Master's thesis

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

Ziyue Zhuang

COLREGs-based Collision Avoidance for ASVs using Modified Timed-Elastic-Band Approach

A Decentralized and Optimal Method

Master's thesis in Marine Technology
Supervisor: Dong Trong Nguyen
June 2023

NTNU
Norwegian University of
Science and Technology

Ziyue Zhuang

COLREGs-based Collision Avoidance for ASVs using Modified Timed-Elastic-Band Approach

A Decentralized and Optimal Method

Master's thesis in Marine Technology
Supervisor: Dong Trong Nguyen
June 2023

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

NTNU
Norwegian University of
Science and Technology

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

MSC THESIS DESCRIPTION SHEET

Name of the candidate: Ziyue Zhuang

Field of study: Marine control engineering

Thesis title (Norwegian): Kollisjonsunngåelse for ASVer basert på COLREGs ved bruk av en Modifisert Timed-Elastic-Band tilnærming: En desentralisert og optimal metode

Thesis title (English): COLREGs-based Collision Avoidance for ASVs using Modified Timed-Elastic-Band Approach: A Decentralized and Optimal Method

Background

Autonomous Surface Vehicles (ASVs) are unmanned watercraft designed for a range of applications, including oceanographic research, environmental monitoring, underwater exploration, and marine transportation. Due to their autonomous operation, ASVs must be capable of safely avoiding collisions with other vessels, stationary objects such as buoys or rocks, and even marine life when following global paths or tracking waypoints. To achieve this, the Timed-Elastic-Band (TEB) approach, a popular local trajectory planner for mobile and car-like robots, provides a promising choice due to its efficiency, optimality, modularity, and flexibility. However, when adapting it to maritime collision avoidance, the algorithm must properly accommodate the International Regulations for Preventing Collisions at Sea (COLREGs).

Work description

1. Perform a background and literature review to provide information and relevant references on:
   - Ship collision avoidance
   - Timed-Elastic-Band approach

   Write a list with abbreviations and definitions of terms, explaining relevant concepts related to the literature study and project assignment.
2. Propose a decentralized collision avoidance method of ASVs by adapting the TEB framework considering COLREGs.
3. Build a simulator in C++:
   - Convert the USV Otter high-fidelity model from MSS Python Vehicle Simulator to C++.
   - Use reference models for heading and surge speed.
   - Adopt line-of-sight guidance law for straight-line path following.
   - Develop PID controller for heading and sliding mode controller for surge speed.
   - Apply straight-line and circular motion prediction techniques.
   - Modify and implement TEB planner using g2o optimization framework.
4. Verify the systems by simulation of waypoints tracking.
5. Validate the systems by simulation of multi-vessel encounters using pre-defined scenarios proposed by Imazu (1987) with two self-defined and more challenging scenarios.
6. Analyze and evaluate the performance of the collision avoidance system.
7. Conclude the thesis and recommend further work.

Specifications

The scope of work may prove to be larger than initially anticipated. By the approval from the supervisor, described topics may be deleted or reduced in extent without consequences with regard to grading.

The candidate shall present personal contribution to the resolution of problems within the scope of work. Theories and conclusions should be based on mathematical derivations and logic reasoning identifying the various steps in the deduction.

The report shall be organized in a logical structure to give a clear exposition of background, results, assessments, and conclusions. The text should be brief and to the point, with a clear language. Rigorous mathematical deductions and illustrating figures are preferred over lengthy textual descriptions. The report shall have font size 11 pts., and it is not expected to be longer than 60-80 A4 pages, from introduction to conclusion, unless otherwise agreed upon. It shall be written in English (preferably US) and contain the following elements: Title page, abstract, acknowledgements, thesis specification, list of symbols and acronyms, table of contents, introduction with objective, background, and scope and delimitations, main body with problem formulations, derivations/developments and results, conclusions with recommendations for further work, references, and optional appendices. All figures, tables, and equations shall be numerated. The original contribution of the candidate and material taken from other sources shall be clearly identified. Work from other sources shall be properly acknowledged using quotations and a Harvard citation style (e.g., *natbib* Latex package). The work is expected to be conducted in an honest and ethical manner, without any sort of plagiarism and misconduct. Such practice is taken very seriously by the university and will have consequences. NTNU can use the results freely in research and teaching by proper referencing, unless otherwise agreed upon.

The thesis shall be submitted with a printed and electronic copy to the main supervisor, with the printed copy signed by the candidate. The final revised version of this thesis description must be included. The report must be submitted according to NTNU procedures. Computer code, pictures, videos, data series, and a PDF version of the report shall be included electronically with all submitted versions.

Start date: 9 January 2023 **Due date:** 11 June 2023

Supervisor: Dong Trong Nguyen
Co-advisor(s): N/A

Trondheim, 9 January 2023

Digitally signed by Dong Trong Nguyen
Date: 2023.06.07 09:06:17 +02'00'

Dong Trong Nguyen
Supervisor

Abstract

Autonomous Surface Vehicles (ASVs) are unmanned watercraft used for various applications, including oceanographic research, environmental monitoring, and marine transportation. Due to their autonomous operation, ASVs must be capable of safely avoiding collisions with other vessels and static obstacles when following global paths or waypoints. This master thesis aims to develop a decentralized and optimal collision avoidance system by adapting the Timed-Elastic-Band (TEB) approach to ASVs. The TEB approach is an efficient and robust algorithm for local trajectory planning of mobile and car-like robots with various advantages, e.g., modularity and flexibility. However, to utilize it for maritime collision avoidance, it must appropriately accommodate the traffic rules at sea, the International Regulations for Preventing Collisions at Sea (COLREGs). Furthermore, an action selection strategy is designed for complex multi-vessel encounters. The system's feasibility and effectiveness will be validated through two types of simulations, waypoints tracking and the extended Imazu problem, thus demonstrating the practical application of the modified TEB approach for ASVs. The proposed method will contribute to the advancement of autonomous navigation systems for ASVs.

Sammendrag

Autonome overflatefartøy (ASV-er) er ubemannede fartøy som brukes til ulike formål, inkludert oseanografisk forskning, miljøovervåking og maritim transport. På grunn av sin autonome drift må ASV-er være i stand til å unngå kollisjoner med andre fartøy og statiske hindringer når de følger globale ruter eller veipunkter. Denne masteroppgaven har som mål å utvikle et desentralisert og optimalt kollisjonsunngåelsessystem ved å tilpasse Timed-Elastic-Band (TEB)-metoden til ASV-er. TEB-metoden er en effektiv og robust algoritme for lokal trajektplanlegging for mobile roboter og bil-lignende roboter med ulike fordeler, for eksempel modularitet og fleksibilitet. Imidlertid må den tilpasses for maritim kollisjonsunngåelse ved å ta hensyn til sjøens trafikkregler, International Regulations for Preventing Collisions at Sea (COLREGs). Videre er det utviklet en handlingsutvelgelsesstrategi for komplekse flerfartøyssituasjoner. Systemets gjennomførbarhet og effektivitet vil bli validert gjennom to typer simuleringer, veipunktsporing og den utvidede Imazu-problemet, for å demonstrere den praktiske anvendelsen av den modifiserte TEB-metoden for ASV-er. Den foreslåtte metoden vil bidra til fremgangen innen autonom navigasjonssystemer for ASV-er.

Contents

Figures

Tables

Abbreviations

ASV Autonomous Surface Vehicle

COLREGs International Regulations for Preventing Collisions at Sea

IMO International Maritime Organization

MASS Autonomous Surface Ship

VTS Vessel Traffic Services

MPC Model Predictive Control

VO Velocity Obstacle

APF Artificial Potential Field

CBF Control Barrier Functions

GA Genetic Algorithm

PSO Particle Swarm Optimization

DRL Deep Reinforcement Learning

TEB Timed-Elastic-Band

DOF Degree-of-Freedom

PID Proportional-Integral-Derivative

SMC Sliding Mode Control

ILOS Integral Line-of-Sight

NED North-East-Down

CPA Closest Point of Approach

TCPA Time to Closest Point of Approach

DCPA Distance at Closest Point of Approach

Chapter 1

Introduction

1.1 Background

Driven by the prospect of promoting safety, eliminating human errors, enhancing performance, lowering costs, and improving efficiency in maritime navigation and operation (Vagale et al. 2021), numerous research and projects are being conducted related to developing Autonomous Surface Vehicles (ASVs). For instance, YARA started to operate a completely electric, autonomous, and emission-free container ship and departed for its maiden voyage in the Oslo fjord (Yara 2021). Later, an autonomous, AI-powered marine research vessel named Mayflower Autonomous Ship crossed the Atlantic Ocean from Plymouth UK (Mayflower 2022). For autonomous vessels, one of the major challenges in autonomous navigation is collision avoidance. Its objective is twofold, avoid grounding and collision with dynamic obstacles such as target vessels. For avoiding other vessels, it is stated by International Maritime Organization (IMO) in 2021 that the International Regulations for Preventing Collisions at Sea, 1972 (COLREGs), shall continue to serve as the framework for collision avoidance of Maritime Autonomous Surface Ships (MASS), which calls for better compliance with COLREGs when designing collision avoidance systems (Burmeister and Constapel 2021).

1.2 Related Work

1.2.1 Literature Review on Ship Collision Avoidance

Due to the difference in communication architecture, collision avoidance algorithms can first be categorized into two types, either centralized or decentralized. In the former approach, a master unit, such as Vessel Traffic Services (VTS) center, is designated to solve the collision avoidance problem for all the participating ships, while the latter relies on self-contained programs for solving the problem by utilizing available information of target ships. According to (Akdağ et al. 2022a), due to its robustness, scalability, and similarity to maritime traffic management, the decentralized collision avoidance system should always be available onboard ASVs. Therefore, only decentralized methods with COLREGs awareness are considered in this review, and three algorithms that gained considerable research attention in maritime collision avoidance are discussed in the following.

The first one is Model Predictive Control (MPC). MPC is a control strategy that optimizes the trajectory of a vehicle over a finite horizon. It is a model-based control technique that predicts the future behavior of a system based on a mathematical model and then computes control actions to achieve desired objectives. Tor A. Johansen et al. (2016) proposed an MPC method that has finite control behaviors: course angle offset and speed. Due to the limited number of control behaviors and resulting scenarios, this method is named Scenario-Based MPC (SB-MPC). To prove its effectiveness, field verification of SB-MPC is conducted in (Ku-

foalor et al. 2020), where the results demonstrate the capability of SB-MPC in computing safe control under challenging situations. Later, to improve the trajectory prediction of target ships for SB-MPC, Informed SB-MPC (Akdağ et al. 2022b) is suggested, which incorporates route exchange in the prediction procedure such that the intentions of other vessels are known. MPC could also be used for trajectory planning by only using the kinematic model of a ship. Eriksen and Breivik (2017) presents a mid-level MPC-based algorithm using nonlinear programming, which enables avoidance of both static and dynamic obstacles while minimizing the deviation to a nominal trajectory. However, this method only complies with COLREGs rule 8. This problem is overcome in (Emil H. Thyri and Breivik 2022), where the target ship domains are carefully designed and constraints related to it are formulated in the MPC, leading to COLREGs compliance of rules 13-15 and 17. Furthermore, Eriksen et al. (2019) introduced a new method called the branching-course MPC (BC-MPC). It is developed to be robust to sensor noise on obstacle detection, where full-scale experiments and simulations display satisfactory performance.

The second method is called Velocity Obstacle (VO). VO algorithms are designed to prevent collision based on finding collision-free and admissible velocity sets. Huang et al. (2018a) applies VO algorithms for collision avoidance at open sea, including Linear-VO, Nonlinear VO, and Probabilistic VO. Later, Generalized Velocity Obstacle (GVO) algorithm (Huang et al. 2018b) is proposed to account for strong hypotheses used in VO, such as the constant-velocity assumption of target ships and simplification of ship dynamics. It is claimed by the author that the GVO algorithm is more robust for close-range encounters. S. Wang et al. (2020) modified the VO algorithm by transforming the traditional VO algorithm that defines velocity obstacle as the region where velocities are completely non-selectable into an optimization problem by considering seamanship. This method is also employed in (Zhang et al. 2022), which further improves its performance by setting up a collision risk assessment method based on asymmetric grey cloud model (AGC).

The last method, Artificial Potential Field (APF), is a popular robot path planning algorithm. Its core idea is about two artificial forces - attractive force and repulsive force, generated by carefully designed potential fields. The attractive force is related to the goal position, while the repulsive force originates from obstacles. By combining these two forces, the resultant force vector implies the direction of a collision-free path. In (Song et al. 2018), a collision avoidance algorithm is presented by utilizing the VO algorithm with the improved APF method, where the APF method is employed to handle emergency situations. Different from the traditional APF method, the improved APF in this study considers two forces: the repulsive force and the novel centrifugal force. The purpose of this centrifugal component is to push the own ship toward the stern of a target ship. To ensure COLREGs compliance, Lyu and Yin (2019) modifies the APF by introducing virtual forces with additional safety considerations. Similar work is done in (Liu et al. 2023), where field tests are carried out to verify the method.

Moreover, other popular methods include Control Barrier Functions (CBF) (Emil H Thyri et al. 2020; Ellenrieder 2022; Gong et al. 2022), evolutionary algorithms like Genetic Algorithm (GA) (Tsou et al. 2010; Ning et al. 2020; H. Wang et al. 2021) and Particle Swarm Optimization (PSO) (Xia et al. 2020; Park 2021; Zheng et al. 2021), and modern AI techniques like Fuzzy logic (Perera et al. 2011; Brcko et al. 2021; Ahmed et al. 2021) and Deep Reinforcement Learning (DRL) (Zhao and Roh 2019; Meyer et al. 2020; Chun et al. 2021).

1.2.2 Literature Review on TEB Approach

Inspired by the classic path planning method called Elastic Band (EB) (Quinlan and Khatib 1993), the Timed-Elastic-Band (TEB) approach is an intuitive trajectory planning algorithm based on nonlinear programming. It is first proposed by (Roesmann et al. 2012) in 2012, which is developed as a weighted multi-objective optimization framework. This method explicitly incorporates temporal information of motions, e.g., dynamic constraints such as limited

velocities and accelerations (Roesmann et al. 2012). The decision variables consist of multiple robot states (position and orientation), and travel time in between these configurations. Despite the considerable number of decision variables for optimization, most objectives are formulated locally (Rösmann et al. 2013) where a small quantity of neighboring decision variables are concerned. This property results in a highly sparse system matrix for optimization (Rösmann et al. 2013), which could be solved efficiently by open-source frameworks such as 'g2o', a sparse system solver, developed by Kümmerle et al. (2011). Over the recent decade, the TEB planner has been developed and plugged into the ROS navigation stack. Due to its efficiency and robustness, the TEB approach is widely utilized as a local planner for differential drives (Roesmann et al. 2012), car-like robots (Rösmann et al. 2017a), manipulators (Magyar et al. 2019), etc. One major advantage of the TEB approach is its modular formulation, which enables the incorporation of additional objectives and constraints. This indicates that it has great potential to be modified to realize COLREGs-compliant behaviors.

1.3 Motivation and Contributions

Despite the availability of numerous collision avoidance algorithms for ASVs, the navigational decisions made by these algorithms have limited coverage of and compliance with the vague traffic rules at sea - COLREGs (Burmeister and Constapel 2021). Particularly, when it comes to multi-vessel encounters, there is more ambiguity. Meanwhile, most simulations only consider target ships that could execute path following without collision avoidance action, which is arguably not a real-world scenario. Additionally, most of the reviewed methods only give sub-optimal results, e.g., SB-MPC, BC-MPC, VO, and APF, which might limit the collision avoidance performance. These problems motivate the objective of this thesis to develop a decentralized and optimal collision avoidance system for ASVs with high compliance with COLREGs, and the following research questions will be addressed:

- *What should COLREGs-compliant navigational decisions of ASVs be in a multi-vessel encounter, especially when there are conflicting roles to take?*
- *How could we develop an optimal collision avoidance system for ASVs?*

Taking advantage of the computational efficiency, optimality, modularity, and flexibility of the TEB approach, the contributions of this thesis are:

1. An action selection strategy for multi-vessel encounters based on COLREGs is proposed.
2. The TEB approach is adapted and improved for collision avoidance of ASVs.
3. The effectiveness of the proposed collision avoidance system is verified by simulating the Imazu problem with additional scenarios.

1.4 Outline

The remainder of this thesis is organized as follows: Chapter 2 covers the basic knowledge about ship modeling, motion control and guidance, and Chapter 3 completes the theories by presenting the proposed collision avoidance method. To validate this method, simulations are conducted, and their results are displayed and discussed in Chapter 4. Lastly, conclusions are drawn in Chapter 5 along with insights into future work.

Chapter 2

Basic Theory

This chapter presents fundamental theories concerning the Modeling, Control, and Guidance of ASVs. Firstly, a general 6-degree-of-freedom (6-DOF) high-fidelity model of a ship is introduced, and corresponding the system matrices of the simulated model used in this project are specified. Subsequently, the motion control system is derived, consisting of Proportional-Integral-Derivative (PID) control for heading autopilot and Sliding Mode Control (SMC) for surge speed autopilot. The calculated yaw moment and surge force are then converted into propeller speed by a simple control allocation algorithm. To guide the vessel in the nominal path following, a guidance system based on reference models and integral line-of-sight (ILOS) law is demonstrated. In addition, the lookahead-based goal point is calculated for the TEB approach as part of the guidance system. The theories presented in this chapter, as well as notations, are based on the comprehensive textbook Handbook of Marine Craft Hydrodynamics and Motion Control, Second Edition (Fossen 2021a).

2.1 Ship Maneuvering Model

For the simulation purpose, a 6-DOF high-fidelity ship model is adopted. The dynamics of a ship can be divided into kinematics and kinetics, where the former treats the geometrical aspects of motion, and the latter analyzes the forces enabling the motion. Assuming that no environmental force exists (induced by wind, wave, and current), the equations of motion can be expressed as:

$$\dot{\boldsymbol{\eta}} = J_{\Theta}(\boldsymbol{\eta})\boldsymbol{\nu} \tag{2.1}$$

$$\boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} \tag{2.2}$$

where the first line is the kinematics equation, and the second is the kinetics equation. The meanings of the matrix notations used in this formula will be briefly explained in Section 2.1.1 and Section 2.1.2.

2.1.1 Kinematics

BODY-NED Transformation

The Equation 2.1 describes the transformation between the NED and BODY frames, where NED represents the North-East-Down coordinate system denoted as $\{n\} = (x_n, y_n, z_n)$ and BODY is the body-fixed reference frame $\{b\} = (x_b, y_b, z_b)$ with a moving origin o_b fixed to the ship. Specifically, it transforms the velocity vector $\boldsymbol{\nu} = [u, v, w, p, q, r]^\top$ in the BODY frame to the generalized velocity vector $\dot{\boldsymbol{\eta}} = [\dot{x}^n, \dot{y}^n, \dot{z}^n, \dot{\phi}, \dot{\theta}, \dot{\psi}]^\top$ in the NED frame. The generalized velocity vector is the time derivative of the generalized position vector $\boldsymbol{\eta} = [x^n, y^n, z^n, \phi, \theta, \psi]$ in the NED frame. We can represent NED position as $\boldsymbol{p}_{nb}^n = [x^n, y^n, z^n]^\top \in \mathbb{R}^3$, and attitude as $\boldsymbol{\Theta}_{nb} = [\phi, \theta, \psi]^\top \in \mathbb{T}^3$, where ϕ, θ, and ψ are Euler angles with a range of $[-\pi, \pi)$. We can

also decompose body-fixed velocity into linear and angular velocities as $\boldsymbol{\nu}_{nb}^{b} = [u, v, w]^{\top} \in \mathbb{R}^3$ and $\boldsymbol{\omega}_{nb}^{b} = [p, q, r]^{\top} \in \mathbb{R}^3$, where $\mathbb{R}^3$ and $\mathbb{T}^3 = \mathbb{S} \times \mathbb{S} \times \mathbb{S}$ are three-dimensional Euclidean space and torus, respectively. With these definitions, the 6-DOF kinematic Equation 2.1 can be expressed as:

$$\begin{bmatrix} \dot{\boldsymbol{p}}_{nb}^{n} \\ \dot{\boldsymbol{\Theta}}_{nb} \end{bmatrix} = \begin{bmatrix} \boldsymbol{R}(\boldsymbol{\Theta}_{nb}) & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{T}(\boldsymbol{\Theta}_{nb}) \end{bmatrix} \begin{bmatrix} \boldsymbol{\nu}_{nb}^{b} \\ \boldsymbol{\omega}_{nb}^{b} \end{bmatrix} \tag{2.3}$$

where $\boldsymbol{R}(\boldsymbol{\Theta}_{nb})$ is the linear velocity transformation matrix following ZYX convention calculated as:

$$\boldsymbol{R}(\boldsymbol{\Theta}_{nb}) = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - s\psi c\phi & c\psi s\theta c\phi + s\psi s\phi \\ s\psi c\theta & s\psi s\theta s\phi + c\psi c\phi & s\psi s\theta c\phi - c\psi s\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \tag{2.4}$$

and $\boldsymbol{T}(\boldsymbol{\Theta}_{nb})$ is the angular velocity transformation matrix given by:

$$\boldsymbol{T}(\boldsymbol{\Theta}_{nb}) = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix} \tag{2.5}$$

Note that for simplicity, $c := \cos$, $s := \sin$, and $t := \tan$. These matrices relate the body-fixed velocity to the NED velocity and angular rates through the rotational motion of the vehicle.

Course and Crab Angle

As it will be presented in Section 2.3.2, it is important to identify the difference between heading and course angle. Course angle, defined as the angle from x_n axis to the ship's velocity vector about z_n axis using right-hand convention, is computed by:

$$\chi = \psi + \beta_c \tag{2.6}$$

where ψ is interpreted as the heading angle and β_c represents the crab angle which denotes the angle from the x_b axis to the ship's velocity vector about z_b axis using right-hand convention, given by:

$$\beta_c = \sin^{-1} \frac{v}{U} \tag{2.7}$$

where $U = \sqrt{u^2 + v^2}$ is the velocity amplitude.

2.1.2 Kinetics

System Inertia Matrix

In Equation 2.2, system inertia matrix $\boldsymbol{M}$, which includes the added mass and the rigid body component, is given by:

$$\boldsymbol{M} = \boldsymbol{M}_{RB} + \boldsymbol{M}_{A} \tag{2.8}$$

By employing a transformation matrix $\boldsymbol{H}(\boldsymbol{r}_{bg}^{b})$ that converts the equations of motion about the center of gravity (CG) to the body frame origin CO, where the CG position vector is denoted by $\boldsymbol{r}_{bg}^{b} = [x_g, y_g, z_g]^{\top}$, we can express the rigid body component as:

$$\boldsymbol{M}_{RB} = \boldsymbol{H}^{\top}(\boldsymbol{r}_{bg}^{b}) \boldsymbol{M}_{GB}^{CG} \boldsymbol{H}(\boldsymbol{r}_{bg}^{b}) \tag{2.9}$$

Here, $\boldsymbol{H}(\boldsymbol{r}_{bg}^{b})$ is defined as follows:

$$\boldsymbol{H}(\boldsymbol{r}_{bg}^{b}) = \begin{bmatrix} \boldsymbol{I}_{3\times3} & \boldsymbol{S}^{\top}(\boldsymbol{r}_{bg}^{b}) \\ \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} \end{bmatrix} \tag{2.10}$$

where $\boldsymbol{S}(\cdot)$ is known as the cross-product operator, defined as:

$$\boldsymbol{S}(\boldsymbol{\lambda})\boldsymbol{a} = \boldsymbol{\lambda} \times \boldsymbol{a} \tag{2.11}$$

Moreover, $\boldsymbol{M}_{GB}^{CG}$ denotes the inertia matrix of the rigid body about the CG, given by:

$$\boldsymbol{M}_{GB}^{CG} = \begin{bmatrix} m\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{g}^{b} \end{bmatrix} \tag{2.12}$$

where m represents the mass of the body, and $\boldsymbol{I}_{g}^{b}$ denotes the inertia dyadic about its own CG. The computation of the hydrodynamic system inertia matrix, denoted by $\boldsymbol{M}_A$, is typically performed numerically using potential theory programs that integrate the pressure distribution over the wet surface of the hull. In the simulation model, we adopt an assumption of decoupling between all motions, allowing us to express the added mass matrix in a diagonal form as follows:

$$\boldsymbol{M}_A = -\text{diag}(X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}, K_{\dot{p}}, M_{\dot{q}}, N_{\dot{r}}) \tag{2.13}$$

This expression implies that the added mass coefficients are dependent solely on the rates of motion for each degree of freedom, as represented by the diagonal elements of the matrix.

