Master's thesis

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

Jakob Stensvik Jensen

Reactive path-planning for autonomous harbor maneuvering

Master's thesis in Marine Technology
Supervisor: Roger Skjetne
June 2020

NTNU
Norwegian University of
Science and Technology

Jakob Stensvik Jensen

Reactive path-planning for autonomous harbor maneuvering

Master's thesis in Marine Technology
Supervisor: Roger Skjetne
June 2020

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

MASTER OF TECHNOLOGY THESIS DEFINITION (30 SP)

Name of the candidate: Jensen, Jakob Stensvik

Field of study: Marine control engineering

Thesis title (Norwegian): Reaktiv baneplanlegging for autonom havnemanøvrering

Thesis title (English): Reactive path-planning for autonomous harbor maneuvering

Background
For ships, the ability to safely maneuver in a harbor are especially critical and involve complex maneuvering by a skilled ship pilot. This involves understanding of:

- the ship dynamics (inertial delays, responses to currents, wind gusts, and propulsion, etc.),
- safety maneuvers in emergency situations (crash stop, evasive turn, Williamson turn, etc.),
- the hydrodynamic effects of maneuvering near harbor structures,
- the marine traffic rules (e.g., COLREGs),
- the optimal paths and speed regulation to avoid collisions with static and dynamic obstacles,
- the sensors, displays, and monitoring variables to use for necessary information feedback, and
- communication between the different crew members involved in the maneuver.

The objective of this thesis is to develop guidance methods for an autonomous ship maneuvering in a harbor, bringing the ship from an auto-voyage state to its final auto-docking state, based on dynamic, reactive, and repeated path-planning to avoid static and dynamic obstacles. One method is to use a discrete topologically organized artificial potential field with navigation functions to represent guidance information of the surrounding environment in the harbor. Then, online optimal and repeated path planning can be done directly on this landscape model to reactively propose new path segments and speed commands that brings the vessel safely to its location to initiate docking. A second method will generate a desired nominal path, but then combine this with the normal vector to the path by using two path parameters and speed assignments to reactively and continuously generate a collision-free path that brings the vessel to its docking location.

The project will consider fully-actuated ships that have DP functionality as a basis. Navigation and control systems must also be designed and put together with the guidance system.

Work description
1. Perform a background and literature review to provide information and relevant references on:
   - Ship maneuvering practices, incl. COLREGs.
   - Relevant ship sensors and instrumentation.
   - Artificial potential field, or similar, methods for path planning.
   - Path generation methods based on waypoints.
   - Relevant motion control designs.

   Write a list with abbreviations and definitions of terms and symbols, relevant to the literature study and project report.
2. Formulate the control problem, including definition of a case study, description of setup, vessel and its equipment, dynamical models, operation workspace, and specific assumptions and delimitations. Conclude with a problem statement.
3. With the task to move from waypoint $p0$ to pt, study, design, and implement a guidance model method to create a topologically organized landscape that represents necessary environment information for safely maneuvering in a harbor. Consider what is sufficiently high resolution to achieve acceptable reactiveness to dynamic obstacles.
4. For maneuvering on a dynamic guidance model (e.g. artificial potential field created by a navigation function), develop a recursive path-generation algorithm that ensures a smooth ($C3$) curve, also in

the connection points. Determine also a heading control strategy and derive the heading reference along the path. Under normal circumstances, there is a nominal path all the way to the target, that will be followed by default. However, the next waypoint can be assumed to be selected by the path-planner at some point along the current path segment, given by the evolution of the guidance model, with enough time to replan and compute next segment. Make a seamless integration between these path-planning and path-generation functions. Present the result by simulation on your case studies.

5. Prepare CSEI for use in experiments (and also as a simulation study if lab becomes unavailable). Tune the VSP servos, measure actuator forces, and verify input to thruster mapping and update the handbook with methods and results. Calibrate the simulation model according to physical testing.
6. Implement and test guidance methods with CSEI in MC-Lab, or in simulations if lab becomes unavailable. Analyze and discuss the results.

Tentatively:

7. Consider an alternative setup, where there is a nominal path all the way to the target, and parametrize the workspace by a path-parallel reference frame {PP} that follows the desired position along the nominal path. Define the desired position $p_d(s)$ as a point that can be offset to the path in {PP}, e.g. along the normal path vector. Suggest how $p_d(s)$ can be reactively updated to achieve a collision-free path that safely makes the vessel pass obstacles and brings it to the docking location.

Specifications

Every weekend throughout the project period, the candidate shall send a status email to the supervisor and co-advisors, providing two brief bulleted lists: 1) work done recent week, and 2) work planned to be done next week.

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, problem, design, results, and critical assessments. 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 70 A4-pages, 100 B5-pages, from introduction to conclusion, unless otherwise agreed upon. It shall be written in English (preferably US) and contain the elements: Title page, abstract, acknowledgement, project definition, list of symbols and acronyms, table of contents, introduction (project motivation, objectives, scope and delimitations), background/literature review, problem formulation, method, results, conclusions with recommendations for further work, references, and optional appendices. 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, which is taken very seriously by the university and cause consequences. NTNU can use the results freely in research and teaching by proper referencing, unless otherwise agreed upon.

The thesis shall be submitted with an electronic copy to the main supervisor and department according to NTNU administrative procedures. The final revised version of this thesis description shall be included after the title page. Computer code, pictures, videos, dataseries, etc., shall be included electronically with the report.

Start date: 15 January, 2020 **Due date:** As specified by administration

Supervisor: Roger Skjetne
Co-advisor(s): Zhengru Ren, Mathias Marley, Einar Ueland

Trondheim, 24.03.2020

Roger Skjetne

Digitally signed by Roger Skjetne
Date: 2020.03.24 11:34:29 +01'00'

Roger Skjetne
Supervisor

Abstract

This thesis has proposed two guidance methods for an autonomous ship maneuvering in a harbor and integrated this guidance system with a navigation and control system. The complete system was put together with the purpose of bringing the ship from a transit state to its final docking state.

The first guidance method path-plans in an online and repeated manner based on the artificial potential field framework with a navigation function. The workspace is partitioned using a grid with a sufficiently high resolution to achieve reactive maneuvering to dynamic obstacles. The path-planner is evolved to achieve fewer changes in navigation direction and to comply with the Convention on the International Regulations for Preventing Collisions at Sea (COLREGs) rules 14 and 15. A hybrid path generation method is used to generate a feasible path that concatenates the waypoints.

The second guidance method parametrizes the workspace by a path-parallel reference frame that follows the position along a nominal path. A path to the destination is generated by reactively and continuously combining the nominal path with a normal vector path using two path parameters and speed assignments. The latter also ensure that a path-tangential heading signal can be provided and that the ship converges to the destination. The proposed reference signal allows for a path to be made in accordance with COLREGs rules 14 and 15 if one assumes only a single obstacle.

The navigation system consists of a nonlinear passive observer that is used to reduce measurement noise and create state estimates. The control system solves the maneuvering control problem by a cascade-backstepping design for each of the guidance methods.

The autonomous system has been verified through simulations using a model of CyberShip Enterprise I with data from bollard pull tests to model the thrusters and the thruster allocation. The results showed that both guidance methods were able to complete the task safely and in accordance with COLREGs rules 14 and 15. The first guidance method was able to construct a collision-free but inefficient path because of unnecessary large safety margins to the obstacle. It was also shown that the hybrid path parametrization yielded a path with too sharp turns for the ship's heading to follow. The second guidance method was able to provide a collision-free and efficient path to the destination which the ship was able to follow well. The speed assignments made the ship follow along the nominal path at the proposed reference signal and converge to the destination.

Sammendrag

Denne avhandlingen har foreslått to guidemetoder for et autonomt skip som manøvrerer i en havn og satt sammen dette guidesystemet med et navigasjons- og kontrollsystem. Det komplette systemet ble satt sammen med det formål å bringe skipet fra en transit-tilstand til sin avsluttende dokkingtilstand.

Den første guidemetoden baneplanlegger på en direktekoblet og gjentakende måte basert på det kunstige potensialfelt-rammeverket med en navigasjonsfunksjon. Arbeidsområdet er partisjonert ved å bruke et nett med en tilstrekkelig høy oppløsning for å oppnå reaktiv manøvrering til dynamiske hindringer. Baneplanleggeren er utviklet for å oppnå færre endringer i navigasjonsretning og for å etterkomme konvensjonen om de internasjonale forskriftene for å forhindre kollisjoner til sjøs (COLREGs) regler 14 og 15. En hybrid banegenerasjonsmetode er brukt til å generere en gjennomførbar bane som konkatenerer veipunktene.

Den andre guidemetoden parametriserer arbeidsområdet gjennom en baneparallell referanseramme som følger posisjonen langs med en nominell bane. En bane til destinasjonen er generert ved å reaktivt og kontinuerlig kombinere den nominelle banen med en normalvektorbane ved å bruke to baneparametere og fartsoppdrag. De sistnevnte forsikrer også at et banetangentielt retningssignal kan bli levert og at skipet konvergerer til destinasjonen. Det foreslåtte referansesignalet lar en bane bli lagd i henhold til COLREGs regler 14 og 15 hvis man antar én enkelt hindring.

Navigasjonssystemet består av en ikke-lineær passiv estimator som brukes til å redusere målestøy og lage estimater av tilstandene. Kontrollsystemet løser manøvreringskontroll-problemet gjennom et tilbakestegsdesign i kaskade for hver av guidemetodene.

Det autonome systemet har blitt verifisert gjennom simuleringer ved å bruke en modell av CyberShip Enterprise I med data fra pullertrekktester for å modellere thrusterene og thrusterallokeringen. Resultatene viste at begge guidemetodene klarte å fullføre oppgaven trygt og i henhold til COLREGs regler 14 og 15. Den første guidemetoden klarte å konstruere en kollisjonsfri, men ineffektiv bane på grunn av unødvendige store sikkerhetsmarginer til hinderet. Det ble også vist at den hybride baneparametriseringen resulterte i en bane med for krappe svinger for skipets navigasjonsretning til å følge. Den andre guidemetoden klarte å levere en kollisjonsfri og effektiv bane til destinasjonen som skipet klarte å følge på en bra måte. Fartsoppdragene gjorde at skipet fulgte langs med den nominelle banen på det foreslåtte referansesignalet og konvergerte til destinasjonen.

Preface

This thesis was written as a finalization of my master's degree in Marine Technology with a specialization in the field of Marine Cybernetics at the Norwegian University of Science and Technology. I engaged in the work presented in this thesis from January 2020 to June 2020 with the guidance of my supervisor Professor Roger Skjetne from the Department of Marine Technology. The literature review of this thesis is partially based on a project thesis that I wrote as a preparatory study in the autumn of 2019.

The initial stages of the process consisted of updating the model ship CyberShip Enterprise I's thrusters and guidelines after experiments in the Marine Cybernetics Laboratory. This was done as a preparation for the testing of the developed system in the laboratory later on in the process. During this period, I started the writing process by formulating the problem and completing the literature review. After this was finalized, I spent most of my time either developing or writing about the methods to solve the problem. Later, the COVID-19 pandemic caused the lab experiments to be canceled which meant that the time spent in the laboratory was not as relevant as we initially hoped. In the end, only simulations were done to verify the developed system.

Throughout the entire process, Professor Roger Skjetne and co-advisors Zhengru Ren and Mathias Marley have been available via email for help and advice. Professor Roger Skjetne has provided me with notes on guidance and maneuvering designs that gave me good ideas on how to solve the problem. Zhengru Ren arranged several individual guidance meetings with me early on in the process and also provided valuable feedback on my problem formulation. Co-advisor Einar Ueland and Torgeir Wahl provided help when we conducted the experiments in the laboratory.

Trondheim, 7th June, 2020

Jakob Stensvik Jensen

Acknowledgments

I would like to thank my supervisor Professor Roger Skjetne for his dedicated support and guidance throughout the project. His insight and knowledge into the subject matter has provided me with good ideas on how to solve the problem.

I would also like to thank my co-advisors Mathias Marley, Einar Ueland, and especially Zhengru Ren. Furthermore, I would like to thank Torgeir Wahl for helping us to solve technical issues at the Marine Cybernetics Laboratory, and Caroline Fleischer for cooperating on the preparatory work done on CyberShip Enterprise I. Finally, special thanks to Elias Gauslaa for his support and encouragement throughout the project.

Contents

List of Figures

List of Tables

Abbreviations and symbols

List of abbreviations

AAWA	Advanced Autonomous Waterborne Applications Initiative
AIS	Automatic Identification System
APF	Artificial potential field
ASV	Autonomous surface vehicle
BT	Bow thruster
CLF	Control Lyapunov function
COLAV	Collision avoidance
COLREG	Convention on the International Regulations for Preventing Collisions at Sea
CSEI	CyberShip Enterprise I
DGNSS	Differential Global Navigation Satellite System
DOF	Degrees-of-freedom
DP	Dynamic positioning
DVL	Doppler Velocity Log
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
IMT	Department of Marine Technology
IMU	Inertial Measurement Unit
Lidar	Light detection and ranging
MC-Lab	Marine Cybernetics Laboratory
NED	North-East-Down
NTNU	Norwegian University of Science and Technology
OS	Own-Ship
PID	Proportional-integral-derivative
SA	Situational awareness
SLAM	Simultaneous Mapping and Localization
TS	Target-Ship
UGAS	Uniformly Globally Asymptotically Stable
UGES	Uniformly Globally Exponentially Stable
USV	Unmanned surface vessel
VRU	Vertical Reference Unit
VSP	Voith Schneider propeller
WP	Waypoint

List of symbols

$\{b\}$	Body frame
$\{n\}$	NED frame
$a_{k,i}, b_{k,i}$	Coefficients of order k for i-th subpath
$\alpha, \alpha_p, \alpha_\psi$	Virtual controls
$\alpha_{i,cmd}$	Commanded azimuth angle to i-th actuator
B	Extended thrust configuration matrix/Beam
b	Bias vector
$\hat{b}$	Estimated bias
b_1, b_2, b_3	Bias
$\hat{b}_1, \hat{b}_2, \hat{b}_3$	Estimated bias
β	Spherical function
C	Coriolis-centripetal matrix
C_A	Added mass Coriolis-centripetal matrix
C_{RB}	Rigid body Coriolis-centripetal matrix
$\mathcal{C}^r$	Curve continuously differentiable r times
χ	Course
χ_{rel}	Relative course between TS and OS
χ_{TS}	Course of TS
D	Linear damping matrix
$D(\nu_r)$	Damping matrix
$D_n(\nu_r)$	Nonlinear damping matrix
$\mathcal{D}$	Euclidean two-dimensional disc
d	Offset from nominal path
$\tilde{d}$	Difference in offset and reference signal
d_0	Neighborhood radius
d_E	Euclidean distance between OS and TS
d_{proj}	Scalar projection
d_{ref}	Reference signal
d_{safe}	Safe passing distance
$\Delta_{p,i}$	I-th gain for hyperbolic function
Δ_ψ	Limit on heading change
δ	Shortest allowable distance to TS
ϵ	Small constant
ε	Path parameter for each subpath
η	Position and heading
$\hat{\eta}$	Estimated position and heading
$\tilde{\eta}$	Errors in position and heading
η_0	Initial position and heading
η_d	Desired position and heading
f	Thrust
f_{cmd}	Commanded thrust
$f_{i,cmd}$	Commanded thrust to i-th actuator
$g(\eta)$	Restoring forces
γ	Distance to destination

I_z	Moment of inertia about the z-axis
i	I-th subpath
J	Skew-symmetric matrix
J_Θ, T_Θ	Transformation matrices
K_1, K_2	Gains
$K_{1,p}, k_{1,\psi}$	Gains for decoupled controller
k_i	Scale of thrust for i-th actuator
κ	Navigation function tuning parameter
L_1, L_2, L_3	Injection gain matrices
L_{cell}	Length of cell
L_{oa}	Length overall
$L_{x,BT}$	Length in x to BT
$L_{x,VSP}$	Length in x to VSP
$L_{y,VSP}$	Length in y to VSP
λ	Positive design parameter
λ_ψ	Weighting factor
M	Inertia matrix
M_A	Added mass matrix
M_{RB}	Rigid body mass matrix
m	Mass/Number of obstacles
μ	Non-negative tuning constant
N_d	Unit normal vector
n	Number of subpaths
n_{HO}	Number of virtual obstacles in head-on scenario
ν	Velocity vector
$\hat{\nu}$	Estimated velocity
$\tilde{\nu}$	Errors in velocity
ν_c	Ocean current velocity
ν_r	Relative velocity
$\mathcal{O}$	Sphere
$\omega, \omega_1, \omega_2$	Update laws
p	Position
p_0	Initial position
$p_{0,TS}$	Initial position of TS
p_d	Desired/Combined path
$p_{d,i}$	I-th desired subpath
p_i, q_i	Center points of sets
p_j	Neighboring cell position
p_k	K-th WP
p_n	Next WP
p_q	Current WP
p_t	Destination
p_{TS}	Position of TS
p, q, r	Angular rates in $\{b\}$
ϕ	Roll angle/Relative bearing
ϕ, θ, ψ	Attitude in $\{n\}$

ϕ_{nf}	Navigation function
ψ_j	Angle between p_q and p_j
ψ_q	Angle from the line between p_q and the previous cell position
ψ_{q_N}	Heading correction
q_d	Nominal path
q_N	Normal vector path
R	Circle of acceptance radius
$R(\psi)$	Rotation matrix
$R_2(\psi)$	Second rotation matrix
r	Yaw rate/Radius of sphere/Number of differentiability/Number of thrusters
r_{GW}	Radius of virtual obstacle in give-way scenario
r_{HO}	Radius of virtual obstacle in head-on scenario
r_{TS}	Radius of TS
ρ	Radius of disc/Hyperbolic function/Tuning function
$S(r)$	Skew-symmetric matrix
$S_2(r)$	Second skew-symmetric matrix
$\mathcal{S}$	Star-shaped set
s, s_1, s_2	Path parameters along nominal and normal vector path
$\sigma_\psi, \sigma_\upsilon$	Activation functions for heading correction and speed assignment
T_1, T_2	Diagonal matrices of time constants for z_1- and z_2-subsystems
$T_{1,p}$	Diagonal matrix of time constants for $z_{1,p}$-subsystem
$T_{1,\psi}$	Time constant for $z_{1,\psi}$-subsystem
$T_d, T_{0,i}$	Unit tangent vector
T_u, T_v, T_r	Time constants for z_2-subsystem
T_x, T_y, T_ψ	Time constants for z_1-subsystem
t	Time
τ	Forces and moment
τ_{cmd}	Commanded generalized forces and moment
τ_d	Desired forces and moment
τ_{env}	Environmental loads
τ_{waves}	Wave loads
τ_{wind}	Wind loads
Θ	Attitude vector
u	Control input
u_d	Reference speed
$u_{i,cmd}$	Control input to i-th actuator
u_N	Desired speed along the normal vector path
u_{TS}	Speed of TS
$\upsilon, \upsilon_1, \upsilon_2$	Speed assignments
V, V_1, V_2	Lyapunov functions
$V_{1,p}, V_{1,\psi}$	Lyapunov functions
v	Zero-mean Gaussian measurement noise
v_u, v_v, w	Linear velocities in $\{b\}$
$\hat{v}_u, \hat{v}_v, \hat{r}$	Estimated velocities
w	Zero-mean Gaussian process noise
$\mathcal{W}$	Workspace

X, Y, N	Forces in surge and sway and moment in yaw
X_d, Y_d, N_d	Desired forces in surge and sway and moment in yaw
$X_u, Y_{v/r}, N_{v/r}$	Hydrodynamic damping coefficients
$X_{\dot{u}}, Y_{\dot{v}}, Y_{\dot{r}}, N_{\dot{r}}$	Hydrodynamic added mass coefficients
x_g, y_g	Center of gravity along x- and y-axis
x_k, y_k	Coordinates of k-th WP
x_n, y_n	Coordinates of next WP
x_{TS}, y_{TS}	Position of TS
$\hat{x}, \hat{y}, \hat{\psi}$	Estimated position and heading
x_d, y_d, ψ_d	Desired position and heading
x_m, y_m, ψ_m	Measurement signal
$x_{q_d}, y_{q_d}, \psi_{q_d}$	Nominal position and heading
x, y, z	Position
y_m	Measurement signal vector
z, z_1, z_2	Error variables
$z_{1,p}, z_{1,\psi}$	Error variables
$z_{2,p}, z_{2,\psi}$	Error variables

Chapter 1

Introduction

1.1 Motivation

Maneuvering a ship in a harbor is a complex task and requires a skilled ship pilot that has knowledge about the ship dynamics and hydrodynamic effects, the marine traffic rules, collision avoidance (COLAV) methods, and information feedback from sensors and displays. Safety of the operation can not be ensured if the pilot fails to meet this requirement.

A large part of the collisions at sea are caused by human errors. Therefore, over the past decade, there has been an increasing amount of interest in the research regarding autonomous vessels. The critical operation of maneuvering a ship could benefit greatly from increasing the level of autonomy. The number of accidents caused by human errors would decrease with human involvement in direct control of the ships. Reducing the size of the crew onboard would reduce the fatality rate of the crew, lower the operational cost, and allow for a ship design that makes for a more efficient use of space.

An autonomous ship will not be given mission commands by a ship pilot. Therefore, the ship's guidance system should be able to plan and replan its path after gathering information about its surroundings in the harbor as the operation is carried out. For an autonomous ship to operate in a safe and reliable way, it needs to ensure that proper and effective action can be taken to avoid collisions on its way to its destination. In the harbor, recreational vessels make up a large part of the traffic. Therefore, it is crucial that reactive maneuvering is achieved. The ship will be operating close to small vessels which may not be detected by the situational awareness (SA) system at an adequate range. In these scenarios, there will be limited time to take evasive action.

1.2 Objectives

The thesis will provide answers to the following objectives that will contribute to developing an autonomous system that brings a ship safely from an auto-voyage state to its final auto-docking state:

1. Provide a literature review on relevant topics such as ship maneuvering practices, ship sensors, path-planning methods using artificial potential field (APF), path generation methods based on waypoints (WPs), and motion control designs.
2. Design a guidance method that uses the navigation function in a path-planner and provide a stepwise path generation method that connects the WPs with a feasible path.
3. Design a guidance method where a nominal path is combined with a normal vector path to create a collision-free path by using two path parameters and speed assignments.
4. Develop and integrate observer and control designs with the guidance system.
5. Verify the autonomous system through simulations.

1.3 Scope and delimitations

The scope of this thesis is narrowed down by the assumptions and simplifications that are listed in Section 3.1.1.

1.4 Contributions

The main contributions of this thesis comprise a complete system for an autonomous ship maneuvering in a harbor. Two guidance methods are proposed and integrated with established navigation and control systems. The complete system has been verified through simulations.

A path-planner based on an established APF framework is evolved to achieve fewer changes in navigation direction and to comply with COLREGs rules 14 and 15. This path-planner is integrated with an established hybrid path generation method.

An established path parametrization method used for docking purposes is evolved to construct a collision-free path to the destination. This is done in accordance with COLREGs rules 14 and 15 if one assumes only a single obstacle.

