing from starboard scenario in Figure 6.7a, the wind and current move in a 20° direction, which in this case also coincide with a course offset of about +45°. Even though the +15° course offset receives quite a high environmental cost, +15° is found to be the optimal course offset choice. This is because there is no land in that direction, meaning the risk of grounding, $\mathcal{R}_{ground}$, is zero. Thus, the AutoNaut can turn starboard and comply with COLREGS, at the same time as it keeps a safe distance to land. If there had been land close by in the +15° course offset direction, the AutoNaut would most likely have had to turn port to avoid moving too close to land, which was the case in the head-on scenario.

The results from the simulations that include environmental factors look promising. Nevertheless, the implementation of the environmental factors can be improved. The goal of the implementation is to have a function that represents the current environmental conditions, which it now does to some extent. The input should consist of actual sensor values from the sensors onboard the AutoNaut or weather forecasts if sensor data is unavailable. In the current implementation, only constant, predefined simulation values have been used. The bathymetry factor needs to be implemented using the approach suggested in Section 5.2.7. More simulations should be done to further explore how the environment affects the system.

7.4 The Complete System

In total, the complete ENC-based collision and grounding avoidance system shows good performance in the simulated scenarios. However, some improvements regarding parameter tuning, simulations, and general system behavior can be made.

The collision and grounding avoidance algorithms contain many tuning parameters, as seen in Table 6.1, which highly affect the behavior of the system. In this work, the tuning of the collision avoidance parameters is based on the values used in Hagen [2], and are then adjusted to fit with the AutoNaut and the integrated anti-grounding system. The parameters related to anti-grounding and environmental factors are tuned through simulations. As pointed out by Hagen, tuning the CAS algorithm is not straightforward because there are many parameters and some depend on each other. Further exploration on how the system can

be tuned optimally should be done to improve the behavior of the system.

The simulation study contains several different scenarios that demonstrate the most important mechanisms of the system. To ensure that the system performs well in all situations and under various conditions, more simulations with other scenarios and different environmental conditions should be done. Real-life testing with the AutoNaut should also be performed to verify that the system functions in practice, and to adjust the system thereafter.

The minimum safe distance to land and to other vessels is set to 100 m or 300 m in the simulations, but the AutoNaut is not always able to keep this distance. Tuning the SB-MPC surveillance distance parameter could improve the behavior, but note that these safe distances are not intended as absolute or explicit constraints. Rather, the distances are set to make the AutoNaut aware of the high risk of keeping a distance shorter than the minimum safe distance. Although the optimal behavior is to keep the minimum distance to all obstacles, the system has not failed if it is not able to achieve that. Other factors have to be taken into account as well. In some situations where multiple obstacles are present, there are conflicting demands, making it impossible to meet all requirements. Then, the focus is to find a solution that will make the situation as safe as possible, considering all obstacles and finding a compromise. The results of the simulations show that the AutoNaut is able to do exactly that when facing both static and dynamic obstacles.

Chapter 8

Conclusion

In this master's thesis an ENC-based anti-grounding system has been developed and integrated into the existing SB-MPC collision avoidance system (CAS) developed by Johansen *et al.* [1] and implemented by Hagen *et al.* [2], [3]. Different extraction and representation methods for ENC data have been studied, and the method developed in Lauvås [37] has been found to be the optimal to use for the AutoNaut. With this method the ENC data is represented as point clouds and stored in a database, which makes it fast and in need of little storage space. The anti-grounding theory has been validated by looking at the evolution of the grounding hazard when the AutoNaut approaches land. Several simulations of scenarios with static and dynamic obstacles have been done, both with and without including environmental factors.

The results look promising and show that the system is able to evaluate the hazard of the different obstacles, in addition to the importance of COLREGS-compliance, against each other. The system then finds a solution where a safe distance is kept to all obstacles. When environmental factors are added, the system takes them into consideration when evaluating the potential hazard of grounding. In rough conditions, grounding avoidance is given a higher priority than in calm conditions, and in some cases a higher priority than COLREGS compliance.

Chapter 9

Future Work

The development of the ENC-based anti-grounding algorithm done in this thesis and the simulation study performed, form a good foundation for further development and real-life testing of the collision and grounding avoidance system. The main objectives for future work on the system are mentioned below.