Coriolis-centripetal Matrix

Likewise, the Coriolis-centripetal matrix $\boldsymbol{C}(\boldsymbol{\nu})$ is the sum of the rigid body and added mass part, namely:

$$\boldsymbol{C}(\boldsymbol{\nu}) = \boldsymbol{C}_{RB}(\boldsymbol{\nu}) + \boldsymbol{C}_A(\boldsymbol{\nu}) \tag{2.14}$$

where

$$\boldsymbol{C}_{RB}(\boldsymbol{\nu}) = \begin{bmatrix} m\boldsymbol{S}(\boldsymbol{\omega}_{nb}^{b}) & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & -\boldsymbol{S}(\boldsymbol{I}_{g}^{b}\boldsymbol{\omega}_{nb}^{b}) \end{bmatrix} \tag{2.15}$$

and

$$\boldsymbol{C}_A(\boldsymbol{\nu}) = \begin{bmatrix} \boldsymbol{0}_{3\times3} & -\boldsymbol{S}(\boldsymbol{A}_{11}\boldsymbol{\nu}_1 + \boldsymbol{A}_{12}\boldsymbol{\nu}_2) \\ -\boldsymbol{S}(\boldsymbol{A}_{11}\boldsymbol{\nu}_1 + \boldsymbol{A}_{12}\boldsymbol{\nu}_2) & -\boldsymbol{S}(\boldsymbol{A}_{21}\boldsymbol{\nu}_1 + \boldsymbol{A}_{22}\boldsymbol{\nu}_2) \end{bmatrix} \tag{2.16}$$

where $\boldsymbol{A}_{ij} \in \mathbb{R}^{3\times3}$ is derived from added mass inertia matrix by:

$$\boldsymbol{M}_A = \begin{bmatrix} \boldsymbol{A}_{11} & \boldsymbol{A}_{12} \\ \boldsymbol{A}_{21} & \boldsymbol{A}_{22} \end{bmatrix} \tag{2.17}$$

Note that $\boldsymbol{C}_A(\boldsymbol{\nu})$ is parameterized to be skew-symmetric, which means that it satisfies $\boldsymbol{C}_A(\boldsymbol{\nu}) = -\boldsymbol{C}_A^{\top}(\boldsymbol{\nu})$.

Damping Matrix

As for the damping matrix, it is commonly separated into linear and nonlinear matrices as the following:

$$\boldsymbol{D}(\boldsymbol{\nu}) = \boldsymbol{D} + \boldsymbol{D}_n(\boldsymbol{\nu}) \tag{2.18}$$

where the linear component $\boldsymbol{D}$ is mostly caused by potential damping and skin friction, and the nonlinear component $\boldsymbol{D}_n(\boldsymbol{\nu})$ is related to higher-order effects such as vortex shedding-induced damping and lifting forces. Since a surface ship is generally xz plane-symmetric, surge motion is deemed to be decoupled from steering dynamics. Hence, linear damping matrix $\boldsymbol{D}$ can be written in the following form:

$$\boldsymbol{D} = -\begin{bmatrix} X_u & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_v & 0 & Y_p & 0 & Y_r \\ 0 & 0 & Z_w & 0 & Z_q & 0 \\ 0 & K_v & 0 & K_p & 0 & K_r \\ 0 & 0 & M_w & 0 & M_q & 0 \\ 0 & N_v & 0 & N_p & 0 & N_r \end{bmatrix} \tag{2.19}$$

In the simulation model, this is further simplified by removing non-diagonal terms given by:

$$\boldsymbol{D} = -\text{diag}(X_u, Y_v, Z_w, K_p, M_q, N_r) \tag{2.20}$$

Nonlinear damping is mostly considered by modeling nonlinear surge damping and applying the cross-flow drag principle for nonlinear damping in sway and yaw for a surface ship. In the simulation model, only sway and yaw nonlinear damping is considered, which is calculated by the following expression disregarding current:

$$Y = -\frac{1}{2}\rho \int_{-\frac{L_{pp}}{2}}^{\frac{L_{pp}}{2}} T(x) C_d^{2D}(x) |v + xr|(v + xr) dx \tag{2.21}$$

$$N = -\frac{1}{2}\rho \int_{-\frac{L_{pp}}{2}}^{\frac{L_{pp}}{2}} T(x) C_d^{2D}(x) x |v + xr|(v + xr) dx \tag{2.22}$$

where $C_d^{2D}(x)$ is the 2D drag coefficient that could be estimated by Hoerner's curve and $T(x)$ is the draft. By curve fitting Equation 2.21 and 2.22 cross-flow drag integrals to second-order modulus functions, we could obtain the expression for $\boldsymbol{D}_n(v)$:

$$\boldsymbol{D}_n(v) = -\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_{|v|v}|v| + Y_{|r|v}|r| & 0 & 0 & 0 & Y_{|v|r}|v| + Y_{|r|r}|r| \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & N_{|v|v}|v| + N_{|r|v}|r| & 0 & 0 & 0 & N_{|v|r}|v| + N_{|r|r}|r| \end{bmatrix} \tag{2.23}$$

Gravitational/Buoyancy Force

Due to the fact that a conventional surface vessel has a small heave, roll and pitch motion during nominal operation, we could use linear theory to approximate restoring force caused by gravity and buoyancy, which means:

$$\boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{G}\boldsymbol{\eta} = \begin{bmatrix} 0 \\ 0 \\ \rho g A_{wp} z^n \\ \rho g \nabla \text{GM}_\text{T} \phi \\ \rho g \nabla \text{GM}_\text{L} \theta \\ 0 \end{bmatrix} \tag{2.24}$$

where z^n denotes the displacement in heave of the center of flotation (CF) located at $\boldsymbol{r}_{bf}^b$ relative to the CO, A_{wp} is the waterplane area, ∇ stands for displaced water volume, and $\mathrm{GM_T}/\mathrm{GM_L}$ represents transverse/longitudinal metacentric height. Thus, we have:

$$\boldsymbol{G}^{CF} = \mathrm{diag}(0, 0, c, \rho g \nabla \mathrm{GM_T}, \rho g \nabla \mathrm{GM_L}, 0) \tag{2.25}$$

where the superscript CF indicates that the matrix is expressed in the CF. Similar to Equation 2.9, $\boldsymbol{G}$ should be transformed from $\boldsymbol{G}_{CF}$ with CF distance vector $\boldsymbol{r}_{bf}^b = [\mathrm{LCF}, 0, 0]^\top$, that is:

$$\boldsymbol{G} = \boldsymbol{H}^\top(\boldsymbol{r}_{bf}^b)\boldsymbol{G}^{CF}\boldsymbol{H}(\boldsymbol{r}_{bf}^b) \tag{2.26}$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \rho g A_{wp} & 0 & -\rho g A_{wp}\mathrm{LCF} & 0 \\ 0 & 0 & 0 & \rho g \nabla \mathrm{GM_T} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\rho g A_{wp}\mathrm{LCF} & 0 & \rho g(A_{wp}\mathrm{LCF}^2 + \nabla \mathrm{GM_L}) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.27}$$

Control Force

Finally, the control force that propels and steers the ship, denoted by $\boldsymbol{\tau}$, can be expressed as a sum of generalized forces $\boldsymbol{\tau}_i$ in 6 DOFs, namely $\boldsymbol{\tau} = \sum \boldsymbol{\tau}_i$. Specifically, the i^{th} component corresponding to a thrust vector $\boldsymbol{f}_{ti}^b = [F_{xi}, F_{yi}, F_{zi}]^\top$ expressed in the body-fixed frame {b} can be written as:

$$\boldsymbol{\tau}_i = \begin{bmatrix} \boldsymbol{f}_{ti}^b \\ \boldsymbol{r}_{ti}^b \times \boldsymbol{f}_{ti}^b \end{bmatrix} = \begin{bmatrix} F_{xi} \\ F_{yi} \\ F_{zi} \\ l_{yi}F_{zi} - l_{zi}F_{yi} \\ l_{zi}F_{xi} - l_{xi}F_{zi} \\ l_{xi}F_{yi} - l_{yi}F_{xi} \end{bmatrix} \tag{2.28}$$

where $\boldsymbol{r}_{ti} = [l_{xi}, l_{yi}, l_{zi}]^\top$ represents the lever arm of the thruster with respect to the CO. In the adopted simulation model, two main propellers are installed at the stern of the ship, and thus, the thrust vector could be rewritten as $\boldsymbol{f}_{ti}^b = [T_i, 0, 0]^\top$. Consequently, we have:

$$\boldsymbol{\tau}_i = \begin{bmatrix} T_i & 0 & 0 & 0 & 0 & -l_{yi}T_i \end{bmatrix}^\top, \quad i = 1, 2 \tag{2.29}$$

where thrust force T_i is calculated by thrust coefficient K and propeller revolution speed $n_i \in [n_{\min}, n_{\max}]$, given by:

$$T_i = K n_i |n_i|, \quad i = 1, 2 \tag{2.30}$$

Therefore, we can express the control force as:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_1 + \boldsymbol{\tau}_2 \tag{2.31}$$

$$= \begin{bmatrix} T_1 + T_2 & 0 & 0 & 0 & 0 & -l_{y1}T_1 - l_{y2}T_2 \end{bmatrix}^\top \tag{2.32}$$

Next, Section 2.2 will present the computation of control commands of propeller speed for autopilot of heading ψ and surge speed u.

2.2 Motion Control

Section 2.1 introduces a maneuvering model that serves as the foundation for model-based control. In this section, decoupled heading and speed controllers for planar motion are developed using two types of control laws: PID and SMC. The controllers calculate the desired surge force and yaw moment, which are then used as input to the thrust allocation module. The thrust allocation module then generates corresponding propeller speed commands. Note that uncontrolled motions (heave, sway, roll, and pitch) are assumed to be relatively small, such that they have limited effect on controlled motions.

2.2.1 Heading PID Control

Consider the linear yaw model derived from Section 2.1.2:

$$(I_z - N_{\dot{r}})\ddot{\psi} - N_r\dot{\psi} = \tau_N + d_N \tag{2.33}$$

where I_z is the moment of inertia about z axis, τ_N is the control force in yaw, and d_N represents unmodeled dynamics. Denoting $m := I_z - N_{\dot{r}}$, $d := -N_r$, $x := \psi$ and $\tilde{x} = \text{ssa}(\psi - \psi_d)$ (ssa$(\cdot)$ is an operator that maps the unconstrained input representing the difference between two angles, e.g, $\psi - \psi_d$ into their smallest difference in range $[-\pi, \pi)$), the PID control law is designed by the following:

$$\tau_N = -K_p\tilde{x} - K_d\dot{x} - K_i\int_0^t \tilde{x}(\tau)d\tau \tag{2.34}$$

where proportional gain K_p and derivative gain K_d are determined by design variables ω_n and ζ, given by:

$$K_p = m\omega_n^2 \tag{2.35}$$

$$K_d = 2\zeta\omega_n m - d \tag{2.36}$$

For integral gain, a rule-of-thumb suggests that the integrator is $1/10$ as the natural frequency ω_n, which indicates that:

$$K_i = \frac{\omega_n}{10}K_p \tag{2.37}$$

Consequently, the closed-loop system becomes a stable mass-damper-spring system:

$$\ddot{x} + 2\zeta\omega_n\dot{x} + \omega_n^2\tilde{x} = 0 \tag{2.38}$$

Here, the integral term was canceled out by the d_N term in the ideal case.

2.2.2 Speed Sliding Mode Control

Consider the linear surge equation derived earlier:

$$(m - X_{\dot{u}})\dot{u} - X_u u = \tau_X + d_X \tag{2.39}$$

where τ_X is the control force in surge, and d_X is the unmodeled dynamics. Let the sliding variable be:

$$\sigma = (u - u_d) + \int_0^t (u - u_d)d\tau \tag{2.40}$$

where u_d is the desired velocity. Thus, the time derivative of σ is:

$$\dot{\sigma} = \dot{u} - \dot{u}_d + u - u_d \tag{2.41}$$

$$= \frac{1}{m - X_{\dot{u}}}(\tau_X + X_u u) - \dot{u}_d + u - u_d \tag{2.42}$$

Let control Lyapunov function be $V = \frac{1}{2}\sigma^2$, by differentiation we have:

$$\dot{V} = \sigma\dot{\sigma} \tag{2.43}$$

$$= \sigma(\frac{1}{m - X_{\dot{u}}}(\tau_X + X_u u) - \dot{u}_d + u - u_d) \tag{2.44}$$

To reach the negative definiteness of $\dot{V}$, we could select SMC control law as the following:

$$\tau_X = (m - X_{\dot{u}})(\dot{u}_d + u_d - u - \frac{1}{m - X_{\dot{u}}}X_u u - K_\sigma \tanh\frac{\sigma}{\phi}) \tag{2.45}$$

where gain $K_\sigma \gg |d_X|$, $\phi > 0$ is a design parameter, and $\tanh(\cdot)$ is adopted in replace of sign function $\text{sgn}(\cdot)$ to avoid chattering for large values of K_σ caused by switching term $K_\sigma \text{sgn}(\sigma)$, which yields the following Lyapunov function:

$$\dot{V} = -K_\sigma \tanh\frac{\sigma}{\phi}\sigma + d_X\sigma < 0, \qquad \forall\sigma \neq 0 \tag{2.46}$$

Hence, it could be concluded from Equation 2.46 that the closed-loop surge system is globally asymptotically stable.

2.2.3 Thrust Allocation

From Equation 2.31 we can derive the following expression through decomposition:

$$\boldsymbol{\tau} = \boldsymbol{B}\boldsymbol{K}\boldsymbol{u} \tag{2.47}$$

where $\boldsymbol{\tau} = [\tau_X, \tau_N]^\top$ is the desired control vector in surge and yaw, $\boldsymbol{K} = \text{diag}(K, K)$ denotes the thrust coefficient matrix, $\boldsymbol{u} = [u_1, u_2]^\top = [n_1|n_1|, n_2|n_2|]^\top$ is the square of propeller speed with revolution direction, $\boldsymbol{B} \in \mathbb{R}^{2\times 2}$ is the thrust configuration matrix given by:

$$\boldsymbol{B} = \begin{bmatrix} 1 & 1 \\ -l_{y1} & -l_{y2} \end{bmatrix} \tag{2.48}$$

Note that the rank of $\boldsymbol{B}$ is 2, which is less than 3, indicating that the vessel is under-actuated, where only surge and yaw direction can be controlled. To obtain $\boldsymbol{u}$, we can invert $\boldsymbol{B}\boldsymbol{K}$ as follows:

$$\boldsymbol{u} = \boldsymbol{K}^{-1}\boldsymbol{B}^{-1}\boldsymbol{\tau} \tag{2.49}$$

Subsequently, corresponding propeller speed commands are derived using the following formulae:

$$n_1 = \text{sgn}(u_1)\sqrt{|u_1|} \tag{2.50}$$

$$n_2 = \text{sgn}(u_2)\sqrt{|u_2|} \tag{2.51}$$

2.3 Guidance

In this section, an open-loop guidance system will be presented with several functionalities, including trajectory generation using reference models for surge speed and yaw angle, integral line-of-sight (ILOS) guidance law that provides heading autopilot reference for straight-line path following, and computation of lookahead-based local goal point (as opposed to global waypoints) for TEB approach that will be introduced in Section 3.4. The first three components of the guidance system are used in path following mode, whereas the last component is called after switching to collision avoidance.

2.3.1 Reference Models

Velocity Reference Model

The surge speed reference model is designed based on a second-order low-pass filter (LP) (or equivalently, mass-spring-damper system) to output a smooth signal of desired speed u_d. Denoting u_{ref} as the step input, we can write the expression in s domain:

$$\frac{u_d}{u_{\text{ref}}}(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}, \tag{2.52}$$

where ζ represents the relative damping ratio, and ω_n is the natural frequency. Alternatively, the reference model can be represented in the time domain as:

$$\ddot{u}_d + 2\zeta\omega_n\dot{u}_d + \omega_n^2 u_d = \omega_n^2 u_{\text{ref}}, \tag{2.53}$$

Defining $a_d := \dot{u}_d$ as the desired acceleration, $j_d := \ddot{u}_d$ as the desired "jerk", Equation 2.53 becomes:

$$j_d + 2\zeta\omega_n a_d + \omega_n^2 u_d = \omega_n^2 u_{\text{ref}} \tag{2.54}$$

Then, the expression for j_d is found as:

$$j_d = \omega_n^2(u_{\text{ref}} - u_d) - 2\zeta\omega_n a_d \tag{2.55}$$

Due to the dynamic limitation of the ship such as surge speed and its acceleration, saturating elements should be added to the reference model to prevent generating physically infeasible reference signals. In this study, Euler's Method is chosen for solving ordinary differential equations (ODEs) as shown in Equation 2.53. Let T_s denote the sample time and t_k as the time at k^{th} step which satisfies $t_{k+1} = t_k + T_s$, the overall surge speed reference model updating the values of u_d and a_d is implemented as the following:

Algorithm 1 Surge Speed Reference Model with Saturating Elements

Input: u_d, a_d at t_k

$u_d \leftarrow \text{sgn}(u_d) \cdot \min(u_{\max}, |u_d|)$
$a_d \leftarrow \text{sgn}(a_d) \cdot \min(a_{\max}, |a_d|)$
$j_d \leftarrow \omega_n^2(u_{\text{ref}} - u_d) - 2\zeta\omega_n a_d$
$u_d \leftarrow u_d + T_s a_d$
$a_d \leftarrow a_d + T_s j_d$
return u_d, a_d at t_{k+1}

where $\min(x, y)$ is maps the input variables x and y to the smallest value among them, $u_{\max}$ and $a_{\max}$ are the maximum acceptable desired surge speed and acceleration.

Attitude Reference Model

Typically, attitude or position reference models are chosen to be of third order for trajectory generation, and a cascade of the second-order LP filter (see Equation 2.52) with a first-order LP filter would satisfy the requirement. Since a first-order LP filter can be formulated as:

$$H(s) = \frac{1}{1 + Ts} \tag{2.56}$$

where $T = 1/\omega_n > 0$ is a time constant, the transfer function of the heading reference model could be written as:

$$\frac{\psi_d}{\psi_{\text{ref}}} = \frac{\omega_n^2}{(1 + Ts)(s^2 + 2\zeta\omega_n s + \omega_n^2)} \tag{2.57}$$

$$= \frac{\omega_n^3}{s^3 + (2\zeta + 1)\omega_n s^2 + (2\zeta + 1)\omega_n^2 s + \omega_n^3} \tag{2.58}$$

In the time domain, this expression is transformed to be:

$$\psi_d^{(3)} + (2\zeta+1)\omega_n\ddot{\psi}_d + (2\zeta+1)\omega_n^2\dot{\psi}_d + \omega_n^3\psi_d = \omega_n^3\psi_{\text{ref}} \tag{2.59}$$

Defining $r_d := \dot{\psi}_d$ as the desired angular velocity, $\alpha_d := \ddot{\psi}_d$ as the desired angular acceleration and $j_d := \psi_d^{(3)}$ as the desired angular "jerk", Equation 2.59 becomes:

$$j_d + (2\zeta+1)\omega_n\alpha_d + (2\zeta+1)\omega_n^2 r_d + \omega_n^3\psi_d = \omega_n^3\psi_{\text{ref}} \tag{2.60}$$

Therefore, the expression for j_d is found as:

$$\begin{aligned} j_d &= \omega_n^3(\psi_{\text{ref}} - \psi_d) - (2\zeta+1)\omega_n\alpha_d - (2\zeta+1)\omega_n^2 r_d && (2.61)\\ &= \omega_n^3(\psi_{\text{ref}} - \psi_d) - (2\zeta+1)(\alpha_d + \omega_n r_d)\omega_n && (2.62)\end{aligned}$$

However, since the range of values for $\psi_{\text{ref}} - \psi_d$ is limited to the interval $[-\pi, \pi)$, it is necessary to modify Equation 2.61 and include the $\text{ssa}(\cdot)$ operator to handle the angular difference term appropriately. Thus, j_d is reformulated as:

$$j_d = \omega_n^3\text{ssa}(\psi_{\text{ref}} - \psi_d) - (2\zeta+1)(\alpha_d + \omega_n r_d)\omega_n \tag{2.63}$$

Similarly, dynamic constraints apply to the desired yaw rate as well as yaw acceleration, and saturating elements should be included in the heading reference model. The updating model representing the equations above is shown as the following:

Algorithm 2 Heading Reference Model with Saturating Elements

Input: ψ_d, r_d, α_d at t_k

$r_d \leftarrow \text{sgn}(r_d) \cdot \min(r_{\max}, |r_d|)$
$\alpha_d \leftarrow \text{sgn}(\alpha_d) \cdot \min(\alpha_{\max}, |\alpha_d|)$
$j_d \leftarrow \omega_n^3\text{ssa}(\psi_{\text{ref}} - \psi_d) - (2\zeta+1)(\alpha_d + \omega_n r_d)\omega_n$
$\psi_d \leftarrow \text{ssa}(\psi_d + T_s r_d)$
$r_d \leftarrow r_d + T_s\alpha_d$
$\alpha_d \leftarrow \alpha_d + T_s j_d$
return ψ_d, r_d, α_d at t_{k+1}

where $r_{\max}$ and $\alpha_{\max}$ are the maximum acceptable desired yaw rate and acceleration.

2.3.2 Integral Line-Of-Sight Guidance Law

LOS and ILOS

Line-Of-Sight (LOS) guidance law is an effective method for computing reference course angle χ_{ref} for path following, defined as the task of following a predefined path without temporal considerations. The formula for the LOS guidance law is:

$$\chi_{\text{ref}} = \pi_p - \tan^{-1}(K_p y_e^p) \tag{2.64}$$

where π_p is the path-tangential angle determined by positions of two consecutive waypoints in straight-line path following, $K_p = \frac{1}{\Delta} > 0$ is the proportional gain parameterized by the look-ahead distance Δ, and y_e^p denotes the lateral distance from the path or equivalently cross-track error. If we express the law with respect to heading reference for its autopilot by expanding χ_{ref}, Equation 2.64 becomes:

$$\psi_{\text{ref}} = \pi_p - \tan^{-1}(K_p y_e^p) - \beta_c \tag{2.65}$$

where the term β_c indicates that measurement of both surge and sway velocity should be available for computation. Alternatively, the crab angle can be regarded as a disturbance to be compensated such that no velocity measurement is needed for this law. In this sense, it is reasonable to add an integral term along with the proportional term to eliminate the disturbance, that is:

$$\psi_{\text{ref}} = \pi_p - \tan^{-1}(K_p y_e^p + K_i \int_0^t y_e^p(\tau) d\tau) \tag{2.66}$$

where $K_i > 0$ is the integral gain. Moreover, another advantage of using ILOS is that its integral term can help remove steady-state offsets due to kinematic modeling errors caused by rolling and pitching motions.

Cross-track Error of Straight-line Path

To compute the path-tangential angle π_p used in Equation 2.66 for a straight line from starting waypoint $\boldsymbol{p}_i = (x_i^n, y_i^n)$ to target waypoint $\boldsymbol{p}_{i+1} = (x_{i+1}^n, y_{i+1}^n)$, the following equation is used:

$$\pi_p = \text{atan2}(y_{i+1}^n - y_i^n, x_{i+1}^n - x_i^n) \tag{2.67}$$

A moving path-tangential reference frame $\{p\}$ is defined to have its origin $\boldsymbol{o}_p = (x_p^n, y_p^n)$ located at the path, such that the following expression is satisfied:

$$(\boldsymbol{p}_{i+1} - \boldsymbol{p}_i) \cdot (\boldsymbol{p} - \boldsymbol{o}_p) = 0 \tag{2.68}$$

where $\boldsymbol{p} = (x^n, y^n)$ denotes the position of the ship. This indicates that the cross-track error y_e^p is given by:

$$\begin{bmatrix} 0 \\ y_e^p \end{bmatrix} = \boldsymbol{R}_p^n(\pi_p)^\top \left(\begin{bmatrix} x^n \\ y^n \end{bmatrix} - \begin{bmatrix} x_p^n \\ y_p^n \end{bmatrix} \right) \tag{2.69}$$

where

$$\boldsymbol{R}_p^n(\pi_p) = \begin{bmatrix} \cos(\pi_p) & -\sin(\pi_p) \\ \sin(\pi_p) & \cos(\pi_p) \end{bmatrix} \tag{2.70}$$

Note that due to the choice of the origin $\boldsymbol{o}_p$ described above, the along-track error x_e^p is constantly 0. Additionally, utilizing the fact that

$$\tan(\pi_p) = \frac{y_{i+1}^n - y_p^n}{x_{i+1}^n - x_p^n} \tag{2.71}$$

three equations with three unknowns can be set up, described by the linear matrix equation:

$$\underbrace{\begin{bmatrix} \cos(\pi_p) & \sin(\pi_p) & 0 \\ -\sin(\pi_p) & \cos(\pi_p) & 1 \\ \tan(\pi_p) & -1 & 0 \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} x_p^n \\ y_p^n \\ y_e^p \end{bmatrix}}_{\boldsymbol{x}} = \underbrace{\begin{bmatrix} \cos(\pi_p)x^n + \sin(\pi_p)y^n \\ -\sin(\pi_p)x^n + \cos(\pi_p)y^n \\ \tan(\pi_p)x_{i+1}^n - y_{i+1}^n \end{bmatrix}}_{\boldsymbol{b}} \tag{2.72}$$

Finally, to obtain the origin $\boldsymbol{o}_p$ of the path-tangential reference frame and the cross-track error y_e^n, we can simply solve Equation 2.72 by inverting $\boldsymbol{A}$ matrix, which is $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$. Here, the calculation result of $\boldsymbol{o}_p = (x_p^n, y_p^n)$ is used in the next section for determining the local goal point denoted by $\boldsymbol{\eta}_g = [x_g^n, y_g^n, \pi_p]^\top$.