Lastly, CyberShip Enterprise I's (CSEI) thrusters and guidelines were updated. The VSP servos were tuned and several bollard pull tests were performed. The work culminated with an update of NTNU (2020) with guidelines on how to perform these tasks.

1.5 Outline

The thesis is organized as follows:

Chapter 2 provides background information and relevant references on topics such as ship maneuvering practices, ship sensors and instrumentation, path-planning methods using APF, path generation methods based on WPs, and motion control designs.

Chapter 3 presents the problem formulation. This includes description of the system, definition of the case study, operation workspace, dynamical models, and a list of specific assumptions and simplifications. It concludes with the problem statement.

Chapter 4 provides a nonlinear passive observer that is used to reduce measurement noise and create state estimates.

Chapter 5 presents the first guidance method where the navigation function is used in a path-planner to guide the vessel safely to its destination. A stepwise hybrid path parametrization is proposed to generate a feasible path connecting the WPs.

Chapter 6 presents the second guidance method where a nominal path is combined with a normal vector path by using two path parameters and speed assignments. This is done to reactively and continuously generate a collision-free path that brings the vessel safely to its destination.

Chapter 7 presents the control system that solves the maneuvering control problem.

Chapter 8 presents the model ship used in the simulation study and the preparatory work done on the vessel.

Chapter 9 presents the results from the simulations of the two guidance methods. The results are analyzed and discussed.

Chapter 10 covers the conclusions and recommendations for further work.

Appendix A presents the videos that show simulations of the two guidance methods.

Chapter 2

Background

This chapter presents a background study that will provide information and references on ship maneuvering practices, ship sensors and instrumentation, path-planning methods using APF, path generation methods based on WPs, and motion control designs. The study is partially based on Jensen (2019), a project thesis that was written as a preparatory study for this thesis.

The ship under control is defined as an Own-Ship (OS), while a stationary or moving obstacle is named a Target-Ship (TS), this notation will be used throughout the thesis.

2.1 Ship maneuvering practices

2.1.1 Rules and regulations

A ship maneuvering in a harbor must comply with the special rules and regulations governing the harbors as well as the rules from COLREG (1972). The latter are important to follow to ensure that safe maneuvering can be guaranteed. COLREGs rules 6, 8, 13-15, and 17 are frequently used in rule-compliant COLAV solutions (Eriksen and Breivik, 2017; Hagen et al., 2018). These rules specify how a ship shall maneuver with a safe speed and how to act to avoid collision and in overtaking, head-on, give-way, and stand-on situations. The following are parts of the descriptions of the rules:

- **Rule 6 (Safe speed):** Safe speed is important to ensure that proper and effective action can be taken to avoid collisions and to stop within a suitable distance of the hazard. The marine traffic rules point out some of the factors that one needs to take into account before determining the speed. Based on the conditions in the harbor and the abilities of the vessel, one should consider:
  - the visibility,
  - the traffic density,
  - the stopping distance and turning ability, and
  - the wind, waves and current.

- **Rule 8 (Action to avoid collision):** Any alteration of course and/or speed to avoid collision shall, if possible, be large enough to be noticeable to an observing vessel. A succession of small alterations of course and/or speed should be avoided. If necessary to avoid collision, a vessel shall slacken its speed or make a complete stop.
- **Rule 13 (Overtaking):** Any vessel overtaking any other shall keep clear of the vessel being overtaken. A vessel approaching another vessel from a direction of more than 22.5° abaft its beam is an overtaking vessel. The overtaking vessel is not relieved of the duty of keeping clear of the overtaken vessel until it is finally past and clear, even with subsequent alteration of the bearing between the two vessels.
- **Rule 14 (Head-on):** Any meeting of two power-driven vessels on reciprocal or nearly reciprocal courses that would involve risk of collision requires each vessel to alter its course to starboard so that each shall pass on the port side of the other.
- **Rule 15 (Give-way):** Any crossing of two power-driven vessels that would involve risk of collision requires the vessel which has the other on its own starboard side to keep out of the way and it shall, if possible, avoid crossing ahead of the other vessel.
- **Rule 17 (Stand-on):** Any situation where one of two vessels is to keep out of the way, the other is required to keep its course and speed. The latter vessel may however take evasive action to avoid collision if it becomes clear that the vessel required to keep out of the way is not taking appropriate action.

The correct vessel behavior for head-on, give-way, stand-on, and overtaking scenarios is shown in Figure 2.1.

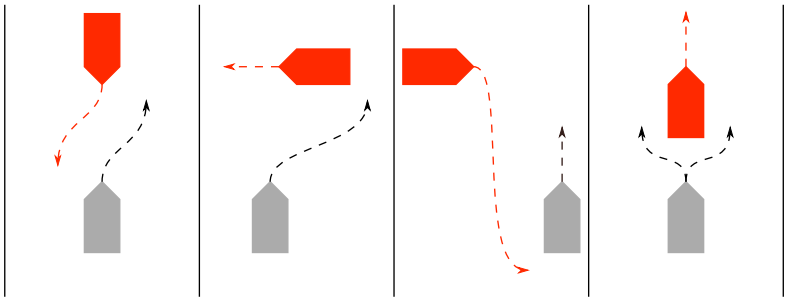


Figure 2.1: From left: The correct vessel behavior for head-on, give-way, stand-on, and overtaking scenarios. Adapted from Hagen et al. (2018).

Figure 2.2 displays a graphical interpretation of the COLREGs scenarios for overtaking, head-on, give-way, and stand-on that is proposed in Eriksen and Breivik (2017). Tam and Bucknall (2010) propose a similar framework to categorize the position of the TS but also take into account the relative course to define the encounter type.

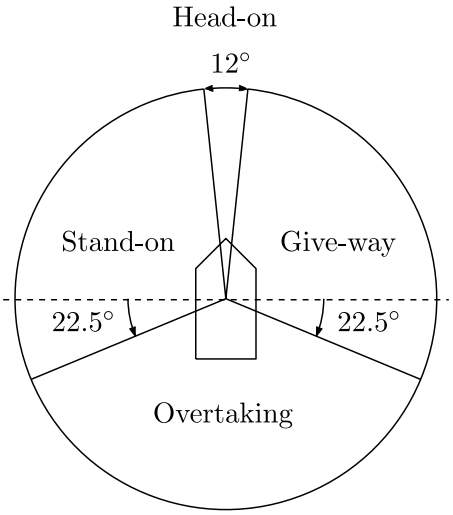


Figure 2.2: Graphical interpretation of overtaking, head-on, give-way, and stand-on situations. Adapted from Eriksen and Breivik (2017).

Murdoch et al. (2012) identify "Golden Rules" that are important to follow when maneuvering in the harbor and during the docking procedure. These include:

- The vessel should arrive with a slow speed and with a controlled approach.
- The bridge team must ensure that a passage plan is made from berth to berth and that the pilot is well briefed about the ship's speed and maneuvering characteristics.
- The crew should be fully aware of the expected maneuvers and what is expected of them to improve safety and efficiency.
- The main engines and thrusters should be checked before approaching the quay so that they are fully operational.

2.1.2 COLAV methods

Tam et al. (2009) review the development of COLAV methods for ships from the 1950s to the early 2000s. They address the shortcomings of ship COLAV and path-planning methods, particularly when operating within close range to other vessels. Some of the common limitations include:

- The environmental conditions are disregarded in the path-planning algorithm.
- None of the studies include true dynamic TSs, only semi-dynamic TSs (i.e., TSs with constant course and speed).
- Most of the ship models were highly idealized (i.e., assumption of small or no change in speed).

Liu et al. (2016) point out other common limitations such as the disregarding of regulations in COLAV algorithms (e.g., COLREGs) and not balancing efficiency and effectiveness.

Huang et al. (2020) discuss how state-of-the-art COLAV studies for manned vessels can benefit the research and development of unmanned vessels. They also provide an overview of COLAV techniques and divide them into three fundamental processes called motion prediction, conflict detection, and conflict resolution. Figure 2.3 shows the information flow between these processes and the ship with observer and actuators.

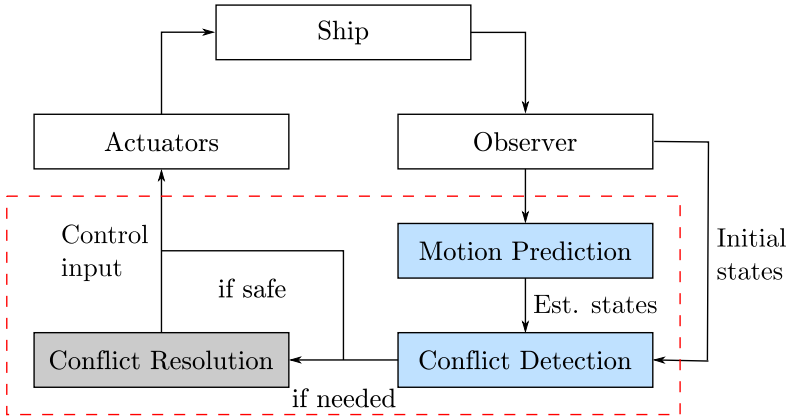


Figure 2.3: Information flow of motion prediction, conflict detection, and conflict resolution modules. Adapted from Huang et al. (2020).

Motion prediction

The methods for predicting the TS's trajectory can be categorized into three groups based on the knowledge of the TS:

- Physics-based methods predict the motion of the TS based on the laws of physics and are the simplest way to predict the trajectory of the TS. These methods are based on the assumption that the TS will keep its speed and course while neglecting environmental disturbances.
- Maneuver-based methods use the ship maneuvering that is learned or estimated from historical traffic data or regulations for marine traffic rules. The algorithms estimate the steering intentions before predicting the TS's trajectory.
- Interaction-aware methods are based on communication between the OS and the TS. The maneuvering intentions (e.g., intended course) are either broadcast, exchanged or negotiated, or the trajectory information is exchanged.

Figure 2.4 shows an illustration of the three different prediction methods. The TS is going southeast with a constant speed and course. The physics-based method predicts that the TS will keep this motion. The maneuver-based method recognizes the give-way intention and predicts that the TS will make a starboard turn as proposed by COLREGs. The interaction-aware method uses the broadcast trajectory from the TS as the predicted trajectory.

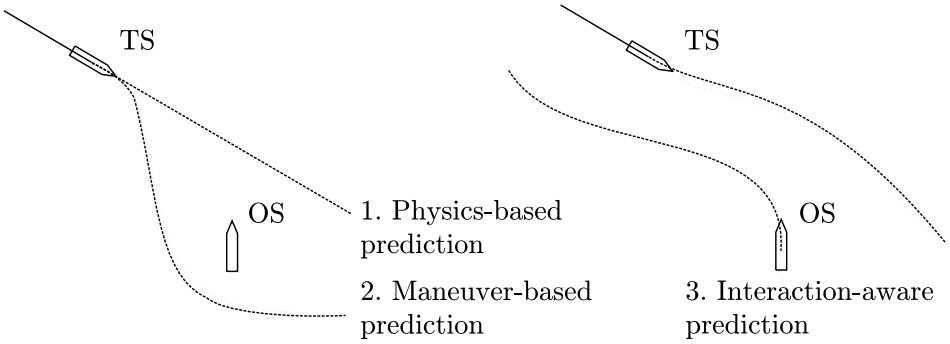


Figure 2.4: Physics-based, maneuver-based, and interaction-aware prediction methods of the TS's trajectory. Adapted from Huang et al. (2020).

The physics-based method is widely used because of its simplicity, but it is less accurate for COLAV as the predicted trajectories are usually represented by straight lines that are unrealistic. The maneuver-based and interaction-aware methods use more information in the prediction. The former is sensitive to errors of the estimated intention, specifically in close range encounters. The latter allows the ships to cooperate, either by exchanging intentions or trajectories. Sharing trajectories is an accurate method as the OS has a better understanding of its own dynamics than that of TSs, but use of this method is sensitive to failure in the communication link between the ships.

Conflict detection

The conflict detection process is about determining whether and when to take evasive action. In this process, a collision risk assessment is needed to determine if it is necessary to trigger an event that informs the human of collision dangers or asks the human or the autonomous system to find a collision-free path. Conflict detection is absolutely necessary in order to support humans or machines in avoiding collisions.

Collision risk, the likelihood of collision, can be represented numerically or graphically. If the former is applied, an event is triggered if a numerical value representing the collision risk violates a pre-set limit. A graphical representation could be rings of warning around the OS in a two-dimensional map. A collision alarm could be triggered if a TS enters this area. Both representations have their benefits, a numerical value representation would make it possible to compare cases, whereas a graphical representation would be easier to understand for the human.

Conflict resolution

The conflict resolution is the evasive action taken to create a collision-free path. The different techniques for COLAV are usually based on one or a hybrid of the following methods:

- Rule-based methods use a rule system to guide the vessel in different scenarios in order to prevent collisions. Regulations from COLREGs are often incorporated in the pre-set rules.

- Virtual vector methods use a generated virtual field to provide the vessel with a direction of motion.
- Resolution searches in the discrete solution-space with collision check are used to provide a collision-free or optimal solution.
- Resolution searches in the continuous solution-space with collision constraints are used to find the optimal solution.
- Replanning methods that search for solutions directly in the workspace.

2.1.3 Ship factors that affect maneuvering

Murdoch et al. (2012) identify the following as ship factors that affect maneuvering:

Actuators

The reduced speed can make it difficult to maintain control of the ship. As the propeller speed is reduced, the water flow into the rudder is also reduced making the rudder less effective. When the speed is low, a method called "kick ahead" could be used to initiate or maintain a turn. This is performed by putting the engines ahead for a short time to increase water flow into the rudder while not increasing the vessel's speed. The rudder should be fully applied before initiating the maneuver.

Pivot point and lateral motion

The ship's pivot point is the point located along its length which the ship rotates about. The placement of the pivot point is dependent on whether the ship is headway or stern-way, and the applied forces (e.g., from rudder and wind) while turning. Since the pivot point is not at the ship's center, the vessel will move laterally while turning. Therefore, it is important to know where the pivot point is located while maneuvering close to hazards in the harbor.

Wind and current

A ship is more vulnerable to wind and current at a low speed. The wind will affect heading and leeway of the ship. Especially high-sided ships will be affected from leeway. The ship will want to settle with the pivot point windward so that the point becomes aligned with the point of influence from wind. The latter depends on the wind direction and the ship's heading, and changes with the profile the ship has toward the wind.

The currents can be complex by having varying rates and directions that change hourly, as well as varying with depth. At many places, there is a counter current close to the bank that flows in the opposite direction of the main current. Therefore, to maneuver safely, local knowledge is needed.

Hydrodynamic forces

The hydrodynamic forces must be taken into account when the OS is interacting with banks and TSs. Shallow water, speed, and distance are factors that affect these hydrodynamic forces. When interacting with a TS, the hydrodynamic effects might make the OS turn toward or be drawn toward the TS. Similar effects can happen when navigating close to the bank and become more prominent with shallow water. It is therefore essential to ensure that a safe speed and passing distance are maintained to avoid collision or contact.

Water depth

The water depth in the harbor can vary from deep water to shallow water where there is danger of touching the bottom. As the water depth reduces, the ship will have increased resistance, the rudder will lose effects of the propeller slipstream, and the turning ability will get worse. These effects will make the ship hard to control.

2.1.4 Safety maneuvers

In emergency situations, the ship will need to perform a safety maneuver to avoid collision. Babicz (2015) presents common safety maneuvers. Crash stop sets the engines to full astern in order to stop the ship in the shortest possible time without turning. Crash stop from full speed is not as effective as turning the ship because the latter usually offers a better COLAV strategy that gives a shorter stopping distance. Therefore, crash stop is only recommended for ships that operate at a low speed.

An evasive turn should be performed if there is enough water around the ship and, if the circumstances admit, to starboard side. Williamson turn (also called man-overboard maneuver) is a useful safety maneuver that can be performed by turning the vessel 60° from its original course, and then turning it back the opposite way to bring the ship back on to the reciprocal course. This is illustrated in Figure 2.5.

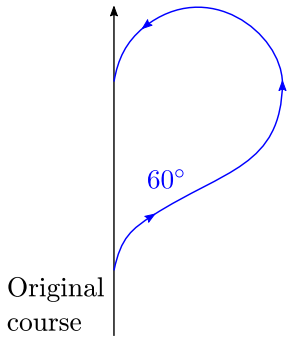


Figure 2.5: Williamson turn.

14th ITTC (1975) proposed some ship stopping and turning trials that can be performed to obtain important maneuvering characteristics of a ship. Crash stop and low-speed stopping trials can be performed to obtain the ship's head reach and maneuverability during emergency situations. The steady turning radius says something about how

the ship responds to course-changing maneuvers, and can be acquired by performing a turning circle.

2.1.5 Autonomous maneuvering

The autonomous aspect of maneuvering has been covered by the Advanced Autonomous Waterborne Applications Initiative (AAWA), a project led by Rolls-Royce with DNV GL involvement. AAWA (2016) investigates technical and safety aspects relevant to unmanned shipping.

Supervision and piloting

An autonomous vessel would need ways to communicate with the control center at the shore and sufficient connectivity is important to guarantee that remote control can be taken. Supervision of the ship would vary depending on the operation mode the ship is currently in. The reason is that the different operation modes (e.g., the harbor maneuvering and docking) require different levels of operator interaction, depending on how complex the mission is. During port approach it is suggested that the operator either can take remote control of the vessel or if just the supervision level is increased.

There are also suggestions for alternatives for future organization of pilots. An alternative is that the autonomous vessel is controlled by a capable pilot, or that the autonomous vessel operator holds a pilot license for the operational area, in this case the harbor.

Fallback strategy

A human off-ship should be able to intervene by remote control of the vessel if operability problems were to occur. A possible fallback strategy is needed in case of sudden reduction in connectivity with the ship simultaneously with harbor maneuvering problems. The following list is the proposed action in prioritized order:

1. Ask the operator to take manual control.
2. Slow down and proceed to the next WP.
3. Stop the ship and stay in dynamic positioning (DP) mode.
4. Return to the previous WP.
5. Navigate to a pre-set safe location.

Cybersecurity

As with other information and communications technology systems, there are concerns that regard the autonomous system's vulnerability to cyberattacks with the purpose to manipulate or exploit the system. Another threat would be intentional jamming and spoofing of the SA sensors, position reference systems, or communication signals between the ship and the control center at the shore.

Liability challenge

Legal thinking will be challenged if accidents involving autonomous ships were to happen. Many accidents will be rooted in system defects rather than human errors, meaning that liability for damages can not be based on human actions in the same way that is done today. There will be uncertainty whether the liability will be placed with, e.g., the manufacturer of the autonomous system, the user, or the owner.

2.2 Relevant ship sensors and instrumentation

Ship sensors and navigation systems are necessary to determine the craft's position, attitude, velocity, and acceleration. The equations of motion of a low-speed vessel can be simplified to a 3 degrees-of-freedom (DOF) system where surge, sway, and yaw need to be measured precisely to ensure DP functionality. Sørensen (2018) presents relevant position reference and sensor systems.

2.2.1 Position reference systems

Global Navigation Satellite System (GNSS): The most commonly used position reference system is GNSS. There are multiple systems under this category, but all of them are based on the same principles. The systems have satellites placed in a constellation in several orbital planes. It is necessary that at least four satellites are visible to the receiver to be able to determine a three-dimensional position. However, only three satellites need to be visible in order to determine the position of a ship at sea level. The most commonly used GNSS for ships is Global Positioning System (GPS) which is owned and operated by the United States Government.

An augmentation system which main purpose is to enhance the accuracy and the integrity of GNSS is differential GNSS (DGNSS). This augmentation system is normally used when measuring the position of a marine craft (Fossen, 2011). The system calculates the GNSS position errors by using a fixed receiver (e.g., on shore) with a known position to transmit corrections to the ship's position. With this system, horizontal position errors less than 1 m are achievable.

DNV GL (2018) proposes that the operational requirements for position fixing for autonomous ships, with an absolute position accuracy with 95% probability, should be 10 m for automatic collision-free operations and navigation in harbor entrances, harbor approaches, and coastal waters. The requirement for maneuvering in port is 1 m.

2.2.2 Sensor systems

Gyrocompass: The heading of a marine craft is usually measured with a gyrocompass. A gyrocompass differs from a magnetic compass by its ability to find the true north.

Vertical Reference Unit (VRU): Measurements from a VRU consist at a minimum of heave, roll, and pitch motions but can also include measurements of angular rates in most cases. The objective of a VRU is to adjust the GPS position measurements by roll

and pitch motions.

Doppler Velocity Log (DVL): Accurate measurements of the vessels velocity can be achieved by using a DVL. Different types of DVLs are laser-Doppler velocimeters and acoustic Doppler velocimeters (Fossen, 2020). Both systems are based on the Doppler shift effect, a change of wave frequency that occurs when an observer moves in relation to a wave source.

Inertial Measurement Unit (IMU): An IMU usually consists of accelerometers and angular rate gyros. Accelerometers measure the specific force in surge, sway, and heave, while angular rate gyros measure the rate of roll, pitch, and yaw. When using an IMU, it is necessary to integrate GPS in a state observer, which compensates for the bias drift terms, to obtain position and attitude.

Environmental sensors: Wind, draft, wave, and current sensors could be classified as environmental sensors. Wind speed and direction relative to the vessel are measured by a wind sensor. These sensors are commonly used for wind feedforward control. Marine crafts operating in a harbor with a wide range of drafts need a draft sensor. Wave sensors can measure significant wave height and direction of the waves, while current sensors measure the speed and direction of the current.

The minimum requirement to a sensor and navigation system for a DP system typically consists of at least one position reference system, one gyrocompass, one VRU for roll and pitch measurements, and one wind sensor. DNV GL (2019) classifies DYNPOS-AUTS, a DP system without redundancy, with the same requirement. The redundancy can be increased by adding multiple measurement devices. Using systems that apply different measurement principles would give full redundancy in hardware configuration.

SA sensors: SA sensors can be used to locate and avoid TSs. Radar is an important SA sensor for ships. This technology allows for ship maneuvering with no visibility at night or during bad weather. Another COLAV system that can supplement radar is Automatic Identification System (AIS). This is used to track TSs by sharing information about position, course, and speed.

Relevant SA sensors for autonomous guidance and navigation include light detection and ranging (lidar) technology and visual sensors. The problem is to combine these in an optimal way with respect to reliability and cost-effectiveness. Partially based on a sensor fusion study for autonomous applications (Mukhtar et al., 2015), AAWA (2016) proposes a sensor fusion for autonomous ships. The proposition consists of a lidar or radar for providing distances, velocities, and angular measurements of objects, while lower-cost and higher spatial resolution cameras could be used for classifying objects. Near-infrared cameras could be used during night-time.

Other SA sensors can be used to track TSs as substitutes when AIS is unavailable. Wolf et al. (2010) study SA systems for unmanned surface vessels (USVs) and develop a solution for an object-level tracking and change detection method using several cameras

to provide a 360° panoramic view. This method ensures that targets are detected and that their locations are confirmed. It also recognizes changes in the surrounding environment when the USV is operating on patrol.