- More simulations should be done of different scenarios, including multiple dynamic and static obstacles. The system should be tested even more thoroughly in order to be completely ready for real-life testing.
- Real-life testing. The system should be tested on the AutoNaut out on the ocean and in the fjord.
- The implementation of the environmental factors can be studied further and should be tested in real-life with real-time sensor data. This would give a better understanding of how it should be designed and tuned, and how the AutoNaut reacts to tough environmental conditions.
- Tune the parameters of the collision avoidance, anti-grounding, and environmental factors, to further improve the behavior of the system. The SB-MPC surveillance distance, and possibly other distance parameters, can be modified to take into account the speed of the obstacle vessels so that the AutoNaut has the opportunity to react early enough to obtain a safe distance to other vessels.
- Include other static obstacles in addition to land, like buoys, beacons, wrecks, and underwater/awash rocks.
- Develop a speed prediction model for the AutoNaut and include it in this system once it has been made. The model will improve the predictions of the AutoNaut's trajectory and thus help the system make more correct course offset choices. It will also be able to tell if some course offsets will lead to maneuvering problems and should be discarded.

A paper about the ENC-based anti-grounding system and the results presented in this thesis will be published in the near future.

Bibliography

[1] T. A. Johansen, T. Perez and A. Cristofaro, 'Ship collision avoidance and colregs compliance using simulation-based control behavior selection with predictive hazard assessment,' *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 12, pp. 3407–3422, 2016. DOI: 10.1109/TITS.2016.2551780.

[2] I. B. Hagen, 'Collision Avoidance for ASVs Using Model Predictive Control,' M.S. thesis, Norwegian University of Science and Technology, 2017.

[3] I. B. Hagen, D. K. M. Kufoalor, E. F. Brekke and T. A. Johansen, 'MPC-based Collision Avoidance Strategy for Existing Marine Vessel Guidance Systems,' in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 7618–7623. DOI: 10.1109/ICRA.2018.8463182.

[4] *Waymo - Our Journey*. [Online]. Available: https://waymo.com/journey/.

[5] *Kongsberg - Autonomous Future*. [Online]. Available: https://www.kongsberg.com/no/maritime/about-us/news-and-media/our-stories/autonomous-future/.

[6] P. McGillivary, J. Borges de Sousa, R. Martins, K. Rajan and F. Leroy, 'Integrating autonomous underwater vessels, surface vessels and aircraft as persistent surveillance components of ocean observing studies,' in *2012 IEEE/OES Autonomous Underwater Vehicles (AUV)*, 2012, pp. 1–5. DOI: 10.1109/AUV.2012.6380734.

[7] A. Dallolio, L. Bertino, L. Chrpa, T. A. Johansen, M. Ludvigsen, K. A. Orvik, L. H. Smedsrud, J. Sousa, I. B. Utne, P. Johnston and K. Rajan, 'Long-duration Autonomy for Open Ocean Exploration: Preliminary Results Challenges,' *RSS 2019 Workshop on Robots in the Wild: Challenges in Deploying Robust Autonomy for Robotic Exploration*, 2019.

[8] EMSA, 'Annual overview of marine casualties and incidents 2020,' 2020. [Online]. Available: http://www.emsa.europa.eu/accident-investigation-publications/annual-overview.html.

[9] B. C. Shah, P. Švec, I. R. Bertaska, A. J. Sinisterra, W. Klinger, K. von Ellenrieder, M. Dhanak and S. K. Gupta, 'Resolution-adaptive risk-aware trajectory planning for surface vehicles operating in congested civilian traffic,'

Autonomous Robots, vol. 40, no. 7, pp. 1139–1163, 2016. DOI: 10.1007/s10514-015-9529-x.

[10] M. Blaich, S. Köhler, J. Reuter and A. Hahn, 'Probabilistic collision avoidance for vessels,' *IFAC-PapersOnLine*, vol. 48, no. 16, pp. 69–74, 2015. DOI: https://doi.org/10.1016/j.ifacol.2015.10.260.

[11] M. Candeloro, A. M. Lekkas and A. J. Sørensen, 'A voronoi-diagram-based dynamic path-planning system for underactuated marine vessels,' *Control Engineering Practice*, vol. 61, pp. 41–54, 2017, ISSN: 0967-0661. DOI: https://doi.org/10.1016/j.conengprac.2017.01.007.

[12] S. M. LaValle, *Rapidly-exploring random trees: A new tool for path planning*. Computer Science Dept., Iowa State University, 1998.

[13] Y. Kuwata, M. T. Wolf, D. Zarzhitsky and T. L. Huntsberger, 'Safe Maritime Autonomous Navigation With COLREGS, Using Velocity Obstacles,' *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 110–119, 2014. DOI: 10.1109/JOE.2013.2254214.