2.3.3 Lookahead-based Goal Point Selection

The TEB approach requires the user to provide a starting point η_s and a goal point η_g as the scope of trajectory planning. Similar to the lookahead-based LOS, one choice is to specify a fixed lookahead distance Δ_g for the η_g along the straight-line path. In the context of path-tangential reference frame $\{p\}$, it is described as:

$$\boldsymbol{p}_g^p = (x_{ge}^p, y_{ge}^p) \equiv (\Delta_g, 0) \tag{2.73}$$

where $\boldsymbol{p}_g^p$ is the position of the goal point expressed in $\{p\}$ frame. To proceed, the coordinates of $\boldsymbol{p}_g$ is obtained using the following expression:

$$\boldsymbol{p}_g = \boldsymbol{R}_p^n(\pi_p) p_g^p + \boldsymbol{o}_p \tag{2.74}$$

Since the goal point preferably aligns with the given path as is depicted in Figure 2.1, the goal point $\boldsymbol{\eta}_g$ is then given by:

$$\boldsymbol{\eta}_g = \begin{bmatrix} \boldsymbol{p}_g \\ \pi_p \end{bmatrix} \tag{2.75}$$

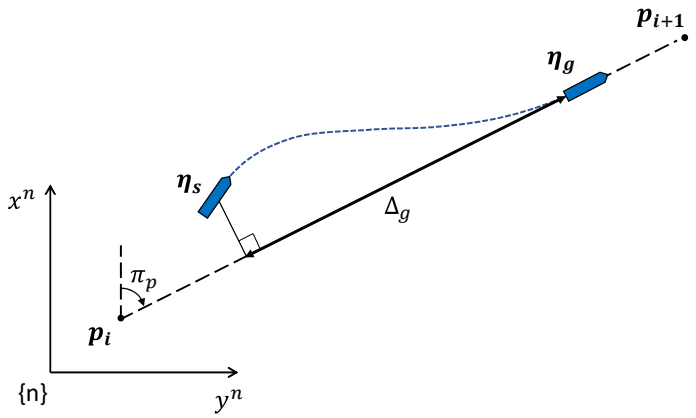


Figure 2.1: Illustration of Lookahead-based Goal Point

However, when the distance between the moving origin $\boldsymbol{o}_p$ and the next waypoint $\boldsymbol{p}_{i+1}$ is less than the threshold Δ_g, which indicates that the ship is approaching its next global goal point, it is recommended to fix the local goal point position $\boldsymbol{p}_g$ at $\boldsymbol{p}_{i+1}$. This strategy can improve the ship's waypoint tracking performance by ensuring that it closely follows the planned global goal.

2.3.4 Waypoint Switching based on Circle of Acceptance

When navigating through a path composed of $n-1$ line segments connected by n waypoints, a certain mechanism is required to determine when to select the next waypoint. One way to accomplish this is by checking if the vessel is positioned within a circular region with radius R centered at the target waypoint $\boldsymbol{p}_{i+1}$. Specifically, if the current position of the vessel at time t satisfies the inequality:

$$\|\boldsymbol{p} - \boldsymbol{p}_{i+1}\| \leq R_{\text{switch}} \tag{2.76}$$

then the next waypoint is selected. In this study, the value of R_{switch} is set to be ten times the length of the ship, that is:

$$R_{\text{switch}} = 10L \tag{2.77}$$

Chapter 3

Collision Avoidance Method

This chapter introduces a comprehensive four-step collision avoidance method. Firstly, a criterion for switching between path following and collision avoidance modes is presented, which is based on two parameters: Time to the Closest Point of Approach and Distance at the Closest Point of Approach. Upon entering collision avoidance mode, the second step identifies the encounter scenarios within the COLREGs framework and determines the appropriate steering direction, such as a starboard turn. The third step relies on a motion tracking system to capture the positions, headings, and velocities of target ships, which are then utilized by a motion prediction module to project the positions of the target ships into the future time. The predicted positions are treated as static obstacles in the TEB approach. Based on a multi-objective function, the last step is to solve an approximate least-squares optimization problem iteratively and generate an optimal collision-free trajectory that is dynamically feasible. Finally, within the planning execution time or update time (1 second in this thesis), a planning command is derived from the optimal trajectory, consisting of desired surge speed, surge acceleration, yaw angle, yaw rate, and yaw acceleration, to ensure safe and COLREGs-compliant navigation.

3.1 TCPA/DCPA-based Mode Switching

One common approach for switching between path following and collision avoidance modes is based on the Closest Point of Approach (CPA) and its two derivatives, Time to CPA (TCPA) and Distance at CPA (DCPA). These concepts are illustrated in Figure 3.1, where the blue and red ship represents own ship and target ship that travel at different speeds. CPA is the closest possible point of the two ships would meet in the future time, supposing that their course and speed are well maintained. Let $\boldsymbol{p} = (x^n, y^n)$ and $\boldsymbol{v} = [\dot{x^n}, \dot{y^n}]^\top$ represent the North-East position and velocity for own ship, while the notations for the target ship or any other dynamic obstacle are $\boldsymbol{p}_o$ and $\boldsymbol{v}_o$. TCPA is calculated using the formula described in (Bergman et al. 2020):

$$\text{TCPA} = \begin{cases} 0 & \text{if } \|\boldsymbol{v} - \boldsymbol{v_o}\| \leq \epsilon \\ \dfrac{(\boldsymbol{p}_o - \boldsymbol{p}) \cdot (\boldsymbol{v} - \boldsymbol{v_o})}{\|\boldsymbol{v} - \boldsymbol{v_o}\|^2} & \text{otherwise} \end{cases} \in (-\infty, \infty) \tag{3.1}$$

where $\epsilon > 0$ is a small value. Subsequently, DCPA is simply computed as:

$$\text{DCPA} = \|(\boldsymbol{p} - \boldsymbol{p}_o) + (\boldsymbol{v} - \boldsymbol{v}_o) \cdot \text{ TCPA}\| \geq 0 \tag{3.2}$$

Similar to the switching mechanism described in (Bergman et al. 2020), the mode is switched from path following to collision avoidance when the following condition is satisfied:

$$(\text{DCPA} \leq d_{\text{switch}}) \ \wedge \ (t_{\text{lower}} \leq \text{TCPA} \leq t_{\text{upper}}) \tag{3.3}$$

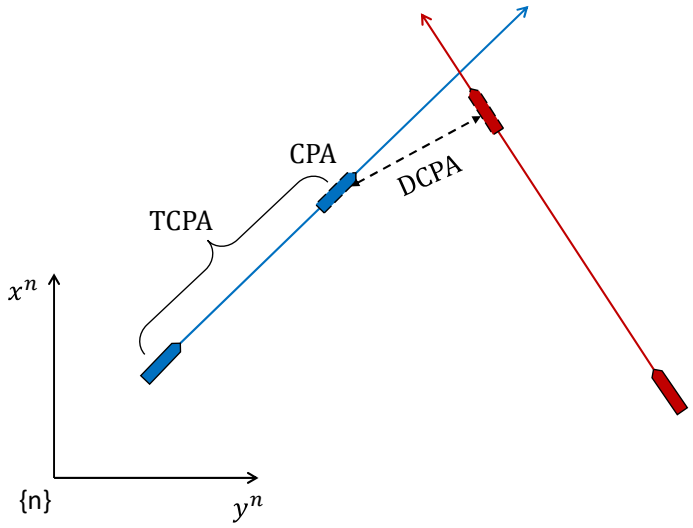


Figure 3.1: Illustration of CPA, TCPA and DCPA

where $d_{\text{switch}} > 0$ is the threshold for DCPA that triggers the mode switch, while t_{lower} and $t_{\text{upper}} > 0$ denote the lower and upper bounds of TCPA for the switch. Typically, t_{lower} is set to zero as only CPA at the current or future time is of concern. However, to prevent rapid mode switching in some marginal scenarios, an anti-hysteresis technique is used for a more 'difficult' switch-back to path following. The condition for the switchback, with anti-hysteresis augmentation, is given by (Bergman et al. 2020):

$$\neg(\text{DCPA} \leq d_{\text{switch}} + d_{\text{hyst}}) \ \vee \ \neg(t_{\text{lower}} - t_{\text{hyst}} \leq \text{TCPA} \leq t_{\text{upper}} + t_{\text{hyst}}) \tag{3.4}$$

where $d_{\text{hyst}} > 0$ and $t_{\text{hyst}} > 0$ are parameters related to hysteresis, which aims to prolong the collision avoidance mode during normal encounters.

3.2 COLREGs

The International Regulations for Preventing Collisions at sea (IMO 1972), also known as COLREGs, provide a framework for safe navigation and collision avoidance at sea. Its rules are designed to ensure that vessels maintain a proper lookout, operate at safe speeds, and take appropriate actions to avoid collisions. In Section 3.2.1, specific rules that are most relevant for collision avoidance are discussed. To comply with these rules during collision avoidance, they must be mathematically modeled and give proper steering action to be taken by the ship, which will be discussed in Section 3.2.2.

3.2.1 COLREGs Rules

Rule 2 - Responsibility: Every vessel has the responsibility to take action to avoid the risk of collision and the dangers of navigation. Departure from the rules is allowed if that is the only way to avoid a potential collision.
Rule 6 - Safe Speed: Every vessel must navigate at a safe speed, considering the visibility, traffic density, maneuverability of the vessel, and prevailing weather conditions. The vessel must be able to stop within a distance appropriate to the prevailing circumstances and conditions.
Rule 8 - Action to Avoid Collision: Every vessel must take early and substantial action to avoid a collision. The action must be positive and follow the COLREGs, taken in ample time.
Rule 13 - Overtaking: When vessels are in an overtaking situation, the overtaking vessel must keep out of the way of the vessel being overtaken and maintain a safe distance from it. The

vessel being overtaken must maintain its course and speed, and it must not alter its course until the overtaking vessel has passed and is clear.
Rule 14 - Head On: When vessels are in a head-on situation, both vessels must alter their course to starboard so that they pass each other on their port side. Both vessels must maintain their course and speed until they are clear of each other. In this situation, the risk of collision is high, and both vessels must take immediate action to avoid a collision.
Rule 15 - Crossing: When vessels are in a crossing situation, the vessel that has the other vessel on its starboard side must keep out of the way of the other vessel. The vessel on the port side must maintain its course and speed, and the vessel on the starboard side must take early and substantial action to keep clear of the other vessel.
Rule 16 - Action by Give-Way Vessel: When a vessel is required to keep out of the way of another vessel, it must take early and substantial action to avoid a collision in ample time, preferably turning to starboard. The give-way vessel must consider the other vessel's course, speed, and maneuverability.
Rule 17 - Action by Stand-On Vessel: When a vessel has the right of way, it must maintain its course and speed. The stand-on vessel must keep a lookout and be prepared to take action if the give-way vessel does not take appropriate action. The stand-on vessel must also avoid unnecessary course or speed changes, which could confuse the other vessel.

According to Rule 13-17 described here, Figure 3.2 further visualizes the encounter types and their corresponding actions that the own ship should take in a conventional two-vessel encounter situation, where the blue and red ships represent the own ship and target ship, respectively.

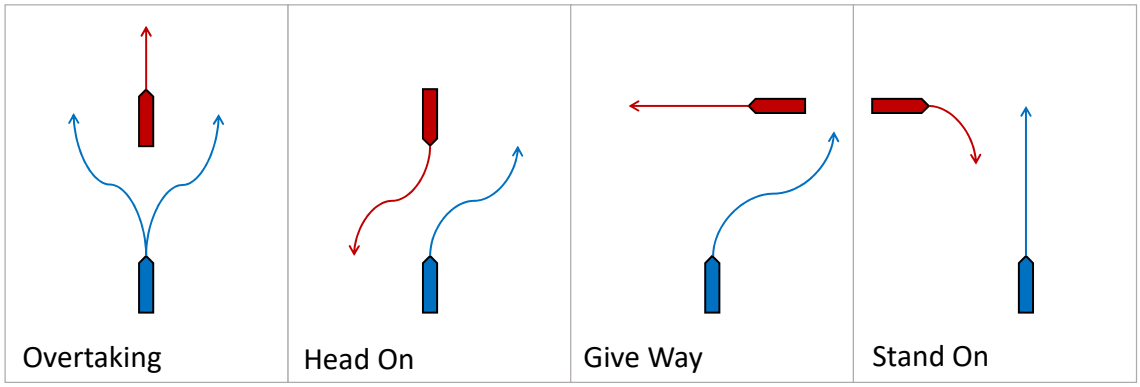


Figure 3.2: Illustration of COLREGs Encounter Types and Actions by the Own Ship

3.2.2 Mathematical Representation

Rule 2: This rule does not have a definite mathematical interpretation, but it indicates that when a target ship is in dangerously close proximity:

$$\|\boldsymbol{p}_o - \boldsymbol{p}\| \leq d_{\text{close}} \tag{3.5}$$

The vessel should always take substantial actions to prevent the potential collision whichever encounter situation the two vessels are in.
Rule 6: This rule requires that the speed of the ship should be lower than some critical value that might be dangerous depending on the situation, that is:

$$\|\boldsymbol{v}\| \leq v_{\text{critical}} \tag{3.6}$$

where $\boldsymbol{v}$ is the previously used notation for velocity in the North-East plane. Moreover, the stopping distance should be under some appropriate value which could be approximated by:

$$d_{\text{stop}} = \frac{\|\boldsymbol{v}\|^2}{2a_{\text{max}}} \leq d_{\text{critical}} \tag{3.7}$$

where d_{stop} denotes the stopping distance, and $a_{\max} > 0$ is the maximum deceleration. The first expression is satisfied by simply imposing a proper speed limit on the vessel, and therefore, the second one is only related to the propulsion system property.

Rule 8: To reach the goal of taking early and substantial action prior to a potential collision, it is required that the motion predictor (see Section 3.3) would over-predict the obstacles motions for an extended period with reference to the planning execution time of a single trajectory plan. Through a process of trial and error, this study has found that the following relationship yields satisfactory outcomes:

$$T_{\text{predict}} = 10T_{\text{exe}} \tag{3.8}$$

This indicates that if planning actions are updated every second, a dynamic obstacle will occupy a continuous two-dimensional space that spans the current and future motion for a duration of ten seconds.

Rule 13-15: These rules consist of 3 encounter scenarios: overtaking, head on and crossing, but no numerical parameters are prescribed for each. Before assigning appropriate relations, two relevant concepts are introduced. The first one is relative bearing β_r between the own ship and the target ship i, given by:

$$\beta_r = \text{ssa}(\text{atan2}(y_o^n - y^n, x_o^n - x^n) - \psi) \ \in [-\pi, \pi) \tag{3.9}$$

Here, the ssa operator is used here due to the form of angular difference. The next angle between the own ship and target ship i is called the relative course angle, denoted as χ_{r} and formulated as:

$$\chi_r = \text{ssa}(\chi - \chi_o) \in [-\pi, \pi) \tag{3.10}$$

where χ_o denotes the relative course angle of the target. This angle could be approximated by relative heading angle $\psi_r = \text{ssa}(\psi - \psi_o)$ if sway velocity is negligible.

In this thesis, the COLREGs classification method is based on (E. Thyri and Breivik 2022), while the major difference is that the encounter types are enhanced with type CS, meaning Close. It has the same classification criteria as type Safe or Stand On regarding relative bearing and relative course but is chosen when the relative distance falls below a threshold d_{close}. Motivated by Rule 2 as discussed above, this extra type sets the ship in collision avoidance mode due to inadequate space between ships, even if the encounter is classified to be Safe or the own ship is allowed to stand on. This modification takes relative distance into account and forces the stand-on ship to take action when the Give Way ship does not cooperate. In the first step, a relative bearing sector (RBS) is chosen based on the calculated β_r among the predefined RBSs, R1 to R4:

$$\text{RBS} = \begin{cases} \text{R1} & \text{if } \ \beta_r \in (-\frac{\pi}{8}, \frac{\pi}{8}] \\ \text{R2} & \text{if } \beta_r \in (\frac{\pi}{8}, \frac{5\pi}{8}] \\ \text{R3} & \text{if } \beta_r \in (\frac{5\pi}{8}, \pi) \cup [-\pi, -\frac{5\pi}{8}) \\ \text{R4} & \text{if } \beta_r \in [-\frac{5\pi}{8}, \frac{\pi}{8}) \end{cases} \tag{3.11}$$

Each RBS contains different situation sectors (SSs), which are subsets of 6 encounter types: Overtaking (OT), Head On (HO), Give Way (GW), Stand On (SO), Safe (SF), and CS (Close). The selection of SS inside a particular RBS is determined by both β_r and χ_r. The angular ranges of SS are divided in the same fashion as RBS with a rotation $\beta_r - \pi$, such that the position vector from the target ship to the own ship points towards the original zero-degree position of the SSs circle. In the last step, an encounter situation is chosen by checking where the χ_r points to. This process is intuitively illustrated in Figure 3.3 where the blue line shows how the situation circle should be rotated and the red arrow decides which type to choose depending on both relative course and distance. Following this classification, Table 3.1 summarizes the desired turning direction in a typical two-vessel encounter, as required by COLREGs.

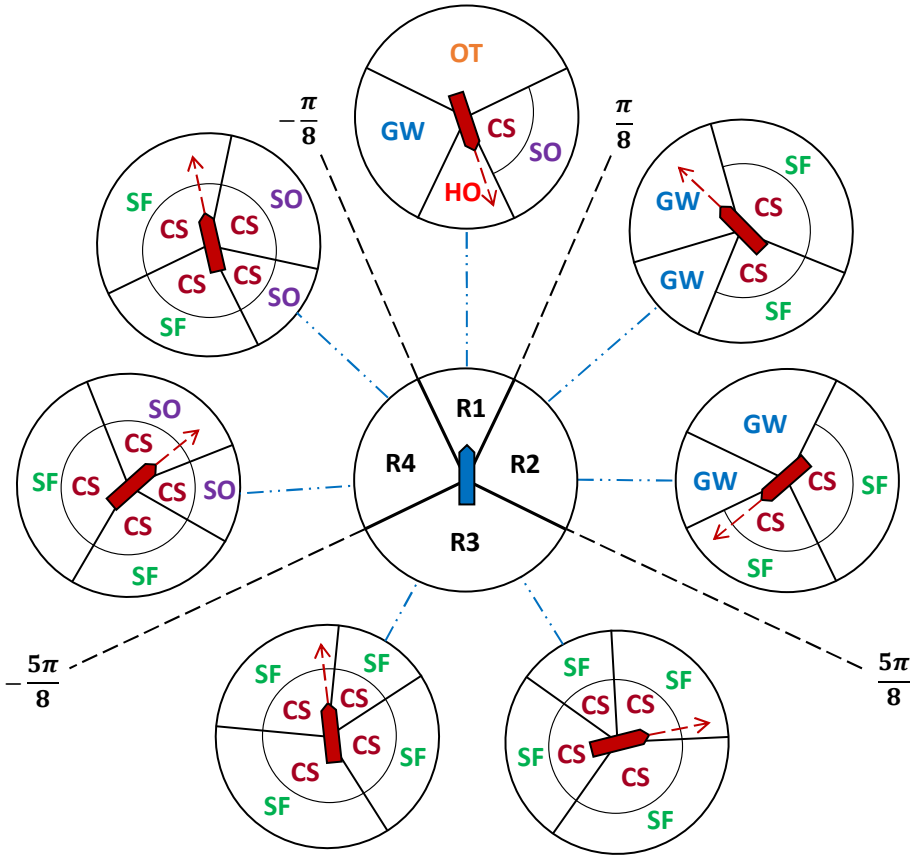


Figure 3.3: COLREGs Classification based on Relative Bearing and Relative Course with 4 Relative Bearing Sectors and Corresponding 4 types of Situation Sectors

Table 3.1: Encounter Types and Actions by the Own Ship according to COLREGs

Encounter Type	Action	COLREGs
Close	Starboard/Port Turn	Rule 2
Overtaking	Starboard/Port Turn	Rule 13
Head On	Starboard Turn	Rule 14
Give Way	Starboard Turn	Rule 15, 16
Stand On	Keep Course	Rule 15, 17
Safe	Keep Course	None

3.2.3 Action Selection Strategy

However, in a multi-vessel encounter, a ship might be in Give Way and Stand On situations at the same time, which demands conflicting actions. An action selection strategy, Algorithm 3, is proposed in this thesis to address this issue. The input set $\mathcal{E}$ captures and categorizes the target ships according to the aforementioned six types within a valid distance, and target ships beyond this range would not be considered. Firstly, the turning direction is initialized to be unrestricted, and the Stand On signal that allows the ship to continue the path following is false by default. Lines 3-8 search for encounter type SO, and reconfigure the Stand On signal. In the last step, lines 9-18 have two objectives. If there is HO or GW in the set, the turning direction is enforced to be starboard, and the Stand On is prohibited even if the reconfigured value is true. Additionally, if OT or CS exists in the set, standing on is also not allowed. This strategy ensures that the ship would stand on only if the followings are satisfied:

- SO exists in the set.
- No other types except SF exist.

and enters collision avoidance mode if the Stand On signal is false. The turning direction is chosen as starboard either HO or GW is in the set.

Algorithm 3 Multi-Vessel Encounter Action Selection

Input: Encounter types set $\mathcal{E}$

Direction ← Starboard/Port
Stand On ← false
for $i = 1, 2, \ldots, n$ **do**
    if $\mathcal{E}_i = \text{SO}$ **then**
        Stand On ← true
        break
    end if
end for
for $i = 1, 2, \ldots, n$ **do**
    if $\mathcal{E}_i = \text{HO}$ or $\mathcal{E}_i = \text{GW}$ **then**
        Direction ← Starboard
        Stand On ← false
        break
    end if
    if $\mathcal{E}_i = \text{OT}$ or $\mathcal{E}_i = \text{CS}$ **then**
        Stand On ← false
    end if
end for

3.3 Motion Prediction

Motion prediction is the process of forecasting the future trajectory of a dynamic obstacle, e.g., a robot or vessel, based on its current state. In this thesis, two simple motion prediction strategies are employed: straight-line motion prediction, and circular motion prediction. The latter differs from the former method by further taking the target ship's course change into consideration (Hornauer et al. 2015). To account for the prediction uncertainty resulting from signal noise and dynamic uncertainty within the prediction horizon (e.g., steering changes in course and speed), two velocity variation sets (translational and angular) are incorporated into the current velocity, and the prediction is performed utilizing different combinations of velocities. This robustness-enhancing technique is inspired by (Tor Arne Johansen et al. 2016). Moreover, to demonstrate the significance of motion prediction related to COLREGs compliance, a crossing situation is depicted in Figure 3.4. The future positions of the stand-on vessel in red are predicted for a time horizon T_{predict} under a certain step T_{step}. By setting up virtual obstacles in those positions, the give-way blue ship's initial route is blocked and is forced to make a starboard turn as stipulated.

3.3.1 Straight-line Motion

Straight-line motion prediction presumes that the target keeps its course and speed constant in the prediction horizon. Denoting ΔT as the prediction time, the predicted position and heading $\boldsymbol{\eta}_{\mathcal{P}}$ after Δt is found by:

$$\boldsymbol{\eta}_{\mathcal{P}} = \boldsymbol{\eta} + \boldsymbol{R}(\psi)\boldsymbol{\nu}\Delta T \tag{3.12}$$

where $\boldsymbol{R}(\psi) \in \mathbb{R}^{3\times3}$ is the rotation matrix expressed as:

$$\boldsymbol{R}(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.13}$$

3.3.2 Circular Motion

Due to the locality of actuators, a more accurate estimation of future motion is to include course change resulting from heading change $r\Delta T$. This assumption results in a circular trajectory, and the traveled path can be deemed as an arc segment. Thus, circular motion prediction is given by:

$$\boldsymbol{\eta}_{\mathcal{P}} = \boldsymbol{\eta} + \begin{bmatrix} \frac{U}{r}(\sin(\chi + r\Delta T) - \sin(\chi)) \\ \frac{U}{r}(-\cos(\chi + r\Delta T) + \cos(\chi)) \\ r\Delta T \end{bmatrix} \tag{3.14}$$

where U is the velocity amplitude, course angle χ is calculated by Equation 2.6, and $\frac{U}{r}$ is interpreted as the radius of the arc segment.

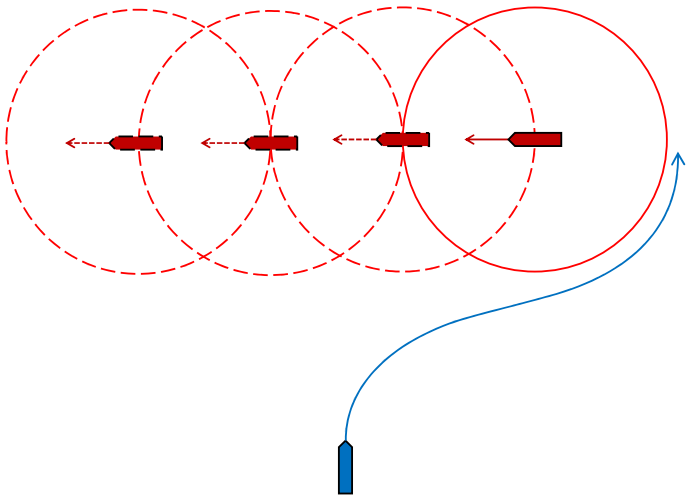


Figure 3.4: A Straight-line Motion Prediction Example in Crossing Scenario

3.3.3 Uncertainty Handling

Since sway velocity is usually much smaller than surge during maneuvering, we simplify the problem by only considering the variation added to surge speed u for translational velocity. Hence, we define $u \in \mathcal{U}$ and $r \in \mathcal{R}$, where $\mathcal{U}$ and $\mathcal{R}$ are discrete sets of surge speed and yaw rate, respectively. These sets include variations of their original values which consider the extent of signal uncertainty such as standard variation and relate to the target vessel's maximum acceleration as well as the execution time T_{exe}. For an execution time of 1 second, one possible choice for $\mathcal{U}$ and $\mathcal{R}$ is as follows:

$$\mathcal{U} = \{u-0.3,\ u-0.2,\ u-0.1,\ u,\ u+0.1,\ u+0.2,\ u+0.3\} \quad [\text{m/s}]$$

$$\mathcal{R} = \{r - \frac{\pi}{60},\ r - \frac{\pi}{90},\ r - \frac{\pi}{180},\ r,\ r + \frac{\pi}{180},\ r + \frac{\pi}{90},\ r + \frac{\pi}{60}\} \quad [\text{rad/s}]$$

3.4 Timed-Elastic-Band Approach

This section presents the TEB approach (Rösmann et al. 2017b), which aims to generate collision-free trajectories and provide planning commands for the motion control system. Firstly, the concept of the Timed Elastic Bands is defined, and two essential TEB nodes (Rösmann et al. 2013) are introduced. The subsequent discussion elaborates on the nonlinear programming problem, where equality and inequality constraints are explained. Due to the computationally expensive nature of nonlinear optimization, the program is converted into an approximative least-squares optimization problem which could efficiently reduce computation. After finding the optimal TEB, the corresponding planning command is generated through interpolation of the discrete trajectory. Finally, the overall algorithm workflow is explained and several modifications are discussed. For consistency, notations in this section follow the previous sections instead of the original literature.