Another problem is that weak GNSS signals are received when operating close to bridges and other sheltered environments. This problem could be addressed by using Simultaneous Mapping and Localization (SLAM) when navigating an ASV in these scenarios. The SLAM method developed in Leedekerken et al. (2014) uses imaging sonar, radar, lidar, and camera sensors to map the marine environment below and above the surface simultaneously. The method makes it possible to navigate in sheltered environments.

2.3 APF method

Khatib (1985) presented a real-time obstacle avoidance technique that uses an APF to guide the motion of a vehicle. The technique generates a repulsive potential field around obstacles and an attractive potential field at the destination. The sum of the negated potential field gradients yields a resultant virtual force that, at each position, guides the motion of the vehicle. This algorithm suffers from some drawbacks: (1) The presence of local minima; (2) It only provides a direction of motion and does not directly provide a collision-free path; (3) It was not developed for handling a dynamic environment.

Ge and Cui (2002) propose a more advanced APF method for motion planning of a mobile robot in a dynamic environment with moving obstacles and a target. This is done by considering both the velocity of the obstacles and the maximal deceleration of the robot. This requires that the positions and velocities must be known or measured online. The relative position of the robot and the target is used to define a function for the attractive field, whereas the relative positions of the robot and the obstacles are used to define a function for the repulsive field.

The speed of the APF algorithms and the potential for applying the method to higher dimensions make it a good alternative to graph searching techniques, even though the latter are more thorough techniques. The main drawback of the APF methods is that they suffer from the presence of local minima. Consider scenarios where the attractive and repulsive forces are equal or almost equal and collinear but in opposite directions. This would result in a trapped vehicle as seen in Figure 2.6a. Figure 2.6b displays another case where the target is too close to the obstacle so that the vehicle can not reach the target.

Li et al. (2012) resolve the local minima problem by proposing an improved APF method that defines the repulsive field about the vertices of polygonal obstacles and changes the direction of the repulsive field around these and circular obstacles.

(a) The obstacle located between the target and the vehicle creates a local minima.

(b) The obstacle located close to the target creates a local minima.

Figure 2.6: Situations where the equal or almost equal and collinear attractive and repulsive forces result in a local minima. Adapted from Li et al. (2012).

2.3.1 Navigation function

Koditschek and Rimon (1990) introduced a special APF method using navigation functions. The navigation function is used as a potential field to transform local minima, which are undesired, to saddle points with regions of attraction that measure zero. Rimon and Koditschek (1992) presented a way to navigate a point-mass robot in *generalized sphere worlds*. The simplest member of these worlds is a space made from a disk that is punctured with disjointed discs representing obstacles. The navigation function can be used on complicated geometric spaces by coordinate transformations into such sphere worlds.

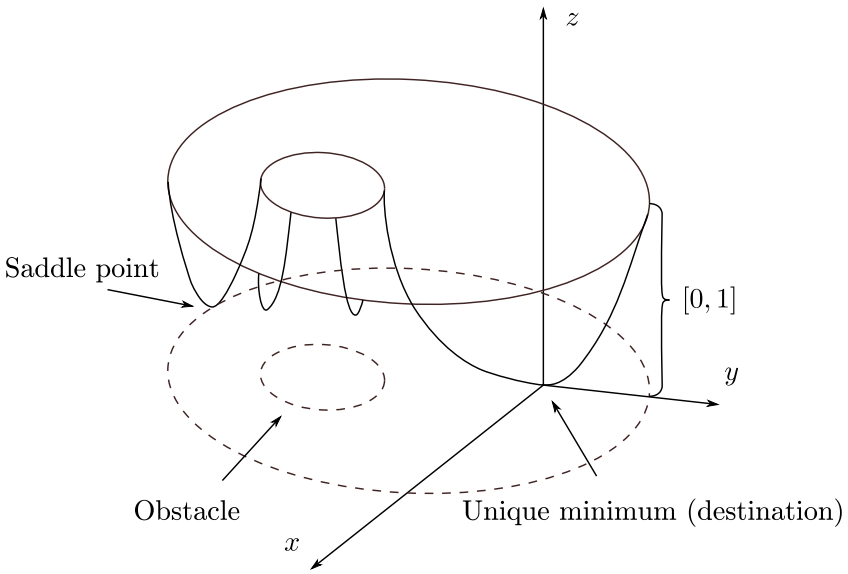


Figure 2.7: A navigation function constructed on a planar configuration space. Adapted from Rimon and Koditschek (1992).

A real-valued map by a navigation function is illustrated in Figure 2.7. This is constructed on a planar configuration space. As can be seen, the map has a unique minimum

at the destination and is uniformly maximal on the boundary of the configuration space.

Valbuena and Tanner (2012) use navigation functions to navigate non-holonomic mobile robots in planar environments with obstacles. This method allows for simultaneous convergence of position and orientation, and reduces the need for online switching of different control laws that would result in chattering. Figure 2.8 shows how the gradients guide the motion of the robot away from the boundaries of the workspace and the spherical obstacle, and toward the destination point which is the origin.

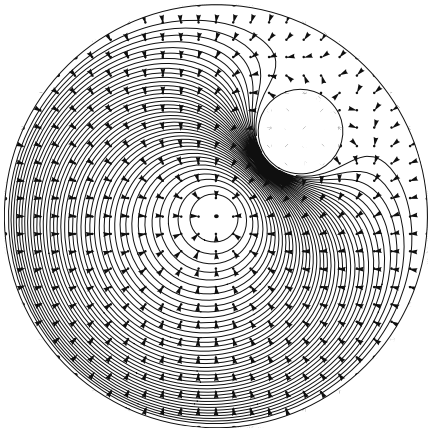


Figure 2.8: The contour plot of a sphere-world navigation function with the negated gradients that guide the robot to the destination. Courtesy of Valbuena and Tanner (2012).

2.4 Path generation based on WPs

2.4.1 Dubins path

Fossen (2011) presents different methods to generate paths based on WPs by using straight lines, circular arc segments, and interpolation. The simplest method uses straight lines to connect the WPs. Another method combines straight lines and circular arcs segments to make a shortest possible path for a craft with a constant speed between two configurations, a result that was established in Dubins (1957). This is a common way to represent the desired path because of its simplicity. A drawback of this path generation method is the jump that occurs in the desired yaw rate. This happens when the craft moves from the straight line where the yaw rate is zero to the circular arc segment where it is a constant. If a smooth reference trajectory is applied (by e.g., interpolation) one can overcome this problem. Figure 2.9 shows an example of a Dubins path constructed between 6 WPs.

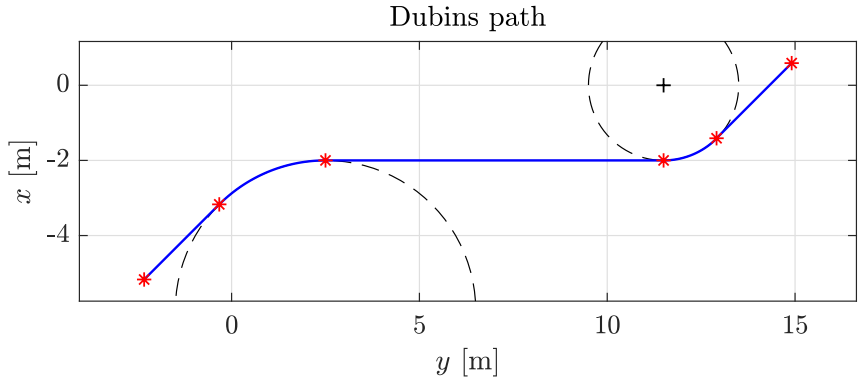


Figure 2.9: A Dubins path generated between 6 WPs.

2.4.2 Interpolation methods

Path generation using interpolation methods are another way of generating a curve through a set of predefined WPs (Fossen, 2011; Skjetne, 2005). This requires the path to be parametrized by a continuous path variable. A cubic spline interpolation is a common interpolation method that considers cubic polynomials to create a smooth spline. If compared to the Dubins path, the cubic interpolation strategy does not have a jump in the desired yaw rate along the path. Figure 2.10 displays an example of a cubic spline path generated using the MATLAB function `spline.m`.

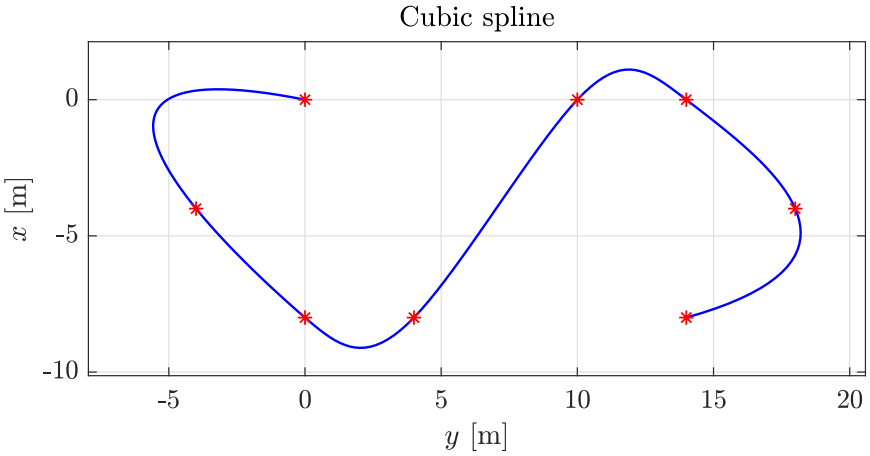


Figure 2.10: A cubic spline generated between 8 WPs.

Skjetne (2020*a*) proposes a path parametrization method that uses two path parameters and speed assignments. This is done to solve the problem of a ship that shall approach the quayside and perform final docking. The overall desired path is constructed by a nominal path along the quayside to the final WP and a normal vector path. The latter is activated when the ship has arrived at the final WP. The additional path parameter

and its speed assignment allow the ship to slowly crab sideways until it has reached the quayside.

2.5 Relevant motion control designs

2.5.1 Proportional-integral-derivative DP control

Fossen (2011) presents a simple maneuvering control design by using a proportional-integral-derivative (PID) control law, a model-free control. The control design model, which is a simplified version of the simulation model, can be used to compute constant gains for the PID controller. Conventional PID controllers were used in the first DP systems in the 1960s with lowpass and/or notch filters in cascade to remove wave-induced motion.

2.5.2 Backstepping DP control

More advanced control systems can be developed by using a dynamic model to generate feedback signals. Fossen (2011) presents backstepping as a design methodology to create such feedback signals. Backstepping constructs a feedback control law by a recursive construction of a control Lyapunov function (CLF).

Nonlinear backstepping design is strongly related to feedback linearization. The difference between the two design methods is that the feedback linearization methods cancel the nonlinearities that exist in the system. The backstepping methods have the possibility to exploit "good" nonlinearities and add nonlinear damping to dominate "bad" nonlinearities. Therefore, backstepping methods allow for a more flexible design that also has additional robustness compared to that of the feedback linearization methods. The latter also require precise models in order to cancel out nonlinear terms and these models can be difficult to obtain.

General maneuvering designs by backstepping and cascade-backstepping are presented in Skjetne et al. (2005); Skjetne (2020*b*). Cascade-backstepping uses cascaded systems theory to prove stability rather than recursively building up a CLF in the second step of the design.

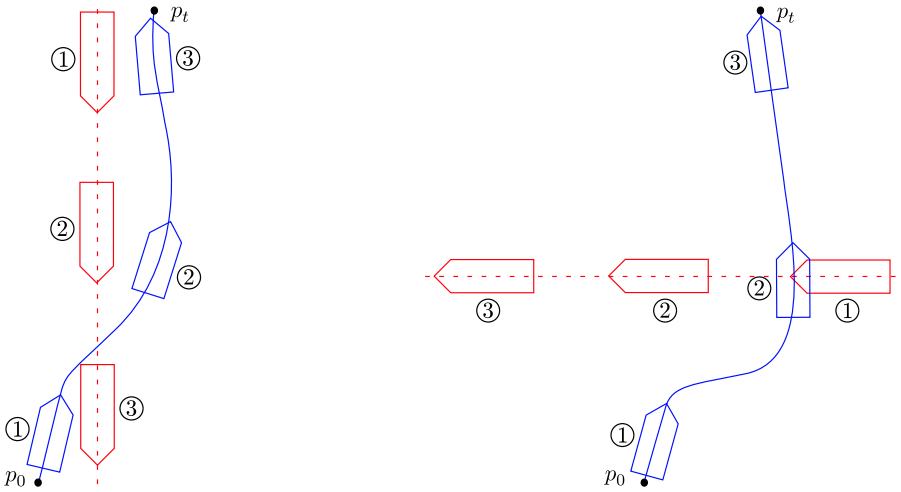
Chapter 3

Problem formulation

3.1 System description

This thesis considers the problem where an autonomous surface vehicle (ASV) is to maneuver in a harbor. The OS shall start at an initial position in the harbor, denoted by p_0, and maneuver to a position close enough to the quay to initiate the docking procedure, denoted by p_t.

The OS shall have a COLAV system such that it avoids TSs in the harbor in a safe and reactive way. The harbor is considered to be an urban waterway where recreational vessels (i.e., sailing and motor vessels, sea kayaks, etc.) make up most of the traffic. In the harbor, the vessel speed is restricted by law. Therefore, all of the moving vessels are considered to have a low speed.



(a) The OS encounters a TS in a head-on situation.

(b) The OS encounters a TS in a give-way situation.

Figure 3.1: The OS encounters a TS while going from p_0 to p_t in two possible scenarios in the harbor. The COLAV system creates a collision-free path for the OS to follow.

Figure 3.1 illustrates two possible scenarios where the OS, in blue, encounters a TS, in red, while going from p_0 to p_t in the harbor. When the OS encounters a TS, the OS's SA system tracks the TS, and the COLAV system shall predict the TS's motion and determine whether and when to take evasive action so that it can create a collision-free path to p_t.

Reactive maneuvering is either achieved by online planning and replanning of the path, or reactively and continuously constructing a path. Both methods are made possible by using information gathered from the surroundings by using the position reference and sensor systems. The GNSS provides the location of the OS and the static obstacles (e.g., land, docks, etc.), and a gyrocompass provides the heading of the OS. The SA system provides the location of the TSs. A sufficiently smooth path shall be generated by the OS's guidance system. The OS shall use a DP system to achieve path-following.

3.1.1 Assumptions and simplifications

The thesis is based on the following assumptions and simplifications:

- The vessel model parameters are known.
- The vessel is fully-actuated and has DP functionality as a basis.
- All moving vessels are low-speed. The dynamic TSs have a constant speed and course.
- The GNSS and the SA system provide the position of the OS and the TSs, respectively. The SA system measurements are perfect. A gyrocompass provides the heading of the OS.
- The guidance methods only consider COLREGs rules 8, 13-15, and 17 with main focus on developing compliance with rules 14 and 15.
- The second guidance method assumes only a single TS from p_0 to p_t.
- The only environmental disturbance is current.

3.1.2 Simulations

The simulations are run in MATLAB R2018B/Simulink.

3.2 Modeling

This section presents a complex 6 DOF simulation model for testing and a simplified 3 DOF control design model valid for low-speed vessels.

3.2.1 Simulation model

Two geographic reference frames are necessary to describe the motion of a marine craft (Fossen, 2011).

- The North-East-Down (NED) frame $\{n\}$ is defined as the tangent plane on the surface of the Earth, moving with the craft. The x-, y-, and z-axis in this frame point in the north, east, and down direction, respectively. Assuming that the craft can be approximated to operate with a constant longitude and latitude, $\{n\}$ is inertial and Newton's laws apply.
- The body frame $\{b\}$ is body-fixed and moves with the craft with the x-, y-, and z-axis pointing in the positive surge, sway, and heave direction, respectively. The linear and angular velocities are expressed in $\{b\}$.

Fossen (2011) presents a 6 DOF high fidelity model describing the ship dynamics as accurately as possible. Such a model is useful for simulation purposes and is usually referred to as a simulation model or a process plant model. This complex model is later used to simulate the response of the model ship. Assuming that the ocean currents are constant and irrotational, the equations of motion yield

$$\dot{\eta} = J_\Theta(\eta)\nu, \tag{3.1a}$$

$$M\dot{\nu}_r + C(\nu_r)\nu_r + D(\nu_r)\nu_r + g(\eta) = \tau + \tau_{\text{wind}} + \tau_{\text{waves}}, \tag{3.1b}$$

where

$$M = M_{RB} + M_A, \tag{3.2}$$

$$C(\nu_r) = C_{RB}(\nu_r) + C_A(\nu_r), \tag{3.3}$$

$$D(\nu_r) = D + D_n(\nu_r), \tag{3.4}$$

where M, $C(\nu_r)$, and $D(\nu_r)$ represent the system inertia matrix, Coriolis-centripetal matrix, and damping matrix, respectively. Let $\eta = [x, y, z, \phi, \theta, \psi]^\top \in \mathbb{R}^3 \times \mathbb{S}^3$ represent the vessel's position (x, y, z) and attitude (ϕ, θ, ψ) given in $\{n\}$, and $\nu = [v_u, v_v, w, p, q, r]^\top \in \mathbb{R}^6$ represent the corresponding linear velocities in surge, sway, and heave (v_u, v_v, w) and angular rates in roll, pitch, and yaw (p, q, r) given in $\{b\}$. Let $\nu_r = \nu - \nu_c$ be the relative velocity vector where ν_c denotes the ocean current velocity. Let $g(\eta)$ represent the restoring forces and τ represent the loads. The transformation matrix $J_\Theta(\eta)$ is given by

$$J_\Theta(\eta) = \begin{bmatrix} R_b^n(\Theta_{nb}) & 0_{3\times3} \\ 0_{3\times3} & T_\Theta(\Theta_{nb}) \end{bmatrix}, \tag{3.5}$$

where $R_b^n(\Theta_{nb})$ is given by

$$R_b^n(\Theta_{nb}) = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi c\phi s\theta \\ s\psi c\theta & c\psi c\phi + s\phi s\theta s\psi & -c\psi s\phi + s\theta s\psi c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix}, \tag{3.6}$$

and $T_\Theta(\Theta_{nb})$ is given by

$$T_\Theta(\Theta_{nb}) = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix}, \tag{3.7}$$

where $s\cdot = \sin(\cdot)$, $c\cdot = \cos(\cdot)$, and $t\cdot = \tan(\cdot)$.

3.2.2 Control design model

A simplified, low-fidelity model capturing the main physical properties of the vessel is sufficient when handling control design. Skjetne (2005) shows that for DP, the equations of motion of a ship can be simplified to a 3 DOF system valid for low-speed vessels. This simplification eliminates the Coriolis and nonlinear damping terms, while the dynamics related to the current and other unmodeled effects are represented by a slowly varying bias. In addition, it is assumed that heave, roll, and pitch dynamics can be neglected. The resulting model is given by

$$\dot{\eta} = R(\psi)\nu, \tag{3.8a}$$

$$M\dot{\nu} + D\nu = \tau + R(\psi)^\top b, \tag{3.8b}$$

where $\eta = [x, y, \psi]^\top \in \mathbb{R}^2 \times \mathbb{S}$ represents the vessel's position (x, y) and heading (ψ) given in $\{n\}$, and $\nu = [v^\top, r]^\top \in \mathbb{R}^2 \times \mathbb{R}$ with $v = [v_u, v_v]^\top$, is the body-fixed linear velocities in surge and sway, and angular rate in yaw (r) given in $\{b\}$. Let $\tau = [X, Y, N]^\top \in \mathbb{R}^3$ be the control forces and moment given in $\{b\}$, and $b = [b_1, b_2, b_3]^\top \in \mathbb{R}^3$ be the bias given in $\{n\}$. Transition between the two frames is performed through the rotation matrix $R(\psi)$ which is given by

$$R(\psi) = \begin{bmatrix} R_2(\psi) & 0_{2\times 1} \\ 0_{1\times 2} & 1 \end{bmatrix}, \quad R_2(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix}. \tag{3.9}$$

The skew-symmetric matrix S is given by

$$S(r) = \begin{bmatrix} S_2(r) & 0_{2\times 1} \\ 0_{1\times 2} & 0 \end{bmatrix}, \quad S_2(r) = \begin{bmatrix} 0 & -r \\ r & 0 \end{bmatrix}. \tag{3.10}$$

Some important rotation matrix properties are $R(\psi)^\top R(\psi) = I$ and $\dot{R}(\psi) = R(\psi)S(r)$. The vessel's inertia matrix M is given by

$$M = M_{RB} + M_A = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ 0 & mx_g - Y_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix} = M^\top \succ 0, \tag{3.11}$$

and the damping matrix D is given by

$$D = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & -Y_r \\ 0 & -N_v & -N_r \end{bmatrix} \succ 0, \tag{3.12}$$

where the parameters of M and D can be found in Table 8.3.

3.3 Problem statement

3.3.1 System overview

The information flow of the motion control system is illustrated in Figure 3.2. It is usual to divide this system into three subsystems denoted as the navigation, guidance, and control systems (Fossen, 2011).

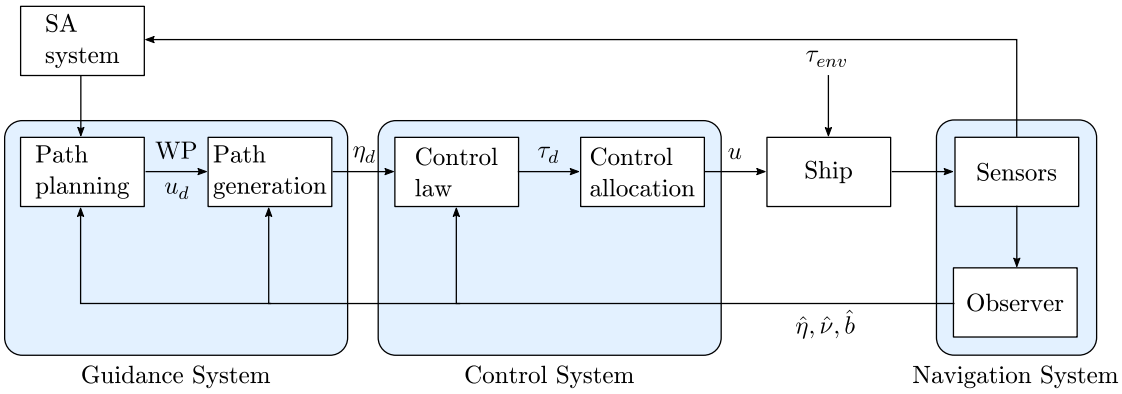


Figure 3.2: The information flow of the motion control system.

3.3.2 Navigation system

The navigation system sends information from the GNSS and the gyrocompass to an observer that creates the state estimates $(\hat{\eta}, \hat{\nu}, \hat{b})$. These state estimates are provided to the guidance and control systems. Information from the GNSS and SA sensors is sent to a SA system that fuses this data into a map of the surroundings and provides the location of the TSs to the guidance system.