[14] D. Fox, W. Burgard and S. Thrun, 'The dynamic window approach to collision avoidance,' *IEEE Robotics Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997. DOI: 10.1109/100.580977.

[15] B.-O. H. Eriksen, M. Breivik, K. Y. Pettersen and M. S. Wiig, 'A modified dynamic window algorithm for horizontal collision avoidance for AUVs,' in *2016 IEEE Conference on Control Applications (CCA)*, 2016, pp. 499–506. DOI: 10.1109/CCA.2016.7587879.

[16] O. Khatib, 'Real-time obstacle avoidance for manipulators and mobile robots,' in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, vol. 2, 1985, pp. 500–505. DOI: 10.1109/ROBOT.1985.1087247.

[17] E. Serigstad, 'Hybrid Collision Avoidance for Autonomous Surface Vessels,' M.S. thesis, Norwegian University of Science and Technology, 2017.

[18] Ø. A. G. Loe, 'Collision Avoidance for Unmanned Surface Vehicles,' M.S. thesis, Norwegian University of Science and Technology, 2008.

[19] A. Bemporad and M. Morari, 'Robust model predictive control: A survey,' in *Robustness in identification and control*, A. Garulli and A. Tesi, Eds., London: Springer London, 1999, pp. 207–226.

[20] P. Scokaert and D. Mayne, 'Min-max feedback model predictive control for constrained linear systems,' *IEEE Transactions on Automatic Control*, vol. 43, no. 8, pp. 1136–1142, 1998. DOI: 10.1109/9.704989.

[21] D. K. M. Kufoalor, T. A. Johansen, E. F. Brekke, A. Hepsø and K. Trnka, 'Autonomous maritime collision avoidance: Field verification of autonomous surface vehicle behavior in challenging scenarios,' *Journal of Field Robotics*, vol. 37, no. 3, pp. 387–403, 2020. DOI: https://doi.org/10.1002/rob.21919.

[22] P. Tang, R. Zhang, D. Liu, L. Huang, G. Liu and T. Deng, 'Local reactive obstacle avoidance approach for high-speed unmanned surface vehicle,' *Ocean Engineering*, vol. 106, pp. 128–140, 2015, ISSN: 0029-8018. DOI: https://doi.org/10.1016/j.oceaneng.2015.06.055.

[23] R. Guardeño, M. J. López, J. Sánchez, A. González and A. Consegliere, 'A robust reactive static obstacle avoidance system for surface marine vehicles,' *Sensors*, vol. 20, no. 21, 2020, ISSN: 1424-8220. DOI: 10.3390/s20216262.

[24] S. Blindheim, S. Gros and T. A. Johansen, 'Risk-based model predictive control for autonomous ship emergency management,' *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 14 524–14 531, 2020, 21th IFAC World Congress, ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2020.12.1456.

[25] A. Bakdi, I. K. Glad, E. Vanem and Ø. Engelhardtsen, 'Ais-based multiple vessel collision and grounding risk identification based on adaptive safety domain,' *Journal of Marine Science and Engineering*, vol. 8, 2020. DOI: 10.3390/jmse8010005.

[26] A. Mazaheri, J. Montewka and P. Kujala, 'Modeling the risk of ship grounding—a literature review from a risk management perspective,' *WMU Journal of Maritime Affairs*, vol. 13, pp. 269–297, 2014. DOI: https://doi.org/10.1007/s13437-013-0056-3.

[27] D. Kang, J. Jung, S. Oh and S. Kim, 'Automatic route checking method using post-processing for the measured water depth,' in *2015 International Association of Institutes of Navigation World Congress (IAIN)*, 2015, pp. 1–5. DOI: 10.1109/IAIN.2015.7352240.

[28] J. Larson, M. Bruch and J. Ebken, 'Autonomous navigation and obstacle avoidance for unmanned surface vehicles,' in *Unmanned Systems Technology VIII*, G. R. Gerhart, C. M. Shoemaker and D. W. Gage, Eds., International Society for Optics and Photonics, vol. 6230, SPIE, 2006, pp. 53–64. DOI: 10.1117/12.663798.

[29] J. Larson, M. Bruch, R. Halterman, J. Rogers and R. Webster, 'Advances in autonomous obstacle avoidance for unmanned surface vehicles,' in *AUVSI Unmanned Systems North America*, 2007, pp. 1–15.