3.4.1 TEB Definition

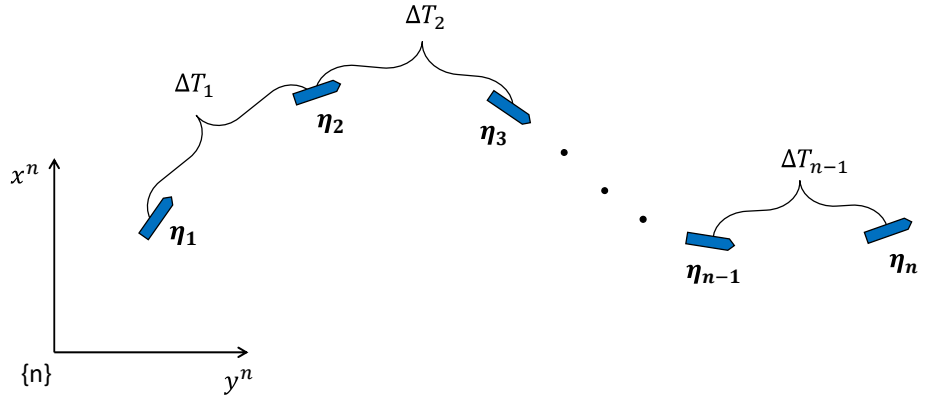


Figure 3.5: TEB representation

A discrete path is described as a sequence $\mathcal{H} = \{\eta_k \in \mathbb{R}^2 \times \mathbb{S} | k = 1, 2, \ldots, n\}$ representing a ship's planar positions and headings in NED frame. To enrich the representation with temporal information, the TEB incorporates strictly positive time intervals $\Delta T_k \in \mathbb{R}^+$ $(k = 1, 2, \ldots, n-1)$ that denotes the time required for the ship to maneuver from η_k to η_{k+1}. The time intervals are then integrated into the sequence $\mathcal{H}$, resulting in the TEB vector denoted as $\mathcal{B}$:

$$\mathcal{B} = [\underbrace{\eta_1^\top, \Delta T_1}_{\mathcal{B}_1}, \underbrace{\eta_2^\top, \Delta T_2}_{\mathcal{B}_2}, \ldots, \underbrace{\eta_{n-1}^\top, \Delta T_{n-1}}_{\mathcal{B}_{n-1}}, \eta_n^\top]^\top \tag{3.15}$$

It could be seen that the TEB is composed of several band points $\mathcal{B}_k$ $(k = 1, 2, \ldots, n-1)$ and the remaining destination point η_n, where a band point consists of two TEB nodes that have different properties, described in Table 3.2.

Table 3.2: TEB Nodes Definition

Variable	Symbol	Increment	Sum
Position & heading	$\eta_k = [x_k, y_k, \psi_k]^\top$	$[\Delta x_k, \Delta y_k, \Delta\psi_k]^\top$	$[x_k + \Delta x_k, y_k + \Delta y_k,$ $\text{ssa}(\psi_k + \Delta T_k)]^\top$
Time interval	ΔT_k	$\Delta\tilde{T}_k$	$\Delta T_k + \Delta\tilde{T}_k$

3.4.2 Optimization Problem

The objective of TEB optimization is to generate an optimal trajectory, respecting the ship's kinematic and dynamic constraints, as well as maintaining a safe distance from obstacles. To accomplish this goal, a nonlinear program is set up to find the optimal TEB:

$$
\begin{aligned}
\textbf{minimize} \quad & f(\mathcal{B}) = \sum_{k=1}^{n-1} \Delta T_k^2 && (3.16)\\
\textbf{subject to} \quad & \boldsymbol{\eta}_1 = \boldsymbol{\eta}_s, \quad \boldsymbol{\eta}_n = \boldsymbol{\eta}_g,\\
& \Delta T_k > 0,\\
& \boldsymbol{h}_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k) = \mathbf{0},\\
& \varrho_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k) \geq 0,\\
& o_k(\boldsymbol{\eta}_k) \geq 0,\\
& \boldsymbol{\nu}_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k, \Delta T_k) \geq 0, && (k = 1, 2, \ldots, n-1)\\
& \boldsymbol{a}_1(\boldsymbol{\eta}_2, \boldsymbol{\eta}_1, \Delta T_1) \geq 0,\\
& \boldsymbol{a}_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k, \boldsymbol{\eta}_{k-1}, \Delta T_k, \Delta T_{k-1}) \geq \mathbf{0}, && (k = 2, 3, \ldots, n-1)\\
& \boldsymbol{a}_n(\boldsymbol{\eta}_n, \boldsymbol{\eta}_{n-1}, \Delta T_{n-1}) \geq 0\\
& c_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k) \geq 0 && (k = 1, 2, \ldots, m)
\end{aligned}
$$

The definition of time interval ΔT_k indicates that the total maneuvering time can be estimated as the sum of time intervals, expressed as $T \approx \sum_{k=1}^{n-1} \Delta T_k$. However, in the objective function (Equation 3.16), minimizing T should not be adopted because this could potentially result in ill-posed TEB Position & heading nodes that maintain distance from obstacles by 'skipping' through them using a longer time interval between two nodes, rather than adjusting headings. This behavior is caused by the fact that time intervals are unconstrained and they have infinite combinations for certain total maneuvering time. To avoid this phenomenon, time intervals should be restricted to some small and equivalent values, and this could be realized by minimizing the sum of the squared ΔT_k, resulting in homogeneous time intervals $\Delta T_k = \frac{T}{n-1}$. It could be proved by Cauchy–Schwarz inequality, where its vector formulation states that:

$$\|\boldsymbol{x}\|\|\boldsymbol{y}\| \geq |\boldsymbol{x} \cdot \boldsymbol{y}| \tag{3.17}$$

Set vector $\boldsymbol{x} = (\Delta T_1, \Delta T_2, \ldots, \Delta T_{n-1})$ and vector $\boldsymbol{y} = (\underbrace{1, 1, \ldots, 1}_{n-1})$, we have:

$$\sqrt{n-1}\sqrt{\sum_{k=1}^{n-1} \Delta T_k^2} \geq \left|\sum_{k=1}^{n-1} \Delta T_k\right| \tag{3.18}$$

$$\Downarrow$$

$$\sum_{k=1}^{n-1} \Delta T_k^2 \geq \frac{1}{n-1}\left(\sum_{k=1}^{n-1} \Delta T_k\right)^2 \tag{3.19}$$

Assume the total maneuvering time is $T \approx \sum_{k=1}^{n-1} \Delta T_k$, it reduces to:

$$\sum_{k=1}^{n-1} \Delta T_k^2 \geq \frac{T^2}{n-1} \tag{3.20}$$

Hence, it could be concluded that the minimum value of the sum of squared ΔT_k is $\frac{T^2}{n-1}$. According to Cauchy–Schwarz inequality, equality occurs if and only if there exists a scalar λ

such that $\boldsymbol{x} = \lambda \boldsymbol{y}$, or if one of the vectors is zero. Only the first statement satisfy this condition, which gives the following:

$$\Delta T_k = \lambda \tag{3.21}$$

$$\Downarrow$$

$$\sum_{k=1}^{n-1} \lambda^2 = (n-1)\lambda^2 = \frac{T^2}{n-1} \tag{3.22}$$

$$\Downarrow$$

$$\Delta T_k = \frac{T}{n-1} \tag{3.23}$$

Thus, by minimizing the sum of squared ΔT_k, the optimized TEB has uniform time intervals.

Next, optimization constraints are explained as follows. $\boldsymbol{\eta}_1$ is fixed as the ship's current position & heading $\boldsymbol{\eta}_s$, and $\boldsymbol{\eta}_n$ is constrained to be the goal of the trajectory $\boldsymbol{\eta}_g$ calculated in Equation 2.75. $\boldsymbol{h}_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k)$ is the kinematic constraint between two consecutive positions and headings. $\varrho(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k)$ imposes geometric constraint of minimum turning radius of the trajectory. Inequality $o_k(\boldsymbol{\eta}_k)$ enforces minimum separation from obstacles. Limitations of dynamics are handled in velocity constraint $\boldsymbol{\nu}_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k, \Delta T_k)$, as well as acceleration constraint $\boldsymbol{a}_k(\cdot)$. Lastly, $c_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k)$ is designed for COLREGs compliance in encounter scenarios such as Head On and Give Way where only the Starboard Turn is highly preferred. The formulations of these constraints are discussed in the following.

Non-holonomic kinematics

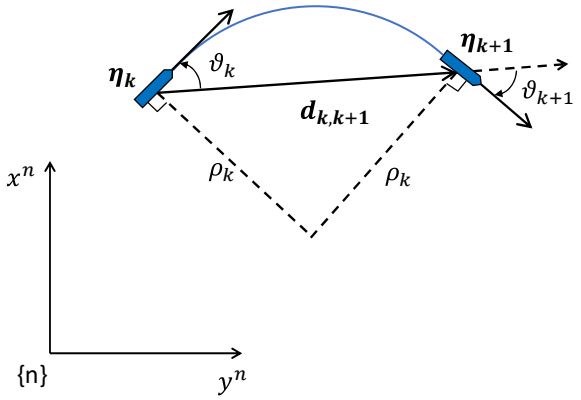


Figure 3.6: Non-holonomic Kinematics

The equality constraint $\boldsymbol{h}_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k)$ is originally designed for mobile robots with only two local DOF (they cannot move sideways), such as car-like robots or differential drives, which can only follow paths composed of linear and arc segments assuming constant speed. This constraint, however, can also be applied to the simulation model, as no thrusters are configured to provide control force in the sway direction (see Equation 2.31), resulting in negligible sway motion. The non-holonomic constraint is defined through a geometric interpretation, where $\boldsymbol{\eta}_k$ and $\boldsymbol{\eta}_{k+1}$ are required to be positioned on a common arc with constant curvature. The angle ϑ_k between $\boldsymbol{\eta}_k$ and the direction vector $\boldsymbol{d}_{k,k+1} = [x_{k+1} - x_k, y_{k+1} - y_k, 0]^\top$ must be equal to the corresponding angle ϑ_{k+1} at $\boldsymbol{\eta}_{k+1}$:

$$\vartheta_k = \vartheta_{k+1} \tag{3.24}$$

This condition indicates:

$$\begin{bmatrix} \cos(\psi_k) \\ \sin(\psi_k) \\ 0 \end{bmatrix} \times \boldsymbol{d}_{k,k+1} = \boldsymbol{d}_{k,k+1} \times \begin{bmatrix} \cos(\psi_{k+1}) \\ \sin(\psi_{k+1}) \\ 0 \end{bmatrix} \tag{3.25}$$

Rearranging the equation above yields the equality constraint:

$$\boldsymbol{h}_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k) = \left(\begin{bmatrix} \cos(\psi_k) \\ \sin(\psi_k) \\ 0 \end{bmatrix} + \begin{bmatrix} \cos(\psi_{k+1}) \\ \sin(\psi_{k+1}) \\ 0 \end{bmatrix} \right) \times \boldsymbol{d}_{k,k+1} \tag{3.26}$$

Minimum turning radius

To generate a smooth path, it must be restricted to a minimum turning radius. Moreover, a large turning radius forces the ship to take early action of course alteration during collision avoidance, as directed by COLREGs. The heading difference between $\boldsymbol{\eta}_k$ and $\boldsymbol{\eta}_{k+1}$ is denoted by $\Delta\psi_{k,k+1} = \text{ssa}(\psi_{k+1} - \psi_k)$, and the corresponding absolute turning radius is given by:

$$\rho_k = \frac{\|\boldsymbol{d}_{k,k+1}\|}{2\sin\left(\frac{|\Delta\psi_{k,k+1}|}{2}\right)} \tag{3.27}$$

Therefore, the resulting inequality constraint is:

$$\varrho_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k) = \rho_k - \rho_{\min} \tag{3.28}$$

Limited velocity

The ship's control force is subject to a limit, which in turn bounds the ship's maximum surge speed u and yaw rate r. To calculate these velocities between two consecutive positions and headings $\boldsymbol{\eta}_k$ and $\boldsymbol{\eta}_{k+1}$, the finite difference method is utilized, where the actual length of the arc segment is taken into account for surge speed computation. If the turning angle $\Delta\psi_{k,k+1}$ is zero, the path segment degenerates into a straight line with the turning radius of infinity. The surge speed is designed to be strictly positive, allowing the ship to move only forward. Thus, the expression for surge speed between $\boldsymbol{\eta}_k$ and $\boldsymbol{\eta}_{k+1}$ is:

$$u_k = \begin{cases} \dfrac{\|\boldsymbol{d}_{k,k+1}\|}{\Delta T_k} & \text{if } \Delta\psi_{k,k+1} = 0 \\ \dfrac{\rho_k |\Delta\psi_{k,k+1}|}{\Delta T_k} = \dfrac{\|\boldsymbol{d}_{k,k+1}\| |\Delta\psi_{k,k+1}|}{2\sin\left(\frac{|\Delta\psi_{k,k+1}|}{2}\right)\Delta T_k} & \text{if } \Delta\psi_{k,k+1} \neq 0 \end{cases} > 0 \tag{3.29}$$

Likewise, the yaw rate is computed as follows:

$$r_k = \frac{\Delta\psi_{k,k+1}}{\Delta T_k} \tag{3.30}$$

Then, the inequality constraint can be written in a vector form:

$$\boldsymbol{v}_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k, \Delta T_k) = \begin{bmatrix} u_{\max} - u_k \\ \omega_{\max} - |r_k| \end{bmatrix} \tag{3.31}$$

where $u_{\max}$ and $r_{\max}$ are maximum surge speed and yaw rate.

Limited acceleration

A similar procedure is applied for limiting surge and yaw accelerations $\dot{u}$ and $\dot{r}$, respectively. The starting point accelerations from the starting velocity $\boldsymbol{\nu}_s$ to $\boldsymbol{\nu}_1 = [u_1, 0, r_1]^\top$ are calculated using forward finite difference:

$$\dot{u}_1 = \frac{2(u_1 - u_s)}{\Delta T_1} \tag{3.32}$$

$$\dot{r}_1 = \frac{2(r_1 - r_s)}{\Delta T_1} \tag{3.33}$$

The initial acceleration inequality is then written as:

$$\boldsymbol{a}_1(\boldsymbol{\eta}_2, \boldsymbol{\eta}_1, \Delta T_1) = \begin{bmatrix} \dot{u}_{\max} - |\dot{u}_1| \\ \dot{r}_{\max} - |\dot{r}_1| \end{bmatrix} \tag{3.34}$$

For accelerations of intermediate points, a central finite difference method is used for better approximation accuracy. The computation procedure is given by:

$$\dot{u}_k = \frac{2(u_{k+1} - u_k)}{\Delta T_k + \Delta T_{k+1}} \tag{3.35}$$

$$\dot{r}_k = \frac{2(r_{k+1} - r_k)}{\Delta T_k + \Delta T_{k+1}} \tag{3.36}$$

$$\boldsymbol{a}_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k, \boldsymbol{\eta}_{k-1}, \Delta T_k, \Delta T_{k-1}) = \begin{bmatrix} \dot{u}_{\max} - |\dot{u}_k| \\ \dot{r}_{\max} - |\dot{r}_k| \end{bmatrix} \tag{3.37}$$

The backward finite difference is employed for the goal point, as follows:

$$\dot{u}_n = \frac{2(u_g - u_{n-1})}{\Delta T_{n-1}} \tag{3.38}$$

$$\dot{r}_n = \frac{2(r_g - r_{n-1})}{\Delta T_{n-1}} \tag{3.39}$$

$$\boldsymbol{a}_n(\boldsymbol{\eta}_n, \boldsymbol{\eta}_{n-1}, \Delta T_{n-1}) = \begin{bmatrix} a_{\max} - |\dot{u}_n| \\ \alpha_{\max} - |\dot{r}_n| \end{bmatrix} \tag{3.40}$$

Obstacle avoidance

The fundamental objective of the collision avoidance system is to ensure a safe distance between the own ship and any potential obstacles. To simplify the problem and improve computational efficiency, only point obstacles are considered in this thesis. The obstacle set is denoted as $\boldsymbol{\mathcal{O}}$ and consists of the 2D positions of reference points of the point obstacles, including the positions of static obstacles, the current positions of dynamic obstacles, and their predicted future positions for a time span of T_{predict} (as described in Section 3.3). The function $\delta(\boldsymbol{\eta}_k, \boldsymbol{\mathcal{O}})$ finds the minimum distance between the ship's position at step k and the obstacle set $\boldsymbol{\mathcal{O}}$. The inequality constraint $\boldsymbol{o}_k(\boldsymbol{\eta}_k)$ is formulated according to:

$$o_k(\boldsymbol{\eta}_k) = \delta(\boldsymbol{\eta}_k, \boldsymbol{\mathcal{O}}) - \delta_{\min}, \tag{3.41}$$

where $\delta_{\min}$ represents the minimum distance between the ship and the reference point of obstacles. The design of this value should take into account safe distance, own ship length, and the target ship length, as well as any other relevant factors.

COLREGs compliance

Inequality constraint $c_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k)$ prohibits the Port Turn ($r < 0$ or $\Delta\psi_{k,k+1} < 0$) for the first m headings when the signal $\kappa \in \{0, 1\}$ representing Starboard Turn is 1. If $\kappa = 0$, no specific turning direction is preferred, and this inequality constraint is always satisfied. This simple strategy is proven to be effective as will be shown in the next chapter.

$$c_k(\boldsymbol{\eta}_{k+1}, \boldsymbol{\eta}_k, \kappa) = \kappa \Delta\psi_{k,k+1} \tag{3.42}$$

3.4.3 Approximative Least-squares Optimization

Solving nonlinear programs with hard constraints is known to be a computationally expensive task. In order to improve computational efficiency, it is common to convert the constrained nonlinear program into an unconstrained optimization problem by transforming hard constraints into soft constraints incorporated in the objective function as penalty terms. One simple but effective approach is to formulate the penalty functions as quadratic functions. For equality constraints, it is formulated as:

$$\mathcal{E}(\boldsymbol{x}, \sigma_x) = \sigma_x \boldsymbol{x}^\top \boldsymbol{x} = \sigma_x \|\boldsymbol{x}\|^2 \tag{3.43}$$

where σ_x is a scalar weight, and $\boldsymbol{x}$ is the equality constraint. For inequality constraint, only the value below 0 is penalized. To achieve this, $\min(\cdot)$ function is utilized, which finds the row-wise smallest value. Thus, the inequality constraint penalty is:

$$\mathcal{I}(\boldsymbol{x}, \sigma_x) = \sigma_x \|\min(\boldsymbol{0}, \boldsymbol{x})\|^2 \tag{3.44}$$

Since the shortest time objective function is quadratic, Equation 3.43 also applies to it, augmenting the time objective with a tuning weight σ_t. Since $\boldsymbol{\eta}_1$ and $\boldsymbol{\eta}_n$ are fixed by current point $\boldsymbol{\eta}_s$ and goal point $\boldsymbol{\eta}_g$ respectively, they can be excluded from the optimization process. Therefore, the overall unconstrained optimization problem becomes:

$$\begin{aligned} \textbf{minimize} \quad \tilde{f}(\mathcal{B}) = \sum_{k=1}^{n-1} \big(& \mathcal{E}(\Delta T_k, \sigma_t) + \mathcal{E}(\boldsymbol{h}_k, \sigma_h) + \mathcal{I}(\rho_k, \sigma_\rho) + \mathcal{I}(o_k, \sigma_o) + \mathcal{I}(\boldsymbol{v}_k, \sigma_v) \\ & + \mathcal{I}(\boldsymbol{a}_k, \sigma_a) \big) + \mathcal{I}(a_n, \sigma_a) + \sum_{k=1}^{m} \big(\mathcal{I}(c_k, \sigma_c) \big) \end{aligned} \tag{3.45}$$

$$\mathcal{B}^* = \min_{\mathcal{B} \setminus \{\boldsymbol{\eta}_1, \boldsymbol{\eta}_n\}} \tilde{f}(\mathcal{B}) \tag{3.46}$$

where $\mathcal{B}^*$ is the optimized TEB, which will later be used to generate a full trajectory and corresponding command.

3.4.4 Optimization Solution

This section introduces the Levenberg-Marquardt algorithm (Marquardt 1963) used by the TEB approach to solve the least-squared optimization problem. By simplifying the notations in (Kümmerle et al. 2011), its general form is:

$$F(\boldsymbol{x}) = \sum_{k=1}^{N} \underbrace{\boldsymbol{e}_k(\boldsymbol{x})^\top \boldsymbol{\Omega}_k \boldsymbol{e}_k(\boldsymbol{x})}_{F_k(\boldsymbol{x})} \tag{3.47}$$

$$\boldsymbol{x}^* = \min_{\boldsymbol{x}} F(\boldsymbol{x}) \tag{3.48}$$

where $\boldsymbol{x}$ is the variable to be optimized, $\boldsymbol{e}_k(\boldsymbol{x})$ represents the error function, and $\boldsymbol{\Omega}$ is called the information matrix. In the context of TEB optimization problem, $\boldsymbol{x} = \mathcal{B} \setminus \{\boldsymbol{\eta}_1, \boldsymbol{\eta}_n\}$, and $\boldsymbol{e}_k(\boldsymbol{x})$

is the 2-norm part in the equality or inequality penalty functions in Equation 3.43 and 3.44. Moreover, the constraint penalty scaling factor σ_x is stored in $\boldsymbol{\Omega}_k$ diagonally. Let $\hat{\boldsymbol{x}}$ denote an initial guess of the solution, The error function can be approximated using first-order Taylor expansion around the current $\hat{\boldsymbol{x}}$, that is:

$$\boldsymbol{e}_k(\hat{\boldsymbol{x}} + \Delta\boldsymbol{x}) \approx \boldsymbol{e}_k + \boldsymbol{J}_k \Delta\boldsymbol{x} \tag{3.49}$$

where $\boldsymbol{J}_k$ is the Jacobian matrix of $\boldsymbol{e}_k(\hat{\boldsymbol{x}})$ derived at $\hat{\boldsymbol{x}}$, formulated as:

$$\boldsymbol{J}_k = \left.\frac{\partial \boldsymbol{e}_k(\hat{\boldsymbol{x}} + \Delta\boldsymbol{x})}{\partial \Delta\boldsymbol{x}}\right|_{\Delta\boldsymbol{x}=\boldsymbol{0}} \tag{3.50}$$

Substitute the error terms in Equation 3.47 using Equation 3.49, $F_k(\hat{\boldsymbol{x}} + \Delta\boldsymbol{x})$ can be approximated by:

$$\begin{aligned}
F_k(\hat{\boldsymbol{x}} + \Delta\boldsymbol{x}) &= \boldsymbol{e}_k(\hat{\boldsymbol{x}} + \Delta\boldsymbol{x})^\top \boldsymbol{\Omega}_k \boldsymbol{e}_k(\hat{\boldsymbol{x}} + \Delta\boldsymbol{x}) && (3.51)\\
&\approx (\boldsymbol{e}_k + \boldsymbol{J}_k\Delta\boldsymbol{x})^\top \boldsymbol{\Omega}_k (\boldsymbol{e}_k + \boldsymbol{J}_k\Delta\boldsymbol{x}) && (3.52)\\
&= \underbrace{\boldsymbol{e}_k^\top \boldsymbol{\Omega}_k \boldsymbol{e}_k}_{c_k} + 2\underbrace{\boldsymbol{e}_k^\top \boldsymbol{\Omega}_k \boldsymbol{J}_k}_{\boldsymbol{b}_k}\Delta\boldsymbol{x} + \Delta\boldsymbol{x}^\top \underbrace{\boldsymbol{J}_k^\top \boldsymbol{\Omega}_k \boldsymbol{J}_k}_{\boldsymbol{H}_k}\Delta\boldsymbol{x} && (3.53)\\
&= c_k + 2\boldsymbol{b}_k\Delta\boldsymbol{x} + \Delta\boldsymbol{x}^\top \boldsymbol{H}_k \Delta\boldsymbol{x} && (3.54)
\end{aligned}$$

Collecting all the terms of $F_k(\hat{\boldsymbol{x}} + \Delta\boldsymbol{x})$, the overall approximation of $F(\hat{\boldsymbol{x}} + \Delta\boldsymbol{x})$ becomes:

$$\begin{aligned}
F(\hat{\boldsymbol{x}} + \Delta\boldsymbol{x}) &= \sum_{k=1}^{N} F_k(\hat{\boldsymbol{x}} + \Delta\boldsymbol{x}) && (3.55)\\
&= \sum_{k=1}^{N} c_k + 2\boldsymbol{b}_k\Delta\boldsymbol{x} + \Delta\boldsymbol{x}^\top \boldsymbol{H}_k \Delta\boldsymbol{x} && (3.56)\\
&= c + 2\boldsymbol{b}^\top\Delta\boldsymbol{x} + \Delta\boldsymbol{x}^\top \boldsymbol{H}\Delta\boldsymbol{x} && (3.57)
\end{aligned}$$

where Equation 3.57 is obtained by letting: $c = \sum_{k=1}^{N} c_k$, $\boldsymbol{b} = \sum_{k=1}^{N} \boldsymbol{b}_k$ and $\boldsymbol{H} = \sum_{k=1}^{N} \boldsymbol{H}_k$. The minimum of $F(\hat{\boldsymbol{x}} + \Delta\boldsymbol{x})$ is found when $\Delta\boldsymbol{x}$ satisfy the following linear system:

$$\boldsymbol{H}\Delta\boldsymbol{x}^* = -\boldsymbol{b} \tag{3.58}$$

Then, the initial guess $\hat{\boldsymbol{x}}$ is updated by adding the step $\Delta\boldsymbol{x}^*$, which would be used in the next iteration.

$$\boldsymbol{x}^* = \hat{\boldsymbol{x}} + \Delta\boldsymbol{x}^* \tag{3.59}$$

While the Gauss-Newton algorithm solves Equation 3.58 and updates the guess $\hat{\boldsymbol{x}}$ for system linearization iteratively until a certain criterion is met for termination, the LM algorithm includes a damping factor $\lambda\boldsymbol{I}$ in Equation 3.58 (Kümmerle et al. 2011):

$$(\boldsymbol{H} + \lambda\boldsymbol{I})\Delta\boldsymbol{x}^* = -\boldsymbol{b} \tag{3.60}$$

The purpose of introducing the damping term is to control the step size of $\Delta\boldsymbol{x}$, and the value of λ is dynamically maintained by evaluating the error $\boldsymbol{e}_k$. If the error in the current step is larger than the previous value, the next step $\Delta\boldsymbol{x}$ should be decreased by increasing λ, which is advantageous for regions with high nonlinearity (Kümmerle et al. 2011). For the opposite situation, λ should be decreased for faster convergence.