Observer

The objective of the observer is to reduce the noise from the measurements and provide state estimates of the unmeasured states or during temporary loss of signals. It is assumed that only the position and heading measurements from the OS are available. Let the measured signal $y_m = [x_m, y_m, \psi_m]^\top \in \mathbb{R}^2 \times \mathbb{S}$ be given by

$$y_m = \eta + v, \tag{3.13}$$

where $v \in \mathbb{R}^3$ denotes a zero-mean Gaussian measurement noise vector. The observer shall provide an estimate of the position and heading that has removed the noise from the measurements such that

$$\lim_{t \to \infty} |\eta(t) - \hat{\eta}(t)| = 0. \tag{3.14}$$

In addition, the observer shall produce estimates of the unmeasured velocity and bias for feedback control.

3.3.3 Guidance system

In the guidance system, the path-planner calculates the WPs and determines the reference speed u_d. Both are provided to the path generator. The path generator constructs a feasible path and heading curve η_d for the vessel to follow.

Path-planning

The first guidance method shall path-plan by using information from online measurements to calculate WPs. The calculations should be done in a stepwise manner from p_0 to p_t such that a collision-free path can be constructed in accordance with COLREGs rules 14 and 15. Each WP is defined in a Cartesian coordinate system in a two-dimensional workspace, and given as $p_k = (x_k, y_k) \in \mathbb{R}^2$, for $k \in \{1, \ldots, n\}$, where n denotes the total number of WPs.

Online replanning is used to update the WP database so that collisions with moving TSs are avoided. In this stepwise setup, the next WP must be calculated by the path-planner algorithm at some point along the current path segment. Given $p_k = (x_k, y_k)$, the objective of the path-planner is to determine $p_{k+1} = (x_{k+1}, y_{k+1})$.

The path-planning module shall, for both methods, provide the reference speed in the current region. Keeping a low speed minimizes the risk of colliding in a busy harbor where the OS will operate close to TSs. In addition, any alteration of speed and/or course should, if possible, be large enough to be observable by TSs, meaning that a succession of small alterations should be avoided.

Path generation

The first guidance method shall generate a feasible path that connects the WPs from the path-planning module. Feasibility is ensured by making the path sufficiently smooth so that it is continuously differentiable at the connection points. That is, the path is required to be $\mathcal{C}^3$, meaning that the curve is 3 times differentiable. The objective is to generate the desired output η_d, parametrized by $s \in \mathbb{R}$, that is, $\eta_d : \mathbb{R} \to \mathbb{R}^2 \times \mathbb{S}$. This is given as

$$\eta_d(s) = \begin{bmatrix} p_d(s) \\ \psi_d(s) \end{bmatrix}, \quad p_d(s) = \begin{bmatrix} x_d(s) \\ y_d(s) \end{bmatrix}. \tag{3.15}$$

The second guidance method shall generate a feasible and collision-free path $p_d(s_1, s_2)$ by combining a nominal path $q_d(s_1)$ with a normal vector path $q_N(s_1, s_2)$ from p_0 to p_t. This overall path should be reactively and continuously constructed based on information from online measurements. Also, the path shall be in accordance with COLREGs rules 14 and 15. The objective is to generate the desired output η_d, consisting of $p_d : \mathbb{R}^2 \to \mathbb{R}^2$, parametrized by $s \in \mathbb{R}^2$, and heading reference $\psi_d : \mathbb{R}_{\geq 0} \times \mathbb{R}^2 \to \mathbb{S}$, that is, $\eta_d : \mathbb{R}^2 \to \mathbb{R}^2 \times \mathbb{S}$.

The desired heading curve shall be path-tangential for both of the guidance methods.

3.3.4 Control system

The control system determines the desired generalized control forces and moment τ_d that are necessary to follow the desired path. It distributes these loads as the control input u to the actuators of the ship.

Maneuvering control

The control problem can be stated as a maneuvering problem. Skjetne (2005) presents the maneuvering problem which comprises two tasks, in prioritized order:

1. The geometric task: Force the output η to converge to the desired output $\eta_d(s)$ with some continuous function s(t), i.e.,

$$\lim_{t\to\infty} |\eta(t) - \eta_d(s(t))| = 0. \tag{3.16}$$

2. The dynamic task: Force $\dot{s}$ to converge to a desired speed assignment $\upsilon(t, s)$, i.e.,

$$\lim_{t\to\infty} |\dot{s}(t) - \upsilon(t, s(t))| = 0. \tag{3.17}$$

The maneuvering control objective is then to design a control law such that

$$\left.\begin{array}{l} \eta(t) \to \eta_d(s(t)) \\ \dot{s}(t) \to \upsilon(t, s(t)) \end{array}\right\} \text{as } t \to \infty. \tag{3.18}$$

Chapter 4

Navigation

4.1 Observer

A nonlinear passive observer from Fossen (2011) is introduced in order to reduce measurement noise and provide state estimates of unmeasured states or during loss of measurement signals, dead reckoning. To prove passivity, the following assumption is needed:

- Assume that the zero-mean Gaussian white noise terms w and v are zero such that they are omitted in the analysis of the observer. In this way, the error dynamics will be uniformly globally asymptotically/exponentially stable (UGAS/UGES) instead of uniformly ultimated bounded in the Lyapunov function analysis.

Given the assumption above and the low-speed vessel model in (3.8), the following observer equations can be chosen:

$$\dot{\hat{\eta}} = R(\psi)\hat{\nu} + L_1\tilde{\eta}, \tag{4.1}$$

$$M\dot{\hat{\nu}} = -D\hat{\nu} + R(\psi)^\top \hat{b} + R(\psi)^\top L_2\tilde{\eta} + \tau, \tag{4.2}$$

$$\dot{\hat{b}} = L_3\tilde{\eta}, \tag{4.3}$$

where $(\hat{\eta}, \hat{\nu}, \hat{b})$ are the state estimates, and the state errors are defined by $\tilde{\eta} := \eta - \hat{\eta}$ and $\tilde{\nu} := \nu - \hat{\nu}$. The injection gain matrices $L_{1,2,3} \in \mathbb{R}^{3\times 3}$ can be tuned using rules from Fossen (2011).

Chapter 5

Guidance with navigation function

This chapter presents how a discrete topologically organized APF with a navigation function is used as a path-planner to guide the OS from p_0 to p_t while avoiding TSs. In addition, a path generation method that connects the WPs with a feasible path is presented.

5.1 Path-planning

5.1.1 Creating the dynamic navigation function

The OS is to operate on a two-dimensional Euclidean space in the harbor. Therefore, it is necessary to show how to construct a two-dimensional Euclidean sphere world to be able to apply the formula for the navigation function. The two-dimensional Euclidean sphere world can be described as a disk with smaller disjointed disk-like punctures that represent obstacles.

Rimon and Koditschek (1992) presented a suitable coordinate transformation to transform any star-shaped space to a sphere world. Figure 5.1 shows two sets that are topological discs, where $\mathcal{S}$ is a general shape that is termed a "star shape". A star-shaped set like $\mathcal{S}$ includes all convex sets and is characterized by having a center point $q_i \in \mathcal{S}$ from which every ray crosses the boundary of the set only once. The other set $\mathcal{D}$ is a Euclidean two-dimensional disc with a center point p_i and radius ρ. The star-to-disc transformation maps the boundary of the star diffeomorphically onto the boundary of the disc. It also maps the star's interior and exterior to the disc's interior and exterior. By performing this mapping, the navigation function can be used on complicated geometric spaces.

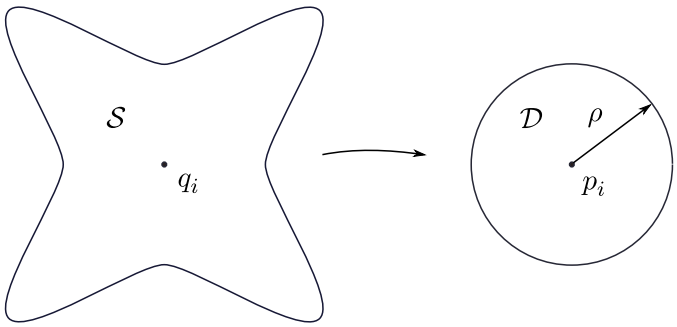


Figure 5.1: A star-shaped set is transformed to a disk by a change of coordinates. Adapted from Rimon and Koditschek (1992).

Valbuena and Tanner (2012) show how the general formula for a navigation function created by Rimon and Koditschek (1992) can be applied to a two-dimensional sphere world. Let $p = [x, y]^\top$ be the position of the OS and $\mathcal{W}$ be the planar workspace of the OS defined as a closed ball in $\mathbb{R}^2$. The origin is contained in $\mathcal{W}$ and is chosen as p_t. It is assumed that $p \in \mathcal{W}$ and that the TSs are represented by spheres, $\mathcal{O}_i \in \mathcal{W}$, for $i \in \{1, \ldots, m\}$, where m denotes the total number of TSs. The latter can be assumed as any star shape can be diffeomorphically transformed to a sphere. It is also assumed that $\mathcal{O}_i \cap \mathcal{O}_j = \emptyset$, for every $i \neq j$, with $i, j \in \{0, \ldots, m\}$. The workspace boundary is represented by $\mathcal{O}_0 = \mathbb{R}^2 \setminus \mathcal{W}$.

Let $\beta_0 = r_0^2 - |p|^2$, where r_0 denotes the radius of the workspace. Let $\beta_i = |p - p_i|^2 - r_i^2$, where r_i and p_i denote the radius and center, respectively, of $\mathcal{O}_i$ for $i \in \{1, \ldots, m\}$. Also let the distance to the destination be defined as $\gamma := |p|^2$, and κ be a sufficiently large parameter so that $\phi_{nf}(p) : \mathcal{W} \setminus \cup_{i=1}^m \mathcal{O}_i \to [0, 1]$ is a navigation function defined as

$$\phi_{nf}(p) := \frac{\gamma}{[\gamma^\kappa + \prod_{i=0}^m \beta_i(p)]^{\frac{1}{\kappa}}}. \tag{5.1}$$

When the parameter κ exceeds a certain sufficiently large value, it can be shown that all the undesired local minima disappear (Koditschek and Rimon, 1990).

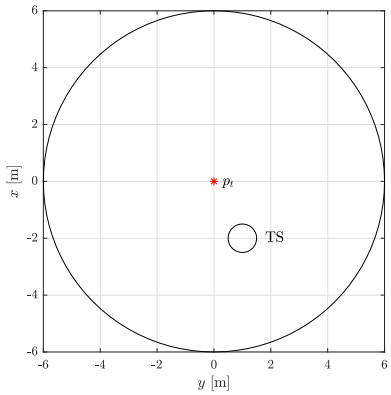
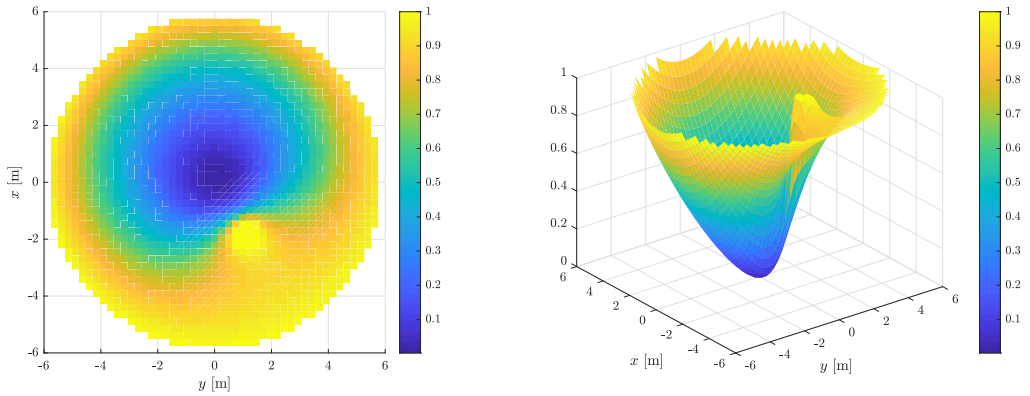


Figure 5.2: An environment containing a TS and the destination.

Figure 5.2 shows an environment with $r_0 = 6$ m that contains a TS with $r_1 = 0.5$ m at $p_1 = [-2, 1]^\top$ and the destination at the origin.

Figure 5.3 displays how the environment in Figure 5.2 can be translated to a topologically organized landscape representation using a navigation function with $\kappa = 2$. As can be seen from figures 5.3a and 5.3b, the navigation function varies between zero at p_t, which gives a unique minimum, and unity on the boundaries of the obstacle and workspace. This is as expected when examining (5.1). From Figure 5.3b, a single saddle point can be observed near the internal obstacle. This is the lowest number that can be achieved considering there are at least m saddle points when using navigation functions subject to sphere world's topology (Rimon and Koditschek, 1992).



(a) A two-dimensional view. (b) A three-dimensional view.

Figure 5.3: The environment in Figure 5.2 is translated to a topologically organized landscape representation using a navigation function.

5.1.2 Workspace partitioning

Using the APF method in a path-planner requires a partitioning of the workspace in order to limit the computational cost of determining the next WP. There are multiple ways to partition the two-dimensional workspace. Meng and Yang (1998) used a grid partitioning method with square cells, whereas Scibilia et al. (2012) partition the workspace using circular cells. The former is used as a decomposition of the workspace. In the discrete partitioning, it is necessary to have a resolution high enough to achieve reactive maneuvering. Before determining this resolution, several factors related to cell size need to be considered.

Reducing the cell size would increase the APF algorithm's computational time significantly. Another drawback of the cell-composition path-planning is that the position of the vehicle cannot be uniquely defined if the cell size is smaller than the vehicle (Šeda, 2007). On the other hand, too large cell size would produce inefficient routes with unnecessary large safety margins and lead to more cases where no solution is found. To achieve reactive maneuvering, a cell size L_{cell} equaling the ship length L_{oa} was deemed necessary.

5.1.3 Calculating the next WP

The next WP is to be calculated by the path-planner algorithm at some point along the current path segment. In this way, the positions of the TSs can be updated and the dynamic guidance model can be evolved. Therefore, a method is needed to know when to calculate the next WP to ensure that there is enough time to replan and compute the next segment. Fossen (2011) presents an updating method using a circle of acceptance and proposes a guideline for it. Let the radius of a circle be R and the next WP be $p_n = (x_n, y_n)$. The next WP should be calculated if the vessel in position $p(t)$ satisfies

$$[x_n - x(t)]^2 + [y_n - y(t)]^2 \leq R^2. \tag{5.2}$$

The guideline proposed is to let R be equal to two ship lengths, that is $R = 2L_{oa}$. Considering that the OS needs to be reactive to the varying environment, $R = L_{oa}$ was chosen.

The proposed method for determining the next WP is to move the vehicle along the direction where the discrete potential field value decreases the most. This must be limited to a defined neighborhood of the vehicle and should be repeated until the OS is deemed close enough to p_t. Given that the OS occupies the current cell position p_q, the next position p_n, which corresponds to a navigation function value $\phi_{nf}(p_n)$, is given by

$$p_n \Leftarrow \phi_{nf}(p_n) = \min\{\phi_{nf}(p_j) | \, p_j \in neig_{d_0}(p_q)\}, \tag{5.3}$$

where $\phi_{nf}(p_j)$ corresponds to a neighboring cell position p_j. The neighboring set of cells to a cell position p_q is defined as

$$neig_{d_0}(p_q) := \{p_j | \; |p_q - p_j| < d_0\}, \tag{5.4}$$

where d_0 is the radius of the neighborhood required to be: (1) Larger or equal to L_{cell}; (2) Small enough to be able to have an acceptable computational cost.

The path-planner can be evolved to take heading changes into account by adopting the same approach as in Scibilia et al. (2012). Given the OS's cell position p_q and heading angle ψ_q from the line between p_q and the previous cell position, the next position is given by

$$p_n \Leftarrow \phi_{nf}(p_n) = \arg \min_{\phi_{nf}(p_j): \, p_j \in neig_{d_0}(p_q)} \left\{ \left(1 + \lambda_\psi \frac{\text{diff}(\psi_q, \psi_j)}{\pi} \right) \phi_{nf}(p_j) \right\}. \tag{5.5}$$

The angle ψ_j is the heading angle from the line between p_q and p_j. The operator diff returns the smallest angle difference between two angles, that is, $\text{diff} : [0, 2\pi) \times [0, 2\pi) \rightarrow [0, \pi]$. The non-negative parameter λ_ψ is a weighting factor determining the heading change. Also, let the neighboring set of cells be limited by too large heading changes such that $\text{diff}(\psi_q, \psi_j) < \Delta_\psi$ is required, where Δ_ψ denotes the limit on heading changes.

5.1.4 COLREGs compliance

In its current form, the navigation function path-planner is already partly in compliance with COLREGs rules 8, 13, and 17. The path-planner complies with parts of rule 8

as it provides large enough alteration of course to be noticeable by TSs when choosing adequate values for L_{cell} and d_0. Even more compliance with this rule can be added by ensuring crash stop functionality. In the situation where the OS and the TS are deemed too close to each other and in danger of colliding, the path-planner can evolved to perform a crash stop.

Let the Euclidean distance between the OS and the TS be $d_E(t) = |p(t) - p_{TS}(t)|$, where $p_{TS}(t)$ denotes the position of the TS. Let the relative bearing of the TS to the OS be

$$\phi(t) = \text{atan2}(y_{TS}(t) - y(t), x_{TS}(t) - x(t)) - \psi(t), \tag{5.6}$$

where the function $\text{atan2}(y, x) \in [-\pi, \pi]$ is the four-quadrant version of $\arctan(y, x)$. Also let δ be the shortest allowable distance between the OS and the TS. The categorization from Tam and Bucknall (2010) is used to determine the position of the TS. If the TS is in a crossing or head-on region and too close, meaning that $\phi(t) \in [-112.5°, 112.5°]$ and $d_E(t) < \delta$, a crash stop is performed by setting $u_d(t) = 0$.

The path-planner already complies to some extent with rules 13 and 17 because in an overtaking scenario the OS would either pass the TS on its port or starboard side. For a stand-on scenario, the OS would take evasive action if the TS is not keeping out of the way.

More COLREGs compliance can be added by modifying the APF. Naeem et al. (2016) adapt an APF method to comply with COLREGs rules 14 and 15 by adding virtual objects to the APF that act as obstacles. These virtual objects need to be positioned close to the TS to encourage a COLREGs compliant "flow" around the TS. A similar step is taken to evolve the navigation function.

First, the encounter type needs to be determined as soon as the TS is located. Using the proposed method from Tam and Bucknall (2010) requires the relative bearing from (5.6) and the relative course $\chi_{rel} = \chi_{TS} - \chi$ between the vessels. Assuming that the COLREGs scenario is classified as either head-on or give-way, the following steps are taken:

- If the OS is in a head-on scenario, a row of n_{HO} virtual obstacles with a radius r_{HO} are added outward from the TS, in the direction 22.5° offset from χ_{TS}. The virtual obstacles are placed as close as possible while still being isolated, thus satisfying $\mathcal{O}_i \cap \mathcal{O}_j = \emptyset$, for every $i \neq j$, with $i, j \in \{0, \ldots, m\}$. This will enforce the OS to change its course to starboard so that the TS passes on the OS's port side.
- If the OS is in a give-way scenario, a row of virtual obstacles extending the entire workspace are added outward from the TS, in the same direction as χ_{TS}. The obstacles have a radius r_{GW} and are placed adjacently in the same manner as in the head-on scenario. This will enforce the OS to pass behind the TS.

A flaw of this approach is that as noted in Section 5.1.1, there will be at least one new saddle point in the navigation function for each virtual obstacle added. Also, as the give-way step adds virtual obstacles that extend the entire workspace, the OS can get trapped if there are multiple TSs that the OS is the give-way vessel to.

5.2 Path generation

The generation of the path is done by selecting WPs that satisfy the spatial constraints and constructing a path between the WPs. The vessel's dynamic constraints and the desired continuity between WPs need to be taken into consideration when generating the path. The path needs to be sufficiently differentiable allowing the path derivatives to be continuous at the connection points.

5.2.1 $\mathcal{C}^r$ path generation

Skjetne (2005) shows how a sufficiently differentiable curve $\mathcal{C}^r$ connecting the WPs using splines and interpolations techniques can be specified. To generate the overall desired path $p_d(s)$, it is split into n subpaths $p_{d,i}(s)$, $i = 1, \ldots, n$, that are concatenated at the WPs. Each subpath is expressed by a polynomial in s of a specific order to ensure that the path is sufficiently differentiable. Let $\mathcal{I} = \{1, 2, \ldots, n\}$ be indices for each subpath. The desired curve is $p_d(s) = [x_d(s), y_d(s)]^\top$, where $s \in [0, n]$, while $p_{d,i}(s) = [x_{d,i}(s), y_{d,i}(s)]^\top$, $i \in \mathcal{I}$, are subpaths, and $p_i = (x_i, y_i)$, $i \in \mathcal{I} \cup \{n+1\}$, are the WPs. Generating a $\mathcal{C}^r$ path $p_d(s)$ requires the following to hold between two subpaths:

$$
\begin{aligned}
\lim_{s \nearrow i-1} x_{d,i-1}(s) &= \lim_{s \searrow i-1} x_{d,i}(s), & \lim_{s \nearrow i-1} y_{d,i-1}(s) &= \lim_{s \searrow i-1} y_{d,i}(s), \\
\lim_{s \nearrow i-1} x^s_{d,i-1}(s) &= \lim_{s \searrow i-1} x^s_{d,i}(s), & \lim_{s \nearrow i-1} y^s_{d,i-1}(s) &= \lim_{s \searrow i-1} y^s_{d,i}(s), \\
&\vdots & &\vdots \\
\lim_{s \nearrow i-1} x^{s^r}_{d,i-1}(s) &= \lim_{s \searrow i-1} x^{s^r}_{d,i}(s), & \lim_{s \nearrow i-1} y^{s^r}_{d,i-1}(s) &= \lim_{s \searrow i-1} y^{s^r}_{d,i}(s),
\end{aligned}
$$

for $i \in \mathcal{I} \setminus \{1\}$. Polynomials of order k are considered, this yields

$$x_{d,i}(s) = a_{k,i}s^k + \ldots + a_{1,i}s + a_{0,i}, \tag{5.7}$$

$$y_{d,i}(s) = b_{k,i}s^k + \ldots + b_{1,i}s + b_{0,i}, \tag{5.8}$$

where the coefficients in (5.7) and (5.8) need to be determined. As there are $2(k+1)$ unknowns for each subpath, this yields a total of $2n(k+1)$ unknown coefficients that need to be determined to construct the full path.

Stepwise hybrid path parametrization

Hybrid path parametrization is a method for calculating the coefficients in the polynomials in (5.7) and (5.8). Skjetne (2019) proposes a strategy for applying this method in a stepwise setup where it is assumed that only the current WP $p_{0,i}$ and the target WP $p_{t,i}$ are known. The method consists of continuously parametrizing each subpath i within $\varepsilon = s - \lfloor s \rfloor \in [0, 1)$, where $i = \lfloor s \rfloor + 1 \in \mathcal{I}$.