[30] S. Reed and V. E. Schmidt, 'Providing nautical chart awareness to autonomous surface vessel operations,' in *OCEANS 2016 MTS/IEEE Monterey*, 2016, pp. 1–8. DOI: 10.1109/OCEANS.2016.7761472.

[31] M. Mąka and J. Magaj, 'Data extraction from an electronic s-57 standard chart for navigational decision systems,' *Zeszyty Naukowe / Akademia Morska w Szczecinie*, vol. 30, no. 102, pp. 83–87, 2012.

[32] O. S. Otterholm, 'Extracting Mapped Hazards from Electronic Navigational Charts for ASV Collision Avoidance,' M.S. thesis, Norwegian University of Science and Technology, 2019.

[33] T. Midjås, 'SBMPC Collision Avoidance for the ReVolt Model-Scale Ship,' M.S. thesis, Norwegian University of Science and Technology, 2019.

[34] T. D. Grande, 'PSB-MPC Collision Avoidance with Anti-Grounding,' M.S. thesis, Norwegian University of Science and Technology, Feb. 2021.

[35] T. Tengesdal, E. F. Brekke and T. A. Johansen, 'On collision risk assessment for autonomous ships using scenario-based mpc,' *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 14 509–14 516, 2020, 21th IFAC World Congress, ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2020.12.1454.

[36] T. Tengesdal, T. A. Johansen and E. Brekke, 'Risk-based autonomous maritime collision avoidance considering obstacle intentions,' in *2020 IEEE 23rd International Conference on Information Fusion (FUSION)*, 2020, pp. 1–8. DOI: 10.23919/FUSION45008.2020.9190212.

[37] N. Lauvås, 'Design and development of a robotic fish tracking vehicle,' M.S. thesis, Norwegian University of Science and Technology, 2020.

[38] The Society of Naval Architects and Marine Engineers, 'Nomenclature for Treating the Motion of a Submerged Body Through a Fluid,' *Technical and Research Bulletin No. 1-5*, pp. 1–15, 1950.

[39] T. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, 2021.

[40] National Ocean Service - National Oceanic and Atmospheric Administration - U.S. Department of Commerce, *What is geodesy?* [Online]. Available: https://oceanservice.noaa.gov/facts/geodesy.html.

[41] Kartverket, *Referanserammer for Noreg*. [Online]. Available: https://www.kartverket.no/til-lands/posisjon/referanserammer-for-noreg.

[42] J. Pinto, P. Calado, J. Braga, P. Dias, R. Martins, E. Marques and J. Sousa, 'Implementation of a control architecture for networked vehicle systems,' *IFAC Proceedings Volumes*, vol. 45, no. 5, pp. 100–105, 2012. DOI: https://doi.org/10.3182/20120410-3-PT-4028.00018.

[43] IMO, *Convention on the International Regulations for Preventing Collisions at Sea (COLREG)*, 1972.

[44] AutoNaut Ltd., *Autonaut*. [Online]. Available: https://www.autonautusv.com/.

[45] P. Johnston and C. Pierpoint, 'Deployment of a passive acoustic monitoring (PAM) array from the AutoNaut wave-propelled unmanned surface vessel (USV),' in *OCEANS 2017 - Aberdeen*, 2017, pp. 1–4. DOI: 10.1109/OCEANSE.2017.8084780.

[46] A. Dallolio, *Autonaut documentation wiki*. [Online]. Available: http://autonaut.itk.ntnu.no/doku.php?id=start.

[47] A. Dallolio, B. Agdal, A. Zolich, J. A. Alfredsen and T. Johansen, 'Long-Endurance Green Energy Autonomous Surface Vehicle Control Architecture,' Oct. 2019, pp. 1–10. DOI: 10.23919/OCEANS40490.2019.8962768.

[48] Airmar, *120WX WeatherStation*. [Online]. Available: https://www.airmar.com/weather-description.html?id=152.

[49] *SenTiBoard*. [Online]. Available: https://sentiboard.com/about.html.

[50] Kartverket, *Elektroniske sjøkart (ENC)*. [Online]. Available: https://www.kartverket.no/til-sjos/kart/elektroniske-sjokart-enc.