3.4.5 Planning Command

The derivation of the planning command is based on a discrete trajectory, denoted as $\mathcal{T}$. It is comprised of a set of trajectory points $\mathcal{T}_i$ $(i = 1, 2, \ldots, n)$, where each point contains three components: accumulated time, position & heading, and velocity. It is represented as:

$$\mathcal{T} = \{\underbrace{t_1, \boldsymbol{\eta}_1, \boldsymbol{\nu}_1}_{\mathcal{T}_1}, \underbrace{t_2, \boldsymbol{\eta}_2, \boldsymbol{\nu}_2}_{\mathcal{T}_2}, \ldots, \underbrace{t_n, \boldsymbol{\eta}_n, \boldsymbol{\nu}_n}_{\mathcal{T}_n}\} \tag{3.61}$$

Algorithm 4 Obtaining Trajectory from TEB

Input: $\mathcal{B}, \boldsymbol{\nu}_s, \boldsymbol{\nu}_g$
$\mathcal{T}_1 \leftarrow 0, \boldsymbol{\eta}_s, \boldsymbol{\nu}_s$
for $i = 2, 3, \ldots, n-1$ **do**
  $t_i \leftarrow t_{i-1} + \Delta T_{i-1}$
  $\boldsymbol{\nu}_{i-1,i} \leftarrow \text{getVelocity}(\boldsymbol{\eta}_{i-1}, \boldsymbol{\eta}_i, \Delta T_{i-1})$
  $\boldsymbol{\nu}_{i,i+1} \leftarrow \text{getVelocity}(\boldsymbol{\eta}_i, \boldsymbol{\eta}_{i+1}, \Delta T_i)$
  $\mathcal{T}_i \leftarrow t_i, \boldsymbol{\eta}_i, \dfrac{\boldsymbol{\nu}_{i-1,i} + \boldsymbol{\nu}_{i,i+1}}{2}$
end for
$t_n \leftarrow t_{n-1} + \Delta T_{n-1}$
$\mathcal{T}_n \leftarrow t_n, \boldsymbol{\eta}_g, \boldsymbol{\nu}_g$
return $\mathcal{T} \leftarrow \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n\}$

Algorithm 4 shows the procedure to obtain the trajectory based on the optimized TEB. The algorithm starts with initializing the first trajectory point at the ship's starting condition, $\boldsymbol{\eta}_s$ and $\boldsymbol{\nu}_s$. The accumulated time is then updated, and we recover the velocities between the TEB band points $i-1$ and i, as well as i and $i+1$, using Equations 3.29 and 3.30 with zero sway speed. The velocity at the trajectory point i is then approximated as the average of these two velocities. Finally, the goal point information is collected and set for the last trajectory point.

Algorithm 5 Obtaining Planning Command from Trajectory

Input: $\mathcal{B}, \mathcal{T}, T_{\text{exe}}$
$i \leftarrow 1$
while true **do**
  if $T_{\text{exe}} > t_i$ **then**
    $i \leftarrow i + 1$
  else
    $i \leftarrow i - 1$
    break
  end if
end while
$\dot{r}_{\mathcal{C}} \leftarrow \dfrac{r_{i+1} - r_i}{\Delta T_i}$
$r_{\mathcal{C}} \leftarrow r_i + \dot{r}_{\mathcal{C}}(T_{\text{exe}} - t_i)$
$\psi_{\mathcal{C}} \leftarrow \text{ssa}\left(\psi_i + \dfrac{r_{\mathcal{C}}^2 - r_i^2}{2\dot{r}_{\mathcal{C}}}\right)$
$\dot{u}_{\mathcal{C}} \leftarrow \dfrac{u_{i+1} - u_i}{\Delta T_i}$
$u_{\mathcal{C}} \leftarrow u_i + \dot{u}_{\mathcal{C}}(T_{\text{exe}} - t_i)$
return $\mathcal{C} \leftarrow \{\psi_{\mathcal{C}}, r_{\mathcal{C}}, \dot{r}_{\mathcal{C}}, u_{\mathcal{C}}, \dot{u}_{\mathcal{C}}\}$

After obtaining the trajectory, its corresponding planning commands, defined as the interpolated dynamics of the trajectory in surge and yaw at the execution time T_{exe}, are found using Algorithm 5. Planning commands is denoted $\mathcal{C}$, which is expressed as:

$$\mathcal{C} = \{\psi_{\mathcal{C}}, r_{\mathcal{C}}, \dot{r}_{\mathcal{C}}, u_{\mathcal{C}}, \dot{u}_{\mathcal{C}}\} \tag{3.62}$$

In Algorithm 5, line 1-9 finds the index i that satisfies $t_i \leq T_{\text{exe}} \leq t_{i+1}$ through iteration. In line 10-12, yaw acceleration is found using finite difference, based on which the yaw rate and yaw angle is interpolated. The same procedure also applies to surge acceleration and surge speed.

3.4.6 Conventional Algorithm Workflow

The overall TEB approach is displayed in Algorithm 6. In the first step, $\mathcal{B}$ is initialized by a straight-line trajectory with uniformly distributed band points $\mathcal{B}_k$ both temporally and spatially, based on the starting point and goal point. In the second step, the TEB is iteratively optimized and its dimension is adjusted such that the TEB time intervals satisfies:

$$\Delta T_{\text{ref}} - \Delta T_{\text{hyst}} \leq \Delta T_k \leq \Delta T_{\text{ref}} + \Delta T_{\text{hyst}} \tag{3.63}$$

where ΔT_{hyst} is an anti-hysteresis parameter. During the adjustment, TEB band points are inserted through simple interpolation, or deleted without complicated adjustment of other band points since this task is handled by the optimizer automatically. This TEB dimension adjustment procedure is significant if the TEB dimension is insufficient which results in large time intervals, potentially leading to *obstacle-skipping* behavior of the optimized TEB as discussed earlier. Lastly, a trajectory is obtained, and corresponding planning commands that guide the vessel to collision avoidance are obtained, which are necessary inputs to the heading and surge speed controllers.

Algorithm 6 Conventional TEB approach

Input: $\eta_s, \nu_s, \eta_g, \nu_g, \mathcal{O}, T_{\text{exe}}$

1: $\mathcal{B}^* \leftarrow$ initializeTEB($\eta_s, \nu_s, \eta_g, \nu_g$)
2: **for** $i = 1, 2, \ldots, I_{\text{iteration}}$ **do**
3: $\quad \mathcal{B} \leftarrow$ adjustTEB($\mathcal{B}^*, \Delta T_{\text{ref}}, n_{\min}, n_{\max}$)
4: $\quad \mathcal{B}^* \leftarrow$ solveNLP($\mathcal{B}, \mathcal{O}$) ▷ see Equation 3.46 - 3.60
5: **end for**
6: $\mathcal{T} \leftarrow$ getTrajectory($\mathcal{B}^*, \nu_s, \nu_g$) ▷ see Algorithm 4
7: **return** $\mathcal{C} \leftarrow$ getCommand($\mathcal{B}^*, \mathcal{T}, T_{\text{exe}}$) ▷ see Algorithm 5

3.4.7 Algorithm Modifications

While the conventional TEB approach yields satisfying results for mobile robots, several issues are identified during experiments in the context of ships:

1. For ships with a 2-meter length, a minimum obstacle distance of 16 meters, and a nominal speed of 2.5 m/s, only a local goal with more than 90 meters in distance ($\Delta_g > 90$ m) is sufficient for complex encounters. This results in too many band points for a trajectory with a temporal resolution of 1 second.
2. One might consequently consider extending the reference interval to, e.g., 2 seconds, as to shorten computation time. However, longer intervals indicate less obstacle avoidance constraint coverage, which implies that more states between the nodes may not be collision-free due to poor resolution.

3. In some cases, the goal point overlaps with obstacles, either static obstacles or predicted dynamic obstacles. Thus, fixing the last node $\boldsymbol{\eta}_n$ as the goal $\boldsymbol{\eta}_g$ could potentially result in a trajectory leading to collisions.

To adapt the TEB framework for ships, and boost the algorithm run time without compromising performance, 4 modifications are made to the original approach:

1. Impose obstacle avoidance constraints for some interpolated states.
2. Unfix $\boldsymbol{\eta}_n$ as the goal $\boldsymbol{\eta}_g$. Instead, convert this equality constraint into a soft objective using the quadratic penalty function (see Equation 3.43).
3. Fix all time intervals as $\Delta T_k = \Delta T_{\text{ref}}$. Combined with the previous step, the overall decision variable dimension decreases by $n+2$.
4. To reach time optimality under modification 3, goal distance is selected as the farthest reachable distance in maximum speed for n band points, namely $\Delta_g = n\Delta T_{\text{ref}} u_{\max}$.

While modification 1 alleviates issue 2 described above, the second one solves issue 3 effectively by improving trajectory flexibility and collision avoidance capability. With these two modifications, experiments verify that a much shorter goal distance could be chosen without sacrificing performance, which basically eliminates issue 1. Modification 3 captures an underlying problem with the TEB approach - the unnecessary inclusion of time intervals as decision variables. By eliminating these variables, the computational effort is greatly reduced. Since fixing individual time intervals also means fixing the overall time, how to ensure the trajectory temporal efficiency becomes a problem. This motivates modification 4 which pushes the TEB nodes toward the farthest possible goal. Due to these modifications, Equation 3.45 and 3.46 are reformulated, and the ultimate optimization problem becomes:

$$\begin{aligned}\textbf{minimize} \qquad \tilde{f}(\boldsymbol{\mathcal{B}}) = \sum_{k=1}^{n-1}\Big(\mathcal{E}(\boldsymbol{h}_k,\sigma_h) + \mathcal{I}(\rho_k,\sigma_\rho) + \mathcal{I}(o_k,\sigma_o) + \mathcal{I}(\boldsymbol{v}_k,\sigma_v) + \mathcal{I}(\boldsymbol{a}_k,\sigma_a)\Big) \\ + \mathcal{I}(a_n,\sigma_a) + \sum_{k=1}^{m}\Big(\mathcal{I}(c_k,\sigma_c)\Big) + \mathcal{E}(\boldsymbol{g}_n,\sigma_g)\end{aligned} \tag{3.64}$$

$$\boldsymbol{\mathcal{B}}^* = \min_{\boldsymbol{\mathcal{B}}\setminus\{\boldsymbol{\eta}_1,\Delta T_1,\ldots,\Delta T_k\}} \tilde{f}(\boldsymbol{\mathcal{B}}) \tag{3.65}$$

where the goal equality constraint is:

$$\boldsymbol{g}_n(\boldsymbol{\eta}_n) = \|\boldsymbol{p}_n - \boldsymbol{p}_g\| + \mu|\text{ssa}(\psi_n - \psi_g)| \tag{3.66}$$

where $\boldsymbol{p}_n$ and $\boldsymbol{p}_g$ are position vectors, and μ is a scalar weight for heading. And the obstacle avoidance inequality constraint is updated as:

$$o_k(\boldsymbol{\eta}_k,\boldsymbol{\eta}_{k+1}) = \sum_{i=1}^{n_o}\Big(\delta(\boldsymbol{\eta}_k + \frac{i(\boldsymbol{\eta}_{k+1}-\boldsymbol{\eta}_k)}{n_o},\boldsymbol{\mathcal{O}}) - \delta_{\min}\Big) \tag{3.67}$$

where n_o denotes the number of interpolated states subject to additional obstacle avoidance penalties.

Chapter 4

Simulation and Analysis

In this chapter, the performance of the collision avoidance system is evaluated using two types of simulations. The first simulation is to test the ASV's capability to track global waypoints in narrow waterways in the presence of both static obstacles and target ships. The second simulation is conducted according to Imazu problem (Imazu 1987) with 22 encounter scenarios, involving 2-4 vessels. Without the assistance of the collision avoidance system, all ships would collide near the central point. All the participating vessels are equipped with the collision avoidance system developed previously and make individual navigational decisions.

4.1 Simulation Setup

As is shown in Figure 4.1, the simulation architecture has three main modules and three external data sources. These sources are essential inputs to the Motion Planning module. The global waypoints are generated either by mission planning or a human operator, and motion information about target ships is typically captured by some motion tracking systems using sensors such as radar, lidar, and AIS (Tor Arne Johansen et al. 2016). Static obstacles can be accurately obtained by digital charts. After receiving the relevant information, the Motion Planning module makes navigational decisions. The first step is to decide which mode to enter, either collision avoidance, or path following. Corresponding desired motion states are calculated, which is then processed by Motion Control. By retrieving the real-time position & heading, velocity, and propeller speed of the own ship using a navigation system and relevant sensors, this module computes the propeller speed command for the high-fidelity USV Otter Simulator.

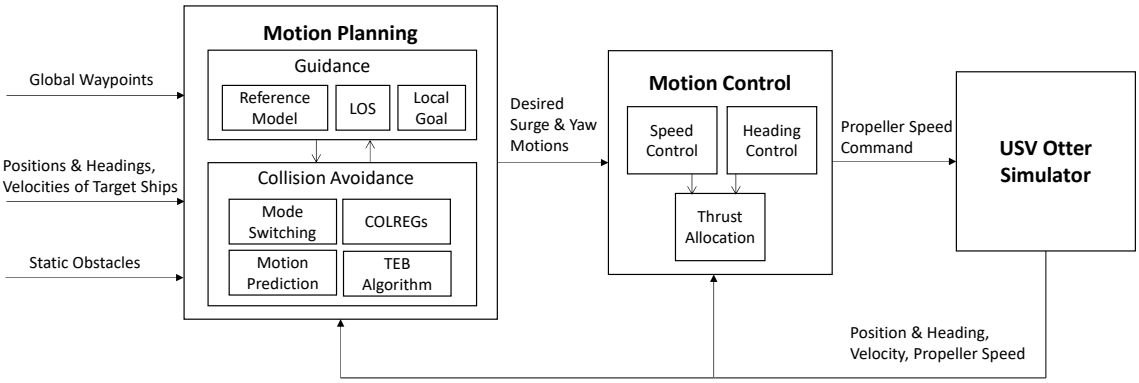


Figure 4.1: Simulation System Architecture

The simulation is implemented as a C++ project (PC: i5-6300 CPU @ 2.40GHz), where the USV Otter Simulator is adapted from the Python version of the MSS toolbox VESSELS catalog

(Fossen 2021b). The main component of the Collision Avoidance block - the TEB Algorithm, utilizes g2o graph optimization framework (Kümmerle et al. 2011) that implements a sparse variant of the LM algorithm efficiently for solving the nonlinear approximative least-squares optimization problem (Rösmann et al. 2017b). The configurations or system parameters with respect to USV Simulator, Motion Control, and Motion Planning are listed in Table A.1 - A.3. To evaluate if the collision avoidance system could work 'online', the computation time of planning commands is crucial. On average, solving the TEB optimization problem consumes around 0.06 s in each period with a peak of 0.085 s, which is relatively short for an execution time of 1 s. This indicates that the resulting delay during real-time operation is generally tolerable.

4.2 Simulation 1 Waypoints Tracking

In Simulation 1, the collision avoidance system is tested by tracking waypoints on a map with other ships. Following these global waypoints using pure path following is not guaranteed to be collision-free, while the collision avoidance system enables the ASV to maintain distance with static obstacles. Simulation results of 3 scenarios including the North-East plot or XY plot, yaw angle, and surge speed as well as relative distance among vessels are presented. These simulations demonstrate the strong capability of the system and show satisfactory results.

4.2.1 Scenario 1

Scenario 1 involves ASV 1 traveling through 3 waypoints within an acceptable range. Typically, if there are no obstacles in the proximity of the path, the ASV can safely execute path following using LOS guidance law and reference models. However, as depicted in Figure 4.2, the dashed straight lines representing the paths are close to, or overlapping with the grey and elliptical obstacles. These obstacles are generated virtually by placing point obstacles around the ellipse outline uniformly. To prevent collisions, a real-time and rolling horizon-based collision avoidance system which updates the planning command every second, is entered when the minimum distance with point obstacles falls below a threshold.

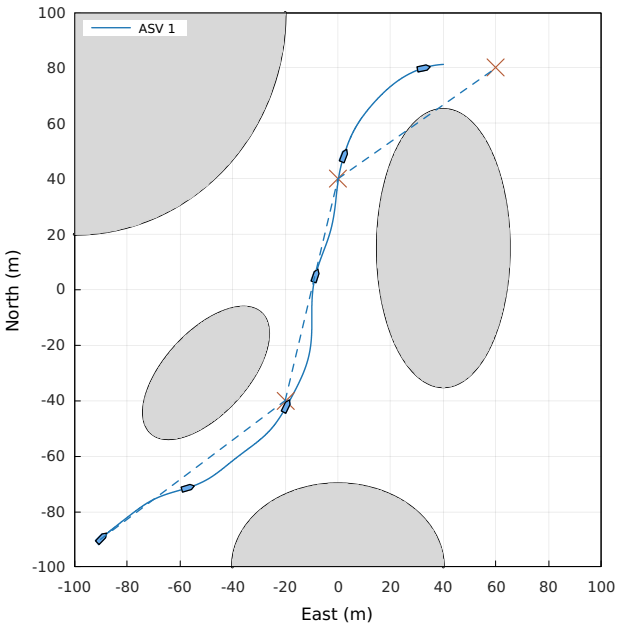
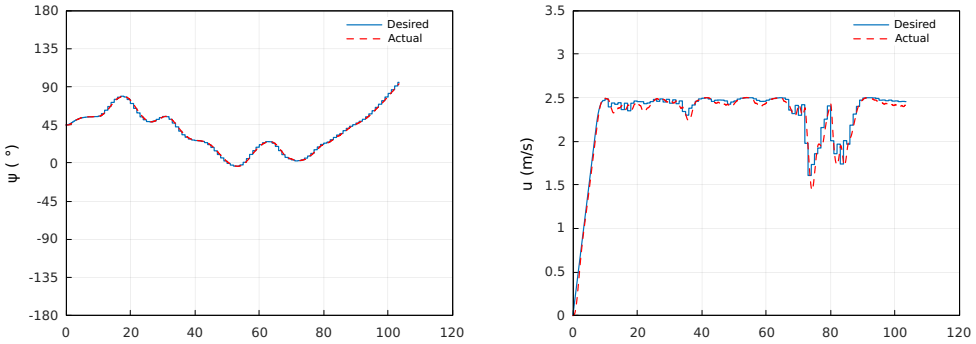


Figure 4.2: XY Plot in Simulation 1 Scenario 1

In Figure 4.3, the desired yaw angle is either provided by the filtered LOS during path following or directly generated by the TEB approach during collision avoidance, while the desired surge speed is obtained by filtering nominal speed of 2.5 m/s in path following mode. To show the temporal information of the resulting trajectory, Figure 4.2 draws the ASV outline with orientation every 20 seconds. It could be seen that the ASV does not strictly follow the given paths and maintains proper distance with the grey zone by changing course and surge speed simultaneously, in a time-optimal fashion. These planning commands are dynamically feasible, which renders a smooth trajectory. Meanwhile, due to an appropriate tuning and integrator anti-windup scheme, the heading and speed controllers demonstrate satisfactory tracking performance without noticeable phase lag or overshooting.



(a) Yaw **(b)** Surge

Figure 4.3: Yaw Angle and Surge Speed of ASV 1 in Simulation 1 Scenario 1

4.2.2 Scenario 2

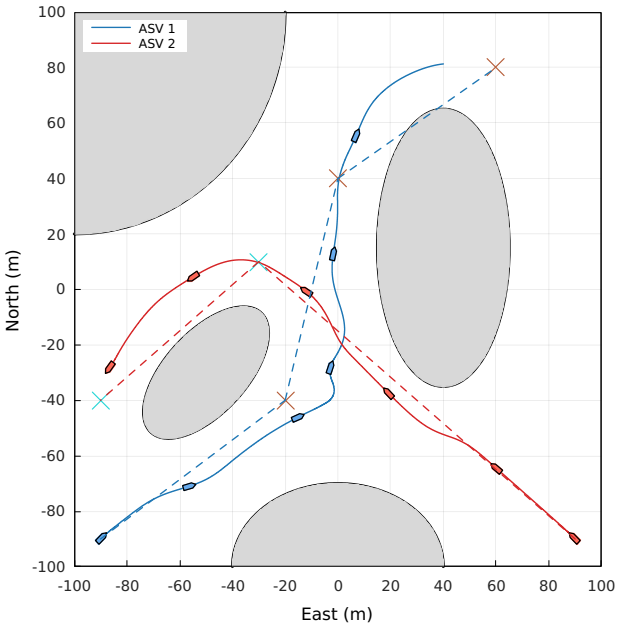


Figure 4.4: XY Plot in Simulation 1 Scenario 2

In scenario 2, ASV 2 is added to the map to evaluate the performance of handling both static and dynamic obstacles. The difference is that dynamic ones are related to COLREGs and their

positions are projected into the future time. In Figure 4.4, two ASVs have the potential to collide around the intersecting point of two dotted lines connecting the waypoints between time 40-60 s. Identifying this crossing scenario, ASV 1 decelerates aggressively in time 40-50 s (see Figure 4.6) and turns to starboard to give its way to ASV 2, the stand-on vessel. Compared with the previous XY plot, a deviation from ASV 1's original trajectory can be recognized. The stand-on role enables ASV 2 to execute path following between time 30-50 s since there is no risk of colliding with static obstacles. From Figure 4.5, it could be noticed that the distance between ASV 1 and 2 is well maintained, with a minimum value of more than 9 times of ship length. After leaving the high-risk collision zone, the two ASVs continue to maneuver under the guidance of collision avoidance systems and track their waypoints successfully, keeping their surge speeds mostly at their nominal values.

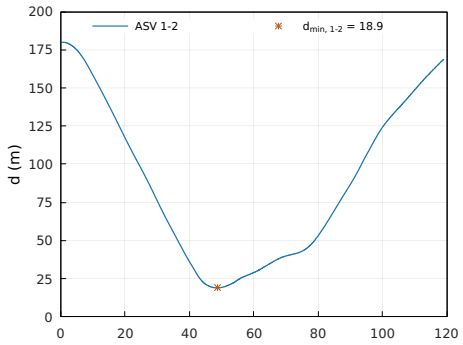
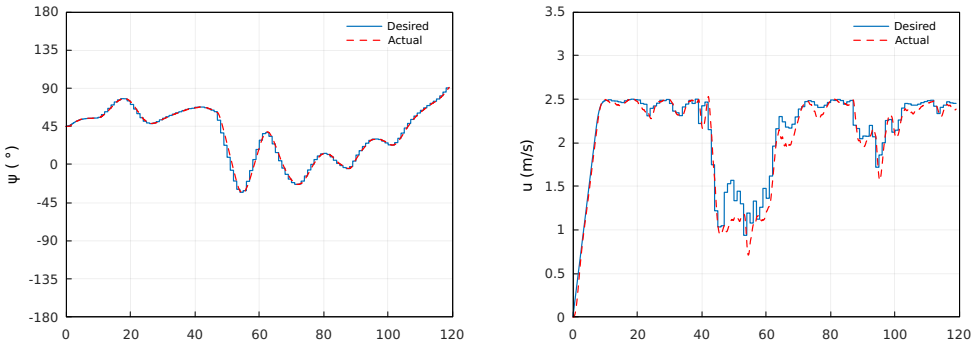
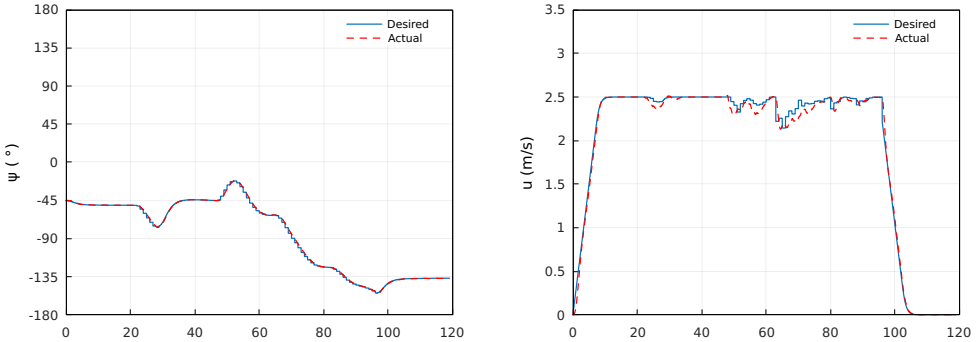


Figure 4.5: Relative Distance of ASVs in Simulation 1 Scenarios 2



(a) ASV 1 Yaw

(b) ASV1 Surge



(c) ASV2 Yaw

(d) ASV 2 Surge

Figure 4.6: Yaw Angle and Surge Speed of ASVs in Simulation 1 Scenario 2

4.2.3 Scenario 3

To further investigate the capability of the collision avoidance system in handling more complex situations, one additional ASV is incorporated into the simulation, which initially travels toward the south, making the waterway at their intersection point more congested. At time 30 s, the three ASVs are in a multi-vessel encounter within a close range in Figure 4.7, where the maneuverable space is considerably limited. Based on the COLREGs classification described previously, the encounter types set for ASV 1 is {GW, SO}. Processed by the action selection strategy, ASV 1 prefers a starboard turn. Similarly, ASV 2 has the set {SO, HO} and ASV 3 has the set {GW, HO}, which all render the same action as ASV 1.