Assuming that the subpath i connects $p_{0,i}$ and $p_{t,i}$ gives the following equations for calculating the coefficients:

$\mathcal{C}^0$: Continuity at $p_{0,i}$ and $p_{t,i}$ gives for subpath i:

$$\begin{aligned} x_{d,i}(0) &= x_{0,i}, & x_{d,i}(1) &= x_{t,i}, \\ y_{d,i}(0) &= y_{0,i}, & y_{d,i}(1) &= y_{t,i}. \end{aligned} \tag{5.9}$$

$\mathcal{C}^1$: The slopes at $p_{0,i}$ and $p_{t,i}$ are set as:

$$\begin{aligned} x^{\varepsilon}_{d,i}(0) &= T_{0x,i}, & x^{\varepsilon}_{d,i}(1) &= \lambda \frac{x_{t,i} - x_{0,i}}{|p_{t,i} - p_{0,i}|}, \\ y^{\varepsilon}_{d,i}(0) &= T_{0y,i}, & y^{\varepsilon}_{d,i}(1) &= \lambda \frac{y_{t,i} - y_{0,i}}{|p_{t,i} - p_{0,i}|}, \end{aligned} \tag{5.10}$$

where $T_{0,i} \in \mathbb{R}^2$ is the unit-tangent vector at $p_{0,i}$ and $\lambda > 0$ is a design parameter.

$\mathcal{C}^j$: Setting the derivatives of order $j \geq 2$ to zero gives for subpath i:

$$\begin{aligned} x^{\varepsilon^j}_{d,i}(0) &= 0, & x^{\varepsilon^j}_{d,i}(1) &= 0, \\ y^{\varepsilon^j}_{d,i}(0) &= 0, & y^{\varepsilon^j}_{d,i}(1) &= 0. \end{aligned} \tag{5.11}$$

It can be shown that (5.9) to (5.11) up to $j = r$ give $4n(r+1)$ equations to solve for $2n(k+1)$ unknowns. Then, the order of the polynomials is given by

$$k = 2r + 1. \tag{5.12}$$

Since the path tangent vector is needed to determine the desired heading, the path needs to be $\mathcal{C}^3$ to ensure bumpless transfer between the WPs. This requires the polynomials in ε to be of order $k = 7$. Figure 5.4 displays an example of a $\mathcal{C}^3$ path generated using a stepwise hybrid path parametrization with $\lambda = 2$.

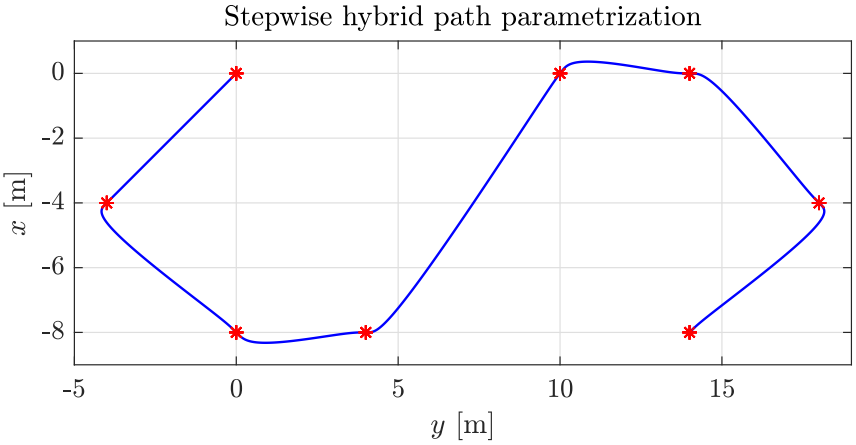


Figure 5.4: A path generated between 8 WPs using a stepwise hybrid path parametrization with $\lambda = 2$.

Given $p_{0,i}$, $p_{t,i}$, and $u_d(t)$ from the path-planner, the path generator function is to generate the desired path, heading, speed assignments, and relevant derivatives.

The desired path and its derivatives up to the second order are

$$\bar{p}_d(i,\varepsilon) = \begin{bmatrix} \bar{x}_d(i,\varepsilon) \\ \bar{y}_d(i,\varepsilon) \end{bmatrix}, \quad \bar{p}_d^{\varepsilon}(i,\varepsilon) = \begin{bmatrix} \bar{x}_d^{\varepsilon}(i,\varepsilon) \\ \bar{y}_d^{\varepsilon}(i,\varepsilon) \end{bmatrix}, \quad \bar{p}_d^{\varepsilon^2}(i,\varepsilon) = \begin{bmatrix} \bar{x}_d^{\varepsilon^2}(i,\varepsilon) \\ \bar{y}_d^{\varepsilon^2}(i,\varepsilon) \end{bmatrix}. \tag{5.13}$$

The desired heading needs to be path-tangential, this yields

$$\bar{\psi}_d(i,\varepsilon) = \operatorname{atan2}\left(\bar{y}_d^{\varepsilon}(i,\varepsilon), \bar{x}_d^{\varepsilon}(i,\varepsilon)\right). \tag{5.14}$$

The heading derivatives up to the second order are

$$\bar{\psi}_d^{\varepsilon}(i,\varepsilon) = \frac{\bar{x}_d^{\varepsilon}(i,\varepsilon)\bar{y}_d^{\varepsilon^2}(i,\varepsilon) - \bar{x}_d^{\varepsilon^2}(i,\varepsilon)\bar{y}_d^{\varepsilon}(i,\varepsilon)}{\bar{x}_d^{\varepsilon}(i,\varepsilon)^2 + \bar{y}_d^{\varepsilon}(i,\varepsilon)^2}, \tag{5.15}$$

$$\begin{aligned} \bar{\psi}_d^{\varepsilon^2}(i,\varepsilon) = {} & \frac{\bar{x}_d^{\varepsilon}(i,\varepsilon)\bar{y}_d^{\varepsilon^3}(i,\varepsilon) - \bar{x}_d^{\varepsilon^3}(i,\varepsilon)\bar{y}_d^{\varepsilon}(i,\varepsilon)}{\bar{x}_d^{\varepsilon}(i,\varepsilon)^2 + \bar{y}_d^{\varepsilon}(i,\varepsilon)^2} \\ & - 2\frac{\left(\bar{x}_d^{\varepsilon}(i,\varepsilon)\bar{y}_d^{\varepsilon^2}(i,\varepsilon) - \bar{x}_d^{\varepsilon^2}(i,\varepsilon)\bar{y}_d^{\varepsilon}(i,\varepsilon)\right)\left(\bar{x}_d^{\varepsilon}(i,\varepsilon)\bar{x}_d^{\varepsilon^2}(i,\varepsilon) + \bar{y}_d^{\varepsilon}(i,\varepsilon)\bar{y}_d^{\varepsilon^2}(i,\varepsilon)\right)}{\left(\bar{x}_d^{\varepsilon}(i,\varepsilon)^2 + \bar{y}_d^{\varepsilon}(i,\varepsilon)^2\right)^2}. \end{aligned} \tag{5.16}$$

Let the speed assignment be defined as

$$\dot{s} := \upsilon_i(t,\varepsilon) = \frac{u_d(t)}{|\bar{p}_d^{\varepsilon}(i,\varepsilon)|}\rho(i,\varepsilon), \tag{5.17}$$

where

$$\rho(i,\varepsilon) = \tanh\left(\frac{n-(i-1+\varepsilon)}{\Delta_{p,1}}\right), \tag{5.18}$$

is a function that makes the speed go to zero at the destination and $\Delta_{p,1}$ is a gain to set the slope at $s = n$. This yields the derivatives

$$\upsilon_i^t(t,\varepsilon) = \frac{u_d^t(t)}{|\bar{p}_d^{\varepsilon}(i,\varepsilon)|}\rho(i,\varepsilon), \tag{5.19}$$

$$\upsilon_i^{\varepsilon}(t,\varepsilon) = -\frac{\bar{p}_d^{\varepsilon}(i,\varepsilon)^{\top}\bar{p}_d^{\varepsilon^2}(i,\varepsilon)}{|\bar{p}_d^{\varepsilon}(i,\varepsilon)|^3}u_d(t)\rho(i,\varepsilon) - \frac{u_d(t)}{|\bar{p}_d^{\varepsilon}(i,\varepsilon)|\Delta_{p,1}}\operatorname{sech}^2\left(\frac{n-(i-1+\varepsilon)}{\Delta_{p,1}}\right). \tag{5.20}$$

Combining the references yields the following guidance system:

$$\dot{s} = \upsilon(t,s) := \upsilon_i(t,\varepsilon) = \frac{u_d(t)}{|\bar{p}_d^{\varepsilon}(i,\varepsilon)|}\rho(i,\varepsilon), \tag{5.21}$$

$$\psi \to \psi_d(s) := \bar{\psi}_d(i,\varepsilon) = \operatorname{atan2}\left(\bar{y}_d^{\varepsilon}(i,\varepsilon), \bar{x}_d^{\varepsilon}(i,\varepsilon)\right), \tag{5.22}$$

$$p \to p_d(s) := \bar{p}_d(i,\varepsilon). \tag{5.23}$$

Chapter 6

Guidance with two path parameters

This chapter presents a guidance method that combines the nominal path with a normal vector path by using two path parameters and speed assignments. This is done to reactively and continuously generate a collision-free path that brings the vessel to its docking location.

6.1 Path-following a nominal path

Let a nominal path $q_d(s_1) = [x_{q_d}(s_1), y_{q_d}(s_1)]^\top \in \mathbb{R}^2$ for the OS to follow be continuously parametrized by a path parameter $s_1 \in [0, 1]$ from p_0 to p_t. Assigning the same speed assignment law on $\dot{s}_1$ as in Skjetne (2020*a*) yields

$$\dot{s}_1 = v_1(t, s_1) := \frac{u_d(t)}{|q_d^{s_1}(s_1)|}\rho(s_1), \tag{6.1}$$

where $u_d(t)$ is the desired speed along the nominal path and

$$\rho(s_1) = \tanh\left(\frac{1 - s_1}{\Delta_{p,1}}\right), \tag{6.2}$$

is a function that makes the speed go to zero at p_t. The gain $\Delta_{p,1}$ sets the slope at $s_1 = 1$. The desired heading along the nominal path is

$$\psi_{q_d}(s_1) = \text{atan2}\left(y_{q_d}(s_1), x_{q_d}(s_1)\right). \tag{6.3}$$

6.2 Constructing a combined path

The workspace of the OS can be parametrized by a path-parallel reference frame $\{PP\}$ that follows the desired position along $q_d(s_1)$. Let a combined path $p_d(s_1, s_2)$ be set as a point that can be offset normally by adding an additional path parameter s_2 to the path in $\{PP\}$. Also let $s = \text{col}(s_1, s_2)$. The path parameter s_2 needs to be updated in an appropriate way to create a collision-free path from p_0 to p_t. The combined path yields

$$p_d(s_1, s_2) = q_d(s_1) + q_N(s_1, s_2), \tag{6.4}$$

where the normal vector path $q_N(s_1, s_2)$ is given by

$$q_N(s_1, s_2) = s_2 N_d(s_1), \tag{6.5}$$

where $N_d(s_1)$ is the unit normal vector that is derived from the unit tangent vector $T_d(s_1)$ along the nominal path. Skjetne (2020*a*) establishes the latter as

$$T_d(s_1) = \frac{q_d^{s_1}(s_1)}{|q_d^{s_1}(s_1)|}, \tag{6.6}$$

giving the unit normal vector

$$N_d(s_1) = JT_d(s_1) = J\frac{q_d^{s_1}(s_1)}{|q_d^{s_1}(s_1)|}, \tag{6.7}$$

where the skew-symmetric matrix J is chosen as

$$J = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} = -J^\top, \tag{6.8}$$

so that $N_d(s_1)$ points in a positive sway direction for a ship moving along the nominal path with a path-tangential heading. This yields

$$T_d(s_1)^\top N_d(s_1) = 0. \tag{6.9}$$

Figure 6.1 illustrates the combined path.

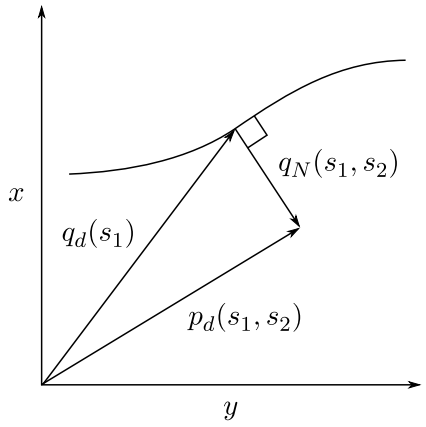


Figure 6.1: The construction of the combined path.

6.2.1 Normal vector path

Reference signal

The proposed method is to construct a reference signal d_{ref} for s_2 to follow so that a collision-free path $p_d(s_1, s_2)$ can be created. Since $N_d(s_1)$ has unit length, s_2 has unit [m]. Assuming that there is only a single TS between p_0 and p_t, d_{ref} needs to have the following properties:

- It needs to be smooth to avoid jumps in the path-tangential heading reference.
- It needs to be sufficiently large (a safe passing distance to the TS) when $q_d(s_1)$ is close to the TS's position.
- It needs to be approximately zero when the OS is close to p_0 and p_t such that $p_d(0, s_2) \approx q_d(0)$ and $p_d(1, s_2) \approx q_d(1)$, respectively.

A suitable smooth reference signal is the derivative of the hyperbolic function $\tanh(x)$ given by

$$\frac{d}{dx}\tanh(x) = 1 - \tanh^2(x). \tag{6.10}$$

Note that (6.10) equals 1 for $x = 0$ and converges to 0 for large absolute values of x.

Let the position of the TS be given by $p_{TS}(t) = [x_{TS}(t), y_{TS}(t)]^\top$. It is assumed that the TS has a constant speed and course. Let d_{ref} depend on the scalar projection d_{proj} of the vector $[p_{TS}(t) - q_d(s_1)]$ on the vector $[p_t - q_d(s_1)]$. This yields

$$d_{proj}(t, s_1) = \frac{[p_t - q_d(s_1)]^\top [p_{TS}(t) - q_d(s_1)]}{|p_t - q_d(s_1)|}. \tag{6.11}$$

The scalar projection becomes negative when the OS has passed the TS. Let the reference signal be

$$d_{ref}(t, s_1) = d_{safe}\left[1 - \tanh^2\left(\frac{d_{proj}(t, s_1)}{\Delta_{p,2}}\right)\right], \tag{6.12}$$

where $d_{safe} > 0$ is a large enough value so that the OS has a safe passing distance to the TS, and $\Delta_{p,2}$ is a gain that determines how fast (6.12) converges to 0.

Speed assignment

A suitable speed assignment law on $\dot{s}_2$ needs to be designed so that the OS will follow $d_{ref}(t, s_1)$. Let d denote an approximation of the offset from $q_d(s_1)$ to the OS's position $p(t)$ and be given by

$$d(t, s_1) = \tanh\Big(N_d(s_1)^\top [p(t) - q_d(s_1)]\Big)|p(t) - q_d(s_1)|, \tag{6.13}$$

where the function tanh determines whether the OS is on the positive or negative side of $q_d(s_1)$ compared to the direction of $N_d(s_1)$. The function also avoids signal chatter when the OS is close to the nominal path. Figure 6.2 illustrates the parameters from (6.11) to (6.13) in an example with a straight-line nominal path.

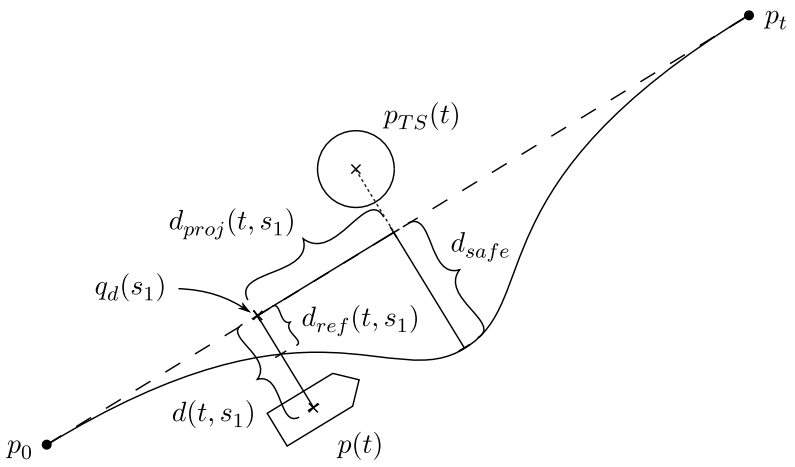


Figure 6.2: The parameters that are used to create the speed assignment for $\dot{s}_2$.

Let $\tilde{d} = d - d_{ref}$ denote the difference in the offset and the reference signal so that a proposed speed assignment law on $\dot{s}_2$ is

$$\dot{s}_2 = \upsilon_2(t, s_1) = u_N \tanh\left(\frac{-\tilde{d}(t, s_1)}{\Delta_{p,3}}\right), \tag{6.14}$$

where u_N is the desired speed along the normal vector path, and $\Delta_{p,3}$ is a gain to set the slope at $\tilde{d} = 0$. A similar speed assignment is used in Skjetne (2020*a*) to slowly take the normal vector path to a quayside. The speed assignment in (6.14) ensures that $\tilde{d} > 0$ gives a negative speed (compared to the direction of $N_d(s_1)$) toward d_{ref}, while $\tilde{d} < 0$ gives a positive speed toward d_{ref}. The speed is zero for $\tilde{d} = 0$. This implies an asymptotic stability property of s_2 at the position where $\tilde{d} = 0$.

By using (6.14), the OS will come to a stop a distance away from the nominal path as d_{ref} is not exactly zero at $s_1 = 1$. However, this speed assignment can be modified by adding the activation function $\sigma_\upsilon(s_1) \in \{0, 1\}$ which will change the speed assignment close to p_t. The final speed assignment law on $\dot{s}_2$ is

$$\dot{s}_2 = \upsilon_2(t, s) := \sigma_\upsilon(s_1) u_N \tanh\left(\frac{-\tilde{d}(t, s_1)}{\Delta_{p,3}}\right) + (1 - \sigma_\upsilon(s_1)) u_N \tanh\left(\frac{-s_2}{\Delta_{p,4}}\right), \tag{6.15}$$

where the added term is a function that makes the normal vector path speed go to zero at the nominal path, and $\Delta_{p,4}$ is a gain to set the slope at $s_2 = 0$.

6.2.2 Heading correction

In order for the heading to be tangential to the combined path, $\psi_{q_d}(s_1)$ needs to be corrected by

$$\psi_{q_N}(t, s) = \text{atan2}\left(\frac{\dot{s}_2(t, s)}{|q_d^{s_1}(s_1)|}, \dot{s}_1(t, s_1)\right). \tag{6.16}$$

This heading correction is deactivated by an activation function $\sigma_\psi(s_1) \in \{0, 1\}$ close to p_t. Deactivating both $\sigma_\upsilon(s_1)$ and $\sigma_\psi(s_1)$ at the same time would only cause a single

jump in the desired heading signal. The final heading reference yields

$$\psi_d(t,s) = \psi_{q_d}(s_1) + \sigma_\psi(s_1)\psi_{q_N}(t,s). \tag{6.17}$$

6.2.3 COLREGs compliance

In its current form, the combined path is partly in compliance with COLREGs rules 8, 14, and 15. Similarly to the navigation function path-planner in Chapter 5, the combined path provides large enough alteration of course to be noticeable by the TS when choosing an adequate value for $\Delta_{p,2}$. The same approach as in Section 5.1.4 is used to add more compliance with rule 8 by performing a crash stop if the OS is in danger of colliding with the TS. For this guidance method, both $u_d(t)$ and u_N need to be set to zero in order for the OS to stop.

The combined path is partly in compliance with head-on and give-way rules as it uses a d_{ref} that only considers a single TS from p_0 to p_t. The combination of this reference signal and the J matrix allows for a construction of $q_N(s_1, s_2)$ on the starboard side of the OS as it moves along the nominal path.

6.2.4 Combined reference

The desired path and its relevant derivatives are the same as in (5.13). The derivatives of the first speed assignment in (6.1) are equal to (5.19) and (5.20). The relevant derivatives of the second speed assignment in (6.15) are set to zero as a simplification.

The nominal and normal vector path references yield the following guidance system:

$$\dot{s}_1 = \upsilon_1(t,s_1) := \frac{u_d(t)}{|q_d^{s_1}(s_1)|}\rho(s_1), \tag{6.18}$$

$$\dot{s}_2 = \upsilon_2(t,s) := \sigma_\upsilon(s_1)u_N \tanh\left(\frac{-\tilde{d}(t,s_1)}{\Delta_{p,3}}\right) + (1-\sigma_\upsilon(s_1))u_N \tanh\left(\frac{-s_2}{\Delta_{p,4}}\right), \tag{6.19}$$

$$\psi \to \psi_d(t,s) := \text{atan2}\left(y_{q_d}(s_1), x_{q_d}(s_1)\right) + \sigma_\psi(s_1)\text{atan2}\left(\frac{\dot{s}_2(t,s)}{|q_d^{s_1}(s_1)|}, \dot{s}_1(t,s_1)\right), \tag{6.20}$$

$$p \to p_d(s_1,s_2) := q_d(s_1) + s_2 N_d(s_1). \tag{6.21}$$

Chapter 7

Control

This chapter presents two different DP maneuvering control designs for the guidance methods in Chapter 5 and Chapter 6.

7.1 DP maneuvering control designs

Given the control design model in Section 3.2.2, a control law that satisfies the maneuvering control objective presented in Section 3.3.4 is designed by cascade-backstepping. Skjetne (2020*b*) and Skjetne (2020*a*) present DP maneuvering control designs for guidance methods that use one and two path parameters, respectively.

- In Chapter 5, the output η_d is parametrized by a scalar variable $s \in \mathbb{R}$, that is, $\eta_d : \mathbb{R} \to \mathbb{R}^2 \times \mathbb{S}$.
- In Chapter 6, the output η_d consists of $p_d : \mathbb{R}^2 \to \mathbb{R}^2$, parametrized by $s \in \mathbb{R}^2$, and heading reference $\psi_d : \mathbb{R}_{\geq 0} \times \mathbb{R}^2 \to \mathbb{S}$, that is, $\eta_d : \mathbb{R}^2 \to \mathbb{R}^2 \times \mathbb{S}$.

Since both designs are performed by cascade-backstepping and require the same Step 2, this is presented first.