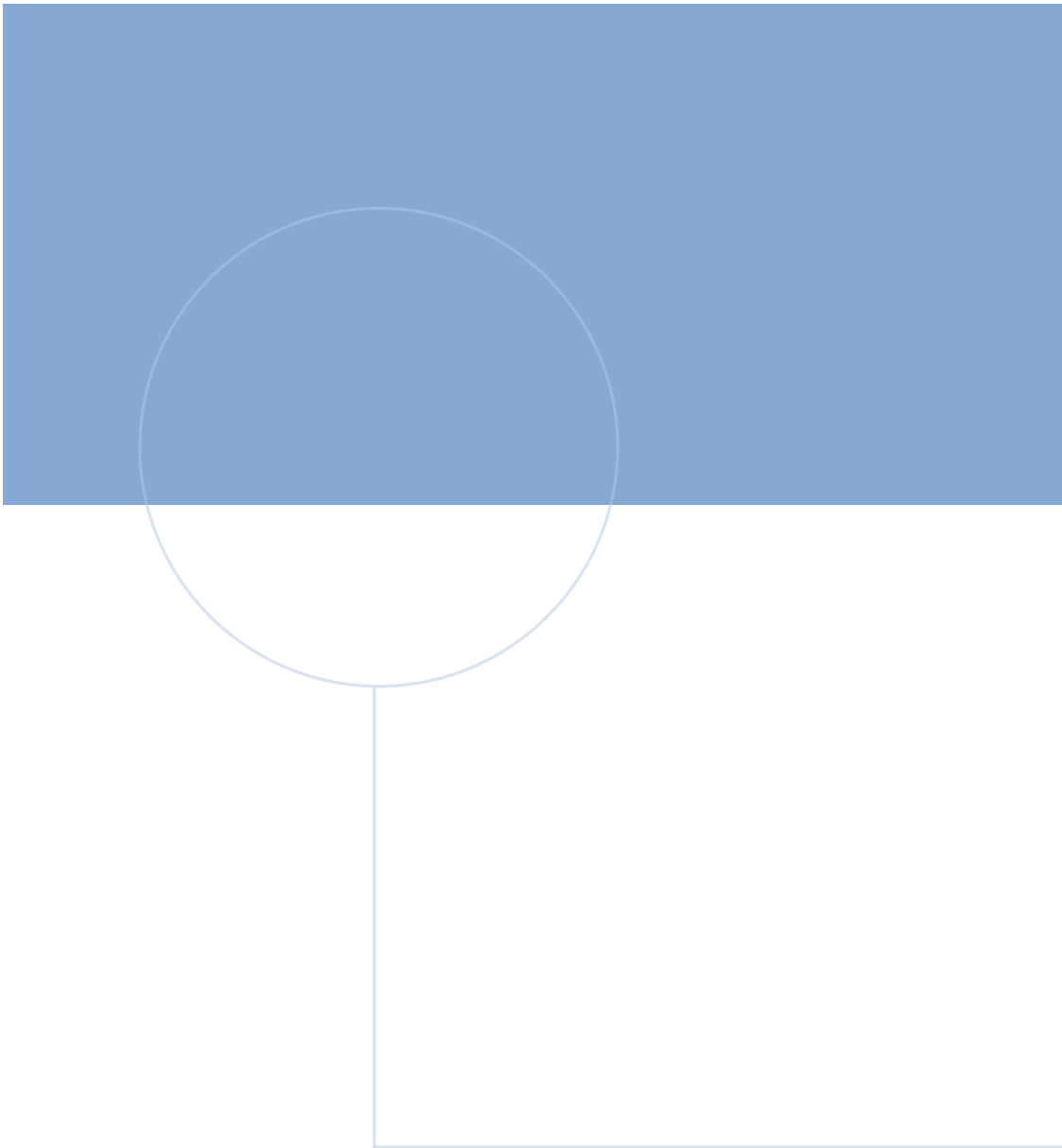
[51] International Hydrographic Bureau Monaco, *IHO Transfer Standard for Digital Hydrographic Data - Special Publication No. 57. 3.1*, Nov. 2000.

[52] International Hydrographic Bureau Monaco, *S-57 Appendix A, IHO Object Catalogue*, Nov. 2000.

[53] ESRI - Environmental Systems Research Institute, *ESRI Shapefile Technical Description*, Jul. 1998.

[54] B. Foss and T. A. N. Heirung, *Merging Optimization and Control*. NTNU, 2016.

[55] S. D. Sæter, 'COLREGs compliant Collision Avoidance System for a Wave and Solar Powered USV,' M.S. thesis, Norwegian University of Science and Technology, 2018.



NTNU

Norwegian University of
Science and Technology