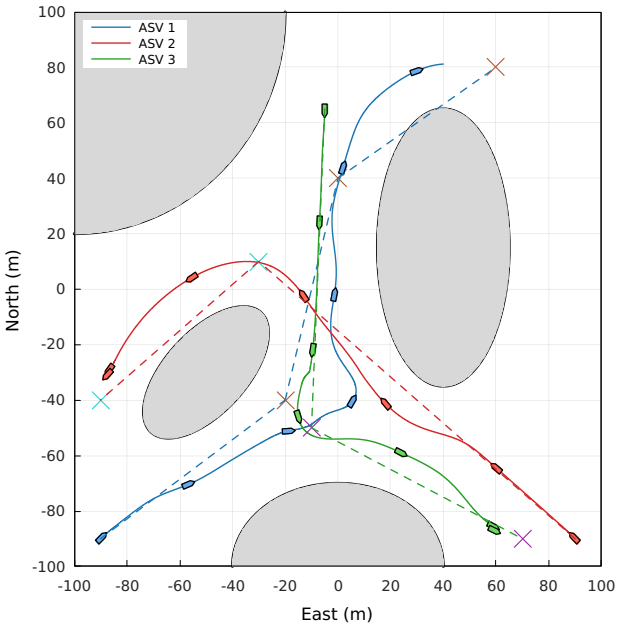


Figure 4.7: XY Plot in Simulation 1 Scenario 3

Additionally, Figure 4.9 indicates that both ASV 1 and 3 drop their surge speed to some degree, enabling ASV 2 to pass through the waterway at near full speed. In particular, the goal point of ASV 3 is almost occupied by ASV 1 at time 40 s. The flexible goal modification allows ASV 3 to contract its total path length to avoid collision and not reach its desired goal, without which ASV 3 would navigate itself to it and suffer the increased risk of collision with ASV 1. According to Figure 4.8, the minimum distance is found between ASV 1 and 3 at time around 50 s, which is equivalent to about 7 times of ship length.

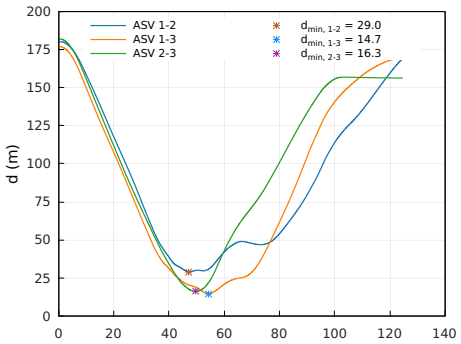
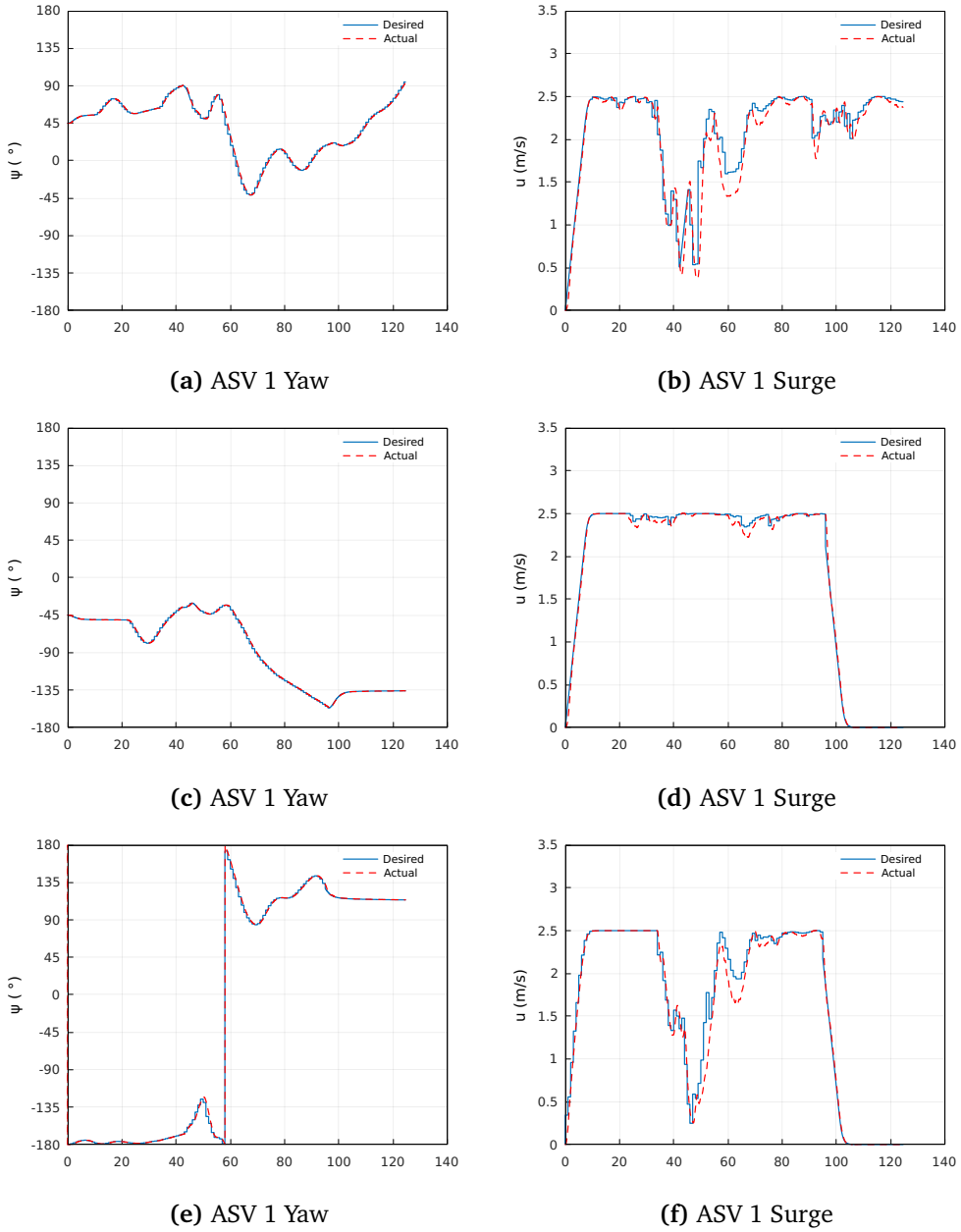


Figure 4.8: Relative Distance of ASVs in Simulation 1 Scenarios 3



(a) ASV 1 Yaw **(b)** ASV 1 Surge

(c) ASV 1 Yaw **(d)** ASV 1 Surge

(e) ASV 1 Yaw **(f)** ASV 1 Surge

Figure 4.9: Yaw Angle and Surge Speed of ASVs in Simulation 1 Scenario 3

4.3 Simulation 2 Imazu Problem

In Simulation 2, the system is tested by simulating a set of scenarios proposed in (Imazu 1987). This problem is a benchmark for testing maritime collision avoidance, which is comprised of 22 scenarios including 2 to 4 vessel encounters where all vessels are to collide in the same spot without intervention. Since COLREGs do not offer guidelines for multi-vessel encounters, the Imazu problem is challenging and has a high requirement for taking evasive actions (Cai and Hasegawa 2013). Inspired by the Imazu problem, two extra scenarios which expand the participating ship number to 5 are included in this simulation. In the Imazu problem experiment conducted in (Sawada et al. 2020), only the own ship is required to take action. In this simulation however, all ships are expected to take evasive maneuvers, except for some non-COLREGs-compliant or non-cooperative ships, drawn in red in Figure 4.10. These ships travel at a lower speed and stop after arriving at their goals.

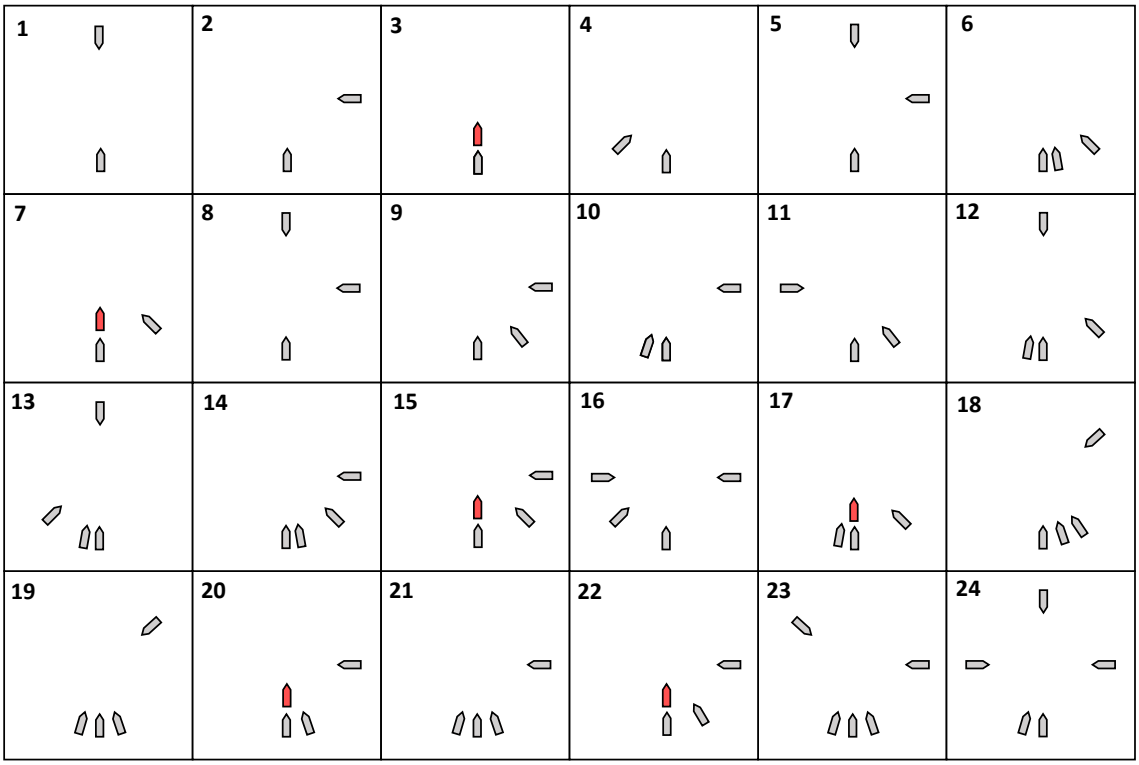


Figure 4.10: Imazu Collision Avoidance Problem with Two Extra Scenarios

To locate the participating ships for simulation, a polar coordinate system that marks the north direction with polar angle 0 is set up, where the expected collision point is located at its pole. Denoting radius as r and polar angle as φ, a point in this coordinate is expressed as $(r, \varphi) \in \mathbb{R}^+ \times (-180°, 180°]$. This indicates that the east and the west directions have polar angles $90°$ and $-90°$, respectively. To see the coordinates of all ASVs, please refer to Table B.1 for details. All the ships start at zero speed, and their nominal speed is uniformly 2.5 m/s, except for red ships that travel at 1 m/s and case 8 with 2 m/s as the sole distinction from case 5. The simulation results of the 24 scenarios are plotted in Figure 4.11, 4.12, 4.13 and 4.14 that also display the ASVs' shape every 20 seconds along the path, and their corresponding relative distance results are shown in Figure B.1, B.2, B.3 and B.4, where every 6 scenarios are grouped in each figure. Table B.2 further summarizes the minimum relative distances among all ASVs in each case. To interpret these distances with respect to the ship size, these values are divided by ship length L as an additional column in Table B.2. Next, the performance of the collision avoidance system will be discussed.

Scenario 1 to 4 are conventional 2-vessel encounter situations, namely Head On, Crossing, Overtaking, and Crossing, where the actions of each vessel are clearly regulated by COLREGs. In scenario 1, both ships turn to starboard before the collision point. In scenario 2, ASV 1 is the give-way vessel, and it slows down and turns to starboard. In scenario 3, ASV 1 takes over ASV 2 by steering to the port of ASV 1. Contrary to scenario 2, ASV 1 is the stand-on vessel which ASV 2 gives way to, by slowing down and turning to starboard.

Scenario 5 to 11 involves 3 participating ships, which scales the difficulty significantly due to less maneuverable space and uncertainty in motion prediction. Initially, each ship has two elements in the encounter types set that potentially require conflicting actions to take according to COLREGs. The proposed action selection strategy is thus heavily relied on in these scenarios. In scenario 5, all ships detect encounter types that force them to make a starboard turn and bypass the central collision point. This strategy is effective since it prevents turnings that direct two vessels toward the same spot. For instance, if ASV 1 is allowed to turn to port, it might lead to a collision with ASV 2 or ASV 3 if they both choose a starboard turn. In scenario 6,

ASV 3 has a uniform set of SO and thus, continues the path following, while ASV 1 and 2 are supposed to turn to starboard. However, ASV 1 turns to port due to potential collision with ASV 2 if a starboard turn is executed. After the crossing of ASV 3, ASV 2 becomes a stand-on vessel and reaches its goal earlier than ASV 1. ASV 2 in scenario 7 does not comply with COLREGs and does not give way to ASV 3 which is allowed to stand on. However, this information is unknown to ASV 3. After it gets too close to ASV 2, the SO type related to ASV 2 is converted to CS, enabling its emergency action - changing its course to cross ASV 2. ASV 1 has a give-way role and makes a small starboard turn. Scenario 8 is identical to scenario 5 except for nominal speeds. ASVs in both cases display similar behaviors by turning to starboard like entering and exiting a virtual roundabout. It could also be concluded from scenario 5 and 8 that not all ships containing SO in their encounter types set should stand on since this would suppress evasive actions of both ASV 2 and 3. Scenario 9 and 10 share many common points, where ASV 3 is the stand-on vessel that forces ASV 1 and 2 to decelerate until ASV 3 finishes crossing. After the crossing, ASV 1 and ASV 2 are also in a crossing situation whose crossing priority is dependent on their relative bearings. In scenario 9, ASV 2 is the second stand-on vessel while ASV 1 is the one in scenario 10. In scenario 11, ASV 3 stands on and forces ASV 2 to turn sharply to starboard. After the crossing of ASV 3, due to the different degrees of course deviation, ASV 2 passes through the conflict zone faster than ASV 1.

Scenario 12 to 22 consists of 11 cases with 4 vessels, among which some are considerably challenging and intractable since they mainly rely on reactive decision-making rather than clear actions described in COLREGs. In scenario 12 and 13, all ships enter collision avoidance mode with the preference for starboard turn since they have encounter types of either HO or GW, with no priority for crossing the conflict zone. Scenario 14 is highly similar to scenario 9. ASV 4 is allowed to stand on since 3 encounter types are uniformly SO. The remaining ships continue maneuvering based on their positions and headings formed after the give-way period. Scenario 15 has a non-cooperative vessel, ASV 2. Due to the lower speed of ASV 2, the stand-on vessel ASV 4 successfully crosses all ships at first. Later on, ASV 1 takes over ASV 2 from its starboard side, and ASV 3 crosses from its starboard as well. Similar to scenario 5, all ships in scenario 16 are supposed to turn to starboard where different degrees of turnings angles are purely reactive. In scenario 17, the stand-on vessel ASV 4's nominal path is blocked by the non-cooperative vessel, ASV 2, which results in a 90-degree turn to starboard after switching to collision avoidance mode. The other two ASVs take over ASV 2 while avoiding ASV 4. In scenario 18, 19 and 21, ASV 4 is the stand-on vessel, and the remaining 3 ASVs give their way by starboard turn. After that, these vessels swap their position in order to reach individual goals. In scenario 20, ASV 1 and 3 take over ASV 2, and ASV 4 crosses at full speed. Scenario 22 shows similar results, and the difference originates from starting position and heading of ASV 3.

Finally, scenario 23 and 24 include 5 participants, where the former is designed based on scenario 21 by adding ASV 5 on the northwest side and the latter is a combination of scenario 5 and 10. Since every ship has a target ship on its starboard side in scenario 23 and 24, all ships prefer starboard turns to avoid collisions. From these two complex scenarios, it could be seen that the action selection strategy is very effective in a multi-vessel encounter regarding prioritizing starboard turn if HO or GW appears in the encounter types set. All these scenarios demonstrate the strong capability of the proposed collision avoidance system where no collision occurs and distances among these ships are more than 4.5 times the ship length, which is sufficient for a 2-meter long ship.

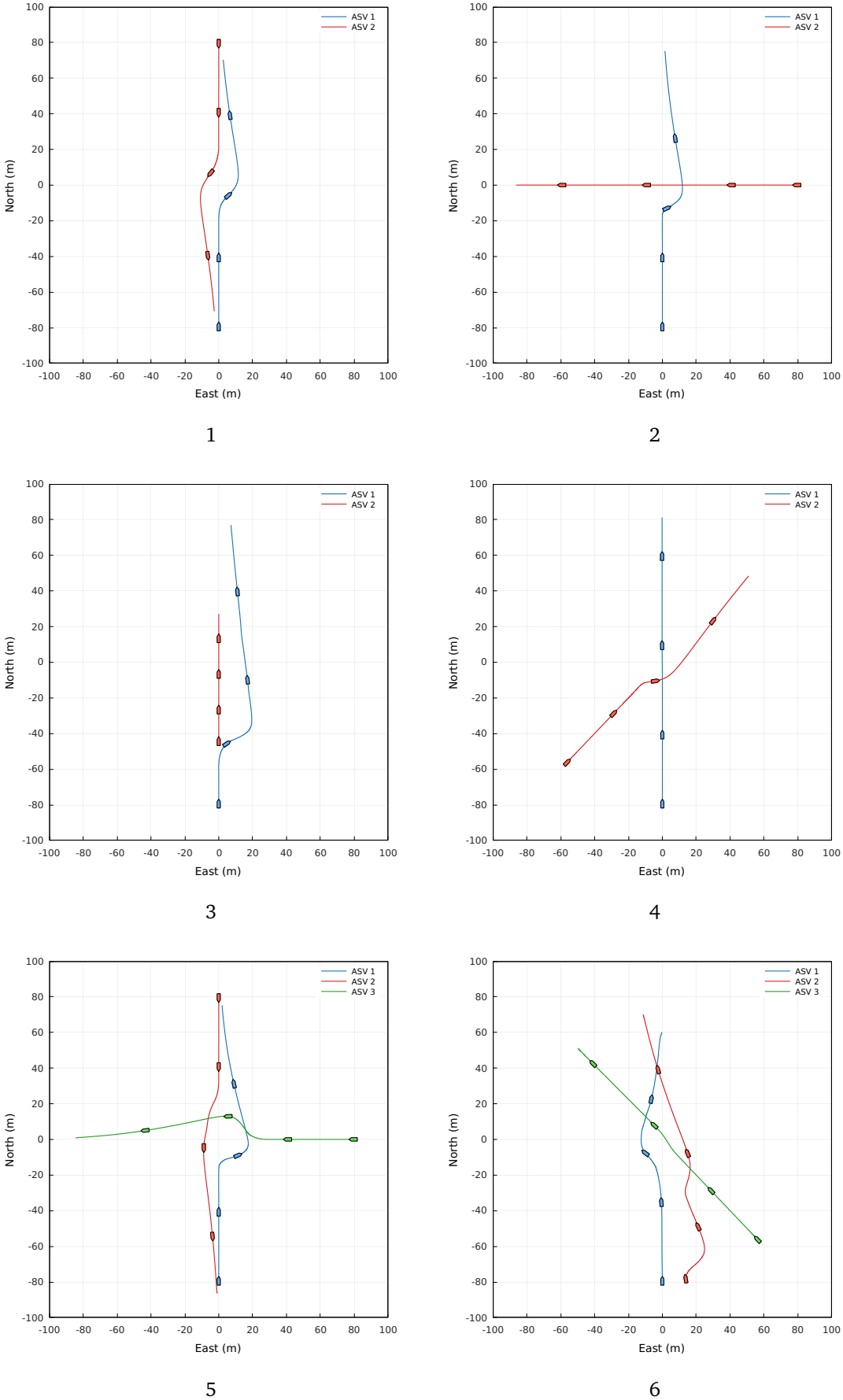
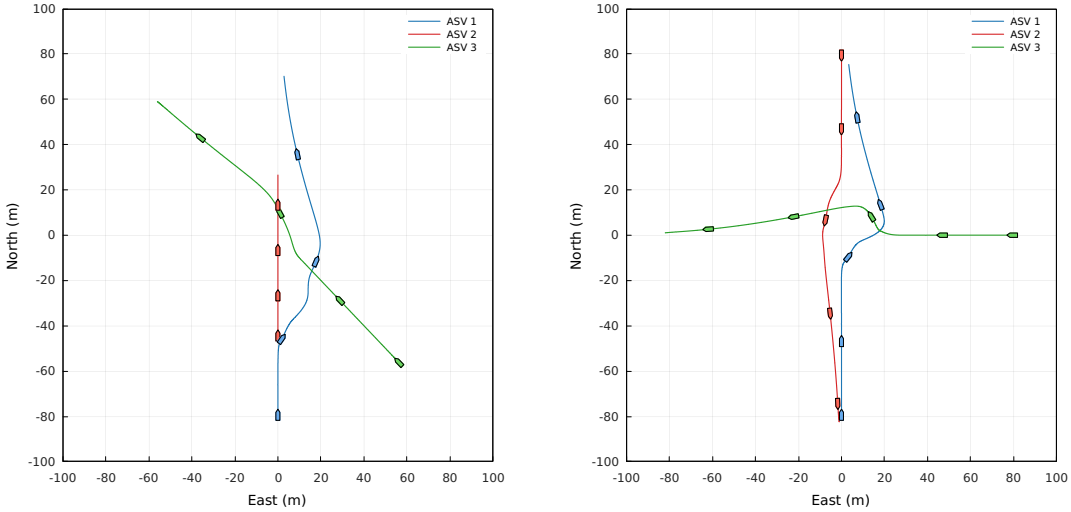
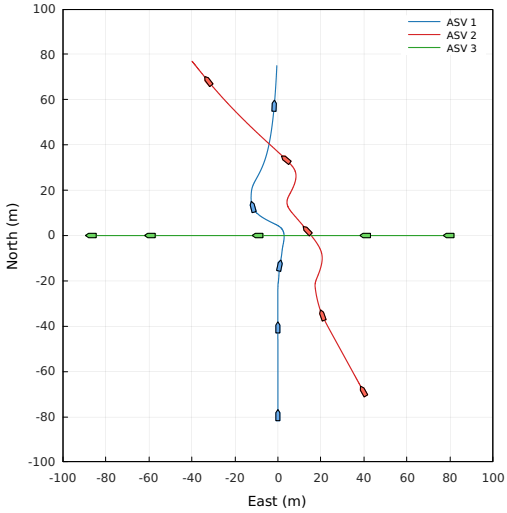