7.1.1 Step 2

Assume that the virtual control $\alpha(t, s, \eta)$ and its derivative $\dot{\alpha}$ are available. Defining the error variable

$$z_2 := \nu - \alpha(t, s, \eta). \tag{7.1}$$

The objective is to control (7.1) exponentially to zero. Differentiating (7.1) yields

$$M\dot{z}_2 = M\dot{\nu} - M\dot{\alpha} = -D\nu + \tau + R(\psi)^\top b - M\dot{\alpha}. \tag{7.2}$$

An estimate of the slowly varying bias $\hat{b} = b$ is assumed provided by the DP observer. Assigning the control law

$$\tau = -K_2 z_2 + D\alpha - R(\psi)^\top \hat{b} + M\dot{\alpha}, \quad K_2 = K_2^\top \succ 0, \tag{7.3}$$

yields

$$M\dot{z}_2 = -(K_2 + D)z_2. \tag{7.4}$$

Let the second CLF be

$$V_2 = \frac{1}{2} z_2^\top M z_2, \tag{7.5}$$

differentiating this yields

$$\dot{V}_2 = z_2^\top M \dot{z}_2 = -z_2^\top (K_2 + D) z_2. \tag{7.6}$$

Then $z_2 = 0$ is UGES for the $\dot{z}_2$-subsystem. Let $T_2 = \text{diag}(T_u, T_v, T_r)$ be a diagonal matrix of preferred time constants for the z_2-subsystem, this gives

$$M\dot{z}_2 = -(K_2 + D) z_2 \quad \Longleftrightarrow \quad T_2 \dot{z}_2 = -z_2 \tag{7.7}$$

$$\Rightarrow K_2 = M T_2^{-1} - D. \tag{7.8}$$

7.1.2 Step 1 using one path parameter

Assume that the desired path map $\eta_d : \mathbb{R} \to \mathbb{R}^2 \times \mathbb{S}$ and the continuous path parameter $s \in \mathbb{R}$ are available. Defining the error variables

$$z_1 := R(\psi)^\top [\eta - \eta_d(s)], \tag{7.9}$$

$$\omega := \dot{s} - \upsilon(t, s), \tag{7.10}$$

where υ is a speed assignment for $\dot{s}$. Differentiating (7.9) with respect to time yields

$$\dot{z}_1 = \dot{R}(\psi)^\top [\eta - \eta_d] + R(\psi)^\top [\dot{\eta} - \eta_d^s \dot{s}] \tag{7.11}$$

$$= -S(r) z_1 + z_2 + \alpha - R(\psi)^\top \eta_d^s (\omega + \upsilon). \tag{7.12}$$

The first CLF is defined as

$$V_1 := \frac{1}{2} z_1^\top z_1, \tag{7.13}$$

differentiating this yields

$$\dot{V}_1 = -z_1^\top S(r) z_1 + z_1^\top z_2 + z_1^\top [\alpha - R(\psi)^\top \eta_d^s (\omega + \upsilon)]. \tag{7.14}$$

The virtual control

$$\alpha = -K_1 z_1 + R(\psi)^\top \eta_d^s \upsilon, \quad K_1 = K_1^\top \succ 0, \tag{7.15}$$

and the tuning function

$$\rho = -z_1^\top R(\psi)^\top \eta_d^s, \tag{7.16}$$

yield the following result of Step 1:

$$\dot{V}_1 = -z_1^\top K_1 z_1 + \rho \omega + z_1^\top z_2, \tag{7.17}$$

$$\dot{z}_1 = -K_1 z_1 - S(r) z_1 + z_2 - R(\psi)^\top \eta_d^s \omega. \tag{7.18}$$

Let $T_1 = \text{diag}(T_x, T_y, T_\psi)$ be a diagonal matrix of preferred time constants for the z_1-subsystem, set $K_1 = T_1^{-1}$, and assume $S(r) z_1 + R(\psi)^\top \eta_d^s \omega = 0$. Then the simplified system

$$T_1 \dot{z}_1 = -z_1 + T_1 z_2, \tag{7.19}$$

$$T_2 \dot{z}_2 = -z_2, \tag{7.20}$$

is used for tuning T_1 and T_2 which determine the gains K_1 and K_2. It is recommended to set $T_2 < T_1$ to make the z_2-subsystem faster than z_1. In Step 2, $\dot{\alpha}$ is required, this is

$$\begin{aligned}\dot{\alpha} = {} & K_1 S(r) z_1 - K_1 \nu - S(r) R(\psi)^\top \eta_d^s \upsilon + R(\psi)^\top \eta_d^s \upsilon^t \\ & + [K_1 R(\psi)^\top \eta_d^s + R(\psi)^\top \eta_d^{s^2} \upsilon + R(\psi)^\top \eta_d^s \upsilon^s] \dot{s}. \end{aligned} \tag{7.21}$$

Update law: The maneuvering update law needs to be decided for it to only act in the output space of η. The objective is to render

$$\dot{V}_1|_{z_2=0} = -z_1^\top K_1 z_1 + \rho\omega, \tag{7.22}$$

negative definite in z_1. Next, ω is chosen and implemented in

$$\dot{s} = \upsilon(t, s) + \omega. \tag{7.23}$$

Choosing the unit-tangent gradient update law yields

$$\omega = \mu \frac{\eta_d^{s\top}}{|\eta_d^s|} R(\psi) z_1, \quad \mu \geq 0. \tag{7.24}$$

For this choice, $\rho\omega \leq 0$.

The final control law and closed-loop system become

$$\boxed{\begin{aligned} \dot{s} &= \upsilon + \omega, && (7.25)\\ \alpha &= -K_1 z_1 + R(\psi)^\top \eta_d^s \upsilon, && (7.26)\\ \tau &= -K_2 z_2 + D\alpha - R(\psi)^\top \hat{b} + M\dot{\alpha}, && (7.27) \end{aligned}}$$

$$\boxed{\begin{aligned} \dot{z}_1 &= -K_1 z_1 - S(r) z_1 + z_2 - R(\psi)^\top \eta_d^s \omega, && (7.28)\\ M\dot{z}_2 &= -(K_2 + D) z_2. && (7.29) \end{aligned}}$$

7.1.3 Step 1 using two path parameters

In this design, the control for the position and heading will be decoupled and later combined into a virtual control α in the end.

Assume that a parametrized path $p_d : \mathbb{R}^2 \to \mathbb{R}^2$, heading reference $\psi_d : \mathbb{R}_{\geq 0} \times \mathbb{R}^2 \to \mathbb{S}$, and path parameters $s := [s_1, s_2]^\top \in \mathbb{R}^2$ are available. Also assume that p_d is designed so that $s \mapsto p_d(s)$ spans $\mathbb{R}^2$. The desired output $\eta_d(s)$ is defined as

$$\eta_d(t, s) := \begin{bmatrix} p_d(s_1, s_2) \\ \psi_d(t, s_1, s_2) \end{bmatrix}, \tag{7.30}$$

so that $\eta_d : \mathbb{R}^2 \to \mathbb{R}^2 \times \mathbb{S}$. The desired speed assignment for s is assumed given by $\upsilon : \mathbb{R}_{\geq 0} \times \mathbb{R}^2 \to \mathbb{R}^2$ defined by

$$\upsilon(t, s) := \begin{bmatrix} \upsilon_1(t, s_1, s_2) \\ \upsilon_2(t, s_1, s_2) \end{bmatrix}. \tag{7.31}$$

Defining the error variables

$$z_{1,p} := R_2(\psi)^\top [p - p_d(s_1, s_2)], \quad z_{1,\psi} := \psi - \psi_d(t, s_1, s_2), \quad z_1 := \mathrm{col}(z_{1,p}, z_{1,\psi}), \tag{7.32}$$
$$z_{2,p} := v - \alpha_p, \quad z_{2,\psi} := r - \alpha_\psi, \quad z_2 := \mathrm{col}(z_{2,p}, z_{2,\psi}), \quad \alpha := \mathrm{col}(\alpha_p, \alpha_\psi), \tag{7.33}$$
$$\omega_1 := \dot{s}_1 - \upsilon_1(t, s_1, s_2), \quad \omega_2 := \dot{s}_2 - \upsilon_2(t, s_1, s_2), \tag{7.34}$$
$$\upsilon := \mathrm{col}(\upsilon_1, \upsilon_2), \quad \omega := \mathrm{col}(\omega_1, \omega_2) = \dot{s} - \upsilon(t, s). \tag{7.35}$$

Then, the position and heading are individually designed. For the position, a maneuvering design is done. Differentiating $z_{1,p}$ with respect to time yields

$$\dot{z}_{1,p} = \dot{R}_2(\psi)^\top [p - p_d] + R_2(\psi)^\top [\dot{p} - p_d^{s_1} \dot{s}_1 - p_d^{s_2} \dot{s}_2] \tag{7.36}$$
$$= -S_2(r) z_{1,p} + z_{2,p} + \alpha_p - R_2(\psi)^\top p_d^{s_1} (\omega_1 + \upsilon_1) - R_2(\psi)^\top p_d^{s_2} (\omega_2 + \upsilon_2). \tag{7.37}$$

The first CLF for position is defined as

$$V_{1,p} := \frac{1}{2} z_{1,p}^\top z_{1,p}, \tag{7.38}$$

differentiating this yields

$$\dot{V}_{1,p} = z_{1,p}^\top [-S_2(r) z_{1,p} + z_{2,p} + \alpha_p - R_2(\psi)^\top p_d^{s_1} (\omega_1 + \upsilon_1) - R_2(\psi)^\top p_d^{s_2} (\omega_2 + \upsilon_2)]. \tag{7.39}$$

The virtual control for position

$$\alpha_p = -K_{1,p} z_{1,p} + R_2(\psi)^\top p_d^{s_1} \upsilon_1 + R_2(\psi)^\top p_d^{s_2} \upsilon_2, \quad K_{1,p} = K_{1,p}^\top \succ 0, \tag{7.40}$$

and the tuning functions with respect to s_1 and s_2

$$\rho_1 = -z_{1,p}^\top R_2(\psi)^\top p_d^{s_1} = V_{1,p}^{s_1}(s_1, s_2, p, \psi), \tag{7.41}$$
$$\rho_2 = -z_{1,p}^\top R_2(\psi)^\top p_d^{s_2} = V_{1,p}^{s_2}(s_1, s_2, p, \psi), \tag{7.42}$$

yield the following result:

$$\dot{V}_{1,p} = -z_{1,p}^\top K_{1,p} z_{1,p} + z_{1,p}^\top z_{2,p} + \rho_1 \omega_1 + \rho_2 \omega_2, \tag{7.43}$$
$$\dot{z}_{1,p} = -(S_2(r) + K_{1,p}) z_{1,p} + z_{2,p} - R_2(\psi)^\top \left(p_d^{s_1} \omega_1 + p_d^{s_2} \omega_2 \right), \tag{7.44}$$

where $z_{2,p} = \begin{bmatrix} I_{2\times 2} & 0_{2\times 1} \end{bmatrix} z_2$.

For the heading, a direct tracking design is done, assuming that $\dot{\psi}_d$ is available as a signal. Differentiating $z_{1,\psi}$ with respect to time yields

$$\dot{z}_{1,\psi} = \dot{\psi} - \dot{\psi}_d = z_{2,\psi} + \alpha_\psi - \dot{\psi}_d. \tag{7.45}$$

The first CLF for heading is defined as

$$V_{1,\psi} := \frac{1}{2} z_{1,\psi}^2, \tag{7.46}$$

differentiating this yields

$$\dot{V}_{1,\psi} = z_{1,\psi} [z_{2,\psi} + \alpha_\psi - \dot{\psi}_d]. \tag{7.47}$$

The virtual control for heading

$$\alpha_\psi = -k_{1,\psi} z_{1,\psi} + \dot{\psi}_d, \quad k_{1,\psi} > 0, \tag{7.48}$$

yields the following result:

$$\dot{V}_{1,\psi} = -k_{1,\psi} z_{1,\psi}^2 + z_{1,\psi} z_{2,\psi}, \tag{7.49}$$
$$\dot{z}_{1,\psi} = -k_{1,\psi} z_{1,\psi} + z_{2,\psi}, \tag{7.50}$$

where $z_{2,\psi} = \begin{bmatrix} 0_{1\times 2} & 1 \end{bmatrix} z_2$. Let $T_{1,p} = \mathrm{diag}(T_x, T_y)$ be a diagonal matrix of preferred time constants for the $z_{1,p}$-subsystem and T_ψ be a time constant for the $z_{1,\psi}$-subsystem. Set $K_{1,p} = T_{1,p}^{-1}$ and $k_{1,\psi} = 1/T_\psi$, and assume $S_2(r) z_{1,p} + R_2(\psi)^\top (p_d^{s_1} \omega_1 + p_d^{s_2} \omega_2) = 0$. Then the simplified system

$$T_{1,p} \dot{z}_{1,p} = -z_{1,p} + T_{1,p} z_{2,p}, \tag{7.51}$$
$$T_\psi \dot{z}_{1,\psi} = -z_{1,\psi} + T_\psi z_{2,\psi}, \tag{7.52}$$
$$T_2 \dot{z}_2 = -z_2, \tag{7.53}$$

is used for tuning $T_{1,p}$, $T_{1,\psi}$, and T_2 which determine the gains $K_{1,p}$, $K_{1,\psi}$, and K_2. It is recommended to set $T_2 < \mathrm{diag}(T_{1,p}, T_{1,\psi})$ to make the z_2-subsystem faster than $z_{1,p}$ and $z_{1,\psi}$. In Step 2, $\dot{\alpha}$ is required, this consists of

$$\begin{aligned} \dot{\alpha}_p = {} & K_{1,p} S_2(r) z_{1,p} - K_{1,p} v + K_{1,p} R_2(\psi)^\top (p_d^{s_1} \dot{s}_1 + p_d^{s_2} \dot{s}_2) \\ & - S_2(r) R_2(\psi)^\top (p_d^{s_1} \upsilon_1 + p_d^{s_2} \upsilon_2) \\ & + R_2(\psi)^\top (p_d^{s_1^2} \upsilon_1 \dot{s}_1 + p_d^{s_1} \upsilon_1^{s_1} \dot{s}_1 + p_d^{s_1} \upsilon_1^{s_2} \dot{s}_2 + p_d^{s_1} \upsilon_1^t) \\ & + R_2(\psi)^\top (p_d^{s_2^2} \upsilon_2 \dot{s}_2 + p_d^{s_2} \upsilon_2^{s_1} \dot{s}_1 + p_d^{s_2} \upsilon_2^{s_2} \dot{s}_2 + p_d^{s_2} \upsilon_2^t), \end{aligned} \tag{7.54}$$
$$\dot{\alpha}_\psi = -k_{1,\psi}(r - \dot{\psi}_d) + \ddot{\psi}_d. \tag{7.55}$$

Update laws: Next, the maneuvering update laws need to be decided for them to only act in the output space of p. The objective is to render

$$\dot{V}_{1,p}\Big|_{z_{2,p}=0} = -z_{1,p}^\top K_{1,p} z_{1,p} + \rho_1 \omega_1 + \rho_2 \omega_2, \tag{7.56}$$

negative definite in $z_{1,p}$. Next, ω_1 and ω_2 are chosen and implemented in

$$\dot{s} = \upsilon(t, s) + \omega \quad \begin{cases} \dot{s}_1 = \upsilon_1(t, s_1, s_2) + \omega_1, \\ \dot{s}_2 = \upsilon_2(t, s_1, s_2) + \omega_2. \end{cases} \tag{7.57}$$

Choosing the unit-tangent gradient update law yields

$$\omega_1 = \frac{-\mu_1 \rho_1}{|p_d^{s_1}| + \epsilon} = \mu_1 \frac{{p_d^{s_1}}^\top}{|p_d^{s_1}| + \epsilon} R_2(\psi) z_{1,p}, \quad \mu_1 \geq 0, \tag{7.58}$$

$$\omega_2 = \frac{-\mu_2 \rho_2}{|p_d^{s_2}| + \epsilon} = \mu_2 \frac{{p_d^{s_2}}^\top}{|p_d^{s_2}| + \epsilon} R_2(\psi) z_{1,p}, \quad \mu_2 \geq 0, \tag{7.59}$$

where $0 < \epsilon \ll 1$ is a small constant. For this choice, $\rho_1\omega_1 \leq 0$ and $\rho_2\omega_2 \leq 0$.

The final control law and closed-loop system yield for the position

$$\boxed{\begin{aligned} \dot{s}_1 &= \upsilon_1 + \omega_1, \\ \dot{s}_2 &= \upsilon_2 + \omega_2, \\ \alpha_p &= -K_{1,p} z_{1,p} + R_2(\psi)^\top p_d^{s_1} \upsilon_1 + R_2(\psi)^\top p_d^{s_2} \upsilon_2, \\ \tau &= -K_2 z_2 + D\alpha - R(\psi)^\top \hat{b} + M\dot{\alpha}, \end{aligned}} \tag{7.60}$$

$$\boxed{\begin{aligned} \dot{z}_{1,p} &= -(S_2(r) + K_{1,p}) z_{1,p} + z_{2,p} - R_2(\psi)^\top \left(p_d^{s_1} \omega_1 + p_d^{s_2} \omega_2 \right), \\ M\dot{z}_2 &= -(K_2 + D) z_2, \quad z_{2,p} = \begin{bmatrix} I_{2\times 2} & 0_{2\times 1} \end{bmatrix} z_2, \end{aligned}} \tag{7.64}$$

and for the heading

$$\boxed{\begin{aligned} \alpha_\psi &= -k_{1,\psi} z_{1,\psi} + \dot{\psi}_d, \\ \tau &= -K_2 z_2 + D\alpha - R(\psi)^\top \hat{b} + M\dot{\alpha}, \end{aligned}} \tag{7.66}$$

$$\boxed{\begin{aligned} \dot{z}_{1,\psi} &= -k_{1,\psi} z_{1,\psi} + z_{2,\psi}, \\ M\dot{z}_2 &= -(K_2 + D) z_2, \quad z_{2,\psi} = \begin{bmatrix} 0_{1\times 2} & 1 \end{bmatrix} z_2. \end{aligned}} \tag{7.68}$$

7.2 Control allocation

The thrust allocation module is a necessary part of the control system. Its objective is to distribute the commanded generalized control forces and moment τ_{cmd} as control input $u_{i,cmd}$ to each actuator i of the vessel. The rectangular coordinate thrust allocation from Fossen (2011) yields the thrust mapping

$$\tau = Bf, \tag{7.70}$$

where $B \in \mathbb{R}^{3\times r}$ is the extended thrust configuration matrix and $f \in \mathbb{R}^r$ is a thrust column vector given in $\{b\}$. The number of thrusters is denoted by r, with the azimuth thrusters being treated as two thrusters. The commanded thrust f_{cmd} is determined by the weighted Moore-Penrose pseudoinverse that yields

$$f_{cmd} = B^\dagger \tau_{cmd}, \quad B^\dagger = B^\top (BB^\top)^{-1}. \tag{7.71}$$

This method does not take into account constraints such as thruster saturation and forbidden sectors. After f_{cmd} is determined, the commanded thrust $f_{i,cmd}$ for each thruster needs to be calculated. Azimuth thrusters yield

$$f_{i,cmd} = \sqrt{f_{i,x,cmd}^2 + f_{i,y,cmd}^2}, \tag{7.72}$$

and angles

$$\alpha_{i,cmd} = \operatorname{atan2}\left(f_{i,y,cmd}, f_{i,x,cmd} \right), \tag{7.73}$$

while tunnel thrusters yield

$$f_{i,cmd} = f_{i,cmd}. \tag{7.74}$$

If each thrust is scaled by a gain k_i so that a normalized signal $u_{i,cmd}$ is obtained, that is

$$f_{i,cmd} = k_i u_{i,cmd}, \tag{7.75}$$

this results in

$$u_{i,cmd} = \frac{f_{i,cmd}}{k_i}. \tag{7.76}$$

Chapter 8

Simulation setup

This chapter presents the model ship CSEI used in the simulation study and the preparatory work done on the vessel.

8.1 CSEI

The simulations were conducted on a high-fidelity 3 DOF simulation model of CSEI using parameters from NTNU (2020). CSEI, shown in Figure 8.1, is used for experimentation in the Marine Cybernetics Laboratory (MC-Lab). CSEI is a 1:50 scale model of a tugboat mounted with two Voith Schneider propellers (VSPs) astern and a bow thruster (BT). The main dimensions of CSEI can be found in Table 8.1.

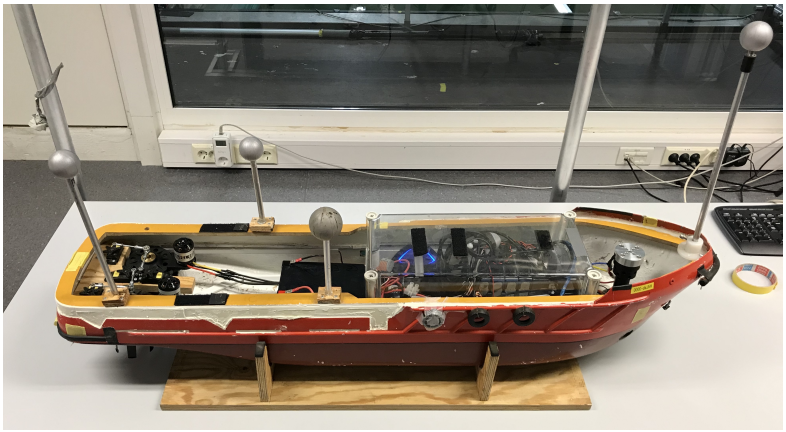


Figure 8.1: CSEI.

Table 8.1: CSEI's main dimensions. Courtesy of NTNU (2020).

Parameter	Value
L_{oa}	1.105 m
B	0.248 m

CSEI's actuators are located as in Figure 8.2 with dimensions from Table 8.2.

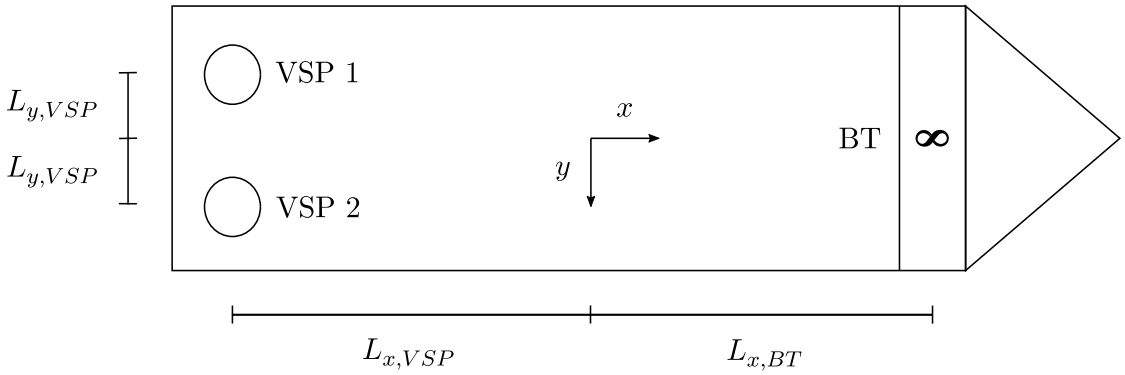


Figure 8.2: Location of CSEI's actuators.

Table 8.2: Location of CSEI's actuators. Courtesy of NTNU (2020).

Parameter	Description	Value
$L_{x,BT}$	x length to BT	0.3875 m
$L_{x,VSP}$	x length to VSP	0.4574 m
$L_{y,VSP}$	y length to VSP	0.055 m

CSEI's rigid body, added mass, and drag coefficients are given in Table 8.3.

Table 8.3: CSEI's rigid body, added mass, and damping coefficients. Courtesy of NTNU (2020).

Rigid body		Added mass		Damping	
Parameter	Value	Parameter	Value	Parameter	Value
m	14.11	$X_{\dot{u}}$	-2	X_u	-0.6555
I_z	1.76	$Y_{\dot{v}}$	-10	Y_v	-1.33
x_g	0.0375	$Y_{\dot{r}}$	0	Y_r	-7.25
y_g	0	$N_{\dot{r}}$	-1	N_v	0
				N_r	-1.9

8.1.1 Preparatory work

CSEI was prepared for use in experiments in the MC-Lab by an update of the mapping between the control input and thrust given from each thruster. The update was done by performing several bollard pull tests on the vessel. Before performing the bollard pull tests, the VSP servos were tuned so that these would give thrust in the desired control allocation direction.