Figure 4.11: XY Plot in Simulation 2 Scenario 1-6

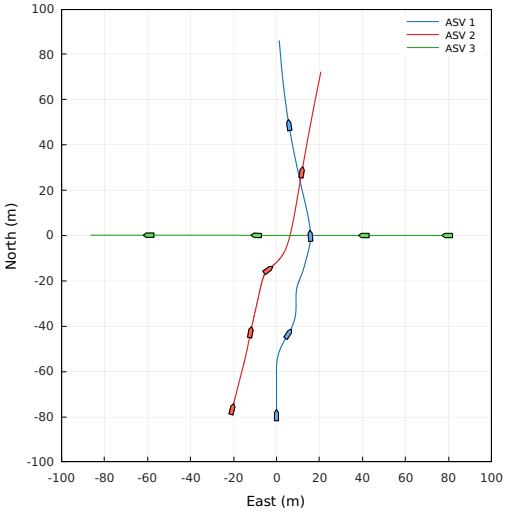


7

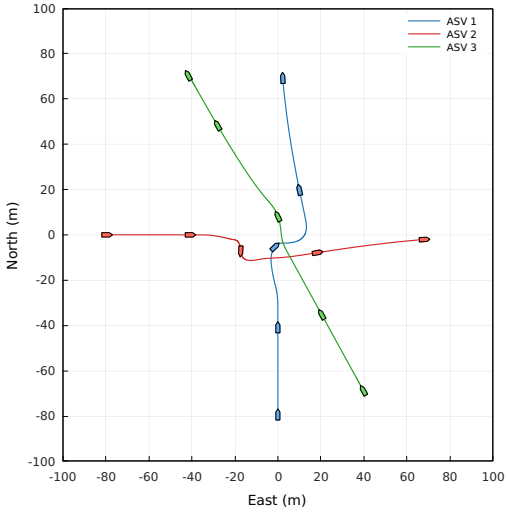
8



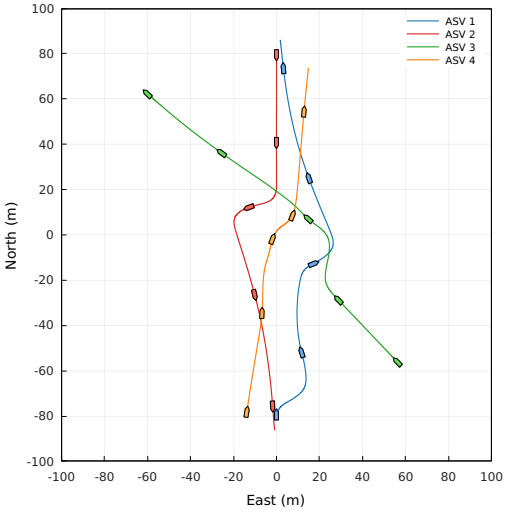
9



10

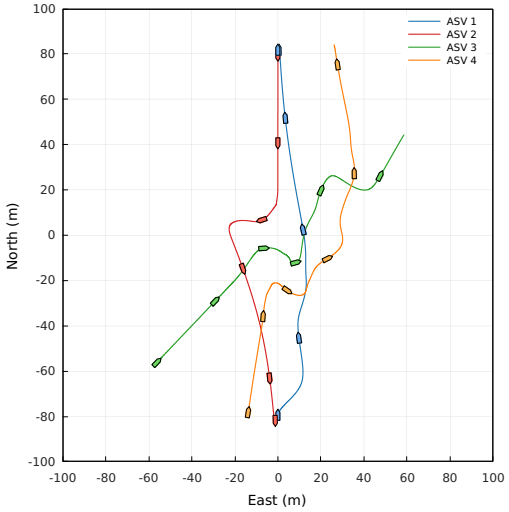


11

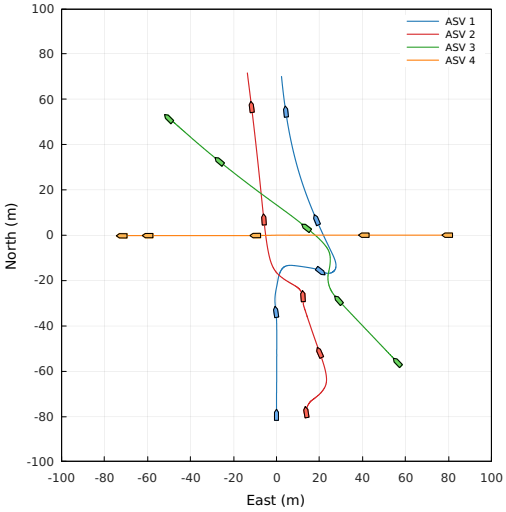


12

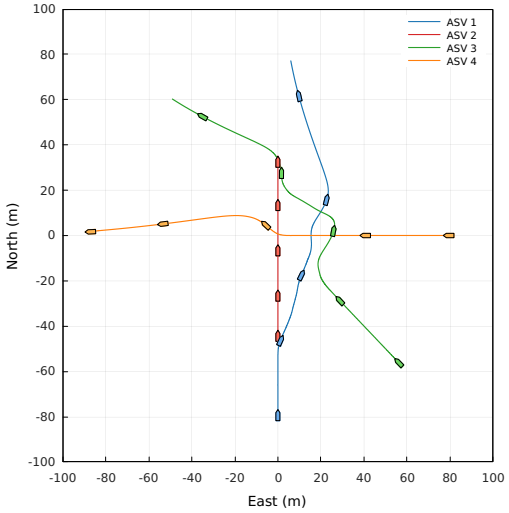
Figure 4.12: XY Plot in Simulation 2 Scenario 7-12



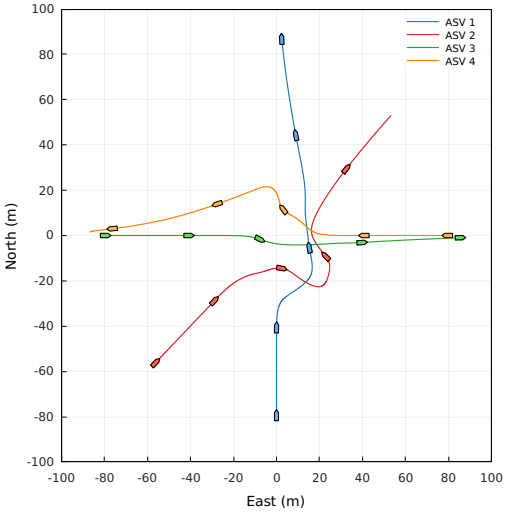
13



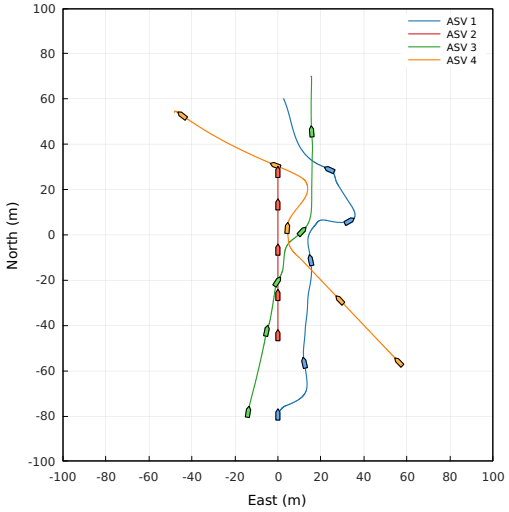
14



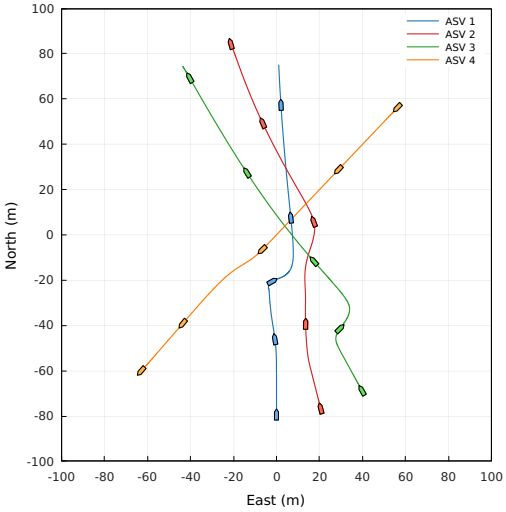
15



16

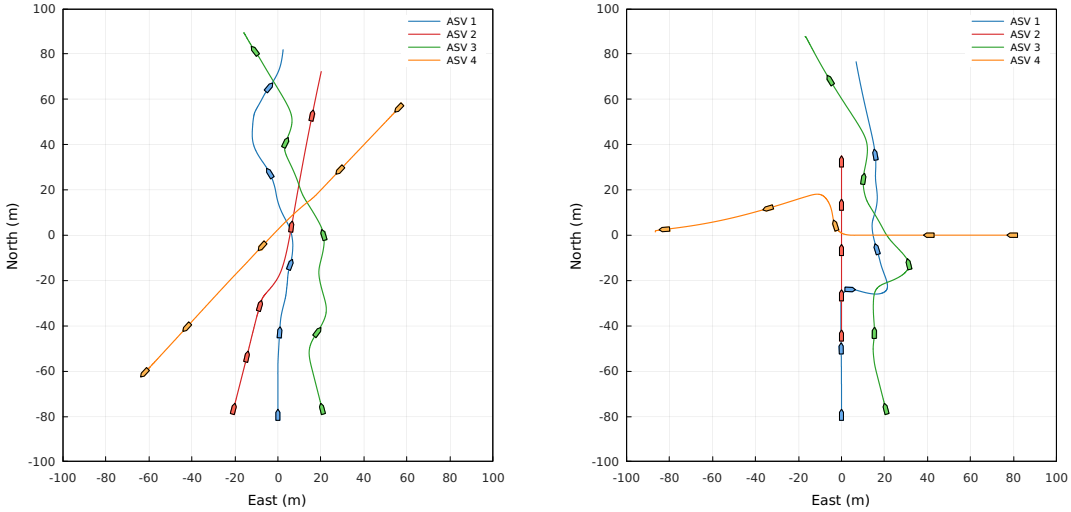


17

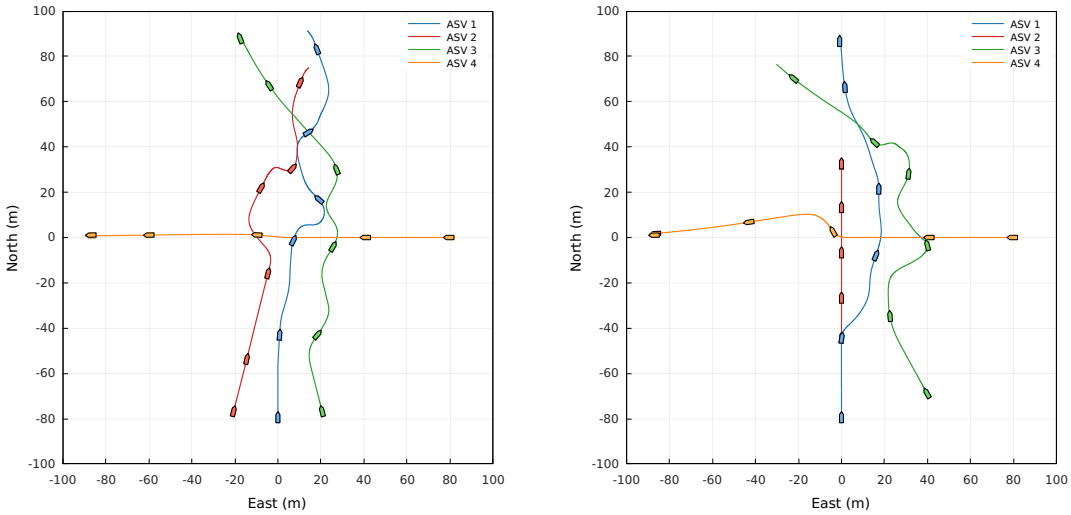


18

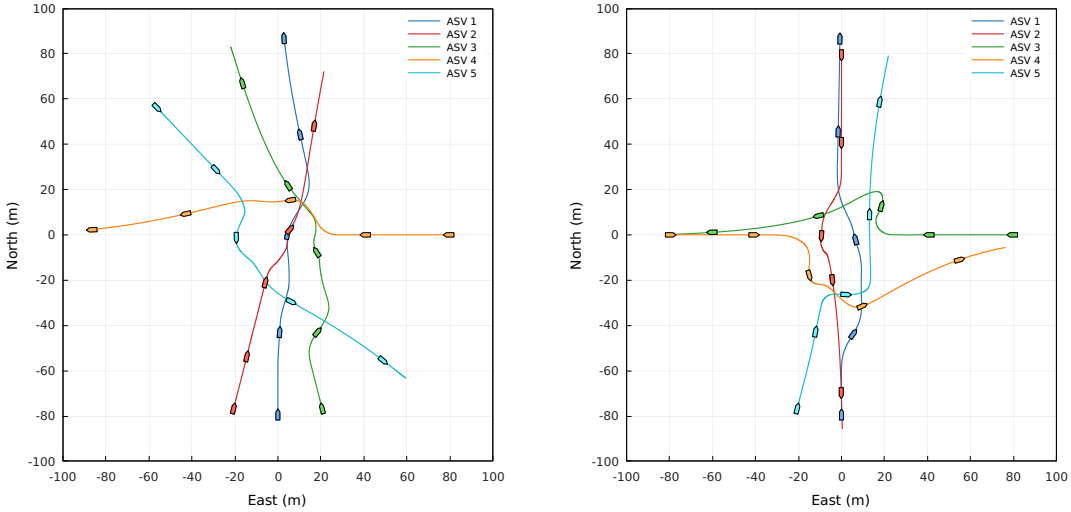
Figure 4.13: XY Plot in Simulation 2 Scenario 13-18



19 20



21 22



23 24

Figure 4.14: XY Plot in Simulation 2 Scenario 19-24

Chapter 5

Conclusion and Future Work

This thesis attempts to answer the two research questions described in Section 1.3, and the proposed solutions show promising results in Chapter 4. Firstly, the action selection strategy for multi-vessel encounters under the COLREGs framework effectively alleviates the uncertainty brought by conflicting roles. Notably, the TEB approach is successfully modified for collision avoidance of ASVs. The adapted method is optimal, robust, fast, flexible, and COLREGs compliant, which has enormous potential to be employed onboard vessels. Additionally, simulations of the challenging Imazu problem with extensions are collision-free and demonstrate the capability of the proposed collision avoidance system in handling complex scenarios.

For future work, a direct improvement is to enhance the motion prediction by considering ship dynamics and environmental loads. Furthermore, the target ship domain in this thesis is based on a circle, and more sophisticated shapes may be used for higher fidelity. Importantly, the developed collision avoidance system is far from being complete, e.g., an emergency mode could be designed, based on the evaluation of the cost in the objective function or simply the minimum distance with other vessels, which to some degree, closes the planning loop. A potential action in this mode is to decrease the weight of goal point equality constraint, or eliminate this constraint and make the planner goal-free, focusing on finding a collision-free trajectory. If conditions allow, field experiments could be conducted to verify the method at the next level. Finally, since the tuning of the collision avoidance system is only customized for small ASVs, it would be advantageous to design the system specifically for larger ships e.g., cargo ships, and explore its potential for providing optimal solutions for maritime collision avoidance.

Bibliography

Vagale, Anete, Rachid Oucheikh, Robin Bye, Ottar Osen, and Thor Fossen (Jan. 2021). "Path planning and collision avoidance for autonomous surface vehicles I: a review." In: *Journal of Marine Science and Technology* 26. DOI: 10.1007/s00773-020-00787-6.

Yara (2021). *Yara to start operating the world's first fully emission-free container ship: Yara International*. URL: https://www.yara.com/corporate-releases/yara-to-start-operating-the-worlds-first-fully-emission-free-container-ship/.

Mayflower (2022). *Mayflower Autonomous Ship*. URL: https://mas400.com/.

Burmeister, Hans-Christoph and Manfred Constapel (Sept. 2021). "Autonomous Collision Avoidance at Sea: A Survey." In: *Frontiers in Robotics and AI* 8. DOI: 10.3389/frobt.2021.739013.

Akdağ, Melih, Petter Solnør, and Tor Arne Johansen (2022a). "Collaborative collision avoidance for Maritime Autonomous Surface Ships: A review." In: *Ocean Engineering* 250, p. 110920. ISSN: 0029-8018. DOI: https://doi.org/10.1016/j.oceaneng.2022.110920. URL: https://www.sciencedirect.com/science/article/pii/S0029801822003444.

Johansen, Tor A., Andrea Cristofaro, and Tristan Perez (2016). "Ship Collision Avoidance Using Scenario-Based Model Predictive Control." In: *IFAC-PapersOnLine* 49.23. 10th IFAC Conference on Control Applications in Marine SystemsCAMS 2016, pp. 14–21. ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2016.10.315. URL: https://www.sciencedirect.com/science/article/pii/S2405896316319024.

Kufoalor, D. K. M., T. A. Johansen, E. F. Brekke, A. Hepsø, and K. Trnka (2020). "Autonomous maritime collision avoidance: Field verification of autonomous surface vehicle behavior in challenging scenarios." In: *Journal of Field Robotics* 37.3, pp. 387–403. DOI: https://doi.org/10.1002/rob.21919. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.21919. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21919.

Akdağ, Melih, Thor I. Fossen, and Tor A. Johansen (2022b). "Collaborative Collision Avoidance for Autonomous Ships Using Informed Scenario-Based Model Predictive Control." In: *IFAC-PapersOnLine* 55.31. 14th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles CAMS 2022, pp. 249–256. ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2022.10.439. URL: https://www.sciencedirect.com/science/article/pii/S2405896322024855.

Eriksen, Bjørn-Olav H. and Morten Breivik (2017). "MPC-Based mid-level collision avoidance for asvs using nonlinear programming." In: *2017 IEEE Conference on Control Technology and Applications (CCTA)*, pp. 766–772. DOI: 10.1109/CCTA.2017.8062554.

Thyri, Emil H. and Morten Breivik (2022). "Collision avoidance for ASVs through trajectory planning: MPC with COLREGs-compliant nonlinear constraints." In: *Modeling, Identification and Control* 43.2, pp. 55–77. DOI: 10.4173/mic.2022.2.2.

Eriksen, Bjørn-Olav H., Morten Breivik, Erik F. Wilthil, Andreas L. Flåten, and Edmund F. Brekke (2019). "The branching-course model predictive control algorithm for maritime collision avoidance." In: *Journal of Field Robotics* 36.7, pp. 1222–1249. DOI: https://doi.org/10.1002/rob.21900. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.21900. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21900.

Huang, Yamin, P.H.A.J.M. van Gelder, and Yuanqiao Wen (2018a). "Velocity obstacle algorithms for collision prevention at sea." In: *Ocean Engineering* 151, pp. 308–321. ISSN: 0029-8018. DOI: https://doi.org/10.1016/j.oceaneng.2018.01.001. URL: https://www.sciencedirect.com/science/article/pii/S0029801818300015.

Huang, Yamin, Linying Chen, and P.H.A.J.M. Gelder (Dec. 2018b). "Generalized Velocity Obstacle Algorithm for Preventing Ship Collisions at Sea." In: *Ocean Engineering*. DOI: 10.1016/j.oceaneng.2018.12.053.

Wang, Shaobo, Zhang Yingjun, and Li Lianbo (Nov. 2020). "A collision avoidance decision-making system for autonomous ship based on modified velocity obstacle method." In: *Ocean Engineering* 215, p. 107910. DOI: 10.1016/j.oceaneng.2020.107910.

Zhang, Ke, Liwen Huang, Yixiong He, Liang Zhang, Weiguo Huang, Cheng Xie, and Guozhu Hao (2022). "Collision Avoidance Method for Autonomous Ships Based on Modified Velocity Obstacle and Collision Risk Index." eng. In: *Journal of advanced transportation* 2022, pp. 1–22. ISSN: 0197-6729.

Song, A. Lifei, B. Yiran Su, C. Zaopeng Dong, D. Wei Shen, E. Zuquan Xiang, and F. Puxiu Mao (2018). "A two-level dynamic obstacle avoidance algorithm for unmanned surface vehicles." In: *Ocean Engineering* 170, pp. 351–360. ISSN: 0029-8018. DOI: https://doi.org/10.1016/j.oceaneng.2018.10.008. URL: https://www.sciencedirect.com/science/article/pii/S002980181831967X.

Lyu, Hongguang and Yong Yin (2019). "COLREGS-Constrained Real-time Path Planning for Autonomous Ships Using Modified Artificial Potential Fields." In: *The Journal of Navigation* 72.3, pp. 588–608. DOI: 10.1017/S0373463318000796.

Liu, Wei, Kun Qiu, Xiaofei Yang, Ronghao Wang, Zhengrong Xiang, Yue Wang, and Weixiang Xu (Apr. 2023). "COLREGS-Based Collision Avoidance Algorithm for Unmanned Surface Vehicles Using Modified Artificial Potential Fields." In: *Phys. Commun.* 57.C. ISSN: 1874-4907. DOI: 10.1016/j.phycom.2022.101980. URL: https://doi.org/10.1016/j.phycom.2022.101980.

Thyri, Emil H, Erlend A Basso, Morten Breivik, Kristin Y Pettersen, Roger Skjetne, and Anastasios M Lekkas (2020). "Reactive collision avoidance for ASVs based on control barrier functions." In: *2020 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, pp. 380–387.

Ellenrieder, Karl D. von (2022). "Control barrier function based collision avoidance control for underactuated USVs." In: *OCEANS 2022 - Chennai*, pp. 1–8. DOI: 10.1109/OCEANSChennai45887.2022.9775402.

Gong, Xiaoxuan, Lu Liu, and Zhouhua Peng (2022). "Safe-critical formation reconfiguration of multiple unmanned surface vehicles subject to static and dynamic obstacles based on guiding vector fields and fixed-time control barrier functions." In: *Ocean Engineering* 250, p. 110821.

Tsou, Ming-Cheng, Sheng-Long Kao, and Chien-Min Su (2010). "Decision support from genetic algorithms for ship collision avoidance route planning and alerts." In: *The Journal of Navigation* 63.1, pp. 167–182.

Ning, Jun, Hanmin Chen, Tieshan Li, Wei Li, and Chunze Li (2020). "COLREGs-Compliant unmanned surface vehicles collision avoidance based on multi-objective genetic algorithm." In: *Ieee Access* 8, pp. 190367–190377.

Wang, Hongjian, Zhongjian Fu, Jiajia Zhou, Mingyu Fu, and Li Ruan (2021). "Cooperative collision avoidance for unmanned surface vehicles based on improved genetic algorithm." In: *Ocean Engineering* 222, p. 108612.

Xia, Guoqing, Zhiwei Han, Bo Zhao, and Xinwei Wang (2020). "Local path planning for unmanned surface vehicle collision avoidance based on modified quantum particle swarm optimization." In: *Complexity* 2020, pp. 1–15.

Park, Jinwan (2021). "Improved collision avoidance method for autonomous surface vessels based on model predictive control using particle swarm optimization." In: *International Journal of Fuzzy Logic and Intelligent Systems* 21.4, pp. 378–390.

Zheng, Yisong, Xiuguo Zhang, Zijing Shang, Siyu Guo, and Yiquan Du (2021). "A decision-making method for ship collision avoidance based on improved cultural particle swarm." In: *Journal of Advanced transportation* 2021, pp. 1–31.

Perera, LP, JP Carvalho, and C Guedes Soares (2011). "Fuzzy logic based decision making system for collision avoidance of ocean navigation under critical collision conditions." In: *Journal of marine science and technology* 16, pp. 84–99.

Brcko, Tanja, Andrej Androjna, Jure Srše, and Renata Boć (2021). "Vessel multi-parametric collision avoidance decision model: Fuzzy approach." In: *Journal of Marine Science and Engineering* 9.1, p. 49.

Ahmed, Yaseen Adnan, Mohammed Abdul Hannan, Mahmoud Yasser Oraby, and Adi Maimun (2021). "COLREGs compliant fuzzy-based collision avoidance system for multiple ship encounters." In: *Journal of Marine Science and Engineering* 9.8, p. 790.

Zhao, Luman and Myung-Il Roh (2019). "COLREGs-compliant multiship collision avoidance based on deep reinforcement learning." In: *Ocean Engineering* 191, p. 106436.

Meyer, Eivind, Amalie Heiberg, Adil Rasheed, and Omer San (2020). "COLREG-compliant collision avoidance for unmanned surface vehicle using deep reinforcement learning." In: *Ieee Access* 8, pp. 165344–165364.

Chun, Do-Hyun, Myung-Il Roh, Hye-Won Lee, Jisang Ha, and Donghun Yu (2021). "Deep reinforcement learning-based collision avoidance for an autonomous ship." In: *Ocean Engineering* 234, p. 109216.

Quinlan, S. and O. Khatib (1993). "Elastic bands: connecting path planning and control." In: *[1993] Proceedings IEEE International Conference on Robotics and Automation*, 802–807 vol.2. DOI: 10.1109/ROBOT.1993.291936.

Roesmann, Christoph, Wendelin Feiten, Thomas Woesch, Frank Hoffmann, and Torsten Bertram (2012). "Trajectory modification considering dynamic constraints of autonomous robots." In: *ROBOTIK 2012; 7th German Conference on Robotics*, pp. 1–6.

Rösmann, Christoph, Wendelin Feiten, Thomas Wösch, Frank Hoffmann, and Torsten Bertram (2013). "Efficient trajectory optimization using a sparse model." In: *2013 European Conference on Mobile Robots*, pp. 138–143. DOI: 10.1109/ECMR.2013.6698833.

Kümmerle, Rainer, Giorgio Grisetti, Hauke Strasdat, Kurt Konolige, and Wolfram Burgard (2011). "G2o: A general framework for graph optimization." In: *2011 IEEE International Conference on Robotics and Automation*, pp. 3607–3613. DOI: 10.1109/ICRA.2011.5979949.

Rösmann, Christoph, Frank Hoffmann, and Torsten Bertram (2017a). "Kinodynamic trajectory optimization and control for car-like robots." In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5681–5686. DOI: 10.1109/IROS.2017.8206458.

Magyar, Bence, Nikolaos Tsiogkas, Jérémie Deray, Sammy Pfeiffer, and David Lane (2019). "Timed-Elastic Bands for Manipulation Motion Planning." In: *IEEE Robotics and Automation Letters* 4.4, pp. 3513–3520. DOI: 10.1109/LRA.2019.2927956.

Fossen, Thor I (2021a). *Handbook of marine craft hydrodynamics and motion control*. eng. Hoboken, NJ.

Bergman, Kristoffer, Oskar Ljungqvist, Jonas Linder, and Daniel Axehill (Dec. 2020). "A COLREGs Compliant Motion Planner for Autonomous Maneuvering of Marine Vessels in Complex Environments." In.

IMO (1972). *Convention on the International Regulations for Preventing Collisions at Sea, 1972 (COLREGs)*. URL: https://www.imo.org/en/About/Conventions/Pages/COLREG.aspx.

Thyri, Emil and Morten Breivik (May 2022). "A domain-based and reactive COLAV method with a partially COLREGs-compliant domain for ASVs operating in confined waters." In: *Field Robotics* 2, pp. 637–677. DOI: 10.55417/fr.2022022.

Hornauer, Sascha, Michael Blaich, Axel Hahn, and Johannes Reuter (Sept. 2015). "Trajectory Planning with Negotiation for Maritime Collision Avoidance." In: *TransNav: International Journal on Marine Navigation and Safety of Sea Transportation* 9, p. 335. DOI: 10.12716/1001.09.03.05.

Johansen, Tor Arne, Tristan Perez, and Andrea Cristofaro (2016). "Ship Collision Avoidance and COLREGS Compliance Using Simulation-Based Control Behavior Selection With Predictive Hazard Assessment." In: *IEEE Transactions on Intelligent Transportation Systems* 17.12, pp. 3407–3422. DOI: 10.1109/TITS.2016.2551780.

Rösmann, Christoph, Frank Hoffmann, and Torsten Bertram (2017b). "Integrated online trajectory planning and optimization in distinctive topologies." In: *Robotics and Autonomous Systems* 88, pp. 142–153. ISSN: 0921-8890. DOI: https://doi.org/10.1016/j.robot.2016.11.007. URL: https://www.sciencedirect.com/science/article/pii/S0921889016300495.

Marquardt, Donald W. (1963). "An Algorithm for Least-Squares Estimation of Nonlinear Parameters." In: *Journal of the Society for Industrial and Applied Mathematics* 11.2, pp. 431–441. DOI: 10.1137/0111030. eprint: https://doi.org/10.1137/0111030. URL: https://doi.org/10.1137/0111030.

Imazu (1987). "Research on collision avoidance manoeuvre." PhD thesis. Tokyo, Japan: The University of Tokyo.

Fossen, Thor I (2021b). *PythonVehicleSimulator*. URL: https://github.com/cybergalactic/PythonVehicleSimulator.

Cai, Yuan Long and Kazuhiko Hasegawa (2013). "Evaluating of Marine Traffic Simulation System through Imazu Problem." In.

Sawada, Ryohei, Keiji Sato, and Takahiro Majima (Aug. 2020). "Automatic ship collision avoidance using deep reinforcement learning with LSTM in continuous action spaces." In: *Journal of Marine Science and Technology* 26. DOI: 10.1007/s00773-020-00755-0.

Appendix A

Simulation Configuration

Table A.1: USV Otter Simulator Parameters

Parameter	Symbol	Value
Length	L	2 m
Beam	B	1.08 m
Total Mass	m_{total}	90 kg
Maximum Speed	u_{max}	3 m/s
Number of Propellers	$n_{\text{propeller}}$	2

Table A.2: Motion Control Configuration

Parameter	Symbol	Value
Heading PID Natural Frequency	ω_n	4.5
Heading PID Damping Ratio	ζ	1
Heading PID Mass Term	m	41.4
Heading PID Damping Term	d	41.4
Speed SMC Gain	K_σ	15
Speed SMC Slope Factor	ϕ	6

Table A.3: Motion Planning Configuration

(a) Guidance

Parameter	Symbol	Value
Speed Reference Model Natural Frequency	ω_n	1.5
Speed Reference Model Damping Ratio	ζ	1
Speed Reference Model Maximum Surge Speed	$u_{\max}$	3 m/s
Speed Reference Model Maximum Surge Acceleration	$a_{\max}$	0.3 m/s^2
Heading Reference Model Natural Frequency	ω_n	1
Heading Reference Model Damping Ratio	ζ	1
Heading Reference Model Maximum Yaw Rate	$r_{\max}$	$\frac{\pi}{9}$ rad/s
Heading Reference Model Maximum Yaw Acceleration	$\alpha_{\max}$	$\frac{\pi}{45}$ rad/s^2
LOS Lookahead Distance	Δ	33.33 m
LOS Integral Gain	K_i	0.001
Lookahead Distance of Goal Point	Δ_g	30 m
Waypoint Switching Radius	R_{switch}	20 m

(b) Collision Avoidance

Parameter	Symbol	Value
DCPA Mode-Switching Threshold	d_{switch}	20 m
DCPA Hysteresis Parameter	d_{hyst}	1 m
TCPA Mode-Switching Lower Bound	t_{lower}	0 s
TCPA Mode-Switching Upper Bound	t_{high}	20 s
TCPA Hysteresis Parameter	t_{hyst}	1 s
Threshold Distance for Conversion to CS	d_{close}	14 m
TEB Reference Time Interval	ΔT_{ref}	2 s
TEB Minimum Samples	n_{min}	3
TEB Maximum Samples	n_{max}	500
TEB Maximum Surge Speed	u_{max}	2.5 m/s
TEB Maximum Surge Acceleration	$\dot{u}_{\text{max}}$	0.25 m/s^2
TEB Maximum Yaw Rate	r_{max}	$\frac{\pi}{18}$ rad/s
TEB Maximum Yaw Acceleration	$\dot{r}_{\text{max}}$	$\frac{\pi}{275}$ rad/s^2
TEB Minimum Turning Radius	ρ_{min}	5 m
TEB Minimum Obstacle Distance	δ_{min}	16 m
TEB COLREGs Weight	σ_c	1e6
TEB Velocity Weight	σ_v	1e4
TEB Acceleration Weight	σ_a	1e4
TEB Non-Holonomic Kinematics Weight	σ_h	1e4
TEB Obstacle Weight	σ_o	100
TEB Turning Radius Weight	σ_ρ	100
TEB Goal Distance Weight	σ_g	1
No. of TEB Nodes s.t. COLREGs Constraint	m	3
Execution Time	T_{exe}	1 s
Prediction Horizon	T_{predict}	10 s
Prediction Time Step	T_{step}	2.5 s

Appendix B

Simulation 2 Details

Table B.1: Polar Coordinates of ASVs of 24 Scenarios in Simulation 2

Scenario No.	ASV 1	ASV 2	ASV 3	ASV 4	ASV 5
1		80, 0			
2		80, 90			
3		45, 180			
4		80, -135			
5		80, 0	80, 90		
6		80, 170	80, 135		
7		45, 180	80, 135		
8		80, 0	80, 90		
9		80, 150	80, 90		
10		80, -165	80, 90		
11		80, -90	80, 150		
12	80, 180	80, 0	80, 135	80, -170	
13		80, 170	80, -135	80, -170	
14		80, 170	80, 135	80, 90	
15		45, 180	80, 135	80, 90	
16		90, -135	90, -90	90, 90	
17		45, 180	90, -170	80, 135	
18		80, 165	80, 150	80, 45	
19		80, -165	80, 165	80, 45	
20		45, 180	80, 165	80, 90	
21		80, -165	80, 165	80, 90	
22		45, 180	80, 150	80, 90	
23		80, -165	80, 165	80, 90	80, -45
24		80, 0	80, 90	80, -90	80, -165

Table B.2: Summary of Minimum Relative Distance in Simulation 2

Scenario No.	Distance d (m)	Multiple of Length (d/L)
1	16.00	8.00
2	15.50	7.75
3	15.90	7.95
4	15.90	7.95
5	18.40	9.20
6	13.80	6.90
7	10.70	5.35
8	14.60	7.30
9	15.60	7.80
10	15.00	7.50
11	11.90	5.95
12	13.80	6.90
13	11.90	5.95
14	13.50	6.75
15	10.60	5.30
16	12.70	6.35
17	9.53	4.77
18	13.60	6.80
19	14.70	7.35
20	10.40	5.20
21	9.03	4.52
22	9.93	4.97
23	11.70	5.85
24	13.00	6.50

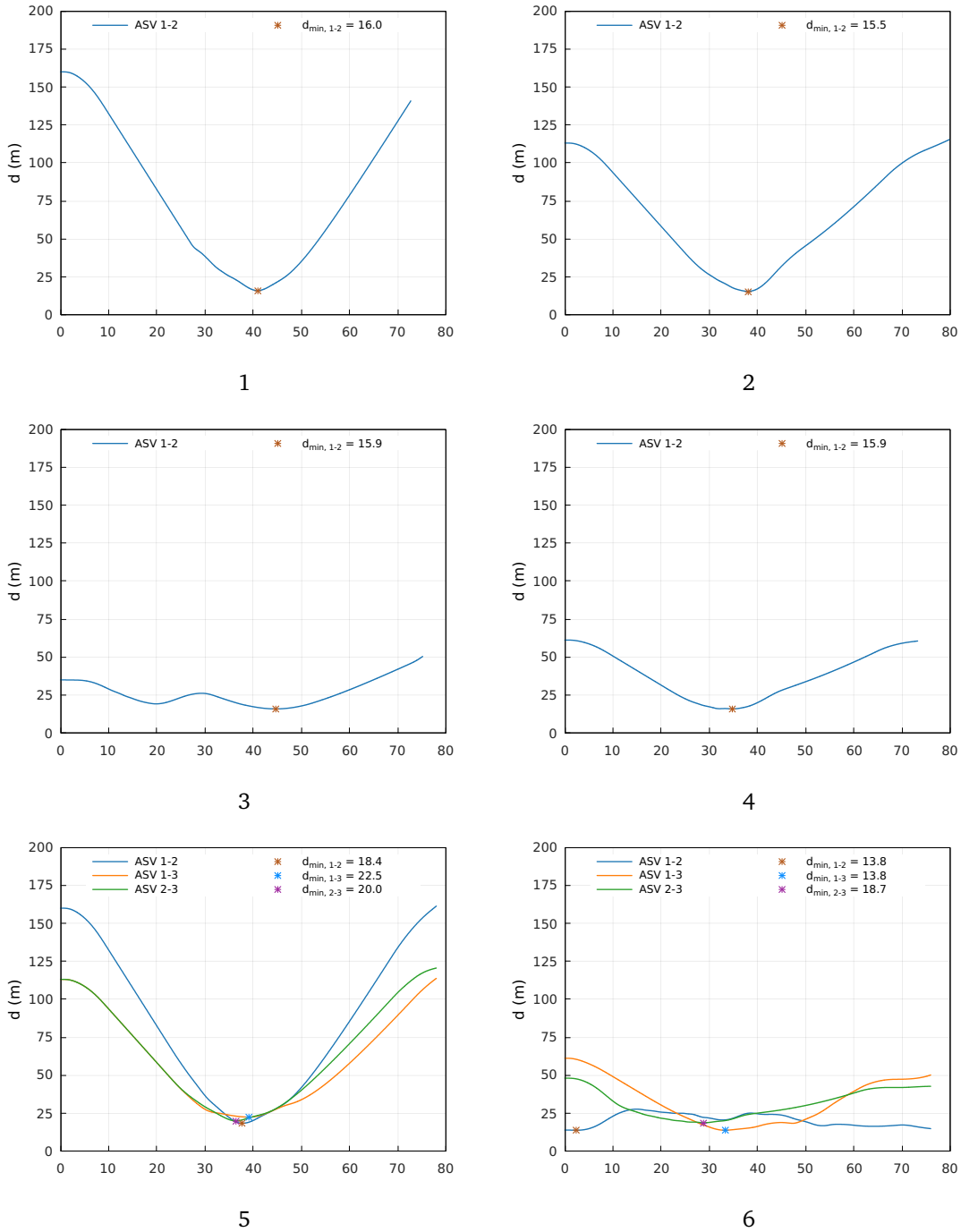


Figure B.1: Relative Distance of ASVs in Simulation 2 Scenario 1-6

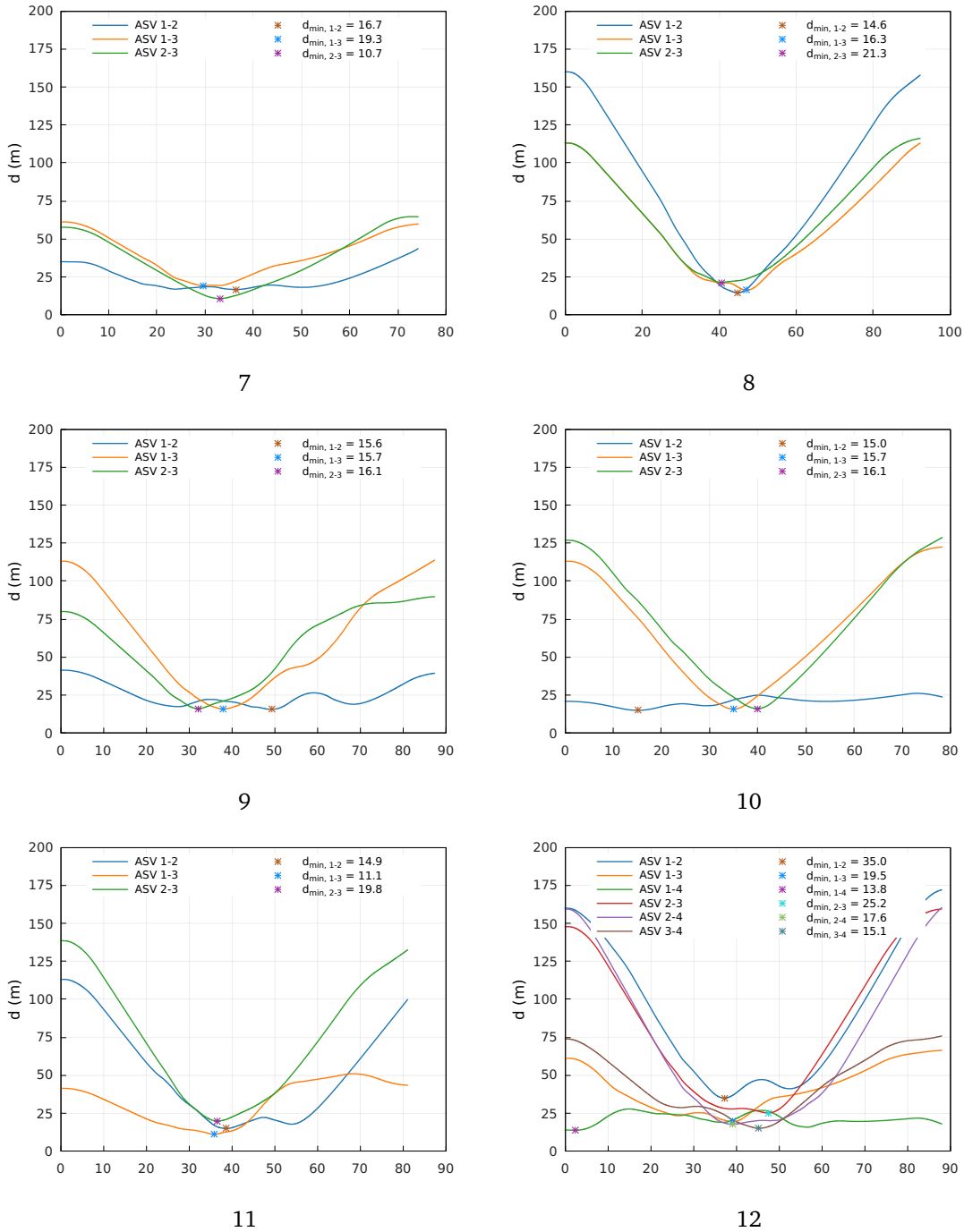


Figure B.2: Relative Distance of ASVs in Simulation 2 Scenario 7-12

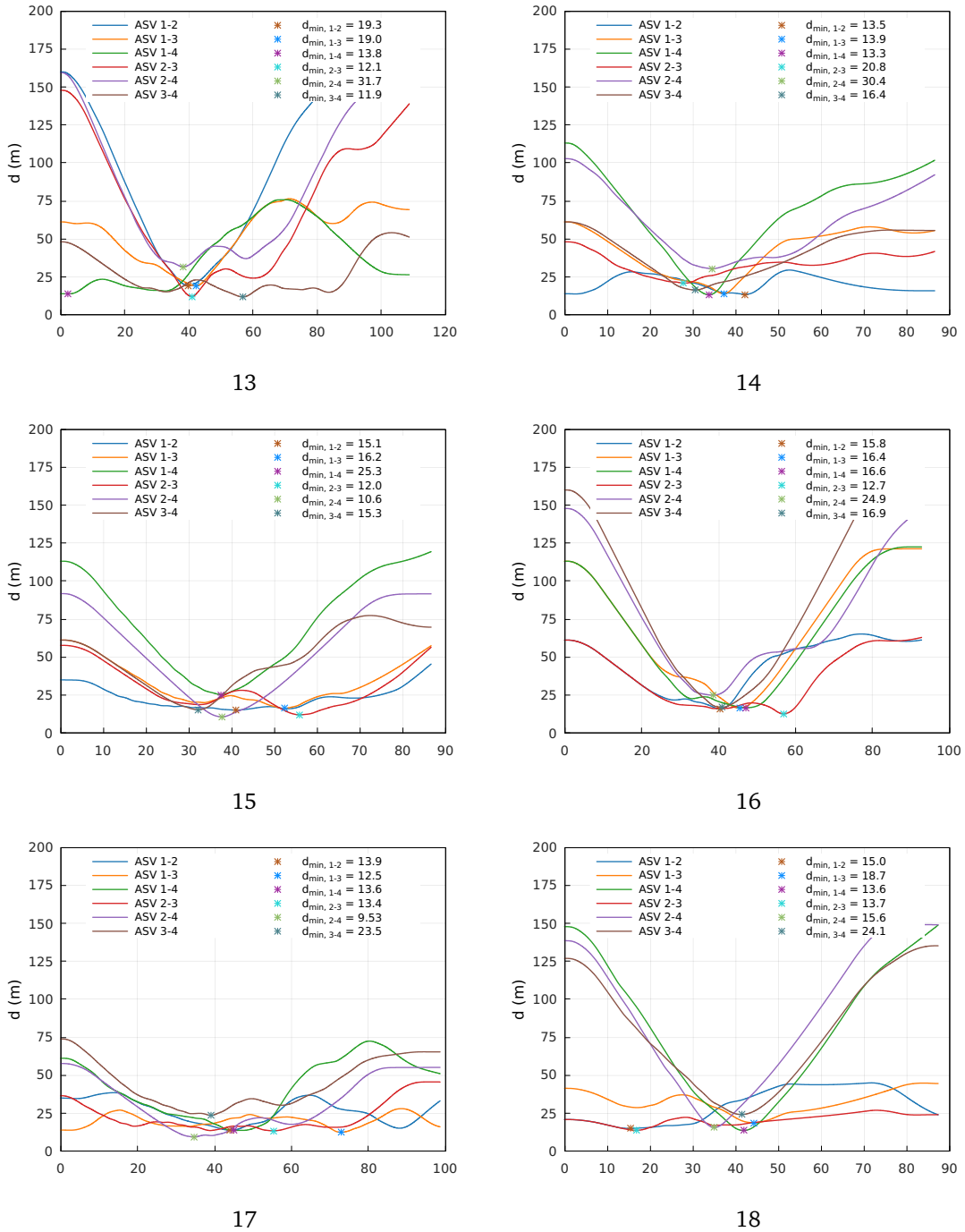


Figure B.3: Relative Distance of ASVs in Simulation 2 Scenario 13-18

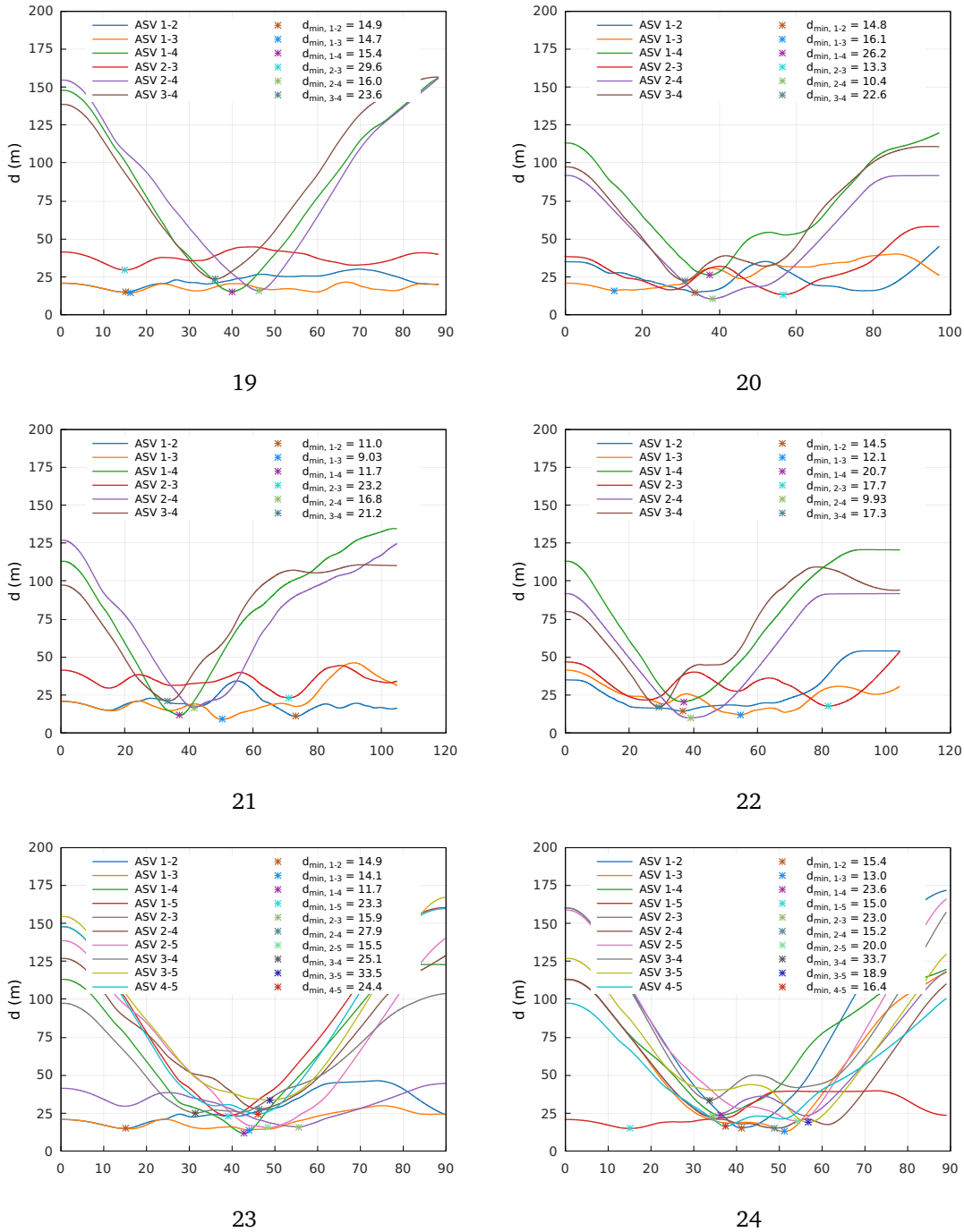
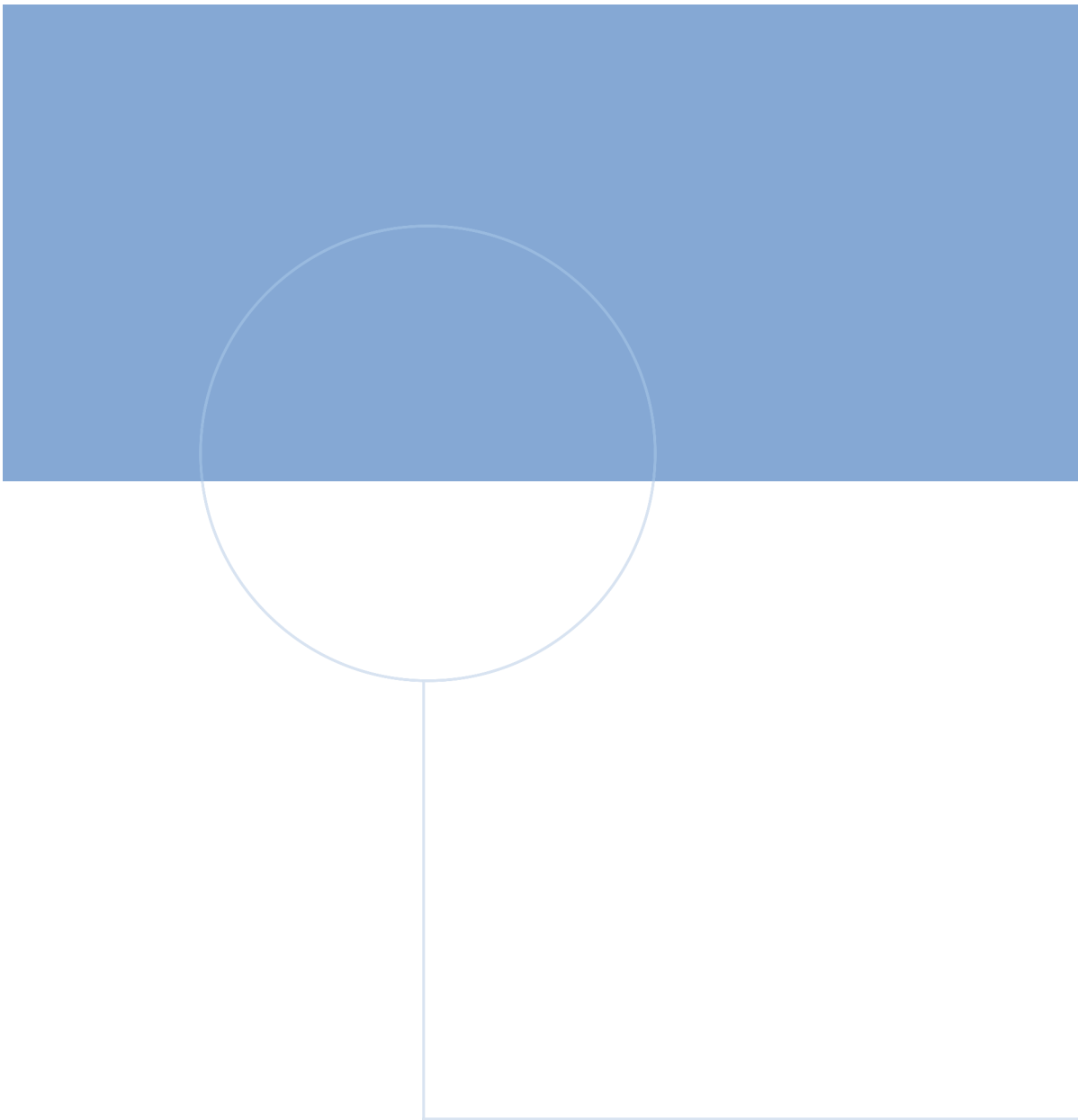


Figure B.4: Relative Distance of ASVs in Simulation 2 Scenario 19-24



NTNU

Norwegian University of
Science and Technology