The work culminated with an update of NTNU (2020) with guidelines on how to tune the VSP servos and how to perform the bollard pull tests.

8.1.2 Thruster allocation and dynamics

To emulate real-world experiments with the model ship, a thrust allocation and thruster dynamics module were implemented in the simulation setup. The rectangular coordinate thrust allocation method from Section 7.2 was implemented using parameters from Table 8.2 and scales obtained from the bollard pull tests. A simplified thruster dynamics module was implemented to map the control input to forces and moment acting on the ship. The thruster losses, forbidden sectors of VSP angles, and turning rate of thrusters were not taken into account.

Chapter 9

Results

This chapter presents the simulations of the guidance methods in two different scenarios and discusses the results.

9.1 Simulations

The following scenarios are used when simulating the guidance methods from Chapter 5 and Chapter 6:

- **Head-on scenario:** The OS and a TS are on near reciprocal courses.
- **Give-way scenario:** The OS and a TS are crossing each other with the OS being the give-way vessel.

Let the initial position and heading of the OS be $\eta_0 = [-14, -2, 0.142]^\top$. Also, let the destination be located at $p_t = [0, 0]^\top$ and the OS's reference speed be $u_d = 0.1$ m/s. The current ν_c is constant with speed 0.01 m/s going northwest.

The simulations have a single circular TS with radius $r_{TS} = 0.5$ m that has a constant speed $u_{TS} = 0.1$ m/s and course χ_{TS}. In the head-on scenario, the TS's initial position is $p_{0,TS} = [0, -1]^\top$ and $\chi_{TS} = \pi$. In the give-way scenario, the TS's initial position is $p_{0,TS} = [-5.8, 6.2]^\top$ and $\chi_{TS} = -1.736$.

Four figures are shown for each simulation. The first is a N/E plot showing the desired and real path of the OS, the path of the TS, and outlines of the vessels at timestamps with 15s intervals. The second shows the desired and real position and heading. The third displays the real and estimated velocities. The fourth shows the desired and real forces and moment. Videos that show simulations of the two guidance methods can be found in Appendix A.

The parameters used to simulate the first and second guidance method are shown in Table 9.1 and Table 9.2, respectively. The following are the gains after tuning the controllers:

$$T_1 = \text{diag}(2, 1, 1), \quad T_2 = \text{diag}(1, 0.8, 0.5), \tag{9.1}$$

$$T_{1,p} = \text{diag}(2, 1), \quad T_{1,\psi} = 1, \quad T_2 = \text{diag}(1, 1, 2). \tag{9.2}$$

Table 9.1: The first guidance method parameters for head-on and give-way scenarios.

Parameter	Value	Parameter	Value
$\Delta_{p,1}$	0.1	Δ_{ψ}	$\pi/2$
μ	0.1	λ_{ψ}	0.001
λ	2	r_{HO}	$2r_{TS}$
κ	6 and 12	n_{HO}	4
R	L_{oa}	r_{GW}	$3r_{TS}$
d_0	4.2 and 3.2		

Table 9.2: The second guidance method parameters for head-on and give-way scenarios.

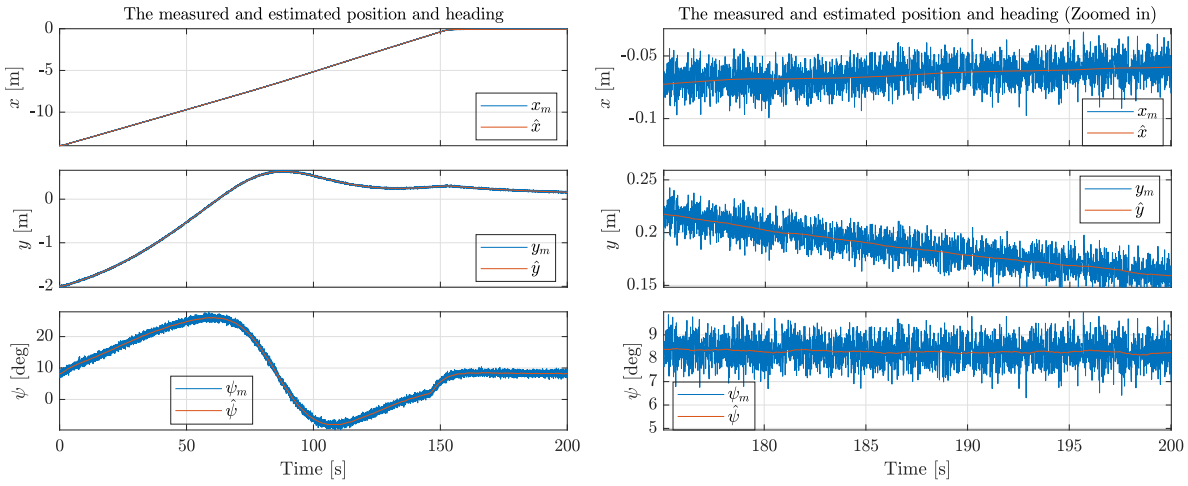
Parameter	Value	Parameter	Value
$\Delta_{p,1}$	0.01	$\Delta_{p,3}$	1
d_{safe}	1.5 and 2.5	$\Delta_{p,4}$	2
u_N	0.1	μ_1	0.01
$\Delta_{p,2}$	6 and 3.5	μ_2	0

Observer verification

The observer was tuned and verified before simulating the scenarios. The following are the injection gain matrices after tuning the observer:

$$L_1 = \text{diag}(0.05, 0.5, 0.75), \quad L_2 = \text{diag}(5, 12.5, 2.5), \quad L_3 = \text{diag}(0.15, 0.7, 0.15). \quad (9.3)$$

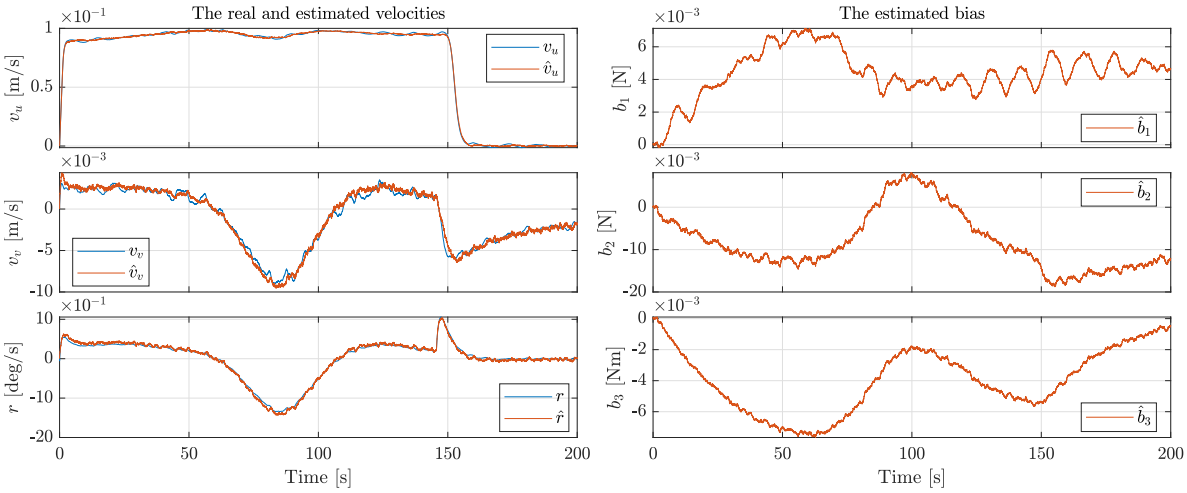
Figure 9.1 shows the measured signal y_m with a zero-mean Gaussian measurement noise v of power 10^{-6} and the estimated position and heading $\hat{\eta}$. As can be seen from Figure 9.1b, the noise is removed from the measured signal.



(a) Complete plot. (b) Zoomed in.

Figure 9.1: The measured and estimated position and heading.

Figure 9.2 shows that the observer provides state estimates of the velocities and bias. Figure 9.2a displays a comparison of the real velocity ν and the estimated velocity $\hat{\nu}$. As can be seen, the latter is a good estimation of the unmeasured velocity. Figure 9.2b shows that the dynamics related to the current and other unmodeled effects are represented by an estimated slowly varying bias $\hat{b}$.



(a) The real and estimated velocities.

(b) The estimated bias.

Figure 9.2: State estimates of the velocities and bias.

9.1.1 Guidance with navigation function

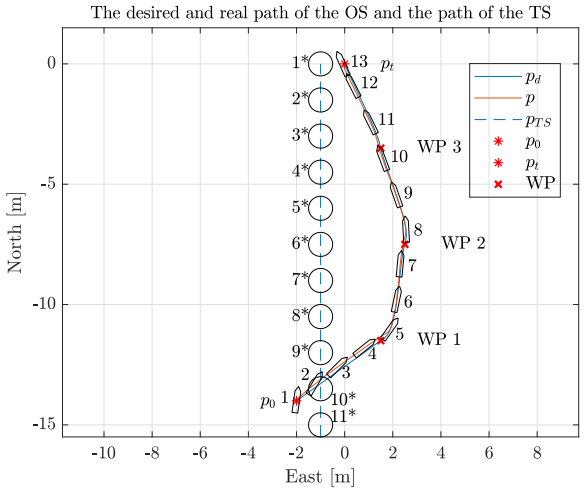
Head-on scenario

The overview in Figure 9.3a displays that the first WP generated by the path-planner makes the OS start a starboard course change. The next WP enforces a port course change for the OS so that the vessels pass on the port side of each other. In this way, the OS avoids collision by a maneuver that is in accordance with COLREGs rule 14. The last WP makes the OS continue its port course change to close in on the destination as the TS has passed and the danger is over. As the desired heading at p_t is not specified for this guidance method, the OS is set to keep its final heading reference created by the last WP and p_t.

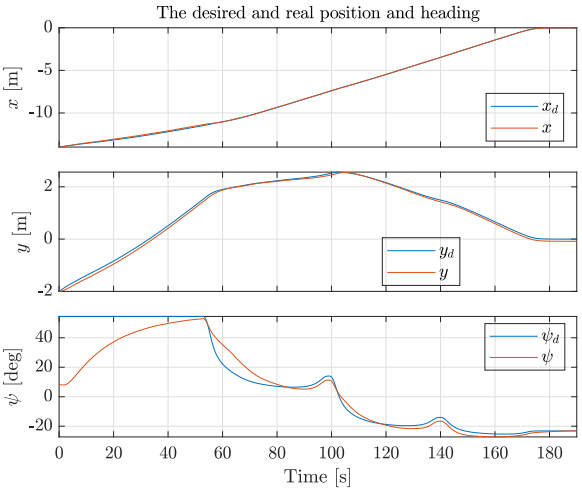
Figure 9.3b shows that the OS is able to follow the desired position, but it is apparent that it struggles with a sudden and large change in the heading reference. A similar behavior is observed in the give-way scenario in Figure 9.4b. Figures 9.3c and 9.4c show that the speed assignment enforces the OS's speed to go to zero at the destination for both scenarios.

Figures 9.3d and 9.4d display that the sway force and yaw moment have spikes after most of the WPs. This causes the OS's thrusters to be saturated and unable to give the desired force and moment. The spikes worsen with the change in heading reference and cause an undesirable oscillating behavior in the forces and in the linear velocities as seen

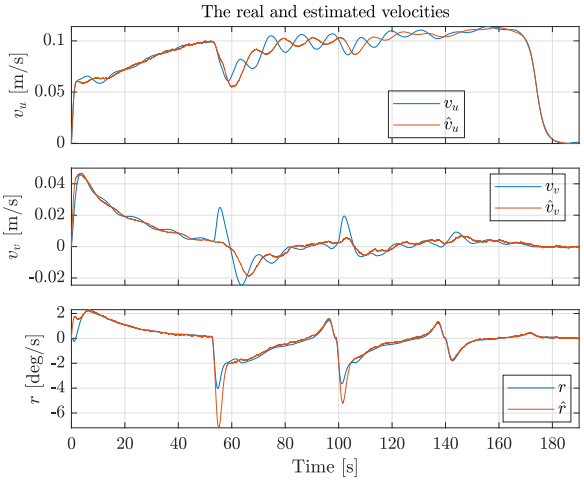
in figures 9.3c and 9.4c.



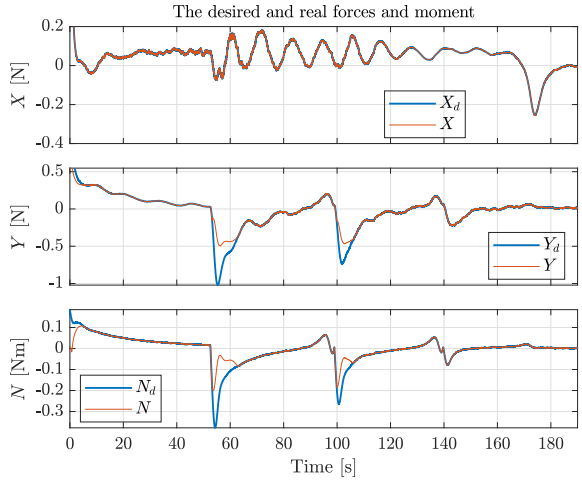
(a) The desired and real path of the OS and the path of the TS.



(b) The desired and real position and heading.



(c) The real and estimated velocities.

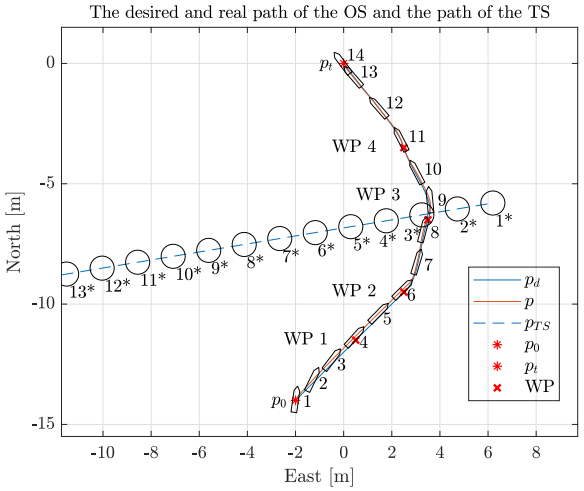


(d) The desired and real forces and moment.

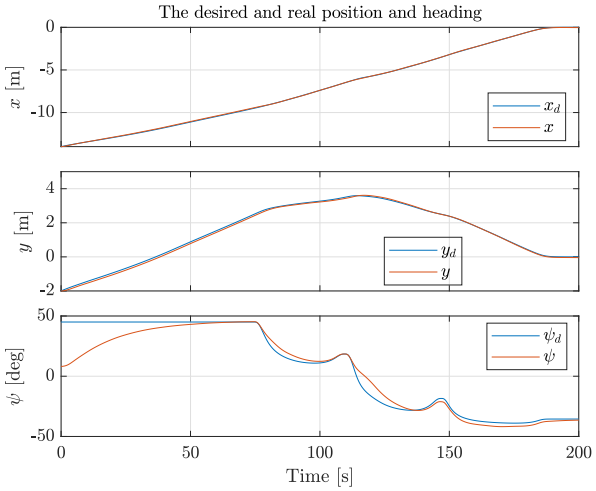
Figure 9.3: A head-on scenario with the first guidance method.

Give-way scenario

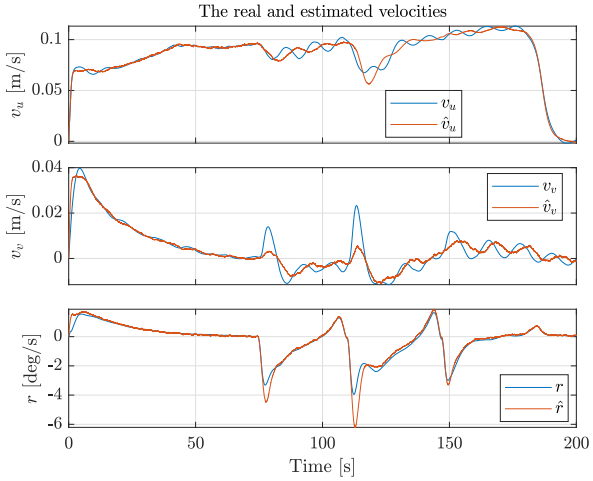
Figure 9.4a shows that the first WP generated by the path-planner makes the OS start a starboard course change so that it keeps out of the way of the TS. The second WP lets the OS continue with the same course in order for it to pass behind the TS. The next WP makes the OS do a port course change as the TS has passed and the danger is over. The last WP continues this port course change to close in on the destination. The WPs are calculated such that the OS avoids crossing ahead of the TS in accordance with rule 15.



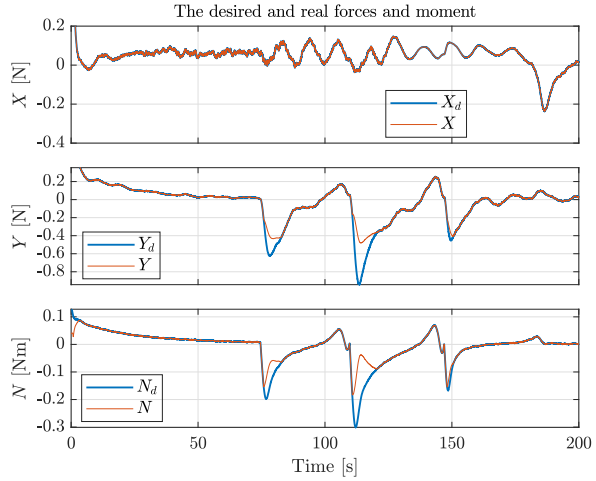
(a) The desired and real path of the OS and the path of the TS.



(b) The desired and real position and heading.



(c) The real and estimated velocities.



(d) The desired and real forces and moment.

Figure 9.4: A give-way scenario with the first guidance method.

Discussion

To summarize, the simulations using the first guidance method showed that the path-planner was able to provide WPs that could bring the OS from p_0 to p_t in accordance with COLREGs rules 14 and 15. The path generation module allowed for a feasible path to be constructed for the OS in a stepwise manner. The OS was capable of following the desired position in a satisfactory way and the speed assignment enforced the OS to slow down at the destination. However, the OS struggled with large and sudden changes in the heading reference.

The OS may not be able to achieve the desired heading for a number of reasons. It could be due to sub-optimal controller gains or that some of the turns simply are too

sharp for the OS given the proposed reference speed. However, reducing the reference speed would also affect the efficiency of the operation and cannot be considered to be the only solution. As the WPs are calculated online, the hybrid path generation method will construct a sharp turn if the distance between the WPs is short and/or the angle created by the previous and next WP is large.

There are many parameters that affect the WP choice and this makes the guidance method difficult to tune. Increasing the radius of the neighborhood d_0 would allow for a less demanding path to be made but would greatly affect how reactive the OS is. Too large value would make it difficult to ensure that a collision-free path can be constructed. This is because: (1) The environment would vary more between WP calculations; (2) The path-planner does not take into account that the cells between the current and next WP can be occupied by TSs.

Increasing the weight on heading changes λ_ψ and reducing the heading change limit Δ_ψ could be a solution to achieve smaller changes in the heading reference. Again, this would affect how reactive the OS would be when encountering TSs as well as the efficiency of the overall path. Figures 9.3a and 9.4a show that these parameters create a large safety margin to the TS resulting in a longer and less efficient path to p_t.

The guidance method's ability to construct a collision-free path could have been improved by applying an even smaller circle of acceptance radius R. Then the OS would calculate the next WP based on a more recent update about the surroundings. Such a change can be crucial when operating in rapidly changing environments which the harbors might be. Reducing R would not only improve the path-planner's reliability but also the efficiency of the path as the path-planner might propose unnecessary large safety margins to a TS that no longer poses a threat. However, too small R would make the circle difficult to reach for the OS. Another step could be to let the dynamic TSs occupy more cells in the TS's course direction to allow the OS to react appropriately by anticipating how the environment will change.

Lastly, it is not realistic to assume that the large values given in the forces and moment after the WPs would be achieved in real-world experiments. This is because the simulations do not take into account the physical limitations of the thrusters such as the rotation time and how fast the propeller spin can be changed. Other unmodeled thruster dynamics that would greatly affect a real-world scenario would be thruster losses. On the whole, it is evident that the responses from the simulations would worsen with real-world experiments.

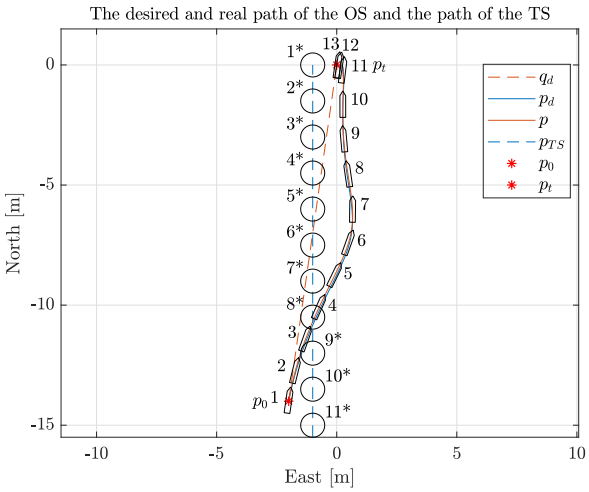
9.1.2 Guidance with two path parameters

Head-on scenario

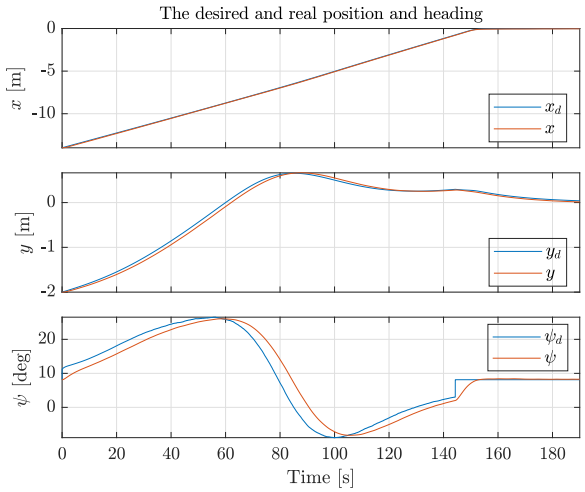
The overview in Figure 9.5a shows that the combined path p_d makes the OS start a starboard course change that endures until about timestamp 5. After this, p_d makes the OS adjust its course to return to the nominal path and the vessels pass on the port side of each other in accordance with COLREGs rule 14. The port course change lasts for

about 45s until the OS adjusts its heading for a return to the heading along the nominal path.

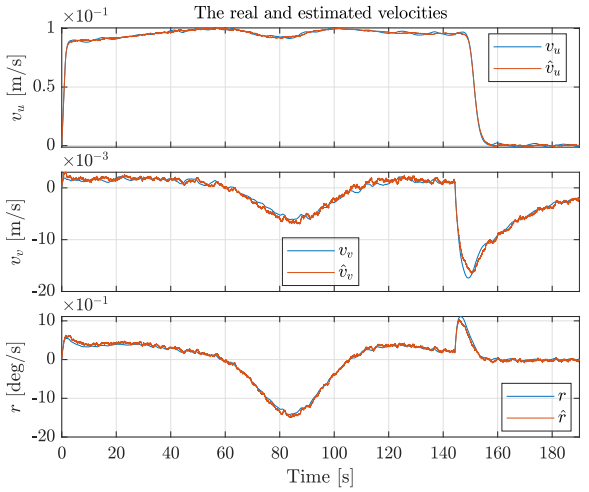
Figures 9.5b and 9.6b show that the OS is able to follow the desired position and heading for both scenarios. The performance is slightly worse for the heading in the give-way scenario as the reference signal has a larger change in a shorter amount of time.



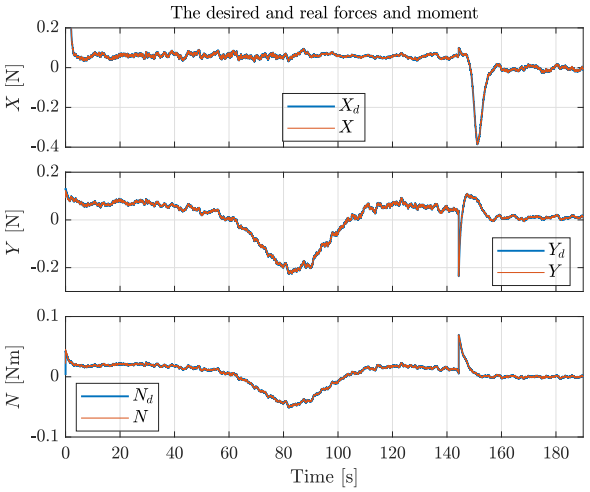
(a) The desired and real path of the OS and the path of the TS.



(b) The desired and real position and heading.



(c) The real and estimated velocities.



(d) The desired and real forces and moment.

Figure 9.5: A head-on scenario with the second guidance method.

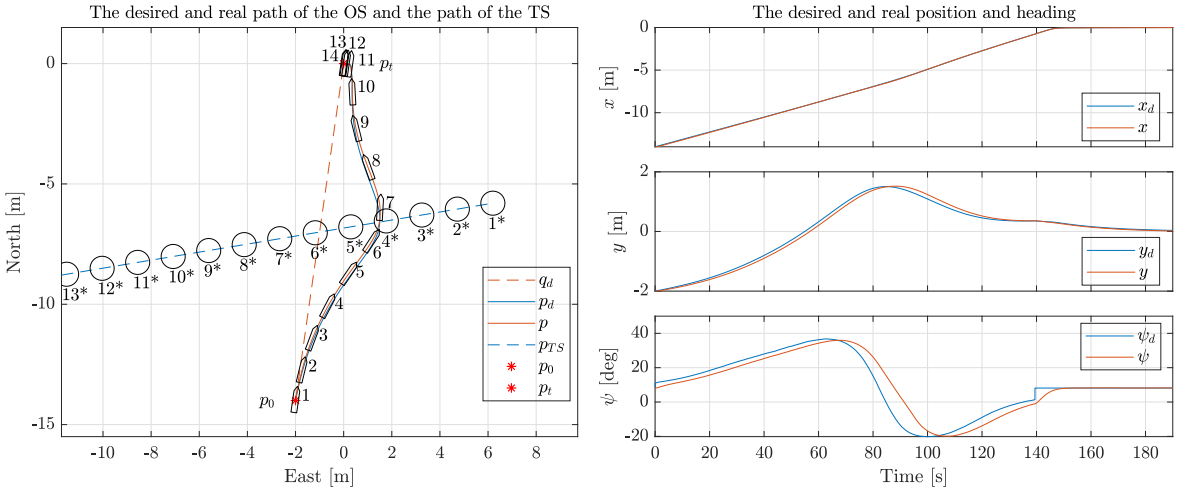
The activation functions σ_ψ and σ_υ are set to be deactivated when $s_1 \geq 0.95$ and this causes the OS to have a jump in the desired heading signal in Figure 9.5b at about 145s. The heading reference jumps from the current value to the nominal path-tangential value. A similar jump can be observed in Figure 9.6b. The second activation function enforces a change in the speed assignment on s_2 which makes the sway speed go to zero at the

nominal path. Figures 9.5c and 9.6c show that the speed assignment on s_1 enforces the OS's surge speed to go to zero at $s_1 = 1$. The activation functions cause spikes in the forces and moment as seen in figures 9.5d and 9.6d.

From Figure 9.5c it is observed that the OS's surge speed is close to the chosen reference speed u_d. A similar but slightly more varying surge speed is achieved in Figure 9.6c given a give-way scenario. As seen from Figure 9.5d, none of the thrusters are saturated as the desired position and heading require less effort to follow compared to the give-way scenario in Figure 9.6d. The sharper turn in the latter scenario causes the forces and moment to saturate for a short time. For both scenarios, the forces and moment are noisy.

Give-way scenario

The overview in Figure 9.6a displays that p_d initiates a starboard course change that endures until between timestamp 5 and 6. This forces the OS to keep out of the TS's way. Then p_d makes the OS adjust its course for a return to the nominal path as the situation is safe. The OS passes behind the TS in accordance with COLREGs rule 15.



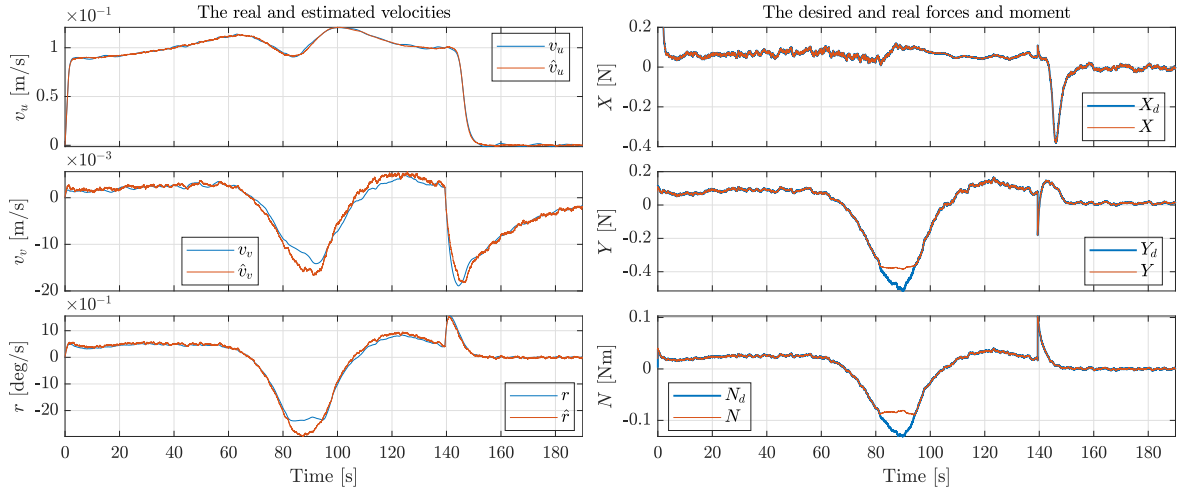
(a) The desired and real path of the OS and the path of the TS.

(b) The desired and real position and heading.

Figure 9.6: A give-way scenario with the second guidance method.

Discussion

In summary, the simulations showed that the second guidance method was able to construct a collision-free path from p_0 to p_t. This was done by combining a nominal path parametrized by s_1 and a normal vector path that can be offset using a second path parameter s_2. The speed assignment on s_1 made the OS follow the nominal path component while keeping the reference speed. The speed assignment on s_2 allowed the OS to follow d_{ref} to avoid the TS in accordance with COLREGs rules 14 and 15. The combination of the two speed assignments provided a path-tangential heading signal and allowed



(c) The real and estimated velocities. (d) The desired and real forces and moment.

Figure 9.6: A give-way scenario with the second guidance method (continued).

the OS to slowly converge to the destination for both scenarios. It would be possible to converge to the nominal path sooner by deactivating σ_υ earlier, but this would also cause an additional jump in the heading reference. The OS was able to follow the desired position and heading in a satisfactory manner for both scenarios. However, the performance was slightly worse for following the desired heading in the give-way scenario. This is due to a sharper turn created by d_{ref} and could be improved by using a larger gain $\Delta_{p,2}$ to construct a smoother reference signal with a less abrupt turn. This solution would remove the saturation in Y and N that occur during the turn, but it would also create a larger reference value at p_0 and p_t making for a rougher transition from the nominal path.

In addition to the saturation in the give-way scenario, there are some other unwanted behaviors related to the loads. There are spikes that occur in the forces and moment when the activation functions are deactivated. As mentioned in Section 9.1.1, it is not realistic to assume that these loads can be given in a real-world scenario due to the simplified thruster dynamics in the simulation model. A simple solution could be to ramp the reference signal in heading to avoid the jump that occurs. Also, the noise in the forces and moment could be reduced by assigning more optimal values for the guidance method parameters and controller gains.

Lastly, it is important to take into account that the proposed reference signal assumes that there is only a single TS between p_0 and p_t. This is far from realistic as the OS is supposed to maneuver in what might be a highly trafficked harbor. Still, the simulations showed that the reference signal yielded an efficient path which the OS was able to follow, and that the guidance method accomplishes its objective.

Chapter 10

Conclusions and further work

10.1 Conclusions

This thesis has proposed two guidance methods for an autonomous ship maneuvering in a harbor with the purpose of bringing the ship from a transit state to its final docking state. Observer and maneuvering control designs were also presented and integrated with the guidance systems. The autonomous system was verified through simulations showing that both guidance methods were able to complete the task safely and in accordance with COLREGs rules 14 and 15.

The path-planner in the first guidance method was able to provide WPs in an online and repeated manner to the destination by translating the environment with a navigation function. An established stepwise hybrid path generation method was used to generate a feasible path connecting the WPs. This yielded a collision-free but inefficient path because of unnecessary large safety margins to the TS which resulted in a long overall path. It was also shown that the hybrid path parametrization concatenating the WPs yielded a path that included too sharp turns for the OS's heading to follow.

The second guidance method was able to combine a nominal path with a normal vector path. This was done by using two path parameters and speed assignments to reactively and continuously generate a path to the destination. The path generation module provided a collision-free and efficient path that the OS was able to follow well. The speed assignments made the OS follow along the nominal path at the proposed reference signal and converge to the destination.

10.2 Recommendations for further work

As mentioned in Section 5.1.4, the added COLREGs compliance with give-way is not a robust solution. This method adds especially many saddle points in the navigation function and can trap the OS if there are multiple TSs. It is therefore suggested that another method should be designed in order for the navigation function to comply with COLREGs rule 15. Another improvement could be made by letting the dynamic TSs occupy more cells in the TS's course direction to allow the OS to react appropriately. In addition, the number of cells could depend on the speed of the TS.

For the second guidance method, the reference signal only considers a single TS and should therefore be modified or replaced with a more appropriate signal that can handle multiple TSs. Compliance with COLREGs rule 13 could also be made by categorizing the overtaking encounter type and using this to determine the direction of the J matrix.

Lastly, it is recommended that an experimental testing of the overall system should be conducted as only simulations of the system were done to verify it.

Bibliography

14th ITTC (1975), Discussion and Recommendations for an ITTC 1975 Maneuvering Trial Code, *in* 'Proceedings of the 14th International Towing Tank Conference', Ottawa, pp. 348–365.

AAWA (2016), *Remote and Autonomous Ships: The Next Steps*, London: Rolls-Royce plc.

Babicz, J. (2015), *Wärtsilä Encyclopedia of Ship Technology*, 2nd edn, Gdańsk: Wärtsilä Corporation.

COLREG (1972), *Convention on the International Regulations for Preventing Collisions at Sea*, International Maritime Organization.

DNV GL (2018), *Class Guideline DNVGL-CG-0264 Autonomous and Remotely Operated Ships*, DNV GL AS.

DNV GL (2019), *Rules for Classification, Ships: Part 6 Additional Class Notations, Chapter 3 Navigation, Manoeuvring and Position Keeping*, DNV GL AS.

Dubins, L. (1957), 'On Curves of Minimal Length with a Constraint on Average Curvature and with Prescribed Initial and Terminal Positions and Tangents', *American Journal of Mathematics* **79**(3), 497–516. doi: 10.2307/2372560.

Eriksen, B. H. and Breivik, M. (2017), MPC-based Mid-level Collision Avoidance for ASVs using Nonlinear Programming, Hawai'i, USA, pp. 766–772.

Fossen, T. I. (2011), *Handbook of Marine Craft Hydrodynamics and Motion Control*, 1st edn, West Sussex: John Wiley & Sons, Ltd.

Fossen, T. I. (2020), *Handbook of Marine Craft Hydrodynamics and Motion Control*, 2nd edn, West Sussex: John Wiley & Sons, Ltd.

Ge, S. S. and Cui, Y. J. (2002), 'Dynamic Motion Planning for Mobile Robots Using Potential Field method', *Autonomous Robots* **13**, 207–222. doi: 10.1023/A:1020564024509.

Hagen, I. B., Kufoalor, D. K. M., Brekke, E. F. and Johansen, T. A. (2018), MPC-based Collision Avoidance Strategy for Existing Marine Vessel Guidance Systems, Brisbane, Australia, pp. 7618–7623.

Huang, Y., Chen, L., Chen, P., Negenborn, R. R. and van Gelder, P. H. A. J. M. (2020), 'Ship Collision Avoidance Methods: State-of-the-art', *Safety Science* **121**, 451–473. doi: 10.1016/j.ssci.2019.09.018.

Jensen, J. S. (2019), 'Dynamic Optimal Path-Planning for Autonomous Harbor Maneuvering - Project thesis'.

Khatib, O. (1985), Real-Time Obstacle Avoidance for Manipulators and Mobile Robots, St. Louis, MO, USA.

Koditschek, D. E. and Rimon, E. (1990), 'Robot Navigation Functions on Manifolds with Boundary', *Advances in Applied Mathematics* **11**, 412–442. doi: 10.1016/0196–8858(90)90017–S.

Leedekerken, J. C., Fallon, M. F. and Leonard, J. J. (2014), Mapping Complex Marine Environments with Autonomous Surface Craft, *in* 'Experimental Robotics', Vol. 79, Springer, pp. 525–539.

Li, G., Yamashita, A., Asama, H. and Tamura, Y. (2012), An Efficient Improved Artificial Potential Field Based Regression Search Method for Robot Path Planning, Chengdu, China, pp. 1227–1232.

Liu, Z., Zhang, Y., Yu, X. and Yuan, C. (2016), 'Unmanned Surface Vehicles: An Overview of Developments and Challenges', *Annual Reviews in Control* **41**, 71–93. doi: 10.1016/j.arcontrol.2016.04.018.

Meng, M. and Yang, X. (1998), A Neural Network Approach to Real-Time Trajectory Generation, Vol. 2, IEEE, Leuven, Belgium, pp. 1725–1730.

Mukhtar, A., Xia, L. and Tang, T. (2015), 'Vehicle Detection Techniques for Collision Avoidance Systems: A Review', *IEEE Transactions on Intelligent Transportation Systems* **16**(5), 2318–2338. doi: 10.1109/TITS.2015.2409109.

Murdoch, E., Dand, I. W. and Clarke, C. (2012), *A Master's Guide to: Berthing*, 2nd edn, London: The Standard.

Naeem, W., Henrique, S. C. and Hu, L. (2016), 'A Reactive COLREGs-Compliant Navigation Strategy for Autonomous Maritime Navigation', *IFAC* **49**(23), 207–213. doi: 10.1016/j.ifacol.2016.10.344.

NTNU (2020), 'CyberShip Enterprise I: User Manual'.

Rimon, E. and Koditschek, D. E. (1992), 'Exact Robot Navigation Using Artificial Potential Functions', *IEEE Transactions on Robotics and Automation* **8**(5), 501–518. Available at: https://repository.upenn.edu/ese_papers/323/ (Accessed: 3/3–2020).

Scibilia, F., Jørgensen, U. and Skjetne, R. (2012), AUV Guidance System for Dynamic trajectory Generation, Vol. 45, Elsevier Ltd., pp. 198–203.

Skjetne, R. (2005), The Maneuvering Problem, PhD thesis, Norwegian University of Science and Technology, Trondheim.

Skjetne, R. (2019), 'Technical note: Maneuvering control design of a low-speed fully-actuated vessel with stepwise path generation Revision D, NTNU'.

Skjetne, R. (2020*a*), 'Notes on: Guidance and maneuvering design for docking Revision A (Draft), NTNU'.

Skjetne, R. (2020*b*), 'Technical note: Maneuvering-based guidance design for dynamic positioning Revision B (Draft), NTNU'.

Skjetne, R., Fossen, T. I. and Kokotović, P. V. (2005), 'Adaptive Maneuvering, with Experiments, for a Model Ship in a Marine Control Laboratory', *Automatica* **41**, 289–298. doi: 10.1016/j.automatica.2004.10.006.

Sørensen, A. J. (2018), Marine Cybernetics, towards Autonomous Marine Operations and Systems - Lecture Notes, Technical Report UK-18-76, Department of Marine Technology, NTNU, Trondheim.

Tam, C. and Bucknall, R. (2010), 'Collision Risk Assessment for Ships', *Journal of Marine Science and Technology* **15**, 257–270. doi: 10.1007/s00773–010–0089–7.

Tam, C., Bucknall, R. and Greig, A. (2009), 'Review of Collision Avoidance and Path Planning Methods for Ships in Close Range Encounters', *The Journal of Navigation* **62**(3), 455–476. doi: 10.1017/S0373463308005134.

Valbuena, L. and Tanner, H. G. (2012), 'Hybrid Potential Field Based Control of Differential Drive Mobile Robots', *Journal of Intelligent & Robotic Systems* **68**, 307–322. doi: 10.1007/s10846–012–9685–6.

Wolf, M. T., Assad, C., Kuwata, Y., Howard, A., Aghazarian, H., Zhu, D., Lu, T., Trebi-Ollennu, A. and Huntsberger, T. (2010), '360-Degree Visual Detection and Target Tracking on an Autonomous Surface Vehicle', *Journal of Field Robotics* **27**(6), 819–833. doi: 10.1002/rob.20371.

Šeda, M. (2007), Roadmap Methods vs. Cell Decomposition in Robot Motion Planning, Corfu Island, Greece, pp. 127–132.

Appendix A

Videos of simulations

This chapter presents the videos that show simulations of the two guidance methods.

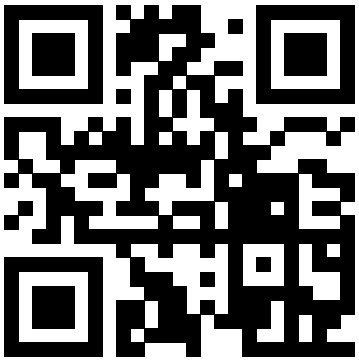
A.1 Guidance with navigation function

A.1.1 Head-on scenario

`https://vimeo.com/425868513`

A.1.2 Give-way scenario

`https://vimeo.com/425867977`



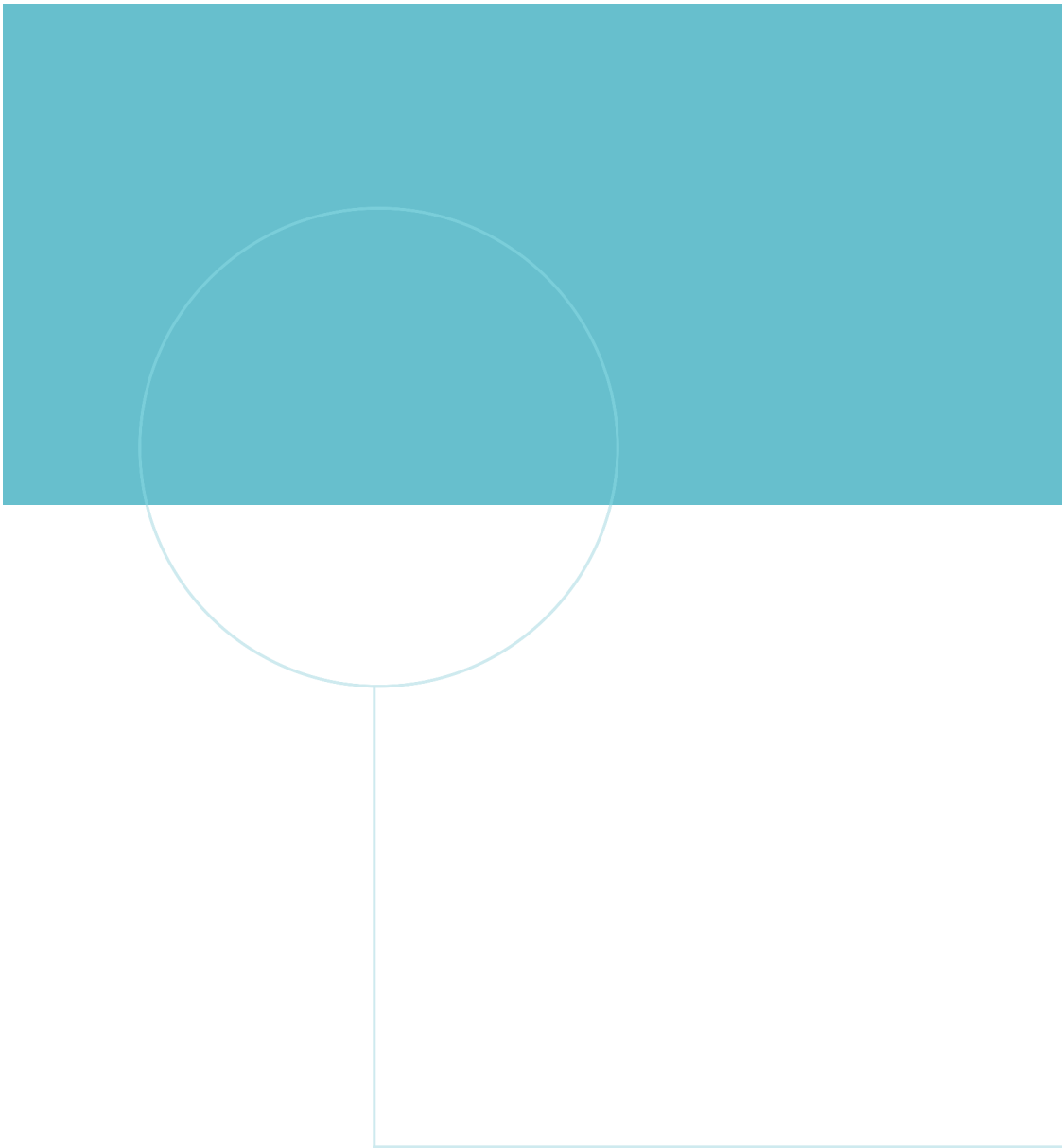
A.2 Guidance with two path parameters

A.2.1 Head-on scenario

https://vimeo.com/425930250

A.2.2 Give-way scenario

https://vimeo.com/425930374



NTNU

Norwegian University of
Science and Technology